|  |
| --- |
| NATIONAL INSTITUTE OF TECHNOLOGY ROURKELA |
| Transport Automation Vehicle System |
| SRS Document |
|  |
| **Yogesh Singla (115CS0243)** |
| **2/1/2019** |

|  |
| --- |
| Software Engineering Laboratory, 8th semester. Department of Computer Science and Engineering, NIT Rourkela |

Table of Contents

[1. Introduction 1](#_Toc536745581)

[1.1 Purpose 1](#_Toc536745582)

[1.2 Document Conventions 1](#_Toc536745583)

[1.2.1 Definitions 1](#_Toc536745584)

[1.2.2 Acronyms 1](#_Toc536745585)

[1.2.3 Abbreviations 1](#_Toc536745586)

[1.3 Intended Audience and Reading Suggestions 2](#_Toc536745587)

[1.4 Product Scope 2](#_Toc536745588)

[1.5 References 2](#_Toc536745589)

[1.6 Overview of Developer's Responsibillites 2](#_Toc536745590)

[2. Overall Description 2](#_Toc536745591)

[2.1 Product Perspective 2](#_Toc536745592)

[2.2 Product Functions 3](#_Toc536745593)

[2.3 User Classes and Characteristics 3](#_Toc536745594)

[2.4 Operating Environment 3](#_Toc536745595)

[2.5 Design and Implementation Constraints 4](#_Toc536745596)

[2.6 User Documentation 4](#_Toc536745597)

[2.7 Assumptions and Dependencies 4](#_Toc536745598)

[3 External Interface Requirements 4](#_Toc536745599)

[3.1 User Interfaces 4](#_Toc536745600)

[3.2 Hardware Interfaces 5](#_Toc536745601)

[3.3 Software Interfaces 5](#_Toc536745602)

[3.4 Communication Interfaces 5](#_Toc536745603)

[4. System Features 5](#_Toc536745604)

[4.1 System Feature 1 : Renting Vehicles 6](#_Toc536745605)

[4.1.1 Description and Priority 6](#_Toc536745606)

[4.1.2 Stimulus/Response Sequences 6](#_Toc536745607)

[4.1.3 Functional Requirements 6](#_Toc536745608)

[4.2 System Feature 1 : Vehicle Management 8](#_Toc536745609)

[4.2.1 Description and Priority 8](#_Toc536745610)

[4.2.2 Stimulus/Response Sequences 9](#_Toc536745611)

[4.2.3 Functional Requirements 9](#_Toc536745612)

[4.3 System Feature 1 : Data Collection and Analysis 10](#_Toc536745613)

[4.3.1 Description and Priority 10](#_Toc536745614)

[4.3.2 Stimulus/Response Sequences 11](#_Toc536745615)

[4.3.3 Functional Requirements 11](#_Toc536745616)

[5. Other Nonfunctional Requirements 13](#_Toc536745617)

[5.1 Performance Requirements 13](#_Toc536745618)

[5.1.1 Response Time 13](#_Toc536745619)

[5.1.2 Capacity 13](#_Toc536745620)

[5.1.3 Availability vs Consistency 13](#_Toc536745621)

[5.1.4 Errors and Debugging 13](#_Toc536745622)

[5.2 Safety Requirements 14](#_Toc536745623)

[5.3 Security Requirements 14](#_Toc536745624)

[5.4 Software Quality Attributes 14](#_Toc536745625)

[5.5 Business Rules 14](#_Toc536745626)

[6. Other Requirements 14](#_Toc536745627)

[Appendix A: Glossary 15](#_Toc536745628)

[Appendix B: Analysis Models 15](#_Toc536745629)

[Appendix C: To Be Determined List 15](#_Toc536745630)

# 1. Introduction

This section describes the background information needed to understand the rest of the SRS.

## 1.1 Purpose

The product is a software for car rental company. This is the first release, version 1. The scope of this SRS is complete with the standalone software system.

## 1.2 Document Conventions

The order of appearance of sections does not account for their relative priority. Where-ever necessary priority has been explicitly stated.

### 1.2.1 Definitions

None

### 1.2.2 Acronyms

None

### 1.2.3 Abbreviations

TAVS : Transport Automation Vehicle System

## 1.3 Intended Audience and Reading Suggestions

The intended audience for the SRS is the design architect for the software, management of the company and developers.

## 1.4 Product Scope

The scope of this project is limited to learning and experience of software developement lifecycle. The authors do not take responsiblity of any deployment in the real world.

## 1.5 References

IEEE Software Requirements Specification Template : https://web.cs.dal.ca/~hawkey/3130/srs\_template-ieee.doc

## 1.6 Overview of Developer's Responsibillites

The developer should stick to the requirements of the system.

Plagiarism should be avoided and due credit given to sources.

# 2. Overall Description

## 2.1 Product Perspective

This is a standalone application. It is not a part of any other software and is independent. Specifically, this product is being developed for educational purposes as the part of Software Engineering Labortory with best possible real world scenario simulation.

## 2.2 Product Functions

The product has one main function:

\* To be deployed in a car rental company.

Users can rent vehicles and company can manage the vehicle fleet and take in statistical data.

This has been elaborated in System Features section.

## 2.3 User Classes and Characteristics

USER

This is the user of the software who wishes to rent vehicles. There characteristic is:

\* Limited knowledge of the software

\* Exposed to client view only

\* Need little to no expertise for the software

ADMINISTRATOR

This is the user who manages the software on the company side. Server and database is also handled. There characteristic is:

\* Higher expertise needed about software

\* Need to collect and understand statistics and debug in case of problems

\* Exposed to all views of the software

## 2.4 Operating Environment

The software will be deployed on a server and connected to internet.

The operating environment should be atleast Windows 7 or Linux server version after 2015.

## 2.5 Design and Implementation Constraints

The server is single point only. The design has not accounted for distributed server and consequentyl distributed database that may have challenges like sharding etc. This limits the maximum users for the software and is scalable upto a limit of few thousand users only.

The implementation is proposed in python only language with NoSQL database.

## 2.6 User Documentation

User documentation for the software will be available on github @yogeshsingla

## 2.7 Assumptions and Dependencies

It is assumed internet connectivity is available at all times. The project has dependencies as the python project.

# 3 External Interface Requirements

## 3.1 User Interfaces

There are two base views.

Client side user interface for renting vehicles.

Company/server side user interface for management and data collection.

Both views will run on different machines with possible one to many relation b/w server and client views.

## 3.2 Hardware Interfaces

The system would need a computer to run on both sides.

Client side should have minimum hardware requirements of a internet connectivity supported machine with at least 512 MB of RAM and 1 GB of storage space.

Server side would need to cater to multiple clients and hence needs significant compute power. It should have basic server setup and internet connectivity.

## 3.3 Software Interfaces

The storage of data should be organised in a database. The system views for server and client should be in the form of GUI or at least CLIs.

## 3.4 Communication Interfaces

Network connectivity is required for client and server to communicated with each other. The database also needs to communicate with the server side similar to a MVP architecture.

# 4. System Features

We arrange the system features in a functional hierarchy as that makes the most sense for this product – Transport Automation Vehicle System. We have three main features of TAVS. First, renting of cars. It is of utmost importance as this feature generates the revenue for the company and will the customer facing. Second, vehicle management. This is important backend feature for the company. Third, data collection and analysis. This is of lower priority than the other two features but will be useful in making the company more effecient and profitable in longer run.

## 4.1 System Feature 1 : Renting Vehicles

### 4.1.1 Description and Priority

This feature will allow customers to browse the available vehicles. The vehicles and charges will be displayed along with the constraints of booking. The customer requests and the software confirms the booking after relevant processing at backend. The customer sets return date and time and renting has started. On return of vehicle the rent transaction is complete.

It is high priority feature as this is going to generate revenue for the company. Specific priority component ratings:

Benefit: 8

Penalty: 6

Cost: 5

Risk: 4

(each rated on a relative scale from a low of 1 to a high of 9).

### 4.1.2 Stimulus/Response Sequences

Sequence of events is: browse, request, book, return.

### 4.1.3 Functional Requirements

DESCRIPTION

These requirements are in serial. These requirements serve the customer and processing happens the the company side only.

#### 4.1.3.1 Requirement 1 : Request Car

STATE

New User Screen

All vehicles availability and charges are shown

INPUT

Car selected for booking with date and time of return

OUTPUT

Booking page confirmation shown to user

PROCESSING

Only vehicles available currently are displayed. This processing is done on company backend side.

#### 4.1.3.2 Requirement 2 : Book Car

STATE

Booking confirmation page

INPUT

User deposits a security amount

OUTPUT

Booking is confirmed subject to all conditions

PROCESSING

The vehicle current mileage and fuel is stored and status marked to rent for the user specified time and dates.

The amount paid by user is recorded.

#### 4.1.3.3 Requirement 3 : Return Car

STATE

Return User Screen

INPUT

Vehicle ID

OUTPUT

Final charges returned to user

PROCESSING

The final payment is decided based on deposit and cost incurred

Car status changed and relevant stats updated

## 4.2 System Feature 1 : Vehicle Management

### 4.2.1 Description and Priority

This feature allows the company the manage it's fleet of vehicles. It also ensures consistency and availability to the customers.

Specific priority component ratings:

Benefit: 8

Penalty: 8

Cost: 9

Risk: 3

(each rated on a relative scale from a low of 1 to a high of 9).

### 4.2.2 Stimulus/Response Sequences

The events can be adding new vehicle, selling old vehicles, monitor state of current vehicle

### 4.2.3 Functional Requirements

DESCRIPTION

#### 4.2.3.1 Requirement 1 : Add Vehicle

STATE

Administrator View

INPUT

Vehicle details to be added

OUTPUT

Unique ID generated for the new vehicle and it is added to the system

PROCESSING

Basic update operations occur on the system end after checking for conflicts

#### 4.2.3.2 Requirement 2 : Remove Vehicle

STATE

Administrator View

INPUT

Vehicle ID to be sold/removed.

OUTPUT

Success or Failure alert

PROCESSING

Basic delete operation occurs on the system end

#### 4.2.3.3 Requirement 3 : Monitor Vehicle

STATE

Administrator View

INPUT

Vehicle ID

OUTPUT

Current state of vehicle from available, rented or in repair.

PROCESSING

Basic search/select operation occurs on the sytem end

## 4.3 System Feature 1 : Data Collection and Analysis

### 4.3.1 Description and Priority

This feature will allow company to make informed choices about prices and vehicles to keep through proper data collection.

It is relatively low priority feature as this does not directly generate revenue for the company. Specific priority component ratings:

Benefit: 4

Penalty: 1

Cost: 6

Risk: 3

(each rated on a relative scale from a low of 1 to a high of 9).

### 4.3.2 Stimulus/Response Sequences

For the administrator side only,

Amount of maintenance expenses per vehicle estimated values.

Statistics page about various types of vehicles

Profitability and charges for each vehicle

### 4.3.3 Functional Requirements

DESCRIPTION

For the data collection and analysis feature, there are are three main requirements. One, maintenance charges calculation. Second, base statistics for each vehicle. Third, profitability and charge calculation per vehicle.

#### 4.3.3.1 Requirement 1 : Maintenance Charge

STATE

Administrator View

INPUT

Vehicle Type and ID

OUTPUT

Maintenance Charges incurred

PROCESSING

Processing occurs on the administrator side only. The maintenance charges are taken from external interface to the repair shops of the company.

#### 4.3.3.2 Requirement 2 : Statistics

STATE

Administrator View

INPUT

No input required

OUTPUT

Tabular display of all statistics till date

PROCESSING

The processing occurs on company side with data collected in each transaction stored in a database

#### 4.3.3.3 Requirement 3 : Profit and charge calculation

STATE

Administrator View

INPUT

Vehicle Type or ID

OUTPUT

Current profitability of the vehicle and the recommended charge that should be levied on the customer for that vehicle

PROCESSING

The statistics collected are processed to come up with charge for the car by linear regression

# 5. Other Nonfunctional Requirements

These requirements do not form the core of the software being developed but do play a major role in the percieved reliabilty and likeability for the users.

## 5.1 Performance Requirements

### 5.1.1 Response Time

The response time should be within the human visual system resolution ie 10ms. The user may tolerate upto 1s but above that would be a potential for the user to get annoyed and may switch products.

### 5.1.2 Capacity

The scale of the software should be sufficient for production environment.

Atleast 100 users at one time can access the system.

Atleast 100 vehicles should be supported by the system.

### 5.1.3 Availability vs Consistency

The availability of the system should be above 90%. That is atleast 9 out of 10 times, the user should be able to access the system in a given time period. Consistency is important during the booking phase and availability of data in monitoring the vehicles.

### 5.1.4 Errors and Debugging

The system should record logs of all levels - errors, warnings, changes, debug and verbose.

This will help operations team to get information and take quick responses.

## 5.2 Safety Requirements

No safety requirements have been considered as of now.

## 5.3 Security Requirements

The system proposed doesn't take mandatory account signup and data collection. This is to promote data privacy policies of GDPR and make users more confident of the system use. This may pose some security concerns including spoofing threats.

## 5.4 Software Quality Attributes

Covered in Section 5.1.3

## 5.5 Business Rules

The users of the software must have valid Driving License. This is not verified as of now although there details will be taken during renting process.

# 6. Other Requirements

The software will be released under GPL license. Reuse is permitted with due credit to this source as follows:

AUTHOR : YOGESH SINGLA

GITHUB : @yogeshsingla

# Appendix A: Glossary

TAVS : Transport Automation Vehicle System

# Appendix B: Analysis Models

Will be covered soon.

# Appendix C: To Be Determined List

None