Software Requirements Specification

For

Metro Application

Version 3.0 approved

Prepared by

Name	Id	Email
Ahmed Ibrahim Mohamed	20170008	Ahibrahim.ai7@gmail.com
Ahmed Mostafa Elsayed	20170033	Kudo7474@gmail.com
Mohamed Sameh Omar	21070232	mohamedsameh99909@gmail.com
Mohamed Bakr Abdelhafez	20170224	mbakr7757@gmail.com
Mohamed Mohsen Abdelsallam	20170253	midomohsen11@gmail.com

Table of Contents

1. Introduction

1.1 Purpose

- *Solve the extremely crowding of metro stations problem with this large number of daily passengers, and its result of queue that each person become binding to stay. By organizing the process of booking metro tickets online.*

1.2 Intended Audience and Reading Suggestions

- *Users: the sections that are most pertinent are 1.1 and 2.2.*
- *Developers: all sections.*
- *Testers: the sections that are most pertinent are 2.2 and 5.*
- *Project Managers: all sections.*
- *Metro Organization: the sections that are most pertinent are 1.1, 1.3, 2.2 and 5.*

1.3 Product Scope

- Handle all the process of booking a ticket by the mobile app.
- Let the user can easily charge his balance using Fawry for example, or any other payment method.
- Handle the ticket validation process by using QR technology instead of normal tickets, by either machines in stations or other mobile application with an employee to read the code and then complete the process.
- Develop some features to make the application better and more usable.

2. Overall Description

2.1 Product Perspective

- This is a self-contained product.

2.2 Product Functions

2.2.1 The system shall allow users to register in the Application.

2.2.2 The system shall allow users to login in the Application.

2.2.3 The system shall allow users to buy one ticket or more at the same process.

2.2.4 The system shall allow users to determine the price of a ticket using source and destination.

2.2.5 The system shall allow users to determine the closest station from a specific destination and calculate estimated time between them.

2.2.6 The system shall allow users to determine estimated time for the entire trip (from Specific location to another including the time spent in subway).

2.2.7 The system shall allow users to get full directions (path) to go from one subway to another.

2.2.8 The system shall allow users to be notified when his destination is coming closer.

2.2.9 The system shall allow users to apply for or renew a normal subscription.

2.2.10 The system shall allow users to add a subscription by reading Metro subscription card.

2.2.11 The system shall allow users to apply for or renew an Educational subscription after complete all the verification process.

2.2.12 There is a map showing all the subways lines.

2.2.13 The system shall allow Admins to add advertisement or announcement.

2.3 User Classes and Characteristic

- One User Class with three roles: Normal User – Student.
- The normal user should be able to do the following functions:
    - Buy one ticket or more at the same process.
    - Determine the price of a ticket using source and destination.
    - Determine the closest station from a specific destination and calculate estimated time between them.
    - Determine estimated time for the entire trip (from Specific location to another).
    - Get full directions (path) to go from one subway to another.
    - Notified when his destination is coming closer.
    - Apply for or renew a normal subscription.
    - Add a subscription by reading Metro subscription card.
    - A map showing all the subways lines.
    - View all trips he booked.
- The student should be able to do the following functions:
    - Normal user Functions.
    - Apply for or renew an Educational subscription after complete all the verification process.
- The admin should be able to do the following functions:
    - View trips for all users, a specific user, time, day and location.
    - Add advertisement or announcement.

2.4 Operating Environment

- *Mobile Application should work on Android and IOS operating systems.*

2.5 Facts

• An annual report on mobility from Ericsson states that smartphones are poised to reach 6.1 billion users by 2020.
• Gartner predicts that by the end of 2017, mobile apps will have been a downloaded excess of 268 billion times which will generate revenue somewhere over $77 billion.
• In a comprehensive report issued by the Egyptian Council of Ministers Information Centre in 2019, which includes numerous figures and details, the Centre indicated in its report published at Facebook, that the number of passengers served by the subway amounts to 3.5 million per day by

completing the implementation of the third line of the metro, as well as 1662 trips per day in 2019, compared to 1544 trips per day in 2014.

2.6 Assumptions

- Though Near Field Communication (NFC) first came to use in the mid-2000s, a far wider population is now familiar with the technology than ever before, and many are using NFC on a regular and frequent basis. And according to LearnBonds and eMarketer estimates there will be 69.4 million NFC mobile payment users by the end of 2020; that number will rise to 80.1 million users by 2023.

2.7 Dependencies

- Google Maps API.
- Payment API (ex: Fawry).

2.8 Constraints

2.8.1 Development Process and Team Constraints

- The system must be ready in 9 months between design, analytics, development and testing

2.8.2 Environmental and Technology Constraints

2.8.2.1 Software Constraints

- Our team has recommended that the System be constructed using
- Cross platform Flutter for reasons of portability and flexibility
- The customer's Phone should have an Android System that supports the NFC technology.

2.8.2.2 Hardware Constraints

- The System will need low memory of Ram, low size of Storage, so it’s suitable for most of Users.

3. System Features

3.1 Register

Description	• User is able to register in the Application. Each user should be identified by username or email. Each user should provide a password which will be used later to login.
Priority	• High Priority.
Stimulus/ Response Sequences	• User types username, email and password and register. • System validates the entered data if it's correct then add user into database
Functional Requirements	REQ-1: 2.2.1 The system shall allow users to register in the Application.

3.2 Login

Description	• User is able to login in the Application. Each user should be identified by username or email.
Priority	• High Priority.
Stimulus/ Response Sequences	• User types username or email and password and register. • System validates the entered data if it's correct then user login into application.
Functional Requirements	REQ-1: 2.2.2 The system shall allow users to login in the Application.

3.3 Buy Tickets

Description	• Each User can buy one ticket or more through his fawry wallet and subtract the price for it.
Priority	• High Priority.
Stimulus/ Response Sequences	• User selects the tickets he wants to buy and their prices. • System validates if his wallet credit is enough to buy the tickets.
Functional Requirements	REQ-1: 2.2.3 The system shall allow users to buy one ticket or more at the same process.

3.4 Ticket Price

Description	• Each User determine the price of a ticket by enter the source station and destination station.
Priority	• Medium Priority.
Stimulus/ Response Sequences	• User selects source station and destination station. • System determines the suitable ticket price for this trip.
Functional Requirements	REQ-1: 2.2.4 The system shall allow users to determine the price of a ticket using source and destination.

3.5 Subscription

Description	• User can apply for or renew a normal subscription or an Educational subscription after complete all the verification process.
Priority	• Medium Priority.
Stimulus/ Response Sequences	• User will enter the data • The data will be validated • Amount of money will be subtracted • User will have his subscription.
Functional Requirements	REQ-1: 2.2.9 The system shall allow users to apply for or renew a normal subscription. REQ-1: 2.2.10 The system shall allow users to add a subscription by reading Metro subscription card. REQ-1: 2.2.10 The system shall allow users to apply for or renew an Educational subscription after complete all the verification process.

3.6 Trip Time & Route

Description	• Each User can determine the closest station from a specific location and calculate estimated time between them or the estimated time for the entire trip (from Specific location to another including the time spent in subway). User also can get full directions (path) to go from one subway to another. User also can be notified when his destination is coming closer.

Priority	• Low Priority.
Stimulus/ Response Sequences	• User selects a specific location. • System determines the closest station from this location and calculates the estimated time between them. • User selects two locations (source and destination). • System determines the estimated time for entire trip.
Functional Requirements	REQ-1: 2.2.5 The system shall allow users to determine the closest station from a specific destination and calculate estimated time between them. REQ-2: 2.2.6 The system shall allow users to determine estimated time for the entire trip (from Specific location to another including the time spent in subway). REQ-3: 2.2.7 The system shall allow users to get full directions (path) to go from one subway to another. REQ-4: 2.2.8 The system shall allow users to be notified when his destination is coming closer.

4. Other Nonfunctional Requirements

4.1 Performance Requirements

Performance requirements define acceptable response times for system functionality. Response time of the Ticket Reservation System should be less than few seconds most of the time. Response time refers to the waiting time while the system accesses, queries and retrieves the information from the databases. The login information shall be verified within less seconds. Response time of the system will not take long time, almost in a few seconds. The system works 24 hours per day 7 days in a week. The passengers' information must save in the database in few minute after the end of registration.

Response time of the System should be minimum. The system should show no visible deterioration in response time as the number of users or reservation data increases. The system does not taking up too much space in memory to store system's data.

4.2 Security Requirements

This is vital in the design and expectation of any system. It should contain mechanism to protect the integrity and avoid changes or access by unauthorized users.

Consideration of the security of the system has a great advantage for this system, because the database should be secured from the unauthorized users. Only authorized user can get access to the database. To prevent from the unauthorized user, the user should have their username and password that help them to login to the system. Additionally, the users should have to take care of their own username and password. They should have to keep in a secret manner.
The System will use an authorized online wallet.

4.3 Software Quality Attributes

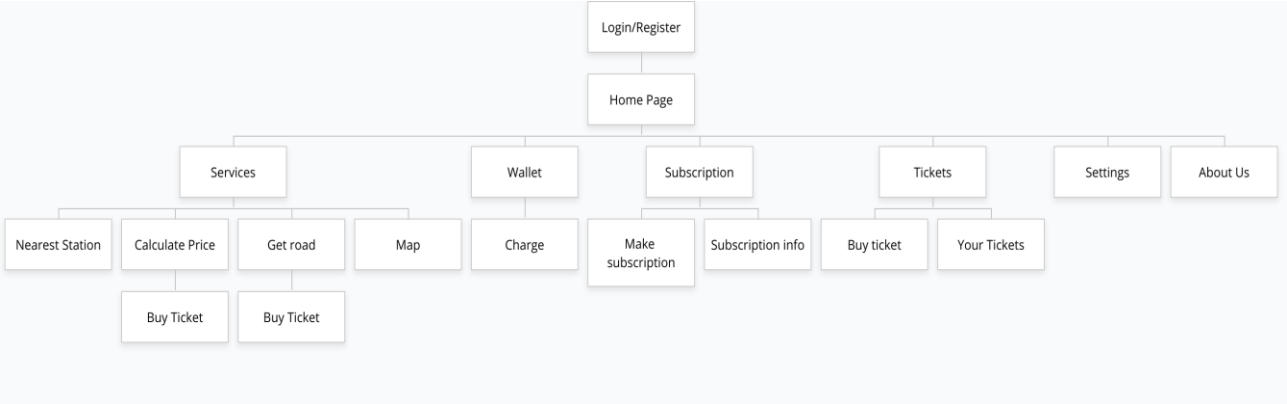
Should be easily maintainable, easy to upgrade and make adjustments as it is known, every system needs to be maintained and modified, so we will try to make the code easily readable and write the comment along with necessary to the codes.
The system will check user inputs to the system to handle error. It handles and show error by displaying the error message when the user enters invalid input.

Our system describes the logical characteristics of each interface between the system and the users. This may include any graphical user interface (GUI) standards or product family style guides, screen layout constraints, standard buttons and functions that will appear on every screen, error message display standards, and so on. So our system does these all functions in easy and efficient way. In other words the system is user interactive

5. Other Requirements

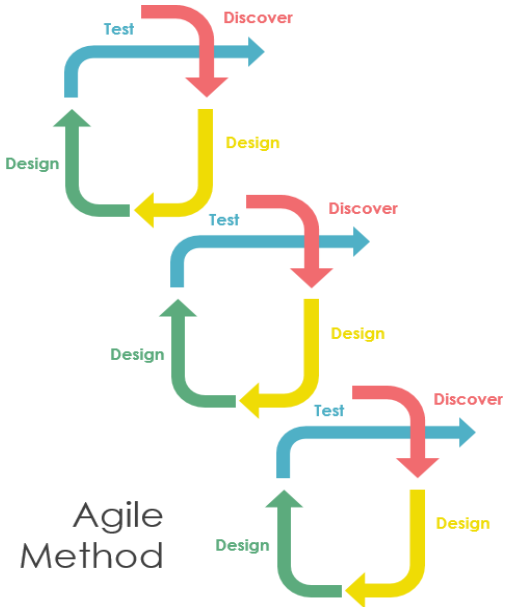
5.1 Site Map:

—

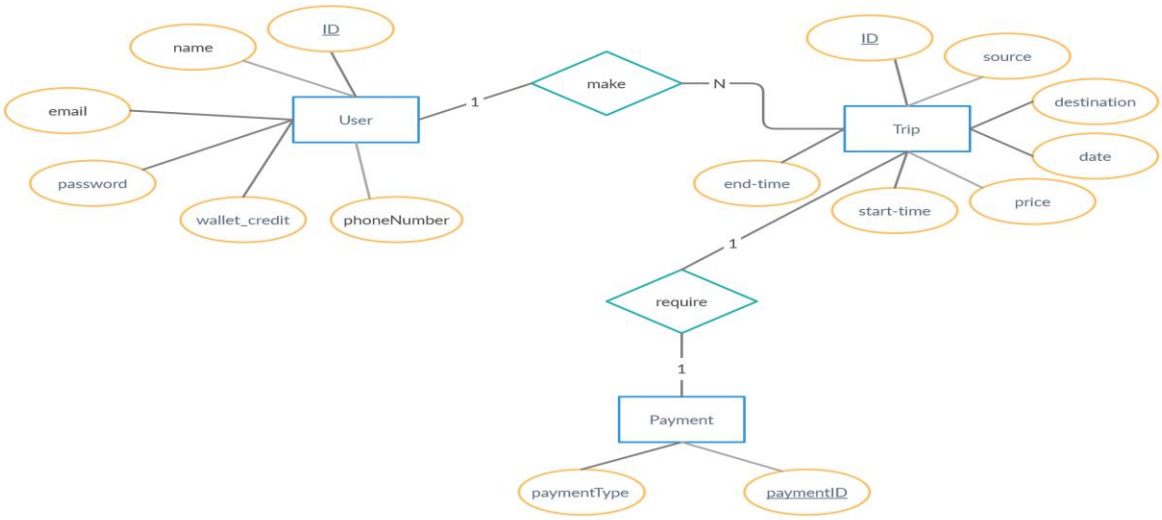


5.2 Process Model:

The used process model is agile method. Agile software development approach is typically operated in rapid and small cycles.



5.3 ERD


ID
name
email
password
wallet_credit
phoneNumber
User
1
make
N
ID
source
destination
date
price
start-time
end-time
Trip
1
require
1
Payment
paymentType
paymentID