POLITECNICO
MILANO 1863

CLup - Customer Line Up
Acceptance Testing Document

Project Details
Course: Software Engineering 2

Academic Year: 2020/2021

Reference Professor: Elisabetta Di Nitto

Contributors:
-Ludovica Lerma
-Federico Mainetti Gambera
https://github.com/LudoLe/LermaMainettiGambera.git

TABLE OF CONTENT

1.Introduction

1.A Tested Group

We tested the implementation made by:

- Francesco Attorre
- Thomas Jean Bernard Bonenfant
- Veronica Cardigliano

their repo can be found at:
https://github.com/FrancescoAttorre/softeng2-attorre-bonenfant-cardigliano

Before ideating the test cases for this acceptance test, we carefully read the DD and RASD they delivered on their repo.

1.B Project Overview

The team we tested implemented a JEE based server hosted on Tomcat which relies on a MySql database. For what concerns the client side, they developed part of the Web Application. In particular, they implemented four views: the home page, the one for the registration of *Activities* on the system, the one for the login of *Activities* and the one that an *Activity* uses to register a new building on the system.

the specific meaning of term “Activity” can be found in the RASD of the team which has been subject of the testing of this document.

2. Installation and set-up

The installation and set-up instructions we found on the I&T Document provided by the other team was clear and easy to follow.

However, we faced a problem related to the database. Indeed, the first time we ran their application we encountered a *server internal error* related to the fact that no database was found. We had to ask the other group for a dump of their database, since none was found on their github repository, and to import it manually through MySql Workbench in order for the application to work properly.

3. Test Cases

Overview

Since we only had the chance to a really limited amount of pages, we cannot really have a clear image of how this application was implemented and developed. Indeed, only four UI views were released by the developing team and only three of those were actually really testable since the home page consists only of a blank page with two hyperlinks, “Login” and “Register”.

The application offer few functionalities in the client side, wich are mapped to the goals [G1] and [G2] of the RASD and which corresponds to the “Tested Features” listed in the table below.

The application also has huge security issues. This is due to the fact that the user inputs’ are only checked by front-end code, and not even properly. Indeed, only the fact that the input cannot be empty is checked by front-end code. Once escaped few basic front-end code security features, basically no controls are made by the server-side, and all of the nonsense inputs go straight to the DB.

Furthermore, some of the inputs result in not handled server exceptions.

Finally, some of the features implemented are not practical: every registered Activity which logs in the system receives a token as an identification, this token is required, for example, for the Activity in order to be allowed to register a new Building on the system. However, this token only lasts one minute, which is a ridiculous little amount of time for the developer to expect the user to fulfil the form, or for us to manually make POST/GET http request to test the API.

Test Cases Reference Table

In the following pages, we show a reference table of the ***test cases*** for each ***goal*** of the application.

The green color highlights the tests passed by the application.

The red color highlights the tests which the application did not pass.

The orange color highlights the partially passed tests.

for further details about “tested requirements” column please read the RASD document of the team which has been subject of the testing of this document.

ID	Tested Requirements	Tested Feature	TEST	RESULT	DETAILS
0	-	installation	installation	problematic	-missing instructions -missing components -resolved after asking the required components to the developing team
1	[R1] [R2]	Registration of an Activity	Base functioning providing correct credentials	passed	
2	[R1] [R2]	Registration of an Activity	Empty Credentials	not passed	once evaded the client side security features, the server let us registered with all of the fields left empty and answered with a “OK” response status. This explicitly contradicts the [R2].
3	[R1] [R2]	Registration of an Activity	-Already in use username -Other credentials left empty	not passed	The Server responded with a “OK” response status. On the view appears a message “Could not register *username*”. This explicitly contradicts the [R2].
4	[R1] [R2]	Registration of an Activity	-Already in use username -Other credentials filled	partially passed	The Server responded with a “OK” response status. On the view appears a message “Could not register *username*”.Although an “OK” Status of the response was received
5	[R1] [R2]	Registration of an Activity	-Already in use P.IVA -Other credentials filled	partially passed	The Server responded with a “OK” response status. On the view appears a message “Could not register *username*”.Although an “OK” Status of the response was received
6	[R1] [R2]	Registration of an Activity	malformed P.IVA 3 different cases: -A standard P.IVA identification with 11 digits but with litterals in it instead of some of the	not passed	in all of the three tested cases the server responded with a “OK” response status and let us register the activty.

			digits -a 3 digit number inserted - a 3 litterals string		
7	[R3] [R4]	Login of an Activity	Base functioning providing correct credentials	passed	
8	[R3] [R4]	Login of an Activity	Empty Credentials	not passed	Once evaded the client side security features, the server let us log in with all of the fields left empty and answered with a “OK” redirecting us to the “Register a building” form. This explicitly contradicts the [R3].
9	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	Base functioning providing correct credentials	passed	
10	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	Empty form	not passed	Once evaded the client side security features, the server responded with a Tomcat default internal server error page. This explicitly contradicts the [R6].
11	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	negative number as “capacity”	partially passed	Server redirected to a page with the writing “ Error, could not insert building” Although an “OK” Status of the response was received.

12	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	closing time set before opening time	not passed	Server let us register the building
13	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	registering a building with the same name of one already registered	partially passed	Server redirected to a page with the writing " Error, could not insert building". Although an "OK" Status of the response was received.
14	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	Empty department form with just one departement	not passed	Once evaded the client side security features, the server responded with a Tomcat default internal server error page.
15	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	Empty department form with more departements: two filled, one empty	not passed	Once evaded the client side security features, the server responded with a Tomcat default internal server error page.
16	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	Filling the "surplus" field of the departement form with a litteral	not passed	Once evaded the client side security features, the server responded with a Tomcat default internal server error page.
17	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	trying to access the "insert a building page" without having made the log in before	passed	The server redirects us to the home page

18	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	inserting two different departments with the exact same surplus field and the exact same name.	partially passed	The server sent us a success message with the identifier of the building we just registered. This let us think that the two undistinguishable departments were successfully added.However, checking in the db, just one departement was added.
19	[R4] [R5] [R6] [R7] [R8] [R9]	Allow an Activity to insert one or more Buildings	inserting the same name of a different building with just some spaces after or before the name	not passed	we could successfully register a new building

Tests 14-15-16 outcome

HTTP Status 500 – Internal Server Error

Type Exception Report

Message For input string: ""

Description The server encountered an unexpected condition that prevented it from fulfilling the request.

Exception

```
java.lang.NumberFormatException: For input string: ""
        java.base/java.lang.NumberFormatException.forInputString(NumberFormatException.java:68)
        java.base/java.lang.Integer.parseInt(Integer.java:662)
        java.base/java.lang.Integer.parseInt(Integer.java:770)
        it.polimi.se2.clup.web.BuildingServlet.doPost(BuildingServlet.java:49)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:652)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:733)
        org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:53)
        org.apache.openejb.server.httpd.EEFilter.doFilter(EEFilter.java:66)
        org.apache.catalina.filters.CorsFilter.handleNonCORS(CorsFilter.java:352)
        org.apache.catalina.filters.CorsFilter.doFilter(CorsFilter.java:171)
```

Note The full stack trace of the root cause is available in the server logs.

Apache Tomcat/9.0.41

Random outcome we got from time to time with no apparent reason while performing tests

HTTP Status 500 – Internal Server Error

Type Exception Report

Message Transaction was rolled back, presumably because setRollbackOnly was called during a synchronization

Description The server encountered an unexpected condition that prevented it from fulfilling the request.

Exception

```
javax.ejb.EJBTransactionRolledbackException: Transaction was rolled back, presumably because setRollbackOnly was called during a synchronizati
        org.apache.openejb.core.ivm.BaseEjbProxyHandler.convertException(BaseEjbProxyHandler.java:422)
        org.apache.openejb.core.ivm.BaseEjbProxyHandler.invoke(BaseEjbProxyHandler.java:353)
        com.sun.proxy.$Proxy114.registerActivity(Unknown Source)
        it.polimi.se2.clup.web.RegisterServlet.doPost(RegisterServlet.java:33)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:652)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:733)
        org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:53)
        org.apache.openejb.server.httpd.EEFilter.doFilter(EEFilter.java:66)
        org.apache.catalina.filters.CorsFilter.handleNonCORS(CorsFilter.java:352)
        org.apache.catalina.filters.CorsFilter.doFilter(CorsFilter.java:171)
```

Root Cause

```
javax.transaction.RollbackException: Unable to commit: transaction marked for rollback
        org.apache.geronimo.transaction.manager.TransactionImpl.commit(TransactionImpl.java:272)
        org.apache.geronimo.transaction.manager.TransactionManagerImpl.commit(TransactionManagerImpl.java:252)
        org.apache.openejb.core.transaction.JtaTransactionPolicy.completeTransaction(JtaTransactionPolicy.java:331)
        org.apache.openejb.core.transaction.TxRequired.commit(TxRequired.java:76)
        org.apache.openejb.core.transaction.EjbTransactionUtil.afterInvoke(EjbTransactionUtil.java:77)
        org.apache.openejb.core.stateless.StatelessContainer._invoke(StatelessContainer.java:271)
        org.apache.openejb.core.stateless.StatelessContainer.invoke(StatelessContainer.java:212)
        org.apache.openejb.core.ivm.EjbObjectProxyHandler.synchronizedBusinessMethod(EjbObjectProxyHandler.java:265)
        org.apache.openejb.core.ivm.EjbObjectProxyHandler.businessMethod(EjbObjectProxyHandler.java:260)
        org.apache.openejb.core.ivm.EjbObjectProxyHandler._invoke(EjbObjectProxyHandler.java:89)
        org.apache.openejb.core.ivm.BaseEjbProxyHandler.invoke(BaseEjbProxyHandler.java:349)
        com.sun.proxy.$Proxy114.registerActivity(Unknown Source)
        it.polimi.se2.clup.web.RegisterServlet.doPost(RegisterServlet.java:33)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:652)
        javax.servlet.http.HttpServlet.service(HttpServlet.java:733)
        org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:53)
        org.apache.openejb.server.httpd.EEFilter.doFilter(EEFilter.java:66)
        org.apache.catalina.filters.CorsFilter.handleNonCORS(CorsFilter.java:352)
        org.apache.catalina.filters.CorsFilter.doFilter(CorsFilter.java:171)
```

Root Cause

```
javax.persistence.PersistenceException: Exception [EclipseLink-4002] (Eclipse Persistence Services - 2.7.7.v20200504-69f2c2b80d): org.eclipse.p
Internal Exception: java.sql.SQLIntegrityConstraintViolationException: Duplicate entry 'nonso' for key 'activity.NAME'
Error Code: 1062
Call: INSERT INTO ACTIVITY (ID, NAME, PIVA, PASSWORD) VALUES (?, ?, ?, ?)
        bind => [4 parameters bound]
Query: InsertObjectQuery(it.polimi.se2.clup.data.entities.Activity@5a5280aa)
        org.eclipse.persistence.internal.jpa.EntityManagerSetupImpl$1.handleException(EntityManagerSetupImpl.java:787)
        org.eclipse.persistence.transaction.AbstractSynchronizationListener.handleException(AbstractSynchronizationListener.java:277)
        org.eclipse.persistence.transaction.AbstractSynchronizationListener.beforeCompletion(AbstractSynchronizationListener.java:172)
        org.eclipse.persistence.transaction.JTASynchronizationListener.beforeCompletion(JTASynchronizationListener.java:70)
        org.apache.geronimo.transaction.manager.TransactionImpl.beforeCompletion(TransactionImpl.java:527)
```

Example of wrong inputs saved in the database

CLOSING	DELTAEXITTIME	LASTEXITTIME	NAME	OPENING	ACTIVITY_ID
21:42:00.000	BLOB	NULL	prodotto1	20:42:00.000	51
00:32:00.000	BLOB	NULL	Building	21:32:00.000	106
18:32:00.000	BLOB	NULL	ciao	21:32:00.000	106
18:32:00.000	BLOB	NULL	pollo	21:32:00.000	106
18:32:00.000	BLOB	NULL	pollo3	21:32:00.000	106
18:32:00.000	BLOB	NULL	pollo34	21:32:00.000	106
18:32:00.000	BLOB	NULL	pollo34s	21:32:00.000	106
22:00:00.000	BLOB	NULL	x	21:59:00.000	110
22:00:00.000	BLOB	NULL	x	21:59:00.000	110
22:00:00.000	BLOB	NULL	x	21:59:00.000	110

ID	NAME	PIVA	PASSWORD
105	nonso	076RT52056C	fcoVlzYNouxztSoQg247jzC7ZltuJiwnD8Bjz+l/O…
106			WJfo6lM5YYE3ln/NwBVy9bHHkEFQ4mrl+/WO…
109	lol	123	vOW66r7u7UxoJwvCdm79u5IPJS7AL3OP6v9L…
110	ale	ale	EeE13+Cnwvy0CM2ohV8wirBvKqKhHPNMlFN…

4. Final remarks

What follows is a summary of our final impressions about the prototype that is the subject of this document.

4.A About the development approach

This acceptance testing was heavily influenced by the fact that the prototype that we tested had basically no user interface. This is a meaningful lack. Due to this lack, we couldn't properly test the user interface nor the underlying backend system. The little we tested cannot be considered sufficient not even for a prototype.

The only thing we can point out, that we evinced by reading the DD, is the fact that the developing team chose to implement the application in a monolithic way, which is a choice we would remark as not really scalable nor reusable. However, as they stated, it adheres to JEE standards.

4.B About the quality of deliverables

Before proceeding and testing the prototype we have been provided, we read the RASD and the DD of the developing team.

The RASD is well structured and well documented. The team really faced several aspects in a very accurate and detailed manner, such as the section B3.
A special regard must be dedicated to the alloy section: very exhaustive and well documented.

About the Design Document: it is nicely developed and remarkably detailed. However, we ran into some minor inconveniences. For example, some unnecessary detailed explanation of standard procedures which could have just been mentioned along with the main reasons which led to the choice of adopting them. Between those, we mention the *Galera Cluster.*
Moreover, the mock ups are little documented.

The code developed by the other team is well written and adequately commented with javadoc.