**Tests Release 2**

**Geplantes Datum für Release 2:** 24. Mai 2018

Bei den Tests der folgenden Funktionen unserer Anwendung, wurde sich dazu entschieden, die Test mithilfe von JUnit durchzuführen und die Einträge, welche durch die Tests in der Datenbank erfolgten, wurden mithilfe von Screenshots dokumentiert.  
Folgende Test wurden geplant und durchgeführt und im nachfolgenden Abschnitt dokumentiert:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Funktion** | **Funktionsbeschreibung** | **Test** | **Testcode** | **Testbeschreibung** |
| 2.1 | Neue Personen erstellen | 2.1.1 | T211 | neue Person wird erstellt |
| 2.1 | Neue Personen erstellen | 2.1.2 | T212 | Vorname der Person setzten |
| 2.1 | Neue Personen erstellen | 2.1.3 | T213 | Nachname der Person setzen |
| 2.1 | Neue Personen erstellen | 2.1.4 | T214 | Vorname der Person ändern |
| 2.1 | Neue Personen erstellen | 2.1.5 | T215 | Nachname der Person ändern |
| 2.1 | Neue Personen erstellen | 2.1.6 | T216 | auf Vorname der Person zugreifen |
| 2.1 | Neue Personen erstellen | 2.1.7 | T217 | auf Nachname der Person zugreifen |
| 2.2 | Zuweisung von Personen zu einem Projekt | 2.2.1 | T221 | eine Person wird einem Projekt zugewiesen |
| 2.2 | Zuweisung von Personen zu einem Projekt | 2.2.2 | T222 | mehrere Personen werden zu einem Projekt zugewiesen |
| 2.3 | Zuweisung von Personen zu einem Aufgabenbereich | 2.3.1 | T231 | eine Person wird zu einem Aufgabenbereich zugewiesen |
| 2.3 | Zuweisung von Personen zu einem Aufgabenbereich | 2.3.2 | T232 | mehrere Personen werden zu einem Aufgabenbereich zugewiesen |
| 2.4 | Zuweisung von Personen zu einer Aufgabe | 2.4.1 | T241 | eine Person wird zu einer Aufgabe zugewiesen |
| 2.4 | Zuweisung von Personen zu einer Aufgabe | 2.4.2 | T242 | mehrere Personen werden zu einer Aufgabe zugewiesen |
| 2.5 | Zuweisung einer Person zu einer Aktivität | 2.5.1 | T251 | eine Person wird einer Aktivität zugewiesen |
| 2.6 | Personen ausgeben | 2.6.1 | T261 | alle Personen eines Projektes ausgeben |
| 2.6 | Personen ausgeben | 2.6.2 | T262 | alle Personen eines Aufgabenbereichs ausgeben |
| 2.6 | Personen ausgeben | 2.6.3 | T263 | alle Personen einer Aufgabe ausgeben |
| 2.6 | Personen ausgeben | 2.6.4 | T264 | Person einer Aktivität ausgeben |
| 2.7 | alle Projekte ausgeben | 2.7.1 | T271 | alle Projekte ausgeben |
| 2.8 | Aufgabenbereiche ausgeben | 2.8.1 | T281 | alle Aufgabenbereiche eines Projekts ausgeben |
| 2.9 | Aufgaben ausgeben | 2.9.1 | T291 | alle Aufgaben eines Aufgabenbereichs ausgeben |
| 2.10 | Aktivitäten ausgeben | 2.10.1 | T2101 | alle Aktivitäten einer Aufgabe ausgeben |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.1 | Neue Personen erstellen | 2.1.1 | T211 | neue Person wird erstellt |

**Auszug JUnit Code:**

@Test

**public** **void** T211CreatePerson() {

Person testPerson = **new** Person();

}

**Ergebnis in der Datenbank:**

table users:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.1 | Neue Personen erstellen | 2.1.2 | T212 | Vorname der Person setzten |

**Auszug JUnit Code:**

@Test

**public** **void** T212setFirstnamePerson() {

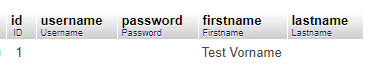
Person testPerson = **new** Person();

testPerson.setFirstName("Test Vorname");

}

**Ergebnis in der Datenbank:**

table users:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.1 | Neue Personen erstellen | 2.1.3 | T213 | Nachname der Person setzen |

**Auszug JUnit Code:**

@Test

**public** **void** T213setLastnamePerson() {

Person testPerson = **new** Person();

testPerson.setLastname("Test Nachname");

}

**Ergebnis in der Datenbank:**

table users:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.1 | Neue Personen erstellen | 2.1.4 | T214 | Vorname der Person ändern |

**Auszug JUnit Code:**

@Test

**public** **void** T214changeFirstnamePerson() {

Person testPerson = **new** Person();

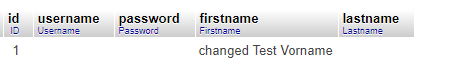
testPerson.setFirstName("Test Vorname");

testPerson.setFirstName("changed Test Vorname");

}

**Ergebnis in der Datenbank:**

table users:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.1 | Neue Personen erstellen | 2.1.5 | T215 | Nachname der Person ändern |

**Auszug JUnit Code:**

@Test

**public** **void** T215changeLastnamePerson() {

Person testPerson = **new** Person();

testPerson.setLastname("Test Nachname");

testPerson.setLastname("changed Test Nachname");

}

**Ergebnis in der Datenbank:**

table users:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.1 | Neue Personen erstellen | 2.1.6 | T216 | auf Vorname der Person zugreifen |

**Auszug JUnit Code:**

@Test

**public** **void** T216getFirstnamePerson() {

Person testPerson = **new** Person();

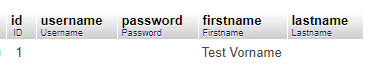
testPerson.setFirstName("Test Vorname");

String vorname = testPerson.getFirstName();

}

**Ergebnis in der Datenbank:**

table users:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.1 | Neue Personen erstellen | 2.1.7 | T217 | auf Nachname der Person zugreifen |

**Auszug JUnit Code:**

@Test

**public** **void** T217getLastnamePerson() {

Person testPerson = **new** Person();

testPerson.setLastname("Test Nachname");

String nachname = testPerson.getLastname();

}

**Ergebnis in der Datenbank:**

table users:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.2 | Zuweisung von Personen zu einem Projekt | 2.2.1 | T221 | eine Person wird einem Projekt zugewiesen |

**Auszug JUnit Code:**

@Test

**public** **void** T221addPersonToProject() {

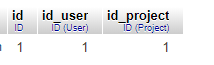
Person testPerson = **new** Person();

testProject.addPerson(testPerson);

}

**Ergebnis in der Datenbank:**

table user\_project:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.2 | Zuweisung von Personen zu einem Projekt | 2.2.2 | T222 | mehrere Personen werden zu einem Projekt zugewiesen |

**Auszug JUnit Code:**

@Test

**public** **void** T222addPersonsToProject() {

Person testPerson1 = **new** Person();

Person testPerson2 = **new** Person();

Person testPerson3 = **new** Person();

testProject.addPerson(testPerson1);

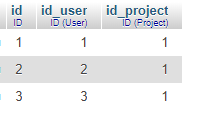
testProject.addPerson(testPerson2);

testProject.addPerson(testPerson3);

}

**Ergebnis in der Datenbank:**

table user\_project:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.3 | Zuweisung von Personen zu einem Aufgabenbereich | 2.3.1 | T231 | eine Person wird zu einem Aufgabenbereich zugewiesen |

**Auszug JUnit Code:**

@Test

**public** **void** T231addPersonToTaskGroup() {

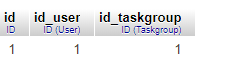
Person testPerson = **new** Person();

testTaskGroup.addPerson(testPerson);

}

**Ergebnis in der Datenbank:**

table user\_taskgroup:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.3 | Zuweisung von Personen zu einem Aufgabenbereich | 2.3.2 | T232 | mehrere Personen werden zu einem Aufgabenbereich zugewiesen |

**Auszug JUnit Code:**

@Test

**public** **void** T232addPersonsToTaskGroup() {

Person testPerson1 = **new** Person();

Person testPerson2 = **new** Person();

Person testPerson3 = **new** Person();

testTaskGroup.addPerson(testPerson1);

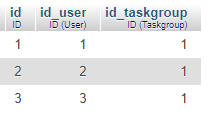
testTaskGroup.addPerson(testPerson2);

testTaskGroup.addPerson(testPerson3);

}

**Ergebnis in der Datenbank:**

table user\_taskgroup:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.4 | Zuweisung von Personen zu einer Aufgabe | 2.4.1 | T241 | eine Person wird zu einer Aufgabe zugewiesen |

**Auszug JUnit Code:**

@Test

**public** **void** T241addPersonToTask() {

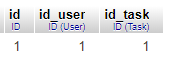
Person testPerson = **new** Person();

testTask.addPerson(testPerson);

}

**Ergebnis in der Datenbank:**

table user\_task:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.4 | Zuweisung von Personen zu einer Aufgabe | 2.4.2 | T242 | mehrere Personen werden zu einer Aufgabe zugewiesen |

**Auszug JUnit Code:**

@Test

**public** **void** T242addPersonsToTask() {

Person testPerson1 = **new** Person();

Person testPerson2 = **new** Person();

Person testPerson3 = **new** Person();

testTask.addPerson(testPerson1);

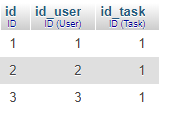
testTask.addPerson(testPerson2);

testTask.addPerson(testPerson3);

}

**Ergebnis in der Datenbank:**

table user\_task:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.5 | Zuweisung einer Person zu einer Aktivität | 2.5.1 | T251 | eine Person wird einer Aktivität zugewiesen |

**Auszug JUnit Code:**

@Test

**public** **void** T251addPersonToActivity() {

Person testPerson = **new** Person();

testActivity.addPerson(testPerson);

}

**Ergebnis in der Datenbank:**

table activity:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.6 | Personen ausgeben | 2.6.1 | T261 | alle Personen eines Projektes ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T261getAllPersonsFromProject() {

Person testPerson1 = **new** Person();

Person testPerson2 = **new** Person();

Person testPerson3 = **new** Person();

testProject.addPerson(testPerson1);

testProject.addPerson(testPerson2);

testProject.addPerson(testPerson3);

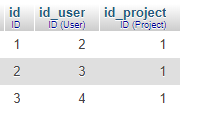
List<Person> projectList = **new** ArrayList<>();

projectList = testProject.getPersonList();

}

**Ergebnis in der Datenbank:**

Table user\_project:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.6 | Personen ausgeben | 2.6.2 | T262 | alle Personen eines Aufgabenbereichs ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T262getAllPersonsFromTaskGroup() {

Person testPerson1 = **new** Person();

Person testPerson2 = **new** Person();

Person testPerson3 = **new** Person();

testTaskGroup.addPerson(testPerson1);

testTaskGroup.addPerson(testPerson2);

testTaskGroup.addPerson(testPerson3);

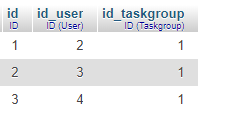
List<TaskGroup> taskGroupList = **new** ArrayList<>();

taskGroupList = testTaskGroup.getPersonList();

}

**Ergebnis in der Datenbank:**

user\_taskgroup



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.6 | Personen ausgeben | 2.6.3 | T263 | alle Personen einer Aufgabe ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T263getAllPersonsFromTask() {

Person testPerson1 = **new** Person();

Person testPerson2 = **new** Person();

Person testPerson3 = **new** Person();

testTask.addPerson(testPerson1);

testTask.addPerson(testPerson2);

testTask.addPerson(testPerson3);

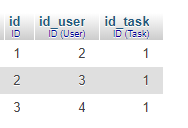
List<Task> taskList = **new** ArrayList<>();

taskList = testTask.getPersonList();

}

**Ergebnis in der Datenbank:**

table user\_task:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.6 | Personen ausgeben | 2.6.4 | T264 | Person einer Aktivität ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T264getPersonFromActivity() {

Person testPerson = **new** Person();

testActivity.addPerson(testPerson);

Person activityPerson = testActivity.getPerson();

}

**Ergebnis in der Datenbank:**

table activity:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.7 | alle Projekte ausgeben | 2.7.1 | T271 | alle Projekte ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T271getAllProjects() {

Project testProject1 = **new** Project();

Project testProject2 = **new** Project();

Project testProject3 = **new** Project();

testProjectList.addProject(testProject1);

testProjectList.addProject(testProject2);

testProjectList.addProject(testProject3);

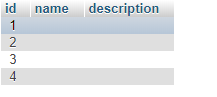
List<Project> projectList = **new** ArrayList<>();

projectList = testProjectList.getProjectList();

}

**Ergebnis in der Datenbank:**

table projects:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.8 | Aufgabenbereiche ausgeben | 2.8.1 | T281 | alle Aufgabenbereiche eines Projekts ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T281getAllTaskGroupsFromProject() {

TaskGroup testTaskGroup1 = **new** TaskGroup();

TaskGroup testTaskGroup2 = **new** TaskGroup();

TaskGroup testTaskGroup3 = **new** TaskGroup();

testProject.addTaskGroup(testTaskGroup1);

testProject.addTaskGroup(testTaskGroup2);

testProject.addTaskGroup(testTaskGroup3);

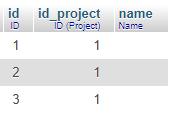
List<TaskGroup> taskGroupList = **new** ArrayList<>();

taskGroupList = testProject.getTsks();

}

**Ergebnis in der Datenbank:**

table taskgroup:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.9 | Aufgaben ausgeben | 2.9.1 | T291 | alle Aufgaben eines Aufgabenbereichs ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T291getAllTasksFromTaskGroup() {

Task testTask1 = **new** Task();

Task testTask2 = **new** Task();

Task testTask3 = **new** Task();

testTaskGroup.addTsk(testTask1);

testTaskGroup.addTsk(testTask2);

testTaskGroup.addTsk(testTask3);

List<Task> taskList = **new** ArrayList<>();

taskList = testTaskGroup.getTaskList();

}

**Ergebnis in der Datenbank:**

table tasks:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Funktion | Funktionsbeschreibung | Test | Testcode | Testbeschreibung |
| 2.10 | Aktivitäten ausgeben | 2.10.1 | T2101 | alle Aktivitäten einer Aufgabe ausgeben |

**Auszug JUnit Code:**

@Test

**public** **void** T2101getAllActivitesFromTask() {

Activity testActivity1 = **new** Activity();

Activity testActivity2 = **new** Activity();

Activity testActivity3 = **new** Activity();

testTask.addActivity(testActivity1);

testTask.addActivity(testActivity2);

testTask.addActivity(testActivity3);

List<Activity> activityList = **new** ArrayList<>();

activityList = testTask.getActivities();

}

**Ergebnis in der Datenbank:**

table activity:

