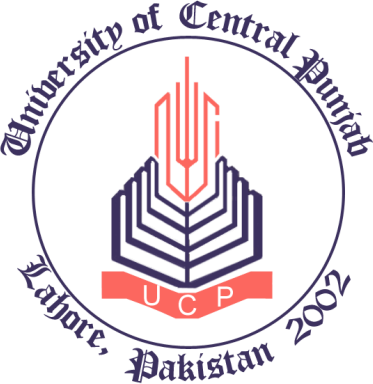
BSCS FINAL PROJECT

Requirements Specification

THE SHIPPING SAINTS



##### Project Advisor

### Presented by:

**Group ID:**

**Faculty of Information Technology**

**University of Central Punjab**

**Software Requirements Specification**

**Version 1**

**THE SHIPPING SAINTS**

# Table of Contents

[Table of Contents 1](#_Toc57118625)

[Revision History 2](#_Toc57118626)

[1. Introduction and Background 1](#_Toc57118627)

[1.1 Product (Problem Statement) 1](#_Toc57118628)

[1.2 Background 1](#_Toc57118629)

[1.3 Scope 1](#_Toc57118630)

[1.4 Objective(s)/Aim(s)/Target(s) 2](#_Toc57118631)

[1.5 Challenges 3](#_Toc57118632)

[1.6 Learning Outcomes 4](#_Toc57118633)

[1.7 Nature of End Product 4](#_Toc57118634)

[1.8 Completeness Criteria 5](#_Toc57118635)

[1.9 Business Goals 5](#_Toc57118636)

[1.10 Related Work/ Literature Survey/ Literature Review 5](#_Toc57118637)

[1.11 Document Conventions 5](#_Toc57118638)

[2. Overall Description 6](#_Toc57118639)

[2.1 Product Features 6](#_Toc57118640)

[2.2 User Classes and Characteristics 6](#_Toc57118641)

[2.3 Operating Environment 6](#_Toc57118642)

[2.4 Design and Implementation Constraints 6](#_Toc57118643)

[2.5 Assumptions and Dependencies 6](#_Toc57118644)

[3. Functional Requirements 7](#_Toc57118645)

[3.1 Login 9](#_Toc57118646)

[3.2 Upload order 9](#_Toc57118647)

[3.3 Requirements Analysis and Modeling 12](#_Toc57118648)

[4. Nonfunctional Requirements 16](#_Toc57118649)

[4.1 Performance Requirements 16](#_Toc57118650)

[4.2 Safety Requirements 16](#_Toc57118651)

[4.3 Security Requirements 16](#_Toc57118652)

[4.4 Additional Software Quality Attributes 16](#_Toc57118653)

[4.4.1 Usability: 16](#_Toc57118654)

[4.4.2 Flexibility: 16](#_Toc57118655)

[5. Other Requirements 16](#_Toc57118656)

[No there is no other requirement 16](#_Toc57118657)

[6. Revised Project Plan 16](#_Toc57118658)

[7. References 17](#_Toc57118659)

[Appendix A: Glossary 19](#_Toc57118660)

[Appendix B: IV & V Report 20](#_Toc57118661)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction and Background

When the industries grows we have many challenges to face. As we know that Ecommerce industry is growing very rapidly. The challenge now a days is late delivery by the ecommerce store owners. We are going to make a system, which helps to make a delivery as soon as possible, or we can also say by our system the delivery can be made in one day. We will make an independent platform where a store owner deliver his/her orders they need to open the mobile app and put the order shipping details there according to the limit of one rider and then press the done and then the delivery riders who are register with our system received request notification for order, if he accepts the request he would have to pick-up order from the warehouse/store and update the status to parcel collected, then our app generate a shortest path using Artificial Intelligence from warehouse/store Destination and calculate the fare based on distance and parcel weight to all destinations, by this the cost is also reduced and when rider deliver a parcel he will update the status to order delivered and receive cash from customer, and if the customer doesn’t receive parcel the status will be updated to not received.

## Product (Problem Statement)

Now days there are many problems of deliveries in ecommerce industry.

* Late delivery is the biggest problem with current delivery system; our idea is for solving this problem.
* Delivery charges are high, current charges are about Rs.200, because of this the profit margins of ecommerce storeowner are reduced.
* Payment clearance of ecommerce storeowners from shipping companies are too late.
* The file of exported orders from the site will be upload on our application, so the user would not have to enter orders manually, which is a time consuming process.

## Background

Our project targets the local ecommerce industry for the fast and affordable delivery cost for merchants. We will make an independent platform where a store owner deliver his/her orders they need to open the mobile app and put the order shipping details there according to the limit of one rider and then press the done and then the delivery riders who are register with our system received request notification for order, if he accepts the request he would have to pick-up order from the warehouse/store and update the status to parcel collected, then our app generate a shortest path using Artificial Intelligence from warehouse/store Destination and calculate the fare based on distance and parcel weight to all destinations, by this the cost is also reduced and when rider deliver a parcel he will update the status to order delivered and receive cash from customer, and if the customer doesn’t receive parcel the status will be updated to not received.

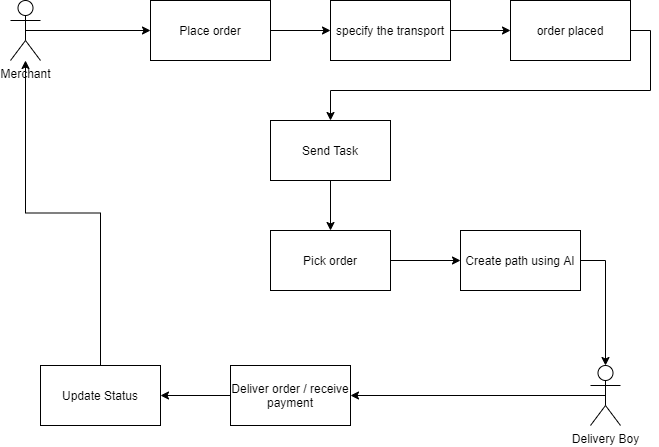
## Scope

Basically it an app and web based application that delivers the products at your door step within one day in city. You can use our app for the below mentioned points

* For overnight delivery of urgent shipments across the country
* Safe and reliable delivery of valuable parcels across the nation within 48 hours
* Send heavy shipments at low prices by road nationwide
* Take advantage of quick, reliable deliveries within days to any corner in the country
* Reliable and speedy delivery of cargo to anywhere in country
* We go that extra mile when it comes to delivering your shipments.
  + Special handling & fragile shipments
  + Time definite shipments
  + Holiday & weekend deliveries
  + Individually customized solutions with us can package, protect and transport your valuables wherever you want.

Offering hassle-free payments, through our convenient collection solutions. The rider will submit the collected cash to the company’s Bank Account, then company will transfer the payment to merchant.

We can provide complete delivery solution for all your ecommerce needs with an matchless service. Being extremely customer focused, we are highly adaptable when it comes to our customer’s requirements. We can offer time-sensitive deliveries within a day, and deliveries outside normal business hours, we can deliver your shipment specifically where you want it, when you want it.



## Objective(s)/Aim(s)/Target(s)

Our app is made for your e-commerce platform. With compatibility for Shopify and WordPress, as an E-Retailer you can rest assured that no matter what your E-Store platform is. Once you click on an order in your E-store dashboard, that order seamlessly falls into our shipping application where you can process it by performing

* Generate Order and search for rider
* Rider will pick up the items
* Track the order
* Bulk orders
* Add or remove products from within the order
* Leaves customer service or operation specific comments against the order
* Shortest path selection
* Order status and Payment received notification

You can also view your outstanding payments from and review your account statements and history.

## Challenges

**Rider selection:**

So in our system The Shipping Saints we are follow some steps in order to register the rider with our platform. We will take care all the security concerns because riders have luggage of many customers which have value in both terms cost and the merchant customer feedback so we will make our steps for rider selection in such a way where above mentioned responsibilities of riders should be catered.

The steps we follow are as follow:

* First rider must have registered vehicle with the government and have the driving license.
* Rider will registered with the original documents.
* Rider must know how to use the apps and how to manage it and must know how to use maps locations.
* Rider will must have Rs25000, which he paid as a security.
* Rider also gave us two guarantors who gives their guarantee that if he break his loyalty agreement the guarantors will pay the loss.

Rider will go through all above process and get the green flag for confirmation with our platform

**System Selection:**

Our system assure the user and the rider best out of best facility from this automation ecommerce buyers, ecommerce merchants and the rider all will be happy we are offering services in our systems:

* Special Pickups and Deliveries
* Insurance
* Packaging
* Tracking
* Invoicing and Billing
* Freight forwarding
* Technology solutions

**Route Selection:**

So the route selection criteria for the riders to delivers the parcels on time and efficiently, for this we are going to implement an Artificial intelligence algorithm and solve travel sales man problem by naive solution. When we implement this the fares of deliveries also reduced because when the orders are placed into our app it will automatically sorted and make the shortest path, where all the orders are arranged in the order. Where a shortest path is made and rider will follow the path and keep on going and delivering the parcels so by this we will reduce.

* Delivery time: This reduced when we put a shortest path for Rider. It will delivers the parcels in less time because all the address are arranged in the manner where from first order till last will follow the continuous path we will avoid the way like the first order is far away from 2nd we will put the orders which next who have most near from previous order deliver.
* Reduced fuel consumption: when we caters the above thing so the fuel consumption will automatically reduce and when the fuel consumption is reduced the fares will also automatically reduced
* More productivity: And when all the system will goes on automation, the productivity will be improved.
* Run time addition of orders: if the riders are delivering orders and at the same mean time some merchant will generates a call and if the riders are available nearby he will received the call if he have free slots then they will added on his path and he will get the parcels from the merchant and delivers them.

## Learning Outcomes

After this project, we will have grips on following language and the frameworks:

**Framework:**

* Flutter

**Languages:**

* Java Script
* HTML
* PHP
* Java
* XML

**Software:**

* Android Studio
* Chrome
* VS Code
* Sublime Text

## Nature of End Product

Our app is made for your e-commerce platform. With compatibility for Shopify and WordPress, as an E-Retailer you can rest assured that no matter what your E-Store platform is. Once you click on an order in your E-store dashboard, that order seamlessly falls into our shipping application where you can process it by performing

* Generate Order and search for rider
* Rider will pick up the items
* Track the order
* Bulk orders
* Add or remove products from within the order
* Leaves customer service or operation specific comments against the order
* Shortest path selection
* Order status and Payment received notification

You can also view your outstanding payments from and review your account statements and history.

## Completeness Criteria

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Criteria** | **Weightage %** |
| 1 | Desktop GUI | 20 |
| 2 | Mobile Application Development | 50 |
| 3 | Algorithm Development | 15 |
| 4 | Communication | 15 |

## Business Goals

As we, all know about the e-commerce, industry is growing very rapidly. So every startup or business modal which is related to e commerce is must be beneficial. Because from last 2 – 3 years this industry is setting up new trends and enabling entrepreneur to make millions of dollars so if we can overcome the major issue which is delivery on time so our idea must be successful and if it will become successful then it must have a great business potential of making millions.

## Related Work/ Literature Survey/ Literature Review

We have seen many applications with respect to this area yet we have not discovered such application in Pakistan. We have the exploration of numerous applications yet we have just discovered some courier services but these don’t ease the e-commerce merchant so we propose this proposition to beat this trouble for the e-commerce industry so merchant can simply signup and deliver his orders to customers.

## Document Conventions

* Font for the description is Arial
* Arial size is 11.
* Heading font is Times New Roman, Size 18.
* Size for Sub-Heading Times New Roman, Size 14.
* Line Spacing between text is 1.5.
* Heading section font is Cambria (Headings) italic and its size is 10.

# Overall Description

## Product Features

The main feature of product is that

* You simply add your order file there
* It will automatically arrange order in such a way that creates less time and fuel consuming path
* You simply select rider according to your convince
* Rider will accept your request and reach there
* You can have life location functionality with currently and the next order that is going to deliver location
* You can track them easily
* You can have your payment and detail there after being updated each order deliver

## User Classes and Characteristics

**We mainly have 2 primary user classes**

**Merchant:**

The ones who is going to use our app for delivering of her products

**Driver:**

The one who is going to deliver the products

## Operating Environment

The shipping Saint is designed in such a way that one can easily access it from mobile using application as we have 2 different application one for driver and one for merchant while merchant will have access to web application if he/she wants to use it can have it

## Design and Implementation Constraints

We always focus on the security of our customers so in this regard we have developed an security plan   
plan for drivers:

They need to visit our office with police verification from his local area and 2 guarantor then we will sign an security agreement with them

Plan for merchant:

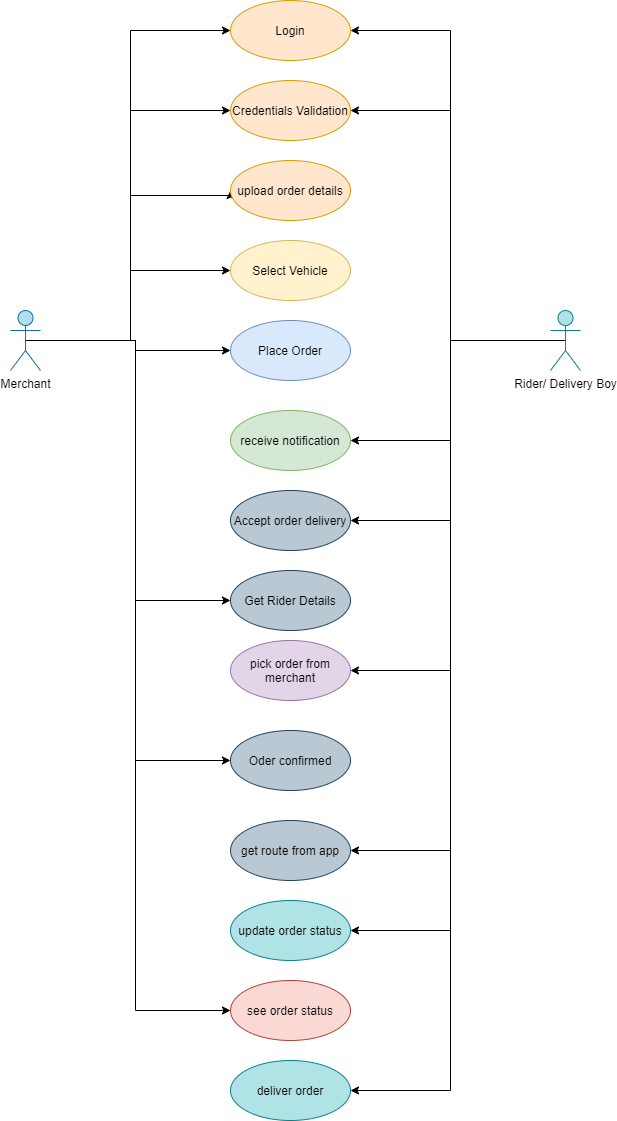
The only need to register their shop/business with our platform and the will be given an agreement to sign

## Assumptions and Dependencies

We assume that if number of rider are busy than fair would be increased ,in case if no user is available in your area the you will have to go to open booking option where the ones who accept the ride will receive extra charges ,If you book on faster delivery then you have to pay double charges

# Functional Requirements

The rider will have mobile app through which he register himself. And store owner have mobile app / web app where he can place the order for delivery. If the storeowner has 10 or 20 order, we will make our app, which supports different cms order files to be uploaded on our system and all the orders details added to database automatically no need of placing details one by one. Our app have different transportation vehicles categories depending on the dimensions of product. Which includes bikes, autos, and pickups, next our app generate a shortest path using Artificial Intelligence from warehouse/store to destination, this is for the rider benefits as we have 10 orders so our system automatically makes the shortest path from where all the orders can be delivered in shortest time and efficiently. so it can be a time and fuel efficient service. When rider deliver a parcel he will update the status to order delivered and receive cash from customer, and if the customer does not receive, parcel the status will be updated to not received, the status notification and fare for each delivery will send to owner. And in the last if the orders are on cod then the cash should be clear at the end of the day and the payment clearance issues also be solved because of this.



## Login

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-1 | |
| **Purpose** | | To login account | |
| **Priority** | | High | |
| **Pre-conditions** | | App must have internet connection | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Enter Login credentials | |  |
| **2** | Press login button | | System will check login validation |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Enter wrong credentials | | System will show error |

#### Table 1: UC-1

## Upload order

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-2 | |
| **Purpose** | | Upload the orders file to add in system | |
| **Priority** | | High | |
| **Pre-conditions** | | App must be logged in | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | User upload orders csv file | |  |
| **2** | Click upload button | | System will check the format and add orders to database |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | If user upload wrong/empty file | | System will show error |

#### Table 1: UC-2

**Select vehicle**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-3 | |
| **Purpose** | | Select vehicle according to requirements | |
| **Priority** | | High | |
| **Pre-conditions** | | Orders files must be uploaded | |
| **Post- conditions** | |  | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Click on vehicle button | | A list of vehicle appears |
| **2** | Select vehicle | | System will search vehicle nearby |

#### Table 1: UC-3

**Place Order**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-4 | |
| **Purpose** | | To confirm order | |
| **Priority** | | High | |
| **Pre-conditions** | | Vehicle must be selected | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | User will click “Place order” Button | | System will send notification to nearby riders |
| **2** | Searching popup will appear | |  |

#### Table 1: UC-4

**Receive Notification/accept/reject**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-5 | |
| **Purpose** | | To book a rider | |
| **Priority** | | High | |
| **Pre-conditions** | | Rider must be available | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Rider receive notification | |  |
| **2** | Rider accepts notification | | System will send rider details to merchant, and show him route to pickup location |
| **Alternate Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Rider reject ride notification | | System will continue search and wait for other rider acceptance |

#### Table 1: UC-5

**Pickup Order/ Start Ride**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-6 | |
| **Purpose** | | Reach pickup location and start ride | |
| **Priority** | | High | |
| **Pre-conditions** | | Must accepted the ride request | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Rider will go to pickup location | |  |
| **2** | Rider press arrived button | | System will send notification of rider arrival |
| **3** | Rider starts ride | | System will send notification of order status to Merchant, and show route to delivery location. |

#### Table 1: UC-6

**Start order deliver**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-7 | |
| **Purpose** | | Order Delivery | |
| **Priority** | | High | |
| **Pre-conditions** | | Pickup order | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Deliver order and update status | |  |
| **2** | Update the status of order | | System will send notification of order delivery status to Merchant. |

#### Table 1: UC-7

**Receive payment**

|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-8 | |
| **Purpose** | | Payment collection | |
| **Priority** | | High | |
| **Pre-conditions** | | Order must be deliver | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Rider will get payment from customer | |  |
| **2** | Rider update payment status | | System will send notification of payment received to merchant |

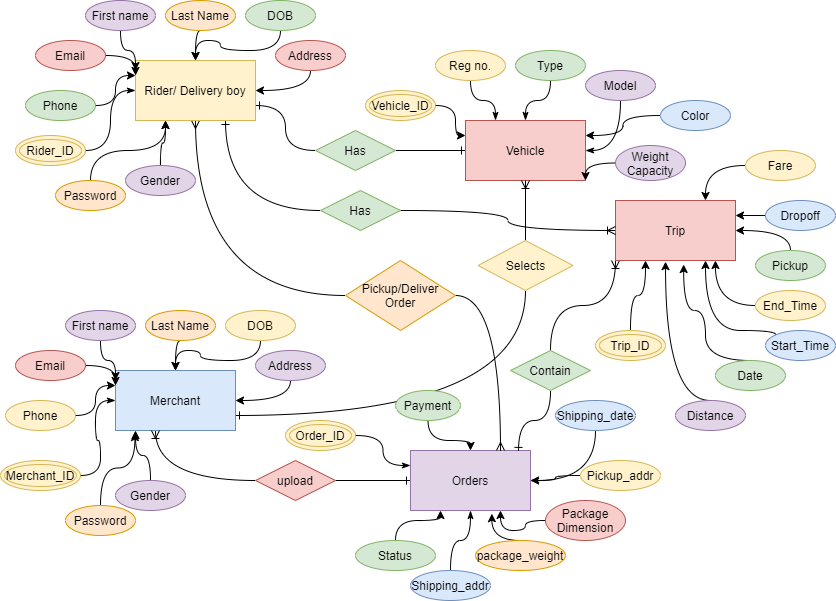
#### Table 1: UC-8

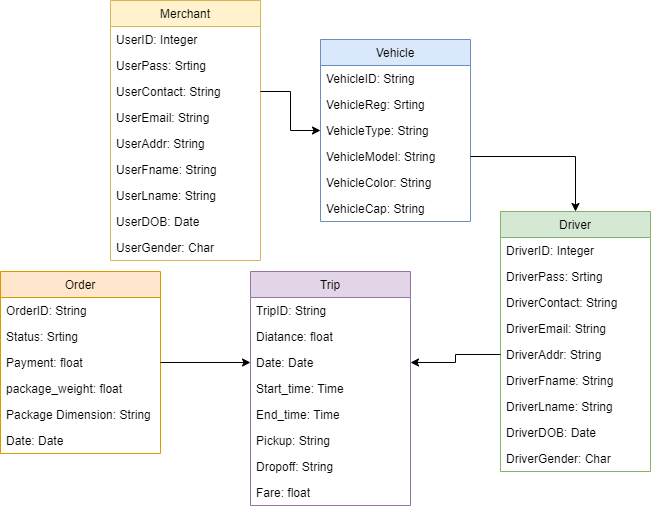
**End Ride**

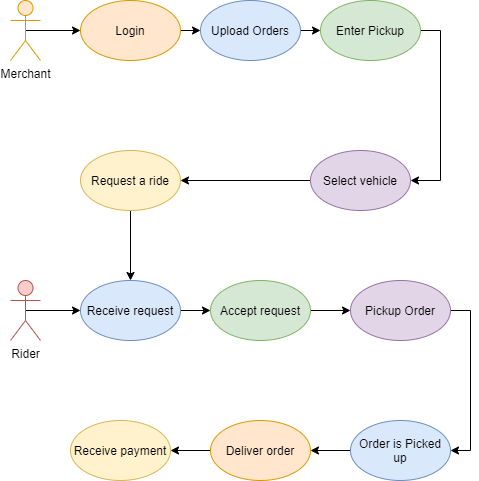
|  |  |  |  |
| --- | --- | --- | --- |
| **Identifier** | | UC-9 | |
| **Purpose** | | To available for new ride | |
| **Priority** | | High | |
| **Pre-conditions** | | Delivered all orders | |
| **Post- conditions** | | … | |
| **Typical Course of Action** | | | |
| **S#** | **Actor Action** | | **System Response** |
| **1** | Rider will end ride | | System will make rider available for new rides |
| **2** | Rider is ready for new rides | |  |

#### Table 1: UC-9

## Requirements Analysis and Modeling







# Nonfunctional Requirements

## Performance Requirements

To use Shipping Saint at its best quality you need to have a high speed internet so there could be less chances of any loss of rider and you can easily use all the functionality

## Safety Requirements

We focus on the security of our users so in this regard we will save our users data very safely and will try our best to save it from any hacking as we have payment option there in our app so we will have a certificate of clearance from specific company so that there is no risk in transaction of money

## Security Requirements

We always focus on the security of our customers so in this regard we have developed an security plan   
 for drivers:  
they need to visit our office with police verification from his local area and 2 guarantor then we will sign an security agreement with them

Plan for merchant:

The only need to register their shop/business with our platform and the will be given an agreement to sign

## Additional Software Quality Attributes

## 4.4.1 Usability:

The shipping saints is easy to use as it is a mobile application and every user know how to operate it.

## 4.4.2 Flexibility:

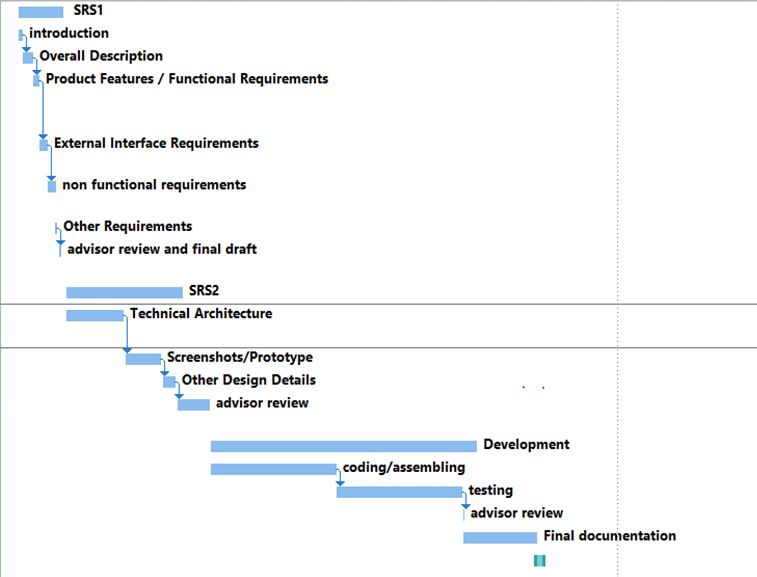
It is flexible to use as it is going to develop keeping the view of an ordinary users

## 

# Other Requirements

# No there is no other requirement

# Revised Project Plan



# References

<List all books, conference papers, journal articles, websites, etc. used in preparing the content of this SRS. Provide enough information so that the reader could access a copy of each reference, including

title, author, volume/edition number, page number(s), and publication year. Mention complete URLs for websites.>

# Appendix A: Glossary

<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>

# Appendix B: IV & V Report

#### (Independent verification & validation) IV & V Resource

##### Name Signature

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **S#** | **Defect Description** | **Origin Stage** | **Status** | **Fix Time** | |
| **Hours** | **Minutes** |
| 1 |  |  |  |  |  |
| 2 |  |  |  |  |  |
| 3 |  |  |  |  |  |
| … |  |  |  |  |  |

#### Table 2: List of non-trivial defects