Software Engineering 2 - Prof. Di Nitto Elisabetta
Dipartimento di Elettronica, Informazione e Bioingegneria
Politecnico di Milano

CodeKataBattle

ATD
Acceptance Test deliverable

February 8, 2024

Marco Cerino 244780
Mattia De Bartolomeis 245022

Contents

1 Introduction

The project analysed in this paper was developed by Antonio Di Paola and Conti. Relevant details on the project are given below:

- **Authors of the Project:** Conti, Antonio Di Paola
- **Link to the Project Repository:** `https://github.com/Dipa0219/ContiDiPaola`
- **Main Reference Documents:**
  - RASD
  - DD
  - ITD
  - Github Repository

2 Installation Setup

During the installation phase of the prototype, I followed the guide provided in the 'INSTALLATION INSTRUCTION' section of the ITD reference. The project was implemented as a Java web application, requiring a web server and an SQL database to run it. The steps followed are summarised below, together with the problems encountered.
We started by installing IntelliJ and importing the project. Next, we configured the project as a Maven module and set up Apache Tomcat as the web server. The import of the database provided in the repository was done via MySQL Workbench. During these steps, I followed the instructions specified in the ITD to correctly configure the development environment.

2.1 Problems and Inconsistencies

During the installation process, I encountered some problems and inconsistencies:

- The documentation did not clearly specify the version of Apache Tomcat and MySQL Workbench to be used.
- The ITD did not mention the need to install MySQL Server in order to use MySQL Workbench. This omission could lead to confusion for less experienced users, who might not be aware of the need to have an active database server to manage the database via Workbench.

3 Acceptance test cases

To start testing the project, we analysed the ITD to find out what requirements were implemented in this prototype. Unfortunately, however, we could not find the information of interest and analysed requirement by requirement listed in the RASD. It was decided to carry out manual tests, directly via the user interface of the web app. These are the results of the tests:

- System allows students to sign up:

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Sign up student	1. Go to the website. 2. Click on Sign up. 3. Compile form. 4. Click on Sign in.	1. Role: Student 2. Name: Mattia 3. Surname: De Bartolomeis 4. Birthdate: 11/09/2001 5. Username: matamattia 6. Email: matamattia@gmail.com 7. Password: password 8. Github Username : matamattia	We are sending you a confirmation email Please check your email	Test passed because the student was succesfully registered. However no confirmation e-mail was sent as reported by the output message and in the scenarios of RASD

- System allows educators to sign up

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**

Sign up educator	1. Go to the website. 2. Click on Sign up. 3. Compile form. 4. Click on Sign in.	1. Role: Educator 2. Name: Marco 3. Surname: Cerino 4. Birthdate: 12/04/2001 5. Username: marco 6. Email: marco@gmail.com 7. Password: password 8. Github Username : marcocerino	We are sending you a confirmation email Please check your email	Test passed because the student was succesfully registered. However no confirmation e-mail was sent as reported by the output message and in the scenarios of RASD.

- System allows students to login

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Login student	1. Go to the website. 2. Click on Login. 3. Compile form. 4. Click on login.	1. Username: matamattia 2. Password: password	Redirected to the dashboard	Test passed

- System allows educators to login

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Login educator	1. Go to the website. 2. Click on Login. 3. Compile form. 4. Click on login.	1. Username: marco 2. Password: password	Redirected to the dashboard	Test passed

- System allows educator to create a new tournament and to add all the needed information

- System notifies students about the creation of a new tournament

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Tournament creation and notication	1. Log as educator 2. Click on Create New Tournament. 3. Compile form. 4. Click on submit.	1. Tournament name: NewTournament 2. Tournament Description: TestDescription 3. Registration Deadline: 17/02/2024	Tournament created and educator notified via email.	Test passed

- System allows educator to grant the permission to other colleagues to create battle in a specific tournament.

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Tournament permission	1. Log as educator 2. Click on the tournament. 3. Click on Add Collaborator. 4. Select a collaborator. 5. Click Submit.	1. Tournament name: NewTournament 2. Tournament Description: TestDescription 3. Registration Deadline: 17/02/2024	The interface does not display any confirmation message; however, if you check the dashboard of the educator who has been added, you can see that the tournament in question has indeed been added to his or her list of tournaments.	Test passed

- System allows educator to create a battle inside a particular tournament by adding CK and the deadlines

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**

Battle creation	1. Log as educator 2. Click on the tournament. 3. Click on create a battle 4. Compile form. 5.	1. Battle name: NewBattle 2. Battle description : descriptionTest 3. Registration deadline: 10/02/2024 18:30 4. Submission deadline: 10/02/2024 19:00 5. Minimum number of student:1 6. Maximum number of student: 3 7. Battle project: file zip 8. Battle test case: file.yaml	Battle succesfully created.	Test passed

Note: In order for the educator to display the button to create the tournament, the tournament registration deadline must expire. This constraint is not stated in the scenarios and use cases of the rasd.

- System allows educator to close a tournament.

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Tournament closure	1. Log as educator 2. Click on the tournament. 3. Click on close tournament	Tournament name: NewTournament	If the tournament has no scheduled battles: Error message: The tournament is not closable now. If the tournament has sceduled battles: Error message: 'You can't acces in this page'.	Test undetermined

Since the conditions on the closure of the tournament are not well specified in the

rasd, it is difficult to say whether the test has failed, or for some reason under those conditions the educator is unable to close the tournament.

- System allow student to join a tournament

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Student registers for the tournament	1. Log as student 2. Search in the search bar for the name of the tournament, (presumably known from the e-mail notification) 3. Click on the tournament. 4. Click on join tornament.	Tournament name: NewTournament	A confirmation message is not displayed, but the button allowing you to join the tournament disappears. Furthermore the student can see his name in the ranking of the tournament, so he is succesfully registered for the tournament.	Test passed

- System allows students to search tournament by key words

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Search for a tournament	1. Log as student 2. Search in the search bar for the name of the tournament, 3. Click on submit.	Tournament name: NewTournament	The searched tournament, if exists, is displayed in the dashboard	Test Passed

- System allows educator to see rankings of each tournament, which he created or in which he has been nominated collaborator.

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**

Visualize tournament ranking	1. Log as educator 2. Search in the search bar for the name of the tournament, 3. Click on the tournament.	Tournament name: NewTournament	Tournament details including ranking.	Test Passed

- System allows educator to see rankings of each tournament, which he created or in which he has been nominated collaborator.

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Visualize tournament ranking	1. Log as educator who has not permission for the tournament. 2. Search in the search bar for the name of the tournament, 3. Click on the tournament.	Tournament name: NewTournament	Tournament details including ranking.	Test Failed

- System notifies students about the creation of a new battle in a specific tournament in which they have subscribed

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Battle creation notification.	1. Create a battle in which a student is subscribed		Nothing happens	Test Failed

- System allows students to see rankings of each tournament, which they are sub scripted

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**

Visualiza tournament ranking as student	1. Log as student 2. Search in the search bar for the name of the tournament, 3. Click on submit. 4. Click on the tournament	Tournament name: newTournament	Torunament details including ranking.	Test Passed

- System allows students to see rankings of each tournament, which they are sub scripted

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**
Non-registered student tries to view a tournament ranking	1. Log as student 2. Search in the search bar for the name of a tournament in which he is not subscribed. 3. Click on submit. 4. Click on the tournament	Tournament name: newTournament	Torunament details including ranking.	Test Failed

- System allow student to join a battle

Test Type	**Test Description**	**Test Input**	**Test Output**	**Pass/Fail**

Battle registration	1. Log as student 2. Search in the search bar for a tournament 3. Click on submit. 4. Click on the tournament 5. Click on a battle of the tournament 6. Attempt to register to the battle	– Tournament name: newTournament – Battle name: NewBattle	None, impossible to register for a battle	Test Failed

When a student clicks on the battle, a page appears showing the details of the battle, but no option is displayed for the student to register for the battle.

3.1 Requirements that cannot be tested manually

Due to the limitation found in the battle registration functionality, the following requirements could not be tested:

- System allows educator to see ranking of a specific battle
- System allows educator to modify the evaluation manually of a specific group in a battle
- System notify student about publication of final ranking of a specific battle
- System notify student about publication of final ranking of a specific tournament
- System allows students to see ranking of a specific battle
- System, at end of the registration period of a battle, create a GH repo and send invitation to all member of the group

4 Quality of code and documentation

4.1 Comments on the code

The project follows a conventional structure of a Java application, with the main source code organised in the `src/main/java` directory and the test code in `src/test/java`. Within these directories, the code is divided into packages reflecting the various components and responsibilities of the system, as follows:

- **servlet**: It contains the servlet classes for handling HTTP requests.
- **bean**: It includes the JavaBean classes used to encapsulate objects.
- **DAO**: It represents access to data via DAO objects.
- **utils**: It provides utility classes that support various cross functionalities of the project.

The documentation within the code varies between components. The most relevant classes and methods include descriptive comments explaining their function and use. However, some areas of the code may benefit from more detailed documentation, especially those that implement complex or less intuitive logic.

4.2 Comments on documents

RASD
The RASD document submitted for the software project is well structured overall and provides a solid basis for understanding the system requirements and specifications. However, some areas were identified that require further improvement to ensure that the documentation is fully aligned with the needs of the project

- **Scenarios and Use Cases**: The scenarios and use cases presented in the document do not accurately describe the operational flows as they occur in the reality of the web application. For example, the creation of a battle within a tournament omits a crucial precondition: the tournament registration deadline. This omission could lead to misunderstandings about the actual functioning dynamics of the system. Similarly, the tournament closure procedure is not adequately detailed, as evidenced by problems encountered during the testing phase, where the absence of sufficient conditions for closure generated error messages.
- **System Requirements**: It turned out that the list of requirements is not complete, particularly with respect to what was outlined in the initial project outline. A significant example concerns the functionality that should allow students to create teams. This aspect, which is fundamental for user interaction with the system, is not adequately mentioned in the current requirements. Its inclusion is essential to ensure that the system fully meets the project objectives.

DD
The DD presented for the project is clear and well organised. Its well-articulated composition facilitates the understanding of architectural choices, implementation strategies and interactions between the various system components.
ATD
The requirements section implemented in the ITD deserves special attention to ensure that there is a clear and direct correspondence with the requirements listed in the RASD. Currently, there are ambiguities in the documentation that make it difficult to discern precisely which functionalities have actually been implemented and which remain to be implemented. This lack of clarity can lead to misunderstandings for testers, as they do not actually know which requirements to test.

5 Time Spent

Name	**Hours**
Mattia De Bartolomeis	4
Marco Cerino	4

6 References

- DD of the revised group
- RASD of the revised group
- ATD of the revised group
- Project of the revised group