

1. LabSWP_2019_SoSe_Geocaching	3
1.1 Dokumentation	5
1.1.1 Decision Log	6
1.1.2 Frontend Development	9
1.1.3 Technologie & Architektur	10
1.1.4 Öffnen und starten des Projekts auf dem localhost	12
1.1.5 Klassendiagramm	14
1.1.6 Geocaching Backend API	15
1.1.7 Installationsanleitung	17
1.1.8 Wichtige Abläufe	19
1.1.9 Leistungen pro Teammitglied	21
1.1.10 Datenbankstruktur, SQL-Anfragen, Trigger	44
1.1.11 Projektorganisation	46
1.1.11.1 Daily Scrums	47
1.1.11.1.1 2019-03-14 Daily Scrum 1	48
1.1.11.1.2 2019-03-18 Daily Scrum 2	49
1.1.11.1.3 2019-03-21 Daily Scrum 3	50
1.1.11.1.4 2019-03-25 Daily Scrum 4	52
1.1.11.1.5 2019-03-28 Daily Scrum 5	53
1.1.11.1.6 2019-04-01 Daily Scrum 6	54
1.1.11.1.7 2019-04-04 Daily Scrum 7	55
1.1.11.1.8 2019-04-05 Daily Scrum 8	56
1.1.11.1.9 2019-04-08 Daily Scrum 9	57
1.1.11.1.10 2019-04-09 Daily Scrum 10	58
1.1.11.1.11 2019-04-10 Daily Scrum 11	59
1.1.11.1.12 2019-04-11 Daily Scrum 12	60
1.1.11.1.13 2019-04-12 Daily Scrum 13	61
1.1.11.1.14 2019-04-15 Daily Scrum 14	62
1.1.11.1.15 2019-04-16 Daily Scrum 15	63
1.1.11.1.16 2019-04-17 Daily Scrum 16	64
1.1.11.1.17 2019-04-18 Daily Scrum 17	65
1.1.11.1.18 2019-04-23 Daily Scrum 18	66
1.1.11.1.19 2019-04-24 Daily Scrum 19	67
1.1.11.1.20 2019-04-25 Daily Scrum 20	68
1.1.11.1.21 2019-04-26 Daily Scrum 21	69
1.1.11.1.22 2019-04-29 Daily Scrum 22	70
1.1.11.1.23 2019-04-30 Daily Scrum 23	71
1.1.11.1.24 2019-05-02 Daily Scrum 24	72
1.1.11.1.25 2019-05-03 Daily Scrum 25	73
1.1.11.1.26 2019-05-06 Daily Scrum 26	74
1.1.11.1.27 2019-05-08 Daily Scrum 27	75
1.1.11.1.28 2019-05-09 Daily Scrum 28	76
1.1.11.1.29 2019-05-10 Daily Scrum 29	77
1.1.11.1.30 2019-05-14 Daily Scrum 30	78
1.1.11.1.31 2019-05-16 Daily Scrum 31	79
1.1.11.1.32 2019-05-17 Daily Scrum 32	80
1.1.11.1.33 2019-05-20 Daily Scrum 33	81
1.1.11.1.34 2019-05-21 Daily Scrum 34	82
1.1.11.1.35 2019-05-22 Daily Scrum 35	83
1.1.11.1.36 2019-05-23 Daily Scrum 36	84
1.1.11.1.37 2019-05-24 Daily Scrum 37	85
1.1.11.1.38 2019-05-27 Daily Scrum 38	86
1.1.11.1.39 2019-05-28 Daily Scrum 39	87
1.1.11.1.40 2019-05-29 Daily Scrum 40	88
1.1.11.1.41 2019-05-31 Daily Scrum 41	89
1.1.11.1.42 2019-06-03 Daily Scrum 42	90
1.1.11.1.43 2019-06-04 Daily Scrum 43	91
1.1.11.1.44 2019-06-05 Daily Scrum 44	92
1.1.11.1.45 2019-06-06 Daily Scrum 45	93
1.1.11.1.46 2019-06-17 Daily Scrum 46	94
1.1.11.1.47 2019-06-18 Daily Scrum 47	95
1.1.11.1.48 2019-06-19 Daily Scrum 48	96
1.1.11.1.49 2019-06-21 Daily Scrum 49	97
1.1.11.1.50 2019-06-24 Daily Scrum 50	98
1.1.11.1.51 2019-06-25 Daily Scrum 51	99
1.1.11.1.52 2019-06-26 Daily Scrum 52	100
1.1.11.2 Retrospectives	101
1.1.11.2.1 2019-03-28 Retrospektive	102
1.1.11.2.2 2019-04-11 Retrospektive	103
1.1.11.2.3 2019-04-25 Retrospektive	104
1.1.11.2.4 2019-05-09 Retrospektive	105
1.1.11.2.5 2019-05-23 Retrospektive	106
1.1.11.2.6 2019-06-06 Retrospektive	107
1.1.11.2.7 2019-06-25 Retrospektive	108
1.1.11.3 Sprint Reviews	109
1.1.12 Bedienungsanleitung	111
1.1.12.1 I. Für BuGa-Besuchende	112
1.1.12.1.1 1. Startseite aufrufen	113
1.1.12.1.2 2. Caches	114

1.1.12.1.3 3. TOP10-Rangliste	116
1.1.12.1.4 4. Konto anlegen	118
1.1.12.1.5 Die Navigationsleiste	119
1.1.12.2 II. Für Caching	121
1.1.12.2.1 1. Anmelden	122
1.1.12.2.2 2. Cache starten	123
1.1.12.2.3 3. Cache durchführen	124
1.1.12.2.4 4. Das Profil und Teams	126
1.1.12.2.5 Abmelden	128
1.1.12.2.6 Angefangene und beendete Caches	129
1.1.12.3 III. Für Administrierende	131
1.1.12.3.1 1. Cache erstellen	132
1.1.12.3.2 2. Station erstellen	134
1.1.12.3.3 3. Caches bearbeiten / löschen	136
1.1.12.3.4 4. Rangpunkte zurücksetzen	137
1.1.13 Anmeldedaten	138
1.1.14 Testberichte	139
1.1.15 Projektmanagement	142
1.1.15.1 Change Requests	143
1.1.15.2 Projektnormen und -spielregeln	144
1.1.15.3 Sprint Burndowns	145
1.1.16 Bug Statistik	148
1.1.17 Liste verbleibender Bugs und Tasks	151
1.2 Dateilisten	152
1.2.1 Besprechung mit anderen Teams	153
1.2.2 Usability Test	154
1.2.3 Präsentationen	160
1.2.4 Product Backlog	161
1.2.5 Projektauftrag	178
1.2.6 Releaseplan	183

LabSWP_2019_SoSe_Geocaching

So lange noch erreichbar, empfehlen wir ihnen die gesamte Dokumentation auf [Confluence](#) nachzulesen.

Über dieses Projekt

Unser Ziel ist es, die Bundesgartenschau 2019 in Heilbronn für junge, technikaffine Menschen attraktiver zu machen. Wir nutzen hierzu ein Zusammenspiel von neuen Technologien und Ideen und altbekanntem. Im Kern unserer Anwendung steht dabei die spielerische Führung über die Bundesgartenschau. Der Cacher kann hierzu eine von vielen vorgefertigten Routen über die BuGa auswählen. Jede dieser Routen unterteilt sich in eine Reihe von Stationen, die der Cacher nacheinander besuchen muss. Findet der Cacher eine Station erhält er einen Code, den er in der Web Anwendung eingeben kann und so Hinweise auf die nächste Station erhält. Die genaue Position muss der Cacher anhand dieser Hinweise, geschicktem Kombinieren und Wissen über die Bundesgartenschau, Heilbronn und Botanik herausfinden. Vervollständigt ein Cacher eine gesamte Route erhält er eine kleine Belohnung und Punkte für eine globale Rangliste. Je mehr Routen ein Nutzer abschließt, desto mehr Punkte erhält er und desto höher wird er in der Rangliste angezeigt. Dieses Feature verleiht dem ganzen noch eine gewisse Kompetivität.

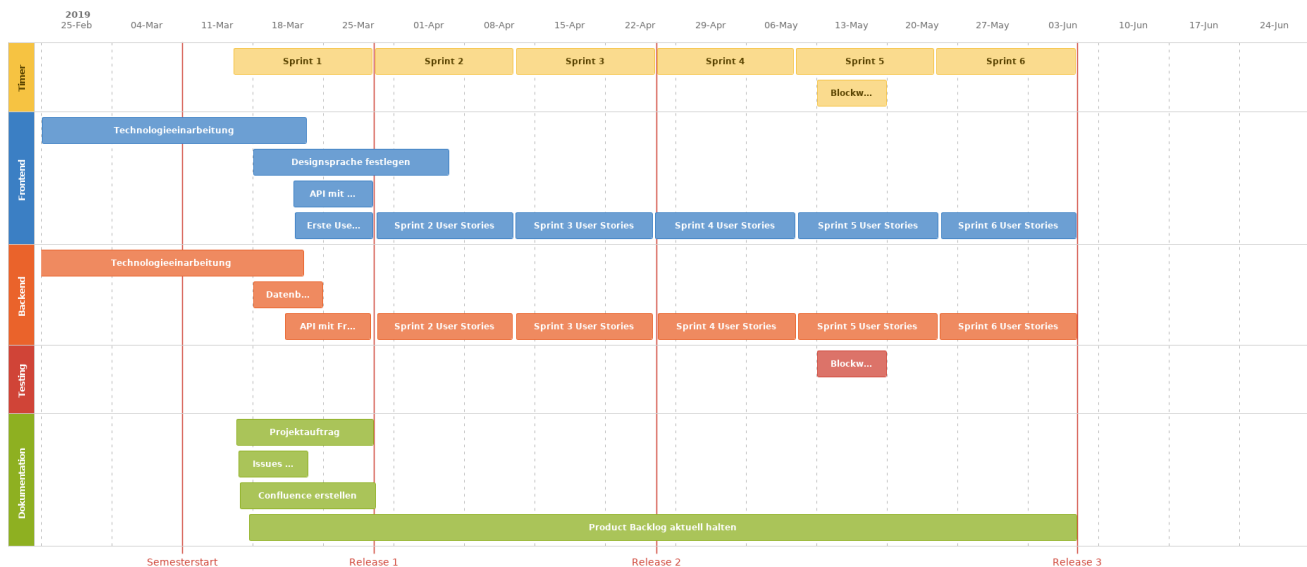
Das Team

Maximilian Leopold Product Owner	Katharina Will Scrum Master	Michael Dietz Developer
Robin Christ Developer	Timo Volkmann Developer	

Issues in Jira

Schlüssel S Erstellt Aktualisiert Fällig Bearbeiter Status Lösung

Keine Vorgänge gefunden



Kürzlich aktualisiert

[Liste verbleibender Bugs und Tasks](#)

vor einer Minute • erstellt von Katharina Will

[Sprint Reviews](#)

[vor etwa einer Stunde](#) • aktualisiert von Maximilian Leopold • [Änderung anzeigen](#)

[Sprint Burndowns](#)

[vor etwa 2 Stunden](#) • aktualisiert von Katharina Will • [Änderung anzeigen](#)

[Projektnormen und -spielregeln](#)

[vor etwa 2 Stunden](#) • aktualisiert von Robin Christ • [Änderung anzeigen](#)

[Angefangene und beendete Caches](#)

[vor etwa 2 Stunden](#) • aktualisiert von Michael Dietz • [Änderung anzeigen](#)

[Öffnen und starten des Projekts auf dem localhost](#)

[vor etwa 2 Stunden](#) • aktualisiert von Maximilian Leopold • [Änderung anzeigen](#)

[Wichtige Abläufe](#)

[vor etwa 2 Stunden](#) • aktualisiert von Maximilian Leopold • [Änderung anzeigen](#)

Terminkalender

Team Calendars Team-Kalender

Dokumentation

Decision Log

Angenommene Entscheidungen sind grün.

Verworfenne Entscheidungen sind **rot**.

Wie sind Rätsel der Stationen aufgebaut:

- Jede Station hat ein Rätsel, dessen Lösung zusammen mit dem Beschreibungstext den genauen Standort der nächsten Station verrät.
- Jede Station hat ein Rätsel. Wenn die Lösung dieses Rätsels in die Anwendung eingegeben wird, wird die nächste Station auf der Karte angezeigt.

API Aufrufe:

- Die Create (bsp. `/api/createCache`) Aufrufe werden mit einem `RequestBody` und einem Objekt (bsp. ein Cache Objekt) übergeben, nicht mit URL Parametern.
- `createCache` und `createStation` Methoden werden zusammengefasst.

Logging im Backend

Wir haben uns dazu entschlossen im Backend sehr ausführlich zu loggen, um für zukünftige Entwickler und Weiterführer des Projekts leicht Fehlerquellen ausmachen zu können. Wir nutzen hierzu die `slf4j` für Tomcat und schreiben per Logback in die Tomcat Logs. Falls Sie sich diese Logs ansehen möchten, können Sie dies jederzeit per Remote Directory unter der Adresse `\\seserver.se.hs-heilbronn.de\Tomcat 9 logs` im File `tomcat9-stdout.YYYY-MM-DD` nachschauen.

Beispiel des Loggings:

tomcat9-stdout.2019-06-21

```

2019-06-24 21:00:56.233 WARN 9052 --- [o-9443-exec-178] h.l.b.controller.Controller : API CALL:
/api/allCaches
2019-06-24 21:00:56.233 DEBUG 9052 --- [o-9443-exec-178] h.l.b.controller.Controller : /api
/allCaches PARAMETERS: -
2019-06-24 21:00:56.617 WARN 9052 --- [o-9443-exec-176] h.l.b.controller.Controller : API CALL:
/api/getMyCaches
2019-06-24 21:00:56.617 DEBUG 9052 --- [o-9443-exec-176] h.l.b.controller.Controller : /api
/getMyCaches: PARAMETERS:
token: eyJhbGciOiJSUzUxMiJ9.
eyJ1c2VyIjoieioiwicm9sZXNcIjpbZWlwcm9sZWVlOlwiQURNSU5cIixcImRvbWpblwiOlwiZ2VvY2FjaGluZy5kZVwifSx7XCXJyb2x1XCI6XCXJD
QUNIRVJcIixcImRvbWpblwiOlwiZ2VvY2FjaGluZy5kZVwifV0sXCJC1c2VySWRcIjoiMn0iLCJleHAiOiJlNjE0MTM2MDB9.
DosFb949utdnvebcla0gHdcHncl8rPkp0rEriisTnZvx8tc-6ZO25opp0fo_LbA3NxL6c6JxCpsrAJVBpHowY-
0SKWBGQvAnRfGwsRfQUnsB80QP5p0-5QBvtloqYlD9DTHb1zh15H8P-
mdyzJObiJ8clu1FGc5SEx4QJjhImlqIj9FSML76hvm_lR9uMhfkpAaI4kMZEeffCyhhAMryzEehhsmae6SXT__1Zs6PJ4-
Zv2cqCOvknX4uWXYlDdb_crNZT7YenVSC5wLbgrjBLMKfkUMLVfSTIPUp6T31PFRxW8dSC4ey4XO2q3UVmtSxRrnjgGFwWwEiMPRyWY28zMJavllY
HZPHFlea76g5rasRM_MW2NNop_JEX0vRdHJhtZlWq6ci3n05ncHmE9K0krOrqBkZMzt4Kuof3o_lJnR5sZZTcCl6IBehnf72_SyaTf8zvr08035
z1RXJiVkf5qOz40UuwVtUL2K3nhP4CxpqvE4SsDTL7mGh5eTQfRWhEdHGfnJS9zTej9S1N5LweBZOfrTpmjFQw09y47_zbRhLsI7BFD4RlTGe3T
lc0cURSArZVMYwV6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5cA5_maMPyTLoAA2fSk4ik6JCWpy0gZNDdG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:00:56.617 WARN 9052 --- [o-9443-exec-176] h.l.b.util.VerificationUtil : VERIFY
TOKEN: token: eyJhbGciOiJSUzUxMiJ9.
eyJ1c2VyIjoieioiwicm9sZXNcIjpbZWlwcm9sZWVlOlwiQURNSU5cIixcImRvbWpblwiOlwiZ2VvY2FjaGluZy5kZVwifSx7XCXJyb2x1XCI6XCXJD
QUNIRVJcIixcImRvbWpblwiOlwiZ2VvY2FjaGluZy5kZVwifV0sXCJC1c2VySWRcIjoiMn0iLCJleHAiOiJlNjE0MTM2MDB9.
DosFb949utdnvebcla0gHdcHncl8rPkp0rEriisTnZvx8tc-6ZO25opp0fo_LbA3NxL6c6JxCpsrAJVBpHowY-
0SKWBGQvAnRfGwsRfQUnsB80QP5p0-5QBvtloqYlD9DTHb1zh15H8P-
mdyzJObiJ8clu1FGc5SEx4QJjhImlqIj9FSML76hvm_lR9uMhfkpAaI4kMZEeffCyhhAMryzEehhsmae6SXT__1Zs6PJ4-
Zv2cqCOvknX4uWXYlDdb_crNZT7YenVSC5wLbgrjBLMKfkUMLVfSTIPUp6T31PFRxW8dSC4ey4XO2q3UVmtSxRrnjgGFwWwEiMPRyWY28zMJavllY
HZPHFlea76g5rasRM_MW2NNop_JEX0vRdHJhtZlWq6ci3n05ncHmE9K0krOrqBkZMzt4Kuof3o_lJnR5sZZTcCl6IBehnf72_SyaTf8zvr08035
z1RXJiVkf5qOz40UuwVtUL2K3nhP4CxpqvE4SsDTL7mGh5eTQfRWhEdHGfnJS9zTej9S1N5LweBZOfrTpmjFQw09y47_zbRhLsI7BFD4RlTGe3T
lc0cURSArZVMYwV6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5cA5_maMPyTLoAA2fSk4ik6JCWpy0gZNDdG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:00:56.619 DEBUG 9052 --- [o-9443-exec-176] h.l.b.util.VerificationUtil : VERIFY
TOKEN: Got Claims: {user:{roles:["ADMIN"], "domain": "geocaching.de"}, {"role": "CACHER", "domain": "
geocaching.de"}}, "userId": 52}, exp=1561413600}
2019-06-24 21:00:56.619 DEBUG 9052 --- [o-9443-exec-176] h.l.b.controller.Controller : /api
/getMyCaches Token verified
2019-06-24 21:00:56.623 DEBUG 9052 --- [o-9443-exec-176] h.l.b.controller.Controller : /api
/getMyCaches Got User: max.leopold2111@gmail.com
2019-06-24 21:00:56.687 DEBUG 9052 --- [o-9443-exec-176] h.l.b.controller.Controller : /api

```

```
getMyCaches Got all bearbeitet entreis of user: max.leopold211l@gmail.com
2019-06-24 21:01:22.802 WARN 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : API CALL: /api/getUser
2019-06-24 21:01:22.802 DEBUG 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : /api/getUser: PARAMETERS: token: eyJhbGciOiJSUzUxMiJ9.eyJ1c2VyIjoielwiczXNcIjpbZWlwcm9sZVwiOlwiQURNSU5cIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifSx7XCJyb2xlXCI6XCJDQUINIRVJcIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifV0sXCJlc2VySWRCIjolMn0iLCJleHAiOjE1NjEOMTM2MDB9.DosFb949utdnvebcla0gHdCHnlc8rpPkpOrEriisTnzvx8tc-6ZO25opp0fo_Lba3NxL6c6JxCpsrAJVBpHowY-OSKWBQGvAnRfGwsRfqUnsB8OQP5pO-5QBvtloqYlD9DTHblzh15H8P-mdyzJObij8clulFGc5SEEX4QJjhImIqIj9FSML76hvm_1R9uMhfkpAaI4kMZEEFFCyhhAMryzEehhmae6SXT__1Zs6PJ4-Zv2cqCOvknX4uWXYlDdb_crNZT7YeNVSC5wLbgrjBLMKfkUMlVFstIPUp6T31PFRxW8dSC4ey4XO2q3UVmtSxRrnjgGFwwEimPRywy28zMjavlllyHZPHFllea76g5rasRM_MW2NNop_JEXovRdhJhtZlWq6ci3n05ncHmE9KOkrOrgBkZMzt4Kuof3o_1JnR5sZZTcCl6IBehn72_SyaTF8zvr08035zlRXJiVkF5qOz40UuwVtUL2K3nhP4CxqpVe4SsDTL7mGH5eTQRWhEdHGfnJS9zTej9S1N5LweBZOfrTpmjFQw09y47_zbRhLSi7BFD4DRltGe3Tlc0CURSArZVMYVV6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5CA5_maMPyTL0AA2fSk4ik6JCWPpyOGZNDDG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:01:22.802 DEBUG 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : /api/getUser Trying to verify Token: eyJhbGciOiJSUzUxMiJ9.eyJ1c2VyIjoielwiczXNcIjpbZWlwcm9sZVwiOlwiQURNSU5cIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifSx7XCJyb2xlXCI6XCJDQUINIRVJcIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifV0sXCJlc2VySWRCIjolMn0iLCJleHAiOjE1NjEOMTM2MDB9.DosFb949utdnvebcla0gHdCHnlc8rpPkpOrEriisTnzvx8tc-6ZO25opp0fo_Lba3NxL6c6JxCpsrAJVBpHowY-OSKWBQGvAnRfGwsRfqUnsB8OQP5pO-5QBvtloqYlD9DTHblzh15H8P-mdyzJObij8clulFGc5SEEX4QJjhImIqIj9FSML76hvm_1R9uMhfkpAaI4kMZEEFFCyhhAMryzEehhmae6SXT__1Zs6PJ4-Zv2cqCOvknX4uWXYlDdb_crNZT7YeNVSC5wLbgrjBLMKfkUMlVFstIPUp6T31PFRxW8dSC4ey4XO2q3UVmtSxRrnjgGFwwEimPRywy28zMjavlllyHZPHFllea76g5rasRM_MW2NNop_JEXovRdhJhtZlWq6ci3n05ncHmE9KOkrOrgBkZMzt4Kuof3o_1JnR5sZZTcCl6IBehn72_SyaTF8zvr08035zlRXJiVkF5qOz40UuwVtUL2K3nhP4CxqpVe4SsDTL7mGH5eTQRWhEdHGfnJS9zTej9S1N5LweBZOfrTpmjFQw09y47_zbRhLSi7BFD4DRltGe3Tlc0CURSArZVMYVV6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5CA5_maMPyTL0AA2fSk4ik6JCWPpyOGZNDDG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:01:22.802 WARN 9052 --- [o-9443-exec-179] h.l.b.util.VerificationUtil : VERIFY TOKEN: token: eyJhbGciOiJSUzUxMiJ9.eyJ1c2VyIjoielwiczXNcIjpbZWlwcm9sZVwiOlwiQURNSU5cIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifSx7XCJyb2xlXCI6XCJDQUINIRVJcIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifV0sXCJlc2VySWRCIjolMn0iLCJleHAiOjE1NjEOMTM2MDB9.DosFb949utdnvebcla0gHdCHnlc8rpPkpOrEriisTnzvx8tc-6ZO25opp0fo_Lba3NxL6c6JxCpsrAJVBpHowY-OSKWBQGvAnRfGwsRfqUnsB8OQP5pO-5QBvtloqYlD9DTHblzh15H8P-mdyzJObij8clulFGc5SEEX4QJjhImIqIj9FSML76hvm_1R9uMhfkpAaI4kMZEEFFCyhhAMryzEehhmae6SXT__1Zs6PJ4-Zv2cqCOvknX4uWXYlDdb_crNZT7YeNVSC5wLbgrjBLMKfkUMlVFstIPUp6T31PFRxW8dSC4ey4XO2q3UVmtSxRrnjgGFwwEimPRywy28zMjavlllyHZPHFllea76g5rasRM_MW2NNop_JEXovRdhJhtZlWq6ci3n05ncHmE9KOkrOrgBkZMzt4Kuof3o_1JnR5sZZTcCl6IBehn72_SyaTF8zvr08035zlRXJiVkF5qOz40UuwVtUL2K3nhP4CxqpVe4SsDTL7mGH5eTQRWhEdHGfnJS9zTej9S1N5LweBZOfrTpmjFQw09y47_zbRhLSi7BFD4DRltGe3Tlc0CURSArZVMYVV6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5CA5_maMPyTL0AA2fSk4ik6JCWPpyOGZNDDG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:01:22.805 DEBUG 9052 --- [o-9443-exec-179] h.l.b.util.VerificationUtil : VERIFY TOKEN: Got Claims: {user:{roles:[{role:"ADMIN","domain":"geocaching.de"},{role:"CACHER","domain":"geocaching.de"}],userId:52},exp=1561413600}
2019-06-24 21:01:22.805 DEBUG 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : /api/getUser Trying to get User
2019-06-24 21:01:22.805 DEBUG 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : /api/getUser Got Claim: { "user": "{ \"roles\": [{ \"role\": \"ADMIN\", \"domain\": \"geocaching.de\" }, { \"role\": \"CACHER\", \"domain\": \"geocaching.de\" } ], \"userId\": 52 }\", \"exp\": 1561413600 }
2019-06-24 21:01:22.813 DEBUG 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : /api/getUser Got User: max.leopold211l@gmail.com
2019-06-24 21:01:22.813 DEBUG 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : /api/getUser User != null max.leopold211l@gmail.com
2019-06-24 21:01:22.813 DEBUG 9052 --- [o-9443-exec-179] h.l.b.controller.Controller : /api/getUser Send User to Frontend
2019-06-24 21:01:22.862 WARN 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : API CALL: /api/getUser
2019-06-24 21:01:22.862 DEBUG 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : /api/getUser: PARAMETERS: token: eyJhbGciOiJSUzUxMiJ9.eyJ1c2VyIjoielwiczXNcIjpbZWlwcm9sZVwiOlwiQURNSU5cIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifSx7XCJyb2xlXCI6XCJDQUINIRVJcIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifV0sXCJlc2VySWRCIjolMn0iLCJleHAiOjE1NjEOMTM2MDB9.DosFb949utdnvebcla0gHdCHnlc8rpPkpOrEriisTnzvx8tc-6ZO25opp0fo_Lba3NxL6c6JxCpsrAJVBpHowY-OSKWBQGvAnRfGwsRfqUnsB8OQP5pO-5QBvtloqYlD9DTHblzh15H8P-mdyzJObij8clulFGc5SEEX4QJjhImIqIj9FSML76hvm_1R9uMhfkpAaI4kMZEEFFCyhhAMryzEehhmae6SXT__1Zs6PJ4-Zv2cqCOvknX4uWXYlDdb_crNZT7YeNVSC5wLbgrjBLMKfkUMlVFstIPUp6T31PFRxW8dSC4ey4XO2q3UVmtSxRrnjgGFwwEimPRywy28zMjavlllyHZPHFllea76g5rasRM_MW2NNop_JEXovRdhJhtZlWq6ci3n05ncHmE9KOkrOrgBkZMzt4Kuof3o_1JnR5sZZTcCl6IBehn72_SyaTF8zvr08035zlRXJiVkF5qOz40UuwVtUL2K3nhP4CxqpVe4SsDTL7mGH5eTQRWhEdHGfnJS9zTej9S1N5LweBZOfrTpmjFQw09y47_zbRhLSi7BFD4DRltGe3Tlc0CURSArZVMYVV6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5CA5_maMPyTL0AA2fSk4ik6JCWPpyOGZNDDG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:01:22.862 DEBUG 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : /api/getUser Trying to verify Token: eyJhbGciOiJSUzUxMiJ9.eyJ1c2VyIjoielwiczXNcIjpbZWlwcm9sZVwiOlwiQURNSU5cIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifSx7XCJyb2xlXCI6XCJDQUINIRVJcIixcImRvbWFPblwiOlwiz2VvY2FjaGluz5kzZVwifV0sXCJlc2VySWRCIjolMn0iLCJleHAiOjE1NjEOMTM2MDB9.DosFb949utdnvebcla0gHdCHnlc8rpPkpOrEriisTnzvx8tc-6ZO25opp0fo_Lba3NxL6c6JxCpsrAJVBpHowY-OSKWBQGvAnRfGwsRfqUnsB8OQP5pO-5QBvtloqYlD9DTHblzh15H8P-mdyzJObij8clulFGc5SEEX4QJjhImIqIj9FSML76hvm_1R9uMhfkpAaI4kMZEEFFCyhhAMryzEehhmae6SXT__1Zs6PJ4-Zv2cqCOvknX4uWXYlDdb_crNZT7YeNVSC5wLbgrjBLMKfkUMlVFstIPUp6T31PFRxW8dSC4ey4XO2q3UVmtSxRrnjgGFwwEimPRywy28zMjavlllyHZPHFllea76g5rasRM_MW2NNop_JEXovRdhJhtZlWq6ci3n05ncHmE9KOkrOrgBkZMzt4Kuof3o_1JnR5sZZTcCl6IBehn72_SyaTF8zvr08035zlRXJiVkF5qOz40UuwVtUL2K3nhP4CxqpVe4SsDTL7mGH5eTQRWhEdHGfnJS9zTej9S1N5LweBZOfrTpmjFQw09y47_zbRhLSi7BFD4DRltGe3Tlc0CURSArZVMYVV6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5CA5_maMPyTL0AA2fSk4ik6JCWPpyOGZNDDG4CUWjygbP8VeAlR49xrjHfxKTXM
```

```
QUNIRVJcIixcImRvbWfPbblwiOlwiz2VvY2FjagLUZy5kZVwifV0sXCJlc2VySWRCIjoIMn0iLCJleHAiOjE1NjE0MTM2M2DB9.
DosFb949utdnvebcla0gHDcHn1c8rPpKp0rEriisTnZvx8tc-6Z025opp0fo_LbA3NxL6c6JxCpsrAJVBpHowY-
0SKwBGQvAnRfGwsRfGUnsB80QP5p0-5QbvtloqY1D9DTHblzhl5H8P-
mdyzJObij8clulFGc5SEX4QJjhIm1qIj9FSML76hvm_1R9uMhfkpAAi4kMZEeffCyhhAMryzEehhsmae6SXT__1Zs6PJ4-
Zv2cqCovknX4uWXY1Ddb_crNZT7YeNVSC5wLbgrjBLMKfkUM1VfSTIPUp6T31PFRxW8dSC4ey4X02q3UVmtSxRrnjgGFWwEiMPRywY28zMJav1ly
HZPHFlea76g5rasRM_MW2NNop_JEX0vRdhJhtZlWq6ci3n05ncHmE9K0krOrqBkZMzjt4Kuof3o_1JnR5sZZTc16IBehnf72_SyaTf8zvr08035
z1RXJiVkf5qOz40UwuVtUL2K3nhP4CxpqvE4SsDTL7mGh5eTQfRwEdHGfnJS9zTej9S1N5LweBZOFRtPmjfQw09y47_zbRhLsI7BFD4DRltGe3T
1c0cURSArZVMYVW6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5cA5_maMPyTLoAA2fSk4ik6JCWpy0gZNDdG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:01:22.862 WARN 9052 --- [o-9443-exec-173] h.l.b.util.VerificationUtil : VERIFY
TOKEN: token: eyJhbGciOiJSUzUxMiJ9.
eyJ1c2VyIjoielwlcw9sZXNcIjpbZWwifV0sXCJlc2VySWRCIjoIMn0iLCJleHAiOjE1NjE0MTM2M2DB9.
QUNIRVJcIixcImRvbWfPbblwiOlwiz2VvY2FjagLUZy5kZVwifV0sXCJlc2VySWRCIjoIMn0iLCJleHAiOjE1NjE0MTM2M2DB9.
DosFb949utdnvebcla0gHDcHn1c8rPpKp0rEriisTnZvx8tc-6Z025opp0fo_LbA3NxL6c6JxCpsrAJVBpHowY-
0SKwBGQvAnRfGwsRfGUnsB80QP5p0-5QbvtloqY1D9DTHblzhl5H8P-
mdyzJObij8clulFGc5SEX4QJjhIm1qIj9FSML76hvm_1R9uMhfkpAAi4kMZEeffCyhhAMryzEehhsmae6SXT__1Zs6PJ4-
Zv2cqCovknX4uWXY1Ddb_crNZT7YeNVSC5wLbgrjBLMKfkUM1VfSTIPUp6T31PFRxW8dSC4ey4X02q3UVmtSxRrnjgGFWwEiMPRywY28zMJav1ly
HZPHFlea76g5rasRM_MW2NNop_JEX0vRdhJhtZlWq6ci3n05ncHmE9K0krOrqBkZMzjt4Kuof3o_1JnR5sZZTc16IBehnf72_SyaTf8zvr08035
z1RXJiVkf5qOz40UwuVtUL2K3nhP4CxpqvE4SsDTL7mGh5eTQfRwEdHGfnJS9zTej9S1N5LweBZOFRtPmjfQw09y47_zbRhLsI7BFD4DRltGe3T
1c0cURSArZVMYVW6Y7h8CprDzsXmk9v_4APi7gehQjne76ynH5cA5_maMPyTLoAA2fSk4ik6JCWpy0gZNDdG4CUWjygbP8VeAlR49xrjHfxKTXM
2019-06-24 21:01:22.864 DEBUG 9052 --- [o-9443-exec-173] h.l.b.util.VerificationUtil : VERIFY
TOKEN: Got Claims: {user={\"roles\":[{\"role\":\"ADMIN\",\"domain\":\"geocaching.de\"},{\"role\":\"CACHER\",\"domain\":\"geocaching.de\"}],\"userId\":52}, exp=1561413600}
2019-06-24 21:01:22.864 DEBUG 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : /api
/getUser Trying to get User
2019-06-24 21:01:22.865 DEBUG 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : /api
/getUser Got Claim: {
  \"user\": \"{\\\"roles\\\":[{\\\"role\\\":\\\"ADMIN\\\",\\\"domain\\\":\\\"geocaching.de\\\"},{\\\"role\\\":\\\"CACHER\\\",\\\"domain\\\":\\\"geocaching.de\\\"}],\\\"userId\\\":52}\",
  \"exp\": 1561413600
}
2019-06-24 21:01:22.876 DEBUG 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : /api
/getUser Got User: max.leopold2111@gmail.com
2019-06-24 21:01:22.876 DEBUG 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : /api
/getUser User != null max.leopold2111@gmail.com
2019-06-24 21:01:22.876 DEBUG 9052 --- [o-9443-exec-173] h.l.b.controller.Controller : /api
/getUser Send User to Frontend
```

Change Requests

Betreff:

Jeder Benutzer sollte ursprünglich auch ohne Anmeldung cachen dürfen.

Grund:

Wir haben uns im Laufe des Projekts dazu entschieden, Benutzer nicht unangemeldet Cachen zu lassen, da dies folgende Probleme verursacht:

- Benutzer sind nicht eindeutig identifizierbar für den Server.
- Benutzer kann unterbrochene Caches nicht fortsetzen.
- Benutzer hat keine Kontrolle über seine Daten (wenn serverseitig gespeichert).
- Benutzer darf aus Missbrauchspräventionsgründen nicht auf Belohnungen zugreifen kein Anreiz für Nutzer mehr.
- Wenn Daten nur im Client liegen, wären diese manipulierbar und nicht aktuell.
- Nutzen/Aufwand Verhältnis zu groß:
Erfordert einiges an zusätzlicher Anwendungslogik für angemeldete und unangemeldete Benutzer Nicht mehr im Zeitrahmen umsetzbar
Usermanagement ist externes System und bietet für diese Funktion gebrauchte Features nicht (temporäre und automatisch generierte Nutzer).

Alternative Lösung:

Benutzer können sich jedoch unangemeldet einen Überblick über alle Caches verschaffen, die Beschreibungen ansehen, Caches auf der Karte ansehen und die Rangliste aufrufen. Wenn dann interesse besteht, ist die Anmeldung schnell vollzogen und der Benutzer kann den vollen Funktionsumfang der Anwendung nutzen.

Frontend Development

Für die Entwicklung mit Vue.js und Quasar sind einige Vorbereitungen zu treffen bevor die Web GUI aufrufbar ist. Frontend-Entwicklern wird empfohlen NPM oder Yarn (beides Paketverwaltungen für Javascript Anwendungen) global installiert zu haben. Wenn nur mit Spring gearbeitet wird, kann darauf verzichtet werden, da Gradle alle Aufgaben bezüglich NPM übernimmt, inklusive Download und lokaler Installation innerhalb des Projekts.

Quasar (Vue.js) Dev Server:

Wenn nur im Frontend gearbeitet wird und für die Arbeit keine Anbindung zum Backend notwendig ist, kann mit einem Node.js & Webpack gestützten Development Server gearbeitet werden. Der Vorteil hierbei ist, dass Änderungen am Code direkt erkannt und mittels HMR (Hot Module Reload) direkt im Browser sichtbar sind. Der Server muss daurch nicht bei jeder kleinen Änderung neu gestartet werden, sondern übernimmt Änderungen im laufenden Betrieb.

Genauere Informationen zum Ausführen des Projekts auf dem localhost erhalten sie [hier](#).

1. nach **<projekt>/frontend** wechseln
2. **quasar dev** oder **npm run dev** ausführen

Build:

Um den aktuellen Stand ins Hauptpaket zu bauen, muss lediglich ein Gradle Build ausgeführt werden.

1. Im Projekt Root Verzeichnis **./gradlew build** oder **gradle build** ausführen

Conventions & Best Practice

Message Dialog:

Um eine Fehlernachricht von jeder Seite aus aufrufen zu können, steht nun eine Methode bereit um ein Popup zu öffnen. Die Methode nimmt ein Objekt für Titel, Nachricht und Farbe entgegen:

```
this.$store.commit('dialog/NEW_MESSAGE_DIALOG', { title: "<Titel>", message: "<Text der Nachricht>", color: "<vordefinierte Farbe>" })
```

Momentan sind mögliche Farben: "red", "amber", "blue".

Die anderen Eigenschaften haben Standardwerte, sodass die Übergabe des Objekts (oder einzelner Properties des Objekts) an die Methode nicht zwingend notwendig ist.

Defaultwerte:

title: "Fehler"
message: "Ein unbekannter Fehler ist aufgetreten. Bitte versuchen Sie es noch einmal."
color: "red"

Technologie & Architektur

Liste der Abhängigkeiten

Clientseitig

Paket	Version	Funktion
Quasar Framework	v1.0.0-rc.2	Frontend-Framework mit Material-Design: ermöglicht schnelle Erstellung von Apps für verschiedene Plattformen mit einer einzigen Codebasis
Vue.js	2.6.10	JS-Framework für Webapps
Vuex	3.1.1	Zentrale Datenhaltung für Vue-Apps
Axios	0.18.0	Bietet einfache Schnittstelle für HTTP-Aufrufe
VueLayers (OpenLayers)	0.11.4 (4.6.5)	Vue-Wrapper für Openlayers (Client-Library für Open Street Maps)
vue-qrcode-reader (jsqr)	1.4.2 (1.2.0)	Vue-Wrapper für jsqr (JS-Library für die Verarbeitung und Auslesen von QR-Codes)

Serverseitig

Paket	Version	Funktion
Spring Boot	2.1.6	Backend Framework für Java Restful Webservices
Gson	2.8.5	JSON-Parser für Java Objekte
JWT	0.10.5	JSON Web Token für Authentifizierung von Usern
Springfox-Swagger	2.9.2 (2.0.2)	Automatische JSON API Dokumentation für Spring
SLF4J	1.7.2	Loggerfacade für Java Anwendungen
Logback	1.0.9	Logger für Spring Boot

Architektur

Clientseitig wurde von uns das Quasar Framework ausgewählt, welches auf Vue.js basiert und uns ermöglicht, durch fertige Komponenten schnell ein gutes Layout aufzusetzen und alle benötigten Pakete und Dienste einzubinden. Auch bezüglich Zielpattform sind wir später noch flexibel und können beispielsweise bei Bedarf auch Smartphone-Apps und Programme für PC/Mac aus der selben Codebase generieren.

Für die Verbindung zu den API-Endpunkten unseres Servers und den Endpunkten des Dienstes zur QR-Code-Generierung haben wir Axios verwendet, welches eine vereinfachte Schnittstelle für HTTP-Requests bietet.

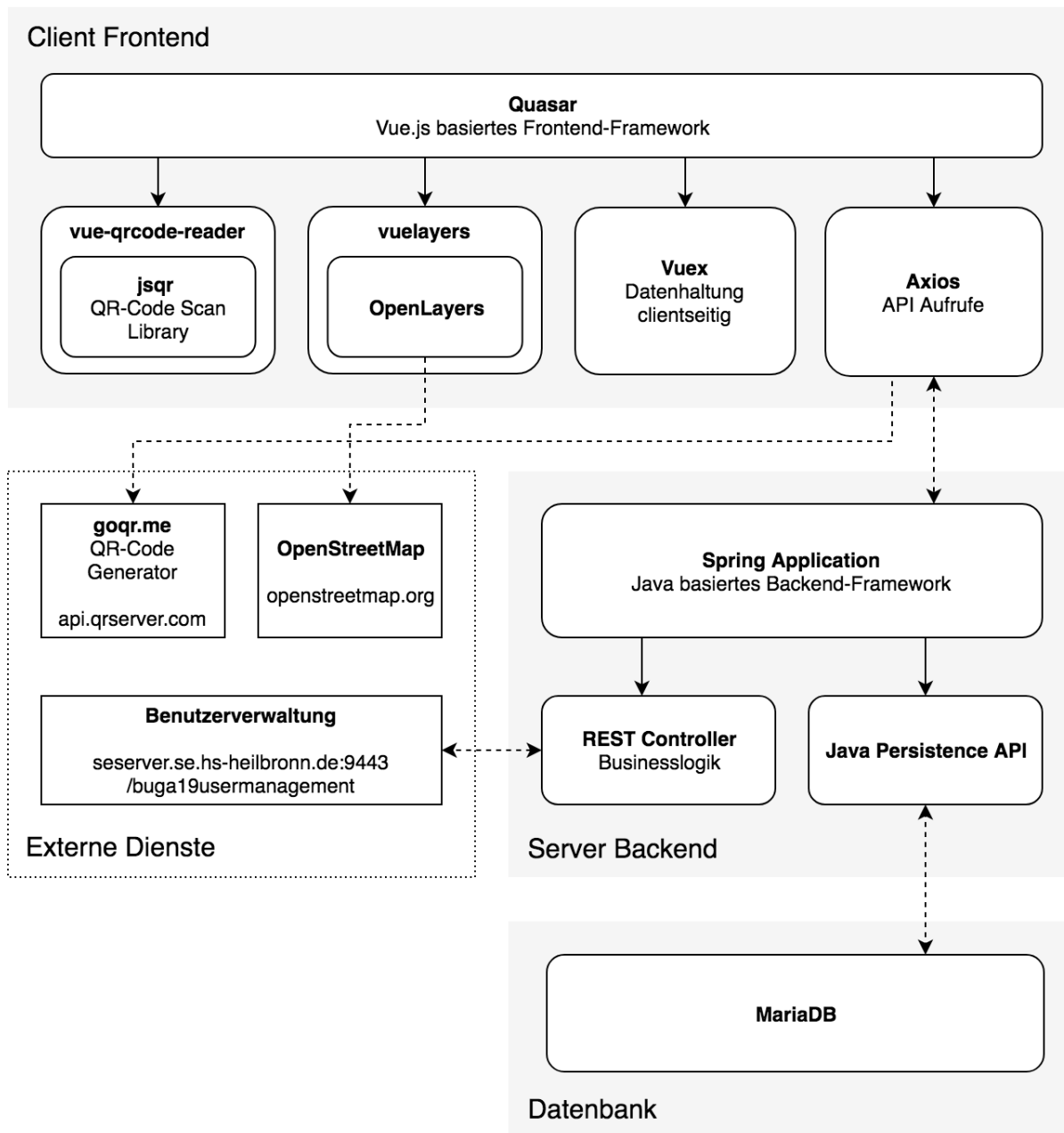
Für OpenStreetMaps gibt es mit vuelayers fertige Wrapper-Komponenten für Vue, welche die Benutzung von OSM mithilfe der Client-Bibliothek Openlayers vereinfachen.

Den QR-Code-Scanner binden wir über vue-qrcode-reader ein. Dieses Paket greift seinerseits auf jsqr (JS-Library für die Verarbeitung und Auslesen von QR-Codes) zurück, welches die Algorithmen zur Bildverarbeitung und Auswertung implementiert und die Browserschnittstelle zur Verwendung der Kamera nutzt. Aufgrund von sicherheitstechnischen Einschränkungen des Browsers ist der Zugriff auf die Kamera des Endnutzers nur mit einer verschlüsselten Verbindung über HTTPS möglich.

Vuex wird von uns benutzt, um Daten clientseitig in der gesamten Webapp zugänglich zu machen und diese an einem zentralen Ort aufzubewahren. Um Daten wie beispielsweise den Benutzer-Token auf der Clientseite persistent zu halten, benutzen wir die von modernen Browsern bereitgestellte LocalStorage-Schnittstelle.

Im Backend nutzen wir Java Spring Boot als Java basiertes Backend Framework. Unsere Spring Application definiert hierbei einen Rest Controller, der für jede URL eine gemappte Methode besitzt. Der Restcontroller besitzt die Backend Logik, nimmt Anfragen entgegen, verarbeitet diese und sendet das Ergebnis ans Frontend zurück. Der Rest Controller ist ebenfalls für die Kommunikation mit der Benutzerverwaltung zuständig. Beim starten der Anwendung holt sich der Controller einmalig den Public Key des Usermanagements um damit User eigenständig verifizieren zu können.

Als Schnittstelle zu unserer MariaDB Datenbank nutzen wir die Java Persistence API (JPA). JPA erlaubt es uns sogenannte Repositories zu definieren, mit denen wir direkten Zugriff auf Datenbanken haben. In diesen Repositories können SQL Anfragen definiert werden, um auch komplexere Anfragen an die Datenbank stellen zu können.



Öffnen und starten des Projekts auf dem localhost

Wir empfehlen Ihnen dringend als IDE JetBrains IntelliJ IDEA zu nutzen!

Da alle Mitglieder unseres Team IntelliJ als ihre vorrangige IDE verwenden, können wir die Kompatibilität zu anderen IDE, wie beispielsweise Eclipse oder NetBeans, nicht sicherstellen. Dies liegt an den IDE spezifischen Projektfiles. Im Projekt ist lediglich der .idea Ordner als IntelliJ Konfiguration vorhanden. Eine .project Datei wie sie von Eclipse benutzt wird ist nicht vorhanden.

Falls Sie IntelliJ nicht besitzen, können Sie es sich unter diesem Link herunterladen: <https://www.jetbrains.com/idea/>

Sie finden das Projekt unter "Projekt" auf dem USB-Stick.

Öffnen des Projekt in IntelliJ IDEA:

Falls Sie IntelliJ das erste mal öffnen, sehen Sie Willkommensfenster. Hier klicken Sie bitte auf **Open** und navigieren zum Speicherort des Projekts und drücken anschließend auf **Öffnen**.

Falls Sie IntelliJ bereits benutzt haben, sollte sich das zuletzt von Ihnen bearbeitete Projekt öffnen. Um ein neues Projekt zu öffnen klicken Sie auf **File Open**. Navigieren Sie anschließend zum Speicherort des Projekts und drücken Sie auf **Öffnen**.

Wenn Sie das Projekt zum ersten mal öffnen, sollten Sie alle benötigten Dependencies einmalig laden. Der einfachste Weg um dies zu tun ist über den eingebauten Gradle Wrapper. Um diesen zu nutzen öffnen Sie das interne Terminal in IntelliJ (zu finden ganz unten in der Mitte. Button **Terminal**. siehe Abb. 1). Wenn sich das Terminal geöffnet hat, vergewissern Sie sich, dass Sie im richtigen Verzeichnis sind (Terminal sollte ähnlich wie Abb. 2 aussehen). Falls dies nicht der Fall sein sollte, navigieren Sie bitte mit dem **cd <verzeichnis>** Befehl in das Projektverzeichnis.



Abb. 1

```
max (develop *) labswp_2019_sose_geocaching
$
```

Abb. 2

Wenn Sie im richtige Verzeichnis angekommen sind, führen Sie den Befehl **gradlew clean build** (**./gradlew clean build** unter **Mac OS X**) aus. IntelliJ lädt dadurch alle erforderlichen Dependencies herunter und baut das Projekt einmal zusammen um zu sehen ob es Compilefehler oder Dependencyfehler gibt.

Um die WebApp anschließend ansehen zu können, müssen Sie sowohl das Backend als auch das Frontend starten. Im folgenden wird dies erklärt.

Starten des Backends:

Wurde das Projekt einmal komplett zusammengebaut, steht der Ausführung des Backends nichts mehr im Weg. Im **Ordner src/main/java, package hhn. labsw.bugageocaching** finden Sie die Klasse **Application.java**. Dies ist die Klasse, die die Spring Anwendung und damit das Backend startet. Bevor Sie jedoch auf 'run' drücken vergewissern Sie sich, dass Sie entweder mit dem Hochschulnetz oder mit der Hochschule per VPN verbunden sind und der Port 8080 auf Ihrem Rechner nicht belegt ist. Wenn Sie mit der Hochschule verbunden sind starten Sie das Backend durch einen **Rechtsklick** auf die **Application.java** Klasse und drücken in dem sich öffnenden Popup auf **Run 'Application'** (siehe Abb. 3).

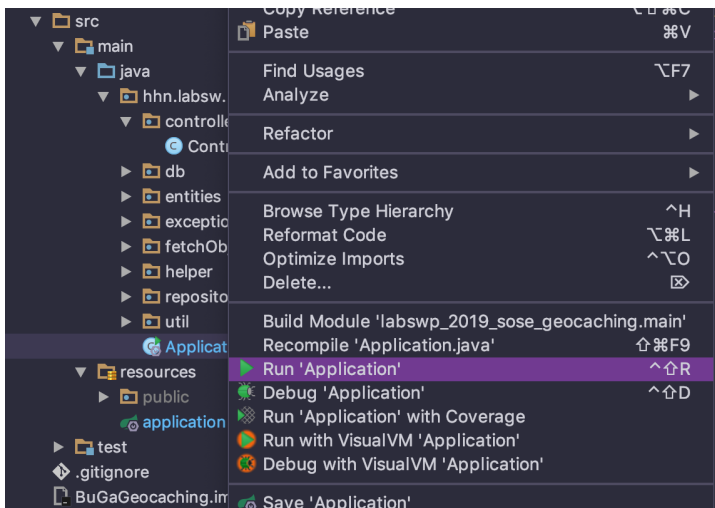


Abb. 3

Starten des Frontends:

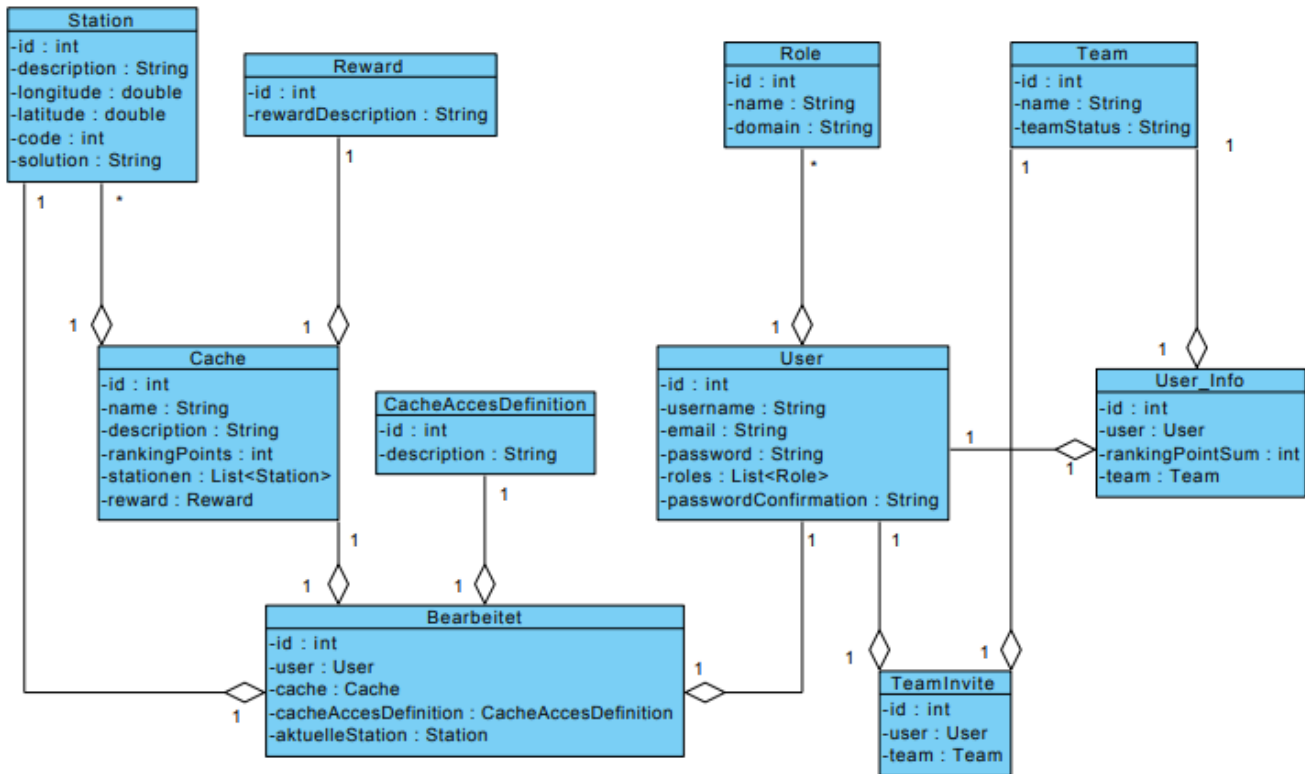
Zum starten des Frontends empfehlen wir das benutzen des **Quasar Dev Servers**. Wir haben die Erfahrung gemacht, dass falls das Frontend ebenfalls durch Spring gestartet wird, Änderungen nicht zum Tragen kommen, da Spring (bzw. IntelliJ) die alten Daten cached und diese auch bei Änderungen wiederverwendet.

Da unser Frontend auf Quasar beruht, empfiehlt sich die Benutzung des Quasar Dev Servers.

Um diesen nutzen zu öffnen Sie erneut das Terminal. Wenn Sie das Backend bereits gestartet haben, benutzen Sie den **+** Button am oberen Rand des Terminals um ein neues Terminal zu erzeugen. Falls nicht schon automatisch geschehen, navigieren Sie wieder in das Projektverzeichnis, sodass ihr Terminal ähnlich wie **Abb. 2** aussieht. Nutzen Sie nun die zwei Befehle **cd labswp_2019_sose_geocaching_frontend/** und **cd frontend/** um ins Frontendverzeichnis zu gelangen. Um den Quasar Dev Server ausführen zu können benötigen Sie noch eine weitere Dependency. Diese können Sie über das Terminal mit dem Befehl **npm install -g quasar-cli** erhalten. Wenn die Dependency erfolgreich installiert wurde, nutzen Sie den Befehl **quasar dev** um des Quasar Dev Server zu starten.

Falls Sie sowohl Backend als auch Frontend erfolgreich gestartet haben, können Sie die WebApp unter der Adresse <http://localhost:8081/> erreichen.

Klassendiagramm



Erklärung zum Klassendiagramm:

Klassen sind bei uns durch Kompositionen miteinander verknüpft. Das liegt daran, da es für uns ausreicht, wenn beispielsweise der Cache weiß welche Stationen zu ihm gehören. Die Station selbst muss nicht wissen zu welchem Cache sie gehört. So sparen wir uns unnötige Spalten in den Datenbanktabellen und damit auch Speicherplatz.

Im Kern unseres Klassendiagramms stehen die Klassen User und Cache.

Der User kann mehrere Rollen einnehmen (Cacher, Administrator). Durch die User_Info, welche aus einem User und einem Team besteht, werden weitere Informationen des Users gespeichert, welche sich in der gemeinsamen Userdatenbank nicht speichern lassen können. Neben den oben sichtbaren Attributen der User_Info lassen sich auch mit Hilfe von SQL-Anfragen die Solo- und Teamrangliste bestimmen. Wir entschieden uns dagegen die Rangliste in der Datenbank zu speichern, da wir ansonsten jedesmal wenn jemand einen Cache abschließt (Punkte bekommt) diese Ranglistentabelle neu berechnen und neu speichern müssen. Mit Hilfe der SQL-Anfragen müssen wir die Rangliste wirklich nur dann berechnen, wenn jemand die Rangliste sehen will.

Auf der anderen Seite besteht der Cache aus mehreren Stationen und einem Reward. Mit Hilfe der Bearbeitet-Klasse, welche aus einer Station, einem Cache, einer cacheAccessDefinition und einem User besteht können wir den Fortschritt eines Users verfolgen. So können wir beispielsweise sehen, welcher User welchen Cache angefangen/beendet hat und welche Station er bei diesem Cache zuletzt abgeschlossen hat.

Geocaching Backend API

Für eine übersichtliche Darstellung der Backend API, [starten Sie das Projekt bitte auf dem localhost](http://localhost:8080/swagger-ui.html) und besuchen Sie die Adresse <http://localhost:8080/swagger-ui.html> um die API als HTML Seite ansehen zu können. Es ist hier auch möglich Methodenaufrufe manuell zu testen, indem man die benötigten Parameter in die Textfelder des jeweiligen Aufrufs einträgt und auf 'Try out' drückt.

Beispiel des Aufrufs der Methode '/api/allCaches'

GET /api/allCaches Retrieves all Caches, including their Stations, from the Database

Response Class (Status 200)
Model | Model Schema

```
{
  "code": 0,
  "description": "string",
  "id": 0,
  "latitude": 0,
  "longitude": 0,
  "solution": "string"
}
```

Response Content Type application/json

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out! Hide Response

Curl

```
curl -X GET --header "Accept: application/json" "http://localhost:8080/api/allCaches"
```

Request URL

`http://localhost:8080/api/allCaches`

Response Body

```
[
  {
    "id": 8,
    "name": "Wolfszipfel",
    "description": "Suche den Aussichtspunkt und genieße die Aussicht",
    "rankingPoints": 341,
    "stationen": [
      {
        "id": 6,
        "description": "Dreh dich 5 mal im Kreis wenn du vor der Eiche stehst und laufe dann gerade aus.",
        "longitude": 9.208858198755664,
        "latitude": 49.14785422283188,
        "code": 592482,
        "solution": "keine"
      },
      {
        "id": 7,
        "description": "Endstation",
        "longitude": 9.2111347,
        "latitude": 49.12199150000001,

```

Response Code

`200`

Response Headers

```
{  
  "date": "Tue, 25 Jun 2019 15:41:59 GMT",  
  "content-length": "5766",  
  "content-type": "application/json; charset=UTF-8"  
}
```

Api Documentation

Api Documentation

[Apache 2.0](#)

controller : All Backend API Operations for the geocaching Application

Show/Hide | List Operations | Expand Operations

GET /api/allCaches Retrieves all Caches, including their Stations, from the Database

Response Class (Status 200)

Model | Model Schema

```
Cache {
  description (string, optional): The Description of the Cache,
  id (integer, optional): The autogenerated CacheID (Primary Key),
  name (string, optional): The name of the Cache,
  rankingPoints (integer, optional): The number of ranking points the user gets fro completing the cache,
  reward (Reward, optional): The Reward for the Cache,
  stationen (Array[Station], optional): All stations included in the Cache
}

Reward {
  id (integer, optional): The autogenerated ID (Primary Key),
  rewardDescription (string, optional): The description for the reward
}

Station {
  code (integer, optional): The unique code for a Station (needed for the QR Code),
  description (string, optional): The riddle for the Station,
  id (integer, optional): The autogenerated ID (Primary Key),
  latitude (number, optional): The latitude of the position for the station,
  longitude (number, optional): The longitude of the position for the station,
  solution (string, optional): The solution for the riddle of the station
}
```

Response Content Type application/json

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/checkAdmin

Checks if the given User has an admin role

Response Class (Status 200)

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<div>(required)</div>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

PUT

/api/checkStation

Checks if the given Station is the correct next Station in the Cache

Response Class (Status 200)

Model | Model Schema

Bearbeitet {

aktuelleStation (Station, optional): The current Station (Foreign Key),

cache (Cache, optional): The Cache (Foreign Key),

cacheAccesDefinition (CacheAccesDefinition, optional): The cacheAccessDefinition (Foreign Key),

id (integer, optional): The autogenerated ID (Primary Key),

user (User, optional): The User (Foreign Key)

}

Station {

code (integer, optional): The unique code for a Station (needed for the QR Code),

description (string, optional): The riddle for the Station,

id (integer, optional): The autogenerated ID (Primary Key),

latitude (number, optional): The latitude of the position for the station,

longitude (number, optional): The longitude of the position for the station,

solution (string, optional): The solution for the riddle of the station

}

Cache {

description (string, optional): The Description of the Cache,

id (integer, optional): The autogenerated CacheID (Primary Key),

name (string, optional): The name of the Cache,

rankingPoints (integer, optional): The number of ranking points the user gets fro completing the cache,

reward (Reward, optional): The Reward for the Cache,

stationen (Array[Station], optional): All stations included in the Cache

localhost:8080/swagger-ui.html

2/23

```
}
CacheAccessDefinition {
  description (string, optional): The description for the CacheAccessDefinition,
  id (integer, optional): The autogenerated ID (Primary Key)
}
User {
  email (string, optional): The email of the user,
  id (integer, optional): The autogenerated ID (Primary Key),
  password (string, optional): The password for the user,
  roles (Array[Role], optional): A List of roles for the User (e.g. Cacher, Admin...),
  username (string, optional): The username of the user
}
Reward {
  id (integer, optional): The autogenerated ID (Primary Key),
  rewardDescription (string, optional): The description for the reward
}
Role {
  domain (string, optional): The domain of the role (e.g. geocaching, flowertours...),
  id (integer, optional): The autogenerated ID (Primary Key),
  name (string, optional): The name of the role
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
cacheID	<input type="text" value="(required)"/>	The CacheID from the scanned QR Code	query	string
durchgefuehrterCacheID	<input type="text" value="(required)"/>	The CacheID from the cache the user does at the time scanning the code	query	string
stationID	<input type="text" value="(required)"/>	The StationID from the scanned QR Code	query	string
token	<input type="text" value="(required)"/>	JWT Token indentifiying the cser	query	string

Response Messages

26.6.2019Swagger UI

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

POST/api/createCacheCreates a new Cache

Response Class (Status 200)

Response Content Typeapplication/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
cache	<div>(required)</div> <div>Parameter content type: application/json</div>	The cache to be created and saved in the database	body	<div>Model</div> <div>Model Schema</div> <div>Cache { description (string, optional): The Description of the Cache, id (integer, optional): The autogenerated CacheID (Primary Key), name (string, optional): The name of the Cache, rankingPoints (integer, optional): The number of ranking points the user gets fro completing the cache, reward (Reward, optional): The Reward for the Cache, stationen (Array[Station], optional): All stations included in the Cache }</div> <div>Reward { id (integer, optional): The autogenerated ID (Primary Key), rewardDescription (string, optional): The description for the reward</div>

Parameter	Value	Description	Parameter Type	Data Type
				<pre>} Station { code (integer, optional): The unique code for a Station (needed for the QR Code), description (string, optional): The riddle for the Station, id (integer, optional): The autogenerated ID (Primary Key), latitude (number, optional): The latitude of the position for the station, longitude (number, optional): The longitude of the position for the station , solution (string, optional): The solution for the riddle of the station }</pre>

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

POST

/api/createTeam

Creates a new Team

Response Class (Status 200)

Model | Model Schema

```
Team {  
  id (integer, optional): The autogenerated ID (Primary Key),  
  name (string, optional): The name of the team,  
  teamStatus (string, optional): The teamstatus  
}
```

Response Content Type

application/json

Parameters

26.6.2019Swagger UI

Parameter	Value	Description	Parameter Type	Data Type
name	<input type="text" value="(required)"/>	Name of the team to be created	query	string
token	<input type="text" value="(required)"/>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

DELETE/api/deleteCacheDeletes a Cache

Response Class (Status 200)

Response Content Typeapplication/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
cacheID	<input type="text" value="(required)"/>	The cache ID to be deleted	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

DELETE/api/deleteCacheForUserDeletes the bearbeitet entry from a User from a Cache

Response Class (Status 200)

Response Content Typeapplication/json

Parameters

26.6.2019Swagger UI

Parameter	Value	Description	Parameter Type	Data Type
cacheID	<input type="text" value="(required)"/>	The cache id for the cache to be deleted	query	string
token	<input type="text" value="(required)"/>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

DELETE /api/deleteTeamInvite Removes a team invite from a user (called if a user declined a team invite)

Response Class (Status 200)

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
teamInviteID	<input type="text" value="(required)"/>	The id of the TeamInvite to be deletd	query	string
token	<input type="text" value="(required)"/>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

PUT /api/editCache Edits a Cache

Response Class (Status 200)

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
newCache	<div>(required)</div> <div>Parameter content type: application/json</div>	The edited cache	body	<div>Model</div> <div>Model Schema</div> <div>Cache {<div>description (string, <i>optional</i>): The Description of the Cache,</div><div>id (integer, <i>optional</i>): The autogenerated CacheID (Primary Key),</div><div>name (string, <i>optional</i>): The name of the Cache,</div><div>rankingPoints (integer, <i>optional</i>): The number of ranking points the user gets fro completing the cache,</div><div>reward (Reward, <i>optional</i>): The Reward for the Cache</div><div>,</div><div>stationen (Array[Station], <i>optional</i>): All stations included in the Cache</div><div>}</div><div>Reward {<div>id (integer, <i>optional</i>): The autogenerated ID (Primary Key),</div><div>rewardDescription (string, <i>optional</i>): The description for the reward</div><div>}</div><div>Station {<div>code (integer, <i>optional</i>): The unique code for a Station (needed for the QR Code),</div><div>description (string, <i>optional</i>): The riddle for the Station,</div><div>id (integer, <i>optional</i>): The autogenerated ID</div></div></div></div>

Parameter	Value	Description	Parameter Type	Data Type
				(Primary Key), latitude (number, <i>optional</i>): The latitude of the position for the station, longitude (number, <i>optional</i>): The longitude of the position for the station , solution (string, <i>optional</i>): The solution for the riddle of the station }

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getAllStations

Returns all Stations

Response Class (Status 200)

Model | Model Schema

Station {
 code (integer, *optional*): The unique code for a Station (needed for the QR Code),
 description (string, *optional*): The riddle for the Station,
 id (integer, *optional*): The autogenerated ID (Primary Key),
 latitude (number, *optional*): The latitude of the position for the station,
 longitude (number, *optional*): The longitude of the position for the station,
 solution (string, *optional*): The solution for the riddle of the station
}

Response Content Type

application/json

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getCurrentStation

Returns the station the user is currently at (for a specific cache)

Response Class (Status 200)

Model | Model Schema

Station {

code (integer, optional): The unique code for a Station (needed for the QR Code),

description (string, optional): The riddle for the Station,

id (integer, optional): The autogenerated ID (Primary Key),

latitude (number, optional): The latitude of the position for the station,

longitude (number, optional): The longitude of the position for the station,

solution (string, optional): The solution for the riddle of the station

}

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
cacheID	(required)	The Cache id for the searched station	query	string
token	(required)	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getCurrentStationMap

Returns the Team of a user

Response Class (Status 200)

Model | Model Schema

POI {

categoryID (integer, optional),

latitude (number, optional),

longitude (number, optional),

name (string, optional)

}

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
cacheID	<div>(required)</div>	Cache id of the searched station	query	string
token	<div>(required)</div>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getMyCaches

Returns all Caches finished/started by a given User

Response Class (Status 200)

Model

Model Schema

```
Bearbeitet {
  aktuelleStation (Station, optional): The current Station (Foreign Key),
  cache (Cache, optional): The Cache (Foreign Key),
  cacheAccesDefinition (CacheAccesDefinition, optional): The cacheAccessDefinition (Foreign Key),
  id (integer, optional): The autogenerated ID (Primary Key),
  user (User, optional): The User (Foreign Key)
}

Station {
  code (integer, optional): The unique code for a Station (needed for the QR Code),
  description (string, optional): The riddle for the Station,
  id (integer, optional): The autogenerated ID (Primary Key),
  latitude (number, optional): The latitude of the position for the station,
  longitude (number, optional): The longitude of the position for the station,
  solution (string, optional): The solution for the riddle of the station
}

Cache {
  description (string, optional): The Description of the Cache,
  id (integer, optional): The autogenerated CacheID (Primary Key),
  name (string, optional): The name of the Cache,
  rankingPoints (integer, optional): The number of ranking points the user gets fro completing the cache,
  reward (Reward, optional): The Reward for the Cache,
  stationen (Array[Station], optional): All stations included in the Cache
}

CacheAccesDefinition {
```

description (string, *optional*): The description for the CacheAcessDefinition,

id (integer, *optional*): The autogenerated ID (Primary Key)

}

User {

email (string, *optional*): The email of the user,

id (integer, *optional*): The autogenerated ID (Primary Key),

password (string, *optional*): The password for the user,

roles (Array[Role], *optional*): A List of roles for the User (e.g. Cacher, Admin...),

username (string, *optional*): The username of the user

}

Reward {

id (integer, *optional*): The autogenerated ID (Primary Key),

rewardDescription (string, *optional*): The description for the reward

}

Role {

domain (string, *optional*): The domain of the role (e.g. geocaching, flowertours...),

id (integer, *optional*): The autogenerated ID (Primary Key),

name (string, *optional*): The name of the role

}

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<div>(required)</div>	JWT Token indentifying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getMyStationPOIS

Returns startstations and all other stations the user already visited as POIS

Response Class (Status 200)

Model | Model Schema

POI {

categoryID (integer, *optional*),

latitude (number, *optional*),

longitude (number, *optional*),

name (string, *optional*)

}

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="(required)"/>	JWT Token identifying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET /api/getMyTeamInvites Returns all User invites for a specific User

Response Class (Status 200)

Model | Model Schema

```
TeamInvite {
  id (integer, optional): The autogenerated ID (Primary Key),
  team (Team, optional): The team the user is invited to join,
  user (User, optional): The user to be invited
}

Team {
  id (integer, optional): The autogenerated ID (Primary Key),
  name (string, optional): The name of the team,
  teamStatus (string, optional): The teamstatus
}

User {
  email (string, optional): The email of the user,
  id (integer, optional): The autogenerated ID (Primary Key),
  password (string, optional): The password for the user,
  roles (Array[Role], optional): A List of roles for the User (e.g. Cacher, Admin...),
  username (string, optional): The username of the user
}

Role {
  domain (string, optional): The domain of the role (e.g. geocaching, flowertours...),
  id (integer, optional): The autogenerated ID (Primary Key),
  name (string, optional): The name of the role
}
```

Response Content Type application/json

Parameters

26.6.2019

Swagger UI

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="(required)"/>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET /api/getRankingList

Returns the rankinglist

Response Class (Status 200)

Model | Model Schema

RankingListHelper {
 rank (integer, optional),
 ranking_Points (integer, optional),
 username (string, optional)
}

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="null"/>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET /api/getRankingPlace

Returns the ranking place on the leaderboard for a specific user

Response Class (Status 200)

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
email	<input type="text" value="(required)"/>	The email of the user as unique identifier	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET /api/getTeam Returns a team by a name

Response Class (Status 200)

Model | Model Schema

Team {
 id (integer, optional): The autogenerated ID (Primary Key),
 name (string, optional): The name of the team,
 teamStatus (string, optional): The teamstatus
}

Response Content Type application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
name	<input type="text" value="(required)"/>	The name of the team	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getTeamMembers

Returns all teammates by the team name

Response Class (Status 200)

Model | Model Schema

User {
 email (string, optional): The email of the user,
 id (integer, optional): The autogenerated ID (Primary Key),
 password (string, optional): The password for the user,
 roles (Array[Role], optional): A List of roles for the User (e.g. Cacher, Admin...),
 username (string, optional): The username of the user
}
Role {
 domain (string, optional): The domain of the role (e.g. geocaching, flowertours...),
 id (integer, optional): The autogenerated ID (Primary Key),
 name (string, optional): The name of the role
}

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
name	<input type="text"/>	The name of the team	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getTeamOfUser

Returns the Team of a user

Response Class (Status 200)

Model | Model Schema

Team {
 id (integer, optional): The autogenerated ID (Primary Key),
 name (string, optional): The name of the team,
 teamStatus (string, optional): The teamstatus
}

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="(required)"/>	JWT Token indentifying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getTeamRankingList

Returns the Team Rankinglist

Response Class (Status 200)

Model | Model Schema

```
TeamRankingListHelper {
  ranking_Points (integer, optional),
  teamname (string, optional)
}
```

Response Content Type

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/getUser

Returns a user from a given token

Response Class (Status 200)

Model | Model Schema

```
User {
  email (string, optional): The email of the user,
  id (integer, optional): The autogenerated ID (Primary Key),
  password (string, optional): The password for the user,
```

roles (Array[Role], *optional*): A List of roles for the User (e.g. Cacher, Admin...),
username (string, *optional*): The username of the user

}

Role {

domain (string, *optional*): The domain of the role (e.g. geocaching, flowertours...),

id (integer, *optional*): The autogenerated ID (Primary Key),

name (string, *optional*): The name of the role

}

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<div>(required)</div>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET

/api/hello

Test method (Changes its purpose often)

Response Class (Status 200)

Model | Model Schema

```
{
  "body": {},
  "statusCode": "100 CONTINUE",
  "statusCodeValue": 0
}
```

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
name	<div>(required)</div>	name	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
------------------	--------	----------------	---------

26.6.2019Swagger UI

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

PUT/api/joinTeamLets the user join a Team

Response Class (Status 200)

ModelModel Schema

Team {
 id (integer, optional): The autogenerated ID (Primary Key),
 name (string, optional): The name of the team,
 teamStatus (string, optional): The teamstatus
}

Response Content Typeapplication/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
teamID	(required)	Team id of the team the user wants to join	query	string
token	(required)	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET/api/leaveTeamRemoves the user from the Team

Response Class (Status 200)

Response Content Typeapplication/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="(required)"/>	JWT Token indentifying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

PUT /api/leaveTeam Removes the user from the Team

Response Class (Status 200)

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="(required)"/>	JWT Token indentifying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

GET /api/resetRankingList Resets the rankinglist

Response Class (Status 200)

Response Content Type

Parameters

26.6.2019Swagger UI

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="(required)"/>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

PUT/api/resetSingleUserResets the points of a single User

Response Class (Status 200)

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
token	<input type="text" value="(required)"/>	JWT Token indentifiying the User (Admin in this case)	query	string
username	<input type="text" value="(required)"/>	The username of the user to be reseted	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

POST/api/sendTeamInviteSends a Team Invite to the invitedUserEmail from the User (token)

Response Class (Status 200)

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
invitedUserEmail	<div>(required)</div>	The email of the invited User	query	string
token	<div>(required)</div>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

PUT

/api/setTeamStatus

Sets the team status

Response Class (Status 200)

Response Content Type

application/json

Parameters

Parameter	Value	Description	Parameter Type	Data Type
teamStatus	<div>(required)</div>	The new team status	query	string
token	<div>(required)</div>	JWT Token indentifiying the User	query	string

Response Messages

HTTP Status Code	Reason	Response Model	Headers
400	Given Station is NOT the correct following station		
401	JWT Token expired		
404	Database error		

Try it out!

basic-error-controller : Basic Error Controller

Show/Hide | List Operations | Expand Operations

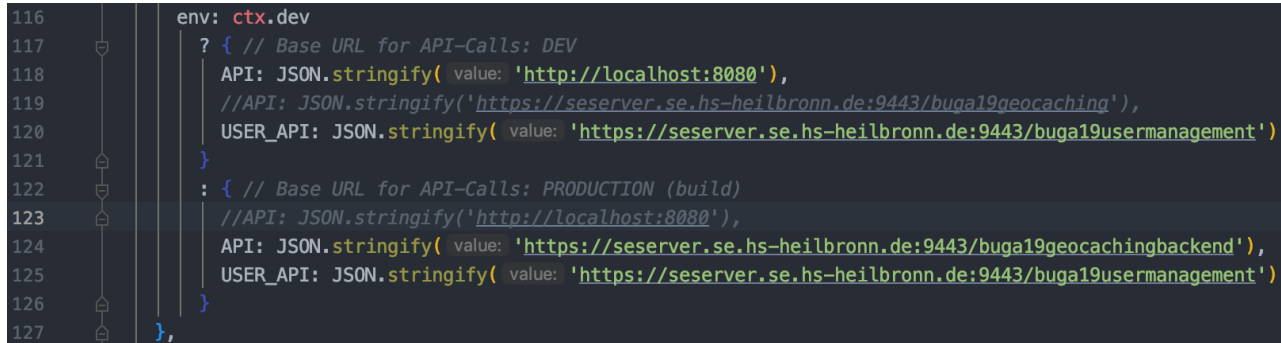
[BASE URL: / , API VERSION: 1.0]

Installationsanleitung

Falls Sie das Projekt lokal öffnen möchten, lesen Sie bitte diese [Anleitung](#) dafür.

Definieren der Backend URL:

Falls Sie die *.war Dateien auf einem anderen Tomcat Server wie dem seserver deployen möchten, ist es vor der Generierung noch wichtig die URL für die Backend API Calls anzupassen. Navigieren Sie hierzu in das *quasar.conf.js* File (relativer Dateipfad *labswp_2019_sose_geocaching_frontend/frontend/quasar.conf.js*). Ab Zeile 116 (siehe Abb. 1) lässt sich hier ein Codeblock finden, der definiert, wohin Backend API Calls geleitet werden.



```
116 env: ctx.dev
117   ? { // Base URL for API-Calls: DEV
118     API: JSON.stringify( value: 'http://localhost:8080'),
119     //API: JSON.stringify('https://seserver.se.hs-heilbronn.de:9443/buga19geocaching'),
120     USER_API: JSON.stringify( value: 'https://seserver.se.hs-heilbronn.de:9443/buga19usermanagement')
121   }
122   : { // Base URL for API-Calls: PRODUCTION (build)
123     //API: JSON.stringify('http://localhost:8080'),
124     API: JSON.stringify( value: 'https://seserver.se.hs-heilbronn.de:9443/buga19geocachingbackend'),
125     USER_API: JSON.stringify( value: 'https://seserver.se.hs-heilbronn.de:9443/buga19usermanagement')
126   }
127 },
```

Abb. 1

Ändern der URL für das lokale starten:

- Zeile 118 gibt an, mit welcher Adresse sich das Backend erreichen lässt. Für lokale Tests ist die meist 'http://localhost:8080'. Falls eine andere URL für Backend Aufrufe benutzt werden soll, beispielsweise ein anderer Port, muss dies hier angegeben werden.
- Zeile 120 gibt an, unter welcher Adresse das zentrale Usermanagement erreichbar ist. Ändert sich dessen Location, muss die URL hier entsprechend angepasst werden.

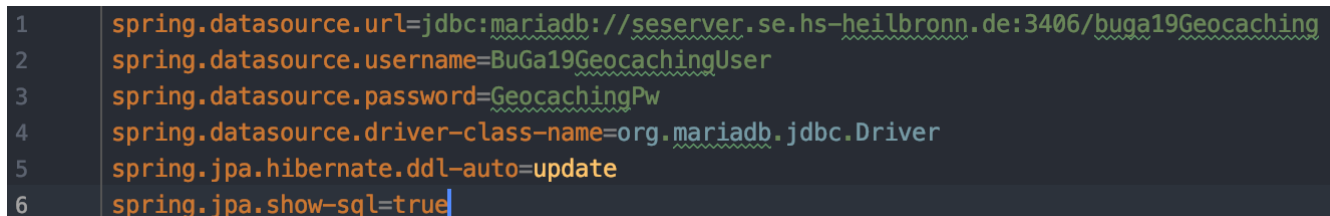
Ändern der URL für den Production Build:

- Zeile 124 gibt an, unter welcher Adresse das Geocaching Backend erreichbar ist. Durch die Trennung von Frontend und Backend ist es möglich, das Backend auf einem völlig anderen Server wie das Frontend laufen zu lassen. Wichtig ist nur, dass die URL des Backends hier richtig angegeben wird.
- Zeile 125 gibt an, unter welcher Adresse das zentrale Usermanagement erreichbar ist. Ändert sich dessen Location, muss die URL hier entsprechend angepasst werden.

Bitte passen Sie die entsprechenden Zeilen vor der Generierung der *.war Dateien an und speichern Sie sie ab.

Definieren der Backend Datenbank

Wenn eine andere Datenbank, wie die MariaDB Datenbank der Hochschule, verwendet werden soll, kann dies in der Anwendung definiert werden. Öffnen Sie hierfür die application.properties Datei (relativer Dateipfad: *labswp_2019_sose_geocaching/src/main/resources/application.properties*).



```
1 spring.datasource.url=jdbc:mariadb://seserver.se.hs-heilbronn.de:3406/buga19Geocaching
2 spring.datasource.username=BuGa19GeocachingUser
3 spring.datasource.password=GeocachingPw
4 spring.datasource.driver-class-name=org.mariadb.jdbc.Driver
5 spring.jpa.hibernate.ddl-auto=update
6 spring.jpa.show-sql=true
```

Abb. 2

Wie in Abb. 2 zu sehen ist, besteht die application.properties Datei aus 6 Zeilen, die die Datenbank definieren:

- Zeile 1: Die URL der Datenbank
- Zeile 2: Der Username für den Zugriff auf die Datenbank
- Zeile 3: Das Passwort für den Zugriff auf die Datenbank
- Zeile 4: Die Treiberklasse für die Datenbank. Der Treiber unterscheidet sich je nach Datenbank und muss ausgetauscht werden
- Zeile 5: Beschreibt, ob neue Datenbanktabellen erzeugt oder geupdated werden sollen
- Zeile 6: Unwichtig für die Definition der Datenbank. Sagt aus, ob in den Logs die ausgeführten SQL Statements geschrieben werden sollen

Falls Sie eine andere Datenbank nutzen wollen, passen Sie dieses File bitte an und speichern es ab.

Generierung der deploybaren *.war Dateien:

Hintergrundinformationen zur Generierung :

Da wir als Buildtool Gradle verwenden, nutzen wir zur Generierung der *.war Dateien das 'war' Plugin für Gradle. Dieses Plugin fügt Gradle die Befehle 'war' und 'bootWar' hinzu. Der Befehl 'war' führt dazu, dass Gradle aus dem Projekt eine ausführbare *.war Datei erzeugt. Die zwei *.war Dateien, die auf den Tomcat deployed (buga19geocaching (frontend) und buga19geocachingbackend) werden durch zwei build.gradle Dateien generiert (jeweils mit 'war' Plugin). Durch die settings.gradle Datei sorgen wir dafür, dass beide *.war Dateien durch nur einen Aufruf von './gradlew build' auf Mac OSX bzw. 'gradlew build' auf Windows generiert werden. In den build.gradle Dateien ist durch das Attribut 'archiveName' zusätzlich festgelegt, wie die Namen der erzeugten *.war Dateien lauten sollen. In unserem Fall *buga19geocachingbackend.war* für das Backend und *buga19geocaching.war* für das Frontend.

Die eigentliche Generierung:

Öffnen Sie das Terminal und navigieren Sie, falls nicht schon automatisch geschehen, zum Verzeichnis, in dem Sie das Projekt gespeichert haben (Terminal ähnlich Abb. 3).

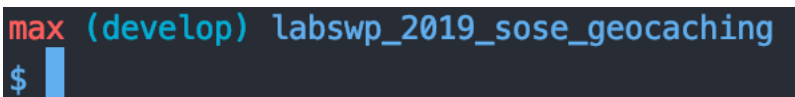


Abb. 3

Nutzen Sie nun den Befehl './gradlew clean build' auf Mac OSX oder 'gradlew clean build' auf Windows um die *.war Dateien zu generieren

Deployen der generierten *.war Dateien:

Wenn die *.war Dateien über den zuvor beschriebenen Weg generiert wurden, sind sie im Ordner '/build/libs' zu finden (Abb. 4). Im Ordner befindet sich auch noch eine *.jar Datei. Diese ist jedoch für die Installation nicht wichtig.

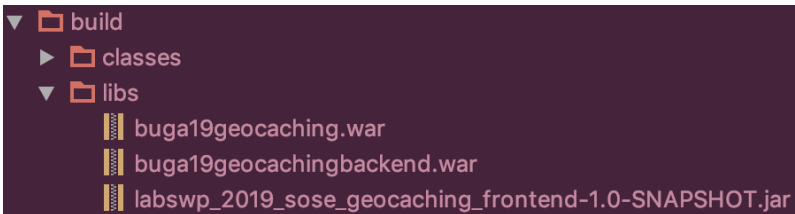


Abb. 4

Unser Backend ist darauf ausgelegt auf dem Tomcat 9 Server der Hochschule zu laufen (<http://seserver.se.hs-heilbronn.de:9080>). Der Tomcat 9 Server für das Frontend kann theoretisch frei gewählt werden. Wir empfehlen jedoch auch den Server der Hochschule zu verwenden.

Um die *.war Dateien auf dem Server der Hochschule zu deployen, besuchen Sie die Seite <http://seserver.se.hs-heilbronn.de:9080/manager/html> und melden sich mit den Credentials an, die Ihnen der Admin (Prof. Dr.-Ing. Permantier) zur Verfügung gestellt hat.

Bevor Sie die neuen *.war Dateien hochladen, achten Sie darauf, dass 'buga19geocaching.war' und 'buga19geocachingbackend.war' nicht bereits deployed sind. Falls dies der Fall ist, suchen Sie die zwei Anwendungen in der Anwendungsliste und drücken Sie den 'Entfernen' Button am rechten Rand. Wenn Sie beide *.war erfolgreich entfernt haben, können Sie die Anwendungen im nächsten Schritt erneut deployen.

Scrollen Sie nun bis zum Ende der Seite nach unten und wählen Sie 'Datei auswählen' aus (Abb. 5).



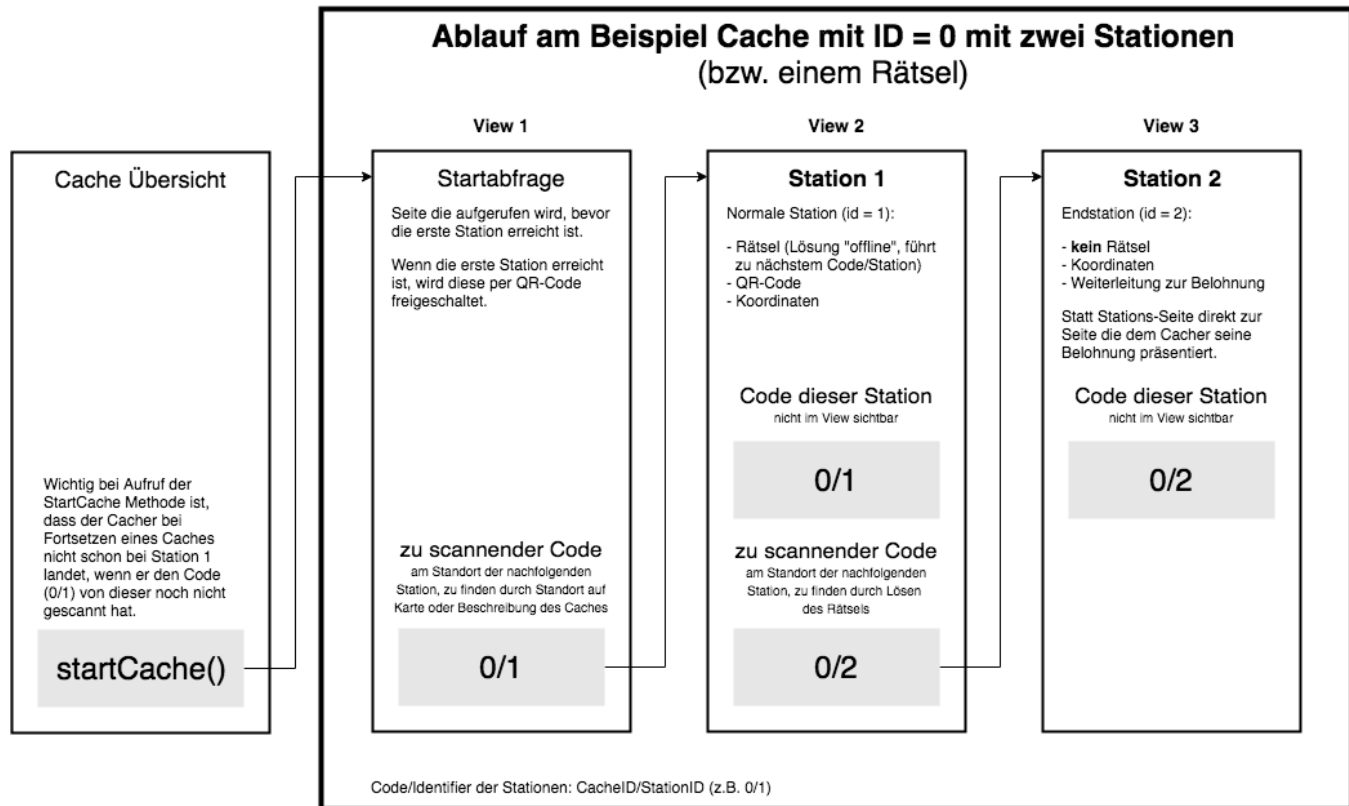
Abb. 5

Wählen Sie nun zuerst die 'buga19geocachingbackend.war' Datei aus und klicken Sie auf 'installieren'. Sie laden nun die Datei auf den Server hoch. Sobald die Seite sich selbst aktualisiert und oben 'OK' angezeigt wird, haben Sie die 'buga19geocachingbackend.war' Datei erfolgreich hochgeladen. Wiederholen Sie diese Schritte nun für die 'buga19geocaching.war' Datei.

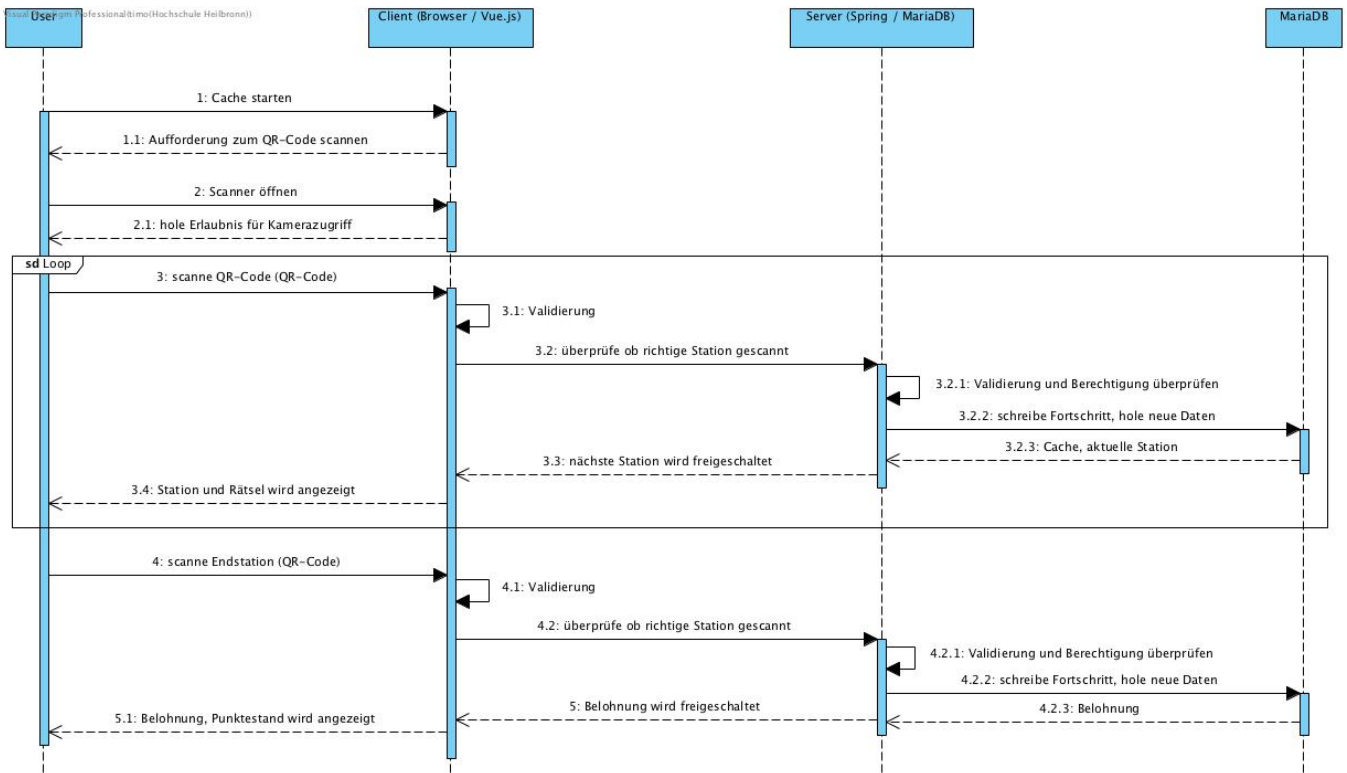
Wenn beide *.war Dateien erfolgreich hochgeladen wurden, ist die Anwendung unter <https://seserver.se.hs-heilbronn.de:9443/buga19geocaching> erreichbar.

Wichtige Abläufe

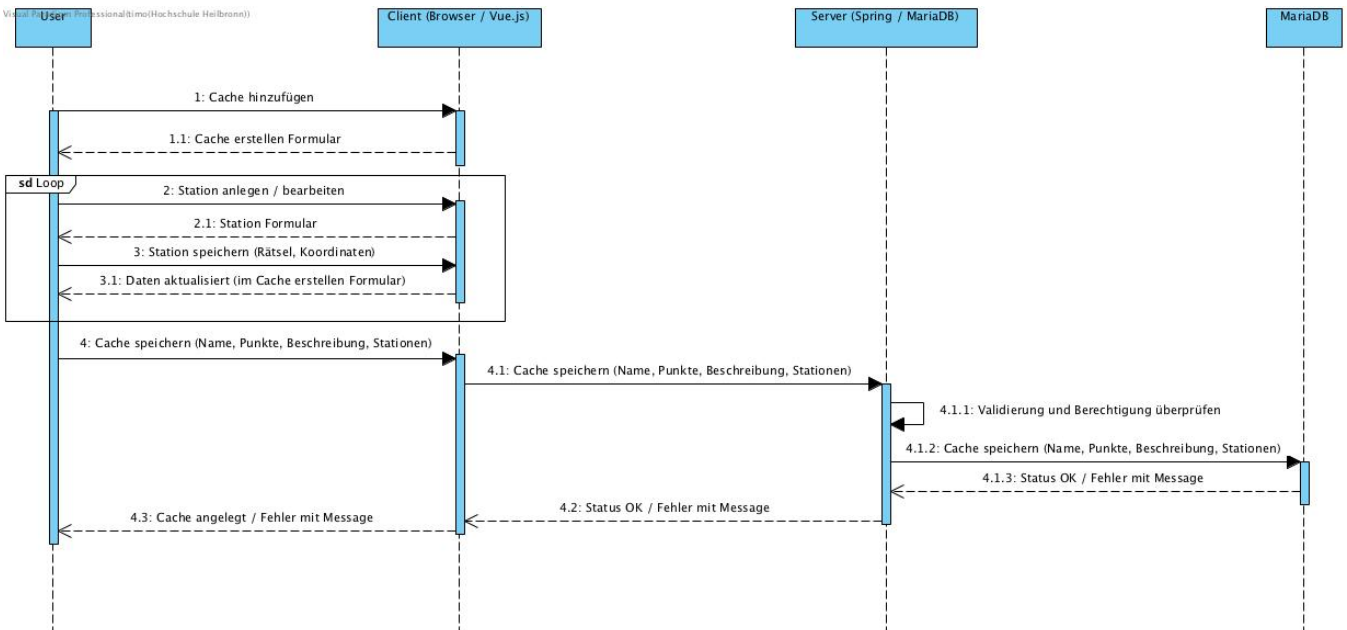
Ablauf der Durchführung eines Caches



Unter "Diagramme" finden sie die Visual Paradigm Dateien für die zwei Sequenzdiagramme.



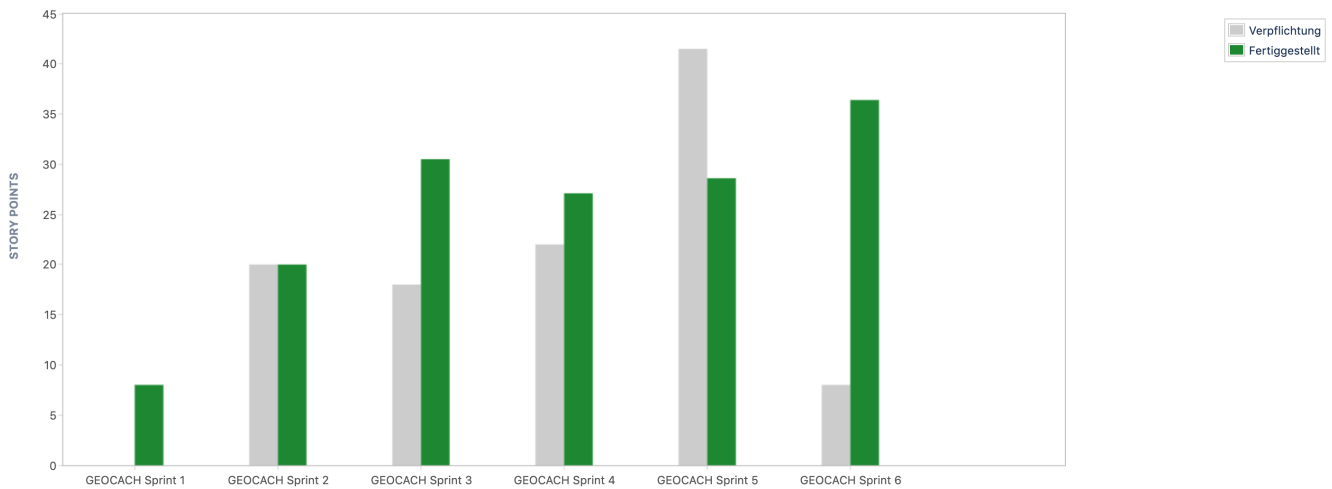
Anlegen eines Caches



Leistungen pro Teammitglied

Velocity über die verschiedenen Sprints hinweg:

Die Userstories für Sprint 1 wurden erst nach Start des Sprints hinzugefügt. Deshalb werden sie hier noch nicht als 'Verpflichtung' angezeigt.



Erledigte Aufgaben pro Teammitglied insgesamt:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	33	33
Maximilian Leopold	65	65
Michael Dietz	56	56
Robin Christ	22	22
Timo Volkmann	33	33
Summe, Vorgänge:	284	284

5 von 6 Statistiken werden angezeigt.

Erledigte Aufgaben pro Teammitglied pro Sprint:

Sprint 1:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	2	2
Maximilian Leopold	4	4
Michael Dietz	4	4
Timo Volkmann	2	2
Nicht zugewiesen	6	6
Summe, Vorgänge:	18	18

5 von 5 Statistiken werden angezeigt.

Sprint 2:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	4	4
Maximilian Leopold	5	5
Michael Dietz	17	17
Robin Christ	2	2
Timo Volkmann	6	6
Summe, Vorgänge:	44	44

5 von 6 Statistiken werden angezeigt.

Sprint 3:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	5	5
Maximilian Leopold	3	3
Michael Dietz	10	10
Robin Christ	9	9
Timo Volkmann	4	4
Summe, Vorgänge:	34	34

5 von 6 Statistiken werden angezeigt.

Sprint 4:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	8	8
Maximilian Leopold	8	8
Michael Dietz	16	16
Robin Christ	8	8
Timo Volkmann	14	14
Summe, Vorgänge:	64	64

5 von 6 Statistiken werden angezeigt.

Sprint 5:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	8	8
Maximilian Leopold	11	11
Michael Dietz	5	5
Robin Christ	4	4
Timo Volkmann	8	8
Summe, Vorgänge:	57	57
5 von 6 Statistiken werden angezeigt.		

Sprint 6:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	6	6
Maximilian Leopold	36	36
Michael Dietz	7	7
Robin Christ	3	3
Timo Volkmann	6	6
Summe, Vorgänge:	84	84
5 von 6 Statistiken werden angezeigt.		

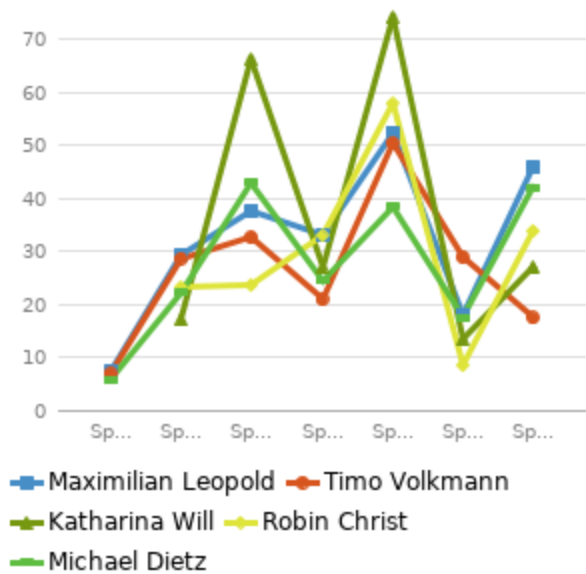
Sprint 7:

Bearbeiter	Status	
	FERTIG	S:
Katharina Will	9	9
Maximilian Leopold	11	11
Michael Dietz	3	3
Robin Christ	3	3
Timo Volkmann	3	3
Summe, Vorgänge:	37	37

5 von 6 Statistiken werden
angezeigt.

Geloggte Zeit pro Teammitglied pro Sprint:

Alle Zeitangaben sind in Stunden



Bearbeitete User Stories/Subtasks

In den ersten zwei Sprints wurde die Zeit noch nicht konsequent geloggt, da wir uns erst in die neuen Technologien rund um Jira einarbeiten mussten und diese Funktion nicht von Anfang an benutzt wurde. Die Zeiten entsprechen also nicht den wirklich gearbeiteten Stunden. Sprint 1 umfasst auch noch viel Zeit für die Einarbeitung in die neuen Technologien.

Syntax des Aufschriebs:

Geocach-SubTask Nummer <(Geocach-User Story Nummer)>

Beschreibung der SubTask

Wenn keine User Story Nummer angegeben wurde, handelt es sich hierbei um eine zusätzliche Aufgabe, die in diesem Sprint abgearbeitet wurde, aber nicht zu einer bestimmten User Story gehört.

User Stories:

Geocach-1

Als Cachender will ich einen Cache von der Website aus starten können

Geocach-2

Als Cachender/Besuchender will ich eine Übersicht über alle Caches sehen können, um mir einen Überblick zu verschaffen

Geocach-3

Als Cachender/Administrierender will ich mich anmelden

Geocach-4

Als Cachender/Administrierender will ich mich abmelden

Geocach-5

Als Besuchender der BuGa will ich Informationen über das GeoCaching Angebot erhalten

Geocach-6

Als Cachender/Besuchender will ich jederzeit die Rangliste einsehen können

Geocach-7

Als Administrierender will ich einen neuen Cache erstellen können

Geocach-8

Als Administrierender will ich Caches entfernen können

Geocach-9

Als Cachender will ich meine angefangenen und abgeschlossenen Caches sehen können

Geocach-10

Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, sodass ich dem Endpunkt des Caches näherkomme

Geocach-11

Als BuGa-Besuchender will ich mich registrieren

Geocach-12

Als Cachender will ich den Fortschritt meines Cache verfolgen können

Geocach-13

Als Cachender will ich Teams erstellen können

Geocach-14

Als Cachender will ich nachdem ich einen Cache gefunden habe, eine Belohnung abholen können

Geocach-15

Als Cachender will ich die Startpunkte eines Caches auf der Karte sehen können

Geocach-16

Als Cachender möchte ich andere zu meinem Team einladen können

Geocach-17

Als Cachender will ich einen QR-Code scannen können, sodass ich einen Cache starte

Geocach-18

Als Administrierender will ich Caches bearbeiten können

Geocach-19

Als Cachender möchte ich meinen Standort auf der Karte sehen können

Geocach-20

Als Cachender möchte ich Caches von der Karte aus aufrufen können

Geocach-41

Als Administrierender will ich eine neue Station erstellen können

Geocach-115

Als Cachender will ich mein Profil mit Informationen über meinen Account ansehen könne

Geocach-139

API Aufruf, der die Daten über die Caches in geeigneter JSON Form zurückgibt um daraus eine Karte bei dem Lageplan Team anzufordern

Geocach-187

Als Anbieter von GeoCaches möchte ich die Bestenliste gelegentlich zurücksetzen, so dass neue Teilnehmer eine Chance haben in die besten 10 zu kommen

Teammitglied	Sprint 1 (14.03.2019 - 28.03.2019)	Geloggte Zeit	Story Point Beteiligung
--------------	------------------------------------	---------------	-------------------------

Maximilian Leopold	<u>Geocach-26 (Geocach-2)</u>		7
	Rückgabe eines JSON mit allen Caches	2h	
	<u>Geocach-27 (Geocach-3)</u>		
	Verschlüsselung des Passworts bei Übergabe	2h	
	<u>Geocach-28 (Geocach-3)</u>		
	Hashen des Passworts ind DB	1h 30m	
	<u>Geocach-30 (Geocach-5)</u>		
	Informationstext verfassen	2h	
Timo Volkmann	<u>Geocach-25 (Geocach-1)</u>		6
	Darstellung aller Caches	7h	
Kathrina Will	<u>Geocach-29 (Geocach-3)</u>		5
	Darstellung des Login Screens		
	<u>Geocach-32 (Geocach-5)</u>		
	Darstellung für Informationstext		
Robin Christ			0
Michael Dietz	<u>Geocach-21 (Geocach-1)</u>		4
	Senden eines POST Requests zum starten eines Caches	3h	
	<u>Geocach-23 (Geocach-1)</u>		
	Datenbanktabelle für Caches aufsetzen	1h	
	<u>Geocach-24 (Geocach-1)</u>		
	Datenbanktabelle für User aufsetzen	1h	
	<u>Geocach-32 (Geocach-5)</u>		
	Logout API Call implementieren	1h	

Teammitglied	Sprint 2 (29.03.2019 - 11.04.2019)	Geloggte Zeit	Story Point Beteiligung
--------------	------------------------------------	---------------	-------------------------

Maximilian Leopold	<u>Geocach-34 (Geocach-6)</u>		2
	Datenbankaufruf für Rangliste erstellen	2h 30m	
	<u>Geocach-35 (Geocach-6)</u>		
	JSON aus Datenbanktabelle generieren	4h	
	<u>Geocach-36 (Geocach-6)</u>		
	API Aufruf für Ranglist erstellen	3h	
	<u>Geocach-40 (Geocach-6)</u>		
	API Dokumentation erstellen	6h	
	<u>Geocach-60</u>		
	Meeting mit anderen Gruppe über Usermanagement und Lageplan	2h 30m	
	<u>Geocach-59</u>		
	JWT implementieren um User Datenbanken mit anderen Gruppe zusammenzuführen	3h 45m	
	<u>Geocach-61 (Bug)</u>		
	Spring Securityconfig funktioniert nichtmehr richtig	30m	
	<u>Geocach-64 (Bug)</u>		
	There can be multiple same database entries holding the same information in the "bearbeitet" table	30m	
	<u>Geocach-62</u>		
Timo Volkmann	Überarbeitung und Spezifizierung aller Akzeptanzkriterien im Product Backlog	3h	13
	<u>Geocach-63</u>		
	Durchlesen und Anmerkungen schreiben für Produkt Auftrag	30m	
	<u>Geocach-58</u>		
	Daily Scrum Sprint 2	1h 15m	
	<u>Geocach-68</u>		
	Sprint Restrospective/Review	2h	
	<u>Geocach-22 (Geocach-1)</u>		
	Darstellung der Caches	7h	
	<u>Geocach-48 (Geocach-7)</u>		
	Seite zum Erstellen von Caches erstellen(cache attribute + stationen hinzufügen)	1d 5h	
	<u>Geocach-49 (Geocach-8)</u>		
	Seite zum Löschen von Caches erstellen/Cache löschen Funktion in vorhandene Cacheübersicht für Administratoren einbauen	2h	
	<u>Geocach-51 (Geocach-41)</u>		
	Im Frontend je nach Ergebnis des Adminchecks die Seite anzeigen lassen oder nicht	1h	
	<u>Geocach-54 (Geocach-7)</u>		
	Darstellung der angefangenen und abgeschlossenen Caches erstellen	2h	
	<u>Geocach-63</u>		
	Durchlesen und Anmerkungen schreiben für Produkt Auftrag	30m	
	<u>Geocach-58</u>		
	Daily Scrum Sprint 2	1h 15m	
	<u>Geocach-68</u>		
	Sprint Restrospective/Review	2h	

Kathrina Will	<u>Geocach-56 (Geocach-6)</u>	7h 30m	2
	View erstellen (für Rangliste)		
	<u>Geocach-57 (Geocach-6)</u>		
	Datenabfrage vom Backend implementieren	4h	
	<u>Geocach-60</u>		
	Meeting mit anderen Gruppe über Usermanagement und Lageplan	2h 30m	
Robin Christ	<u>Geocach-58</u>		5
	Daily Scrum Sprint 2	1h 15m	
	<u>Geocach-68</u>		
	Sprint Restrospective/Review	2h	
Robin Christ	<u>Geocach-55 (Geocach-9)</u>	2d 4h	5
	Darstellung der angefangenen und abgeschlossenen Caches erstellen		
	<u>Geocach-58</u>		
	Daily Scrum Sprint 2	1h 15m	
	<u>Geocach-68</u>		
	Sprint Restrospective/Review	2h	

Michael Dietz	<u>Geocach-37 (Geocach-9)</u>		18
	Zusammenstellen aller Datenbankeinträge für einen bestimmten User	1h	
	<u>Geocach-38 (Geocach-9)</u>		
	API Aufruf für Datenbankeinträge erstellen	2h	
	<u>Geocach-39 (Geocach-9)</u>		
	API Dokumentation schreiben	30m	
	<u>Geocach-42 (Geocach-41)</u>		
	API-Aufruf mit den benötigten Daten erstellen	2h	
	<u>Geocach-43 (Geocach-41)</u>		
	Neu erstellte Station in der Datenbank speichern	30m	
	<u>Geocach-45 (Geocach-7)</u>		
	API-Aufruf mit den benötigten Daten erstellen	4h	
	<u>Geocach-46 (Geocach-7)</u>		
	Neu erstellten Cache in der Datenbank speichern mit all seinen Verknüpfungstabellen	30m	
	<u>Geocach-47 (Geocach-7)</u>		
	API-Aufruf, der ein JSON von allen Stationen zurückgibt(die man dann dem cache hinzufügen kann)	2h	
	<u>Geocach-50 (Geocach-8)</u>		
	Funktion um Cache aus Datenbank zu löschen	1h	
	<u>Geocach-65</u>		
	User Story details im product backlog überarbeiten	1h	
	<u>Geocach-66 (Bug)</u>		
	deleteCache soll auch die Stationen löschen, die zu dem Cache gehören	1h	
	<u>Geocach-67 (Bug)</u>		
	Ein Cache soll nur erstellt werden, wenn dieser mindestens 2 Stationen hat	15m	
	<u>Geocach-63</u>		
	Durchlesen und Anmerkungen schreiben für Produkt Auftrag	15m	
	<u>Geocach-58</u>		
	Daily Scrum Sprint 2	1h 15m	
	<u>Geocach-68</u>		
	Sprint Restrospective/Review	2h	
	<u>Geocach-53 (Geocach-41)</u>		
	API zum checken, ob der user ein Admin ist und damit Zugriff auf diese Funktion hat	1h	
	<u>Geocach-52 (Geocach-7)</u>		
	API zum checken, ob der user ein Admin ist und damit Zugriff auf diese Funktion hat	2h	
Teammitglied	Sprint 3 (12.04.2019 - 25.04.2019)	Geloggte Zeit	Story Point Beteiligung

Maximilian Leopold	<u>Geocach-94</u>		8,5
	Gemeinsame Userverwaltungs API durchlesen und verstehen	1h 15m	
	<u>Geocach-91</u>		
	Präsentationsvorbereitung	1d 2h	
	<u>Geocach-96</u>		
	Code Refactoring und Klassenunterteilung	4h	
	<u>Geocach-86 (Geocach-10)</u>		
	Veränderter Fortschritt in einem Cache verfolgen	2h	
	<u>Geocach-106</u>		
	API methoden verbessern/simplifizieren	3h	
	<u>Geocach-95</u>		
	Anpassen des Backends auf die neue zentrale Userverwaltung	1d 2h	
	<u>Geocach-111 (Bug)</u>		
Timo Volkmann	Delete Triggers wont work correctly	1h	19
	<u>Geocach-89</u>		
	Daily Scrum Sprint 3	2h	
	<u>Geocach-104</u>		
	Meetings	2h 30m	
	<u>Geocach-90</u>		
	Review & Retrospective Sprint 3	2h	
	<u>Geocach-110</u>		
	Confluence API Dokumentation überarbeiten	2h	
	<u>Geocach-101</u>		
	URL zentral verwalten im Frontend	2h 30m	
	<u>Geocach-70 (GEOCACH-10)</u>		
	QR-Code Scanner im Frontend implementieren	2d 7h	
	<u>Geocach-89</u>		
	Daily Scrum Sprint 3	2h	
	<u>Geocach-104</u>		
	Meetings	1h 15m	
	<u>Geocach-90</u>		
	Review & Retrospective Sprint 3	2h	

Kathrina Will	<u>Geocach-71 (Geocach-10)</u>		13
	Darstellung die einem zeigt, dass die gefundene Station die Falsche ist	7h	
	<u>Geocach-91</u>		
	Präsentationsvorbereitung	3d 4h	
	<u>Geocach-89</u>		
	Daily Scrum Sprint 3	2h	
	<u>Geocach-104</u>		
	Meetings	1h 15m	
Robin Christ	<u>Geocach-90</u>		8
	Review & Retrospective Sprint 3	2h	
	<u>Geocach-79 (Geocach-10)</u>		
	Seite die dem Cachenden die nächste Station anzeigt	1d 6h	
	<u>Geocach-93 (Geocach-10)</u>		
	Anzeige, dass Cache abgeschlossen ist	1d 4h	
	<u>Geocach-75 (Geocach-11)</u>		
	Seite die die benötigten Daten anzeigt und dem User ermöglicht diese Daten auszufüllen (für Registrierung)	1d	
	<u>Geocach-73 (Geocach-11)</u>		8
	Seite die einen Button "registrieren" enthält	1h	
	<u>Geocach-74 (Geocach-11)</u>		
	API-Aufruf der Registrieren Funktion	3h	
	<u>Geocach-78 (Geocach-11)</u>		
	Fehlermeldungen für falsche Eingaben ausgeben	1h	
	<u>Geocach-76 (Geocach-11)</u>		
	Validierung der Daten bei Eingabe	3h	
	<u>Geocach-99</u>		8
	Fehlermeldungen implementieren	2h	
	<u>Geocach-77 (Geocach-11)</u>		
	Seite die anzeigt, dass die Registrierung erfolgreich war	30m	
	<u>Geocach-89</u>		
	Daily Scrum Sprint 3	2h	
	<u>Geocach-104</u>		
	Meetings	1h 15m	
	<u>Geocach-90</u>		8
	Review & Retrospective Sprint 3	2h	

Michael Dietz	<u>Geocach-105</u> API-Dokumentation überarbeiten 1h <u>Geocach-80 (Geocach-10)</u> API-Dokumentation 20m <u>Geocach-72 (Geocach-10)</u> API-Aufruf Korrektheit der Station prüfen und Übergabe nächster Station 7h <u>Geocach-86 (Geocach-10)</u> Veränderter Fortschritt in einem Cache verfolgen 3h <u>Geocach-107 (Bug)</u> Fix a bug which causes users not to be found correctly 2h <u>Geocach-109 (Bug)</u> Fixed a bug which caused createCache not to work anymore 2h <u>Geocach-106</u> API methoden verbessern/simplifizieren 4h <u>Geocach-95</u> Anpassen des Backends auf die neue zentrale Userverwaltung 2d <u>Geocach-110</u> Confluence API Dokumentation überarbeiten 30m <u>Geocach-113 (Bug)</u> Alle Stationsaufrufe in einer methode vereinheitlichen 2h <u>Geocach-89</u> Daily Scrum Sprint 3 2h <u>Geocach-104</u> Meetings 1h 15m <u>Geocach-90</u> Review & Retrospective Sprint 3 2h	20
---------------	---	----

Teammitglied	Sprint 4 (26.04.2019 - 09.05.2019)	Geloggte Zeit	Story Point Beteiligung
Maximilian Leopold	<u>Geocach-141 (Bug)</u> Punkteanzeige in Rangliste fixen <u>Geocach-154 (Geocach-16)</u> API Call, der alle Teammitglieder eines Teams zurückliefert <u>Geocach-155 (Geocach-16)</u> API Call, der ein Team anhand seines Namens zurückgibt <u>Geocach-91</u> Präsentationsvorbereitung <u>Geocach-118 (115)</u> API Aufruf, der alls Daten über einen bestimmten User aus der Datenbank als JSON zurückgibt <u>Geocach-170</u> Umstellen der API Dokumentation auf Swagger <u>Geocach-186</u> API Call, der die letzte besuchte Station eines Caches in geeigneter Lageplan Form zurückgibt <u>Geocach-96</u> Code Refactoring und Klassenunterteilung	2h 1h 30m 30m 1d 5h 6h 4h 4h 2h	17,1

Timo Volkmann	<u>Geocach-140</u>		17,8
	Falls User eingeloggt ist "Registrieren"-Button ausblenden	10m	
	<u>Geocach-159 (Bug)</u>		
	Email wird zweimal eingetragen automatisch bei Registrierung	20m	
	<u>Geocach-163 (Bug)</u>		
	Username wird bei Registrierung abgefragt, aber nie benutzt	10m	
	<u>Geocach-169 (Bug)</u>		
	Automatisches Logout, bzw. Weiterleitung zu Login, falls Token expired ist	45m	
	<u>Geocach-168 (Bug)</u>		
	Problem bei "Meine Caches"	30m	
	<u>Geocach-187 (Bug)</u>		
	Scannen eines QR-Codes führt zu unverständlicher roter Fehlermeldung	45m	
	<u>Geocach-165 (Bug)</u>		
	Nachdem QR Code gescannt wurde (beim Cachen) soll Kamera sofort zentriert sein (es muss automatisch nach unten gescrollt werden)	1h	
	<u>Geocach-161 (Bug)</u>		
Katharina Will	Automatische Weiterleitung auf Registrierung, bzw. Login nach Registrierung oder sofortiger Login	3h 45m	16
	<u>Geocach-100</u>		
	Router Guard einrichten (User/Cacher/Admin)	2h	
	<u>Geocach-164 (Bug)</u>		
	Nach Abschluss des Caches soll nicht nur Seite mit drei Buttons angezeigt werden. Eine der Optionen soll standardmäßig angezeigt werden (Bsp. Belohnung)	30m	
	<u>Geocach-166 (Bug)</u>		
	QR Code Generator beim erstellen von Caches	1h	
	<u>Geocach-145</u>		
	Frontend Refactoring, Layout optimieren und Bugfixes	1d 2h 15m	
Katharina Will	<u>Geocach-136 (Geocach-115)</u>		16
	Darstellen der Team Rangliste	1d 1h	
	<u>Geocach-124 (Geocach-12)</u>		
	Karte die den Startpunkt eines gerade gestarteten Cache anzeigt implementieren	3h	
	<u>Geocach-246 (Geocach-15)</u>		
	Einbinden der Karte	1d 7h	

Robin Christ	<u>Geocach-121 (Geocach-16)</u>		14
	Seite zur Verwaltung eines Teams	2h	
	<u>Geocach-119 (Geocach-115)</u>		
	Darstellung des Profils mit allen Daten über einen Nutzer	1d 1h	
	<u>Geocach-120 (Geocach-13)</u>		
	Seite zum Team erstellen erstellen	2h	
	<u>Geocach-122 (Geocach-16)</u>		
	Option Nutzer ins Team einzuladen (über Email)	1d 3h	
	<u>Geocach-123 (Geocach-16)</u>		
	Option Nutzer aus dem Team zu entfernen	1h	
	<u>Geocach-132 (Geocach-16)</u>		
	Option für alle Teammitglieder in der Teamübersicht einen Teamstatus zu setzen	2h	
	<u>Geocach-128 (Geocach-16)</u>		
	Seite zur Anzeige aller Teammitglieder	3h	
	<u>Geocach-98</u>		
	Eingabe Validierung im Frontend	3h	

Michael Dietz	<u>Geocach-116 (Geocach-13)</u>		18,2
	API-Aufruf zum Team erstellen	3h	
	<u>Geocach-130 (Geocach-13)</u>		
	Neu erstelltes Team in der Datenbank speichern	30m	
	<u>Geocach-125 (Geocach-16)</u>		
	API Call, der einen Nutzer einem Team hinzufügt	3h	
	<u>Geocach-126 (Geocach-16)</u>		
	API Call, der einen Nutzer aus einem Team entfernt	2h	
	<u>Geocach-149</u>		
	API Aufruf getTeamRankingList erstellen	2h	
	<u>Geocach-147 (Geocach-16)</u>		
	API Aufruf zum Versenden einer Teameinladung	2h	
	<u>Geocach-148 (Geocach-16)</u>		
	Teameinladungsdatenbanktabelle erstellen	30m	
	<u>Geocach-150 (Geocach-115)</u>		
	API methode, die alle Teameinladungen zurückliefert	1h	
	<u>Geocach-151 (Geocach-16)</u>		
	API Methode zum Löschen einer Teameinladung	1h	
	<u>Geocach-152</u>		
	Alte unbenutzte Methoden löschen	30m	
	<u>Geocach-153</u>		
	Teamstatus hinzugefügen und setTeamStatus API methode erstellen	1h	
	<u>Geocach-156 (Bug)</u>		
	fixed a bug with checkStation	2h	
	<u>Geocach-157 (Bug)</u>		
	Fixed mistakes done in previous merges	2h	
	<u>Geocach-158</u>		
	Confluence API Dokumentation updaten	2h	
	<u>Geocach-162 (Geocach-12)</u>		
	API methode, die alle stationen als POIs zurückliefert die startStationen sind oder bereits vom benutzer gefunden wurden(wird dann vom frontend zur lageplangruppe weitergeleitet)	2h	
	<u>Geocach-137 (Geocach-115)</u>		
	API Aufruf zum ändern des Team Status	1h	
Teammitglied	Sprint 5 (10.05.2019 - 23.05.2019)	Geloggte Zeit	Story Point Beteiligung

Maximilian Leopold	<u>Geocach-180 (Bug)</u>		17,5
	Verschiedenste kleine Bugfixes in verschiedenen API Calls, die für die Anzeige des Profils nötig sind	4h	
	<u>Geocach-185 (Geocach-18)</u>		
	API aufruf, welcher den bearbeiteten Cache in der Datenbank speichert	3h	
	<u>Geocach-186 (Geocach-12)</u>		
	API Call, der die letzte besuchte Station eines Caches in geeigneter Lageplan Form zurückgibt	4h	
	<u>Geocach-252</u>		
	In der Rangliste unterhalb der Top 10, seinen eigenen Rang + Punkte anzeigen, eventuell highlighten. Durch Anklicken des eigenen Namens auf Profil kommen	6h	
	<u>Geocach-254 (Bug)</u>		
	Email bei registrierung darf nur erlaubte symbole verwenden. Ansonsten korrekte fehlermeldung ausgeben	3h	
	<u>Geocach-262</u>		
	Punkte eines bestimmten Nutzers sollen von eine Admin zurückgesetzt werden	4h	
	<u>Geocach-178</u>		
	Review & Retrospective Sprint 5	2h	
	<u>Geocach-177</u>		
	Daily Scrum Sprint 5	2h 30m	
	<u>Geocach-235</u>		
	Blockwoche	3d	

Timo Volkmann	<u>Geocach-143 (Geocach-12)</u>		24,5
	Progress Stepper	2h	
	<u>Geocach-114 (Geocach-18)</u>		
	Refactoring der Cache erstellen und bearbeiten Logik im Frontend	4h	
	<u>Geocach-160 (Bug)</u>		
	Richtige Fehlermeldungen anzeigen (Für den User aussagekräftig!)	1h 45m	
	<u>Geocach-172 (Bug)</u>		
	Wenn Liste leer, anzeigen dass leer	15m	
	<u>Geocach-260</u>		
	Cache Übersicht: Buttons größer machen	2h	
	<u>Geocach-270 (Bug)</u>		
	Karte sollte ausgeblendet werden, wenn Scanner aktiviert wird (CacheStart, StationView)	15m	
	<u>Geocach-145</u>		
	Frontend Refactoring, Layout optimieren und Bugfixes	2d	
	<u>Geocach-246 (Geocach-15)</u>		
	Einbinden der Karte	4h	
	<u>Geocach-117 (Geocach-12)</u>		
	Karte einbinden mit bereits besuchten Stationen	3h	
	<u>Geocach-255 (Bug)</u>		
	Breitengrad/Längengrad-Textreihfolge ändern bei der Stationserstellung	30m	
	<u>Geocach-173 (Bug)</u>		
	Cacheübersicht: Cache neu starten nicht möglich, wenn bereits gestartet oder beendet	5h	
	<u>Geocach-234 (Bug)</u>		
	switch "deleteCache" to DELETE	10m	
	<u>Geocach-249</u>		
	Sichtbar machen, dass ein User eingeloggt ist	4h	
	<u>Geocach-178</u>		
	Review & Retrospective Sprint 5	2h	
	<u>Geocach-177</u>		
	Daily Scrum Sprint 5	2h 30m	
	<u>Geocach-179</u>		
	Meeting	3h	
	<u>Geocach-235</u>		
	Blockwoche	3d	

Katharina Will	<u>Geocach-237 (Geocach-19)</u>		20
	Darstellung des Standorts auf Karte	1d 3h 20m	
	<u>Geocach-124 (Geocach-12)</u>		
	Karte die den Startpunkt eines gerade gestarteten Cache anzeigt implementieren	1h	
	<u>Geocach-246 (Geocach-15)</u>		
	Einbinden der Karte	2d	
	<u>Geocach-178</u>		
	Review & Retrospective Sprint 5	2h	
	<u>Geocach-177</u>		
	Daily Scrum Sprint 5	2h 30m	
Robin Christ	<u>Geocach-235</u>		3
	Blockwoche	3d	
	<u>Geocach-179</u>		
	Meeting	1h	
	<u>Geocach-204</u>		
	Darstellung des Profils und des Teams überarbeiten	1d 1h	
	<u>Geocach-266 (Geocach-259)</u>		
	Disable "Cache erstellen"-Button, wenn nicht alle notwendigen Felder ausgefüllt sind (ink. Stationen)	1d	
	<u>Geocach-98</u>		
	Eingabe Validierung im Frontend	3h	
	<u>Geocach-178</u>		
	Review & Retrospective Sprint 5	2h	
	<u>Geocach-177</u>		
	Daily Scrum Sprint 5	2h 30m	
	<u>Geocach-235</u>		
	Blockwoche	3d	

Michael Dietz	<u>Geocach-185 (Geocach-18)</u> API aufruf, welcher den bearbeiteten Cache in der Datenbank speichert 6h <u>Geocach-254 (Bug)</u> Email bei registrierung darf nur erlaubte symbole verwenden. Ansonsten korrekte fehlermeldung ausgeben 1h <u>Geocach-162 (Geocach-12)</u> API methode, die alle stationen als POIs zurückliefert die startStationen sind oder bereits vom benutzer gefunden wurden(wird dann vom frontend zur lageplangruppe weitergeleitet) 2h <u>Geocach-248 (Bug)</u> error occurs when someone leaves a team 30m <u>Geocach-256 (Bug)</u> Fehlermeldungen bei der Cacheerstellung verbessern 1h <u>Geocach-267 (Geocach-187)</u> Backend API methode schreiben 1h <u>Geocach-265 (Bug)</u> Teamranglistenpunkte zeigt Teams nicht in richtiger Reihenfolge an 30m <u>Geocach-178</u> Review & Retrospective Sprint 5 2h <u>Geocach-177</u> Daily Scrum Sprint 5 2h 30m <u>Geocach-235</u> Blockwoche 3d	4,6
---------------	--	-----

Teammitglied	Sprint 6 (24.05.2019 - 06.06.2019)	Geloggte Zeit	Story Point Beteiligung
Maximilian Leopold	<u>Geocach-186 (Geocach-12)</u> API Call, der die letzte besuchte Station eines Caches in geeigneter Lageplan Form zurückgibt <u>Geocach-268 (Geocach-187)</u> Button zum Zurücksetzen der Rangliste hinzufügen(nur für Admin sichtbar) <u>Geocach-236</u> Logging im Backend hinzufügen <u>Geocach-258 (Bug)</u> Wenn die Koordinaten der Station nicht im richtigen Format eingegeben werden, muss eine korrekte Fehlermeldung ausgegeben werden <u>Geocach-257 (Bug)</u> Punktwert muss eine ganzzahlige positive Zahl sein, falls etwas anderes eingegeben wird, muss korrekte Fehlermeldung ausgegeben werden <u>Geocach-271</u> Daily Scrums Sprint 6 <u>Geocach-269</u> Review und Retrospective Sprint 6	4h 3h 2h 3h 1h 30m 2h 30m 2h	11,5

Timo Volkmann	<u>Geocach-263</u>		18,5
	Cache zurücksetzen im Frontend (aus bearbeitet löschen)	1h 30m	
	<u>Geocach-273</u>		
	Karte bei Station erstellen/bearbeiten einbinden und Position auswählbar machen	1d 7h	
	<u>Geocach-145</u>		
	Frontend Refactoring, Layout optimieren und Bugfixes	1d	
	<u>Geocach-271</u>		
Katharina Will	Daily Scrums Sprint 6	2h 30m	11
	<u>Geocach-269</u>		
	Review und Retrospective Sprint 6	2h	
	<u>Geocach-124 (Geocach-12)</u>		
	Karte die den Startpunkt eines gerade gestarteten Cache anzeigt implementieren	4h	
	<u>Geocach-271</u>		
	Daily Scrums Sprint 6	2h 30m	
Robin Christ	<u>Geocach-117 (Geocach-12)</u>		2,5
	Karte einbinden mit bereits besuchten Stationen	5h	
	<u>Geocach-269</u>		
	Review und Retrospective Sprint 6	2h	
	<u>Geocach-271</u>		
	Daily Scrums Sprint 6	2h 30m	
	<u>Geocach-269</u>		
Michael Dietz	Review und Retrospective Sprint 6	2h	6,9
	<u>Geocach-264</u>		
	Admin kann andere zum Admin machen	4h	
	<u>Geocach-271</u>		
	Daily Scrums Sprint 6	2h 30m	
	<u>Geocach-269</u>		
	Review und Retrospective Sprint 6	2h	
	<u>Geocach-272 (Bug)</u>		
	Es werden unterschiedliche Positionen in der Rangliste und im Profil angezeigt	3h	
	<u>Geocach-274</u>		
	Benutzeroberflächenfarben vereinheitlichen	3h	
	<u>Geocach-275</u>		
	Profil-Teamranglistentab überarbeiten	1h	
	<u>Geocach-276</u>		
	Startseite überarbeiten(Text sofort anzeigen statt ausklappbar zu machen)	1h	
	<u>Geocach-277</u>		
	Rangliste anpassen	2h	

Teammitglied	Sprint 7 (07.06.2019 - 27.06.2019)	Geloggte Zeit
--------------	------------------------------------	---------------

Maximilian Leopold	<u>Geocach-298 (Dokumentation)</u>	1d
	Anleitung zum öffnen und lokalen Starten des Projekts auf dem localhost	
	<u>Geocach-281 (Dokumentation)</u>	1d
	Installationsanleitung	
	<u>Geocach-299 (Dokumentation)</u>	1d 1h
	API Dokumentation Backend	
	<u>Geocach-292 (Dokumentation)</u>	1d 3h
	Darstellung der Leistung pro Teammitglied pro Sprint aus Jira	
	<u>Geocach-296 (Dokumentation)</u>	1h
	Architekturbeschreibung	
	<u>Geocach-285 (Dokumentation)</u>	3h
	Eingesetzte Pakete	
	<u>Geocach-294 (Dokumentation)</u>	2h
	Testberichte aus Blockwoche	
Timo Volkmann	<u>Geocach-279</u>	2h
	Sprint Review und Retrospective	
	<u>Geocach-278</u>	2h
	Daily Scrum Sprint 7	
	<u>Geocach-296 (Dokumentation)</u>	1d 2h
	Architekturbeschreibung	
	<u>Geocach-285 (Dokumentation)</u>	2h
	Eingesetzte Pakete	
Katharina Will	<u>Geocach-287 (Dokumentation)</u>	3h 30m
	Wichtige Abläufe als Sequenzdiagramm	
	<u>Geocach-279</u>	2h
	Sprint Review und Retrospective	
	<u>Geocach-278</u>	2h
	Daily Scrum Sprint 7	
	<u>Geocach-290 (Dokumentation)</u>	5h
	Projektorganisation	
	<u>Geocach-282 (Dokumentation)</u>	2d
	Bedienungsanleitung	
	<u>Geocach-293 (Dokumentation)</u>	2h
	Dokumente zum Projektmanagement	
	<u>Geocach-286 (Dokumentation)</u>	15m
	Zugangsdaten	
	<u>Geocach-279</u>	2h
	Sprint Review und Retrospective	
	<u>Geocach-278</u>	2h
	Daily Scrum Sprint 7	

Robin Christ	<u>Geocach-294 (Dokumentation)</u>	6h
	Testberichte aus Blockwoche	
	<u>Geocach-280</u>	3d
	Präsentationsvorbereitung	
	<u>Geocach-279</u>	2h
	Sprint Review und Retrospective	
Michael Dietz	<u>Geocach-278</u>	2h
	Daily Scrum Sprint 7	
	<u>Geocach-284 (Dokumentation)</u>	4h
	Datenbankstruktur, SQL Anfragen, Trigger...	
	<u>Geocach-283 (Dokumentation)</u>	2h
	Klassenmodell des Anwendungsgebiets (=Entity-Klassen)	
	<u>Geocach-280</u>	4d
	Präsentationsvorbereitung	
	<u>Geocach-279</u>	2h
	Sprint Review und Retrospective	
	<u>Geocach-278</u>	2h
	Daily Scrum Sprint 7	
	<u>Geocach-304</u>	5m
	Email-Validierung funktioniert nicht richtig	

Datenbankstruktur, SQL-Anfragen, Trigger

Trigger:

Da mitten im Projekt eine zentrale Usermanagementdatenbank eingeführt wurde, wir aber weiterhin den User als Fremdschlüssel in unseren Datenbanktabellen benutzen wollten, mussten wir uns die Tabellen User, Role, Role_User aus der Usermanagementdatenbank in unsere Datenbank rüber kopieren. Für das Kopieren haben wir uns entschieden, da es nicht möglich ist Fremdschlüssel auf Datenbankeinträge einer anderen Datenbank zu haben. Wir mussten deshalb eine konsistente Usertabelle in unserer Datenbank anlegen, die **immer** die selben Einträge wie die Usermanagementdatenbank enthält. Dies wird durch update, insert und delete Trigger umgesetzt, die in den Tabellen der Usermanagementdatenbank untergebracht sind und dafür sorgen, dass jede Änderung auch auf unsere Datenbank übertragen wird, damit die User, Role, Role_User-Tabellen in unserer Datenbank mit denen der Usermanagementdatenbank synchron sind.

Insert-Trigger:

Hier sieht man einen Insert-Trigger in der Usertabelle, welcher dafür sorgt, dass nach jedem neuen Eintrag in der Usertabelle der Usermanagementdatenbank ebenfalls ein neuer Eintrag in der Usertabelle der Geocachingdatenbank erzeugt wird. Außerdem wird in der User_Info-Tabelle der Geocachingdatenbank ebenfalls ein neuer Eintrag generiert, bei welchem ranking_points_sum mit 0 initialisiert wird und der neu erzeugte User als Fremdschlüssel gespeichert wird, damit identifiziert werden kann, zu welchem User die User_Info gehört.

Abb. 1, Insert-Trigger in der Usertabelle in der Usermanagementdatenbank

Update-Trigger:

Im folgenden sieht man einen Update-Trigger in der Usertabelle, welcher dafür sorgt, dass jede Änderung eines Eintrags in der Usertabelle der Usermanagementdatenbank ebenfalls in der Usertabelle der Geocachingdatenbank übernommen wird.

Abb. 2, Update-Trigger in der Usertabelle in der Usermanagementdatenbank

Delete-Trigger:

Zuletzt noch der Delete-Trigger in der Usertabelle, welcher dafür sorgt, dass ein User der in der Usertabelle der Usermanagementdatenbank gelöscht wurde ebenfalls in der Usertabelle der Geocachingdatenbank gelöscht wird. Außerdem werden Einträge aus Tabellen, bei denen der User als Fremdschlüssel agiert ebenfalls aus der Geocachingdatenbank gelöscht.

Abb. 3, Delete-Trigger in der Usertabelle in der Usermanagementdatenbank

Analoge Trigger gibt es in der Role- und Role_User-Tabelle der Usermanagementdatenbank.

Beispieleinträge:

In diesem Abschnitt möchte ich auf einige wichtige Tabellen eingehen und anhand von Beispieldaten erklären, welche Funktionen die einzelnen Tabellen erfüllen.

User_Info:

So sieht ein typischer Eintrag in der User_Infotabelle aus. Durch den User-Fremdschlüssel wird festgelegt, zu welchem User dieser User_Info-Eintrag gehört. In dieser Tabelle werden dem User zusätzliche Attribute gegeben. So erfährt man hier beispielsweise, dass der User mit der ID 34 derzeit 2000 Ranglistenpunkte hat und zu dem Team mit der ID 6047 gehört. Diese Informationen wurden damals in der Usertabelle gespeichert, mussten aber aufgrund der zentralen Usermanagementdatenbank auf eine andere Tabelle verlagert werden.

Abb. 4, Eintrag in der User_Infotabelle in der Geocachingdatenbank

Cache:

Dieser Cache trägt den Namen Wolfszipfel. Durch die Beschreibung "Suche den Aussichtspunkt und genieße die Aussicht" wird gezeigt, um was es bei diesem Cache geht. Außerdem wird gespeichert, dass das Abschließen dieses Caches 341 Punkte gibt und man die Belohnung mit der ID 0 erhält.

Abb. 5, Eintrag in der Cachetabelle in der Geocachingdatenbank

Cache Stationen:

In dieser Tabelle wird festgehalten, welche Stationen zu einem Cache gehören. Zu dem oben beschriebenen Cache "Wolfszipfel" mit der ID 8 gehören die Stationen mit der ID 6 und 7.

Abb. 6, Einträge in der Cache_Stations-tabelle in der Geocachingdatenbank

Station:

Eine der Stationen die zu dem Cache namens Wolfszipfel gehört ist hier zu sehen. Sie hat eine Beschreibung, bzw. ein Rätsel welches man lösen muss um die Nachfolgestation zu finden. Außerdem werden hier die Koordinaten gespeichert an dem sich die Station befindet. Die Station wird vor Ort durch einen QR-Code repräsentiert. Diesen muss man einscannen und erhält dann das Rätsel.

Abb. 7, Eintrag in der Stationstabelle in der Geocachingdatenbank

Bearbeitet:

In der Bearbeitetabelle wird der Fortschritt eines Users bezüglich eines Caches gespeichert. Nachdem der User einen Cache gestartet hat wird ein Eintrag in dieser Tabelle generiert. In diesem Beispiel, hat der User mit der ID 237 den Cache mit der ID 1099 begonnen. Dass er diesen begonnen und noch nicht abgeschlossen hat, sieht man an der Cache_Acces_Definition mit der ID 0, welche für "angefangen" steht. Die zuletzt besuchte Station, die der User während des Durchführen des Caches besucht hat war die Station mit der ID 1093.

Abb. 8, Eintrag in der Bearbeitetabelle in der Geocachingdatenbank

Cache Acces Definition:

Die Cache_Acces_Definitiontabelle hat lediglich zwei Einträge. Diese werden für die Bearbeitetabelle benötigt, um anzuzeigen ob der User einen Cache bereits abgeschlossen hat oder nicht.

Abb. 9, Eintrag in der Cache_Acces_Definitiontabelle in der Geocachingdatenbank

SQL-Anfragen:

Wie im Kapitel "Klassendiagramm" beschrieben und erklärt verwenden wir SQL-Anfragen, die hier aufgezeigt und erklärt werden.

Rangliste:

Eine SQL-Anfrage kümmert sich um die Ranglistenbestimmung. Mit Hilfe der Ranglistenpunkte welche in der User_Infotabelle gespeichert werden lässt sich die Rangliste leicht bestimmen, indem man die Einträge nach den Ranglistenpunkten sortiert.

Abb. 10, SQL-Anfrage zur Bestimmung der Rangliste

Ranglistenplatz:

Die vorherige SQL-Anfrage wird zur Hilfe genommen um den Ranglistenplatz eines Nutzers, repräsentiert durch die Email, zu bestimmen. Dazu werden einfach die Zeilen der generierten Ranglistentabelle gezählt, bis der User mit der Email auftaucht. Die Zeilennummer entspricht dann dem Ranglistenplatz.

Abb. 11, SQL-Anfrage zur Bestimmung der Ranglistenposition eines Nutzers

Teamrangliste:

Die Teamrangliste wird ähnlich wie die normale Rangliste bestimmt. Zusätzlich werden aber alle Nutzer die zum selben Team gehören gruppiert und die Summe ihrer Ranglistenpunkte gebildet.

Abb. 12, SQL-Anfrage zur Bestimmung der Teamrangliste

Teammitglieder:

Mit dieser SQL-Anfrage lassen sich alle User bestimmen, die zu einem gewissen Team, repräsentiert durch seinen Namen, gehören.

Abb. 13, SQL-Anfrage zur Bestimmung der Teammitglieder

Projektorganisation

In diesem Abschnitt sind alle Protokolle zu Daily Scrums, Sprint Reviews und Sprint Retrospectives zu finden.

Weitere Informationen zu den jeweiligen Unterabschnitten sind in den Ordnern "Daily Scrums", "Sprint Reviews" und "Retrospectives".

Im Abschnitt "Innere Organisation" handelt von der Vorgehensweise innerhalb des Teams, z.B. Aufgabenteilung, Vorgehen bei Auftreten von Problemen.

[Daily Scrums](#)

[Projektnormen und -spielregeln](#)

[Sprint Reviews](#)

[Retrospectives](#)

Daily Scrums

[Create meeting note](#)

Hier finden Sie die Protokolle aller stattgefundenen Daily Scrum-Treffen.

Daily Scrums fanden montags und donnerstags, nach dem ersten Release sogar täglich, statt. Ausnahmen gab es lediglich an Feiertagen und in den Ferien, in denen keine Daily Scrums durchgeführt wurden.

Die Treffen fanden räumlich in der Regel in A106, A212 oder über Discord statt. Zeitlich nahmen die Treffen nie mehr als 15 Minuten in Anspruch und fanden i.d.R. um 13 Uhr statt, bzw. zum nächstmöglichen Zeitpunkt, wenn die Vorlesung geendet hat.

Moderiert und protokolliert wurden die Treffen stets vom Scrum Master Katharina Will.

Incomplete tasks from meetings

Aufgabenbericht

Das sieht gut aus! Keine unvollständigen Aufgaben.

All meeting notes

Titel	Ersteller	Geändert
2019-06-26 Daily Scrum 52	Katharina Will	vor etwa 3 Stunden
2019-06-25 Daily Scrum 51	Katharina Will	gestern um 1:42 PM
2019-06-24 Daily Scrum 50	Katharina Will	Jun 24, 2019
2019-06-21 Daily Scrum 49	Katharina Will	Jun 21, 2019
2019-06-18 Daily Scrum 47	Katharina Will	Jun 21, 2019
2019-06-19 Daily Scrum 48	Katharina Will	Jun 21, 2019
2019-04-15 Daily Scrum 14	Katharina Will	Jun 21, 2019
2019-06-17 Daily Scrum 46	Katharina Will	Jun 17, 2019
2019-06-06 Daily Scrum 45	Katharina Will	Jun 17, 2019
2019-06-04 Daily Scrum 43	Katharina Will	Jun 06, 2019
2019-06-05 Daily Scrum 44	Katharina Will	Jun 06, 2019
2019-06-03 Daily Scrum 42	Katharina Will	Jun 06, 2019
2019-05-29 Daily Scrum 40	Katharina Will	Jun 03, 2019
2019-05-31 Daily Scrum 41	Katharina Will	Jun 03, 2019
2019-05-28 Daily Scrum 39	Katharina Will	Jun 03, 2019
2019-05-27 Daily Scrum 38	Katharina Will	Mai 27, 2019
2019-05-24 Daily Scrum 37	Katharina Will	Mai 27, 2019
2019-05-23 Daily Scrum 36	Katharina Will	Mai 23, 2019
2019-05-21 Daily Scrum 34	Katharina Will	Mai 23, 2019
2019-05-22 Daily Scrum 35	Katharina Will	Mai 23, 2019

[Weitere Ergebnisse suchen](#)

2019-03-14 Daily Scrum 1

Datum

14.03.2019

Von: 15:55 Uhr Bis: 16:10 Uhr

Ort: Mensa

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Robin Christ
- Michael Dietz

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Prototyp für die Registrierung erstellt	In Backend-Technologien einarbeiten: Tomcat und Java Spring	Viel Lernaufwand, da keine Kenntnisse vorhanden sind
Maximilian Leopold	Prototyp für die Einsicht in die Rangliste erstellt In Product Backlog eingearbeitet	Einarbeitung in Tomcat und Java Spring	Viel Lernaufwand, da noch keine Kenntnisse mit Backend vorhanden sind
Timo Volkmann	Prototyp für die Profilübersicht erstellt	Einarbeitung in vue.js	Keine
Robin Christ	Prototyp zum Erstellen eines Caches erstellt In Product Backlog eingearbeitet	In Backend-Technologien einarbeiten: Tomcat und Java Spring	Viel Lernaufwand, da keine Kenntnisse vorhanden sind
Katharina Will	Prototyp für die Cacheübersicht erstellt	Jira mit vorhandenen Daten füllen Über vue.js informieren & lernen	Viel Lernaufwand, da keine Kenntnisse vorhanden sind

Sonstiges:

- ☒ Gruppe in Scrum-Rollen eingeteilt

Bemerkungen für nächstes Treffen:

Datum: 18.03.2019

- ☒ Projektauftrag aufteilen

2019-03-18 Daily Scrum 2

Datum

18.03.2019

Von: 13:10 Uhr Bis: 13:30 Uhr

Ort: A106

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Robin Christ
- Michael Dietz

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Mit Tomcat und Spring auseinandergesetzt	Kapitel 2 des Projektauftrags schreiben	Keine
Maximilian Leopold	Mit Tomcat und Spring boot beschäftigt Bitbucket Projekt gestartet	Kapitel 1 des Projektauftrags schreiben	Wenig Zeit bis zur geplanten Deadline für den Projektauftrag (21.03.2019)
Timo Volkmann	Mit vue.js beschäftigt Über fertige Pakete informiert, die man für unser Projekt verwenden kann	Kapitel 4 des Projektauftrags schreiben Demoprojekt aufsetzen und in Bitbucket einstellen	Keine
Robin Christ	Über Spring boot informiert	Kapitel 5 des Projektauftrags schreiben Demoprojekt aufsetzen	Keine
Katharina Will	Jira aktualisiert HTML & vue.js Tutorial bearbeitet	Kapitel 3 des Projektauftrags schreiben Daily Scrum-Protokolle in Confluence einpflegen Weiter mit vue.js auseinandersetzen	Wenig Zeit bis zur geplanten Deadline (21.03.2019)

Sonstiges:

- ☒ Kapitel des Projektauftrags aufgeteilt
- ☒ René Schiesswohl wurde Kapitel 6 des Projektauftrags zugeteilt

Bemerkungen für nächstes Treffen:

Datum: 21.03.2019

- ☒ Projektauftrag sollte bis 21.03.2019 fertig sein

2019-03-21 Daily Scrum 3

Datum

21.03.2019

Von: 13:20 Uhr Bis: 13:35 Uhr

Ort: A106

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Robin Christ
- Michael Dietz

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Kapitel 2 des Projektauftrags geschrieben Mit Spring auseinandergesetzt	Entity Klassen editieren	Sehr knapp die User Stories umzusetzen
Maximilian Leopold	Kapitel 1 des Projektauftrags geschrieben Mit JPA und Datenbanken beschäftigt	Bis Samstag Datenbank neu aufsetzen (JPA Entity Klassen neu schreiben) Überlegen, wie wir die API aufbauen werden	Zeitlich sehr knapp eine Präsentation und die Datenbank vorzubereiten
Timo Volkmann	Kapitel 4 des Projektauftrags geschrieben Gradle konfiguriert, um Frontend einzubinden Mit Vue weiter beschäftigt Vue aufgesetzt	Präsentation vorbereiten Frontend-Teil der User Stories bearbeiten Mit Axios beschäftigen	Sehr enger Zeitplan Vermutlich werden User Stories nicht zufriedenstellend fertig
Robin Christ	Kapitel 5 des Projektauftrags geschrieben Mit Java Spring auseinandergesetzt Projekt aus Bitbucket	Präsentation vorbereiten Projektauftrag überarbeiten	Unsicherheit bei der Präsentation über anzusprechende Themen
Katharina Will	Kapitel 3 des Projektauftrags geschrieben Daily Scrum Protokolle in Confluence eingepflegt Vue.js und JavaScript Tutorials bearbeitet	Vue.js weiterstudieren Erstellung der Präsentation assistieren	Sehr wenig Zeit bis zum ersten Release

Sonstiges:

- ☒ Timo Volkmann und Robin Christ werden die erste Release-Präsentation durchführen

Bemerkungen für nächstes Treffen:

Datum: 25.03.2019

2019-03-25 Daily Scrum 4

Datum

25.03.2019

Von: 13:45 Uhr Bis: 13:55 Uhr

Ort: A106

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Robin Christ
- Michael Dietz

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Restlichen Entity Klassen angelegt Für Cache starten eine Backend-Methode angelegt	Daten in der Datenbank mit sinnvollen Daten füllen	Knappe Zeit
Maximilian Leopold	Java Spring Security Login Funktion implementiert Datenbank umgebaut	Implementierung der Login Funktion mit Token	In 3 Tagen ist Release, sehr knapp, aber Backend steht.
Timo Volkmann	Präsentation angefertigt Login implementieren Tutorials bearbeitet	Layout und Login fertig machen	Schwierigkeiten mit der Authentifizierung
Robin Christ	Präsentation fertig gemacht (bis auf die Gliederung)	Auf die Präsentation vorbereiten An der Datenbank arbeiten	Noch keine Ahnung, wie Änderungen an der Datenbank umgesetzt wird
Katharina Will	Frontend für Login angefertigt, für Cache-Übersicht angefangen	Layout für die Cache-Übersicht überarbeiten	Großer Zeitdruck

Sonstiges:

Bemerkungen für nächstes Treffen:

Datum: 28.03.2019

2019-03-28 Daily Scrum 5

Datum

28.03.2019

Von: 15:10 Uhr Bis: 15:25 Uhr

Ort: A106

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Robin Christ
- Michael Dietz

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Datenbank mehrmals überarbeitet und mit Daten gefüllt API-Methoden überarbeitet, damit sie mit Token funktionieren Logout-Methode implementiert	In Confluence die API-Methoden überarbeiten Auf Abruf	Keine mehr
Maximilian Leopold	Projektauftrag geschrieben Product Backlog überarbeitet API Dokumentation geschrieben	Als Cacher will ich meine angefangenen und abgeschlossenen Caches sehen können Rangliste einsehen	Java Security wird nicht ganz verstanden und kann nicht implementiert werden
Timo Volkmann	Präsentation fertig gestellt Letzte Änderungen am Frontend gemacht Kleine Anleitung für Frontend Entwicklung erstellt (in Confluence)	Detailansicht für einen Cache anfertigen	Keine
Robin Christ	Präsentation fertig gestellt, geübt	Wieder einarbeiten in aktuellen Backend-Stand Nächste User Story anfangen	Einarbeitung in Backend benötigt
Katharina Will	Cache-Übersicht Layout erstellt Layout der Startseite fertig gestellt	Layout für Ansicht eines Caches anfertigen Layout für Rangliste erstellen	Keine

Bemerkungen für nächstes Treffen:

Datum: 01.04.2019

2019-04-01 Daily Scrum 6

Datum

01.04.2019

Von: 13:10 Uhr Bis: 13:20 Uhr

Ort: A106

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann

Abwesend:

- Rene Schiesswohl
- Robin Christ
- Michael Dietz

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Maximilian Leopold	Product Backlog final überarbeitet Zugewiesene User Stories bearbeitet Mit named queries und queries in JPA beschäftigt	mit Spring Security beschäftigen	Sind nur 5 Leute, statt 6 seit 1,5 Sprints
Timo Volkmann	In Jira User Story Unteraufgaben erstellt	Die ersten User Stories anfangen Layout fürs Frontend festlegen	Keine
Katharina Will	Product Backlog final überarbeitet	Layout für Ansicht eines Caches anfertigen Layout für Rangliste erstellen	Keine

Sonstiges:

- ☒ Robin Christ wird für das Frontend eingesetzt, um Ausgeglichenheit im Frontend und Backend zu schaffen

Bemerkungen für nächstes Treffen:

Datum: 04.04.2019

2019-04-04 Daily Scrum 7

Datum

04.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: A212

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Robin Christ
- Michael Dietz
- Rene Schiesswohl

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Backend einige User Stories implementiert (mit Admin-Funktionen) mit http-Responses auseinandergesetzt und teilweise implementiert	Restliche Methoden überarbeiten, damit auch da die http-Responses funktionieren Dokumentation überarbeiten	Keine
Maximilian Leopold	Mit Java Spring Security auseinandergesetzt	Restliche Methoden überarbeiten, damit auch da die http-Responses funktionieren Dokumentation von den API-Aufrufen überarbeiten	Keine
Timo Volkmann	Letzte User Story des ersten Sprints bearbeitet Layout für die Ansicht der Stationen angefangen	Cache erstellen und deren Problematik bearbeiten	Viel Logik im Frontend wird versucht zu vermeiden, genaue Umsetzung noch unklar
Robin Christ	Vue.js angeschaut	Übungsprojekt mit Vue.js und Quasar erstellen	Keine
Katharina Will	Layout für die Rangliste angefertigt	Dokumentation überarbeiten	Keine
Rene Schiesswohl	In html und vue.js eingearbeitet	Quasar anschauen Übungsprojekt mit Vue.js und Quasar erstellen	Keine

Bemerkungen für nächstes Treffen:

Datum: 05.04.2019

- ☒ Daily Scrums finden ab sofort 3-mal wöchentlich statt: montags, mittwochs und freitags

2019-04-05 Daily Scrum 8

Datum

05.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Robin Christ
- Michael Dietz
- Rene Schiesswohl

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Methode so überarbeitet, dass sie einen http Response zurückgeben API Methoden auf Confluence dokumentiert createCache überarbeitet startStation als Attribut von Cache entfernt	createCache überarbeiten bzw. optimieren nach Absprache mit Max Confluence updaten	Keine
Maximilian Leopold	API Aufrufe im Backend überarbeitet API Dokumentation in Confluence überarbeitet	Mit anderen Gruppen über Umsetzung der gemeinsamen Datenbank für User sprechen Security Aspekte anschauen API Dokumentation weiterschreiben	Eine gemeinsame Datenbank für alle User könnte schwierig anzulegen sein, da alle Gruppen andere Sicherheitsmechanismen und Implementierung haben
Timo Volkmann	Cache und Station erstellen Ansichten und Frontendlogik angefangen	Cache und Station erstellen weiter bearbeiten	Keine
Robin Christ	Projektauftrag überarbeitet	Umsetzung von "Als Cachender will ich angefangene/beendete Caches sehen" angefangen	Erstes Arbeiten mit vue.js, keine Erfahrung bisher
Katharina Will	Projektauftrag überarbeitet An Rangliste ansehen weiter gearbeitet	An Rangliste ansehen weiter arbeiten	Keine
Rene Schiesswohl	Projektauftrag überarbeitet	Tests mit vue.js	Verständnisprobleme, da das erste Mal mit vue.js gearbeitet wird

Bemerkungen für nächstes Treffen:

Datum: 08.04.2019

☒ Daily Scrums finden ab sofort täglich statt

2019-04-08 Daily Scrum 9

Datum

08.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Rene Schiesswohl

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	In der Datenbank Attribute angepasst in JWT eingelesen finale Version von createCache fertiggestellt	Details von User Stories im Product Backlog überarbeiten	Sprint geht nur noch ein paar Tage und es sind noch einige Subtasks nicht umgesetzt
Maximilian Leopold	Token neu implementiert mit geprüften Sicherheitskonzepten und plattformübergreifend Akzeptanzkriterien in Jira ergänzt Akzeptanzkriterien im Product Backlog neu geschrieben	Die neue Implementierung der Token mit Develop mergen und anpassen Mit Frontend über neue Implementierung sprechen	Sprint ist bald zuende und es gibt noch ein paar Aufgaben zu erledigen
Timo Volkmann	Frontend-Logik für Cacheerstellung inkl. Stationen	Frontend-Logik fertig Zentrale und dynamische Konstante für API-URL User Objekt inkl. Logik in vuex Store umziehen Adminabfrage implementieren	Keine
Katharina Will	An Rangliste ansehen weiter gearbeitet	An Rangliste ansehen weiter arbeiten	Keine
Rene Schiesswohl	Einarbeitung in vue.js und quasar Framework	Tasks von User Stories im Product Backlog formulieren Evtl erste Übernahme einer User Story	Wenig Restzeit im aktuellen Sprint

Bemerkungen für nächstes Treffen:

Datum: 09.04.2019

2019-04-09 Daily Scrum 10

Datum

09.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz

Abwesend:

- Rene Schiesswohl
- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Dokumentation in Confluence bearbeitet	Details in User Stories im Product Backlog überarbeiten	Keine
Maximilian Leopold	Alle Änderungen in Develop gemerged An Meeting mit anderen Gruppen bzgl Nutzerverwaltung, Blumendaten und Kartendaten teilgenommen Bug mit Spring Security gelöst	Bearbeitet-Tabellen bearbeiten	Keine
Timo Volkmann	Login angepasst, aufgrund vorgenommener Änderungen Cache erstellen funktioniert	Fehler ausbessern	Keine
Katharina Will	An Rangliste ansehen weiter gearbeitet An Meeting mit anderen Gruppen bzgl Nutzerverwaltung, Blumendaten und Kartendaten teilgenommen	Rangliste ansehen fertigstellen Projektauftrag final überarbeiten & abgeben	Keine

Bemerkungen für nächstes Treffen:

Datum: 10.04.2019

2019-04-10 Daily Scrum 11

Datum

10.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Rene Schiesswohl
- Robin Christ

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	User Story Details im Product Backlog überarbeitet	Eigentlich nichts mehr im Backend zu erledigen	Keine
Maximilian Leopold	Die startCache-Methode überarbeitet, sodass nicht jedes Mal ein neuer Eintrag hinzugefügt wird, wenn schon einer existiert	API Dokumentation überarbeiten	Eine User Story fehlt noch. Morgen ist aber Ende des zweiten Sprints
Timo Volkmann	User Stories fertig gestellt	Plan machen, was für den dritten Sprint gemacht werden muss	Verteilung der Aufgaben
Katharina Will	Projektauftrag final überarbeitet und abgegeben Rangliste ansehen fertig gestellt	Sprint Review und Sprint Retrospective vorbereiten	Keine
Rene Schiesswohl	Tasks im Product Backlog ergänzt	Erste Schritte mit User Story	Sprint endet morgen
Robin Christ	User Story Angefangene Caches einsehen bearbeitet	Arbeit weiterführen	Kleines Problem mit Axios Request mit einem gespeicherten Token im Browser als Parameter

Bemerkungen für nächstes Treffen:

Datum: 11.04.2019

2019-04-11 Daily Scrum 12

Datum

11.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Robin Christ

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Bugs in den API Methoden gefixed	Subtasks anschauen und aufteilen	Keine
Maximilian Leopold	API Dokumentation überarbeitet	User Stories vom dritten Sprint anschauen, Subtasks Aufteilen der User Stories mit Michael API Rückgabewerte mit Frontend absprechen	Keine, alle User Stories fertig
Timo Volkmann	Robin bei der Entwicklung unterstützt	Subtasks anschauen und aufteilen	Keine
Katharina Will	Sprint Review und Sprint Retrospective vorbereitet	User Stories, Subtasks anschauen und aufteilen	Keine
Robin Christ	User Story angefangene/beendete Caches einsehen fertig gestellt	-	Bisher noch keine Aufgabe

Bemerkungen für nächstes Treffen:

Datum: 12.04.2019

2019-04-12 Daily Scrum 13

Datum

12.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Rene Schiesswohl
- Michael Dietz
- Robin Christ

Abwesend:

- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	-	Erste Subtasks umsetzen	Keine
Maximilian Leopold	-	Subtasks anschauen und zuteilen, anfangen zu bearbeiten	Keine
Rene Schiesswohl	-	Mit Timo abklären, welche User Story bearbeitet werden soll	Keine
Katharina Will	-	Aufgaben aufteilen	Keine
Robin Christ	-	Aufgabe suchen	Keine

Bemerkungen für nächstes Treffen:

Datum: 15.04.2019

2019-04-15 Daily Scrum 14

Datum

15.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Robin Christ

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	checkStation implementiert (API Methode zum Durchführen eines Cache) Mit neuen Datenbankgegebenheiten beschäftigt	checkStation verbessern /dokumentieren	Probleme in der Datenbank, da die User in einer anderen Datenbank liegen und wir eigentlich ihre foreign keys verwenden müssen, das aber so jetzt nicht mehr können
Maximilian Leopold	Angefangen die neue Userverwaltung zu implementieren Struktur im Backend umgestellt, sodass sie besser lesbar ist checkStation peer-reviewed	Userverwaltung fertig implementieren Trigger schreiben, sobald Berechtigung dafür erteilt wird	Keine Berechtigung Trigger in der Userverwaltung zu schreiben. Email wurde verfasst und Berechtigung wird hoffentlich heute erteilt
Timo Volkmann	Tasks gesammelt, die heute zu verteilen sind	Tasks, die zugewiesen wurden, bearbeiten	Keine
Katharina Will	-	Tasks, die zugewiesen wurden, bearbeiten	Keine
Robin Christ	-	"Fehlermeldungen ausgeben" bearbeiten	Noch nicht damit gearbeitet, evtl. erst Informationen sammeln

Sonstiges:

- ☒ Lösungsfeld bei der Aufgabenstellung wird entfernt, da Rätsel zur Position zur nächsten Station gestellt werden und keine verschiedenen Aufgabentypen

Bemerkungen für nächstes Treffen:

Datum: 16.04.2019

2019-04-16 Daily Scrum 15

Datum

16.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Rene Schiesswohl

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	checkStation Methode Rückgabewert verändert und vereinfacht Mit der Usermanagementimplementierung beschäftigt	Mit Herrn Keller besprechen, ob man die Usernamen unique machen kann, weil das für die Rangliste von Vorteil wäre Dokumentation in Confluence updaten	Gemeinsame Usermanagementimplementierung anders als die eigene. Daher müssen viele Dinge, die bereits in früheren Sprints erledigt wurden, neu überarbeiten
Maximilian Leopold	Backend weiter umorganisiert mit Hilfsklassen, um Konsistenz in den Rückgaben zu erzeugen Herrn Permantier eine Mail geschrieben, wegen der neuen Userverwaltung Mit der neuen Userverwaltung auseinandergesetzt	Auf Herrn Permantiers Antwort warten Sobald die Rechte vorhanden sind, unsere Datenbank weiter an die neue Userverwaltung anpassen	Immer noch keine Rechte von Herrn Permantier erhalten
Timo Volkmann	Zentrale Stelle geschaffen für URL Erste Versuche mit QR-Code-Scanner Implementierung	Alle Aufrufe anpassen, sodass diese BaseURL verwenden Weitere Tests mit QR-Code-Scanner	Schlechte Testbarkeit von Scanner auf iOS Geräten, da https benötigt wird
Katharina Will	Angefangen Seite anzufertigen, die das aktuelle Rätsel darstellt	An Subtask weiterarbeiten	Keine
Rene Schiesswohl	Erste Schritte bei der Registrierung eines Users	Weiterführung der User Story	Werden sich noch ergeben, bisher noch keine vorhanden

Bemerkungen für nächstes Treffen:

Datum: 17.04.2019

2019-04-17 Daily Scrum 16

Datum

17.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Michael Dietz

Abwesend:

- Robin Christ
- Rene Schiesswohl
- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Datenbank angepasst, Trigger modifiziert für user_info Einige API Methoden neu angepasst checkStation soweit fertig implementiert	Rückgabewerte von API-Methoden bearbeiten, falls Frontend dies wünscht	Keine
Maximilian Leopold	Datenbank Trigger geschrieben Datenbank repariert Datenbank neu aufgesetzt mit korrekten Attributen und Tabellen an Userverwaltung angepasst API Methoden im Backend angepasst	Letzte Fehler im neuen User Management beseitigen Funktioniert alles? Rückgabewerte anpassen, je nachdem welche Formate das Frontend braucht	Keine
Katharina Will	Seite angefertigt, die das aktuelle Rätsel darstellt	An Subtask weiterarbeiten	Keine

Bemerkungen für nächstes Treffen:

Datum: 18.04.2019

2019-04-18 Daily Scrum 17

Datum

18.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: A212

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz

Abwesend:

- Robin Christ
- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Die Suche nach einem User mit Max geändert Weiter Datenbank umstrukturiert, fertig mit Userverwaltung Ranglisten API-Methode geändert, sodass nur noch eigene User angezeigt werden, nicht alle	API Dokumentation in Confluence updaten	Datenbank Umstrukturierung sehr zeitaufwendig
Maximilian Leopold	Die Suche nach einem User mit Michael geändert Weiter Datenbank umstrukturiert, fertig mit Userverwaltung	Schauen, ob alles funktioniert; Bugs beheben	Info zu kurzfristig, dass Datenbank wieder umstrukturiert werden muss
Timo Volkmann	Probleme mit der Anmeldung behoben URL für die API im Frontend zentral hinterlegt	QR-Code scannen weiter bearbeiten User API Feinschliff	Keine
Katharina Will	Seite anfertigen, die das aktuelle Rätsel darstellt weiter bearbeiten	Subtask fertigstellen	Keine

Bemerkungen für nächstes Treffen:

Datum: 23.04.2019

2019-04-23 Daily Scrum 18

Datum

23.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: F229

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Robin Christ

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	API Dokumentation in Confluence überarbeitet Kleine Fehler behoben in der Datenbank und in den API Methoden	Auf Abruf, falls im Frontend Probleme zu API Abrufen auftreten	Server wird ab 16 Uhr abgeschaltet
Maximilian Leopold	Präsentation angefangen vorzubereiten Backend Code reorganisiert Neue Methode, dass man den Rang eines bestimmten Users erhält	Auf Präsentation fokussieren Auf Abruf, falls im Frontend Probleme zu API Abrufen auftreten	Server wird ab 16 Uhr abgeschaltet
Timo Volkmann	WR-Code Scanner fertig implementiert	User Story zum Cachen fertig stellen	Server wird ab 16 Uhr abgeschaltet
Katharina Will	Präsentation angefangen vorzubereiten An der Start und Endseite zum Cachen gearbeitet	Start und Endseite zum Cachen fertig stellen	Zeit wird knapp, keine der beiden User Stories fertig Server wird ab 16 Uhr abgeschaltet
Robin Christ	Fehlermeldungen anzeigen angefangen Elemente für die Registrierung angefangen	Registrierung fertigstellen	Server wird ab 16 Uhr abgeschaltet

Bemerkungen für nächstes Treffen:

Datum: 24.04.2019

2019-04-24 Daily Scrum 19

Datum

24.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Michael Dietz
- Robin Christ

Abwesend:

- Rene Schiesswohl
- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	startCache und checkStation vereinheitlicht	-	Keine
Maximilian Leopold	Präsentationsvorbereitung Änderungen von gestern durchgelesen	Auf Präsentation fokussieren Gespräch mit den Professoren vorbereiten	Keine
Katharina Will	Präsentationsvorbereitung An der Start und Endseite zum Cachen gearbeitet	Präsentation vorbereiten Sprint Review, Retrospective und Gespräch mit Professoren vorbereiten	Keine
Robin Christ	Registrieren fast komplett abgeschlossen	Fehlenden Subtask "User informieren, falls Registrierung erfolgreich" abschließen	User Story zur Registrierung wird zeitlich nur sehr knapp fertig

Bemerkungen für nächstes Treffen:

Datum: 25.04.2019

2019-04-25 Daily Scrum 20

Datum

25.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: A212

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Robin Christ

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	-	User Stories anschauen und aufteilen Subtasks abarbeiten	Server fast dauerhaft offline, testen nicht möglich
Maximilian Leopold	Präsentation vorbereitet	Präsentation vorbereiten User Stories anschauen und mit Michael aufteilen	Server fast dauerhaft offline, testen nicht möglich
Timo Volkmann	Scanner in den Ablauf implementiert Änderungen am Cache Erstellen	User Stories anschauen und aufteilen	Server fast dauerhaft offline, testen nicht möglich
Katharina Will	Präsentation vorbereitet	Präsentation vorbereiten User Stories anschauen und aufteilen	Server fast dauerhaft offline, testen nicht möglich
Robin Christ	-	Registrierung fertigstellen	Server fast dauerhaft offline, testen nicht möglich

Bemerkungen für nächstes Treffen:

Datum: 26.04.2019

2019-04-26 Daily Scrum 21

Datum

26.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Robin Christ

Abwesend:

- Rene Schiesswohl

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	createTeam und joinTeam initial implementiert	Weiter an team-Methoden arbeiten	Server immer noch offline, testen nicht möglich
Maximilian Leopold	Rangliste wird wieder korrekt angezeigt Präsentationsvorbereitung	Präsentation vorbereiten	Die se-Server sind wieder offline (Usermanagement) Server immer noch offline, testen nicht möglich
Timo Volkmann	Frontend Bugs und To-Do's sammeln	Prüfen, ob alles funktioniert, ggf fixen	Server immer noch offline, testen nicht möglich
Katharina Will	Präsentationsvorbereitung	Präsentation vorbereiten	Server immer noch offline, testen nicht möglich Live Demo für die Präsentation kann nicht geübt werden
Robin Christ	Layout zur Profil- und Teamansicht angelegt	Design davon überarbeiten und Funktionen dafür ausarbeiten	Server immer noch offline, testen nicht möglich

Bemerkungen für nächstes Treffen:

Datum: 29.04.2019

2019-04-29 Daily Scrum 22

Datum

29.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Michael Dietz
- Robin Christ

Abwesend:

- Rene Schiesswohl
- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Folgende API Methoden implementiert, inklusive nötiger Datenbankänderungen: <ul style="list-style-type: none">▪ getTeamRankingList▪ leaveTeam▪ sendTeamInvite▪ getMyTeamInvites▪ deleteTeamInvite▪ setTeamStatus	Mögliche Fehler finden und entfernen	Server immer noch offline, testen nicht möglich
Maximilian Leopold	Präsentationsvorbereitung	Präsentation vorbereiten Letzte Bugfixes, damit alles zur Präsentation funktioniert	Es funktioniert noch nicht alles für die Präsentation
Katharina Will	Präsentationsvorbereitung	Präsentation vorbereiten	Es funktioniert noch nicht alles für die Präsentation Live Demo für die Präsentation kann kaum geübt werden, weil Server unbeständig
Robin Christ	Layout zur Profil- und Teamansicht (Mockup Version) angelegt	Logik hinter Profil- und Teamkomponente implementieren	Server offline, testen nicht möglich

Bemerkungen für nächstes Treffen:

Datum: 30.04.2019

2019-04-30 Daily Scrum 23

Datum

30.04.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: F229

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Timo Volkmann

Abwesend:

- Rene Schiesswohl
- Michael Dietz

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Maximilian Leopold	Präsentationsvorbereitung	Präsentation vorbereiten Letzte Bugfixes, damit alles zur Präsentation funktioniert	Keine
Timo Volkmann	Frontend Bugfixes Layout Optimierung Code Refactoring	Frontend Bugfixes Layout Optimierung Code Refactoring	Keine
Katharina Will	Präsentationsvorbereitung	Präsentation vorbereiten	Live Demo für die Präsentation kann kaum geübt werden, weil Server unbeständig
Robin Christ	An Logik von Profil- und Teamansicht weitergearbeitet	Logik fertig implementieren	Keine

Bemerkungen für nächstes Treffen:

Datum: 02.05.2019

2019-05-02 Daily Scrum 24

Datum

02.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Timo Volkmann
- Michael Dietz

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Bugfixes Confluence Dokumentation aktualisiert	Alles auf Korrektheit prüfen	Keine
Maximilian Leopold	Präsentationsvorbereitung	Bugfixes Diverse Backend Methoden schreiben Kooperation mit Lageplan-Gruppe testen	Keine
Timo Volkmann	Bugfixes	Weitere Verbesserungen anhand von Feedback der Präsentation	Keine
Katharina Will	Präsentationsvorbereitung	-	Keine
Robin Christ	Weitere Arbeit an Logik der sozialen Komponenten	Arbeit abschließen	Keine

Bemerkungen für nächstes Treffen:

Datum: 03.05.2019

2019-05-03 Daily Scrum 25

Datum

03.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Michael Dietz

Abwesend:

- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Bugfixes	Alles auf Korrektheit prüfen	Keine
Maximilian Leopold	Angefangen Dokumentation auf Swagger umzustellen	Dokumentation abschließen Schauen, was noch zu tun ist	Keine
Katharina Will	-	Subtasks von User Stories bearbeiten	Keine
Robin Christ	-	Arbeit an sozialen Komponenten abschließen	Keine

Bemerkungen für nächstes Treffen:

Datum: 06.05.2019

2019-05-06 Daily Scrum 26

Datum

06.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: A106

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Michael Dietz
- Timo Volkmann

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Karten-POI Methode implementiert	Auf Abruf	Keine
Maximilian Leopold	Dokumentation umgeschrieben Bugs in Jira eingefügt	Auf Abruf	Keine
Timo Volkmann	Bugfixes	Bugfixes Verbesserungen anhand des Präsentationsfeedbacks	Keine
Katharina Will	Team-Rangliste angefangen zu implementieren	Team-Rangliste implementieren	Keine

Bemerkungen für nächstes Treffen:

Datum: 07.05.2019

2019-05-08 Daily Scrum 27

Datum

08.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Michael Dietz
- Timo Volkmann
- Robin Christ

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	-	Auf Abruf	Keine
Maximilian Leopold	Verschiedene Bugfixes und kleine Verbesserungen	Auf Abruf	Keine
Timo Volkmann	Bugfixes und Verbesserungen QR-Code Generator angefangen	Verbesserungen an der Fehlerbehandlung	Keine
Katharina Will	Karten in CacheStart implementiert	Team-Rangliste fertig implementieren	getTeamRankingList funktioniert nicht
Robin Christ	Teamseite API calls implementiert	Fehler in Logik beheben	createTeam funktioniert nicht

Bemerkungen für nächstes Treffen:

Datum: 09.05.2019

2019-05-09 Daily Scrum 28

Datum

09.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: A212

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Timo Volkmann

Abwesend:

- Michael Dietz

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Maximilian Leopold	-	Aufgaben anschauen und aufteilen Mit Michael absprechen, was im Backend gemacht werden muss für den kommenden Sprint Mit Frontend absprechen, sodass für Fehlermeldungen ein Konsens geschaffen wird	Keine
Timo Volkmann	Probleme bei Team-Erstellen-Anfrage behoben	Aufgaben anschauen und aufteilen Progress Stepper weiter implementieren	Keine
Katharina Will	Team-Rangliste fertig implementiert Angefangen Progress Stepper in angefangenen Caches einzubinden	Aufgaben anschauen und aufteilen Cache Übersicht in Kartenform implementieren	Keine
Robin Christ	Problem von gestern der Funktion createTeam behoben	Restliche User Story zur Team Implementierung abschließen	Keine

Bemerkungen für nächstes Treffen:

Datum: 10.05.2019

2019-05-10 Daily Scrum 29

Datum

10.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Timo Volkmann
- Michael Dietz

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	-	In neue User Stories einarbeiten	Keine
Maximilian Leopold	Verschiedene Bugfixes in den API Calls, die für Profil und Team zuständig sind	Aufgaben im Backend einteilen Noch mehr Bugfixes	Keine
Timo Volkmann	Cache-Übersicht verbessert und Fehlerquellen beseitigt Fortschrittsanzeige implementiert	Download Button für QR-Codes Cache bearbeiten ermöglichen	Keine
Katharina Will	Mehrere POI's mithilfe der Lageplan Gruppe versucht darzustellen	In Karten-Frameworks einlesen	Keine
Robin Christ	Team Komponente fertig gestellt	In neue User Stories einarbeiten Eingabevalidierung im Frontend	Keine

Bemerkungen für nächstes Treffen:

Datum: 14.05.2019

Am 13.05. entfällt das Daily Scrum, aufgrund vom Start der Blockwoche.

2019-05-14 Daily Scrum 30

Datum

14.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Timo Volkmann
- Michael Dietz

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Systemtests egschrieben editCache überarbeitet	Usability Tests vorbereiten	Keine
Maximilian Leopold	Systemtests aufgestellt Angefangen Backend zu loggen Study Script geschrieben Bugfixes in Datenbank	Blockwochen Aufgaben machen	Keine
Timo Volkmann	Cache bearbeiten implementiert Fehlermeldungen überarbeitet Bugfixes	Bugfixes Tests (Blockwoche)	Keine
Katharina Will	System-Testfälle in Jira geschrieben	Tests der Terminkalender Gruppe durchführen Testaufgaben für Usability Tests schreiben	Keine
Robin Christ	Designanpassungen und Bugfixes an Profil- und Teamseite	Eingabvalidierung im Frontend	Keine

Bemerkungen für nächstes Treffen:

Datum: 16.05.2019

Aufgrund des erhöhten Arbeitsaufwands während der Blockwoche werden Daily Scrums spontan durchgeführt

2019-05-16 Daily Scrum 31

Datum

16.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Timo Volkmann
- Michael Dietz

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Zahlreiche Bugfixes Fehlermeldungen nochmals verbessert Klassendiagramm gezeichnet	Weitere Bugfixes	Keine
Maximilian Leopold	Logger für Backend Bugfixes Usability Test Testaufgaben schreiben für Abnahmetests	Blockwochen Aufgaben machen Bugfixes	Keine
Timo Volkmann	Bugs gefixed Präsentation der Testergebnisse Feedback umgesetzt	Fehlermeldungen verbessern Feedback umsetzen	Keine
Katharina Will	Tests bei der Terminkalender Gruppe durchgeführt Usability Tests an Probanden durchgeführt Präsentation der Usability Test Ergebnisse	Testfälle für Abnahmetests schreiben	Keine

Bemerkungen für nächstes Treffen:

Datum: 17.05.2019

2019-05-17 Daily Scrum 32

Datum

17.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ

Abwesend:

- Michael Dietz
- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Maximilian Leopold	Präsentation der Abnahmetests	Weiterentwicklung	Keine
Katharina Will	Testfälle für Abnahmetests schreiben	In Kartenframeworks einlesen und Karten implementieren	Keine
Robin Christ	Präsentation der Abnahmetests	Weiterentwicklung	Keine

Bemerkungen für nächstes Treffen:

Datum: 20.05.2019

2019-05-20 Daily Scrum 33

Datum

20.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Robin Christ
- Timo Volkmann
- Michael Dietz

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	-, da im Backend mehr oder weniger fertig geschrieben und dokumentiert ist	Absprechen, was noch zu tun ist	Keine
Maximilian Leopold	-, da im Backend mehr oder weniger fertig geschrieben und dokumentiert ist	Absprechen, was noch zu tun ist	Keine
Katharina Will	Karten implementiert Meeting mit Timo wegen Karten	Marker für Startstationen implementieren GeoLocation des Nutzers anzeigen	Keine
Timo Volkmann	Meeting mit Katharina wegen Karten	-	Keine

Bemerkungen für nächstes Treffen:

Datum: 21.05.2019

2019-05-21 Daily Scrum 34

Datum

21.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- [Katharina Will](#)
- [Maximilian Leopold](#)
- [Timo Volkmann](#)
- [Michael Dietz](#)

Abwesend:

- [Robin Christ](#)

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	-, da im Backend mehr oder weniger fertig geschrieben und dokumentiert ist	Absprechen, was noch zu tun ist	Keine
Maximilian Leopold	Bugfixes Refactoring	Absprechen, was noch zu tun ist	Keine
Katharina Will	GeoLocation des Nutzers weiter implementiert	Marker für Startstationen implementieren GeoLocation des Nutzers anzeigen	Marker anzuzeigen ist komplexer als gedacht, da eine Karte zusätzliche Layer braucht
Timo Volkmann	-	Weiter Feedback aus Usability Tests umsetzen	Keine

Bemerkungen für nächstes Treffen:

Datum: 22.05.2019

2019-05-22 Daily Scrum 35

Datum

22.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	-	Absprechen, was noch zu tun ist	Keine
Maximilian Leopold	Bugfixes Refactoring	Absprechen, was noch zu tun ist	Keine
Katharina Will	GeoLocation des Nutzers fertig implementiert Angefangen in Marker setzen auf Karten einzulesen	Marker für Startstationen implementieren	Marker anzuzeigen ist komplexer als gedacht, da eine Karte zusätzliche Layer braucht
Timo Volkmann	-	Weiter Feedback aus Usability Tests umsetzen	Keine

Bemerkungen für nächstes Treffen:

Datum: 23.05.2019

2019-05-23 Daily Scrum 36

Datum

23.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	API Methode zum Zurücksetzen der Rangliste verfasst Rangliste gefixed, sodass sie in der Rangliste angezeigt wird	Im neuen Sprint schauen, ob evtl auf Frontend gewechselt wird, da im Backend eigentlich nichts mehr zu tun ist	Keine
Maximilian Leopold	Diverse Bugfixes in der Rangliste	Im neuen Sprint schauen, ob evtl auf Frontend gewechselt wird, da im Backend eigentlich nichts mehr zu tun ist	Keine
Katharina Will	Marker setzen auf Karten angefangen zu implementieren	Marker für Startstationen weiter implementieren	Marker anzuzeigen ist komplexer als gedacht, da eine Karte zusätzliche Layer braucht
Timo Volkmann	Übersicht geschaffen in den zu erledigenden Aufgaben	Weiter Feedback aus Usability Tests umsetzen Bugfixes	Keine

Bemerkungen für nächstes Treffen:

Datum: 24.05.2019

2019-05-24 Daily Scrum 37

Datum

24.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Michael Dietz

Abwesend:

- Robin Christ
- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Frontend etwas angeschaut	Frontend weiter anschauen	Keine
Maximilian Leopold	Mit Timo besprochen, wie die Webapp in zwei .war-Dateien aufgeteilt wird Anfangen Frontend anzuschauen und einzuarbeiten	Weiter an Frontend arbeiten .war-Datei deployen fertig machen	Keine
Katharina Will	Marker auf Karte anzeigen ermöglichen	CacheStart wieder funktionsfähig machen	Keine

Bemerkungen für nächstes Treffen:

Datum: 27.05.2019

2019-05-27 Daily Scrum 38

Datum

27.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Katharina Will
- Maximilian Leopold
- Timo Volkmann
- Michael Dietz
- Robin Christ

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Bisschen ins Frontend eingesehen und mit Technologien auseinandergesetzt	Problem mit der Rangliste anschauen	Keine Erfahrung in Frontend und Projekt ist kurz vor dem Ende
Maximilian Leopold	Anfangen im Frontend mitzuarbeiten Rangliste angepasst, Nutzender ist anklickbar Backend fertig gestellt	Validierungsaufgaben bearbeiten Karten anschauen	In der Rangliste werden per SQL die Ränge durcheinander ausgegeben, bei gleicher Punktzahl
Katharina Will	CacheStart wieder funktionsfähig gemacht	Cache Overview Karte weiter bearbeiten	Karte wird nicht angezeigt, erst wenn das Smartphone gedreht wird
Timo Volkmann	Cache Listen überarbeitet, Buttons zusammengefasst Cache zurücksetzen hinzugefügt Projekt umgebaut Dokumentation	Karten anschauen	Karte wird nicht angezeigt, erst wenn das Smartphone gedreht wird
Robin Christ	Designanpassungen	Validierung im Frontend Dokumentation erweitern	Keine

Bemerkungen für nächstes Treffen:

Datum: 28.05.2019

2019-05-28 Daily Scrum 39

Datum

28.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Michael Dietz
- Timo Volkmann

Abwesend:

- Katharina Will
- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Bug gefixed, welcher unterschiedliche Platzierungen in der Rangliste und im Profil anzeigte	Schauen, was noch gefixed werden kann	Keine
Maximilian Leopold	Kleine Änderungen an den gradle Dateien und bug fixes	Frontend Validierungsaufgaben	Keine
Timo Volkmann	Einarbeiten in Karten Kartenanzeige auf Übersicht gefixed	Weiter mit Karten beschäftigen	Keine

Bemerkungen für nächstes Treffen:

Datum: 29.05.2019

2019-05-29 Daily Scrum 40

Datum

28.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Michael Dietz
- Timo Volkmann
- Katharina Will

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Frontend Dateien angeschaut und versucht zu verstehen	Schauen, welche Kleinigkeiten es im Frontend zu tun gibt und versuchen diese zu machen	Keine
Maximilian Leopold	Mehrere Validierungsaufgaben im Frontend	Weitere Validierungsaufgaben und je nachdem was im Frontend zu erledigen ist	Keine
Katharina Will	-	Mit Karte in der Cache Übersicht beschäftigen	Keine
Timo Volkmann	-	Weiter mit Karten beschäftigen	Keine

Bemerkungen für nächstes Treffen:

Datum: 31.05.2019

2019-05-31 Daily Scrum 41

Datum

31.05.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Timo Volkmann
- Katharina Will

Abwesend:

- Robin Christ
- Michael Dietz

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Maximilian Leopold	Versucht Schwierigkeiten beim deployen unserer Webanwendung zu beheben	Frontend Validierungsaufgaben	Keine
Katharina Will	-	Karten mit Timo bearbeiten	Keine
Timo Volkmann	-	Karten mit Katharina bearbeiten	Keine

Bemerkungen für nächstes Treffen:

Datum: 03.06.2019

2019-06-03 Daily Scrum 42

Datum

03.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Timo Volkmann
- Katharina Will
- Michael Dietz

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Mit dem Cache bearbeiten bug beschäftigt	Farbdesign anpassen	Keine
Maximilian Leopold	Etwas mehr in vue etc eingearbeitet Versucht den bug beim deployen zu verstehen und zu beheben	Mehrere kleine Frontend	Keine
Katharina Will	Karte der Cache Übersicht implementiert mit den Startpunkten der Caches, welche in verschiedenen Farben angezeigt werden, je nachdem, wie weit der Nutzer diese bearbeitet hat	Weiter an den Karten arbeiten	Keine
Timo Volkmann	Mit dem Anklickbar-machen der Karte auseinandergesetzt	Popups für Übersichtskarte implementieren	Keine

Bemerkungen für nächstes Treffen:

Datum: 04.06.2019

2019-06-04 Daily Scrum 43

Datum

04.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Timo Volkmann
- Katharina Will
- Michael Dietz

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Verwendete Farben bei der Benutzeroberfläche vereinheitlicht	Weiter optimieren	Keine
Maximilian Leopold	Einarbeiten in Karten-Thematik mit vue und vuelayers	Weitere Versuche mit Karten und Bugfixes	Keine
Katharina Will	StationView zeigen Marker aller Stationen eines Cache an	Farbe der angezeigten Marker in StationView anpassen, sowie die Sichtbarkeit der Marker	Keine
Timo Volkmann	Popups auf der Karte anzeigen Cache starten über die Karte implementiert	Schauen, was bei Dokumentation noch gemacht werden muss	Keine

Bemerkungen für nächstes Treffen:

Datum: 05.06.2019

2019-06-05 Daily Scrum 44

Datum

05.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Timo Volkmann
- Katharina Will
- Michael Dietz

Abwesend:

- Robin Christ

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Farbschema endgültig angepasst	Auf Abruf, falls etwas gemacht werden muss	Keine
Maximilian Leopold	Bug im Backend gefixed Farbschema angepasst Alles, was gemacht wurde, angeschaut	Auf Abruf, je nachdem was vor Sprintende noch zu tun ist	Keine
Katharina Will	StationView zeigen Marker aller Stationen eines Cache an	Farbe der angezeigten Marker in StationView anpassen, sowie die Sichtbarkeit der Marker	Keine
Timo Volkmann	Popups auf der Karte anzeigen	Schauen, was bei Dokumentation noch gemacht werden muss	Keine

Bemerkungen für nächstes Treffen:

Datum: 06.06.2019

2019-06-06 Daily Scrum 45

Datum

06.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: A212

Anwesende Teilnehmer

- Maximilian Leopold
- Timo Volkmann
- Katharina Will
- Michael Dietz
- Robin Christ

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Die Profilseite, Team-Tab Design bearbeitet	Startseite überarbeiten	Keine
Maximilian Leopold	Bug im Backend gefixed Von einem User Punkte resettet	Bugs fixen Dokumentation	Keine
Katharina Will	StationView zeigen Marker aller Stationen eines Cache an	Farbe der angezeigten Marker in StationView anpassen, sowie die Sichtbarkeit der Marker Marker-path in CacheStart anpassen	Keine
Timo Volkmann	Popups auf der Karte anzeigen bug gefixed	Schauen, was in der Dokumentation gemacht werden muss Bugs fixen	Keine
Robin Christ	[Krankheitsbedingt ausgefallen]	Dokumentation Bugfixes	Keine

Bemerkungen für nächstes Treffen:

Datum: 17.06.2019

2019-06-17 Daily Scrum 46

Datum

17.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: A106

Anwesende Teilnehmer

- Maximilian Leopold
- Timo Volkmann
- Katharina Will
- Michael Dietz
- Robin Christ

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Dokumentation anfangen Präsentation vorbereitet	Präsentation	Keine
Maximilian Leopold	Dokumentation geschrieben Dokumentationsaufgaben aufgeteilt	Dokumentationsaufgaben durchführen	Keine
Katharina Will	Dokumentationsaufgaben aufgeteilt Cache Fortschritt wird auf der Karte angezeigt	Dokumentationsaufgaben durchführen	Keine
Timo Volkmann	Überlegt, was für die Dokumentation gemacht werden muss Dokumentation aufgeteilt Cache Fortschritt wird auf der Karte angezeigt	Dokumentationsaufgaben durchführen	Keine
Robin Christ	Validierung bei Cache und Stationen anfangen Präsentation vorbereitet	Dokumentationsaufgaben durchführen Präsentation weiter vorbereiten	Keine

Bemerkungen für nächstes Treffen:

Datum: 18.06.2019

2019-06-18 Daily Scrum 47

Datum

18.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Katharina Will
- Michael Dietz
- Robin Christ

Abwesend:

- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Präsentationsplanung	Präsentation vorbereiten	Keine
Maximilian Leopold	Installationsanleitung als Doku geschrieben	Weiter an Dokumentation arbeiten	Keine
Katharina Will	Projektorganisation angefangen zu dokumentieren	Weiter an Dokumentation arbeiten	Keine
Robin Christ	Validierung beim Cache erstellen fertig gestellt Präsentationsplanung	Präsentation weiter vorbereiten	Keine

Bemerkungen für nächstes Treffen:

Datum: 19.06.2019

2019-06-19 Daily Scrum 48

Datum

19.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- [Maximilian Leopold](#)

Abwesend:

- [Timo Volkmann](#)
- [Katharina Will](#)
- [Michael Dietz](#)
- [Robin Christ](#)

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Maximilian Leopold	Architekturdiagramm vervollständigt	Weiter an Dokumentation arbeiten	Keine

Bemerkungen für nächstes Treffen:

Datum: 21.06.2019

2019-06-21 Daily Scrum 49

Datum

21.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Katharina Will
- Michael Dietz
- Robin Christ

Abwesend:

- Timo Volkmann

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Datenbankstruktur, SQL-Anfragen, Trigger Doku geschrieben Präsentationsvorbereitung	Präsentationsvorbereitung	Keine
Maximilian Leopold	Doku für Leistungen pro Teammitglied pro Sprint geschrieben	Weiter an Dokumentation arbeiten	Keine
Katharina Will	Projektorganisation soweit fertig gestellt	Bedienungsanleitung schreiben	Keine
Robin Christ	Präsentationsvorbereitung und Besprechung bzw. Planung	Präsentationsvorbereitung	Keine

Bemerkungen für nächstes Treffen:

Datum: 24.06.2019

2019-06-24 Daily Scrum 50

Datum

24.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Katharina Will
- Michael Dietz
- Robin Christ
- Timo Volkmann

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Dokumentation geschrieben Präsentationsvorbereitung	Präsentationsvorbereitung	Keine
Maximilian Leopold	Dokumentation geschrieben	Dokumentation schreiben	Keine
Timo Volkmann	Dokumentation geschrieben	Dokumentation schreiben	Keine
Katharina Will	Dokumentation geschrieben	Dokumentation schreiben	Keine
Robin Christ	Dokumentation geschrieben Präsentationsvorbereitung	Präsentationsvorbereitung	Keine

Bemerkungen für nächstes Treffen:

Datum: 25.06.2019

2019-06-25 Daily Scrum 51

Datum

25.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: F229

Anwesende Teilnehmer

- Maximilian Leopold
- Katharina Will
- Michael Dietz
- Robin Christ
- Timo Volkmann

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Dokumentation geschrieben Präsentationsvorbereitung	Präsentationsvorbereitung	Keine
Maximilian Leopold	Dokumentation geschrieben	Dokumentation schreiben	Keine
Timo Volkmann	Dokumentation geschrieben	Dokumentation schreiben	Keine
Katharina Will	Dokumentation geschrieben	Dokumentation schreiben	Keine
Robin Christ	Dokumentation geschrieben Präsentationsvorbereitung	Präsentationsvorbereitung	Keine

Bemerkungen für nächstes Treffen:

Datum: 26.06.2019

2019-06-26 Daily Scrum 52

Datum

26.06.2019

Von: 13:00 Uhr Bis: 13:10 Uhr

Ort: virtuell

Anwesende Teilnehmer

- Maximilian Leopold
- Katharina Will
- Michael Dietz
- Robin Christ
- Timo Volkmann

Abwesend:

Moderation: Katharina Will

Protokoll: Katharina Will

Daily Scrum

Teilnehmer	Was habe ich seit letztem Treffen gemacht?	Was mache ich bis zum nächsten Treffen?	Welche Probleme können auftreten?
Michael Dietz	Dokumentation geschrieben Präsentationsvorbereitung	Präsentationsvorbereitung Dokumentation korrekturlesen	Keine
Maximilian Leopold	Dokumentation geschrieben	Dokumentation korrekturlesen	Keine
Timo Volkmann	Dokumentation geschrieben	Dokumentation korrekturlesen	Keine
Katharina Will	Dokumentation geschrieben	Dokumentation korrekturlesen	Keine
Robin Christ	Dokumentation geschrieben Präsentationsvorbereitung	Präsentationsvorbereitung Dokumentation korrekturlesen	Keine

Bemerkungen für nächstes Treffen:

Datum: ---

Retrospectives

[Create retrospective](#)

Die Sprint Retrospektiven sind ein Mittel zur Selbstverbesserung des Teams. In einem solchen Treffen, für das sich jedes Scrum-Teammitglied gut vorbereitet hat, wird gelobt, was gut gemacht wurde im Hinblick auf Teamzusammenarbeit, und kritisiert, was besser hätte laufen können.

Auf die Kritik wird, falls notwendig, reagiert, sodass die Teamarbeit verbessert wird.

Title	Date	Participants
2019-05-23 Retrospektive		
2019-06-06 Retrospektive		
2019-06-25 Retrospektive		
2019-05-09 Retrospektive		
2019-04-25 Retrospektive		
2019-04-11 Retrospektive		
2019-03-28 Retrospektive		

2019-03-28 Retrospektive

Datum	28.03.2019
Teilnehmer	Robin Christ Maximilian Leopold Timo Volkmann Katharina Will Michael Dietz Abwesend: Rene Schiesswohl

Retrospektive

Was haben wir gut gemacht?

- Als Team gut zusammengearbeitet, alle haben etwas gemacht (mit Ausnahmen)
- Gute Kommunikation im Team
- Hohe Anwesenheitsquote (mit Ausnahmen)
- Gut organisiert, z.B. jeder hat etwas bearbeitet
- Schnelle Reaktionszeit, Fragen wurden zügig beantwortet
- Erster Sprint wurde in praktisch einer Woche fast komplett umgesetzt, obwohl ein Teammitglied nicht anwesend war

Was hätten wir besser machen können?

- Vorstellungen miteinander teilen, einen Konsens schaffen
- Früher auf das häufige Fehlen des neuen Teammitglieds reagieren

Aktionen

- ☒ Gemeinsame Nachricht an René Schießwohl verfassen & verschicken
- ☒ Separates Meeting, um Design der Anwendung zu besprechen, um Konsens zu schaffen (Geplant für 01.04.2019 nach dem Daily Scrum)

2019-04-11 Retrospektive

Datum	11.04.2019
Teilnehmer	Katharina Will Michael Dietz Maximilian Leopold Timo Volkmann

Retrospektive

Was haben wir gut gemacht?

- Man kann mit jedem reden, man bekommt sinnvolle Antworten
- Jeder hat etwas beigetragen, 6. Teammitglied wurde eingebunden
- Alle User Stories wurden bearbeitet
- Hohe Anwesenheitsquote
- Gute Kommunikation zwischen Frontend und Backend
- Gute Zusammenarbeit über Distanz, man muss nicht immer zusammensitzen, um etwas fertig zu machen

Was hätten wir besser machen können?

- Besser Arbeitszeiten loggen
- Früher Tasks zuweisen, damit jeder besser Bescheid weiß, wer was macht

2019-04-25 Retrospektive

Datum	25.04.2019
Teilnehmer	Katharina Will Maximilian Leopold Timo Volkmann Robin Christ Michael Dietz

Retrospektive

Was haben wir gut gemacht?

- Angelegenheit mit 6. Gruppenmitglied geklärt
- gute Hilfsbereitschaft, wenn man nach Hilfe fragt, erhält man sie
- Wenn in der Gruppe jemand bestätigt, dass er etwas macht, wird es erledigt; zuverlässig
- gute Kommunikation zwischen Frontend und Backend, schnelle Reaktionszeit auf Probleme
- gute Kommunikation mit anderen Gruppen, um mit diesen zu kooperieren
- Es wird selbstständig geloggt

Was hätten wir besser machen können?

- Viele Feiertage, daher weniger produktiv
- Früher fertig werden, damit technische Probleme am Ende eines Sprints weniger Probleme verursachen

Aktionen

- ☒ neu organisieren, da ein Gruppenmitglied weniger mitarbeitet

2019-05-09 Retrospektive

Datum	09.05.2019
Teilnehmer	Katharina Will Maximilian Leopold Robin Christ Timo Volkmann Abwesend: Michael Dietz

Retrospektive

Was haben wir gut gemacht?

- Gute Arbeitsteilung, trotz und mit Präsentationsvorbereitung
- Schnelle Reaktionszeiten auf externe Vorschläge
- Schnelle Reaktionszeiten, wenn man Fragen oder Hilfe brauchte
- Gut organisiert, jeder hat seinen Teil beigetragen
- Es wird selbstständig geloggt
- gute Kommunikation und Kooperation mit anderen Gruppen
- alle Gruppenmitglieder sind zufrieden mit der Zusammenarbeit

Was hätten wir besser machen können?

- Mehr Fokus auf Dokumentation setzen im Frontend
- im Falle von Merge Konflikten kurz miteinander kommunizieren

Aktionen

- ☐ Gedanken über Aufbesserung der Frontend-Dokumentation machen

2019-05-23 Retrospektive

Datum	23.05.2019
Teilnehmer	Katharina Will Michael Dietz Maximilian Leopold Timo Volkmann Abwesend: Robin Christ

Retrospektive

Was haben wir gut gemacht?

- Aufgaben innerhalb der Blockwoche gut aufgeteilt, sodass man mit der Entwicklung voran gekommen ist und nicht nur den Tests
- gute Kommunikation im Team
- Hilfsbereitschaft
- Wenn ein Bug gefunden wird, wird dieser sehr schnell behoben
- Logging im Jira wird immer besser (Arbeitszeiten, wie auch Bug reporting)
- Anwesenheit sehr gut, Fehlen wegen Vorstellungsgesprächen war immer entschuldigt

Was hätten wir besser machen können?

- Es werden nicht alle User Stories fertig
- Mehr Dokumentation im Frontend

Aktionen

- ☒ Wegen Arbeitsmangel im Backend, werden Michael Dietz und Maximilian Leopold ins Frontend wechseln

2019-06-06 Retrospektive

Datum	06.06.2019
Teilnehmer	Katharina Will Timo Volkmann Robin Christ Michael Dietz Maximilian Leopold

Retrospektive

Was haben wir gut gemacht?

- Extrem viele Story Points umgesetzt
- Wechsel ins Frontend von Maximilian und Michael war sehr hilfreich
- Sehr gute Hilfsbereitschaft
- Schnelle Reaktionszeiten von Teammitgliedern auf Probleme und Fragen
- Gute Kommunikation
- Jeder kommuniziert klar, was er macht und sagt wenn er fertig ist
- Kritik wird angenommen

Was hätten wir besser machen können?

- Dokumentation hätte man besser pflegen können, daher wird es zum Ende des Projekts ziemlich umfangreich

Aktionen

- ☒ Maximilian Leopold, Michael Dietz, Timo Volkmann, Robin Christ, Katharina Will: Dokumentation pflegen und aufbereiten

2019-06-25 Retrospektive

Datum	25.06.2019
Teilnehmer	Katharina Will Michael Dietz Robin Christ Timo Volkmann Maximilian Leopold

Retrospektive

Was haben wir gut gemacht?

- gute Arbeitsteilung trotz Präsentationsvorbereitung
- Dokumentation wurde rechtzeitig fertig
- Dokumentation wurde unter allen Teammitgliedern aufgeteilt
- gute Kooperation und gute Kommunikation
- Mitglieder waren anwesend bei gemeinsamen Treffen
- Änderungsvorschläge werden angenommen
- Man hat eine gemeinsame Vorstellung, wie das Ergebnis aussehen soll

Was hätten wir besser machen können?

- Man hätte früher mit der Dokumentation beginnen können
- Change Request früher stellen

Sprint Reviews

Was ist eine Sprint Review:

Eine Sprint Review wird immer am Ende eines Sprint abgehalten. Hierzu treffen sich der Product Owner und die Stakeholder mit dem Team und das Team präsentiert den derzeitigen Anwendungsstand. Der Product Owner und die Stakeholder geben hierzu dann ihre Gedanken, Anmerkungen und Kritik ab und das Team ist dazu angehalten diese in die Umsetzung des nächsten Sprints und dessen Aufgaben mit einfließen zu lassen.

Product Owner: Maximilian Leopold

Team: Timo Volkmann, Michael Dietz, Robin Christ, Katharina Will

Protokollantin: Katharina Will

Es empfiehlt sich nicht die alten Versionen der *.war Dateien auf einen Server zu deployen, da sich die Datenbank über mehrer Iterationen von Sprint zu Sprint weiterentwickelt und verändert hat. Viele der alten Versionen werden also mit der aktuellen Datenbank nicht funktionieren und können die Datenbank auch völlig kaputt machen, sodass sie auch mit der neusten Version nichtmehr funktionieren würde!

Sie finden alle *.war Dateien, die hier verlinkt sind auf dem USB-Stick unter "Artefakte".

Wichtig:

Falls versucht werden sollte die .war Dateien älterer Sprint Reviews auf den TomCat Server zu deployen, müssen diese nach dem Download umbenannt werden!

Namenkonvention:

Sprint 1 - 5: Es existiert nur eine .war Datei. Diese muss in **buga19geocaching.war** umbennant werden!

Sprint 6 - 7: Die Datei mit dem Namen **buga19geocaching_backend_sprintX[_releaseX].war** muss in **buga19geocachingbackend.war** umbenannt werden. Die Datei mit dem Namen **buga19geocaching_frontend_sprintX[_releaseX].war** muss in **buga19geocaching.war** umbenannt werden!

Die Namenkonvention ist so gewählt um Verwirrungen zwischen den Versionen auszuschließen.

Rot markiert sind kritische Anmerkungen

Grün markiert sind positive Anmerkungen

Schwarz markiert sind neutrale Anmerkungen

Sprint	Datum	Bemerkungen des Product Owners	Stand der Anwendung
1	28.03.2019	Design einheitlicher gestalten Cache starten-Knopf vorhanden, aber ohne Funktionalität Menü sollte links sein Startseite sieht etwas seltsam aus	Es existiert noch keine deploybare .war Datei. Die Anwendung konnte nur lokal ausgeführt werden.
2	11.04.2019	Fehlermeldungen sollen ausgegeben werden. Man sieht nicht wenn etwas falsch gemacht wurde Im Profil geht die Leiste über die Bildschirmbreite hinaus Einheitliche Farben verwenden Es existiert ein Profil-Knopf aber das wurde nirgendwo als User Story definiert Startseite sieht etwas seltsam aus	Es existiert noch keine deploybare .war Datei. Die Anwendung konnte nur lokal ausgeführt werden.

3	25.04.2019	<p>Server sind offline, Review wird verschoben</p> <p>Ersatztermin für Review: 02.05.2019</p> <p>Fehlermeldungen verbessern. Informationen die den Nutzern etwas bringt, nicht nur den Entwicklern</p> <p>Vereinheitlichen der Button-Farben</p> <p>Automatische Weiterleitung zur Registrierung wenn man nicht angemeldet ist beim scannen eines QR-Code</p> <p>Falls Cache schon durchgeführt wurde, dürfen beim erneuten Durchführen keine Punkte gutgeschrieben werden</p> <p>Team-Seite anpassen, sodass man nicht scrollen muss. Design vereinheitlichen</p> <p>Beim Cachen: Nach QR-Code scannen automatisch nach unten scrollen</p>	buga19geocaching_sprint3.war
4	09.05.2019	<p>Bei der Cache Erstellung einen QR-Code Generator implementieren</p> <p>Neues Symbol für QR-Code Scanner ist gut, besser erkennbar</p> <p>Wenn man logout drückt, wird man direkt ausgeloggt; Sehr gut, da man nicht nochmal extra logout bestätigen muss</p> <p>Karte für die Cache Übersicht noch nicht implementiert</p> <p>Team Übersicht Design vereinheitlichen und anpassen</p> <p>Ranglistenposition ändert sich ständig wenn man 0 Punkte hat</p> <p>In der Cache Übersicht sollte evtl farblich markiert sein wenn der Cache bereits beendet wurde</p>	buga19geocaching_sprint4.war
5	23.05.2019	<p>Karte lädt nicht richtig; Karte bleibt weiß</p> <p>Rangliste zeigt nicht den eigenen Rang an der 11. Position an</p> <p>Buttonfarben vereinheitlichen</p> <p>Eigener Standort wird auf PO MacBook nicht angezeigt</p> <p>In Rangliste kann man nicht auf eigenen Namen klicken</p>	buga19geocaching_sprint5.war
6	06.06.2019	<p>Design ist toll; Buttonfarben sind einheitlich</p> <p>Auf der Startseite muss man die Informationen zur Anwendung nicht mehr ausklappen</p> <p>Alle Karten sind implementiert</p> <p>Man startet in der Kartenansicht auf dem Buga-Gelände</p> <p>Admin-Funktionen die vom Kunde gewünscht wurden, wurden implementiert</p> <p>Wenn Listen leer sind, wird "Keine Einträge" angezeigt</p> <p>Auf der Rangliste kann man seinen eigenen Namen unterhalb der Rangliste finden, dieser ist sogar anklickbar</p> <p>Man muss Koordinaten nicht mehr mühselig eingeben beim Anlegen einer neuen Station, sondern auf der Karte auswählen oder per GPS-Knopf den eigenen Standort auswählen</p> <p>Beim Auswählen von Koordinaten auf der Karte, wird der letzte ausgewählte Punkte immer noch angezeigt</p>	buga19geocaching_backend_sprint6_release3.war buga19geocaching_frontend_sprint6_release3.war
7	25.06.2019	<p>Keine großen ersichtlichen Änderungen</p> <p>E-Mail Validierung in der Registrierung funktioniert wieder</p> <p>Rechtschreibfehler in den Fehlermeldungen korrigiert</p> <p>Im Tab "Beendete Caches" in "Meine Caches" wird bei einem aufgeklappten Cache wieder ein Knopf angezeigt zum Belohnung anzeigen</p> <p>Admin-Funktion zur Beförderung eines Cachenden zu einem Adminisitrierenden wurde gestrichen, da Schnittstellen mit zentralem Usermanagement fehlerhaft. Da nur Zusatzfeature nicht weiter tragisch.</p>	buga19geocaching_frontend_sprint7_endrelease.war buga19geocaching_backend_sprint7_endrelease.war kann leider nicht eingefügt werden, da die Datei zu große ist. Auf dem USB-Stick ist sie jedoch unter "Artefakte" vorhanden.

Bedienungsanleitung

Nachfolgend finden Sie die Bedienungsanleitung zur Webanwendung von BuGa-Geocaching.

Die Bedienungsanleitung ist unterteilt in mehrere Kapitel, die sich an die verschiedenen Nutzergruppen richtet: Besuchende, Cachende und Administrierende. In den Kapiteln wird unter Anderem auf andere Kapitel oder auf Zusatzkapitel verwiesen. Sie können selbst entscheiden, ob Sie Zusatzkapitel lesen möchten. Diese Zusatzkapitel erkennen Sie daran, dass diese in der Navigation nicht nummeriert sind.

Zum einfacheren Verständnis sind Bilder der Anwendung in den jeweiligen Kapiteln vorhanden. Bitte beachten Sie, dass die Bilder der Desktop-Ansicht im Vivaldi-Browser erstellt wurden und daher leichte Differenzen im Aussehen der Webanwendung in Ihrem Browser vorhanden sein können. Die Mobilansicht wurde anhand eines Google Pixel 2 erstellt und können daher ebenfalls Differenzen in der Skalierung aufweisen. Die Webanwendung passt sich automatisch Ihrer Auflösung an Ihrem Mobilgerät an und skaliert dementsprechend mit.

Nachfolgend ein Beispiel, um dies zu verdeutlichen:

Google Pixel 2:



iPhone X:



I. Für BuGa-Besuchende

Inhalt:

1. [Startseite aufrufen](#)
2. [Caches](#)
3. [TOP10-Rangliste](#)
4. [Konto anlegen](#)

Zusatzkapitel:

[Die Navigationsleiste](#)

1. Startseite aufrufen

Öffnen Sie Ihren Internet-Browser und geben Sie in der Adressleiste

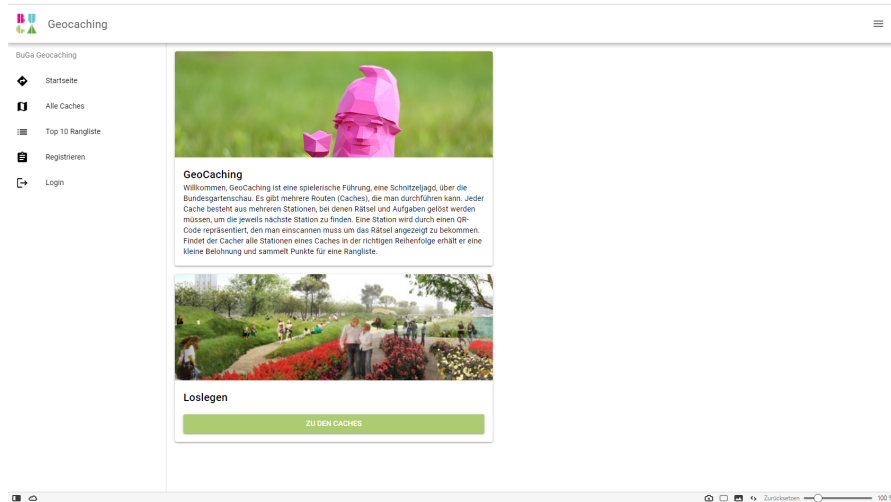
<https://seserver.se.hs-heilbronn.de:9443/buga19geocaching/#/>

ein. Nachdem Sie die Eingabetaste gedrückt haben, ist die Startseite von BuGa-GeoCaching zu sehen.

Auf der Startseite erhalten Sie Informationen zu dem GeoCaching-Angebot, worum es sich dabei handelt und wie es grob funktioniert. Außerdem kann man direkt zu den Caches navigieren, indem man "Zu den Caches" auswählt. *Dazu mehr in 2. Caches.*

Auf der Desktop-Ansicht ist die Navigationsleiste automatisch geöffnet, in der Mobilansicht ist sie geschlossen. Man öffnet die Navigationsleiste, indem man den Menü-Knopf in der oberen rechten Ecke des Bildschirms auswählt, dies gilt für beide Ansichten. *Dazu mehr in Die Navigationsleiste.*

Desktop-Ansicht:



Mobilansicht:



2. Caches

Hier gelangen Sie automatisch hin, wenn Sie

1. sich soeben mit Ihrem Konto angemeldet haben.
2. über die Navigationsleiste "Alle Caches" ausgewählt haben.
3. auf der Startseite "Zu den Caches" ausgewählt haben.

Hier finden Sie alle verfügbaren Caches. Sie sind in der Listenansicht, wenn Sie auf die Seite gelangen. Man kann auf die Kartenansicht wechseln, indem man den Tab "Karte" auswählt.

In der Kartenansicht werden die Startstationen aller verfügbaren Caches angezeigt (Mehr dazu in [2. Cache starten](#)). Wenn Sie Ihrem Browser gewähren Ihre Standortdaten nutzen zu dürfen, wird Ihre Position als blauer Kreis dargestellt, dies gilt auch für alle Kartendarstellungen, die in folgenden Kapiteln erwähnt werden.

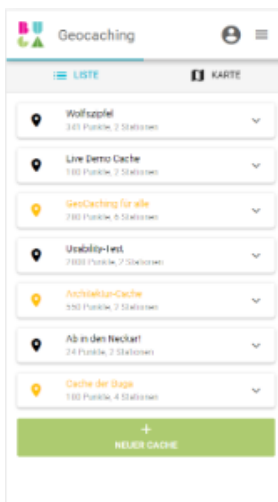
In beiden Ansichten kann man die Caches ausklappen, um weitere Informationen über sie zu erhalten. Sollte man in der Kartenansicht sein, muss man vorher den jeweiligen Marker auf der Karte auswählen, danach kann man ebenso die selben Daten sehen, wie in der Listenansicht.

Wenn Sie "Starten" in einem der Caches auswählen und nicht angemeldet sind, werden Sie automatisch zur Anmeldemaske weitergeleitet (Mehr dazu in [2. Cache starten](#)). Sollten Sie angemeldet sein, werden Sie auf die Startposition des ausgewählten Caches hingewiesen.

Sollten Sie angemeldet sein, sehen Sie die Caches in der Listenansicht, wie auch in der Kartenansicht in Farben hinterlegt, die für Ihren Bearbeitungsstatus stehen:

1. schwarz (bzw. blau in der Kartenansicht): der Cache wurde noch nicht angefangen
2. gelb: der Cache wurde angefangen, allerdings noch nicht abgeschlossen
3. grün: der Cache wurde bereits abgeschlossen

Mobilansicht der Cache-Liste, als angemeldeter Nutzer:



Mobilansicht der Cache-Karte, als angemeldeter Nutzer:



3. TOP10-Rangliste

Hier können Sie sehen, welche 10 Nutzer die meisten Punkte beim Cachen gesammelt haben.

Wenn der Tab "Team" ausgewählt wird, oder auf der Fläche, in der die Rangliste angezeigt wird, mit dem Finger von rechts nach links gewischt wird, gelangen Sie zu den 10 Team, sofern so viele vorhanden sind, die die meisten Punkte erreicht haben. (Mehr dazu im Kapitel "Für Cachernde" in [4. Das Profil und Teams](#))

Man kann, um zum Tab "Solo" zu gelangen diesen Tab entweder auswählen oder auf der Fläche, in der die Rangliste angezeigt wird, von links nach rechts wischen.

Sollten Sie angemeldet sein, wird Ihre eigene Position als 11. Position in der Rangliste angezeigt, wenn Sie nicht in den TOP10 sind. Ihre Position, Ihre E-Mail-Adresse und Ihre Punktzahl ist in grüner Schrift geschrieben.

Bei Auswahl der eigenen E-Mail-Adresse gelangen Sie zu Ihrem Profil. (Mehr dazu im Kapitel "Für Cachernde" in [4. Das Profil und Teams](#))

Mobilansicht der Solo-TOP10:



The screenshot shows the Geocaching mobile app interface. At the top, there's a header with the Geocaching logo and a user profile icon. Below the header, there are two tabs: "SOLO" (selected) and "TEAM". The main content area displays a list of the top 10 solo cachers. Each entry consists of a rank number, an email address, and a score in points.

Rang	E-Mail-Adresse	Punkte
1.	vollmann@geocaching.de	2000 Punkte
2.	michael@1978@gmx.de	24 Punkte
3.	aburton@web.de	0 Punkte
4.	admiral@bunnen.de	0 Punkte
5.	adu@attraktionen.de	0 Punkte
6.	amies@geoc.de	0 Punkte
7.	blumetse@geoc.de	0 Punkte
8.	cache.description@web.de	0 Punkte
9.	Dietz@web.de	0 Punkte
10.		

Mobilansicht der Solo-TOP10 mit eigener Position:



This screenshot is similar to the previous one, but it shows the user's position at the 11th rank. The user's email address and score are highlighted in green. The list shows the top 10 cachers followed by the user's entry at rank 11.

Rang	E-Mail-Adresse	Punkte
1.	vollmann@geocaching.de	2000 Punkte
2.	michael@1978@gmx.de	24 Punkte
3.	aburton@web.de	0 Punkte
4.	admiral@bunnen.de	0 Punkte
5.	adu@attraktionen.de	0 Punkte
6.	amies@geoc.de	0 Punkte
7.	blumetse@geoc.de	0 Punkte
8.	cache.description@web.de	0 Punkte
9.	Dietz@web.de	0 Punkte
10.		
11.	kwil@stad.ho-hellborn.de	0 Punkte

Mobilansicht der Team-TOP10:



4. Konto anlegen

Sie können sich ein eigenes Konto anlegen, wenn Sie in der Navigationsleiste "Registrieren" auswählen oder in der Anmeldemaske "Registrieren" auswählen. Um diese Knöpfe auswählen zu können, dürfen Sie aktuell nicht bereits mit einem Konto angemeldet sein.

Das Formular benötigt Ihren vollständigen Namen, Ihre E-Mail-Adresse, Ihre wiederholte E-Mail-Adresse, sowie ein Passwort und die Wiederholung des Passworts.

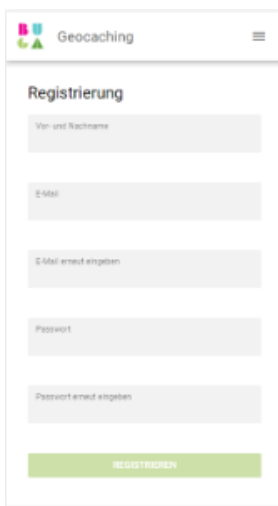
Sollten die Felder nicht korrekt befüllt sein, wird in roter Schrift eine Meldung unter dem betroffenen Feld ausgegeben.

Nachdem Sie das Formular vollständig ausgefüllt haben, wählen Sie den grünen "Registrieren"-Knopf im unteren Teil der Ansicht aus. Dieser Knopf ist nicht auswählbar, solange das Formular nicht vollständig ausgefüllt ist.

Sie werden automatisch mit Ihrem neuen Konto angemeldet und zur Cache-Übersicht weitergeleitet, nachdem Sie "Registrieren" ausgewählt haben.

Bitte beachten Sie, dass Sie bei der Registrierung automatisch damit Ihr Einverständnis geben mit Ihrer E-Mail-Adresse in der TOP10-Rangliste sichtbar zu werden.

Mobilansicht des leeren Registrierungsformulars:



The screenshot shows the mobile app interface for Geocaching. At the top is a header bar with the Geocaching logo and a menu icon. Below the header is a section titled "Registrierung". It contains five input fields: "Vor- und Nachname", "E-Mail", "E-Mail erneut eingeben", "Passwort", and "Passwort erneut eingeben". At the bottom of the form is a green button labeled "REGISTRIEREN".

Mobilansicht des mit Beispieldaten ausgefüllten Registrierungsformulars:



This screenshot shows the same mobile registration form as the previous one, but it is filled with example data. The "Vor- und Nachname" field contains "Monika Musterfrau". The "E-Mail" field contains "monika.musterfrau@mustermail.de". The "E-Mail erneut eingeben" field also contains "monika.musterfrau@mustermail.de". The "Passwort" and "Passwort erneut eingeben" fields are filled with masked characters (dots). The green "REGISTRIEREN" button remains at the bottom.

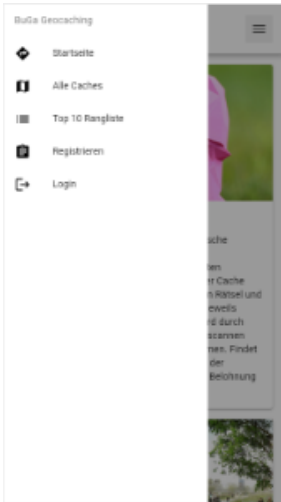
Die Navigationsleiste

Die Navigationsleiste kann stets im oberen rechten Bildschirmrand auf- oder zugeklappt werden. Dazu wählt man den Menü-Knopf aus.

Über die Navigationsleiste gelangt man schnell und einfach auf die Startseite der Anwendung, zur Übersicht über alle Caches, zur TOP10-Rangliste und zu weiteren Funktionen. Im Folgenden wird anhand der Beispielbilder erklärt, welcher Menü-Punkt Sie wohin weiterleitet.

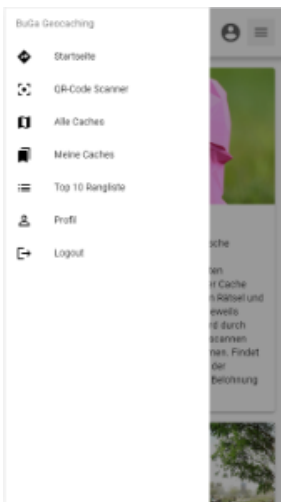
Diese weiteren Funktionen sind abhängig davon, ob man zur Zeit angemeldet ist oder nicht.

Sind Sie gerade nicht angemeldet, erhalten Sie folgende Ansicht der Navigationsleiste (hier in der Mobilansicht):



1. Startseite: Sie werden zur Startseite der Anwendung weitergeleitet.
2. Alle Caches: Sie werden zur Übersicht aller vorhandenen Caches weitergeleitet. (Mehr dazu in [2. Caches](#))
3. TOP 10 Rangliste: Sie werden zur Rangliste weitergeleitet, die die 10 Spieler mit den meisten Gesamtpunkten zeigt. (Mehr dazu in [3. TOP10-Rangliste](#))
4. Registrieren: Sie werden zu einem Registrierungsformular weitergeleitet, um sich ein eigenes Konto im zentralen Usermanagement anzulegen. (Mehr dazu in [4. Konto anlegen](#))
5. Login: Sie werden zum Anmeldeformular weitergeleitet, in welchem Sie, sollten Sie bereits ein existierendes Konto haben, sich einfach anmelden können. (Mehr dazu im Kapitel "Für Cachende" in [1. Anmelden](#))

Wenn Sie aktuell angemeldet sind, werden Sie diese Ansicht erhalten:



1. Startseite: Sie werden zur Startseite der Anwendung weitergeleitet.
2. QR-Code Scanner: Sie werden zum QR-Code Scanner weitergeleitet. (Mehr dazu im Kapitel "Für Cachende" in [2. Cache starten](#))
3. Alle Caches: Sie werden zur Übersicht aller vorhandenen Caches weitergeleitet (Mehr dazu in [2. Caches](#)).
4. Meine Caches: Sie werden zur Liste angefangener und beendeter Caches weitergeleitet. (Mehr dazu im Kapitel "Für Cachende" in [Angefangene und beendete Caches](#))

5. TOP 10 Rangliste: Sie werden zur Rangliste weitergeleitet, die die 10 Spieler mit den meisten Gesamtpunkten zeigt. (Mehr dazu in [3. TOP10-Rangliste](#))
6. Profil: Sie werden zu Ihrem eigenen Profil weitergeleitet. (Mehr dazu im Kapitel "Für Cachende" in [4. Das Profil und Teams](#))
7. Logout: Sie melden sich von Ihrer aktuellen Sitzung ab und werden zum Anmeldeformular weitergeleitet.

II. Für Cachende

Es wird dringend empfohlen das Kapitel "Für Besuchende" [4. Konto anlegen](#) vorher zu lesen und zu bearbeiten, bevor Sie weiterlesen, sollten Sie dies noch nicht getan haben.

Inhalt:

1. [Anmelden](#)
2. [Cache starten](#)
3. [Cache durchführen](#)
4. [Das Profil und Teams](#)

Zusatzkapitel:

- [Abmelden](#)
- [Angefangene und beendete Caches](#)

1. Anmelden

Hierfür ist ein registriertes Konto im zentralen Usermanagement nötig. Wenn Sie [4. Konto anlegen](#) im Kapitel "Für BuGa-Besuchende" abgeschlossen haben, sollten Sie ein solches Konto haben. Außerdem wenn Sie bereits ein Konto auf BuGa-FlowerTours oder BuGa-Blumenwiki haben, können Sie dieses hier auch verwenden.

Sollten Sie sich gerade auf GeoCaching registriert haben, sind die folgenden Schritte für Sie nicht sichtbar, da Sie bereits angemeldet sind, allerdings werden Sie nach spätestens 24 Stunden abgemeldet, daher wird es empfohlen dennoch die folgenden Informationen genau zu lesen. (Mehr dazu in [Abmelden](#))

Auf das Anmeldeformular gelangen Sie auf zwei verschiedenen Wegen:

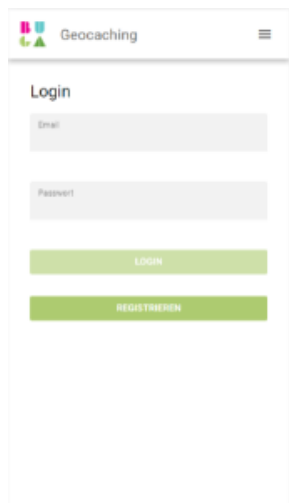
1. Sie wählen "Login" in der Navigationsleiste aus
2. Sie wählen "Starten" in einem aufgeklappten Cache in der "Alle Caches"-Übersicht aus

Geben Sie Ihre registrierte E-Mail-Adresse und Ihr Passwort, das Sie bei der Konto-Erstellung vergeben haben, jeweils in die dafür vorgesehenen Textfelder ein. Erst wenn beide Felder ausgefüllt sind, ist der "Login"-Knopf auswählbar.

Sollten die Daten inkorrekt sein, wird dies als Fehlermeldung ausgegeben.

Wenn die eingegebenen Daten korrekt sind, werden Sie auf die "Alle Caches"-Übersicht weitergeleitet und Sie können einen Cache starten. (Mehr dazu in [2. Cache starten](#))

Mobilansicht des Anmeldeformulars:



The image shows a mobile application interface for GeoCaching. At the top, there is a header bar with the GeoCaching logo on the left and a hamburger menu icon on the right. Below the header, the word "Login" is displayed. There are two input fields: "Email" and "Passwort". Below these fields are two green buttons: "LOGIN" and "REGISTRIEREN".

2. Cache starten

Bitte beachten Sie, dass Sie mit einem Konto, das in dem zentralen Usermanagement registriert ist, angemeldet sein müssen, um folgende Schritte durchführen zu können.

Ein Cache kann auf verschiedenen Wegen gestartet werden:

1. Sie wählen "Starten" in einem aufgeklappten Cache in der "Alle Caches"-Übersicht aus (sowohl in der Liste als auch auf der Karte möglich)
2. Sie finden auf der BuGa eine Startstation und scannen den QR-Code mit dem "QR-Code Scanner", der in der Navigationsleiste zu finden ist

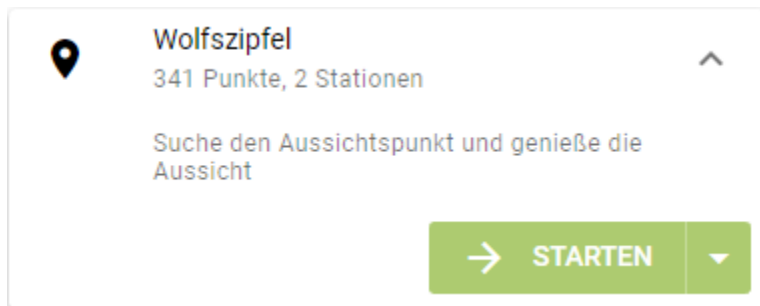
Wenn Sie, wie in 2. beschrieben die Startstation bereits gefunden und gescannt haben, können Sie direkt das nächste Kapitel lesen: [3. Cache durchführen](#)

Sollten Sie über den "Starten"-Knopf einen Cache gestartet haben, werden Sie zu einer Informationsseite weitergeleitet, die erklärt, wie GeoCaching auf der BuGa funktioniert.

Im oberen Teil des Bildschirms wird eine Karte angezeigt, auf der die Startstation, die es zu finden gilt, markiert ist. Unterhalb der Karte befindet sich der Name des Caches (im u.g. Beispiel "Wolfszipfel"). Weiter darunter befindet sich die Anleitung, wie GeoCaching funktioniert. Ganz unten wird ein "QR-Code scannen"-Knopf angezeigt. Wenn dieser ausgewählt wird, aktiviert sich die Kamera und man kann einen QR-Code scannen, sogar ohne einen installierten QR-Code Scanner auf dem Smartphone zu haben.

Bitte beachten Sie, dass der QR-Code Scanner Probleme hat einzelne QR-Codes zu erkennen, wenn mehrere nebeneinander liegen.

Der "Starten"-Knopf in einem aufgeklappten Cache:



Mobilansicht der Cache starten-Informationseite:



3. Cache durchführen

Bitte beachten Sie, dass Sie mit einem Konto, das in dem zentralen Usermanagement registriert ist, angemeldet sein müssen, um folgende Schritte durchführen zu können.

Sie sollten [2. Cache starten](#) durchgelesen haben, bevor Sie hier weiterlesen.

Nachdem Sie die Startstation des gestarteten Caches gefunden haben, beginnt das richtige GeoCaching.

Die Karte zentriert auf die zuletzt gefundene Station, diese ist mit einem roten Marker gekennzeichnet (siehe Mobilansicht Teil 1). Allerdings werden auch alle vergangenen Stationen angezeigt, diese Marker sind grau (siehe Mobilansicht Teil 2).

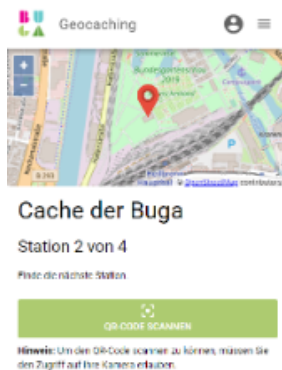
Unter dem Namen des Caches steht, welche Station man gerade gefunden hat und wie viele Stationen noch zu finden sind. Darunter befindet sich das Rätsel, welches zur nächsten Station, bzw. zum nächsten QR-Code, führen soll.

Der "QR-Code scannen"-Knopf funktioniert genauso, wie in [2. Cache starten](#).

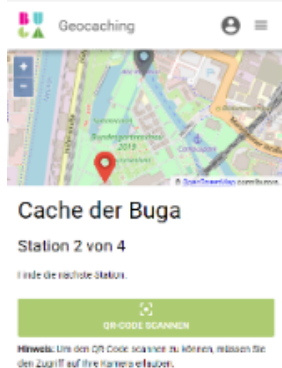
Nachdem alle Stationen, bzw. QR-Codes, gefunden und gescannt wurden, gelangt man zur Endansicht (siehe Mobilansicht Teil 3). Ihrem Konto werden Rangpunkte gutgeschrieben, sodass Ihr Rang in der TOP10-Rangliste steigt. Sie erhalten einen motivierenden Spruch oder eine Weisheit als Belohnung und können sich entweder zur TOP10-Rangliste oder zu "Alle Caches"-Übersicht navigieren lassen.

Ein Cache muss nicht an einem Tag vollständig durchgeführt werden, man kann Caches auch pausieren und an einem anderen Tag, auf der man auf der BuGa ist, fortfahren. Der Fortschritt wird gespeichert. Außerdem können Sie eine Belohnung erneut ansehen, wenn Sie dies möchten. (Mehr dazu in [Angenehme und beendete Caches](#))

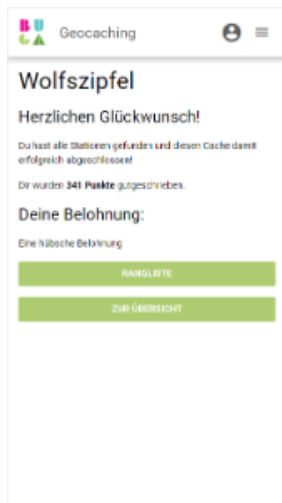
Mobilansicht (Teil 1):



Mobilansicht (Teil 2):



Mobilansicht (Teil 3):



4. Das Profil und Teams

Bitte beachten Sie, dass Sie mit einem Konto, das in dem zentralen Usermanagement registriert ist, angemeldet sein müssen, um folgende Schritte durchführen zu können.

Auf das Profil gelangt man auf verschiedenen Wegen:

1. Man wählt "Profil" in der Navigationsleiste aus
2. Man wählt das "Profilbild"-Icon aus, das stets rechts oben neben dem Menü-Icon ist, welches das Navigationsmenü öffnet
3. Man wählt seine eigene E-Mail-Adresse in der TOP10-Rangliste aus

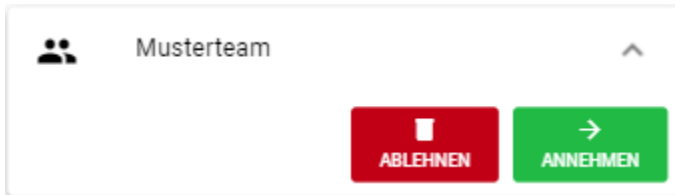
Im Profil wird Ihr bei der Registrierung eingegebener Name angezeigt, darunter Ihre E-Mail-Adresse, Ihren aktuellen Rang in der TOP10-Rangliste und Ihren Teamnamen, falls Sie aktuell in einem Team sind. Teamnamen sind einzigartig, sodass man keine 2 Teams mit dem gleichen Namen finden wird.

Wenn Sie sich aktuell in keinem Team befinden, befindet sich darunter ein Textfeld zur Eingabe eines Teamnamen zur Erstellung eines eigenen Teams. Allerdings kann auch ein anderer Cachender Sie zu seinem Team einladen. Solche Teameinladungen finden Sie unter "Teameinladungen" aufgelistet, rechts daneben ein Knopf zur Annahme einer Einladung. Man kann niemanden einladen, der kein Konto im zentralen Usermanagement hat. (Wie man sich ein solches Konto anlegen kann, finden Sie hier heraus: [4. Konto anlegen](#))

Nachdem man einem Team beigetreten ist, erhält man Zugang zu einem weiteren Tab "Team". Sie können keine neuen/weiteren Einladungen empfangen bzw. annehmen, solange Sie Mitglied in einem Team sind.

Im "Team"-Tab kann für das gesamte Team ein Status gesetzt werden, welcher von jedem Teammitglied gelesen werden kann, hier kann man sich zum gemeinsamen Cachen verabreden o.Ä.

Darunter ist eine Liste aller Teammitglieder, wobei sich weiter darunter ein Textfeld befindet zur Eingabe einer E-Mail-Adresse eines Cachenden, den Sie in Ihr Team einladen möchten. Nach Auswahl des "Einladen"-Knopfes, rechts neben dem Textfeld, wird der Cachende eine Einladung in seinem Profil vorfinden können, die dieser annehmen oder ablehnen kann, nachdem er das Team aufgeklappt hat.



Am unteren Rand befindet sich der rote "Team verlassen"-Knopf, mit welchem Sie Ihr aktuelles Team verlassen können. Wenn keine Cachende in einem Team verbleiben, wird die Gruppe gelöscht.


Mobilansicht eines Beispiel-Profiles, wenn dieser aktuell in keinem Team ist:



Mobilansicht eines Beispiel-Teams, das Test Cachender erstellt hat:

 Geocaching




 PROFIL

 TEAM

Musterteam

Unser Rang -

Unser Status: 

Teammitglieder

Name	Rang
David Cichewski	11

Nutzer einladen

E-Mail des Nutzers

EINLADEN

TEAM VERLASSEN

Abmelden

Man kann über verschiedene Wege von seiner aktuellen Sitzung abgemeldet werden:

1. Nach 24 Stunden ab dem Zeitpunkt der Anmeldung wird die Sitzung automatisch beendet
2. Man wählt über die Navigationsleiste "Logout" aus

Bitte beachten Sie, dass man nicht unangemeldet einen Cache durchführen kann.

Angefangene und beendete Caches

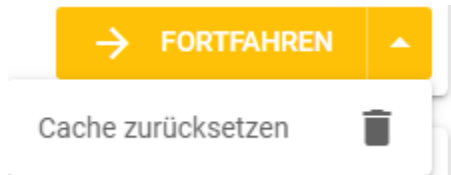
Bitte beachten Sie, dass Sie mit einem Konto, das in dem zentralen Usermanagement registriert ist, angemeldet sein müssen, um folgende Schritte durchführen zu können.

In der Navigationsleiste kann "Meine Caches" ausgewählt werden. Man wird zu einer Cache-Übersicht weitergeleitet, die im Tab "Angefangen" alle gestarteten Caches anzeigt, und im Tab "Beendet" alle bereits abgeschlossenen Caches anzeigt.

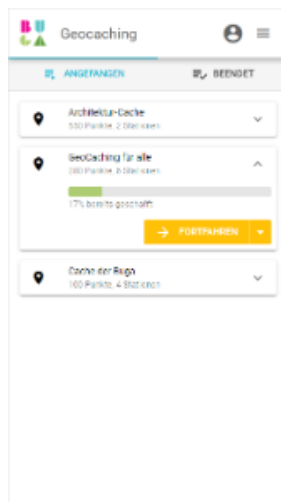
In beiden Tabs können die Caches aufgeklappt werden, wie es auch in der "Alle Caches"-Übersicht möglich ist. (Mehr dazu in [2. Caches](#))

Beim Aufklappen eines angefangenen Caches, erhält man einen Fortschrittsbalken, der anzeigt, wie viele Stationen in diesem Cache bereits gefunden wurden.

Unterhalb des Fortschrittsbalken kann man den Cache nun mit dem "Fortfahren"-Knopf weiter durchführen. Im Dropdown-Menü, welches rechts neben dem "Fortfahren"-Knopf bzw. bei den beendeten Caches rechts vom "Belohnung"-Knopf, kann man einen Cache zurücksetzen, sodass man ihn erneut durchführen kann:

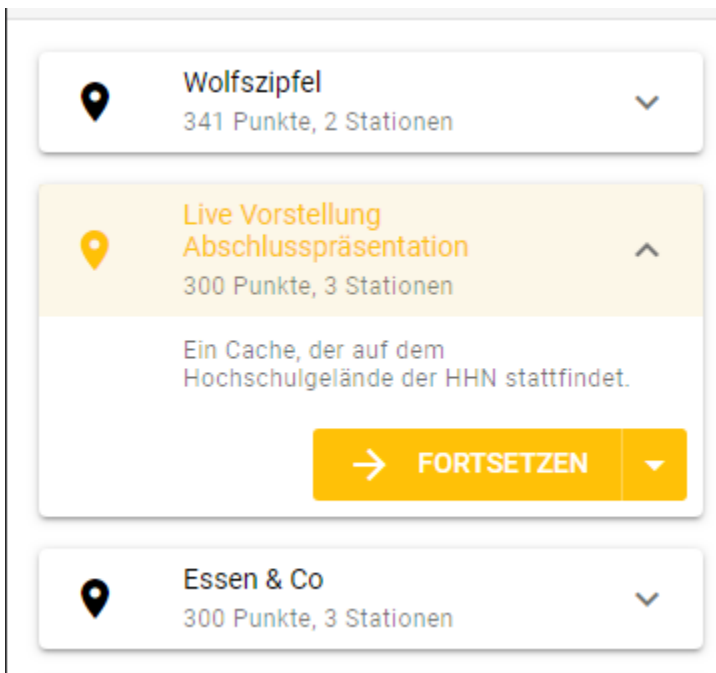


Mobilansicht:

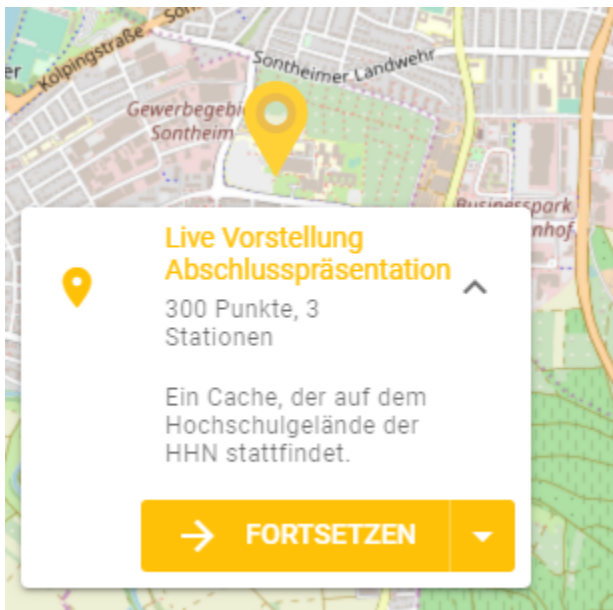


Alternativ lässt sich der Cache unter dem "Alle Caches"-Tab sowohl über die Liste als auch über die Karte fortführen.

Mobilansicht Liste:



Mobilansicht Karte:



III. Für Administrierende

Administrierende haben Zusatzfunktionen. Um diese nutzen zu können muss man sich als Administrierenden anmelden. Die Anmeldedaten erhalten Sie, falls sie Ihnen nicht vorliegen, von anderen Administrierenden.

Wie man sich an- und abmeldet finden Sie im Kapitel "Für Cachende" in [1. Anmelden](#)

Inhalt:

1. [Cache erstellen](#)
2. [Station erstellen](#)
3. [Caches bearbeiten / löschen](#)
4. [Rangpunkte zurücksetzen](#)

1. Cache erstellen

Als Administrierender können Sie Caches in der Cache Übersicht anlegen. (Mehr dazu im Kapitel "Für Besuchende" in [2. Caches](#))

Um einen neuen Cache anlegen zu können, müssen Sie als Administrierender eingeloggt sein und unter "Alle Caches" den Knopf "+ neuer Cache" auswählen:



Dieser Knopf befindet sich unter der Liste an bereits verfügbaren Caches, ggf. müssen Sie scrollen.

Nach Auswahl des o.g. Knopfes erscheint das Formular zur Erstellung eines neuen Cache.

Sie werden aufgefordert dem neuen Cache einen Namen zu geben, der für alle Nutzenden in der Cache Übersicht angezeigt wird.

Der Punktwert, der danach erfragt wird, ist die Rangpunktebelohnung die man bei Abschluss eines Caches erhält. (Mehr dazu im Kapitel "Für Cachende" in [3. Cache durchführen](#))

Für die Beschreibung ist ein Textfeld vorgesehen. In diesem kann man den Text auch **fett**, *kursiv* oder unterstrichen eintragen. Die Beschreibung ist für Nutzende sichtbar, wenn sie einen Cache in der Übersicht auswählen und aufklappen.

Die Belohnung wird dem Cachenden angezeigt, wenn er die letzte Station eines Cache gefunden hat. Daher sollte die Belohnung ein kurzer Spruch zur Motivation oder mit Ratschlägen sein.

Ein Cache besteht aus mehreren Stationen, mindestens allerdings zwei, darauf wird auch im Formular hingewiesen. Eine Station kann über den Knopf "Station hinzufügen +" hinzugefügt werden. (Mehr dazu in [2. Station erstellen](#))

Eine Endstation wird automatisch generiert, allerdings muss man die Koordinaten dieser Station noch entsprechend anpassen. (Mehr dazu in [2. Station erstellen](#))

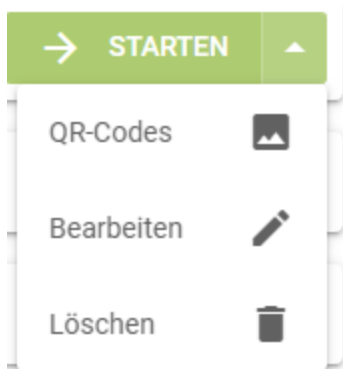
Sollten Fehler bei der Eingabe passiert sein, werden auf diese bei Auswahl von "Cache speichern" hingewiesen. Wenn alle Felder korrekt ausgefüllt sind, lässt sich der Cache speichern.

Wenn der Cache nicht gespeichert werden soll, kann man die Erstellung eines neuen Caches abbrechen, indem man den roten "Abbrechen"-Knopf auswählt. Man gelangt zurück zu "Alle Caches", welche unverändert ist.

Nachdem der Cache gespeichert wurde, wird man zurück in "Alle Caches" weitergeleitet. In der Übersicht ist nun bereits der neue Cache sichtbar und steht allen Cachenden zur Verfügung.




Da die Cachenden zur Durchführung des neuen Caches auch QR-Codes brauchen die sie scannen können, muss man diese auch nach Cache-Erstellung an den Positionen der gespeicherten Stationen verstecken.

Die QR-Codes können hierfür bereits in der Anwendung generiert werden. Hierzu wählen Sie den neu erstellten Cache aus und klappen diesen auf. Neben dem "Starten" Knopf befindet sich ein Dropdown-Menü, in welchem "QR-Codes" zur Verfügung steht:



Dort sind alle nötigen QR-Codes aller Stationen verfügbar.

Mobilansicht des Formulars (obere Hälfte):

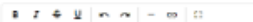

Geocaching



Neuen Cache erstellen

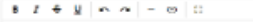
Name

Zustimmung
0

Beschreibung



Belohnung



Stationen

Hinweis: Ein Cache muss mindestens aus zwei Stationen bestehen (inklusive der Endstation).

STATION HINZUFÜGEN +

Mobilansicht des Formulars (obere Hälfte):

Belohnung



Stationen

Hinweis: Ein Cache muss mindestens aus zwei Stationen bestehen (inklusive der Endstation).

STATION HINZUFÜGEN +

Endstation

1

Keine
Endstation
Koordinaten: 49.1474100, 9.2044202




CACHE SPEICHERN

ABBRUCHEN

2. Station erstellen

Wenn Sie eine Station erstellen möchten, müssen Sie einen neuen Cache erstellen und "Station erstellen +"-Knopf auswählen. (Mehr dazu in [1. Cache erstellen](#))

In das Textfeld wird das Rätsel eingetragen das zur nächsten Station führen soll. Eine Endstation wird niemals Rätsel enthalten, da nach einer Endstation keine weitere Station folgt zu der ein Rätsel führen könnte.

Unter dem Rätsel wird die aktuelle Station auf der Karte markiert, hierzu gibt es 3 Eingabemöglichkeiten:

1. In den Textfeldern unterhalb der Karte die Koordinaten händisch eintragen. Der Breitengrad muss hierbei zwischen 49 und 50 liegen, der Längengrad zwischen 9 und 10. Bitte beachten Sie, dass die Nachkommastellen mit einem Punkt (.) getrennt werden und nicht wie im Deutschen mit einem Komma (,).
2. Die Karte ziehen und skalieren. Der Marker bleibt immer zentriert auf der Karte, sobald man nach dem Ziehen der Karte loslässt, wird der Marker an die zentrale Position gesetzt.
3. Die eigene Position nutzen. Mit der Auswahl des GPS-Knopfes, rechts von den beiden Textfeldern zur Eingabe von Breiten- und Längengrad, wird, sofern der Anwendung Zugriff auf die Standortinformationen des Nutzers gewährt wurde, der aktuelle Standort als Position der Station gesetzt. TIPP: Man kann trotzdem die Karte ziehen und skalieren, sodass der Marker präziser an der gewünschten Position steht.

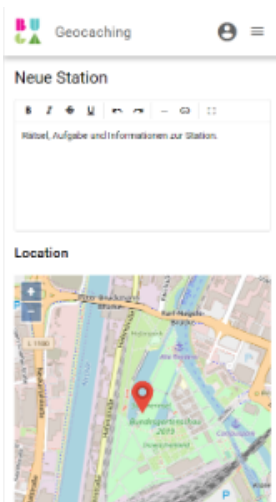
Nachdem das Rätsel zum Finden der nächsten Station und die Position der aktuellen Station eingetragen sind, kann die Station gespeichert werden. Hierzu den "Speichern"-Knopf auswählen. Wenn keine neue Station erstellt werden soll kann man diese Station auch wieder mit Auswahl des "Verwerfen"-Knopfes verwerfen.

Bitte beachten Sie, dass wenn Sie fehlerhafte Koordinaten eingetragen haben, erst bei Speichern des gesamten Caches eine Fehlermeldung ausgegeben wird. In dieser wird auch beschrieben in welcher Station die fehlerhaften Daten liegen.

GPS-Knopf:



Mobilansicht (obere Hälfte):



Mobilansicht (untere Hälfte):

Raster, Aufgabe und Informationen zur Station.

Location



Rechteckig
49.1474392

Quadrat
9.2065282



VERWERFEN

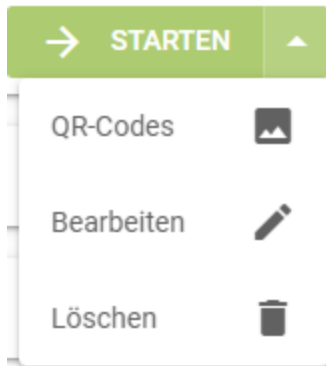
SPEICHERN



3. Caches bearbeiten / löschen

Wenn man als Administrierenden angemeldet ist, kann man bereits vorhandene Caches bearbeiten und auch löschen.

Hierzu muss man in der "Alle Caches"-Übersicht sein. Wählen Sie den zu löschenden oder zu bearbeitenden Cache aus und klappen Sie diesen Cache auf. Rechts neben dem "Starten"-Knopf befindet sich ein Dropdown-Menü:



Wählen Sie in diesem Menü die gewünschte Funktion aus.

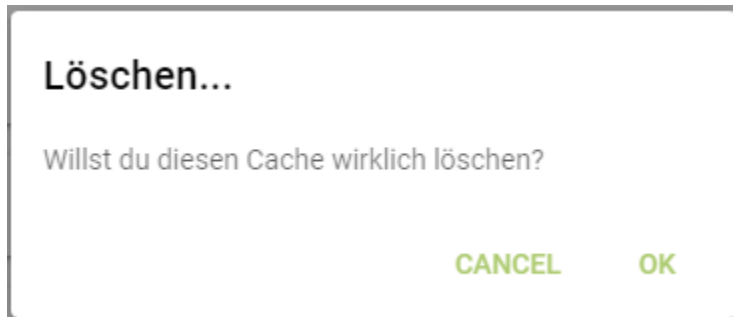
Bearbeiten:

Wenn Sie "Bearbeiten" auswählen, werden Sie in ein Formular weitergeleitet, die der "Cache-Erstellung" (mehr dazu in [1. Cache erstellen](#) und [2. Station erstellen](#)) ähnelt, allerdings sind alle Daten des ausgewählten Caches in die entsprechenden Felder eingetragen. Diese kann man nun editieren.

Nachdem alle gewünschten Datenänderungen vorgenommen wurden, können Sie den "Cache speichern"-Knopf auswählen. Sie werden nun zur Liste "Alle Caches" weitergeleitet.

Löschen:

Wenn Sie "Löschen" auswählen erscheint ein Dialogfeld, dass eine erneute Bestätigung erfragt, ob dieser Cache wirklich gelöscht werden soll:

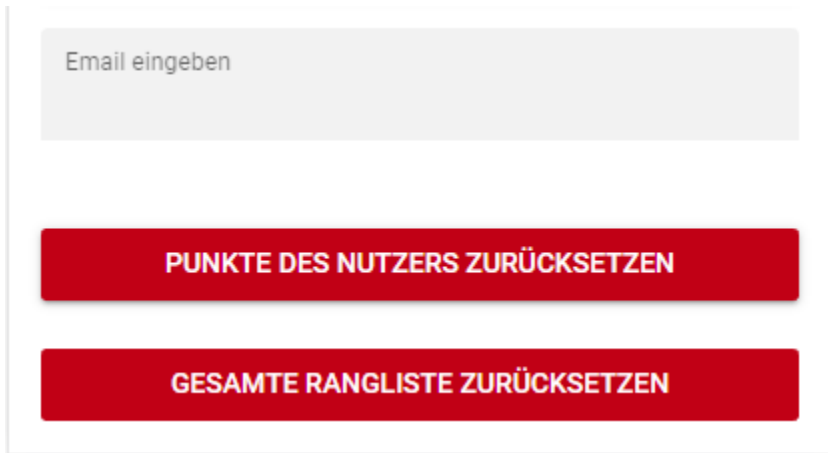


Nach dieser erneuten Bestätigung wird der Cache aus der Cache-Übersicht verschwinden, der Cache ist gelöscht. **ACHTUNG: DIESER VORGANG IST NICHT RÜCKGÄNGIG ZU MACHEN.**

4. Rangpunkte zurücksetzen

Wenn Sie als Administrierender angemeldet sind, können Sie die komplette TOP10-Rangliste oder die Punktzahl eines bestimmten Nutzers zurücksetzen. Im Nachfolgenden wird erklärt wie dies funktioniert:

Besuchen Sie die TOP10-Rangliste und scrollen Sie bis nach unten. In der Solo-TOP10 wird unter der letzten Position der Rangliste ein Textfeld und zwei Knöpfe zum Zurücksetzen der Rangliste angezeigt:



The screenshot shows a user interface for resetting points. It features a light gray rectangular input field at the top with the placeholder text "Email eingeben". Below this field are two prominent red buttons with white text. The first button is labeled "PUNKTE DES NUTZERS ZURÜCKSETZEN" and the second button is labeled "GESAMTE RANGLISTE ZURÜCKSETZEN". The entire interface is enclosed in a thin gray border.

Wenn Sie die Punktzahl eines bestimmten Nutzenden zurücksetzen möchten, können Sie dessen E-Mail-Adresse in das dafür vorgesehene Feld eintragen und den Knopf "Punkte des Nutzers zurücksetzen" auswählen. ACHTUNG: DIESER VORGANG KANN NICHT RÜCKGÄNGIG GEMACHT WERDEN.

Sollten Sie die gesamte Rangliste zurücksetzen wollen und damit die gesammelten Rangpunkte der Cachenden löschen wollen, wählen Sie den Knopf "Gesamte Rangliste zurücksetzen" aus. ACHTUNG: DIESER VORGANG KANN NICHT RÜCKGÄNGIG GEMACHT WERDEN.

Anmeldedaten

Anmeldedaten für einen Cachenden:

E-Mail: cachender@geocaching.de

Passwort: geocaching



























Anmeldedaten für einen Administrierenden:











E-Mail: admin@geocaching.de

Passwort: superadmin







Testberichte



















Demonstrative Tests (in Reihenfolge der Testdurchführung)

Test	Related Userstory	Executed by	Status	Started
 GEOGACH-188 - Test: Als Cachender will ich mein Profil mit Informationen über meinen Account ansehen können FERTIG	 GEOGACH-115 - Als Cachender will ich mein Profil mit Informationen über meinen Account ansehen könne FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-200 - Test: Als Administrierender will ich eine neue Station erstellen können FERTIG	 GEOGACH-44 - Als Administrierender will ich eine neue Station erstellen können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-203 - Test: Als Administrierender will ich eine neue Station erstellen können FERTIG	 GEOGACH-44 - Als Administrierender will ich eine neue Station erstellen können FERTIG	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
 GEOGACH-197 - Test: Als Administrierender will ich einen neuen Cache erstellen können FERTIG	 GEOGACH-7 - Als Administrierender will ich einen neuen Cache erstellen können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-199 - Test: Als Cachender /Administrierender will ich mich abmelden FERTIG	 GEOGACH-4 - Als Cachender/Administrierender will ich mich abmelden FERTIG	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
 GEOGACH-201 - Test: Als Cachender /Administrierender will ich mich anmelden FERTIG	 GEOGACH-3 - Als Cachender/Administrierender will ich mich anmelden FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-192 - Test: Als Cachender will ich einen Cache von der Website aus starten können FERTIG	 GEOGACH-1 - Als Cachender will ich einen Cache von der Website aus starten können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-208 - Test: Als Cachender will ich einen QR-Code scannen können, sodass ich einen Cache starte FERTIG	 GEOGACH-17 - Als Cachender will ich einen QR-Code scannen können, sodass ich einen Cache starte FERTIG	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
 GEOGACH-213 - Test Als Besuchender der BuGa will ich Informationen über das GeoCaching Angebot erhalten FERTIG	 GEOGACH-5 - Als Besuchender der BuGa will ich Informationen über das GeoCaching Angebot erhalten FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-214 - Test: Als Cachender/Besuchender will ich jederzeit die Rangliste einsehen können FERTIG	 GEOGACH-6 - Als Cachender/Besuchender will ich jederzeit die Rangliste einsehen können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-215 - Test: Als Cachender möchte ich andere zu meinem Team einladen können FERTIG	 GEOGACH-16 - Als Cachender möchte ich andere zu meinem Team einladen können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-218 - Test: Als Cachender will ich meine angefangenen und abgeschlossenen Caches sehen können FERTIG	 GEOGACH-9 - Als Cachender will ich meine angefangenen und abgeschlossenen Caches sehen können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-217 - Test: Als Administrierender will ich Caches entfernen können FERTIG	 GEOGACH-8 - Als Administrierender will ich Caches entfernen können FERTIG	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19

 GEOGACH-249 - Test: Als Cachender will ich Teams erstellen können FERTIG	 GEOGACH-13 - Als Cachender will ich Teams erstellen können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-222 - Test: Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, sodass ich dem Endpunkt des Caches näherkomme FERTIG	 GEOGACH-10 - Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, sodass ich dem Endpunkt des Caches näherkomme FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-223 - Test Als BuGa-Besuchender will ich mich registrieren FERTIG	 GEOGACH-11 - Als BuGa-Besuchender will ich mich registrieren FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-234 - Test: Als Cachender will ich die Startpunkte eines Caches auf der Karte sehen können FERTIG	 GEOGACH-15 - Als Cachender will ich die Startpunkte eines Caches auf der Karte sehen können FERTIG	Maximilian Leopold	PASS	24/June /19
 GEOGACH-230 - Test: Als Administrierender will ich Caches bearbeiten können FERTIG	 GEOGACH-18 - Als Administrierender will ich Caches bearbeiten können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-198 - Test: Als Cachender/Besuchender will ich eine Übersicht über alle Caches sehen können, um mir einen Überblick zu verschaffen FERTIG	 GEOGACH-2 - Als Cachender/Besuchender will ich eine Übersicht über alle Caches sehen können, um mir einen Überblick zu verschaffen FERTIG	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
 GEOGACH-232 - Test Als Cachender will ich nachdem ich einen Cache gefunden habe, eine Belohnung abholen können FERTIG	 GEOGACH-14 - Als Cachender will ich nachdem ich einen Cache gefunden habe, eine Belohnung abholen können FERTIG	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
 GEOGACH-307 - Test Als Anbieter von GeoCaches möchte ich die Bestenliste gelegentlich zurücksetzen, so dass neue Teilnehmer eine Chance haben in die besten 10 zu kommen FERTIG	 GEOGACH-187 - Als Anbieter von GeoCaches möchte ich die Bestenliste gelegentlich zurücksetzen, so dass neue Teilnehmer eine Chance haben in die besten 10 zu kommen FERTIG	Maximilian Leopold	PASS	26/June /19
 GEOGACH-308 - Test Als Cachender möchte ich Caches von der Karte aus aufrufen können FERTIG	 GEOGACH-20 - Als Cachender möchte ich Caches von der Karte aus aufrufen können FERTIG	Maximilian Leopold	PASS	26/June /19
 GEOGACH-309 - Test Als Cachender möchte ich meinen Standort auf der Karte sehen können FERTIG	 GEOGACH-19 - Als Cachender möchte ich meinen Standort auf der Karte sehen können FERTIG	Maximilian Leopold	PASS	26/June /19
 GEOGACH-314 - Test Als Cachender will ich den Fortschritt meines Cache verfolgen können FERTIG	 GEOGACH-12 - Als Cachender will ich den Fortschritt meines Cache verfolgen können FERTIG	Maximilian Leopold	PASS	26/June /19

Destruktive Tests (in Reihenfolge der Testdurchführung)

Test	Related Userstory	Executed by	Status	Started
 GEOGACH-195 - Test: Als Administrierender will ich eine neue Station erstellen können FERTIG	 GEOGACH-44 - Als Administrierender will ich eine neue Station erstellen können FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-205 - Test: Als Cachender will ich einen QR-Code scannen können, sodass ich einen Cache starte FERTIG	 GEOGACH-17 - Als Cachender will ich einen QR-Code scannen können, sodass ich einen Cache starte FERTIG	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
 GEOGACH-206 - Test: Als Cachender/Administrierender will ich mich anmelden FERTIG	 GEOGACH-3 - Als Cachender/Administrierender will ich mich anmelden FERTIG	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19

<p> GEOGACH-207 - Test: Als Administrierender will ich einen neuen Cache erstellen können(mit negativem Punktwert, korrekte Fehlermeldung wird ausgegeben) FERTIG</p>	<p> GEOGACH-7 - Als Administrierender will ich einen neuen Cache erstellen können FERTIG</p>	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
<p> GEOGACH-209 - Test: Als Cachender will ich einen Cache von der Website aus starten können (Nutzer ist nicht eingeloggt) FERTIG</p>	<p> GEOGACH-4 - Als Cachender will ich einen Cache von der Website aus starten können FERTIG</p>	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
<p> GEOGACH-210 - Test: Als Administrierender will ich einen neuen Cache erstellen können(ohne Stationen hinzuzufügen, korrekte Fehlermeldung wird ausgegeben) FERTIG</p>	<p> GEOGACH-7 - Als Administrierender will ich einen neuen Cache erstellen können FERTIG</p>	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
<p> GEOGACH-211 - Test: Als Cachender möchte ich andere zu meinem Team einladen können FERTIG</p>	<p> GEOGACH-16 - Als Cachender möchte ich andere zu meinem Team einladen können FERTIG</p>	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
<p> GEOGACH-224 - Test: Als Cachender will ich Teams erstellen können FERTIG</p>	<p> GEOGACH-13 - Als Cachender will ich Teams erstellen können FERTIG</p>	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
<p> GEOGACH-227 - Test Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, sodass ich dem Endpunkt des Caches näherkomme (Falscher QR-Code wird eingescannt) FERTIG</p>	<p> GEOGACH-10 - Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, sodass ich dem Endpunkt des Caches näherkomme FERTIG</p>	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
<p> GEOGACH-228 - Test: Als Cachender will ich Teams erstellen können FERTIG</p>	<p> GEOGACH-13 - Als Cachender will ich Teams erstellen können FERTIG</p>	Simon Stratemeier Maximilian Leopold	FAIL PASS	14/May/19 24/June /19
<p> GEOGACH-239 - Test: Als Cachender/Administrierender will ich mich abmelden FERTIG</p>	<p> GEOGACH-4 - Als Cachender/Administrierender will ich mich abmelden FERTIG</p>	Simon Stratemeier Maximilian Leopold	PASS PASS	14/May/19 24/June /19
<p> GEOGACH-240 - Test :Als BuGa-Besuchender will ich mich registrieren FERTIG</p>	<p> GEOGACH-11 - Als BuGa-Besuchender will ich mich registrieren FERTIG</p>	Maximilian Leopold	PASS	24/June /19

Projektmanagement

Change Requests

Am 24.06.2019 haben wir unseren Kunden einen Change Request gestellt. Dieser ist hier zu finden: [Decision Log](#)

Projektnormen und -spielregeln

Vor Beginn eines Sprints wurde ein Sprint Planning Meeting mit allen Mitgliedern der Gruppe durchgeführt, in dem geklärt wurde, wer welche Aufgaben übernimmt.

Wenn Probleme jeglicher Art auftraten, konnte man jederzeit nach Hilfe fragen. Es wurde stets zügig auf Fragen geantwortet und Lösungen gefunden, z.B. für Probleme zum Code, und bestimmter Funktionalitäten von Methoden.

Projektablauf

In den zweiten Sprint wurde lediglich eine User Story aus dem ersten Sprint mitgenommen, diese wurde dann auch sehr schnell implementiert.

Die User Stories aus dem zweiten und dritten Sprint konnten alle termingerecht abgeschlossen werden.

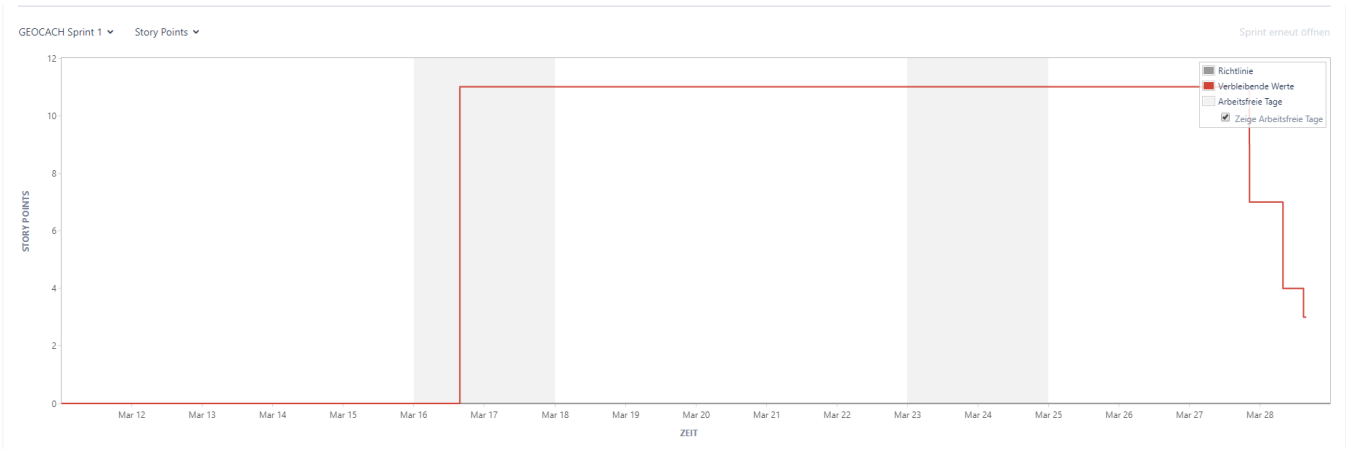
Erst wieder im vierten Sprint mussten User Stories mit in den nächsten Sprint übernommen werden. Hierbei handelte es sich um die User Stories, die die Karte der Lageplan Gruppe betrafen. Da die Lageplan Gruppe nicht die erforderlichen Funktionen bieten konnte, haben wir uns dazu entschlossen, eine eigene Karte einzubauen, die alle Funktionen beherrscht, die wir benötigen.

Auch im fünften Sprint mussten User Stories in den nächsten Sprint übertragen werden, da die Gruppe sich in die Kartentechnologie einarbeiten musste.

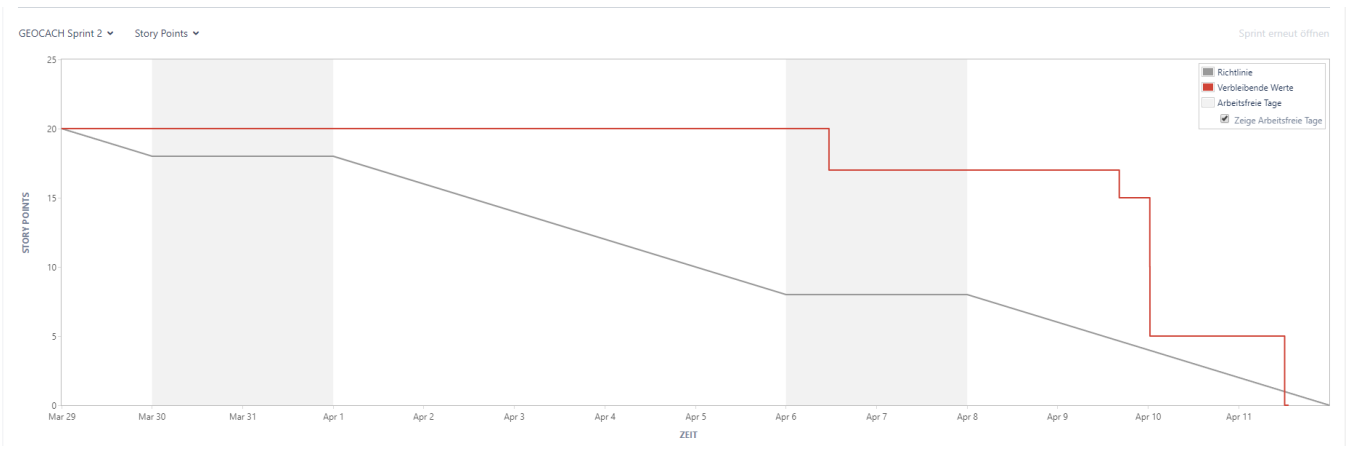
Im Laufe des sechsten Sprints wurden alle geplanten User Stories umgesetzt, sodass der Fokus des siebten Sprints komplett auf der Dokumentation lag.

Sprint Burndowns

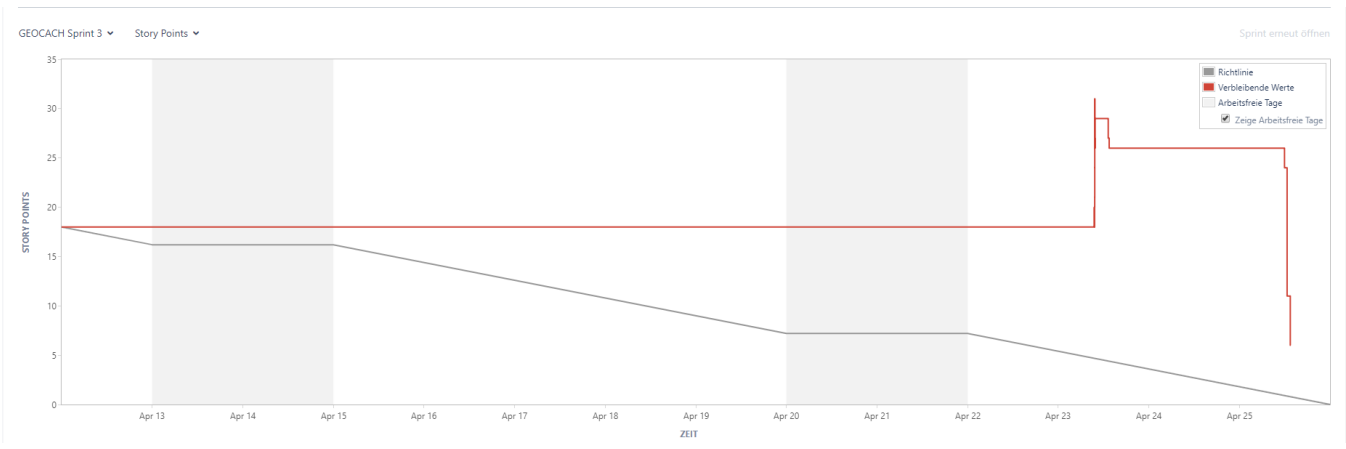
Sprint 1:



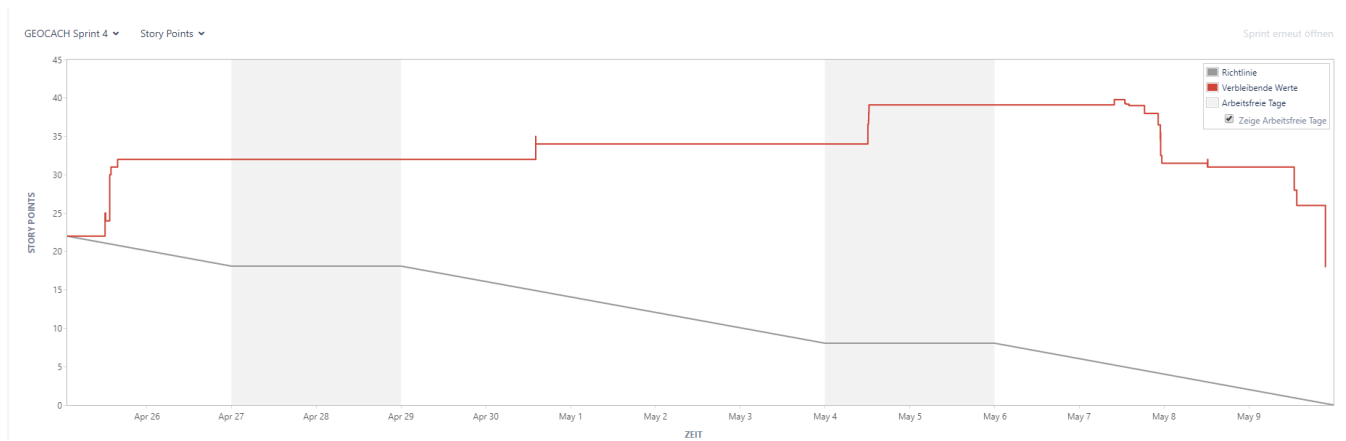
Sprint 2:



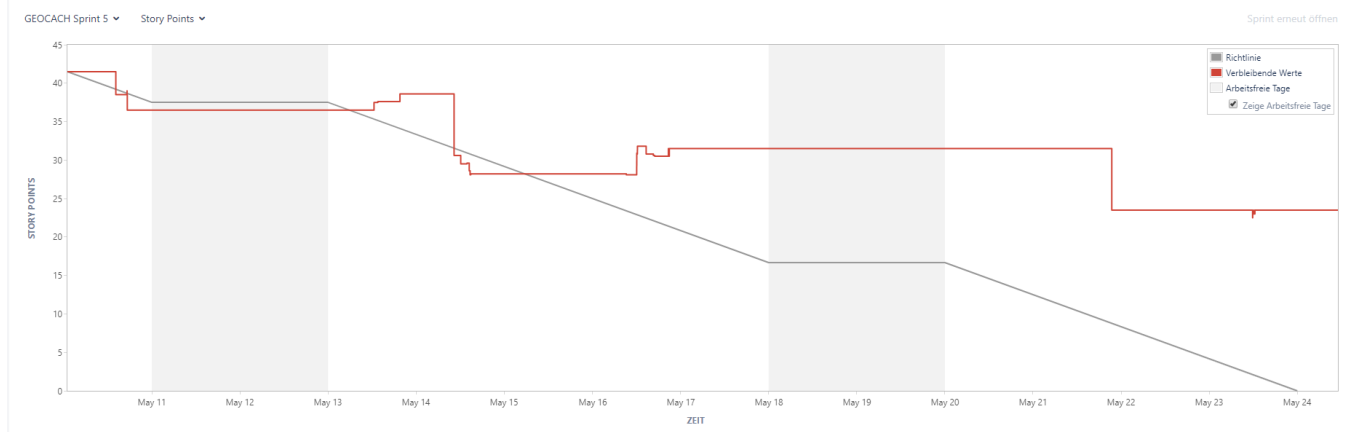
Sprint 3:



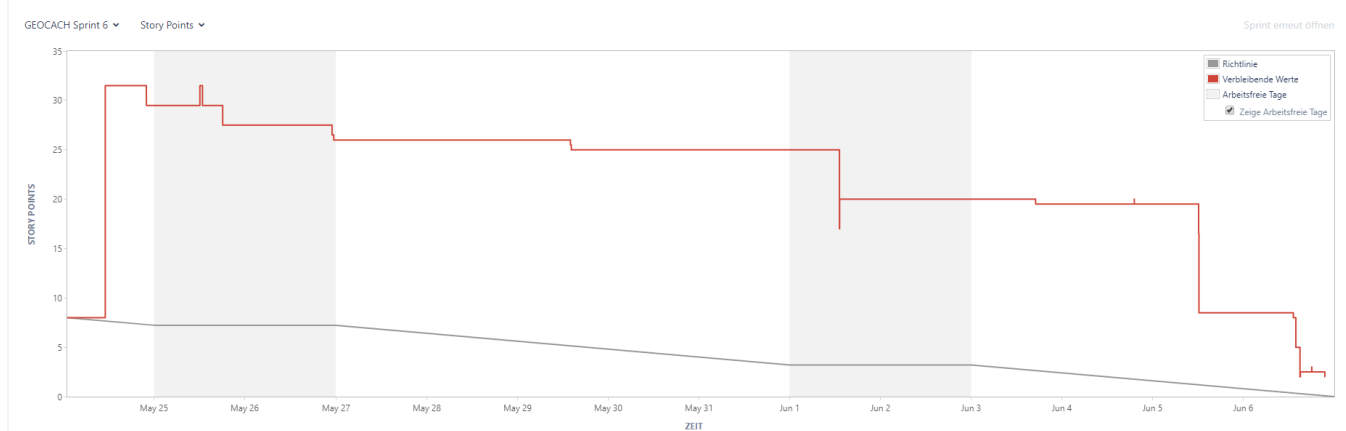
Sprint 4:



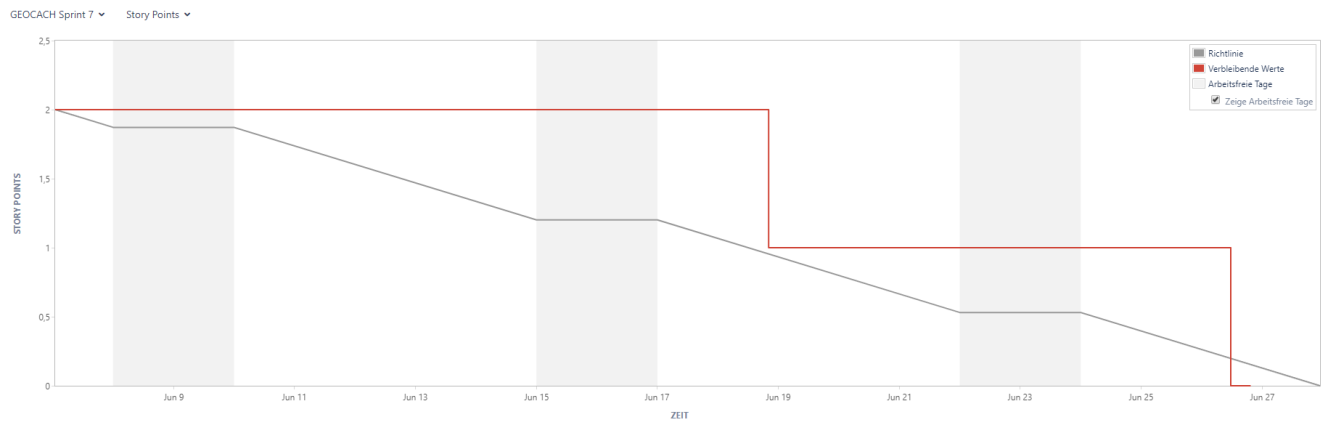
Sprint 5:



Sprint 6:


















Sprint 7: (Stand 26.06.2019)



Bug Statistik

Schlüssel	Zusammenfassung	S	Erstellt	Aktualisiert	Fällig	Bearbeiter	Autor	Priorität	Status	Lösung
GEOCACH-304	Email-Validierung funktioniert nicht richtig		Jun 24, 2019	Jun 26, 2019		Michael Dietz	Michael Dietz	=	FERTIG	Fertig
GEOCACH-272	Es werden unterschiedliche Positionen in der Rangliste und im Profil angezeigt		Mai 28, 2019	Mai 28, 2019		Michael Dietz	Michael Dietz	=	FERTIG	Fertig
GEOCACH-270	Karte sollte ausgeblendet werden, wenn Scanner aktiviert wird (CacheStart, StationView)		Mai 23, 2019	Mai 27, 2019		Timo Volkmann	Timo Volkman n	=	FERTIG	Fertig
GEOCACH-265	Teamranglistenpunkte zeigt Teams nicht in richtiger Reihenfolge an		Mai 23, 2019	Mai 23, 2019		Nicht zugewiesen	Michael Dietz	=	FERTIG	Fertig
GEOCACH-261	Rangliste zeigt unterschiedliche Ränge		Mai 16, 2019	Mai 23, 2019		Nicht zugewiesen	Timo Volkman n	=	FERTIG	Fertig
GEOCACH-258	Wenn die Koordinaten der Station nicht im richtigen Format eingegeben werden, muss eine korrekte Fehlermeldung ausgegeben werden		Mai 16, 2019	Mai 29, 2019		Maximilian Leopold	Michael Dietz	≡	FERTIG	Fertig
GEOCACH-257	Punktwert muss eine ganzzahlige positive Zahl sein, falls etwas anderes eingegeben wird, muss korrekte Fehlermeldung ausgegeben werden		Mai 16, 2019	Mai 29, 2019		Maximilian Leopold	Michael Dietz	=	FERTIG	Fertig
GEOCACH-256	Fehlermeldungen bei der Cacheerstellung verbessern		Mai 16, 2019	Mai 16, 2019		Michael Dietz	Michael Dietz	=	FERTIG	Fertig
GEOCACH-255	Breitengrad /Längengrad-Textreihenfolge ändern bei der Stationerstellung		Mai 16, 2019	Mai 16, 2019		Michael Dietz	Michael Dietz	=	FERTIG	Fertig
GEOCACH-254	Email bei registrierung darf nur erlaubte symbole verwenden. Ansonsten korrekte fehlermeldung ausgeben		Mai 16, 2019	Jun 06, 2019		Maximilian Leopold	Michael Dietz	=	FERTIG	Fertig
GEOCACH-250	Im Profil vor der Liste der Teameinladungen einen Titel einbauen wie "Teameinladungen", damit klar ist um was es sich bei der Liste handelt		Mai 16, 2019	Mai 16, 2019		Nicht zugewiesen	Michael Dietz	=	FERTIG	Fertig
GEOCACH-249	Sichtbar machen, dass ein User eingeloggt ist		Mai 16, 2019	Mai 16, 2019		Nicht zugewiesen	Michael Dietz	=	FERTIG	Fertig
GEOCACH-248	error occurs when someone leaves a team		Mai 16, 2019	Mai 16, 2019		Michael Dietz	Michael Dietz	=	FERTIG	Fertig

GEOCACH-247	API Aufruf für Cache zurücksetzen		Mai 15, 2019	Mai 23, 2019	Nicht zugewiesen	Timo Volkman n	=	FERTIG	Fertig
GEOCACH-245	Längen/Breitengrad Eingabe bei Cache tauschen		Mai 14, 2019	Mai 16, 2019	Nicht zugewiesen	Maximilian Leopold	=	FERTIG	Fertig
GEOCACH-244	Nach der Teamerstellung muss die Teamseite neu geladen werden		Mai 14, 2019	Mai 16, 2019	Nicht zugewiesen	Maximilian Leopold	=	FERTIG	Fertig
GEOCACH-234	switch "deleteCache" to DELETE		Mai 13, 2019	Mai 14, 2019	Nicht zugewiesen	Timo Volkman n	=	FERTIG	Fertig
GEOCACH-180	Verschiedenste kleine Bugfixes in verschiedenen API Calls, die für die Anzeige des Profils nötig sind.		Mai 10, 2019	Mai 14, 2019	Maximilian Leopold	Maximilian Leopold	=	FERTIG	Fertig
GEOCACH-173	Cacheübersicht: Cache neu starten nicht möglich, wenn bereits gestartet oder beendet.		Mai 09, 2019	Mai 14, 2019	Nicht zugewiesen	Timo Volkman n	=	FERTIG	Fertig
GEOCACH-172	Wenn Liste leer, anzeigen dass leer.		Mai 09, 2019	Mai 14, 2019	Timo Volkmann	Timo Volkman n	✓	FERTIG	Fertig
GEOCACH-169	Automatisches Logout, bzw. Weiterleitung zu Login, falls Token expired ist.		Mai 07, 2019	Mai 07, 2019	Timo Volkmann	Timo Volkman n	=	FERTIG	Fertig
GEOCACH-168	Problem bei "Meine Caches"		Mai 06, 2019	Mai 07, 2019	Timo Volkmann	Gerald Permanti er	⤴	FERTIG	Fertig
GEOCACH-167	Scannen eines QR-Codes führt zu unverständlicher roter Fehlermeldung		Mai 04, 2019	Mai 07, 2019	Timo Volkmann	Gerald Permanti er	⤴	FERTIG	Fertig
GEOCACH-166	QR Code Generator beim erstellen von Caches		Mai 04, 2019	Mai 08, 2019	Timo Volkmann	Maximilian Leopold	=	FERTIG	Fertig
GEOCACH-165	Nachdem QR Code gescannt wurde (beim Cachen) soll Kamera sofort zentriert sein (es muss automatisch nach unten gescrollt werden)		Mai 04, 2019	Mai 07, 2019	Timo Volkmann	Maximilian Leopold	=	FERTIG	Fertig
GEOCACH-164	Nach Abschluss des Caches soll nicht nur Seite mit drei Buttons angezeigt werden. Eine der Optionen soll standardmäßig angezeigt werden (Bsp. Belohnung)		Mai 04, 2019	Mai 08, 2019	Timo Volkmann	Maximilian Leopold	=	FERTIG	Fertig
GEOCACH-163	Username wird bei Registrierung abgefragt, aber nie benutzt		Mai 04, 2019	Mai 07, 2019	Timo Volkmann	Maximilian Leopold	=	FERTIG	Fertig
GEOCACH-161	Automatische Weiterleitung auf Registrierung, bzw. Login nach Registrierung oder sofortiger Login		Mai 02, 2019	Mai 07, 2019	Timo Volkmann	Maximilian Leopold	=	FERTIG	Fertig

GEOCACH-160	Richtige Fehlermeldungen anzeigen (Für den User aussagekräftig!)		Mai 02, 2019	Mai 14, 2019	Timo Volkmann	Maximilian Leopold		FERTIG	Fertig
GEOCACH-159	Email wird zweimal eingetragen automatisch bei Registrierung		Mai 02, 2019	Mai 07, 2019	Timo Volkmann	Maximilian Leopold		FERTIG	Fertig
GEOCACH-157	Fixed mistakes done in previous merges		Mai 02, 2019	Mai 04, 2019	Michael Dietz	Michael Dietz		FERTIG	Fertig
GEOCACH-156	fixed a bug with checkStation		Apr 30, 2019	Mai 04, 2019	Michael Dietz	Michael Dietz		FERTIG	Fertig
GEOCACH-144	fix nullpointer in Profile page (token)		Apr 27, 2019	Mai 04, 2019	Nicht zugewiesen	Timo Volkman n		FERTIG	Fertig
GEOCACH-142	Fehlermeldung Login fixen!		Apr 26, 2019	Mai 04, 2019	Nicht zugewiesen	Timo Volkman n		FERTIG	Fertig
GEOCACH-141	Punkteanzeige in Rangliste fixen!		Apr 26, 2019	Mai 07, 2019	Maximilian Leopold	Timo Volkman n		FERTIG	Fertig
GEOCACH-113	Alle Stationsaufrufe in einer methode vereinheitlichen		Apr 24, 2019	Apr 24, 2019	Michael Dietz	Michael Dietz		FERTIG	Fertig
GEOCACH-111	Delete Triggers wont work correctly		Apr 23, 2019	Apr 23, 2019	Nicht zugewiesen	Maximilian Leopold		FERTIG	Fertig
GEOCACH-109	Fixed a bug which caused createCache not to work anymore		Apr 18, 2019	Apr 18, 2019	Michael Dietz	Michael Dietz		FERTIG	Fertig
GEOCACH-107	Fix a bug which causes users not to be found correctly		Apr 18, 2019	Apr 18, 2019	Michael Dietz	Michael Dietz		FERTIG	Fertig
GEOCACH-67	Ein Cache soll nur erstellt werden, wenn dieser mindestens 2 Stationen hat		Apr 11, 2019	Apr 11, 2019	Michael Dietz	Michael Dietz		FERTIG	Fertig
GEOCACH-66	deleteCache soll auch die Stationen löschen, die zu dem Cache gehören		Apr 11, 2019	Apr 11, 2019	Michael Dietz	Michael Dietz		FERTIG	Fertig
GEOCACH-64	There can be multiple same database entries holding the same information in the "bearbeitet" table		Apr 09, 2019	Jun 21, 2019	Maximilian Leopold	Maximilian Leopold		FERTIG	Fertig
GEOCACH-61	Spring Securityconfig funktioniert nichtmehr richtig		Apr 09, 2019	Apr 09, 2019	Nicht zugewiesen	Maximilian Leopold		FERTIG	Fertig

43 Vorgänge

Liste verbleibender Bugs und Tasks

Alle User Stories, Tasks und Bugs wurden erledigt.

Es sind keine dem Team bekannten Bugs in der Anwendung vorhanden.

Dateilisten

[Create file list](#)

Titel	Ersteller	Geändert
Präsentationen	Timo Volkmann	vor etwa 7 Stunden
Releaseplan	Katharina Will	vor etwa 9 Stunden
Besprechung mit anderen Teams	Michael Dietz	vor etwa 9 Stunden
Product Backlog	Katharina Will	gestern um 9:25 PM
Projektauftrag	Maximilian Leopold	gestern um 8:38 PM

Besprechung mit anderen Teams

Im Ordner "Zusätzliche Dokumentationsdateien" (USB-Stick) liegen die Besprechungen mit anderen Teams auch noch als Word oder PDF Datei vor.

Datei	Geändert 
Microsoft Word Dokument Besprechung_Schnittstelle_Lageplan_Geocache.docx	Apr 08, 2019 by Michael Dietz
PDF-Datei Besprechung_Schnittstelle_Lageplan_Geocache.pdf	gestern um 9:22 PM by Maximilian Leopold

Ziehen Sie Dateien an diese Stelle, um sie hochzuladen, oder [Dateien suchen](#)

↓ [Alle herunterladen](#)

Schnittstellenbesprechung zwischen den Gruppen Lageplan und Geocache

Datum: 04.04.2019

Raum: A212

Anwesende Gruppe Lageplan:

Lucas Hinderberger, Patrick Gebhardt, Luca Göttle

Anwesende Gruppe Geocache:

Katharina Will, Maximilian Leopold, Timo Volkmann

Themen:

1. Darstellung von POIs von Geocaches im Lageplan
2. Einbindung eines Lageplans in den Webservice von Geocache

Gedanken:

(1):a) Die Geocache-POIs werden durch eine API von Geocache an den Lageplan gesendet. Das Format ist JSON. Notwendige Datenattribute können unten entnommen werden.

b) Die Geocaches werden in der Datenbank von Lageplan gespeichert. Notwendige Attribute können unten entnommen werden.

(2): Die Lageplangruppe bietet eine URL an, über die eine Karte mit einem einzelnen POI geladen werden kann. Dies erfolgt über eine REST-API.

Notwendige Parameter sind der Name des Geocaches, sowie einen Längen- und einen Breitengrad. Der POI soll auf der Karte zentriert dargestellt werden. Als Marker / Darstellungsform soll eine Stecknadel dienen. Diese Karte kann mittels eines iframes in den Webservice der Gruppe Geocache eingebunden werden.

Datenattribute:

Attribute der Tabelle *poi* der Gruppe Lageplan:

ID <PK>, INT *eindeutiger Primärschlüssel*

AdminID, INT *Der Ersteller des POIs*

GroupID, INT *Die Gruppe, zu der der POI gehört*

CategoryID, INT *Unterscheidung von Unterkategorien einer Gruppe*

Name, VARCHAR *Der Name des POIs – wird als Label auf der Karte angezeigt*

Description, VARCHAR *Eine Beschreibung – wird im INFO-Popup angezeigt*

Link, VARCHAR *Weiterführender Link*

Longitude, FLOAT *Der Längengrad*

Latitude, FLOAT *Der Breitengrad*

WheelchairAccessible, TINYINT *Ist der POI per Rollstuhl zu erreichen*

Attribute notwendig für die Darstellung auf der Karte nach (1) a:

Name *Der Name des POIs – wird als Label auf der Karte angezeigt*

Latitude *Der Längengrad*

Longitude *Der Breitengrad*

Icons:

Es ist ein Icon nötig für die Darstellung im Lageplan.

Offene Fragen:

Wo wird das Icon gespeichert?


Welches Icon wird verwendet?

Ist eine Darstellung mehrerer POIs auf der Karte zum Einbinden ebenfalls gewünscht?

Wenn der Nutzer im Lageplan auf einen Geocache klickt, öffnet sich ein Info-Popup – müssen Daten (bspw. Bilder) aus der Geocache-Datenbank dafür nachgeladen werden?

Usability Test

Im Ordner "Zusätzliche Dokumentationsdateien" (USB-Stick) liegt das Usability Test Protokoll auch noch als Word oder PDF Datei vor.

Datei	Geändert 
Microsoft Word Dokument GeoCaching_Usabilitytest.docx	vor etwa 9 Stunden by Maximilian Leopold
PDF-Datei GeoCaching_Usabilitytest.pdf	vor etwa 9 Stunden by Maximilian Leopold

Ziehen Sie Dateien an diese Stelle, um sie hochzuladen, oder [Dateien suchen](#)

 [Alle herunterladen](#)

GeoCaching Study Script

Damit ich nichts vergesse, lese ich dir hier kurz dieses Skript durch, damit du auf dem gleichen Stand wie alle anderen Tester bist.

Wir sind Studierende des Studiengangs Software Engineering Bachelor im vierten Semester. Es ist diese Woche die Blockwoche, eine Art Projektwoche, an unserer Hochschule und unser Schwerpunktfach in diesem Semester ist das Labor für Software Entwicklung. Wir entwickeln hier eine Anwendung für die Bundesgartenschau. Unser Thema diese Woche ist das Testen verschiedener Komponenten unserer Anwendung, in diesem Fall jetzt das Usability Testing, ein Test zur Nutzerfreundlichkeit. Unser Projekt hierbei ist eine Geocaching Anwendung auf der Bundesgartenschau.

Ich weiß nicht, ob du Geocaching bereits kennst, deshalb erkläre ich es dir kurz:

Geocaching ist eine Art moderne Schnitzeljagd. Ein Nutzer, in diesem Fall du, führt eine Schnitzeljagd durch, bei der du in einer bestimmten Reihenfolge QR Codes scannst. Um den nächsten QR Code zu finden, wird dir ein Hinweis zur Verfügung gestellt. Wenn du alle QR Codes einer Schnitzeljagd gefunden hast, hast du diese abgeschlossen und erhältst eine kleine Belohnung und Punkte für die globale Rangliste. Hast du das Grundkonzept verstanden? Sehr gut, weiter zu den Einzelheiten des Tests.

Was bei diesem kompletten Test wichtig ist, ist, dass es nicht darum geht alle Aufgaben fehlerfrei und in Bestzeit abzuschließen, sondern es für uns wichtig ist, dass wir sehen wie du die Anwendung benutzt. Wir testen hierbei nicht dich, sondern die Anwendung. Du kannst also nichts falsch machen und deine Fehler und Gedanken helfen uns am meisten weiter. Da für uns besonders deine Gedanken wichtig sind, bitte ich dich hiermit alles was du dir denkst laut auszusprechen, damit wir es notieren können.

Wenn du während der Durchführung irgendwelche Fragen hast, frag sie bitte einfach. Ich werde versuchen sie alle zu beantworten, teilweise ist es für uns aber auch wichtig zu wissen wie du die Aufgabe ohne meine Hilfe bewältigen würdest.

Ich werde dir dann gleich die verschiedenen Testfälle vorlesen und du wirst versuchen sie alle durchzuführen.

Damit du nicht dein eigenes Handy verwenden musst, gebe ich dir meins. Hier ist die Website bereits geöffnet und du kannst direkt loslegen.

Hast du bisher irgendwelche Fragen? Ok sehr gut.

Aufgaben

1. Legen Sie ein eigenes Konto an

	Optimal	1	2	3	4	5	6	7
Zeit	65s	60s	65s	110s	150s	87s	140s	80s
Klicks	8	7	13	15	30	18	17	24
Bedienfehler	0	0	0	1	0	0	0	0

Bemerkungen:

Proband 1: **Erst Startseite angesehen und rumprobiert**-> dann navigationsleiste gefunden

Ganz normale Registrierungswebseite. War erfolgreich.

Proband 3: **Sucht auf Startseite**. Geht über den Login zur Registrierung. Hat einen Namen bei Email-Adresse eingegeben.

Proband 4: **Sucht auf der Startseite**. In der Navigationsleiste gesehen und intuitiv. registrieren Knopf sehr klein. "unbekannter Fehler" geworfen -> wegen umlaut in email-adresse. Findet gut, dass wenn daten korrekt in felder eingegeben werden, Felder blau werden.

Proband 5: Automatische Ausfüllung des Formulars erwünscht.

Proband 6: Name: Vorname Nachname? Groß-/Kleinschreibung bei E-Mail ignorieren. Weiterleitung nach Registrierung ist unklar. **Sieht nicht dass er eingeloggt ist.**

Proband 7: **Was für ein Name? Username?** Registrierung erfolgreich

1. Führen Sie den Cache „Usability-Test“ durch

Was muss ich zuerst scannen?

Die erste Station, die du scannen musst, ist die, die hier vor dir liegt.

Wo kann ich einen Cache starten?

Versuche es mal über das Menü.

Wie kann ich einen QR Code scannen? Ich sehe keine Kamera.

Du kannst den Scanner über den Button 'QR Code scannen' starten.

Hier steht ich soll zu der auf der Karte angezeigten Position. Ich sehe jedoch keine Karte.

Diese Funktion ist noch nicht implementiert. Ignorier sie bitte einfach.

	Optimal	1	2	3	4	5	6	7
Zeit	80s	70s	80s	50s	100s	13s	126s	120s
Klicks	5	11	5	5	7	5	6	6
Bedienfehler	0	0	1	0	0	0	0	0

Bemerkungen:

Proband 1: QR-Scanner funktionierte nicht richtig (nur Querformat auf Android)

Proband 2: Hat etwas gedauert bis der QR-Code gescannt wurde. Zweiter QR-Code wurde schneller gescannt. Freude über abgeschlossenen Cache.

Proband 3: Liest aufmerksam die Instruktion. Keine Probleme. Vorschlag: Stichworte als Instruktion, weil es lang dauert zu lesen. Kann man auch wieder zurück? Design ist voll ansprechend. Sieht gut aus.

Proband 4: Das muss ich jetzt so darüber halten? Funktioniert. Was sind das für Punkte?

Proband 5: -

Proband 6: -

Proband 7: **aufpassen, dass nur ein QR-Code im Screen ist**

1. Besuchen Sie die Rangliste

Wie werden die Punkte für die Rangliste berechnet?

Jeder Cache hat eine bestimmte Anzahl von Punkten. Wenn du den Cache abschließt, erhältst du diese Punkte.

Wieso werde ich nicht auf der Rangliste angezeigt?

Es werden immer nur die Top 10 aller Spieler angezeigt.

	Optimal	1	2	3	4	5	6	7
Zeit	2s	4s	10s	1s	2s	4s	8s	10s
Klicks	1	1	1	1	1	1	2	4
Bedienfehler	0	0	0	0	1	0	0	0

Bemerkungen:

Proband 1: Keine Probleme.

Proband 2: Direkte Hinleitung vom Screen des beendeten Caches. Wenn es mehr Nutzer gibt, werden dann trotzdem nur 10 Nutzer angezeigt? Rangliste eher TOP10 nennen, da der Rang in der Profilsansicht zu finden ist. Evtl. als 11. Position auf der Rangliste die eigene Position setzen, falls man nicht in der TOP10 ist.

Proband 3: Keine Probleme. Etwas doof, dass alle die die gleiche Rangpunktzahl haben den gleichen Rang haben.

Proband 4: Keine Probleme. **Klickt auf Name**, gelangt auf Team-Ansicht aus Versehen. man kann swipen.

Proband 5: -

Proband 6: **Rangliste in Top 10 umbenennen**. Gesamtanzahl der Nutzer anzeigen.

Proband 7: -

1. Besuchen Sie Ihre Profilsansicht

	Optimal	1	2	3	4	5	6	7
Zeit	3s	5s	18s	3s	16s	5s	12s	8s
Klicks	2	2	7	2	3	2	2	4
Bedienfehler	0	0	1	0	0	0	0	0

Bemerkungen:

Proband 1: Keine Probleme.

Proband 2: Kann ich in der Rangliste auf eigenen Namen klicken? Nein.

Proband 3: Keine Probleme, intuitiv gefunden.

Proband 4: **Kann von der Rangliste nicht auf eigenen Namen klicken**. Gelangt über die Navigationsleiste drauf

Proband 5: -

Proband 6: **Ränge sind nicht korrekt: Profil steht 7 und in der Rangliste 8**

Proband 7: -

1. Erstellen Sie sich ein eigenes Team und laden den Nutzer mit der E-Mail volkmann@geocaching.de dazu ein.

Wen soll ich einladen?

Den Nutzer mit der E Mail volkmann@geocaching.de

Wie kann ich ein neues Team erstellen?

Such mal unter der Profilsansicht.

In der Profilsansicht ist ein Teamtab.

Ist der Name des Teams egal?

Ja, der Name muss lediglich unique sein.

Bin ich jetzt automatisch in diesem Team?

Ja

Ist der eingeladene User automatisch im Team?

Nein, er muss erst die Team Einladung annehmen.

	Optimal	1	2	3	4	5	6	7
Zeit	40s	75s	48s	40s	76s	50s	100s	82s
Klicks	7	8	7	10	17	9	11	11
Bedienfehler	0	0	0	0	2	0	0	0

Bemerkungen:

Proband 2: Team Erstellung ohne Probleme. Beim Erforschen die Funktion der Teameinladung intuitiv gefunden. **Verleitung auf "Team Verlassen" zu drücken**, weil er so rot und groß ist. Teamstatus ist cool.

Proband 3: Intuitiv gelöst. Schön, dass es bunt ist und Signalfarben verwendet wurden.

Proband 4: Name intuitiv eingegeben. Wollte Team erstellen, ohne Namen einzugeben, Fehlermeldung. Will Status setzen. Findet Nutzer einladen Funktion. Keine Probleme. **Sehe ich das irgendwo, dass ich jemanden eingeladen habe?** Keine Rückmeldung, wenn Nutzer eingeladen. Wollte nochmal einladen, Fehlermeldung provoziert.

Proband 5: Gute Fehlermeldung. Sofort verständlich. Bei Email Eingabe sollte auf Tastatur @ angezeigt werden.

Proband 6: hat Team-Tab nicht sofort gefunden. Pfeil nach rechts ist nicht ganz klar. Nutzer einladen anders gestalten. **Team verlassen Button hat verunsichert**. Mitgliederanzahl in Profilübersicht

Proband 7: -

1. Finden Sie Ihren vorhin abgeschlossenen Cache und rufen Sie die Belohnung erneut auf

Wo finde ich meine abgeschlossenen Caches?

Schau mal unter 'Meine Caches'

	Optimal	1	2	3	4	5	6	7
Zeit	5s	6s	18s	9s	20s	18s	13s	17s
Klicks	3	5	7	5	8	6	5	6
Bedienfehler	0	0	0	0	1	1	0	0

Bemerkungen:

Proband 1: Menü- Knopf ist etwas klein

Proband 2: **Problemlos gefunden**.

Proband 3: Intuitiv gefunden. **Keine Probleme**, keine Fragen.

Proband 4: Findet intuitiv den beendeten Cache. Aus versehen gewischt statt Button gedrückt.

Proband 5: Erst auf 'Alle Caches'

Proband 6: -

Proband 7: **Intuitiv gefunden**. Keine Probleme

1. Verlassen sie ihr Team wieder

Der Button 'Team verlassen' ist ausgegraut

Im Textfeld steht die Anweisung, was du machen musst.

	Optimal	1	2	3	4	5	6	7
Zeit	13s	15s	30s	10s	30s	18s	7s	24s
Klicks	5	8	8	6	5	5	3	7
Bedienfehler	0	0	3	0	0	0	0	0

Bemerkungen:

Proband 1: **Fehler beim Team verlassen**. Beim zweiten Mal geklappt.

Proband 2: Vorhin schon gesehen. "Problem with response" - Drück ok. Ein neues Team anzulegen funktioniert nicht. Funktioniert doch - Name war zu kurz, Rückmeldung kam allerdings erst als alles gelöscht wurde. Nun hat Team verlassen geklappt. Besteht das Team weiterhin, wenn der letzte Nutzer aus dem Team ausgetreten ist. Profil leer - Tab musste erneut angeklickt werden.

Proband 3: **Bekannter Fehler wieder aufgetreten**. Seite neu geladen, dann nicht mehr in Gruppe. Wieso sind die Profilinformationen nicht zentriert? Sehr leicht gestaltet.

Proband 4: Ist nett, dass nochmal gefragt wird. Was ist "cancel"? Wird ausprobiert. **Team verlassen-Fehler tritt erneut auf**.

Proband 5: Fehler tritt auf.

Proband 6: Fehler tritt auf.

Proband 7: Fehler tritt auf

1. Melden Sie sich ab, melden Sie sich als Administrator an. Die Anmeldedaten dafür sind: email: volkmann@geocaching.de pw: 0123456789

	Optimal	1	2	3	4	5	6	7
Zeit	25s	30s	50s	32s	38s	57s	48s	65s
Klicks	7	8	9	7	5	8	6	8
Bedienfehler	0	0	1	0	0	1	0	0

Bemerkungen:

Proband 1: Keine Probleme

Proband 2: Bei Logout schließt sich die Navigationsleiste, **kurz verwirrt, ob der Logout geklappt hat** und unpraktisch, um sich mit einem anderen Account anzumelden.

Proband 3: Keine Probleme

Proband 4: Keine Probleme

Proband 5: **E mail ist Case sensitive.**

Proband 6: Keine Probleme

Proband 7: Keine Probleme

1. Legen Sie einen neuen Cache an

Welche Koordinaten liegen innerhalb der Buga?

Breitengrad zwischen 9 und 10, Längengrad zwischen 49 und 50

Darf ich den Name frei wählen?

Ja

Wieviel Stationen muss ich erstellen?

Mindestens eine plus die Endstation

	Optimal	1	2	3	4	5	6	7
Zeit	60s	75s	100s	170s	210s	300s	107s	360s
Klicks	15	33	24	40	33	42	15	72
Bedienfehler	0	1	0	2	2	4	0	5

Bemerkungen:

Proband 1: Warum ist kein standardmäßig ein gültiger Wert eingetragen? Endstation muss extra Koordinaten geändert werden. Koordinaten sind nicht Allgemeinwissen. Koordinaten nicht zu ändern, führt zu Fehlermeldung.

Proband 2: Irgendwas bestimmtes eingeben? Hinweis gegeben, dass Koordinaten eingegeben werden müssen. Was macht der GPS-Knopf? Hinweis, dass Koordinaten bei Endstation auch geändert werden müssen. "War auch relativ einfach".

Proband 3: Hinweis mit Koordinaten gegeben. Hat GPS-Knopf ausprobiert - nichts ist geschehen. Hat Probiert ob ein Cache nur aus einer Endstation bestehen kann - Fehlermeldung provoziert. Nachdem eine Station angelegt wurde, hat dies funktioniert. Wäre es am Laptop nicht einfacher. Hat beides Vor- und Nachteile. Wie funktioniert das mit den QR-Codes? Sehr coole Funktion.

Proband 4: Gibt keine Zahl als Punktzahl an. Könnte der Text verschwinden, wenn man auf das Textfeld klickt bei Station Erstellung? Fehler, da kein Punkt, sondern Komma eingegeben wurde. Fehler, da Punktzahl keine Zahl. **Bessere Fehlermeldungen bitte!!**

Proband 5: Hat versucht auf Karte zu klicken. Längengrad und Breitengrad bisschen unverständlich. Nachfrage nach Längengrad, Breitengrad auf Buga.

Hat bei Koordinaten Punkt statt Komma benutzt. **Fehlermeldung unverständlich.** Leichte Probleme bei koordinaten erstellung. Formatierung unverständlich. Benötigt man Endstation? Längengrad, Breitengrad Label falsch herum.

Proband 6: keine Probleme.

Proband 7: **Längen- und Breitengrad bereiten große Probleme.** Wenn Koordinaten außerhalb der Bundesgartenschau liegen, sollte genauer gemeldet werden bei welcher Station dies so ist. Fehlgeschlagen, konnte nicht beendet werden (Kein Sichtkontakt zu Probanden). GPS-Knopf noch ohne Funktion.

1. Löschen Sie diesen Cache wieder und melden Sie sich ab

	Optimal	1	2	3	4	5	6	7
Zeit	12s	4s	15s	6s	28s	9s	22s	15s
Klicks	5	4	7	5	6	3	7	7
Bedienfehler	0	0	1	0	0	0	0	0

Bemerkungen:

Proband 1: Keine Probleme.

Proband 2: Gedrückt halten funktioniert nicht, dann ausklappen. Funktion gefunden. Keine Probleme.

Proband 3: Keine Probleme.

Proband 4: Welchen Cache hab ich erstellt? Erneute Nachfrage beim Löschen ist gut. **Buttons zu klein**, besonders in der Mobilansicht.

Proband 5: -

Proband 6: Stört dass man nicht sieht ob man eingeloggt ist und als wen

Proband 7: -

Metriken

1. Zeitmessung zur Erfüllung jeder individuellen Aufgabe
2. Anzahl der benötigten Klicks zur Erfüllung jeder individuellen Aufgabe
3. Anzahl Bedienfehler

Fragebogen

Link zum Fragebogen: (+)https://docs.google.com/forms/d/e/1FAIpQLSetQJmsuCBHpE-XEkXY3M_4nJmDlwD5G5f7fEhh4H3oVSG-Aw/viewform?usp=sf_link

Auswertung Fragebogen:

(+)https://docs.google.com/forms/d/1hD7qHzZNwil8_laP49We01hdkEL-kYsnOdXd73HG2TU/edit#responses

Rollen

Durchführer: Timo

Protokoll: Katharina

Präsentationen

Die Präsentationsdateien finden Sie im Ordner "Präsentationen" auf dem USB-Stick.


Datei	Geändert 
PDF-Datei LabSWSP_Release1_Präsentation_Volkmann_Christ.pdf	Mär 28, 2019 by Maximilian Leopold
PDF-Datei LabSWSP_Release2_Presentation_Will_Leopold.pdf	Jun 11, 2019 by Maximilian Leopold
PDF-Datei LabSWPS_Release3_Presentation_Dietz_Christ.pdf	vor etwa 7 Stunden by Maximilian Leopold

Ziehen Sie Dateien an diese Stelle, um sie hochzuladen, oder [Dateien suchen](#)

 [Alle herunterladen](#)

Product Backlog

Im Ordner "Zusätzliche Dokumentationsdateien" (USB-Stick) liegt der Product Backlog auch noch als Word oder PDF Datei vor.

Datei	Geändert 
PDF-Datei ProductBacklog.pdf	Apr 11, 2019 by Michael Dietz
Microsoft Word Dokument ProductBacklog.docx	gestern um 9:25 PM by Maximilian Leopold
PNG-Datei Übersicht Mockup.png imported from a Word document	gestern um 9:27 PM by Maximilian Leopold
PNG-Datei Stationen Detailansicht Mockup.png imported from a Word document	gestern um 9:28 PM by Maximilian Leopold
PNG-Datei Profil Mockup.png imported from a Word document	gestern um 9:28 PM by Maximilian Leopold
PNG-Datei Cache Ansicht Mockup.png imported from a Word document	gestern um 9:28 PM by Maximilian Leopold
PNG-Datei Cache Detailansicht Mockup.png imported from a Word document	gestern um 9:29 PM by Maximilian Leopold

Ziehen Sie Dateien an diese Stelle, um sie hochzuladen, oder [Dateien suchen](#)

 [Alle herunterladen](#)

Inhaltsverzeichnis

- 1 Ziel, Vision, Überblick**
- 2 Glossar**
- 3 Referenzen auf andere Dokumente**
- 4 Offene Fragen und grundlegende Entscheidungen**
 - 4.1 Offene Fragen
 - 4.2 Grundlegende Entscheidungen
- 5 User Stories**
- 6 Benutzerrollen und eingesetzte Software**
 - 6.1 Benutzerrollen
 - 6.2 Schnittstellen
 - 6.3 Eingesetzte Software
 - 6.4 Constraints
- 7 Datenhaltung**
 - 7.1 User
 - 7.2 Team
 - 7.3 Station
 - 7.4 Cache
- 8 Detaillierte Anforderungen**
 - 8.1 Cachender
 - 8.2 Administrierender
 - 8.3 Besuchender
- 9 Installation**
- 10 GUI-Entwürfe, Bedienoberfläche**
- 11 Geschätzter Aufwand in Story Points**

Ziel, Vision, Überblick

Ab dem 17. April 2019 öffnet die Bundesgartenschau in Heilbronn für 173 Tage ihre Pforten und wir haben uns zur Aufgabe gemacht, die BuGa in Heilbronn mithilfe eines Gamification Ansatzes für Interessenten schmackhaft zu machen. Geocaching scheint hier prädestiniert dafür zu sein: Besuchende werden hier spielerisch über das Gelände gelotst und können nebenbei noch etwas über Heilbronn und die BuGa erfahren und kleine Gewinne wie z.B. Gutscheine erhalten. Das Ganze macht natürlich auch im Team, als Familie oder mit Freunden Spaß. Eine abwechslungsreiche und spannende Freizeitaktivität für Jedermann.

Dabei hat man die Wahl zwischen mehreren auf den Gelände verteilten Caches mit unterschiedlichem Umfang, die aus bis zu fünf Stationen bestehen, die durch das Lösen einfacher Rätsel oder kleinen Aufgaben den Weg zur nächsten Station offenbaren. An jeder Station gibt es einen QR-Code zu scannen, der dann über die Web-Anwendung Informationen und Belohnungen zugänglich macht. Wer angemeldet ist, kann sich über seine gefundenen Caches und die gesammelten Gesamtpunktzahl mit anderen Teams oder Cachernden messen und in der Rangliste einen Platz unter den besten 25 sichern.

Glossar

Administrator:

Eine Person, die Verwaltungsaufgaben in der Anwendung übernimmt (Beispiel: Verwaltung der Inhalte der Datenbank).

Akzeptanzkriterien:

Kriterien, gegen die eine fertig entwickelte User Story getestet wird. Akzeptanzkriterien stehen vor der Entwicklung der User Story fest und dienen als Anhaltspunkte für den Umfang der Funktionen dieser Story.

Benutzerrolle:

Eine Gruppe von Personen, unterteilt nach ihren Verantwortungen, Bedürfnissen etc.

Bootstrap:

Ein Framework zur Erstellung von responsive Websites.

BuGa:

Bundesgartenschau 2019 Heilbronn. Event, für das die Anwendung entwickelt wird.

Cache:

Eine Folge von bis zu 5 Stationen, die ein Nutzender der Reihe nach zu finden hat um den Cache zu beenden.

Cachernder:

Eine Person die die Anwendung zur Ausführung von Geocaches nutzt.

CSS:

Cascading Style Sheet. Eine Stylesheet Sprache für elektronische Dokumente.

Datenbank:

Ein Server, auf dem alle Daten über und um die Anwendung abgespeichert werden und jederzeit abrufbereit sind.

Framework:

Ein Software Rahmen, der bestimmte Funktionen zur Verfügung stellt, jedoch kein fertiges Programm ist, sondern ein Software Gerüst in den Entwickler ihren Code einbetten.

Gamification:

Die Anwendung spieltypischer Elemente auf einen spielfremden Kontext.

Geocaching:

Eine Art moderne Schnitzeljagd, bei der ein Cacher auf Basis von Hinweisen oder Koordinaten versucht einen Cache zu finden.

HTML:

Hypertext Markup Language. Textbasierte Auszeichnungssprache und Grundlage des World Wide Web.

Java Spring:

Backend Framework auf Basis von Java.

JPA (Java Persistence API):

Schnittstelle zur Übertragung von Datenbankeinträgen in Java Anwendungen.

QR-Code:

Ein Zweidimensionaler Code, der eine Zeichenkette verschlüsselt und von jedem modernen Smartphone gelesen werden kann

Rangliste:

Eine nach Zeit oder Punkten geordnete Tabelle, in die jeder Cacher seine Ergebnisse eintragen kann.

Scrum:

Framework zur agilen Softwareentwicklung.

Station:

Eine Station ist ein Teil eines Cache. Eine Station enthält einen Hinweis auf die nächste Station, die in dem Cache gefunden werden muss.

Tomcat:

Webserver Architektur auf Basis von Java Servlets.

User Story:

Eine Form zum Festhalten von Anforderungen an eine Anwendung. Geschrieben nach dem Schema: Als <Benutzerrolle> will ich <das Ziel> [, so dass <Grund für das Ziel>].

Vue.js:

Frontend Framework auf Basis von JavaScript zur Realisierung von responsive/progressive Websites

Wireframe:

Ein konzeptioneller Entwurf einer Website, der auf die nötigsten Elemente reduziert wird.

Referenzen auf andere Dokumente

- BUGA Heilbronn 2019 GmbH. (2018). *Bundesgartenschau Heilbronn*. Abgerufen 25. Oktober, 2018, von <http://www.buga2019.de>
- Schwaber, K., & Sutherland, J. (2017, November). *The Scrum Guide*. Abgerufen 25. Oktober, 2018, von <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>
- Wirdemann, R. (2017). *Scrum mit User Stories (3. Aufl.)*. München, Deutschland: Carl Hanser Verlag.
- Permantier, G. (2017). *Vorlesung Grundlagen des Software Engineering 1*.
- Permantier, G. (2018). *Vorlesung Grundlagen des Software Engineering 2*.

- Permantier, G. (2018). *Vorlesung Projektmanagement und Tools des Software Engineering*.

Offene Fragen und grundlegende Entscheidungen

Offene Fragen

- Wie finanziert sich GeoCaching? Werbung?

Grundlegende Entscheidungen

HTML 5 und CSS 3: HTML 5 und CSS 3 sind der gängige Standard für Websiteprogrammierung. Beide Sprachen sind leicht zu erlernen, sind jedoch auch mächtig genug alle Funktionen für unsere Anwendung anzubieten. Wissen über diese beiden Sprachen ist in der Gruppe ebenfalls schon vorhanden.

Webanwendung: Die Anwendung wird eine Webanwendung in Form von responsive Websites sein. Die Webanwendung wurde gewählt, damit der spätere Nutzer keine App herunterladen muss. Außerdem stehen in einer Webanwendung alle Funktionen, die wir benötigen zur Verfügung.

Java Spring: Aufgrund der Expertise unseres Teams ist Java die sinnvollste Technologie für unser Vorhaben. Als Framework wurde hierzu Java Spring Boots gewählt, da es eine leichtgewichtige Lösung für alle notwendigen Backendaufgaben zur Verfügung stellt.

JPA: JPA wurde als Technologie für die Schnittstelle zwischen Datenbank und Backend gewählt, da es eine leichte Integration in Java Spring bietet und ohne viel Vorwissen zu implementieren ist.

QR-Code: QR-Codes sind dank ihrer automatischen Fehlerkorrektur sehr robust und weit verbreitet. Jedes moderne Smartphone ist in der Lage ohne zusätzliche App (e.g. QR-Code Reader), allein durch die Kamerafunktion QR Codes zu entschlüsseln. Ebenfalls sind QR Codes leicht zu erstellen. Mithilfe dieses Codes wird die Identifikation von gefundenen Caches realisiert.

Tomcat: Tomcat soll als Webserver verwendet werden, da auf dem gemeinsamen Server schon andere Anwendungen mit Tomcat gehostet sind und dieser daher schon vorhanden ist. Weiter ist dies eine allgemein anerkannte Software für diesen Zweck.

Vue.js und Quasar: Vue.js ist ein Webframework, welches schlanker ist, als Angular und React, und einfach zu lernen ist, da es lediglich Kenntnisse über JavaScript und HTML erfordert. Quasar baut auf Vue.js auf und bietet viele Komponenten, die Nutzenden eine gute Nutzererfahrung bieten kann.

User Stories

Benutzerrollen: Cachender, Administrierender, Besuchender

Must-have: 8 User Stories

Should-have: 7 User Stories

Nice-to-have: 5 User Stories

User Story	Kategorie	User Story	Kategorie
Als Cachender will ich einen Cache von der Website aus starten können	Must-have	Als BuGa-Besuchender will ich mich registrieren	Must-have
Als Cachender/Besuchender will ich eine Übersicht über alle Caches sehen können, um mir einen Überblick zu verschaffen	Must-have	Als Cachender will ich den Fortschritt meines Caches verfolgen können	Should-have
Als Cachender/Administrierender will ich mich anmelden	Must-have	Als Cachender will ich Teams erstellen können	Nice-to-have
Als Cachender/Administrierender will ich mich abmelden	Must-have	Als Cachender will ich nachdem ich einen gefunden habe, eine Belohnung abholen können	Should-have
Als Besuchender der BuGa will ich Informationen über das GeoCaching Angebot erhalten	Should-have	Als Cachender will ich die Startpunkte eines Cache auf der Karte sehen können	Nice-to-have
Als Cachender/Besuchender will ich jederzeit die Rangliste einsehen können	Should-have	Als Cachender möchte ich andere zu meinem Team einladen können	Nice-to-have
Als Administrierender will ich einen neuen Cache erstellen können	Must-have	Als Cachender will ich einen QR-Code scannen können, sodass ich einen Cache starte	Should-have
Als Administrierender will ich Caches entfernen können	Must-have	Als Administrierender will ich Caches bearbeiten können	Should-have
Als Cachender will ich meine angefangenen und abgeschlossenen Caches sehen können	Should-have	Als Cachender möchte ich meinen Standort auf der Karte sehen können	Nice-to-have
Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, sodass ich dem Endpunkt des Cache näherkomme	Must-have	Als Cachender möchte ich Caches von der Karte aus aufrufen können	Nice-to-have
Als Administrierender will ich eine neue Station erstellen können	Must-have		

Benutzerrollen und eingesetzte Software

Benutzerrollen

BuGa-Besuchender: Besuchender der Bundesgartenschau 2019 in Heilbronn
Cachender: Ein auf der Webanwendung angemeldeter Nutzer, der einen Cache bearbeitet.
Administrierender: Verwaltender des Systems (Datenbanken etc.)

Schnittstellen

Datenbank:

- Von der Webanwendung zur Datenbank,

Webanwendung:

- vom Nutzenden zur Webanwendung
- vom Administrierenden zur Webanwendung

Eingesetzte Software

- IntelliJ: Wird zur Quellcodeentwicklung in Java verwendet
- DataGrip: Zur Datenbank Verwaltung eingesetzt
- Browser-Entwicklertools (Chrome/Firefox)
- Adobe XD: User Interface Design
- Jira: Projektmanagement
- WebStorm / Atom / SublimeText 3 für Frontendentwicklung

Constraints

- Das System muss bis zu 50 Benutzer gleichzeitig unterstützen.
- Das System muss eine Antwortzeit von weniger als einer Sekunde haben.
- Das System muss eine Verfügbarkeit von 99,9% haben.
- Das System muss einen Nutzer auf eine Genauigkeit von 3 Metern orten können.
- Der Webserver wird mit Tomcat betrieben.
- Die Anwendung wird in Java geschrieben.
- Die QR-Codes müssen mit einer Genauigkeit/Erfolgschance von 99,9% gelesen werden können.

Datenhaltung

Die Datenbank verwaltet zentral die Nutzer- und Administratordaten und speichert dort die erstellten Caches, dessen Stationen und die Informationen dazu. Außerdem werden zu den Caches jeweils Belohnung und weitere Informationen, wie Aufgaben zu bestimmten Caches hinterlegt.
Vereinfachte Übersicht über die Tabellen der Datenbank:

User

- Id
- Email
- Vorname
- Nachname
- Passwort (verschlüsselt)
- Summe der gesammelten Rankingpoints
- Team Id (Verweis auf ein Team)
- Session Token
- Rollen

Team

- Id
- Name
- Summe der Rankingpoints aller Teammitglieder

Station

- Id
- Rätsel

- Koordinaten (Breitengrad, Längengrad)
- Code

Cache

- Id
- Beschreibung
- Name
- Ranking Points, die man für das Erledigen dieser Caches erhält
- Belohnung
- Liste aller Stationen in geordneter Reihenfolge

Detaillierte Anforderungen

Cachender

8.1.1 Als Cachender will ich Teams erstellen können

8.1.1.1 Aufwand: 2

8.1.1.2 Details:

- Erstellen eines Teams aus verschiedenen Nutzenden
- Mit diesem Team lässt sich später gemeinsam cachen/Punkte sammeln
- Man darf maximal nur in einem Team sein
- Jedes Team bekommt einen einzigartigen Namen

8.1.1.3 Tasks:

- API-Aufruf für entsprechende Datenbankeinträge
- Seite zum Erstellen des neuen Teams (Teamname, Mitglieder)
- Neu erstelltes Team wird in Datenbank gespeichert
- API-Aufruf mit Check des Teamnamens in der Datenbank

8.1.1.4 Akzeptanzkriterien:

- Ein Cachender hat auf seinem Profil die Möglichkeit über einen Button ein Team zu erstellen
- Der Cachende kann den Namen seines Teams bei der Erstellung angeben
- Im Backend wird geprüft ob der Name des Teams bereits vergeben ist
- Es existiert ein API Aufruf, der einen Team Namen und einen User Token entgegennimmt und ein neues Team in der Datenbank anlegt und den Ersteller automatisch hinzufügt (/api/createTeam?name=&token=)
- Falls der Cachende bereits in einem Team ist, kann er kein neues Team erstellen. Er muss dafür zuerst aus seinem bisherigen Team austreten
- In Confluence existiert eine Dokumentation für den Aufruf

8.1.2 Als Cachender möchte ich meinen Standort auf der Karte sehen können

8.1.2.1 Aufwand: 8

8.1.2.2 Details:

- Anzeigen des eigenen Standorts auf der Karte des Bundesgartenschau
- Der Nutzende kann sich so besser orientieren und einen Cache in seiner Nähe auswählen
- Ist optional, also man muss seinen Standort nicht preisgeben

8.1.2.3 Tasks:

- Seite zum Freigeben des Standorts
- API-Aufruf des korrekten Standortes
- API-Aufruf der mit korrektem Standort und Karte

8.1.2.4 Akzeptanzkriterien:

- Nutzer hat Möglichkeit Standort freizugeben
- Auf der Karte wird der korrekte Standort des Nutzers angezeigt
- Die Realisierung der Karte und der Anzeige der Standorte ist Schnittstelle mit Lageplan Gruppe
- Es existiert ein API Aufruf, der eine Karte inklusive Standort zurückliefert (beispielsweise ein IFrame)
- In Confluence existiert eine Dokumentation des API Aufrufs

8.1.3 Als Cachender will ich einen QR-Code scannen können, damit ich einen Cache starte

8.1.3.1 Aufwand: 3

8.1.3.2 Details:

- Das Scannen eines QR-Codes startet den Cache
- Alternativ soll es auch möglich sein Caches von der Webseite aus zu starten

8.1.3.3 Tasks:

- Seite die es dem Chachenden ermöglicht die Kamere zu aktivieren
- API-Aufruf des QR was den die korrekte Station aus Datenbank liefert
- Seite die dem Cachenden nun ermöglicht den Cache zu starten.
- API-Aufruf der den Cache startet
- Start des Cache wird in Datenbank hinterlegt

8.1.3.4 Akzeptanzkriterien:

- Cachender kann Kamera für Web Anwendung freigeben
- Die Webanwendung kann QR Code scannen
- Es existiert für jeden Cache eine Startstation mit eine QR Code
- Falls eine Startstation gescannt wird(oder der Cache von der Webseite aus gestartet wird), wird ein API Aufruf getriggert, der den Nutzer den Cache starten lässt (api/startCache?token=&cacheID=)
- In der Datenbank wird gespeichert, dass der Cachende diesen Cache gestartet hat
- In Confluence existiert eine Dokumentation für den API Aufruf

8.1.4 Als Cachender (oder Besuchender) will ich jederzeit die Rangliste einsehen können

8.1.4.1 Aufwand: 2

8.1.4.2 Details:

- Rangliste soll jederzeit einsehbar sein
- Rangliste zeigt den Benutzernamen und die Punkte der besten 100 Cachenden nach Punkte geordnet in absteigender Reihenfolge

8.1.4.3 Tasks:

- Datenbankaufruf für Rangliste erstellen
- JSON aus Datenbankquelle generieren
- API-Aufruf für Rangliste erstellen
- API-Dokumentation erstellen
- View erstellen
- Datenbankabfrage vom Backend implementieren

8.1.4.4 Akzeptanzkriterien:

- Über einen API Aufruf wird ein JSON mit den ersten 100 Ranglisteneinträgen zurückgegeben.
- Der User kann sich eine Liste anzeigen lassen, die die Usernamen und punktestände der ersten 100 User anzeigt
- In Confluence existiert eine Dokumentation für den API Aufruf.

8.1.5 Als Cachender will ich meine angefangenen und abgeschlossenen Caches sehen können

8.1.5.1 Aufwand: 3

8.1.5.2 Details:

- Dem Cachenden werden alle Caches angezeigt, die er angefangen oder bereits beendet hat
- Des Weiteren bekommt er Informationen über den Fortschritt dieser Caches

8.1.5.3 Tasks:

- API-Aufruf für entsprechende Datenbankeinträge
- Seite die dem Cachenden seine Chaches anzeigt
- Zusammenstellen der Datenbankeinträge für einen User
- API-Dokumentation

8.1.5.4 Akzeptanzkriterien:

- In der Datenbank wird gespeichert, wenn ein Nutzer einen Cache anfängt und an welcher Station er gegebenenfalls aufhört

- Es existiert ein Backendaufruf, der einen Token entgegennimmt und ein JSON zurückgibt mit Informationen über die angefangenen und abgeschlossenen Caches eines speziellen Users
- Es existiert eine API Dokumentation über diesen Aufruf in Confluence
- Die angefangenen und abgeschlossenen Caches werden sind nur für den jeweiligen Nutzer abrufbar
- Die angefangenen und abgeschlossenen Caches werden in einer Liste auf dem Profil des Cachenden dargestellt

8.1.6 Als Cachender will ich, nachdem ich einen Cache gefunden habe, eine Belohnung abholen

8.1.6.1 Aufwand: 1

8.1.6.2 Details:

- In Form von Gutscheinen beispielsweise
- Kann mit Hilfe von Sponsoren realisiert werden
- Ist optional, nicht jeder Cache endet mit einer Belohnung

8.1.6.3 Tasks:

- API-Aufruf der letzten Station des Caches
- Seite die dem Cachenden zeigt er hat den Cache erfolgreich abgeschlossen
- Cache wird als abgeschlossen in Datenbank eingetragen
- Seite die dem Cachenden zeigt er habe eine Belohnung erhalten
- JSON aus Belohnung generieren
- API-Aufruf die Belohnung übergibt
- API-Dokumentation

8.1.6.4 Akzeptanzkriterien:

- Für bestimmte Caches existiert eine Belohnung, die in der Datenbank gespeichert ist
- Falls ein Cachender den letzten QR Code einer solchen Station scannt, wird in der Datenbank gespeichert, dass der Cachende diesen Cache abgeschlossen hat und ein JSON mit der Belohnung zurückgeliefert
- Es existiert eine Dokumentation für diesen API Aufruf in Confluence

8.1.7 Als Cachender/Besuchender will ich eine Übersicht über alle Caches sehen können, um mir einen Überblick zu verschaffen

8.1.7.1 Aufwand: 2

8.1.7.2 Details:

- Es werden alle Caches angezeigt mit all ihren Informationen, wie Beschreibung, Name und Position
- Wird sowohl auf der Karte als auch in einer Liste angezeigt

8.1.7.3 Tasks:

- Seite die alle verfügbaren Caches darstellt
- API-Aufruf der Caches im JSON Format
- Seite die Details der Caches darstellt und auch einen Start Button hat
- API-Dokumentation

8.1.7.4 Akzeptanzkriterien:

- Es existiert ein API Aufruf, der alle verfügbaren Caches aus der Datenbank lädt und als JSON zurückliefert (api/allCaches)
- Alle Caches werden in einer Liste dargestellt und haben eine "Start" Knopf
- Jeder Cache in der Liste ist ein ausklappbares Element, dass, wenn ausgeklappt, Informationen über den Cache darstellt
- Es existiert eine API Dokumentation über diesen Aufruf in Confluence

8.1.8 Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, so dass ich dem Endpunkt des Cache näherkomme

8.1.8.1 Aufwand: 13

8.1.8.2 Details:

- Ein Cache besteht aus mehreren Stationen
- Stationen beinhalten Rätsel, mit dessen Hilfe man die nächste Station finden kann
- Wenn man an der Endstation angekommen ist, also alle Rätsel in der richtigen Reihenfolge gelöst hat, hat man den Cache erfolgreich beendet
- Jede Station hat einen zugehörigen QR-Code, den man einscannen muss, um das dazugehörige Rätsel zu erhalten

8.1.8.3 Tasks:

- Seite die es dem Cachenden ermöglicht die Kamera zu aktivieren

- API-Aufruf der Station in der Datenbank
- Seite die einem zeigt das die Station die richtige ist
- API-Aufruf Korrektheit der Station und Übergabe nächster Station
- Seite die dem Cachenden die nächste Station anzeigt
- API-Dokumentation

8.1.8.4 Akzeptanzkriterien:

- Cachender kann Kamera für Web Anwendung freigeben
- Die Webanwendung kann QR Code scannen
- Durch das Scannen eines QR Codes wird ein API Aufruf getriggert, der überprüft ob die gefundene Station die richtige ist und gegebenenfalls die nächste Station zurückliefert
- In Confluence existiert eine Dokumentation der API Aufrufe
- Wenn alle Stationen eines Caches gescannt wurde wird ein API Aufruf getriggert, der abspeichert, dass der Cache von diesem Cacher bearbeitet wurde und gegebenenfalls die Belohnung für das Abschließen zurückliefert
- Die Anwendung kennt die korrekte Reihenfolge aller Stationen für jeden Cache
- Die Anwendung weiß an welche Station sich welcher Cacher zurzeit befindet

8.1.9 Als Cachender/Administrierender will ich mich anmelden

8.1.9.1 Aufwand: 3

8.1.9.2 Details:

- Anmeldefunktion für die Geocaching Webanwendung
- Einzigartiger username + passwort nötig für die Anmeldung

8.1.9.3 Tasks:

- Seite die den Login mit username und passwort ermöglicht
- API-Aufruf vergleicht die Login Daten
- API-Aufruf JWT Token wird zurückgeliefert Login erfolgreich
- JWT Token wird im Browser gespeichert
- Seite die den Home Bildschirm des Cachenden oder Admins liefert
- API-Dokumentation

8.1.9.4 Akzeptanzkriterien:

- Es existiert ein Formular auf der Website, in den Username und Passwort eingetragen werden können
- Falls der "Login" Button betätigt wird, werden Username und Passwort an das Backend gesendet über einen API Aufruf (api/login)
- Das Passwort wird mit dem gehashten Passwort in der Datenbank abgeglichen
- Falls die Passwort/Username Kombination korrekt ist, wird im Backend ein JSON Web Token (JWT) erstellt und zurück ans Frontend geliefert
- Im Browser des Cachenden wird der JWT Token gespeichert
- Im Backend existiert die Möglichkeit eine JWT Token zu verifizieren
- Es existiert eine API Dokumentation über diesen Aufruf in Confluence

8.1.10 Als Cachender/Administrierender will ich mich abmelden

8.1.10.1 Aufwand: 1

8.1.10.2 Details:

- Abmeldefunktion für Nutzende der GeoCaching Webanwendung

8.1.10.3 Tasks:

- Seite die einen Button Logout enthält
- JWT Token wird aus Browser des Cachendem gelöscht
- Seite die den Logout bestätigt
- Cachender wird zur Startseite zurückgeführt

8.1.10.4 Akzeptanzkriterien:

- Auf dem Profil des Nutzers ist ein Button "Logout"
- Falls der "Logout" Button betätigt wird, wird der Token aus dem Browser des Cachenden gelöscht

8.1.11 Als Cachender möchte ich andere zu meinem Team einladen können

8.1.11.1 Aufwand: 8

8.1.11.2 Details:

- Teammitglieder können anderen Nutzenden Einladungen in ihr Team schicken
- Andere Nutzende können diese Einladungen annehmen und treten damit dem Team bei
- Wie zuvor erwähnt, darf man nur Mitglied eines Teams sein

8.1.11.3 Tasks:

- Seite die Cachendem sein Profil anzeigt mit Teamzugehörigkeit
- Seite die es ermöglicht dem Team Mitglieder hinzuzufügen
- Seite die das Suchen von Cachenden per username ermöglicht und hinzufügen lässt
- API-Aufruf der dem angefragten User die TeamID in der Datenbank hinzufügt
- Seite die dem angefragten User die Teamanfrage anzeigt die er annehmen oder ablehnen kann
- API-Aufruf der dem User bei Annahme die TeamID und Name endgültig einträgt
- API-Dokumentation

8.1.11.4 Akzeptanzkriterien:

- Cachender sieht in seinem Profil seine Teamzugehörigkeit und hat, wenn er einem Team angehört einen Button "Teammitglied hinzufügen"
- Drückt der Cachender den Button "Teammitglied hinzufügen" öffnet sich ein Suchfenster in den der Cachende einen Usernamen eingeben kann. Drückt der Cachende auf "absenden" wird ein API Aufruf getriggert, der den angefragten Username und die TeamID entgegennimmt und in der Datenbank abspeichert
- Loggt sich der angefragte User wieder ein, erhält er ein kleines Popup Fenster, bei dem er die Teamanfrage annehmen oder ablehnen kann
- Es existiert eine API Dokumentation über diesen Aufruf in Confluence

Administrierender

8.2.1 Als Administrierender will ich neue Caches erstellen können

8.2.1.1 Aufwand: 5

8.2.1.2 Details:

- Der Cache braucht einen Namen, eine Beschreibung und die Anzahl an Punkten, die der Cachende bekommt, wenn dieser ihn abschließt
- Außerdem besteht der Cache aus vielen Stationen
- Nur der Admin hat Zugriff auf diese Funktion

8.2.1.3 Tasks:

- Seite zum Erstellen von Caches
- API-Aufruf mit den Stationen der Datenbank in JSON
- Neu erstellen Cache in der Datenbank speichern
- API-Admincheck
- API-Aufruf mit den Daten des Caches um sie der Datenbank hinzuzufügen
- Seite mit neuem Cache anzeigen lassen

8.2.1.4 Akzeptanzkriterien:

- Das Programm kann zwischen Nutzern mit verschiedenen Rollen unterschieden (Admin, User)
- Im Frontend existiert ein Formular über das der Admin einen neuen Cache erstellen kann. (Name des Caches, Beschreibung, Ranglistenpunkte, eventuelle Belohnung)
- Bei abschicken des Formulars werden die Daten in korrekter Form in der Datenbank gespeichert
- Es existiert eine API Dokumentation über den API Aufruf in Confluence
- Falls der Nutzer keine Admin Rolle hat, wird seine Anfrage abgelehnt
- Zu einem Cache können bis zu 5 Stationen hinzugefügt werden
- Wenn eine neue Station hinzugefügt wird, öffnet sich ein Formular, bei dem eine neue Station hinzugefügt werden kann (mehr dazu in der User Story "Als Administrator will ich eine neue Station erstellen können")
- Die Stationen werden in korrekter Reihenfolge in der Datenbank abgespeichert

8.2.2 Als Administrierenden will ich Caches bearbeiten können

8.2.2.1 Aufwand: 8

8.2.2.2 Details:

- Es lassen sich alle Eigenschaften des Caches verändern, also z.B. Name, Beschreibung, Punkte, Stationen.
- Nur der Admin hat Zugriff auf diese Funktion

8.2.2.3 Tasks:

- Seite bei der alle Caches angezeigt werden mit der Option zur Bearbeitung
- API-Admin Check
- Seite zum Erstellen von Caches wird geladen und zeigt vorhandene Daten an
- API-Aufruf der Cachedaten zum Bearbeiten des Caches
- Seite lässt Admin eingetragene Daten bearbeiten und abspeichern.
- API-Aufruf Daten werden aktualisiert der Datenbank übergeben.
- API-Dokumentation

8.2.2.4 Akzeptanzkriterien:

- Ein Administrator kann einen Cache anklicken und erhält daraufhin über einen Button die Möglichkeit ihn zu bearbeiten
- Wenn der Administrator einen Cache bearbeiten will, öffnet sich dasselbe Formular wie beim Cache erstellen
- Alle vorhandenen Daten über den Cache sind bereits in dem Formular eingetragen
- Der Administrator kann alle Felder des Formulars bearbeiten und neue Daten eintragen
- Der Administrator kann über einen Klick auf den Button "Speichern", den bearbeiteten Cache in der Datenbank abspeichern
- Es existiert ein API Aufruf, der eine CacheID und die neuen bearbeiteten Daten entgegennimmt und den alten Cache in der Datenbank updatet
- Es existiert eine API Dokumentation über den API Aufruf in Confluence
- Das Programm kann zwischen Nutzern mit verschiedenen Rollen unterschieden (Admin, User)
- Falls der Nutzer keine Admin Rolle hat, wird seine Anfrage abgelehnt.

8.2.3 Als Administrierender will ich Caches entfernen können

8.2.3.1 Aufwand: 2

8.2.3.2 Details:

- Der Cache wird vollständig gelöscht, also aus allen Listen, Karten und der Datenbank.
- Punkte, die Benutzer gesammelt haben, indem sie diesen Cache gelöst haben werden ebenfalls gelöscht
- Der Cache taucht auch nichtmehr unter den erledigten Caches der Benutzer auf, die diesen Cache bereits gelöst haben
- Nur der Admin hat Zugriff auf diese Funktion

8.2.3.3 Tasks:

- Seite bei der alle Caches angezeigt werden mit der Option zur Löschung
- API-Admin Check
- Seite die den Admin nach Löschung fragt
- API-Aufruf mit CacheID der den Cache aus Datenbank löscht
- API-Dokumentation

8.2.3.4 Akzeptanzkriterien:

- Das Programm kann zwischen Nutzern mit verschiedenen Rollen unterschieden (Admin, User)
- Wenn ein Administrator einen Cache anklickt erhält er einen Button "Cache löschen"
- Falls der Nutzer keine Admin Rolle hat, wird seine Anfrage abgelehnt
- Wenn ein Administrator den Button "Cache löschen" anklickt, wird ein API Aufruf getriggert, der eine CacheID entgegennimmt und diese aus der Datenbank entfernt
- Außerdem wird der Cache auch aus allen anderen Datenbanktabellen gelöscht in denen er auftaucht
- Es existiert eine API Dokumentation über den API Aufruf in Confluence

8.2.4 Als Administrierender will ich eine neue Station erstellen können

8.2.4.1 Aufwand: 3

8.2.4.2 Details:

- Die Station braucht eine Beschreibung(Rätsel), Koordinaten, einen Code und eine Lösung zu dem Rätsel
- Eine Station gehört zu einem Cache und lässt sich während der Cacheerstellung erstellen
- Nur der Admin hat Zugriff auf diese Funktion

8.2.4.3 Tasks:

- Seite die alle Stationen darstellt bearbeiten lässt und neue hinzufügen lässt
- API-Admin Check
- API-Aufruf der Daten
- Seite zum Erstellen von Stationen
- API-Aufruf der Daten in Datenbank einträgt
- Seite die neue Station anzeigt

8.2.4.4 Akzeptanzkriterien:

- Das Programm kann zwischen Nutzern mit verschiedenen Rollen unterschieden (Admin, User)
- Falls der Nutzer keine Admin Rolle hat, wird seine Anfrage abgelehnt
- Der Administrator kann im Formular für "Cache erstellen" einen Button drücken mit "Station erstellen"
- Es existiert ein API Aufruf, der alle Informationen für eine Station entgegennimmt und diese in der Datenbank abspeichert
- Es existiert eine API Dokumentation über den API Aufruf in Confluence
- Im Frontend gibt es ein Formular, in das alle Informationen über eine Station eingetragen werden können
- Die Stationen, die zu einem Cache gehören werden in der richtigen Reihenfolge gespeichert

Besuchender

8.3.1 Als Besuchender der BuGa will ich Informationen über das GeoCaching-Angebot erhalten

8.3.1.1 Aufwand: 2

8.3.1.2 Details:

- Auf Angebot aufmerksam machen und Erklärung zur Benutzung bieten
- Informationstext bereitstellen, der eine kleine Einführung in das GeoCaching bietet

8.3.1.3 Tasks:

- Seite die dem Besuchenden das GeoCaching näher bringt und erläutert

8.3.1.4 Akzeptanzkriterien:

- Die Informationsseite ist leicht zu finden und macht einen Besuchenden auf das GeoCaching-Angebot aufmerksam und erklärt ihm, wie dieses funktioniert
- Der angezeigte Text ist leicht verständlich

8.3.2 Als Besuchender/Administrierender der BuGa will ich mich registrieren

8.3.2.1 Aufwand: 5

8.3.2.1 Details:

- Folgende Angaben sind erforderlich: einzigartiger username, passwort, email, firstname, lastname

8.3.2.2 Tasks:

- Seite die einen Button registrieren enthält
- API-Aufruf der Registrieren Funktion
- Seite die die benötigten Daten anzeigt und dem User ermöglicht diese Daten zu füllen.
- API-Aufruf Validierung der Daten und anschließende Übergabe an Datenbank
- Seite die anzeigt das die Registrierung erfolgreich war

8.3.2.3 Akzeptanzkriterien:

- Es existiert ein Formular, in dass der Nutzer seinen Username, Email, first und lastname und Passwort eintragen kann
- Drückt der Nutzer auf "Registrieren" wird ein API Aufruf getriggert, der alle Nutzerdaten entgegennimmt
- Es existiert eine API Dokumentation über den API Aufruf in Confluence
- Der API Aufruf überprüft, ob die Daten valide sind
- Alle Daten werden in der Datenbank gespeichert und das Passwort wird gehasht

Installation

Installationsanleitung:

Das gesamte Projekt ist jederzeit in BitBucket als Git Projekt aufzufinden. Durch den 'git clone' Befehl ist es möglich das Projekt auf dem Computer zu kopieren. Zum Starten des Projekts muss entweder das Projekt in einer IDE oder das Verzeichnis in dem das Projekt liegt in einem Command-line interface geöffnet werden. Mit dem Befehl `'gradle build'` oder `'./gradlew build'` kann das Projekt gebaut und als .jar oder .war Datei zusammengefasst werden. Das .war File dient dazu, dass die Anwendung auf einen Server deployed werden kann. Das .jar File beinhaltet zusätzlich noch einen lokalen Tomcat Server, der automatisch beim Starten der Anwendung geöffnet wird. Dies dient dazu die Anwendung auch lokal laufen lassen zu können. Die Webseiten sind dann unter der Adresse 'http://localhost:8080/' erreichbar.

User Stories, die erfüllt werden müssen, damit die obige Installation ohne Probleme funktioniert:

9.1 Als Administrierender möchte ich die Software auf dem Server installieren, um sie der Öffentlichkeit zugänglich zu machen.

9.1.1 Details:

- Anwendung soll auf einen Server korrekt deployed werden. Hierfür muss ein WAR File korrekt kompiliert werden können.

9.1.2 Tasks:

- WAR Datei oder Java-Programmpaket in Tomcat bereitstellen
- Konfiguration an Umgebung anpassen
- WAR Datei auf Server deployen

9.1.3 Akzeptanzkriterien:

- Softwarepaket muss funktionierendes Front- und Backend enthalten
- Das Softwarepaket ist ohne Fehler als WAR Datei exportierbar
- Das Softwarepaket kann auf einen Server deployed werden
- Die Adresse des Servers ist für jeden erreichbar

9.2 Als Besucher möchte ich die Anwendung benutzen können ohne eine zusätzliche Software installieren zu müssen.

9.2.1 Details:

- Anwendung soll von einem Browser aus aufgerufen werden können. Es muss keine App oder ähnliches heruntergeladen werden

9.2.2 Tasks:

- Besucher muss die Webseite aufrufen

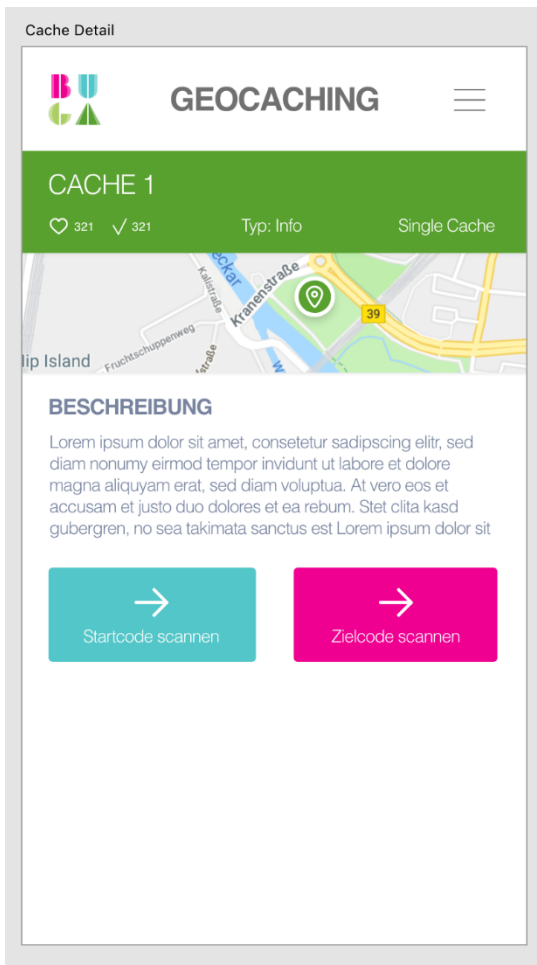
9.2.3 Akzeptanzkriterien:

- Softwarepaket muss funktionierendes Front- und Backend enthalten
- Softwarepaket muss über einen Webbrowser zugänglich sein
- Serveradresse muss für jeden zugänglich sein

GUI-Entwürfe, Bedienoberfläche

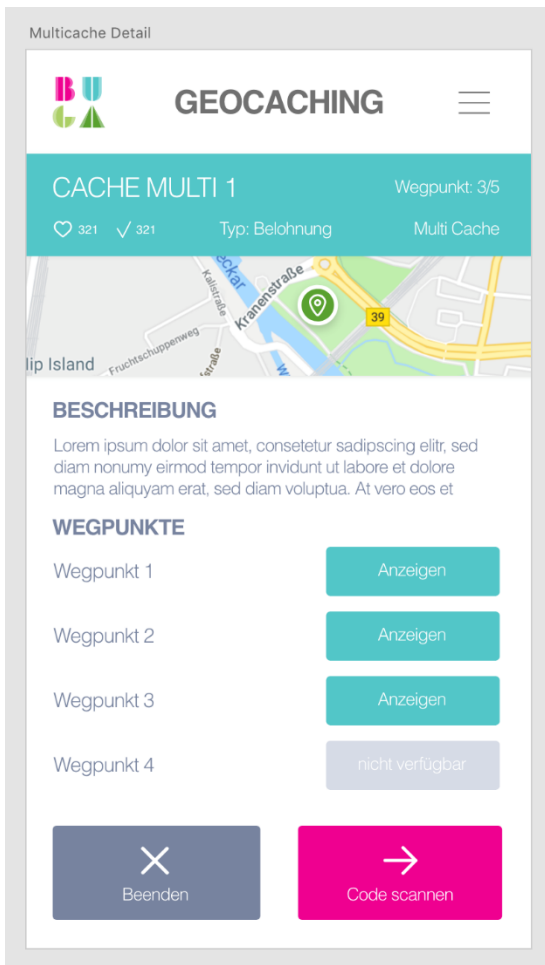
Die Übersichtsseite zeigt alle verfügbaren Caches auf einer schwenk- und zoombaren Karte an. Wenn ein Cache angetippt wird, erscheint ein kleines Übersichtsmodul mit den wichtigsten Informationen zum Cache und einem Link zur Detailsansicht.





Dies ist die Detailansicht eines Caches. Hier können alle Informationen eingesehen werden, auch ob der Cache schon gefunden wurde. Wenn das der Fall ist, kann auf dieser Seite auch auf die freigeschalteten Informationen und die Belohnung zugegriffen werden. Zudem wird auf dieser Seite der Cache gestartet und beendet. (Zugriff auf QR-Scanner)

Eine weitere Detailansicht eines Cache. Hier ein bereits angefangener Cache, bei dem schon Stationen (Wegpunkte) gefunden wurden. Auf die spezifischen Informationen zu den Stationen kann hier jederzeit zugegriffen werden, wenn diese gefunden und gescannt wurden.



In der Profilansicht kann sich der Cacher seinen Fortschritt anzeigen lassen und die Teamzugehörigkeit verwalten. Außerdem hat er Zugriff zu seinen



USER XY

Caches: 1 Rang: 11

Team: BuGa Cacher

Austreten

Team ansehen

GEFUNDENE CACHES



CACHE 1

♡ 321 ✓ 321



Details

ANGEFANGENE CACHES



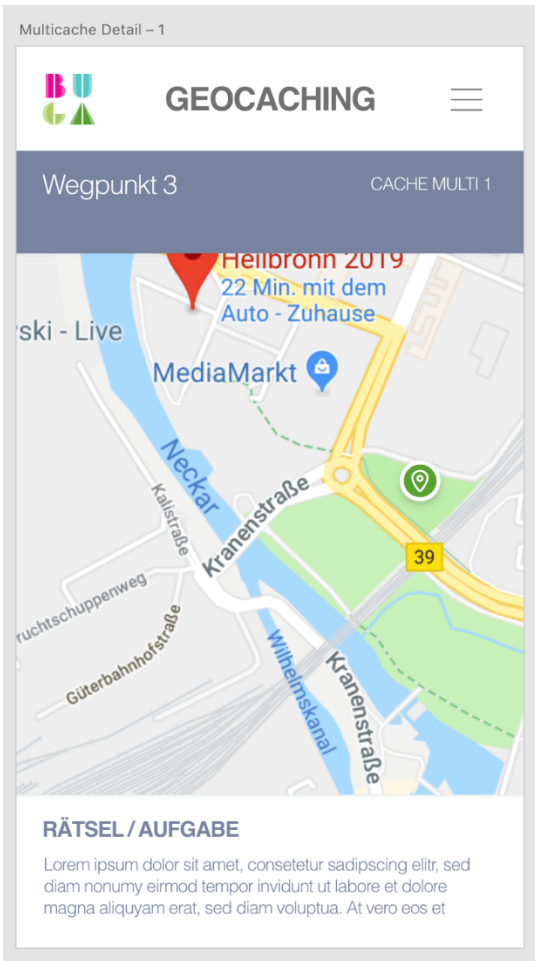
CACHE 1

♡ 321 ✓ 321



Details

angefangenen oder gefundenen Caches.



Ansicht der spezifischen Informationen zu den Stationen (Wegpunkten). Hier werden Aufgaben, Rätsel oder Informationen über die nächste Station aufgeführt.

Geschätzter Aufwand in Story Points

Legende:


1 = hohe Priorität 3 = niedrige Priorität

User Story	Kosten in StoryPoints	PRIO
Als Cachender will ich Teams erstellen können	2	3
Als Cachender möchte ich andere zu meinem Team einladen können	8	3
Als Cachender will ich meine angefangen und beendeten Caches sehen können	5	2
Als Cachender will ich einen QR-Code scannen können, damit ich einen Cache starte	3	2
Als Cachender möchte ich einen Cache von der Website aus starten können	3	1
Als Cachender (oder Besuchender) will ich jederzeit die Rangliste einsehen können	2	2
Als Cachender will ich den Fortschritt meines Cache verfolgen können	3	2
Als Cachender will ich, nachdem ich einen Cache gefunden habe, eine Belohnung abholen	1	2
Als Cachender/Besuchender will ich eine Übersicht über alle Caches sehen können, um mir einen Überblick zu verschaffen	2	1
Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, so dass ich dem Endpunkt des Cache näherkomme	13	1
Als Administrierender will ich neue Caches erstellen können	5	1
Als Administrierender will ich Caches bearbeiten können	8	2

Als Administrierender will ich Caches entfernen können	2	1
Als Besuchender der BuGa will ich Informationen über das GeoCaching-Angebot erhalten	2	2
Als BuGa-Besuchender will ich mich registrieren	5	1
Als Cachender/ Administrierender will ich mich anmelden	3	1
Als Cachender/ Administrierender will ich mich abmelden	1	1
Als Cachender möchte ich Startpunkte eines Cache auf der Karte sehen können	8	3
Als Cachender möchte ich meinen Standort auf der Karte sehen können	8	3
Als Cachender möchte ich die Caches von der Karte aus aufrufen können	8	3
Als Administrierender will ich neue Stationen erstellen können	3	1

Projektauftrag

Im Ordner "Zusätzliche Dokumentationsdateien" (USB-Stick) liegt der Projektauftrag auch noch als Word oder PDF Datei vor.

Datei	Geändert 
PDF-Datei Projektauftrag_GeoCaching-1.pdf	Mär 28, 2019 by Katharina Will
PDF-Datei Projektauftrag_GeoCaching_V1.3.pdf	Apr 11, 2019 by Katharina Will
PDF-Datei Projektauftrag_Geocaching_V1.3_signed.pdf	Mai 14, 2019 by Timo Volkmann

Ziehen Sie Dateien an diese Stelle, um sie hochzuladen, oder [Dateien suchen](#)

 [Alle herunterladen](#)

Teammitglieder:	Robin Christ, Michael Dietz, Maximilian Leopold, René Schießwohl, Timo Volkmann, Katharina Will
Autoren:	Robin Christ, Michael Dietz, Maximilian Leopold, René Schießwohl, Timo Volkmann, Katharina Will

Inhaltsverzeichnis

- [1 Zusammenfassung/Executive Summary](#)
- [2 Ausgangslage und Problemsituation](#)
- [3 Vision/Projektziel](#)
- [4 Projektkinhalt/Projektumfang](#)
- [5 Planung](#)
- [6 Information der Auftraggeber](#)
- [7 Projektorganisation](#)
- [8 Genehmigung](#)

Änderungen

Version	Datum/ Stand	Grund der Änderung, Hinweise	Bearbeitung durch
V0.1	21.03.2019	Kapitel 4	Timo Volkmann
V0.2	22.03.2019	Kapitel 1	Maximilian Leopold
V0.3	22.03.2019	Kapitel 2	Michael Dietz
V0.4	23.03.2019	Kapitel 3	Katharina Will
V0.5	23.03.2019	Kapitel 5	Robin Christ
V0.6	26.03.2019	Kapitel 6	René Schießwohl
V0.7	26.03.2019	Kapitel 1, 3, 5	Timo Volkmann
V0.8	27.03.2019	Kapitel 2, 6	Maximilian Leopold
V1.0	27.03.2019	Kapitel 1, 2, 3	Katharina Will
V1.1	05.04.2019	Kapitel 4	Timo Volkmann
V1.2	09.04.2019	Kapitel 3, 6	Maximilian Leopold

V1.3	09.04.2019	Kapitel 3, 4, 5	Katharina Will
------	------------	-----------------	----------------

Zusammenfassung/Executive Summary

Unser Ziel ist es, die Bundesgartenschau 2019 in Heilbronn für junge, technikaffine Menschen attraktiver zu machen, sowie auch moderne Technik den anderen Generationen näherzubringen. Wir nutzen hierzu ein Zusammenspiel von neuen Technologien und Ideen, und Altbekanntem. Im Kern unserer Anwendung steht dabei die spielerische Führung über die Bundesgartenschau, als eine Art moderne Schnitzeljagd. Der Cachende kann hierzu einen von vielen vorgefertigten Caches (Routen) über die BuGa auswählen. Jedes dieser Caches unterteilt sich in eine Reihe von Stationen, bei der man einen Hinweis bekommt oder ein kleines Rätsel lösen muss, um die nächste Station zu finden. An einer Station, sobald gefunden, kann der Nutzer über die Webanwendung einen QR-Code einscannen und erhält so die nächste Aufgabe gestellt, um die nächste Station zu finden. Die genaue Position muss der Cachende anhand dieser Hinweise, geschicktem Kombinieren und Wissen über die Bundesgartenschau, Heilbronn und Botanik herausfinden. Findet ein Cachender alle Stationen eines Cache erhält er eine kleine Belohnung und sammelt Punkte für eine Rangliste. Je mehr Routen ein Nutzer abschließt, desto mehr Punkte erhält er und desto höher wird er in der Rangliste angezeigt. Dieses Feature verleiht dem ganzen noch eine gewisse Kompetitivität. Our Goal is to make the Bundesgartenschau in Heilbronn more attractive for young, tech-savvy people and also to bring this technology to older generations. To achieve this, we use a mix of new technologies and ideas and well-known. The core of our application consist of many premade guided tours on the Bundesgartenschau, like a modern kind of scavenger hunt. Each of these caches is divided into several stations the user has to visit one after another. If the user finds the next station for his cache he can scan the QR-Code provided by station to gets hints about the location of the next station. The exact position of the next station is obtained by the user, by combining the hints and his knowledge about the Bundesgartenschau, botany and Heilbronn. By finding all stations in the correct order the user completes a cache and gets a small reward and points in the global leaderboard. By completing more caches the user gains more and more points for the global leaderboard. This feature gives the application a certain degree of competitiveness.

Ausgangslage und Problemsituation

Die Bundesgartenschau 2019 startet am 17. April und hat für 173 Tage für Besucher unterschiedlichster Herkunft die Tore geöffnet. Diese umfasst ein großes Gebiet und bietet reichlich Attraktionen. Man kann sich Blumen anschauen, Konzerte besuchen, etc. Jedoch gibt es nur eine geringe Menge an Angeboten, um persönlich aktiv zu werden und sich innerhalb des großen Geländes der Bundesgartenschau spielerisch zu unterhalten.

Um sich durch das umfangreiche Gelände der Bundesgartenschau zu bewegen, kann schlichtweg die Motivation fehlen. Besonders mit einem spielerischen Ansatz sollte es möglich sein diese Motivation zu entfachen, um sich über das große Gelände zu bewegen.

In einer Zeit in der Smartphone-Sucht bei Jugendlichen und jungen Erwachsenen kein Thema mehr ist, das belächelt werden kann, sollte gerade auf der Bundesgartenschau, in der es vorwiegend um die Schönheit der Natur geht, aufmerksam gemacht werden. Anhand verschiedener Ansätze kann man auf verschiedene Attraktionen, wie Blumen, Denkmäler und Gewässer hinweisen.

Ein weiteres Problem der Bundesgartenschau ist, dass viele Attraktionen zeitlich begrenzt sind. Was soll man tun, wenn es derzeit keine Aufführungen gibt, bereits alle Attraktionen, wie Denkmäler und Statuen, gesehen wurden? Es wäre äußerst hilfreich ein Angebot bieten zu können, welches zeitlich unbegrenzt, also jederzeit durchführbar ist. Dadurch würde man Besuchern einen Grund geben, die Bundesgartenschau mehrmals zu besuchen, auch zu Zeiten, zu denen es gerade keine großen Attraktionen, wie Konzerte gibt. Viele Events finden vor allem am Anfang der Bundesgartenschau statt, also braucht man ein langfristig motivierendes Angebot, um die Bundesgartenschau auch ebenso langfristig attraktiv zu machen.

Dazu sollte das Angebot eine große Nutzergruppe abdecken, egal ob jung oder alt, Mann oder Frau, Einzelperson oder Gruppe. Das Angebot sollte Interesse, unabhängig all dieser Faktoren, bieten.

Im letzten Semester starteten wir mit einem Team aus fünf Personen. Nun kam ein neues Teammitglied hinzu. Mit diesem Team aus sechs Personen möchten wir nun die oben genannten Probleme angehen und mit Hilfe einer Webanwendung, welche vor allem auf mobilen Geräten nutzbar ist, lösen.

Vision/Projektziel

Ziel des Projekts ist eine lauffähige Webanwendung, die auf eine spaßige, spielerische und abwechslungsreiche Weise Besuchenden die Bundesgartenschau 2019 in Heilbronn näherbringt. Die Anwendung soll den Erkundungstrieb der Besuchenden fördern und Wissen über die Bundesgartenschau vermitteln.

Eine langfristige Motivation wird gewährleistet, indem GeoCaching eine Art Schnitzeljagd darstellt, die nicht zeitlich gebunden ist, d.h. die zu suchenden Hinweise sind nicht mit bestimmten Terminen auf der Bundesgartenschau verbunden, sondern mit Attraktionen, die jederzeit verfügbar sind.

Die Besuchenden der Bundesgartenschau können sich über die Webanwendung die vorhandenen Caches anschauen und einen davon starten. Caches sind versteckte Stationen, an denen QR-Codes zu finden sind. Diese müssen gescannt werden, um den gesamten Cache abzuschließen oder Hinweise auf die nächste Station zu erhalten.

Der gestartete Cache stellt eine Aufgabe oder ein Rätsel, das es für den Cachenden zu lösen gilt, um die nächste Station oder das Ziel des Cache erreichen zu können. Eine Station bzw. das Ziel eines Cache ist in Form eines QR-Codes zu suchen, der auf der Bundesgartenschau versteckt liegt. Die erwähnten Rätsel und Aufgaben sollen es den Besuchenden ermöglichen diese versteckten QR-Codes zu finden.

Besuchende müssen kein Konto haben, um am GeoCaching teilnehmen zu können. Wenn ein Besuchender sich allerdings als Cachender registriert, erhält er einige Vorteile, die im Folgenden beschrieben werden.

Abgeschlossene Caches bieten Punkte für angemeldete Cachende, um in der Rangliste höher zu steigen. Umso mehr Caches ein Cachender oder ein Cacher-Team abgeschlossen hat, desto höher steigt es in der Rangliste, die jederzeit für Nutzende einsehbar ist. Angefangene und abgeschlossene Caches sind im Profil für angemeldete Nutzende einsehbar.

Multi-Caches sind die einzige Art von Caches, die von der ursprünglichen Idee von GeoCaching übernommen wird. Mit Multi-Caches (nachfolgend Caches genannt) werden allerdings auch andere Arten von Caches abgedeckt, wie z.B. Single-Caches und Mystery-Caches. Jeder Cache hat mindestens eine Start- und eine Zielstation, sie unterscheiden sich jedoch in der Anzahl an zu findenden Stationen, und somit die Anzahl zu lösender Rätsel, bis der Cache abgeschlossen ist. Durch den höheren Zeitaufwand ist die Punktbewertung von längeren Caches höher, im Vergleich zu kürzeren Caches. Die Schwierigkeit der zu lösenden Rätsel entscheidet ebenfalls über die Anzahl der Punkte, die man als Belohnung erhält. Der Fortschritt eines Cache soll vom Cachenden verfolgbar sein.

Angemeldete Cachende sollen auch Teams bilden können, in denen sie gemeinsam Stationen eines Cache suchen, da das gemeinsame Erkunden und Knobeln mehr Vergnügen bereitet.

Außerdem soll es über die Webanwendung möglich sein, die Position von QR-Codes, mit deren Scannung man einen Cache starten kann, auf einer Karte sehen. Die eigene Position soll man ebenfalls auf dieser Karte erkennen können, um sich auf dem großen Gelände der Bundesgartenschau nicht zu verlieren.

Administrierende sind für die Verwaltung von Caches zuständig.

Projekthalt/Projektumfang

Wir stellen mit unserer Geocaching-Webapplikation eine Plattform bereit, die Besuchern das Entdecken der BuGa auf eine spielerische Art und Weise ermöglicht.

Um Anreize zu schaffen, können abgeschlossene Caches mit Belohnungen versehen werden und Cachende bzw. Teams aus bis zu 10 einzelnen Cachenden über Ranglisten sich miteinander messen. Dazu muss der Nutzende sich authentifizieren und seinen Fortschritt wieder aufrufen können. Das wird mit der Implementierung einer Userverwaltung erreicht, bei der sich Nutzende selbst registrieren können und Administrierende Nutzende, Rechte, Caches und sonstige Inhalte (Start-/Informationsseiten) der Webapplikation verwalten können.

Aus einer Liste oder über eine OSM-Karte auf der die Startpunkte markiert sind, kann sich der Cachende einen Cache auswählen, den er gerne spielen möchte. Er wählt sich also einen Cache aus und bekommt am Startpunkt (Markierung auf der Karte) nach dem Einscannen eines QR-Codes eine Aufgabe /Rätsel gestellt, deren Lösung ihn zur nächsten Station führt. Dort ist ein weiterer Code verborgen, mit dem er dann entweder direkt an die Belohnung kommt oder eine neue Aufgabe gestellt bekommt, um wiederum eine neue Station zu finden. Ein Cache kann bis zu 5 solcher Stationen (inkl. Endstation) umfassen. Am Ende werden dem Nutzenden je nach Umfang und Schwierigkeit des Cache, Punkte auf seinem Konto gutgeschrieben. Dabei ist es nicht möglich diese durch mehrfache Durchführung desselben Cache mehrmals zu erhalten. Unangemeldete Nutzende können keine Belohnung erhalten, ihren Fortschritt nicht speichern, sowie nicht in die Rangliste aufgenommen werden. Ausgeschlossen soll auch der Quereinstieg in einen Cache sein um eine Belohnung zu erhalten, wenn man zufällig auf die Endstation trifft. Es ist erforderlich, dass die Stationen eines Cache immer in der richtigen Reihenfolge freigeschaltet werden.

Um einen Cache zu erstellen, ruft der Administrierende in der Cache-Übersicht die Funktion "neuen Cache erstellen" auf. Nun müssen zunächst alle Informationen, wie Name, Beschreibung, Punktwert und Stationen eingegeben werden. Stationen werden über eine Liste hinzugefügt. Um eine neue Station hinzuzufügen werden in einer neuen Eingabemaske die dafür notwendigen Informationen abgefragt: Beschreibung, Rätsel (und ggf. Lösung) und der Ort der Station. Der Ort soll über einen Klick auf der Karte oder optional über Standortbestimmung festgelegt werden können. Wenn alle Informationen gespeichert sind, wird für die Station der Code generiert, über den der Fund mithilfe des QR-Codes abgeglichen und bestätigt werden kann.

Die Scan-Mechanik soll über eine Dritt-Bibliothek ins Frontend integriert werden. Dabei wird der QR-Code im Frontend in einen Code umgewandelt und im Backend mit dem bei der Station Hinterlegten abgeglichen.

Ein weiterer Punkt ist die Integration von Points-of-Interest aus anderen Teams, um diese, wenn vom Nutzenden gewünscht, auf unserer Karte anzeigen zu können. Dazu nutzen wir den Lageplan des entsprechenden Teams. Details zur Einbindung wurden mit der Gruppe Lageplan bereits besprochen und ein Protokoll hierzu ist in Confluence verfügbar.

Als Kriterien für Bewertung der Caches um sich in der Rangliste messen zu können gibt es folgende Ansatzpunkte:

Umfang des Cache (Anzahl Stationen)

Schwierigkeit der Rätsel

Der Einsatz dieser Kriterien und deren Balancing wird noch durch Feldtests mit echten Daten evaluiert.

Zusammenfassend müssen folgende Funktionen (als Epics formuliert) – hier aus Sicht der Rollen in der App – umgesetzt werden:

Administrierender	Cachender	Besuchender
-------------------	-----------	-------------

<ul style="list-style-type: none"> • Verwalten der Infoseiten • Verwalten der Caches, Rätsel und Belohnungen (inklusive Wahl des Standorts auf der Karte bei Caches /Stationen) • Verwalten der Rollen/Cachenden 	<ul style="list-style-type: none"> • Verwalten der gestarteten und gefundenen Caches • Verwaltung des eigenen Profils • Verwaltung eines eigenen Teams • Nutzende zum eigenen Team einladen • Einem anderen Team (auf Einladung) beitreten oder dieses verlassen. • Anzeige der gefundenen Stationen eines Cache auf der Karte • Freischaltung der nachfolgenden Station durch QR-Code 	<ul style="list-style-type: none"> • Übersicht aller verfügbaren Caches • Anzeigen der Startpunkte eines Cache auf der Karte • Aufrufen eines Cache oder einer bestimmten Station eines Caches über die Karte • Informationen über das Spiel und App bekommen

Verwalten bedeutet hier: Übersicht, Erstellen, Bearbeiten und Löschen der Daten/Elemente. Sämtliche Punkte unter "Cachender" gelten auch für "Administrierender". Sämtliche Punkte unter "Besuchender" gelten auch für "Cachender".

Definition of Done

Die Anforderungen werden als erledigt angesehen, wenn die entsprechende Story technisch vollständig funktionsfähig im Backend, sowie im Frontend implementiert ist, sprich die Akzeptanzkriterien aus dem Backlog alle erfüllt sind. Der Product Owner hat dabei die Entscheidungshoheit, ob die Story als final erledigt anerkannt wird. Die Akzeptanzkriterien werden in Jira, als auch im Product Backlog dokumentiert.

Ausschlüsse

Belohnungen werden nicht von uns bereitgestellt. Wir bieten dem Kunden hier lediglich einen Kanal an, um beispielsweise Vergünstigungen oder Gutscheine von Partnerunternehmen zu verteilen und den Besuchenden gleichzeitig auf unkonventionelle Art in die Welt der BuGa eintauchen zu lassen.

Planung

Wir werden unseren Daily Scrum werktags (außer Samstag) täglich über Discord abhalten. Die Uhrzeit für das Daily Scrum ist auf 13:00 Uhr angelegt. Dies beinhaltet die Mitteilung der Ergebnisse seit dem letzten Daily Scrum aus der Arbeit im Home Office und die Ziele/Probleme beim Erreichen der Ziele bis zum nächsten Daily Scrum. Ein Daily Scrum wird etwa 15 Minuten dauern. Die Dokumentation der Daily Scrums wird daraufhin vom Scrum Master in Confluence vorgenommen.

Außerhalb der Daily Scrums wird sich das Entwicklerteam Montags und Donnerstags um 13.15 treffen. Hier werden wir innerhalb von einer Stunde unsere weitere Vorgehensweise besprechen und uns gegenseitig unsere Ergebnisse präsentieren. Die Termine dienen auch zur Kommunikation zwischen Team Frontend und Team Backend, um gemeinsame Schnittstellen zu diskutieren und diese auszuarbeiten.

Hinzu werden unregelmäßige Treffen nötig sein, die wir vermutlich mittwochvormittags abhalten. Die Treffen werden wir im Voraus planen und veröffentlichen.

Ausserterminliche Treffen zwischen einzelnen Teammitgliedern können auch nötig sein, diese weit im Vorhinein zu planen ist jedoch nicht sinnvoll.

Fragen, die sich im Laufe des Tages ergeben, werden dabei oftmals über WhatsApp geklärt.

Außerhalb der Meetings werden wir in Einzelarbeit unterschiedlich viel Zeit aufwenden und dies in Jira dokumentieren. Wir gehen davon aus, dass die Gesamtarbeitszeit einer Person im Team wöchentlich 25 Arbeitsstunden beträgt, was für jeden einzelnen ungefähr 50 Arbeitsstunden pro zweiwöchigen Sprint bedeutet.

Die Sprint Review wird an dem letzten stetigen Termin der zuvor genannten Meetings (donnerstags) vor Ende des Sprints abgehalten. Hierzu werden wir uns zuvor einzeln mit maximal einer Stunde Zeitaufwand im Home Office vorbereiten und die Ergebnisse des Sprints dann innerhalb von maximal einer Stunde den Anwesenden präsentieren (timeboxed). Dies beinhaltet Zeit für konstruktive Kritik der Stakeholder und des Product Owners.

Zur Retrospective ist geplant, den Termin nach dem Sprint Review (ebenfalls donnerstags) zu nutzen. Hier wird ein kürzeres Zeitfenster von einer Stunde angesetzt. Diskutiert wird hier, was im Sprint positiv und negativ gelaufen ist. Aktive Teilnehmer sind hier das Entwicklerteam, der Scrum Master und der Product Owner.

Information der Auftraggeber

Die Auftraggebenden Prof. Dr. Nicola Marsden, Dr. phil. Kerstin Raudonat und Prof. Dr.-Ing. Gerald Permantier können den aktuellen Projektstand jederzeit in[[Jira](https://jira-student.it.hs-heilbronn.de/secure/RapidBoard.jspa?projectKey=GEOCACH&rapidView=263)][<https://jira-student.it.hs-heilbronn.de/secure/RapidBoard.jspa?projectKey=GEOCACH&rapidView=263>] und[[Confluence](https://confluence-student.it.hs-heilbronn.de/display/Geocaching2019/LabSWP_2019_SoSe_Geocaching)][https://confluence-student.it.hs-heilbronn.de/display/Geocaching2019/LabSWP_2019_SoSe_Geocaching] nachvollziehen. In Jira ist der Product Backlog mit den enthaltenen User Stories eingestellt, zusätzlich ist es möglich sich den aktuellen Sprint anzuzeigen, um somit zu sehen, an welchen Stories gerade gearbeitet wird. In Confluence werden Dateien und auch Ergebnisse der Meetings abgelegt. Außerdem existiert in Confluence eine API Dokumentation der Backend Aufrufe, die immer auf dem neuesten Stand gehalten wird. Im Reiter "Meeting notes" werden die Protokolle der Daily Scrums abgelegt, die Ergebnisse der Retrospektive findet man unter dem Reiter "Retrospectives".

Zuletzt findet man auch noch verschiedenste Dateien wie den Projektauftrag, Releaseplan und auch die Präsentationen im Reiter "Dateilisten". Meetings mit den Auftraggebenden werden Donnerstags stattfinden, diese dienen dazu, Fragen zu klären und auch Vorschläge der Auftraggebenden entgegen zu nehmen. Solche Meetings werden auf Wunsch der Auftraggebenden oder des Projektteams terminiert. Des Weiteren werden 3 Präsentationen abgehalten, dort wird der aktuelle Stand verdeutlicht und es wird auch auf den Projekt Ausblick eingegangen.

Diese Präsentationen finden zu folgenden Terminen statt:

- 28.03.2019
- 02.05.2019
- 27.06.2019

Diese Termine entsprechen Releases, in denen die in der Präsentation und dem Release Backlog genannten Features funktionsfähig eingebunden sind.

Projektorganisation

Projekt-Auftraggeber:

Prof. Dr.-Ing. Gerald Permantier

Prof. Dr. Nicola Marsden

Dr. phil. Kerstin Raudonat

Scrum Master:

Katharina Will

Product Owner:

Maximilian Leopold

Projektteam:


Timo Volkmann

Michael Dietz

Robin Christ

Releaseplan

Im Ordner "Zusätzliche Dokumentationsdateien" (USB-Stick) liegt der Releaseplan auch noch als Word oder PDF Datei vor.

Datei	Geändert 
PDF-Datei Releaseplan_GeoCaching.pdf	Mär 28, 2019 by Katharina Will
Microsoft Word Dokument Releaseplan_GeoCaching.docx	vor etwa 9 Stunden by Maximilian Leopold

Ziehen Sie Dateien an diese Stelle, um sie hochzuladen, oder [Dateien suchen](#)

 [Alle herunterladen](#)

Beschreibung	Start	Ende	Story Points	Priorität	Velocity
Sprint #1	11.03.2019	28.03.2019			11
Als Cachender will ich einen Cache von der Website aus starten können			3	1	
Als Cachender/Besuchender will ich eine Übersicht über alle Caches sehen können, um mir einen Überblick zu verschaffen			2	1	
Als Cachender/Administrierender will ich mich anmelden			3	1	
Als Cachender/Administrierender will ich mich abmelden			1	1	
Als Besucher der BuGa will ich Informationen über das GeoCaching Angebot erhalten			2	2	
Release #1			Summe: 11		
Sprint #2	29.03.2019	11.04.2019			17
Als Cachender/Besuchender will ich jederzeit die Rangliste einsehen können			2	2	
Als Administrierender will ich einen neuen Cache erstellen können			5	1	
Als Administrierender will ich Caches entfernen können			2	1	
Als Administrierender will ich eine neue Station erstellen können			3	1	
Als Cachender will ich meine angefangenen und abgeschlossenen Caches sehen können			5	2	
Sprint #3	12.04.2019	02.05.2019			18
Als Cachender will ich an QR-Stationen Rätsel lösen oder Aufgaben erfüllen, sodass ich dem Endpunkt des Cache näherkomme			13	1	
Als BuGa-Besuchender will ich mich registrieren			5	1	
Release #2			Summe: 35		
Sprint #4	03.05.2019	16.05.2019			14
Als Cachender will ich den Fortschritt meines Multicaches verfolgen können			3	2	
Als Cachender will ich Teams erstellen können			2	3	
Als Cachender will ich nachdem ich einen gefunden habe, eine Belohnung abholen können			1	2	
Als Cachender will ich die Startpunkte eines Cache auf der Karte sehen können			8	3	
Sprint #5	17.05.2019	06.06.2019			19
Als Cachender möchte ich andere zu meinem Team einladen können			8	3	
Als Cachender will ich einen QR-Code scannen können, sodass ich einen Cache starte			3	2	

Als Administrierender will ich Caches bearbeiten können			8	2	
Sprint #6	07.06.2019	27.06.2019			16
Als Cachender möchte ich meinen Standort auf der Karte sehen können			8	3	
Als Cachender möchte ich Caches von der Karte aus aufrufen können			8	3	
Release #3			Summe: 49		