university of groningen

faculty of mathematics and natural sciences

Advanced Software Architecture

Train ticketing system

Aida Baxhaku (3242528)

Petar Hariskov (3226735)

Alexandra Matreata (3179265)

Eric Rwemigabo (3040356)

January 10, 2017

version 1.0.0

Revision history

Contents

List of figures

List of tables

Glossary

Chapter 1

Introduction and goals

Chapter 2

Architecture constraints

Chapter 3

System scope and context

This section outlines the main relations between the system and its environment, external systems or entities with which it interacts. The business and technical contexts in which the system will perform as well as different inter-component communication solutions will be presented in this chapter.

3.1 Business context

The system focuses on simplifying the management of information related to train traveling and ticket payment for its users. The system will provide an account for each user which will store information regarding travel distance, destinations, payments, etc. This will help both users travelling by train and the companies providing the travelling services by keeping track of all these components in a centralized manner.

The system will present the user with the choice of making an automated payment for each trip through the connection between the mobile app and the beacon in the train cart. Alternatively, an external payment system will be presented to the user I every train station where they can log in and perform the transaction.

A second important target for our system will be represented by companies providing travelling services by train, which may include governmental institutions or different private companies.

3.2 Logical view

The following diagram displays the main use cases of the system which will be further described in use case scenarios:

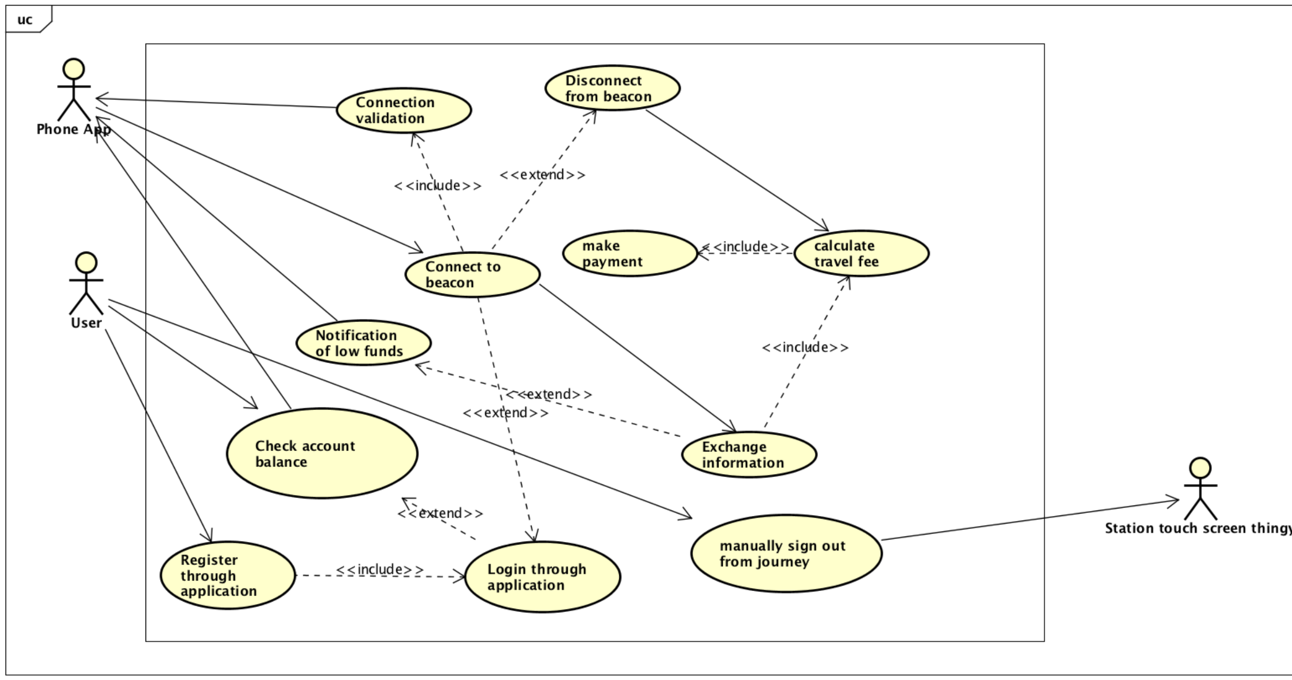

Figure 3.1: Main use cases

Usecase	Connect phone to beacon
Main actor	Phone app
Goal	Connect to the beacon
Preconditions	• Bluetooth is switched on • App is installed • Train has left the station
Main scenario	• The beacon discovers the phone • The phone receives a notification from the beacon • The beacon saves the account information locally
Exception scenarios	• The app doesn't get the notification because the phone is off • The app doesn't get the notification, due to a communication failure • The beacon fails to detect the phone • The app is installed but the owner doesn't have an account
Postconditions	• Phone is connected • Information exchange between phone and beacon
Related requirements	NFR2, FR2, NFR3

Table 3.1: Main use cases -

Usecase	Disconnect from beacon
Main actor	Beacon
Goal	Register a phone disconnection
Precondition	• Bluetooth is switched on • Use case 1
Main scenario	• The user leaves the train • The beacon checks available connections • The beacon sends gathered information to local server • Local server sends information to main server • Main server computes distance travelled by user
Exception scenarios	• Phone battery dies before user leaves the train • The beacon malfunctions and cannot check for connections
Postconditions	Travel distance and price are computed
Related requirements	NFR3, FR3, FR10, FR9

Table 3.2: Main use cases - use case 2

Usecase	Calculate travel fee
Main actor	Server
Goal	Charge user for travelled distance
Precondition	• Usecase 1 • Usecase 2
Main scenario	• Main server receives information regarding users which are still connected • Main server compares accounts currently connected with accounts connected for previous stop • Users which were connected previously and are not connected anymore are identified • Travel distance and fee are calculated for identified users
Exception scenarios	• Information cannot be transmitted between local and main servers • Account is not identified
Postconditions	• Notification regarding payment is sent to user • Payment is made
Related requirements	FR3, FR9, FR10, FR13

Table 3.3: Main use cases - use case 3

3.3 Technical context

The system will need to perform in a specifically designed technical context consisting of three main environments: a central server collecting all information and performing necessary computations, the setup inside the trains themselves (comprised of beacons and small servers which will collect information per train and send it to the central one every time the train leaves the station) and a login system in each station allowing users to manually check-out of their trip and perform payment.

A set of measures will need to be taken into consideration for the system to be able to operate inside this context by using these different components and environments. These measures will be related to the key attributes required of the system as follows:

- **security**: since the system will have access to sensitive information (such as location, financial transactions), every connection between components of the system will need to be secured and trusted. The main connections which will need to be verified each time they are created will be: beacon to mobile application, beacon to server inside the train, small local server to central server.

- **reliability**: for the system to be considered reliable, the main components should be provided with a back-up in case of failure, such as the central server, the beacons in each cart, etc. Also, users should be presented with alternative scenarios in case the main desired activity flow gets interrupted (e.g. possibility to manually check-out of a trip using login system in train station in case of beacon to phone communication being severed)

- **compatibility**: the system will need to be compatible with at least the main and most commonly used mobile OS and frequencies

3.4 External interfaces

Channel	Beacon ⇒ Mobile app
Description	Beacons detect mobile phones in their proximity having installed the application and initialise a connection.
Connection	Bluetooth
Protocol	iBeacon
Frequency	The train has left a station and a mobile phone is in range of the beacon.

Table 3.4: Interface - Beacon to phone

Channel	Beacon ⇒ Local train server
Description	Beacons send information gathered every time a train departs from a station.
Connection	Bluetooth
Protocol	iBeacon
Frequency	The train has left a station.

Table 3.5: Interface - Beacon to server

Channel	Local server ⇒ Central server
Description	The local server situated inside the train sends information gathered periodically to the central server if an internet connection is possible.
Connection	Internet
Protocol	UDP
Frequency	Once every 20 min or when a connection is possible.

Table 3.6: Interface - Local server to central server

Chapter 4

Solution strategy

This chapter aims to discuss and present a concrete solution space for the problems described in chapters 2 and 3. The solutions provided will directly follow the main key drivers of the project and focus on the problem space using views limited only to these most important attributes.

4.1 Security

4.2 Compatibility

4.3 Reliability

Chapter 5

Building block view

Chapter 6

Runtime view

Chapter 7

Deployment view

Chapter 8

Concepts

8.1 Domain model

8.2 Architectural patterns

8.3 Security

8.4 Compatibility

8.5 Reliability

Chapter 9

Design decisions

Chapter 10

Quality scenarios

Notes

Appendix A

Time tracking