POLITECNICO
MILANO 1863

Software Engineering 2
CLup project by
Robert Medvedec
Toma Sikora

Acceptance Test Deliverable

Deliverable:	ATD
Title:	Acceptance Test Deliverable
Authors:	Robet Medvedec, Toma Sikora
Version:	1.0
Date:	11-February-2021
Download page:	https://github.com/robertodavinci/Software_Engineering_2_Project_Medvedec_Sikora/Implementation/Clup
Copyright:	Copyright © 2021, R. Medvedec, T. Sikora – All rights reserved

Analysed project by:	Samuele Negrini, Giorgio Piazza
Analysed repository:	https://github.com/snegrini/NegriniPiazza/tree/main/DeliveryFolder
Reference documents:	RASD1.2.pdf, DD1.2.pdf, ITD1.0.pdf
Reference applications:	CustomerApp.apk, StoreApp.apk

Contents

List of Figures

List of Tables

1 Introduction

This document is a testing and acceptance document of the CLup project developed by Samuele Negrini and Giorgio Piazza.

This document focuses on all three reference documents created by the group, while the main focus is on the app itself along with ITD1.0.pdf document that follows the app and explains some of its design choices.

In the second chapter of the document, the process of installation of the app has been described, along with any potential difficulties regarding it (there weren't any).

In the third chapter the testing done by our team has been described and some of the test results in terms of both desktop and mobile screenshots have been presented. Several issues have been found, but the apps themselves work really well and no major problems were encountered while using and testing the app.

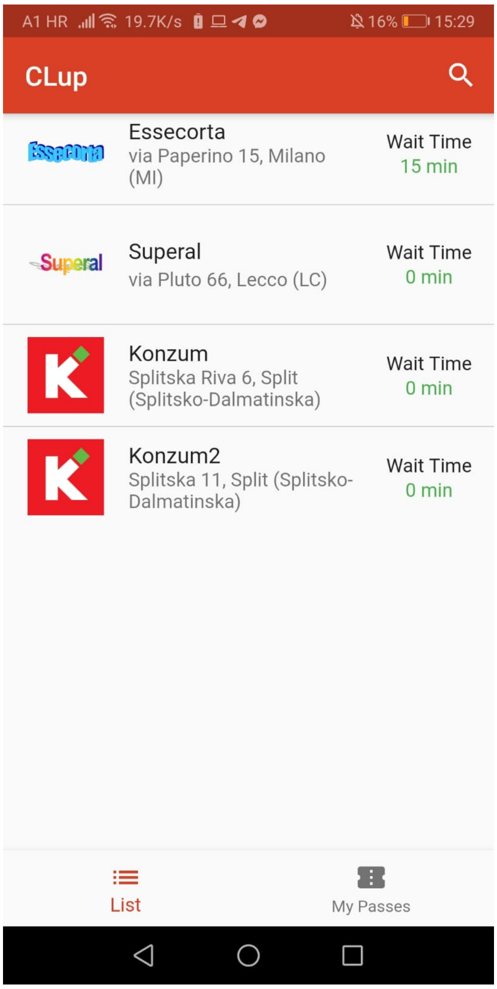
In the fourth and final chapter, we've added some additional comments as well as noted which things are not in line with the rest of the documentation and which things were supposed to be done differently. Finally, our overall experience with the whole system is also described.

2 Installation Process

Installation instructions required installing the whole MySQL environment which includes MySQL Server and Workbench, Java JDK, TomEE, and Maven. The installation instructions were very detailed and easy to follow.

Although this installation process is not the most elegant and it takes a while to set it all up, we were able to complete it in a reasonable amount of time. This process would usually not have to be repeated by a user of the app since the whole backend would be uploaded to the server. The creators did that as well, so we've done our testing on both our computers with their databases set up and over the servers with connection to their computers. Both cases gave the same results.

The app installation for the Android device went without any issues, was pretty straightforward and worked on multiple devices that we have tested it on (Samsung Galaxy S6, Huawei P Smart, Samsung Galaxy Note 9) which all have different Android OS versions; 7, 8, and 9, respectively.

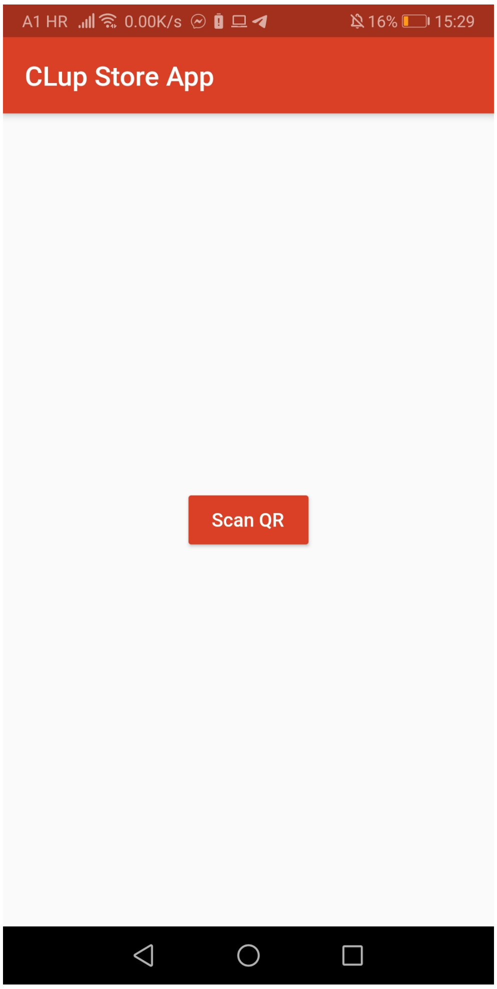


A1 HR 19.7K/s 16% 15:29

CLup

Essecorta	Essecorta via Paperino 15, Milano (MI)	Wait Time 15 min
Superal	Superal via Pluto 66, Lecco (LC)	Wait Time 0 min
K	Konzum Splitska Riva 6, Split (Splitsko-Dalmatinska)	Wait Time 0 min
K	Konzum2 Splitska 11, Split (Splitsko-Dalmatinska)	Wait Time 0 min

Figure 1: **CLup app running on Android 8**

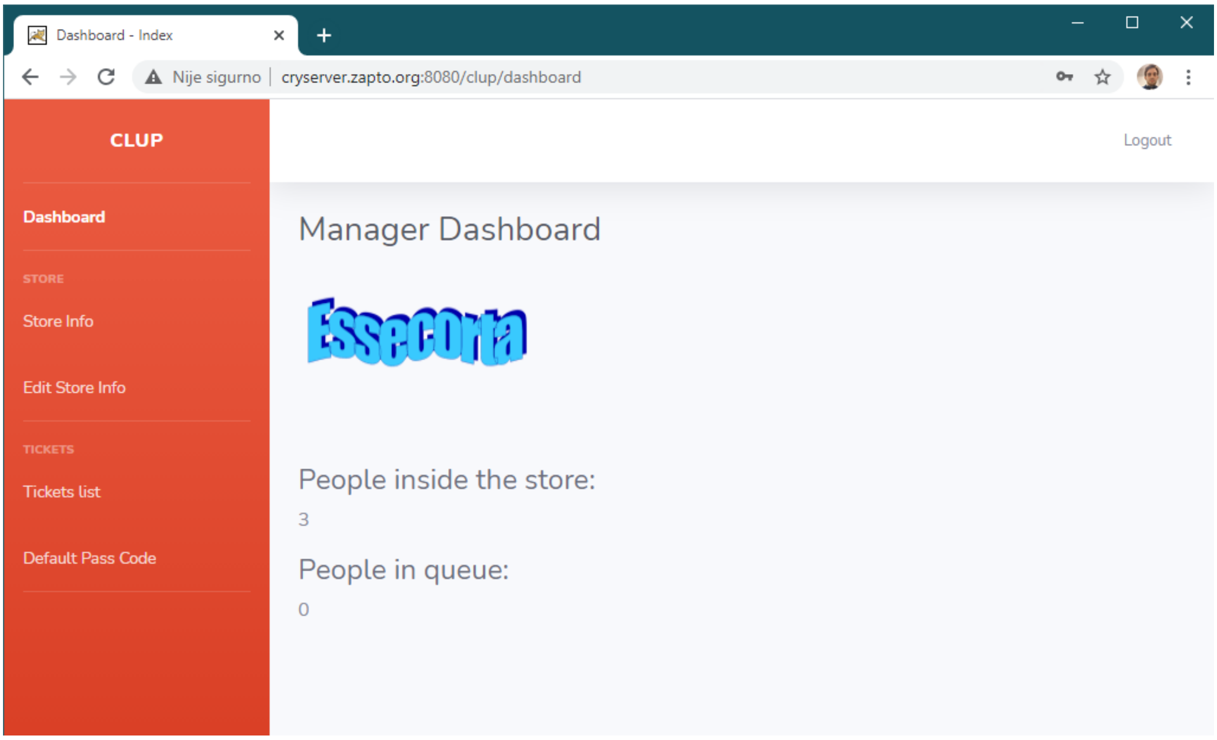


A1 HR 0.00K/s 16% 15:29

CLup Store App

Figure 2: **CLup employee app running on Android 8**

The desktop (browser) version of the app worked in all tested browsers (Firefox, Chrome, Brave) and was easy to log into and use.

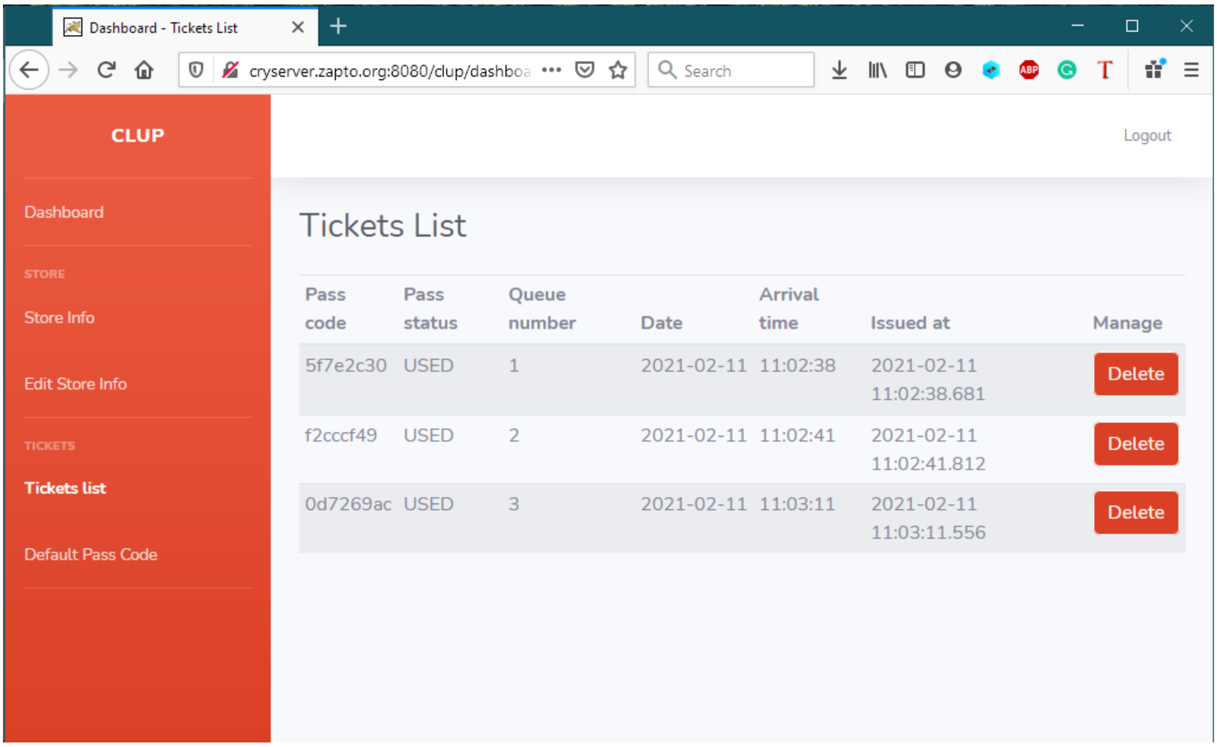


Manager Dashboard

People inside the store:
3

People in queue:
0

Figure 3: **CLup desktop (browser) app running on Google Chrome**



Tickets List

Pass code	Pass status	Queue number	Date	Arrival time	Issued at	Manage
5f7e2c30	USED	1	2021-02-11	11:02:38	2021-02-11 11:02:38.681	Delete
f2cccf49	USED	2	2021-02-11	11:02:41	2021-02-11 11:02:41.812	Delete
0d7269ac	USED	3	2021-02-11	11:03:11	2021-02-11 11:03:11.556	Delete

Figure 4: **CLup desktop (browser) app running on Mozilla Firefox**

3 Acceptance Testing

3.1 Implemented requirements

The list of requirements and their implementation status is included in ITD file. Testing has been done on these implementations and the comments on whether they are correctly implemented or not are next to the requirement name in the following list:

R.1 The system shall allow customers to line-up remotely in a store queue.
Implementation status: Implemented
Comments: The line-up implementation doesn't really exist, queueing is not managed by the time the ticket has been requested or the ticket number, but rather every ticket scan is accepted, making the store employee in charge for the decision on who can enter the store and not the system. The virtual line-up exists, but it's not ordered in any way, making the whole point of the line-up non existent.

R.2 The system shall generate a new ticket when a customer enters a queue
Implementation status: Implemented
Comments: Implementation tested and working correctly.

R.3 The system shall allow customers which do not have a smartphone to get a ticket in place.
Implementation status: Not implemented
Comments: /

R.4 The system shall allow customers to view the number of people lined up in a queue
Implementation status: Implemented
Comments: Implementation tested and working correctly.

R.5 The system shall give customers an estimated waiting time.
Implementation status:Partially implemented
Comments: Time estimation is equal to 1 person per 15 minutes. The estimation doesn't really work correctly since there is no store queueing and no way or determining how many people are in front, making estimation always be either 0 or 15 minutes.

R.6 The system shall fetch the GPS position while the user has retrieved a store pass.
Implementation status:Not implemented
Comments: /

R.7 The system shall allow customers to leave a queue.
Implementation status:Implemented
Comments: Customer can delete the ticket anytime, therefore leaving the queue. Implementation tested and working correctly.

R.8 **The system shall allow customers to filter stores by name.**
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.9 **The system shall notify customers when it's time to leave for the store.**
Implementation status:Not implemented
Comments: /

R.10 **The system shall allow customers to book-a-visit to the store and send them the confir-mation link and receipt via email.**
Implementation status:Not implemented
Comments: /

R.11 **The system shall allow book-a-visit customers to specify the main categories of itemthey intend to buy.**
Implementation status:Not implemented
Comments: /

R.12 **The system shall allow customers to delete a store pass.**
Implementation status:Partially implemented
Comments: There are no bookings because "Book a visit" feature doesn't exist. Ticket deletion working correctly.

R.13 **The system shall notify customers when a ticket or booked visit is deleted.**
Implementation status:Not implemented
Comments: /

R.14 **The system shall accept bookings based onto the already booked category items.**
Implementation status:Not implemented
Comments: /

R.15 **The system shall allow a registered store manager to login by using their credentials.**
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.16 **The system shall allow store managers to view the current status of people inside the store.**
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.17 **The system shall allow store managers to view the current status of people in the queue.**
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.18 **The system shall allow store managers to view the booked visits to the store.**
Implementation status:Not implemented
Comments: /

R.19 The system shall allow store managers to set a maximum cap of people inside the store.
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.20 The system shall allow store managers to delete tickets and booked visits.
Implementation status:Partially implemented
Comments: There are no bookings because "Book a visit" feature doesn't exist. Ticket deletion working correctly.

R.21 The system shall allow a registered store employee to login by using their credentials.
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.22 The system shall allow store employee to view the current status of people inside the store.
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.23 The system shall allow store employee to view the current status of people in the queue.
Implementation status:Implemented
Comments: Implementation tested and working correctly.

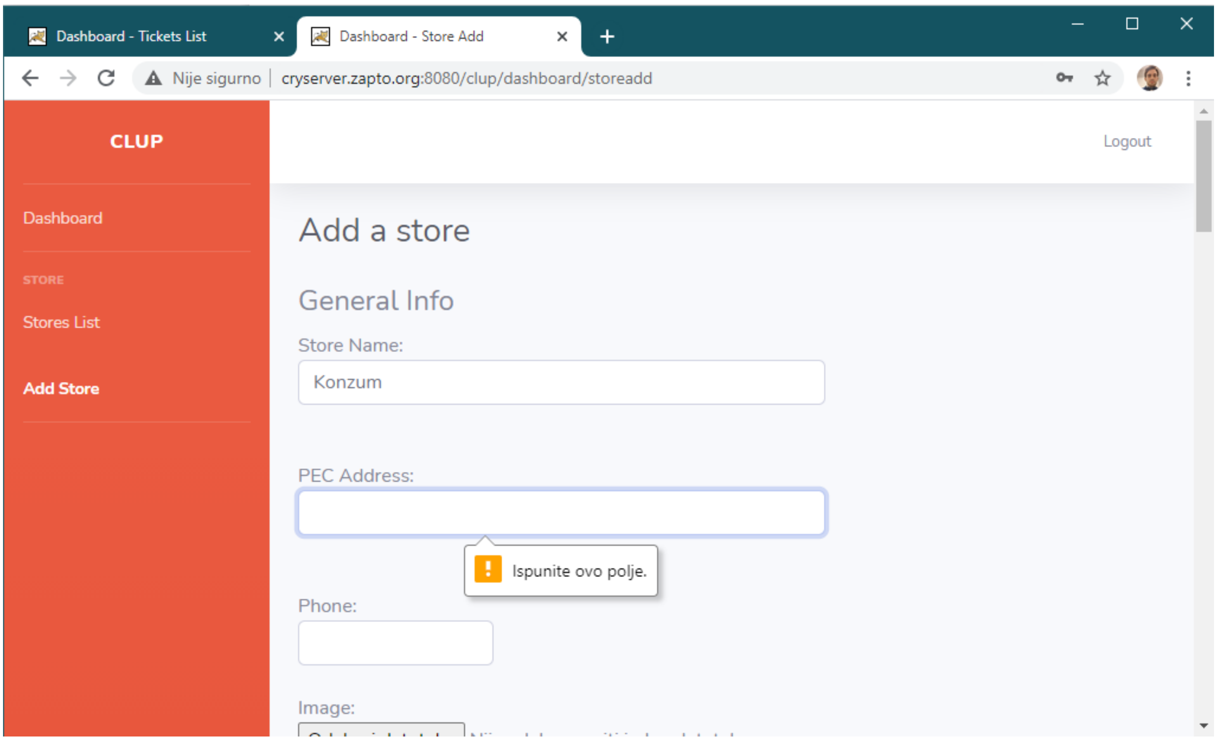
R.24 The system shall allow store employee to scan QR codes.
Implementation status:Implemented
Comments: Implementation tested and working correctly.

R.25 The system shall allow store employee to validate store passes.
Implementation status:Partially implemented
Comments: there are no bookings because "Book a visit" feature doesn't exist. Ticket scanning working correctly.

R.26 The system shall allow CLup admins to register new supermarkets.
Implementation status:Implemented
Comments: Implementation tested and working correctly. New supermarkets can be registered with name, full address, image, and work hours. No two stores of the same name and faulty work hours can be registered.

R.27 The system shall generate new manager and staff credentials for each supermarket registered.
Implementation status:Implemented
Comments: Implementation tested and working correctly.

We have also replicated (manually) all the Unit and System tests defined in ITD1.0.pdf file under chapters 5.1 and 5.3. We have gotten the same results as the creators of the system. Here are a couple of examples and error messages received when trying different "bad" actions.



Add a store

General Info

Store Name: Konzum

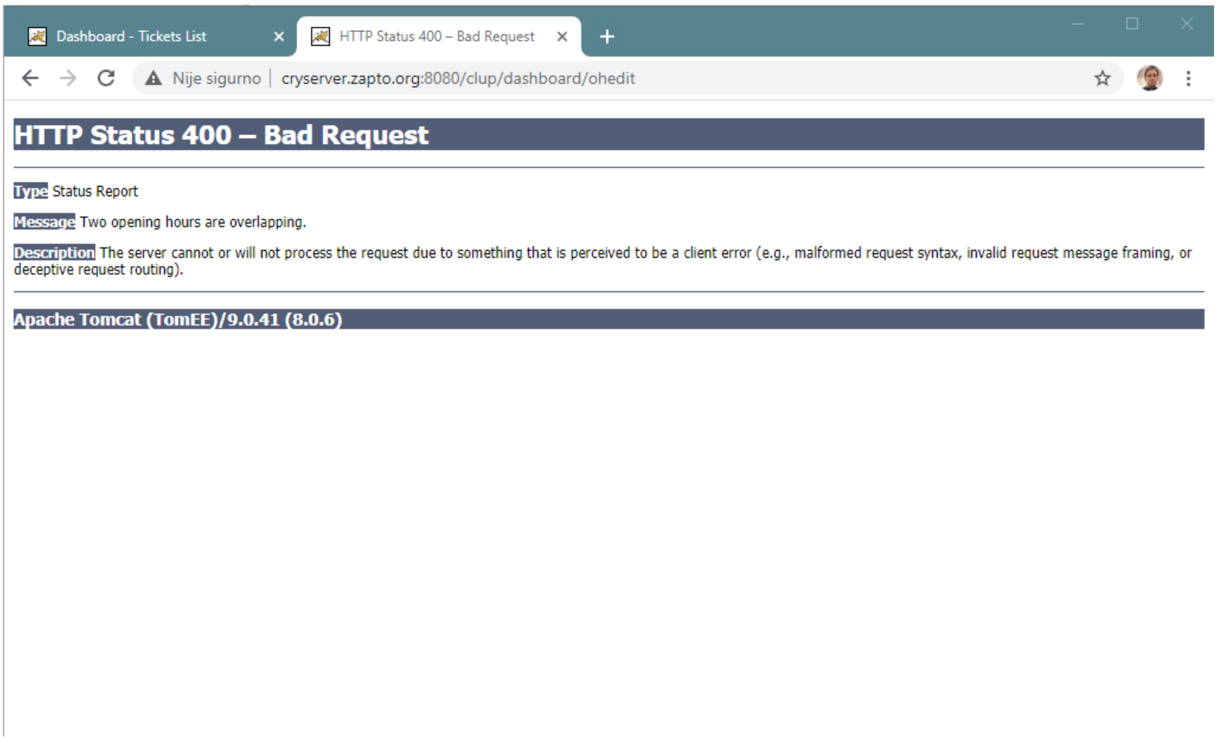
PEC Address:

Ispunite ovo polje.

Phone:

Image:

Figure 5: **Left empty field when creating a store**


Dashboard - Tickets List
HTTP Status 400 – Bad Request
Nije sigurno
cryserver.zapto.org:8080/clup/dashboard/ohedit

HTTP Status 400 – Bad Request

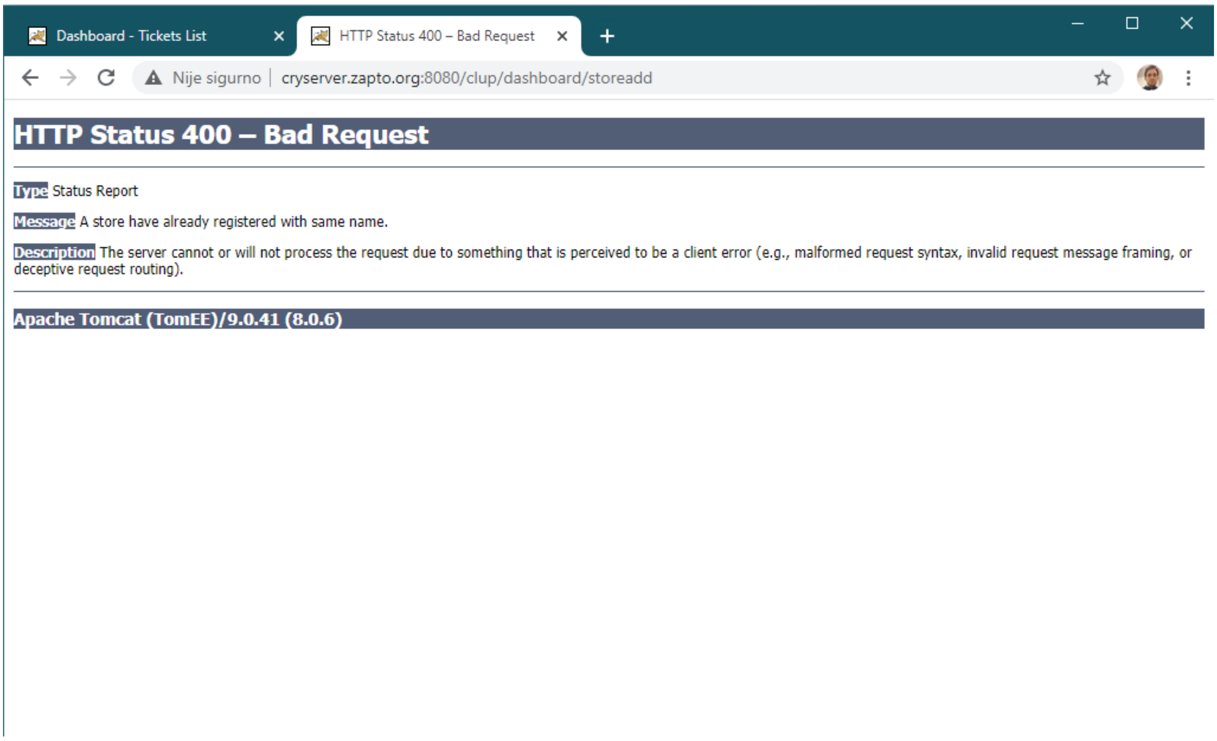
Type Status Report

Message Two opening hours are overlapping.

Description The server cannot or will not process the request due to something that is perceived to be a client error (e.g., malformed request syntax, invalid request message framing, or deceptive request routing).

Apache Tomcat (TomEE)/9.0.41 (8.0.6)

Figure 6: **Bad work hours written - two overlapping intervals**


Dashboard - Tickets List
HTTP Status 400 – Bad Request
Nije sigurno | cryserver.zapto.org:8080/clup/dashboard/storeadd

HTTP Status 400 – Bad Request

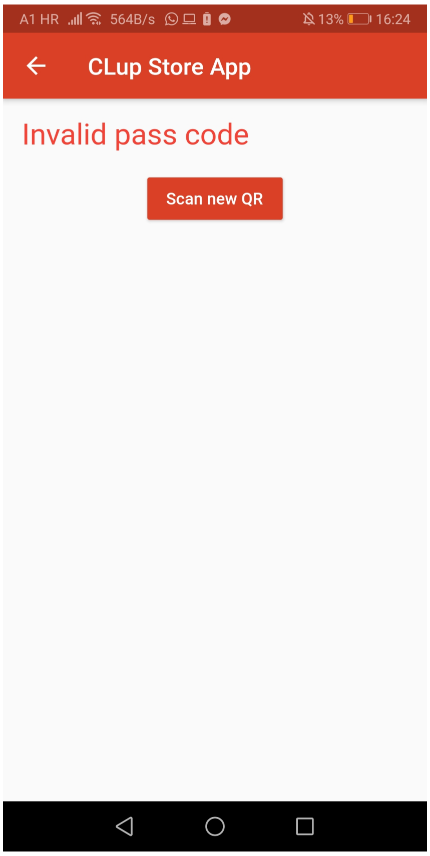
Type Status Report

Message A store have already registered with same name.

Description The server cannot or will not process the request due to something that is perceived to be a client error (e.g., malformed request syntax, invalid request message framing, or deceptive request routing).

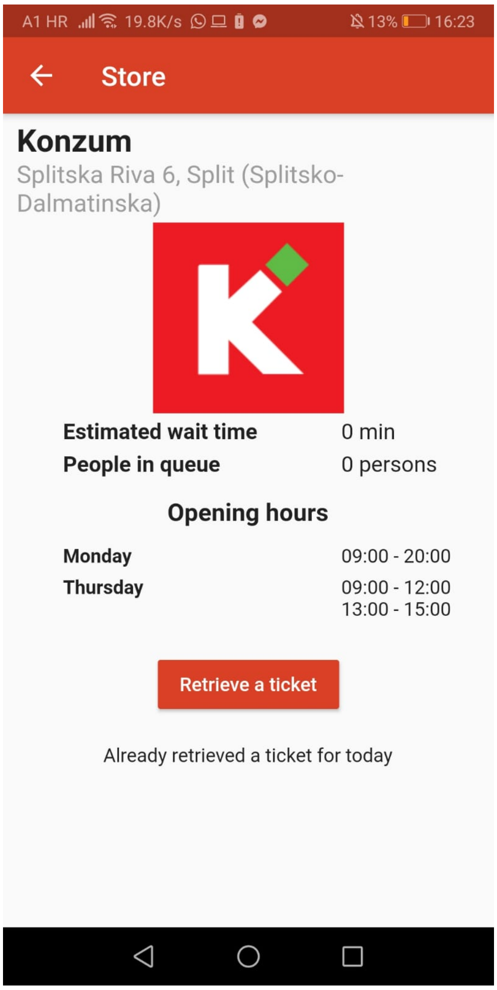
Apache Tomcat (TomEE)/9.0.41 (8.0.6)

Figure 7: **Two stores of the same name error**



Invalid pass code

Figure 8: **Invalid QR code scan error**



A1 HR 19.8K/s 13% 16:23

Store

Konzum

Splitska Riva 6, Split (Splitsko-Dalmatinska)

Estimated wait time 0 min

People in queue 0 persons

Opening hours

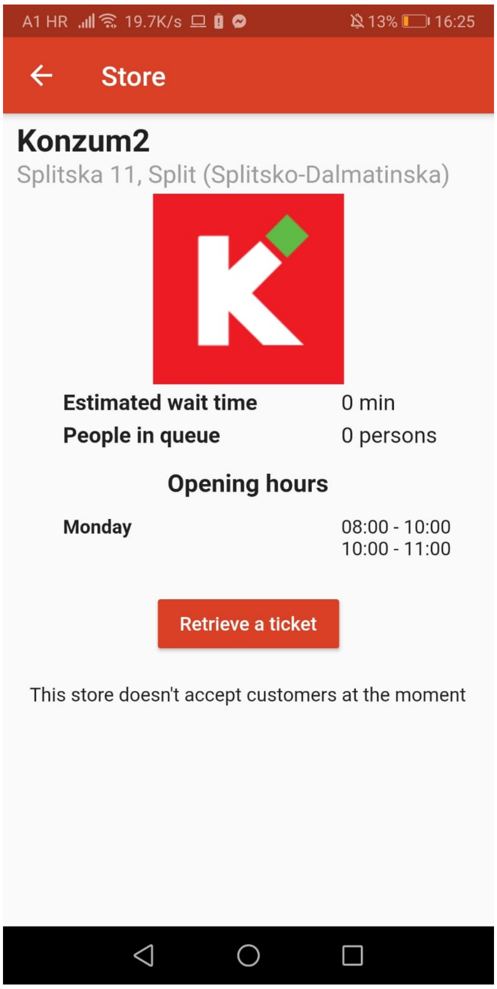
Monday 09:00 - 20:00

Thursday 09:00 - 12:00
13:00 - 15:00

Retrieve a ticket

Already retrieved a ticket for today

Figure 9: **Ticket already retrieved for the same store that day (and not deleted yet)**



A1 HR 19.7K/s 13% 16:25

Store

Konzum2

Splitska 11, Split (Splitsko-Dalmatinska)

Estimated wait time 0 min

People in queue 0 persons

Opening hours

Monday 08:00 - 10:00
10:00 - 11:00

Retrieve a ticket

This store doesn't accept customers at the moment

Figure 10: **The store is currently closed error**

4 Project Comments

The implementation of the desired functionalities of the project described in the RASD and DD meets most of the set goals.

Overall, the implementation for both the mobile and web client looks and feels professional. Clean, minimalistic, and simple to use, it enables customers to switch to a virtual lineup instead of a physical one without any major complications other than installing it on their phone.

Additionally, the store managers and the employees should have no problems whatsoever with the use of the system, as for them, it is equally as clear. The overall experience really is as easy as possible (**G.3 - Grant customers an overall experience as easy as possible**).

Moreover, the system does indeed allow the supermarkets to better monitor and limit the access to the store (**G.5 - Allow supermarkets to monitor access to stores in a better way** and **G.7 - Allow supermarkets to limit the number of access to stores**), and precisely know how many people are coming to the store in advance(**G.6 - Allow supermarkets to know in advance how many people are coming to stores**). The current implementation is also ready for simple upgrades that would allow even the customers who do not have access to the technology to physically retrieve a ticket at the store(**G.4 - Allow customers, even the ones who don't have access to technology, to enjoy the service**). The addition of new functionalities such as waiting time estimation through Google Maps integration and "Book a visit" feature should entail no serious rework of the database and the backend.

It should, however, be noted that the implementation does not in its current form meet all the goals set in RASD. The most noteworthy cause of that is the lack of queue management logic. While the store employee can with the current version of the system easily check if the store is currently at capacity and check if the person has a valid ticket ("valid" meaning just that it has not yet been used), it is up to him to decide whether to let the new customer in (the system allows new customers to enter even when the store is full). The system offers no support with the influx control.

Furthermore, the only way a customer can estimate how much time is left until his turn is to go to the store and check in person. That almost completely beats the purpose of virtual queueing itself, as this way the main difference between a physical number printing machine and the application is the number retrieval. Although that by itself does reduce physical contact considerably (**G1.3 - Shorten the amount of time a customer is in queue**), it does not use the full potential of a well-managed virtual queue. This means that frequent checkups at the store at or near full capacity can result in hazardous situations(**G1.1 - Allow customers to avoid the creation of hazardous situations**) and the customer still has to spend considerable time in front of the store (**G.2 - Grant customers an overall experience as easy as possible**). Another thing to notice is that the store employee who is scanning the tickets has to have two devices near him at all times - on scanning device (mobile phone) and one monitoring device (a desktop) which shows him how many people are currently in the store and in queue. If the employee hasn't got a desktop near him, there is no way of telling how many places are still free inside the store, since the mobile app doesn't show that information on the screen, which is a small detail but one that could be very problematic for some stores.

In the end, it is important to note that all mentioned drawbacks stem from a small set of design choices on the backend and as such should be easily changeable and/or upgradeable. The database, the frontend, and most of the backend work as intended and require no changes whatsoever.

Overall, the system works robustly and coherently, and is ready for professional deployment, with minimal changes.