Software requirements specification (SRS)

Christof Sy, Michael Trittibach, Michael Brunner, Manuela Eschler

Introduction to Software Engineering

Computer Science
University Bern

Version: 1.0

October 2017

Contents

1 Introduction

1.1 Purpose

The System will inquire the following purposes:

- Reduce work/staff
  The System will reduce time to do the same work. Less persons can do the same job with the System.
- Get overview of all orders and drivers
  The logistician will have a good overview of all the orders, drivers and deliveries.
- Always available and up to date
  The informations on the system are always available and up to date.
- Analyze for improvement
  The estimated time for the deliveries and the actual time used will differ. The system allows to analyze this difference for improvements.

1.2 Stakeholders

The stakeholders represent the groups of people who will use the system. The system will serve their purposes in the best way possible.

- **Driver**
  The driver makes the deliveries. He needs at any time access to his current and future deliveries and their informations.
- **Logistician**
  The logistician organizes the tours for the drivers. He keeps the overview and reschedules failed deliveries.
- **User Admin**
  The User admin manages all users.

1.3 Definitions

No definitions required yet.

1.4 System Overview

The new System's overall purpose is to manage logistics. It assists the stakeholders driver and logistician in the best possible way.
The logistician has a overview of all the pending orders at every time. Out of all the pending orders the system supports him organizing tours for every driver. With this system the logistician is able to adjust and reorganize the tours if necessary with little effort.
The system supports analyzing completed tours for improvement.
The driver has an overview as well as details of his few next tours. He can update the status of every delivery in real-time.
The driver can take a look at his tours on a desktop as well as on a mobile. **The user admin** administrates all users on the system.

1.5 References

No references required yet.

1.6 Open Questions

1. Is a forth stakeholder necessary? Like a customer? Where are the orders from?
2. If there is a customer, can he follow his delivery?
3. Login, private link or open source?

2 Overall Description

2.1 Use Cases

2.1.1 Use cases of the logistician

The logistician can login to the website with his username and password. Here he has a broad overview of all deliveries and their information and he can assemble the open ones to tours. He than can assign tours to the drivers. After a tour is finished, the logistician can evaluate the estimated against the used time. When a delivery was unsuccessful, he then can reschedule these to different future tours.

1. Logistician login to website
   The logistician opens the website with a browser and logs in with username and password to the backend.
2. Logistician organizes tours
   The logistician has overview of all deliveries and their informations. He assembles the deliveries to tours.
3. Logistician assigns tours to drivers
   The logistician assigns the tour to a driver.
4. Logistician evaluates tours
   The logistician evaluates for every tour the estimated vs the used time.
5. Logistician reschedules failed deliveries
   The logistician adds failed deliveries to future tours.

2.1.2 Use cases of the driver

Just like the logistician, the diver can login to the website with his username and password. Here he has an overview of his tours with details of every delivery. He can change the status of a delivery to passed or failed and also adds the used time for the tour.

1. Driver login to website
   The driver opens the website with a browser and logs in with username and password to the frontend.
2. Driver has overview of his tours
   The driver has overview of his tours with details of every delivery.
3. Driver sets status of delivery
   The driver changes the status of a delivery to passed or failed.
4. Driver adds used time for tour
   The driver adds the used time for the tour.

2.1.3 Use cases of the user admin

The user admin logs in to the website like the others with a username and a password. He can add new users to the system and can set their settings.

1. user admin login to website
   The user admin opens the website with a browser and logs in with username and password to the backend.
2. User admin creates users on system
   The User admin adds new users to the system and set the settings right.

2.1.4 Use cases of the system

The system notifies the logisticians when a delivery status changes to failed.

1. System notifies logistician when delivery status is failed The system sends a notification to the logistician when the status of a delivery changes to failed.

2.2 Actor Characteristics

2.2.1 Logistician

The logistician would like to:

- Organize tours with help from the system
- Improve tours with little effort by evaluating past tours
- Reschedule tours without additional paperwork
- Have an Overview of the status of the deliveries

2.2.2 Driver

The driver would like to:

- Have an overview of his upcoming tours
- Update of the status of the deliveries immediately

2.2.3 User Admin

The user admin would like to:

- Add new users to the system
- Remove users from the system
- Modify users on the system

2.3 Open Questions

1. These are our initial recommendations to accomplish your request. Open for discussion.

3 Specific Requirements

3.1 Functional Requirements

Backend:

- User management: Add, modify and remove users
  Every user needs a username , a password and a status (logistician, driver, user admin or inactive).
  Inactive users cant log in to the system.
- Tour management: Add, modify and assign tours
  Logisticians have to be able to combine deliveries to a tour and assign them to a driver.
  Drivers have to be able to watch their upcoming tours and details of the tour. Drivers have to be able to report the used time of the tour.
  Logisticians have to be able to get a overview of the statistics of the passed tours.
- Delivery management: Add, update and modify deliveries
  Logisticians have to be able to get a overview of the deliveries witch are not already assigned to a tour or witch are failed.
  Drivers have to be able to set the status of a delivery to passed or failed.
- Report management: Compare estimated and used time of tours
  The system have to compare estimated and used time of tours, how many deliveries and how much of them passed or failed.

Frontend:

- Individual tour overview: Present tour and the assigned deliveries incl. details
  Drivers have to get a overview of their upcoming tours directly after logging in to the system. Before they start the tour the have to tell it to the system and at the end of the tour they have to tell it to the system.
- Delivery Status management: Update status of delivery
  After every passed or failed delivery the driver have to report the status of the delivery to the system.

3.2 Non-functional Requirements

- The web application has responsive design

3.3 Open Questions

1. How many user will be on this system?
2. If a delivery fails, should the driver be able to try it again on the same tour?
3. Where is the server located?
4. How many times a delivery can fail? What happens after it fails to much?
5. With what kind of data are we dealing?
6. How does the paperwork look like? take this as template?