FLEET OF TAXI SERVICE MANAGEMENT SYSTEM - LABORATORY EXERCISE 2: SOFTWARE REQUIREMENTS SPECIFICATION

Danielle Winter (563795), Frederick Nieuwoudt (386372), Stephen Friedman (360938) & Sello Molele (0604606X)

School of Electrical & Information Engineering, University of the Witwatersrand, Private Bag 3, 2050, Johannesburg, South Africa

1. Introduction

2. User Requirements

A brief description of the different users and their basic requirements is given. Furthermore, potential users were surveyed for additional requirements. The user requirements are specified based on the MOSCOW and FURPS methodologies [?].

2.1 Basic Customer Requirements

- Passengers
  - Registration
  - Account Maintenance
  - Lift Requests
- Drivers
  - Receive Fare
  - Complete Fare
- Administrators
  - Add drivers
  - Maintain driver details
  - View logs

2.2 Detailed User Requirements

Passenger Registration

- Passenger login details set up
- Enter personal details (name, age, gender)
- Enter contact details (email address, cellphone number)
- Enter banking details (Credit Card number and type, expiry date of card, CVV number)

Passenger Account Maintenance

- Ability to update details (Change in name or contact details and updated payment details)
- Account cancellation (optional)

Passenger Lift Requests

- Log a destination
- Log a current location
- Specify number of passengers to be transported
- Passenger payment confirmation
- Order a lift in advance (e.g. collection from the airport at a specified time)
- Passenger feedback (ride comfort, satisfaction)
- Job update (passenger is able to say if they have been collected)

Driver Fare Received

- Driver receives an order for a nearby customer
- Driver accepts order
- Passenger updated about taxi arrival time
- Driver status changed to unavailable

Driver Fare Completed

- Driver reports drop-off
- Driver status changed to free

Administrator Driver Addition

- Driver personal details added to database (Name, age, Identity number, gender, medical aid details, tax number)
- Driver contact details (residential and postal address, email address, cellphone number)
- Driver vehicle details (number plate, car model, colour and insurance details)

Administrator Driver Maintenance

- Update driver details for personal, contact and vehicle categories

Administrator Log Viewing

- View current job data
- View historical job data

3. System Specifications

3.1 Scope

The taxi fleet management system is a tool that provides a service where a user can request a taxi and the system will dispatch the closest available driver. The system back-end will be required to store information about drivers, customers, and the current pricing scheme. The system front end will be presented in the form of a web gui. This will provide a means for customers to input their current location and destination. The web interface will also provide a quoted price and allow for the customer to either accept or reject the quote.

3.2 Testing Plan

To achieve a quality product the test driven development(TDD) paradigm will be adopted. TDD calls for the writing of tests before writing the code to be used in the implementation. This encourages simple modular code. The TDD workflow follows the sequence of writing a test, validating that the new test fails, writing the application code, running all the tests together, and then refactoring. This sequence allows for the project to move from getting code working to refactoring the code into simple understandable modules.

3.3 Implementation Plan

Following the Scrum SDLC methodology, will provide shippable incrementally improved products. Though a minimum of required features will be needed for a product that can meet the project scope. Additional functionality can be incrementally provided over the life span of the product.