CMP

Master Clinic

Software Requirements Specification (SRS)
Version: 1.0

Revision History

Sl. No.	Prepared/ Modified by	E-mail	Version	Date	Approved by	Descriptions/ Remarks
1.	Abdelrahman Mohmed Aisha Mousa Hady Maher Reham Hamdy	am535150@gmail.com aishamousa632@yahoo.com hadymaher311@gmail.com rehamhy6@gmail.com	1.0	Mar, 11, 2018	-----------------	-----------------

Distribution list

Name	E-mail	Notes
Abdelrahman Mohamed Aisha Mousa Hady Maher Reham Hamdy	am535150@gmail.com aishamousa632@yahoo.com hadymaher311@gmail.com rehamhy6@gmail.com	

Master Clinic	CM-identifier: Mc _SR01 _v1.0
Software Requirements Specification	Date: 11/Mar/18

Table of Contents

Software Requirements Specification

1.   Introduction

Dental care is very important as teeth affects both health and appearance. However lots of people go to the dentist with an average ratio of 64% of a yearly visit for adults. Managing patients' paper and registering a dentist becomes really an annoying task. That's why we are creating **MASTER-CLINIC** which is a website that hopefully would help you register a dentist, manage your treatment paper and expenses and also help dentists in managing their patients' files and easily getting statistics and finding specific patient's data at any time.

1.1   Purpose

Master Clinic is a web-based application to help both doctors and patients. Patients would be able to make appointments, see his files, review payment (for more details see section 5 Requirements Specifications). doctors would be able manage their clinics, workers, nurses, inventory, patient files, get different statistics.

1.2   Scope

A complete system for connecting patients with their dentists which translates to a website with multiple users, every user would have their pages that'd give them their allowed functionalities. There would be three users: Admin (dentist), Nurse, Patient. Every user would have different functionality for example:

1. Admin (dentist): can see different statistics about his clinics, patient files, review his schedule, add prescriptions, add or remove a clinic, add or remove a nurse, etc.
2. Nurse: create patients accounts and files, confirm reservations, etc.
3. Patient: make a reservation, review his prescriptions, review his case and history, etc.

There would be an authentication system to handle these different users, a database to store users data. Primarily a layout of the users pages would be presented to the client as to have an insight of what the end product would look like. After that there would be an implementation of an authentication system to handle users and give them access to their pages. Implementation of basic functionality starts where work is distributed among team members equally and every member is to implement some functionalities. First delivery to the client is when basic functionalities are implemented. Feedback and any extra functionalities are to be added and remaining functions are distributed and implemented. Final version is delivered to the client after testing and debugging.

1.3   Definitions, Acronyms and Abbreviations

Worker: anyone working in the clinic and not a doctor or nurse.

Patient file: a file that contains all patient's prescriptions, photos and any other comments or data dentist noted about the patient like case description and progress.

Super admin: clinic owners

invoice: doesn't exist in reality however it's created virtually to keep track of patient payments

admin: in this project admin is the dentist himself

1.4   Overview

At the beginning of this document we discuss the average cost of any application with similar scope then we present a feasibility study that show why such a project would be a good choice for such a business and how it would achieve the targets and goals of the customers using the applications. The sections after it show the proposed system requirements with UML diagram and different operating scenarios.

2.   Cost Estimate

This section shows average cost of different parts of any web application with similar scope the numbers in table 2 are all from recent statistics and shall give a good insight about todays market and demands.

Item	Estimated Cost	
	US Dollars	L.E.
1. **Domain and Hosting**		
1.1. Hosting (per month)	10	176.12
1.2. Domain (per year)	48	845.37
Total	58	1021.49
2. Web design and Development		
1.1 Web design	120	2113.42
1.2 Development and implementation of system functionalities	230	4050.72
Total	350	6164.14
2. Deployment and Testing		
1.1 Unit testing	100	1761.18
1.2 deployment and system integration	250	4402.96
Total	350	6164.14
Total Estimated Cost	758	13349.77

Table 2: Cost Estimate

3.   Feasibility Study

Feasibility is a critical attribute of any system as it's what determines if the system really worth the effort and money or not. Giving a look at our system we find it's mainly makes it's easier to connect between the client (the patient in this specific case) and the service provider (the dentist in our specific case). It also helped organizing the organization (clinic in this specific case) and reducing amount of human effort which may result in reduction of human resources. So eventually the system did help the organization to reduce it's expenses on human resources and also made it easier to deal with the client and deliver a better service after all. In short the system did achieve the organization goals. The system is website which is great since most of today applications are web based applications which make much easier to find developers with decent price to build the website from scratch. Even the maintainability and integrity of web applications is much better than any other software (it could be used on any device that connects to the Internet like mobile phones, desktops, etc).

4.   Requirements Specifications

After observing and conducting meetings with the client. It was proposed that the system

would a web application that helps a dentist managing a group of clinics with different nurses and workers. The system was also required to manage patients' files and help patients reserving appointments. And finally, it was proposed that the clinics may contain some products that may be sailed to the patients. From that perspective we came out with these simple requirements that we hope cover the system completely and precisely.

4.1   User Requirements (User Stories)

4.1.1   register a new nurse in the clinic or remove an existing nurse from the clinic (when new nurse is hired doctor would create her an account to work with and if a nurse is leaving then her account is no longer needed and thus deleted from system).

Identifier:US01

4.1.2   Register a new clinic or remove an existing clinic.

Identifier:US02

4.1.3   Register a new patient or remove an existing patient.

Identifier:US03

4.1.4   Register a new admin (when a new doctor is using the clinic may be a co-doctor then he would be registered as admin but with less functionality)

Identifier:US04

4.1.5   Remove an existing admin (when a doctor was working temporarily in the clinic then he leaves his admin account is no longer useful then only a super admin could delete it.

Identifier:US05

4.1.6   Creating patient file or removing an existing file (a new file is created for every patient in his first appointment. When the patient no longer needs the treatment then his file is useless and could be deleted)

Identifier:US06

4.1.7   Editing and updating patient files (after an appointment the doctor would note the progress of the patient may be give him a prescription or something like that then is must be noted in his file to make sure that both he and the doctor are UpToDate with his case progress)

Identifier:US07

4.1.8   Add or remove inventory items (an inventory have a number of items then any item could be added or removed)

Identifier:US08

4.1.9   Create invoices (when patient had an appointment they pay after it and an invoice is created to save the amount of money they paid for future statistics)

Identifier:US09

4.1.10   Review inventory statistics (once in a while the admin would need to see statistics of the inventory to know how the sails is going, numbers of existing items, what the highest sailed item, etc.)

Identifier:US10

4.1.11   Searching over patients using their data like: mobile number, name, email or case (this would help in a lot of cases like: finding patient file easily and quickly)

Identifier:US11

4.1.12   Reviewing patients' statistics (once in a while the doctor would want to know statistics about his patients like: how many patients uses different services? how many patients that are treated already, etc.)
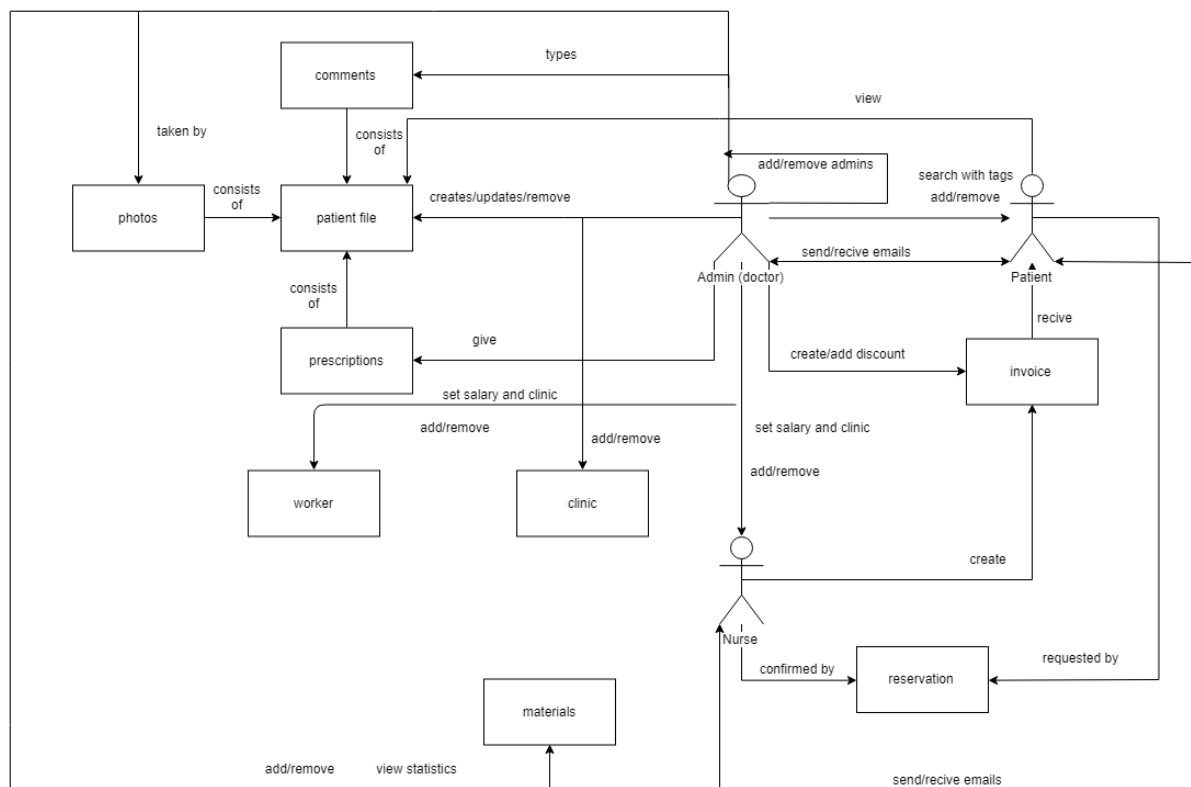
Identifier:US12

4.1.13   Setting nurses and workers' salaries (doctor would want to set his stuff salaries taking into perspective that every worker or nurse take different money than the others or the doctor simply wants to change the salary of a worker or nurse)

Identifier:US13

4.1.14 Setting discounts on some invoices (could be used as some sort of marketing as doctor makes discount to attract more patients)
Identifier:US14

4.1.15 Setting alarm notification on inventory items as if they reached a specific value (helps as a reminder if some inventory item is about to finish then doctor would need to buy more of it)
Identifier:US15

4.1.16 Sending emails to patients (as some times doctor or the nurse would want to inform the patient of may be a discount or something like that)
Identifier:US16

4.1.17 Confirming reservations (nurse would confirm patients reservations as to make sure patients don't waste their time waiting for a free time to visit the doctor the nurse would see patients' reservation request they confirm the patient with the most suitable time to visit the doctor)
Identifier:US17

4.1.18 Request a reservation (patient would make a reservation then it would be pending until the nurse confirm it with the most suitable appointment)
Identifier:US18

4.1.19 Viewing invoices and their total summation to help the user realize his expenses
Identifier:US19

4.1.20 Patient sending mails to doctor or nurse could be useful in case if there is a critical matter that needs immediate consultation
Identifier:US20

4.2   System Requirements (Use Cases)

4.2.1 Add new clinics

Name: add clinic

Identifier: UC1
Preconditions
- Successful login for admin

Basic Course
- Fill a form of clinic data
- Post it to backend components and store in database

Post conditions
- System can then register patients and nurses and appointments.

Map to:
- US02

4.2.2 Remove clinics

Name: remove clinic

Identifier: UC2
Preconditions
- Successful login for admin

Basic Course
- Delete request from admin view to remove a clinic and remove it's nurses and all data of this clinic

Post conditions
- All data of the clinic well be deleted

Map to:
- US02

4.2.3 Add new category

Name: add category

Identifier: UC3
Preconditions
- Successful login for admin

Basic Course
- Fill a form of category data
- Post it to backend components and store in database

Post conditions
- System can then add new material to these categories.

Map to:
- US08

4.2.4 Remove category

Name: remove category

Identifier: UC4
Preconditions
- Successful login for admin

Basic Course
- Delete request from admin view to remove a category and remove it's all material data

Post conditions
- System can then add new material to these categories.

Map to:
- US08

4.2.5 Admin login

Name: admin login

Identifier: UC5
Preconditions
- Admin is added once to database

Basic Course
- Fill a form for admin email and password
- Get the admin data from database and check if he is authenticated
- If the admin authenticated then login or fail

Post conditions
- Admin login to his dashboard and can't go back to login page

Map to:
- US04

4.2.6 Admin register

Name: admin register

Identifier: UC6
Preconditions
- There is at least one super admin

Basic Course
- Fill a form of admin data from super admin view

- Add the new admin to the database

Post conditions
- The new admin can login to his dashboard

Map to:
- US04

4.2.7 Add new file to specific patient

Name: add patient file

Identifier: UC7
Preconditions
- any admin is logged in
- patient must be in the database

Basic Course
- Fill a form for the file of the patient
- Post it to database tables

Post conditions
- Admin can follow the patient status
- User can see his file on his view

Map to:
- US06

4.2.8 Nurse register

Name: nurse register

Identifier: UC8
Preconditions
- There is at least one admin is login
- There is at least one clinic

Basic Course
- Fill a form of nurse data from admin view
- Add the new nurse to the database

Post conditions
- The new nurse can login to his dashboard

Map to:
- US01

4.2.9 Patient register

Name: patient register

Identifier: UC9

Preconditions
- There is at least one admin or one nurse is login
- There is at least one clinic

Basic Course
- Fill a form of patient data from admin or nurse view
- Add the new patient to the database

Post conditions
- The new patient can login to his dashboard and add new reservation

Map to:
- US03

4.2.10 Remove Patient

Name: Remove patient

Identifier: UC10
Preconditions
- There is at least one admin or one nurse is login
- There is at least one clinic

Basic Course
- Delete request from admin or nurse to delete the patient

Post conditions
- All data for this patient will be removed (files, receipts, etc.)

Map to:
- US03

4.2.11 Update Patient's file

Name: update patient's file

Identifier: UC11
Preconditions
- There is at least one admin
- There is at least one clinic
- The file must be exists

Basic Course
- Fill a form of the new status of the patient
- Post it to database and cascade to the old file

Post conditions
- Admin can follow the patient status
- User can see his file on his view
- This data is cascading with the old data

Map to:
- US07

4.2.12 Nurse login

Name: nurse login

Identifier: UC12
Preconditions
- Nurse exists in database.

Basic Course
- Fill a form for nurse email and password.
- Get the nurse's data from database and check if she is authenticated.
- If the nurse authenticated then login or fail.

Post conditions
- Nurse login to her dashboard and can't go back to login page.

Map to:
- US01

4.2.13 Set discount

Name: set discount

Identifier: UC13
Preconditions
- Successful login for admin
- Receipts are once added to the database.

Basic Course
- Get access to patients' receipts
- Set the value of discount in those receipts.

Post conditions
- System can then subtract this discount from the total invoice of some patients.

Map to:
- US14

4.2.14 Set alarm

Name: set alarm

Identifier: UC14
Preconditions
- Successful login for admin.
- There are clinics in the database

Basic Course
- Get access to the number of items in inventory of each clinic.

Post conditions
- System can then subtract this discount from the total invoice of some patients.

Map to:
- US14

4.3   Non-functional Requirements

This section contains non-functional requirements that consists of three main parts External requirements, organizational requirements, product requirements. However, since there is no organizational requirements provided till now we will mention just external and product requirements

4.3.1   Patient files should be hidden from all users excepts himself and the doctor to ensure privacy

4.3.2   System shall be available to all clinics during normal working hours (Sat-Thu, 8:30 AM-9:00PM)

4.3.3   Many nurses shall be able to login the application at the same time even if they are in the same clinic

5.   High level plan

Iteration 1:

User Stories	Estimated Time
1. register and login new admin	2days
2. register and login new nurse	2days
3. register and login new patient	2days

Total Time	**6days**

Iteration 2:

User Stories	**Estimated Time**
1.create new category and materials	2days
2.create and update file for patient	5days
3. create new invoices	3days
4. create statistics	3days
Total Time	**13days**

Iteration 3:

User Stories	**Estimated Time**
1.testing the website	3days
2.	
Total Time	**3days**

<div align="center">Table 1: High Level Plan</div>

6.   Supporting Information

[The supporting information makes the SRS easier to use. It includes: a) Table of contents, b) Index, c) Appendices. These may include use-case storyboards or user-interface prototypes. When appendices are included, the SRS should explicitly state whether or not the appendices are to be considered part of the requirements.]

Appendix A

Traceability Matrix

RID	US1.1	US1.2							....
UC1.1		⚭							
		⚭							
			⚭						
NF1.1									
NF1.2									

...									⚘

Appendix B

Dependability Matrix

RID	US1.1	US1.2							....
US1.1		⚘							
US1.2									
		⚘							