**TDTP**

**Documentation**

Academic year: 2021-2022, Spring Semester

Team Name: Team 2

Team members:

- Mardar Stefania-Florentina

- Raileanu Alexandru

- Doreata Andrei-Stefan

- Peptenaru Marin-Vasile

- Babeii Denisa

Contents

[1. Application Details. Investigated Features 3](#_Toc68371936)

[2. AC. IOs 3](#_Toc68371937)

[3. Testing Mission 3](#_Toc68371938)

[4. Testing Strategy 3](#_Toc68371939)

[5. Selected Test Design Techniques 4](#_Toc68371940)

[6. [optional] Test Design Techniques Improvement 4](#_Toc68371941)

[7. Test Design. Test implementation. Test execution. Test Report 4](#_Toc68371942)

[7.1. Test Design 4](#_Toc68371943)

[7.2. Test Implementation. Test Execution 5](#_Toc68371944)

[7.3. Test Report 5](#_Toc68371945)

[8. Issue Reporting 5](#_Toc68371946)

[9. Conclusions. Lessons Learned 6](#_Toc68371947)

# Application Details. Investigated Features

The tested web application deals with managing office buildings for a company, providing data regarding bookings, buildings, users and desks.

The user is able to add bookings for personal use or for other users and view all existing bookings. In order to have a better overview of the available floors, there is a map displayed with the desks’ placement and information. This information includes existing equipment, desk type and availability.

In the Buildings section the user has the possibility to either add a new building or to manage the existing buildings, being able to edit the information about the buildings or to enter additional information related to the floors, such as floor name, map and details.

In the Buildings section the user has the possibility to either add a new building, or to manage the existing buildings, being able to edit the information about the buildings or to add additional information related to the floors.

TBA: Descriptions for Users, Desks and Reports tabs

# AC. IOs

**Application Context**: You are part of a beginner testing team, which is very passionate and ambitious. The testing team is given an almost finished product to be tested.

**Part I**: The project manager wants your team to test the latest sprint delivered to the customer.

**Part II**: The customer wants the team to check a list of scenarios for the use of the website.

# Testing Mission

**Testing mission for Part I**

TBA: The latest sprint covered

* Search User
* Add New Building
* Etc.

We are planning on testing these features, in order to make sure that they work as expected after the delivery.

**Testing mission for Part II**

We want to check that scenarios provided by the customer work with certainty.

# Testing Strategy

**Testing strategy for Part I**

This part will follow a regression-averse strategy, because we want to make sure that the newly introduced features work well with the existing ones.

**Testing strategy for Part II**

This part will follow a methodical and directed strategy, because the customer provides us with a list of the fundamental features. The customer is outside of the test team and outside the organization itself.

TBA: This list includes checking the following features:

- Manage Equipment

- User Presence Reports

- Etc.

# Selected Test Design Techniques

*<Include details (1 paragraph) on the test design techniques selected. Emphasize the different test attributes and dimensions (coverage, risk, activity, etc.). The investigation of the same technique following two dimensions counts as distinct techniques) in order to address the AC+IOs covered. At the end of this section you can use the following table to summarize your work. See* ***Lecture01 slides 39-55, 58*** *and* ***Lecture04+Lecture06 for test attributes*** *details.>*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Part** | **Test Strategy** | **Test Design Technique** | **Test Attributes** | **Dimension covered** | **Students and Features** |
| Part I | Regression-averse | Feature Integration Testing | Coverage, Information value, Easy to evaluate | Coverage | Babeii Denisa (Features Search User, View User Information) |
| Boundary Testing | Coverage, Easy to evaluate, Motivating, Value | Coverage | Mardar Florentina-Stefania (Feature Add New Building) |
| Quick Tests (QT) | Power, Representative, … | Risk | Student5 (Feature Y) |
| Scenario-based Testing (ST) | Credible, Motivating, … | Activity | Student4 (Features, X, Z, Y) |
| Part II | Methodical, Directed | Function Testing | Coverage, Easy to evaluate, Motivating | Scope | Babeii Denisa (Feature Manage Equipment) |
|  |  | Feature Integration Testing | Information value (tester), Coverage, Easy to evaluate, Validity | Coverage | Mardar Florentina-Stefania (Feature User Presence Reports) |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Test Design. Test implementation. Test execution. Test Report

# Test Design

*<Include all the information associated to test design when a specific test design technique is used. Each team member will fill the details corresponding to Part I and Part II. The table below indicates a sample for Part Il.>*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Information objective (Part I):* The project manager wants your team to test the latest sprint delivered to the customer. | | | | |
| **Student** | **Feature(s)** | **Test Design Technique** | **Details** | **Input, Expected output** |
| *Babeii Denisa* | Features Search User, View User Information | Feature Integration Testing | Check whether the newly added feature ‘Search User’ behaves well with older feature ‘View User Information’ | |  |  | | --- | --- | | ***Input*** | ***Expected Output*** | | *Search for user ‘Babeii’ and enter its corresponding information page* | *Assert that the information page is displayed and belongs to the searched user* | |
| *Mardar Florentina Stefania* | *Feature Add New Building* | *Boundary Testing (BVA)* | *We are testing the boundaries of our Equivalent Classes (EC) variables: buildingName and safeDistance. We need to find the Boundary Value Conditions (BVC) built on the limits of the valid EC.*  *Properties:*  *buildingName – String -> cannot start with “-”, cannot have only one character which is a digit (only one letter character is accepted)*  *safeDistance - float -> no upper limit, no negative numbers, no letters excepting “e”, no special characters excepting “.”* | |  |  |  | | --- | --- | --- | | ***Variable*** | ***Valid*** | ***Invalid*** | | *Building Name* | *Test* | *0* | | *00* | | *X* | *1* | | *123* | *Starting with ‘-[0-9]’* | | *Any combination starting with a digit or letter* | | *Safe Distance* | *1* | *0* | | *2* | *-1* | | *1234* | *Letters* | | *1.456* | *Special charact.* | | *[0, 10..00]* | *(-inf, 0)* |  |  |  |  | | --- | --- | --- | | ***Variable*** | ***Bounds*** | ***BVC*** | | *Building Name* | *Lower Bound* | *1. 0* | | *2. 00* | | *3. abc* | | *Upper Bound* | *No upper bound* | | *Safe Distance* | *Lower Bound* | *1. -1* | | *2. 0* | | *3. 1* | | *Upper Bound* | *No upper bound* | |
| *Student4* | *Feature Z* | *LE* | *Business rules for variable a and b are tested* | |  |  |  | | --- | --- | --- | | ***Variable*** | ***Rule1*** | ***Rule2*** | | *a* | *a>0* | *a<=0* | | *b* | *b>0* | *b<=0* | | *result* | *False* | *True* | |  |  |  | |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Information objective (Part II):* The customer wants the team to check a list of scenarios for the use of the website. | | | | |
| **Student** | **Feature(s)** | **Test Design Technique** | **Details** | **Input, Expected output** |
| *Babeii Denisa* | Feature Manage Equipment | Function Testing | Focus on Manage Equipment feature | |  |  | | --- | --- | | ***Input*** | ***Expected Output*** | | *Add new equipment* | *The equipment list is incremented* | | *Add the same equipment twice* | *The equipment list is incremented once* | |
| *Mardar Florentina Stefania* | Feature  User  Presence Reports | Feature  Integration Testing | Check if the report regarding user presence is updated correctly after a new booking. | |  |  | | --- | --- | | ***Input*** | ***Expected Output*** | | *Add a new booking for a specific desk* | *The new booking data is added in the User Presence Table from the REPORTS.* | |
| *Student4* | *Feature Z* | *LE* | *Business rules for variable a and b are tested* | |  |  |  | | --- | --- | --- | | ***Variable*** | ***Rule1*** | ***Rule2*** | | *a* | *a>0* | *a<=0* | | *b* | *b>0* | *b<=0* | | *result* | *False* | *True* | |  |  |  | |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Test Implementation. Test Execution

*<Each student will choose* ***ONE*** *of the test design techniques she/he included in the previous section. For the picked technique the test cases designed will be implemented using an automation framework (see* ***Lab04****) (SeleniumWebDriver + SerenityBDD, Postman, JMeter) should be emphasized. The team can decide if each student will have her/his own automation project, or they will share the same project on git such that it will include a package with the tests implemented by each team member. The table below will consist of the test cases implemented by each student.>*

|  |  |  |  |
| --- | --- | --- | --- |
| **Part** | **Student** | **Feature(s)** | **Input, Expected Output, Actual Output** |
| *Part II* | *Babeii Denisa* | Feature Manage Equipment | |  |  |  |  | | --- | --- | --- | --- | | ***TCs*** | ***Input*** | ***Expected Output*** | ***Actual Output*** | | *testAddNonExistentEquipment* | *Add new equipment* | *The equipment list is incremented* | ***Passed*** | | *testAddExistentEquipment* | *Add the same equipment twice* | *The equipment list is incremented once* | ***Passed*** | |
| *Part I* | *Mardar Florentina Stefania* | Feature Add New Building | |  |  |  |  | | --- | --- | --- | --- | | ***TCs*** | ***Input*** | ***Expected Output*** | ***Actual Output*** | | *test\_DOES\_NOT\_ADD\_WHEN\_invalid\_*  *\_safeDistance\_zero* | *buildingName = ‘Test\_Building’,*  *safeDistance = 0* | *False* | ***False*** | | *test\_ADD\_new\_building\_WHEN\_valid\_*  *\_inputs\_safeDistance\_in\_interval* | *buildingName = ‘Test\_Building’,*  *safeDistance = 2* | *True* | ***True*** | | *test\_ADD\_new\_building\_WHEN\_valid\_*  *\_inputs\_safeDistance\_lower\_bound* | *buildingName = ‘Test\_Building’,*  *safeDistance = 1* | *True* | ***True*** | | *test\_ADD\_new\_building\_WHEN\_valid\_*  *\_inputs\_name\_accepted* | *buildingName = ‘abc’,*  *safeDistance = 100* | *True* | ***True*** | | *test\_DOES\_NOT\_ADD\_WHEN\_invalid\_*  *\_name\_zero* | *buildingName = ‘0’,*  *safeDistance = 100* | *False* | ***False*** | |
| *Part I* | *Student4* | *Feature Z* | |  |  |  |  | | --- | --- | --- | --- | | ***TCs*** | ***Input*** | ***Expected Output*** | ***Actual Output*** | | *TC01* | *a=3, b=5* | *False* | ***False*** | | *TC02* | *a=-1, b=-3* | *True* | ***False*** | | *…* |  |  |  | |  |  |  |  | |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Test Report

*<This section will include the reports of the test execution, e.g., pie charts generated by the used tool, with #TCs run = #TCs passed + #TCs failed.>*

# Issue Reporting

*<This section includes the description of* ***at least one issue*** *found while performing testing. This can be a coding bug or design issue. The provided details will focus on the RIMGEA bug reporting strategy. Highlight 2-3 relevant RIMGEA elements for the described issue(s). In order to report the bug or issue to Evozon, please use the corresponding template (see* **IssueReport**).*>*

# Conclusions. Lessons Learned

*<Please include in this section final conclusions, lessons learned and personal considerations while working on TDTP (3-4 paragraphs). You can focus on the following aspects: type of application to be tested, amount of knowledge to use (related or not to testing), tools required to apply, team collaboration, test project organization, collaboration with the software company, amount of time needed to fulfil the tasks, etc.>*