SAFE STREETS
ACCEPTANCE TEST DELIVERABLE
FOR: ASPESI BATTISTON CARABELLI

Version 1 – 21/01/2020

Authors:

Giulio A. Abbo   10538950

Gianmarco Accordi   10587213

Massimiliano Bonetti   10560496

Professor:

Elisabetta Di Nitto

Academic year:

2019 – 2020

CONTENTS

1 ANALYSED PROJECT DETAILS

The analysed project is available at *https://github.com/EliaBattiston/AspesiBattistonCarabelli*.

The authors of the analysed project are:

- 10522850 Aspesi Andrea (@AndreaAspesiPoli)
  andrea.aspesi@mail.polimi.it
- 10522687 Battiston Elia (@EliaBattiston)
  elia.battiston@mail.polimi.it
- 10518898 Carabelli Alessandro (@AleCarabelli)
  alessandro2.carabelli@mail.polimi.it

We have considered the following documents:

- RASD2
  https://github.com/EliaBattiston/AspesiBattistonCarabelli/blob/master/DeliveryFolder/RASD2.pdf
- DD2
  https://github.com/EliaBattiston/AspesiBattistonCarabelli/blob/master/DeliveryFolder/DD2.pdf
- ITPD1
  https://github.com/EliaBattiston/AspesiBattistonCarabelli/blob/master/DeliveryFolder/ITPD1.pdf

1.A EXTRACTED DATA

Extracted goals from *RASD I.2.3 Goals*:
- [G1] Allow citizens to register providing basic information that certifies their identities and become certified users;
- [G2] Allow authorities to register through a verification procedure;
- [G3] Enable certified users to report new traffic or parking violations;
- [G4] Enable certified users to see their own past reports;
- [G5] Enable certified users to access the map showing the most reported streets.
- [G6] Enable authorities to access all certified users services;
- [G7] Enable authorities to access all report data (including data about the user who created it);
- [G8] Enable authorities to acquire report data in their own systems to generate tickets;
- [G9] Enable authorities to access data analysis results with suggestions;
- [G10] Enable administrators to certify authorities;
- [G11] Enable administrators to block or remove users;
- [G12] Enable administrators to assign roles to users.

From *RASD II.4.2.1 Regulatory policies*, are extracted requirements on the interaction:
- [RP1] The user has to be clearly informed about how its data will be used by SafeStreets, and allow this use before completing the registration process.
- [RP2] SafeStreets has to provide users a way to review and eventually request the deletion of any data about him on the service's databases.

The requirements are taken from the mapping at *RASD III.2.5 Requirements*:
- [R1] Every citizen that wants to register has a valid identification document and a smartphone (with GPS and camera systems).
- [R2] Every citizen that wants to register owns a valid e-mail address.

- [R3] The municipality allows SafeStreets administrator to verify the belonging of the requesting officer to the traffic police (via an automated system or sending an official request).
- [R4] The user has his smartphone with him, is able to login in the application and is not suspended.
- [R5] The user's smartphone has Internet connection enabled and functioning.
- [R6] The user's smartphone has GPS enabled and functioning.
- [R7] The user is able to take at least one picture of the violation to correctly choose the type of violation he is reporting.
- [R8] The user has already made at least one report.
- [R9] The officer access has the same base functions of the basic user access, plus specific ones.
- [R10] The officer has a computer with Internet access and is able to login to the system.
- [R11] The municipality and/or the authorities use the provided APIs.
- [R12] The administrator has a computer with Internet access and is able to login to the system.
- [R13] The administrator has the possibility to use an official communication channel with the municipality.
- [R14] The administrator has received a block/removal request from the authorities.
- [R15] The administrator has received a sign-up request from the authorities and/or from the municipality.

From *RASD III.3 Performance requirements:*
- [PR1] There are no limits to the number of simultaneously connected users;
- [PR2] Latency must be stable for different amount of users load and under a reasonable time;
- [PR3] The app won't make blocking requests such as for the upload of a picture, these kind of requests needs to run in background;
- [PR4] The AI algorithm responsible for the extraction of data from pictures has to run on devices only if dedicated hardware is provided; otherwise it will be run on SafeStreet's servers.

Software system attributes:
- [SA1] availability and reliability, SafeStreets should provide a guarantee of 99+% uptime for its services;
- [SA2] security, use of the protocol HTTPS and hash of the password on the client;
- [SA3] maintainability, using consolidated and well documented technologies such as Xamarin Framework;
- [SA4] portability, possibility to use the application on Android and on iOS and on a web browser;

1.B PARTIALLY DEVELOPED FEATURES
- [G11] Enable administrators to block or remove users; it is possible only to block a user not to remove him.
- [SA2] The deployment of the official site on Altervista makes use of the HTTPS protocol through e certificate of Altervista, but the hash of the password is calculated on the server, instead that calculate the hash on the client.

1.C NOT DEVELOPED FEATURES
- [G9] Enable authorities to access data analysis results with suggestions;

- [RP1] The user has to be clearly informed about how its data will be used by SafeStreets, and allow this use before completing the registration process.
- [RP2] SafeStreets has to provide users a way to review and eventually request the deletion of any data about him on the service's databases.
- [PR4] The AI algorithm has not been implemented.

2 INSTALLATION SETUP

In this section we describe what software we have installed and how we have installed it. Furthermore, we describe some issues that we have found and some incoherencies of the implementation with the provided documentation, the RASD, the DD and the ITD.

2.A HOW IT WAS INSTALLED

The official deployed system, that can be found at safestreets.altervista.org, works correctly and the only thing that has to be install is the Application on the mobile device, but this method offers a few ways of testing the system.

Instead, the installation of a local server was a bit more complicated: first of all we had to install and configure an Apache Server, then also the PHP compiler to execute the code, and for the testing we also needed to install the suit PHPUnit. To perform debug, we have also installed and configured XDebug. To work more easily with the code, we have used PHPStorm as IDE for PHP.

We have installed the Database using MySql and we have imported the given .sql file inside the database without problems.

To run the application on an iOS mobile device we have installed Xcode on macOS, we have registered an iCloud account as developer and we have created a blank project in order to create a Provisioning Profile for the application. Then we have set the Signing Identity and the Provisioning Profile in Project Options/Build of Microsoft Visual Studio.

To run the application for android in debug mode from Windows, we have installed Visual Studio, configured the debug, and then we have tested the application from the device.

2.B ISSUES

The installation of the Database was easy, except for the fact that the library mysqli of PHP used an older method of authentication that the one used by MySql, so we had to create a different user that authenticated with a legacy method.

The installation instruction also doesn't specify how to change the client's code in order to make request to the local server instead of the request going to the official deployed system on the Altervista server.

It is not sufficient to have Microsoft Visual Studio installed to run the application on an iOS mobile device. As described in the section 2.A we have created a Provisioning Profile to give it to Microsoft Visual Studio.

No information is present on how to run the code and the tests of the application server, for example by using PHPStorm.

3   ACCEPTANCE TEST CASES

In this chapter the performed tests are presented with their motivations and the corresponding outcomes; test cases are extracted from the RASD and the other documents, keeping in count the features that were actually implemented. The motivation for tests on requirements is omitted.

3.A   APPLICATION SERVER

The tests on the application server can be found at:

https://github.com/gianfi12/AbboAccordiBonetti/tree/master/DeliveryFolder/testingForATD

Name	**Correct Sign up**
Requirements/Goals involved	G1
Input	Correct parameters, the username was not already present.
Output	Correct registration
Expected output	Correct registration

Name	**Sign up with an already present username**
Requirements/Goals involved	G1
Input	Correct parameters, but the username was already present.
Output	Response code 405
Expected output	Response code 405

Name	**Login with a wrong username**
Requirements/Goals involved	Login
Input	Login with a username that was not registered
Output	Response code 401
Expected output	Response code 401

Name	**Login with a registered username and a correct password**
Requirements/Goals involved	Login
Input	Login with a registered username and a correct password
Output	Correct login
Expected output	Correct login

Name	**Login with a registered username but wrong password**
Requirements/Goals involved	Login
Input	Login with a registered username but wrong password
Output	Response code 401
Expected output	Response code 401

Name	**Login from a not yet accepted user**

Requirements/Goals involved	Login
Input	Login with a not yet accepted username and a correct password
Output	Response code 403
Expected output	Response code 403

Name	**Login from a suspended user**
Requirements/Goals involved	Login
Input	Login with a suspended username and a correct password
Output	Response code 402
Expected output	Response code 402

Name	**See past Reports**
Requirements/Goals involved	R8
Input	Try to see past report if the user hasn't made previous report
Output	201 Report list is empty
Expected output	200

Name	**Officer make a report**
Requirements/Goals involved	R9
Input	Try to see if also the municipality can make a report
Output	200
Expected output	200

3.B WEB APPLICATION

Name	**Access to all reports from authority**
Requirements/Goals involved	G7
Input	On the web browser with the officer account we have selected the function Reports.
Output	View of all reports.
Expected output	View of all reports.

Name	**User acceptance**
Requirements/Goals involved	G10
Input	On the web browser with the administrator account we have accepted an officer account.
Output	The officer was accepted.
Expected output	The officer was accepted.

Name	**User block**

Requirements/Goals involved	G11
Input	On the web browser with the administrator account we have suspended a user.
Output	The user was suspended, he can no more do the login.
Expected output	The user was suspended, he can no more do the login.

Name	**Change the role of the user**
Requirements/Goals involved	G12
Input	On the web browser with the administrator account we have change the role of a user.
Output	The user has the new role.
Expected output	The user has the new role.

Name	**The administrator has received a sign-up request from the authorities and/or from the municipality**
Requirements/Goals involved	R15
Input	The administrator uses the functionality User Acceptance
Output	The administrator can view the users that have requested a sign up
Expected output	The administrator can view the users that have requested a sign up

Name	**Use of the protocol HTTPS**
Requirements/Goals involved	SA2
Input	Access to the web application with a web browser by writing manually "https://"
Output	The browser indicates that the site is secure
Expected output	The browser indicates that the site is secure

3.C MOBILE APPLICATION

Name	**Sign up**
Requirements/Goals involved	PR3, G1
Input	Sign up from the mobile application on Android and on iOS
Output	Waiting time during the confirmation of the registration, so the it was not made in background.
Expected output	The app won't make blocking requests such as for the upload of a picture, these kind of requests needs to run in background.

Name	**Registration of an authority**
Requirements/Goals involved	G2
Input	Registration of a user, acceptance by the administrator and change role to officer done by the administrator
Output	New account as authority (officer role)

Expected output	New account as authority (officer role)

Name	**New report**
Requirements/Goals involved	PR3, G3
Input	New report from the mobile application on Android and on iOS
Output	Waiting time to upload the report, so the it was not made in background. Then the report was sent.
Expected output	The app won't make blocking requests such as for the upload of a picture, these kind of requests needs to run in background. The report is sent.

Name	**Enable certified users to see their own past reports**
Requirements/Goals involved	G4
Input	On the mobile application with a regular account we have selected the Past Reports function.
Output	View of the own past reports.
Expected output	View of the own past reports.

Name	**Street safety on Android**
Requirements/Goals involved	G5
Input	Use of the function Street safety from the mobile application on Android
Output	Map with the streets with most reports
Expected output	Map with the streets with most reports

Name	**Street safety on iOS**
Requirements/Goals involved	G5
Input	Use of the function Street safety from the mobile application on iOS
Output	Blank page
Expected output	Map with the streets with most reports

Name	**The authority can access to all the services of the certified users**
Requirements/Goals involved	G6, R9
Input	On the mobile application with an officer account we try to access all the services.
Output	Access to all the services of the certified users.
Expected output	Access to all the services of the certified users.

Name	**Take at least one picture for the report**
Requirements/Goals involved	R7
Input	Try to send the report with a picture and the other information
Output	Report sent

Expected output	Report sent

Name	**Take at least one picture for the report**
Requirements/Goals involved	R7
Input	Try to send the report without a picture
Output	Alert, report not sent
Expected output	Alert, report not sent

3.D PERFORMANCE

We have analysed the performance servers with Apache JMeter Tool.

Name	**Login with 10000 users**
Requirements/Goals involved	SA1
Input	Login with 10000 users
Output	Correct login
Expected output	Correct login
Performance	The server has served 663 requests per minute. Average response time: 2,4 s.

Name	**Login with 1000 users**
Requirements/Goals involved	SA1
Input	Login with 1000 users
Output	Correct login
Expected output	Correct login

Performance during the login with 1000 users:

Label	# Samples	Average	Median	90% Line	95% Line
HTTP Request	2000	2463	7	7947	8304

99% Line	Min	Max	Error %	Throughput	KB/sec
15449	1	15503	0.00%	112.5/sec	51.3

JMeter has done 2 logins for each user so in total 2000 samples.

Average: 2463 ms for 2 logins.

Median: time in the middle of a set of samples result

90% Line: 90th percentile

95% Line: 95th percentile

99% Line: 99th percentile

Min: shortest time taken by a sample for specific label

Max: shortest time taken by a sample for specific label

Error %: Percentage of Failed requests per Label.

Throughput: number of requests that are processed per time unit by the server

KB/sec: amount of data downloaded from server during the performance test execution. In short, it is the Throughput measured in Kilobytes per second.

4 ADDITIONAL COMMENTS

The tests in the ITD are well described with the input, the output and the description. Also the tests are completely automated through the use of the GitHub's CI/CD platform.

The source code on the client and on the server is not commented. There are only some rare comments in English and in Italian.

In the Design Document is said that the system will rely on the services of Amazon, but in the ITD and in the implementation the system relies on the services of Altervista.

The requirements (apart from the requirements R7, R9, and R15) in the RASD are not prescriptive assertions formulated in terms of the shared phenomena and they don't represent a bridge between the World and the System. They have been formulated as assertions assumed to be hold in the world, and as so we cannot test them in a precise way.