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

CLup - Customer Line-up

ATD
Acceptance Test Deliverable

Andrea Franchini(10560276)
Ian Di Dio Lavore (10580652)
Luigi Fusco(10601210)

12-02-2021

Contents

1 Introduction

1.1 Analyzed Project

`https://github.com/emassaro98/DigregorioMassaroTamma`

- Digregorio Gabriele (10802375) | gabriele180698
- Massaro Enrico (10602730) | emassaro98
- Tamma Vanessa (10793785) | vanessatamma

1.2 Acronyms

- **RASD:** Requirement Analysis and Specification Document
- **DD:** Design Document
- **ITD:** Implementation and Test Derivable
- **APK:** Android Application Package

1.3 Reference Documents

- I&T assignment goal, schedule, and rules
- Software Engineering 2 course slides
- RASD - Di Gregorio, Massaro, Tamma
- DD - Di Gregorio, Massaro, Tamma
- ITD - Di Gregorio, Massaro, Tamma

2 Installation Setup

The provided installation setup consisted of the following steps:

- Download the APK from the Delivery Folder inside of the repository
- Install the APK on an Android phone
- Run the application with success

Overall the installation process was very easy to follow as the service was already deployed in an external environment.

Note: There is no real way to check that the code executing on the server side is the same one present on the repository.

3 Acceptance

We performed testing on the application in order to verify the correct implementation of the requirement described in the ITD.

- **R1:** In the "Home" page the QR code relative to a single reservation is available through a dedicated button.
- **R2:** In the "Book" page there is a "Booking" button for each available store. Once clicked the user can then choose a date, a duration, and a time slot among the available ones. Once the "BOOK" button is clicked the reservation is available in the "Home" page.

- **R3:** In the "Book" page there is a "Lineup" button for each available store. Once clicked the user can choose a duration. Once the "BOOK" button is clicked the ticket is available in the "Home" page.
- **R4:** While having a booking for a store active, when clicking the "BOOK" button after having selected the preferred date, duration, and time slot, an error message is shown blocking the progress.
- **R9:** Both in the booking and the queue-up page the user is asked to insert their expected duration. If the duration is not selected the "BOOK" button is inactive.
- **R13:** The "Book" page shows a list of stores.
- **R14:** When first opened the application asks the user to login with email and password, and shows an option to create an account. In the account creation page the user is asked for their full name, email, and password. Once the account is created the credentials can be successfully used to log into the application, and a label containing the initials of the user is shown in the upper right corner in the "Home" and "History" pages.
- **R16:** The application requires a registration or login in order to access the main functionalities.
- **R17:** In the "Home" page the current reservations are shown.
- **R18:** The customer reservation is accessible in the section of the application.
- **R19:** In the "Home" page, below every reservation, a "DELETE" button is shown. When pressed the reservation disappears.
- **R20:** An error is displayed if the user tries to line up when already has a booking, or vice versa.

All requirements were adequately implemented and passed our tests. However, we believe that the functionalities implemented are limited:

- We would have appreciated a logout function. It would have made the testing process a little bit faster since in order to log out you have to uninstall the application from the device and reinstall it.
- There is no real way to check if tickets and reservations are actually created, as there is *no functionality related to their usage*.

4 Additional Points

- No instructions present in the ITD on how to deploy the application server on our own machines to do white-box testing.
- Lots of leftover files in the application server, like the `docker-compose.yml` file, are not mentioned in the DD or ITD.
- Incoherence between the DD, ITD and the actual software developed. In fact *no Unit testing or Integration testing were performed*. Even in this case no way to perform ourself those tests.
  The only form of testing that is present in the prototype is System testing through an external system (Postman) to test the responses of the `Rest API` endpoints. The use of Postman must have been done only at the end of the development phase, as it needs the whole system to be functioning. This is in contrast with the methodologies presented in the DD and ITD documents.
- Code structure in the ITD document doesn't have a relative path to the GitHub repository, which makes locating files a little difficult.