UNIVERSITY OF ZAGREB

FACULTY OF ORGANISATION AND INFORMATICS

V A R A Ž D I N

**Viktor Lazar**

**Martina Šestak**

**Goran Vodomin**

**Matej Vuković**

BLUETOOTH LE SHOWCASE

SOFTWARE ANALYSIS AND DEVELOPMENT PROJECT

Varaždin, 2014.

UNIVERSITY OF ZAGREB

FACULTY OF ORGANISATION AND INFORMATICS

V A R A Ž D I N

**Team number:** T1

**Team name:** Heisenbug

**Team members:**

Viktor Lazar, 0108063551

Martina Šestak, 0016091250

Goran Vodomin, 0016092445

Matej Vuković, 0016094754

BLUETOOTH LE SHOWCASE

SOFTWARE ANALYSIS AND DEVELOPMENT

Mentors:

Dr.sc. Zlatko Stapić

Ivan Švogor, mag.inf.

Varaždin, November 2014.

Content

[1. Introduction 1](#_Toc403196091)

[2. Bluetooth Low Energy Technology 1](#_Toc403196092)

[3. General project details 1](#_Toc403196093)

[4. SCRUM details 1](#_Toc403196094)

[5. Project diagrams 1](#_Toc403196095)

# Introduction

Evolaris is a company that has been established in 2000 and since then its teams of experts have been developing many web and mobile applications. Besides that, they are also constantly studying new technologies and evaluating if it's possible to apply them to build useful and modern applications.

One of Evolaris' projects is NFC bonus programme that was recently launched at the Shopping City Seiersberg located in Graz. Customers can, by using this application, register online and become „Friend of Seiersberg“. When registering every customer receives a traditional plastic card that contains a “Friend Chip”. When he comes to Shopping City Seiersberg he has to find a so called “Friends Kiosk”

# Bluetooth Low Energy Technology

# General project details

# SCRUM details

# Project diagrams

In Graz our team was introduced with user specifications. Based on these information’s we created some UML diagrams. Firstly we will create use case diagram which describes how user can use our application. After that based on use case diagrams, activity diagrams were made. Each activity diagram describes specific use case.

* 1. Use case diagram

In use case diagram we have two actors, user and the system. System is made up of two parts, web service and terminal. Main role of system is to provide data for user about discounts, points and friend status. It also, through terminal, notifies the user when the user arrive into shopping center. On the other hand, user have to be registered so that he could use the application. When the user is registered there are three use cases that describes how to user can interact with application, these are “Starts application”, “Checks points” and “Checks for discounts”. Listed use cases are described on activity diagrams in next chapter.

* 1. Activity diagram

Activity diagram is used to describe dynamic aspects of system. Activity diagram shows flow of control or object flow with emphasis on the sequence and conditions. Based on use case diagram we can see three dynamic aspects of our system, they are “Starts application”, “Checks points” and “Checks for discounts”

* + 1. Starts application

To use application user has to be signed in, so for that purpose application has login form where users enter their username and password. When the data is entered, application passes data to web service. Web service verifies the data and sends information if the entered data is correct. Application receives data and based on information proceeds on or display error message. During entire time applications listens for beacon signals that terminals transmit. If entered data is correct and application received beacon signal, welcome message is displayed and user can use the application. Anytime user can send request for termination which closes the application.

* + 1. Checks points

To check points and see friend status user has to start module for that. In the beginning application instantiates form and send data request to web service. Web service further creates queries on database to retrieve data. When the data is retrieved, web service sends it to application which display the data. Users then reads number of points and friend status. Anytime user can send request for termination which closes the application.

* + 1. Checks for discounts

In order to see list of discounts user starts module for this purpose. Further logic is quite the same as at previous case, application instantiates form, send data request to web service. After that web service creates queries on database to retrieve data. In the end when is retrieved, web service sends it to application which displays the data. User reads list of discounts and chooses two options, to see discount details or goes to menu. Base on the choice application creates menu form or discount details form. Anytime user can send request for termination which closes the application.

* + 1. Checks for discount details

For every discount displayed user can see discount details and for that purpose starts module. After module is started, application instantiates form and sends data request to web service. Web service creates queries to retrieve data, and retrieved data proceeds to application. Application displays data which user read. Anytime user can send request for termination which closes the application.