
Software Requirements Specification

for

Tarmem APP

Version 1.0 approved

Prepared by ahmed hamed

RIMDEV Compnay

Friday, January 17, 2020

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction.....	1
1.1 Purpose	1
1.2 Document Conventions.....	1
1.3 Intended Audience and Reading Suggestions.....	1
1.4 Product Scope	1
1.5 References.....	1
2. Overall Description	1
2.1 Product Perspective	1
2.2 Product Functions	1
2.3 User Classes and Characteristics	2
2.4 Operating Environment.....	2
2.5 Design and Implementation Constraints.....	2
2.6 User Documentation	2
2.7 Assumptions and Dependencies	2
3. External Interface Requirements	3
3.1 User Interfaces	3
3.2 Hardware Interfaces	3
3.3 Software Interfaces	3
3.4 Communications Interfaces	4
4. System Features.....	4
4.1 System Feature 1	4
4.2 System Feature 2 (and so on).....	5
5. Other Nonfunctional Requirements.....	5
5.1 Performance Requirements.....	5
5.2 Safety Requirements	5
5.3 Security Requirements	5
5.4 Software Quality Attributes.....	5
5.5 Business Rules	5
6. Other Requirements	6
Appendix A: Glossary.....	6
Appendix B: Analysis Models.....	6
Appendix C: To Be Determined List.....	6

Revision History

Name	Date	Reason For Changes	Version
Ahmed Hamed	17/1/2020	System requirement	V1.0

1. Introduction

1.1 Purpose

- #1 Tarmem Services brings Home Services at Your Doorstep. All you have to do is, Place Request and relax. Tarmem Services is for Home and Professional Services.*
- #2 Repair and Maintenance - Plumbers, Electricians, Carpenters, Painters, Computer and Laptop Repair, Mobile Repair, Interior designer, Air Conditioner Repair, Washing Machine Repair, TV Repair and Other appliances Repair*
- #3 Make Payment Via Cash.*

1.2 Document Conventions

1.3 Intended Audience and Reading Suggestions

This document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers.

1.4 Product Scope

Our scope to make mobile and web application to make a home repair service and make a registration scope for workers.

1.5 References

This is the first document have a system requirement.

2. Overall Description

2.1 Product Perspective

This application interface between normal users and workers to make home repair service.

2.2 Product Functions

- Worker register to choose one of our services*
- Admin approve after complete company paper*

- *Client register with location based*
- *Client request a worker to make one of our service*
- *Request go to nearest worker or client can choose*
- *Worker show client location and route to it by google maps*
- *Worker add money to client wallet*
- *Client approve to pay service*
- *Worker wallet display worker percentage*
- *Company wallet have a remain*

2.3 User Classes and Characteristics

User classes

- *Admin of application*
- *Normal Users*
- *Workers*

2.4 Operating Environment

Environment

- *Web for admin*
- *Mobile for Clients and workers*

2.5 Design and Implementation Constraints

- *hardware limitations*
 - *8 GB RM And i7 Core for database server*
 - *8 GB RM And i7 Core for FTP server*
 - *16 GB RM And i7 Core for web-service server*
 - *16 GB RM And i7 Core for publish service server*
- *databases*
 - *MYSQL DB*
- *language requirements*
 - *Arabic*
 - *French*
 - *English*
- *Application servers*
 - *Apache for webservice*
 - *Nodejs server for web and mobile*

2.6 User Documentation

Document that will be delivered along with the software

- *user manuals*
- *on-line help*
- *admin tutorials*

2.7 Assumptions and Dependencies

SRS document have all requirement and other requirement will be another requirement with other fund.

3. External Interface Requirements

3.1 User Interfaces

- *Admin web UI*
 - *bootstrap UI*
 - *page for configuration of Application*
 - *pages to add all parameter of application*
 - *page to view Worker request*
 - *page to view all user request status*
 - *dashboard (count of app visiting users, count of user, Count of workers)*
- *User Web and mobile UI*
 - *First page choose service type*
 - *List of worker with sort by nearest*
 - *Page to write more details of order with date and time*
 - *Page for login or registration*
 - *Page to appear current request with status and confirm order after worker finish*
 - *Page to appear history request and status.*
 - *Page to appear wallet balance*
 - *Page to appear contact us*
 - *Page to appear about us*
 - *Page for setting (language)*
- *Worker Web and mobile UI*
 - *First Page for login or registration*
 - *List of orders with date and time and change status (ready, inway, reach, inprogress, finish)*
 - *Page to credit money to user wallet*
 - *Page to appear history request and status.*
 - *Page to appear wallet balance*

3.2 Hardware Interfaces

We have four server

- *database server*
 - *MYSQL sever for saving data and information*
- *FTP server*
 - *FTP for saving images and documents*
- *web-service server*
 - *apache server for deploy application web services (Spring framework restful webservices)*
- *publish service server*
 - *node JS with angular CLI for deploy web and mobile application*

3.3 Software Interfaces

- *Core application*
 - *Web services will be restful spring with JPA technology*
- *publish service server*
 - *use angular JS to make web modules*
 - *use flutter to make mobile application for IOS and Andriod*

3.4 Communications Interfaces

Communication standards that will be used, such as FTP and HTTP . if client have SSL certification will deploy in FTP and HTTP to be SFTP and HTTPS

4. System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

4.1 System Feature 1

<Don't really say "System Feature 1." State the feature name in just a few words.>

4.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

4.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

4.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use "TBD" as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:

REQ-2:

4.2 System Feature 2 (and so on)

5. Other Nonfunctional Requirements

5.1 Performance Requirements

<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>

5.2 Safety Requirements

We provide one year warranty to any issues appear to Customer.

5.3 Security Requirements

- *Login module will use encryption password*
- *Token expiration time to make session secure*
- *Encryption to any VIP data*

5.4 Software Quality Attributes

- *Adaptability*
- *Availability*
- *Correctness*
- *Flexibility*
- *Interoperability*
- *Maintainability*
- *Portability*
- *Reliability*
- *Reusability*
- *Robustness*
- *Testability*
- *usability*

5.5 Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

6. Other Requirements

<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>

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: Analysis Models

<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>

Appendix C: To Be Determined List

<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>