Project Report:

Dental Clinic Management System

Version 2.0 : Laboratory Exercise-03

Front-End Development:
Prince Ngema (754774)
Luyanda Makhoba (834867)
Back-End Development:
Takatso Molekane (569869)
Tholithemba Mngomezulu (1512124)

Contents

1 Introduction **3**

1.1 Purpose . . . 3
1.2 Problem Statement . . . 3
1.3 Project Objectives . . . 4
1.4 Stakeholders . . . 4
1.5 Scope . . . 5
1.5.1 Software Benefits and Objectives . . . 5
1.6 Definitions,Acronyms and Abbreviations . . . 6
1.7 References . . . 6
1.8 Project Overview . . . 7
1.8.1 Existing System . . . 7
1.8.2 Proposed System . . . 8

2 Software Requirements Specifications **8**

2.1 Overall Description . . . 8
2.1.1 Product perspective . . . 8
2.2 Functionality . . . 8
2.3 Usability . . . 8

3 Project Design and Architecture **10**

3.0.1 Architecture . . . 10
3.0.2 Implemented Database Tables . . . 11
3.0.3 Entity Relationship Diagram . . . 11
3.0.4 Software Tools . . . 11
3.0.5 Hardware Requirements . . . 11
3.1 Product functions . . . 13
3.2 Business Rules . . . 13
3.2.1 Use Cases . . . 15
3.3 Fully Dressed Use Cases . . . 20
3.3.1 Create patient use case . . . 20
3.3.2 Create appointment use case . . . 21

3.3.3 Use Case Diagram 22
3.4 Project Constraints 22

4 Agile Approach : SCRUM **23**
4.1 Scrum Roles 23

5 Agile Approach:SCRUM **23**
5.1 Scrum Roles 23
5.2 Scrum Artifects 23
5.2.1 User Stories 23
5.2.2 Product Backlog 25
5.2.3 Sprint Backlog 25
5.3 Sprint planning documents 29

6 Module Descriptions and Demonstrations **31**
6.1 Sceenshots 31

7 System development review method **37**
7.1 Sprint retrospective 37

8 System Testing **37**
8.1 Unit Tests 37

1 Introduction

1.1 Purpose

This document serves to describe the processes undertaken in the inception of the Dental Clinic Management system software project. The purpose of this document is to provide a detailed description of the DCMS ,a web application.It will give in detail the purpose of the system, features of the system and the constraints under which the system will operate. An outline of what the software aims to do at each stage, ie problem to be solved.

1.2 Problem Statement

Managing a dental clinic may be cumbersome at times, the paperwork that the receptionist have to do and the time patients have to spend waiting in queue is excessive. DCMS is system that will remedy this situation by allowing multiple patients at the same time

Hard copy files stored in cabinnets pose a security threat since it is possible for unauthorized personel to gain access because of negligence. The system will allow for the use of username and passwords, a secure measure that ensures that only permited users can see and do certain tasks. Where in the contrary, files can easily fall into the wrong hands, be tampered with or lost.

Human error in the collection and capturing of all data occures where patients either fill in their details incorrectly or the receptionist captures the data wrongly. DCMS will allow for data validation to occur, where the user will be alerted immedeiately if any data is incorrect or missing,ensuring the data is consistent in the database.

Paper files are hard to back up, the database storing capabilities adopted by DCMS will offer the ability to back up all data.

The automation of calculations and instantaneos syncing of events will allow for a well intergrated clinic with real time updates and time saving processing.

1.3 Project Objectives

The software is aimed at replacing manual paper systems that currently exists at a dental clinic.Users will remotely have access to relevant services based on requirements.The project objectives are :

- reduce the paper work the receptionist have to do on daily basis
- cut the amount of waiting time in queues
- ensure and protect patient's privacy
- reduce human error in capturing data
- reduce paper work for doctors

1.4 Stakeholders

Anyone that is influenced by or influences a project is a stakeholder. There are two types of stakeholders , internal and external stakeholders.

- **External**
  - Patients
    Patients are able to make appointments and view their bill
  - Dentist
    Dentists can login ,view and set their own schedule of appointments. Write out a prescription for a patient and view a patient's profile(medical record).
  - Receptionist
    receptionist logs in with their username and password, views and manages appointments, performs day open and close activities. He also sends reports to admin and help with registering those patients who that are having problems with registering.
  - Admin
    The administrator has the authority to add or remove a doctors

and receptionist.He grants permission to receptionist and dentists the authority to view and generates report.He also has the authority to add or delete patients from system. He also manages the system

- **Internal**
  - Scrum team
    responsible for developing the software
  - Product owner
    someone in charge of the entire project
  - Scrum master
    The link between the scrum team and the product owner(project manager)
  - Equipment suppliers
    They supply the hardware needed for the operation of the system.(i.e) Computers

1.5 Scope

DCMS (Dental Clinic Management System) is a web application that provides support for managing the services of a small dental clinic.

1.5.1 Software Benefits and Objectives

The software is aimed at replacing manual paper systems that currently exists at a dental clinic. Users will remotely have access to relevant services based on requirements. Having a digital filing system will reduce human error by having text validations before data is captured. Having database will allow for backups.

1.6 Definitions,Acronyms and Abbreviations

Term	Definition
DCMS	A Dental Clinic Management System application
User	Anyone who will be interacting directly with the system..
Netbeans	an integrated development environment for java
Java	A general-purpose computer-programming language that is concurrent, class-based,object-oriented
PHP	Hypertext Preprocessor is a server-side scripting language designed for web development.
Json	JavaScript Object Notation is an open-standard file format that uses human readable text to transmit data objects consisting of attribute-value pairs and array data types

1.7 References

- IEEE Recommended Practice for Software Requirements Specifications
- https://www.bmc.com/blogs/software-requirements-specification-how-to-write-srs-with-examples/ (Accessed Aug 2018)
- Zainab Murtadha- Dentist Web Based Patient Information System and Services in Cloud
- Virtual Medical Home SRS-Bapuju Institute
- https://krazytech.com/projects

54

1.8 Project Overview

Front End tasks: This involves the making of User Interfaces. These are the screens that the users will be seeing when using the system.

- Create Patient(Input will be patient details)
- Log in(Username and Password)
- Create Appointment(PatientId and Date/Time)
- Create Bill(PatientID, DoctorID and Consultation Details)
- View Schedule(DoctorID and Date/Time)
- View Bill(PatientID)

Back End tasks:

- Create Database with table and entities as listed in ERD
- Use back-end frameworks to build server-side software. PHP and JSON
- Cloud computing integration-Allowing Database to be accessed remotely.

1.8.1 Existing System

The present system is manual based. It involves paper work in the form of mantaining files, making appointments and billing.The manually based system has the following disadvantages:

- it is a limited system.
- looking for a patient's file may take a long time
- patients have to queue to make an appointment

- There is no backup files.
- files are prone to damage.
- editing file problems. storage space may be limited.
- Patient's personal information is not protected, it can be accessed by anyone.

1.8.2 Proposed System

DCMS is an automated system that can be accessed via the internet.It has the following advantages.

- Easy to store and search for files.
- Patients can make appointments online and avoid long queues.
- Each patient has a profile that can only be accessed by authorized users i.e(doctor or receptionist).
- The system can be accessed remotely.

2 Software Requirements Specifications

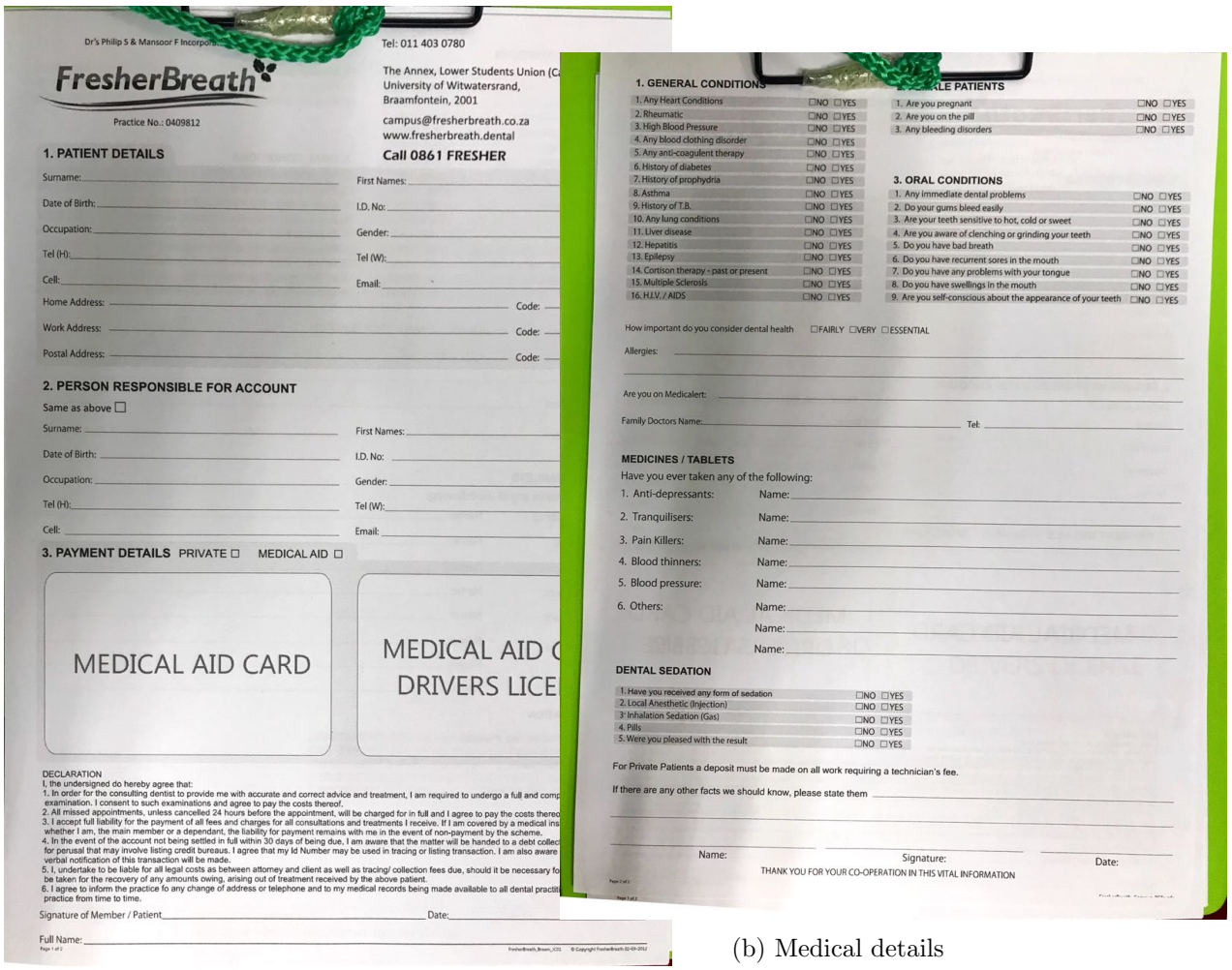
2.1 Overall Description

Based on

2.1.1 Product perspective

2.2 Functionality

2.3 Usability



(a) Patient personal details

(b) Medical details

Figure 1: Paper forms to be replaced

3 Project Design and Architecture

3.0.1 Architecture

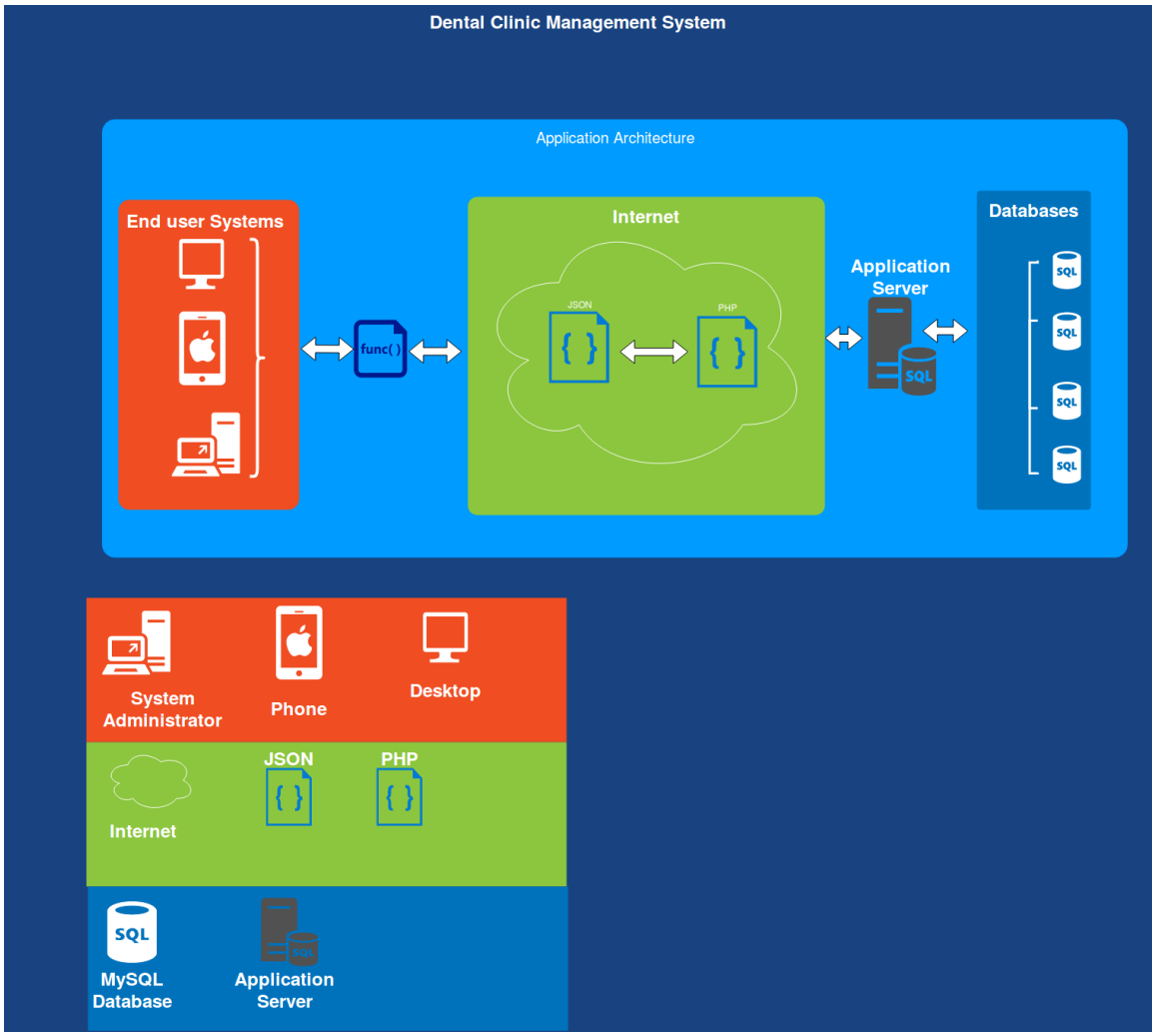


Figure 2: architecture

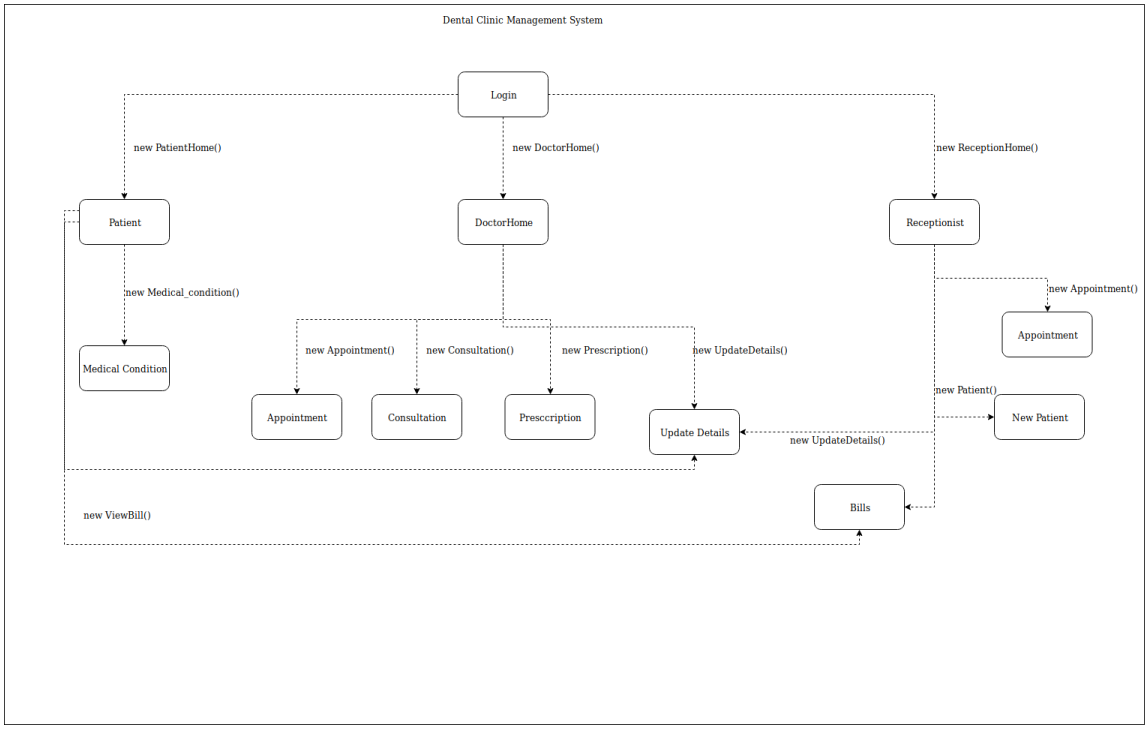


Figure 3: DCMS login

3.0.2 Implemented Database Tables

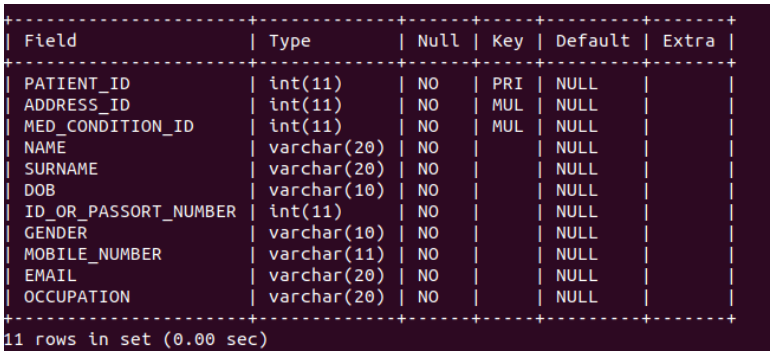
3.0.3 Entity Relationship Diagram

3.0.4 Software Tools

- Database Server: Microsoft SQL Server
- Client: Any web browser
- Programming Language:Java
- Development Tools:Netbeans IDE 8.2

3.0.5 Hardware Requirements

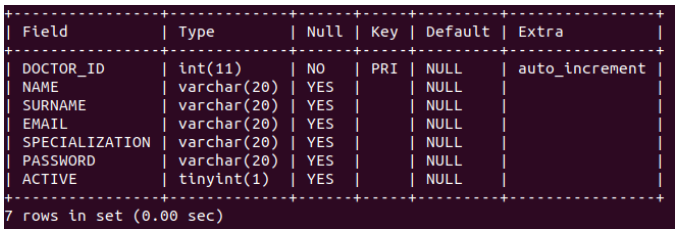
The supported Operating Systems:



Field	Type	Null	Key	Default	Extra
PATIENT_ID	int(11)	NO	PRI	NULL	
ADDRESS_ID	int(11)	NO	MUL	NULL	
MED_CONDITION_ID	int(11)	NO	MUL	NULL	
NAME	varchar(20)	NO		NULL	
SURNAME	varchar(20)	NO		NULL	
DOB	varchar(10)	NO		NULL	
ID_OR_PASSORT_NUMBER	int(11)	NO		NULL	
GENDER	varchar(10)	NO		NULL	
MOBILE_NUMBER	varchar(11)	NO		NULL	
EMAIL	varchar(20)	NO		NULL	
OCCUPATION	varchar(20)	NO		NULL	

11 rows in set (0.00 sec)

Figure 4: Patient Table



Field	Type	Null	Key	Default	Extra
DOCTOR_ID	int(11)	NO	PRI	NULL	auto_increment
NAME	varchar(20)	YES		NULL	
SURNAME	varchar(20)	YES		NULL	
EMAIL	varchar(20)	YES		NULL	
SPECIALIZATION	varchar(20)	YES		NULL	
PASSWORD	varchar(20)	YES		NULL	
ACTIVE	tinyint(1)	YES		NULL	

7 rows in set (0.00 sec)

Figure 5: Doctor Table

- **Microsoft Windows Vista SP1/Windows 7 Professional:**
  - Processor: 800MHz Intel Pentium III or equivalent
  - Memory: 512 MB
  - Disk space: 750 MB of free disk space
- **Ubuntu 9.10:**
  - Processor: 800MHz Intel Pentium III or equivalent
  - Memory: 512 MB
  - Disk space: 650 MB of free disk space
- **Macintosh OS X 10.7 Intel:**
  - Processor: Dual-Core Intel
  - Memory: 2 GB

Field	Type	Null	Key	Default	Extra
APPOINTMENT_ID	int(11)	NO	PRI	NULL	auto_increment
PATIENT_ID	int(11)	NO	MUL	NULL	
DOCTOR_ID	int(11)	NO	MUL	NULL	
RECEPTIONIST_ID	int(11)	NO	MUL	NULL	
DATE	varchar(10)	NO		NULL	
CHECKIN	varchar(10)	NO		NULL	
CHECKOUT	varchar(10)	NO		NULL	

7 rows in set (0.00 sec)

Figure 6: Doctor Table

  - Disk space: 650 MB of free disk space

- **Smartphone Requirements:**
  - Android running OS 4.0+
  - iPhone running iOS 8+
  - Windows Phone 8.1+

3.1 Product functions

DCMS will enable patients to book or make appointment and the output will the be date and time in which it is inline with the Doctors schedule. System will also provide a clear schedule which allows patients to see which Doctor is available at a particular slot. Who ever will be using the system has to go through registration first if he/she is first time user or login by providing username and password to access the DCMS.The system allows patients to request their bill and the patient can view or print the through system.

3.2 Business Rules

- Before a user can log in, they are required to be an existing user on the System. Existing users access the system (log in) using username and password.

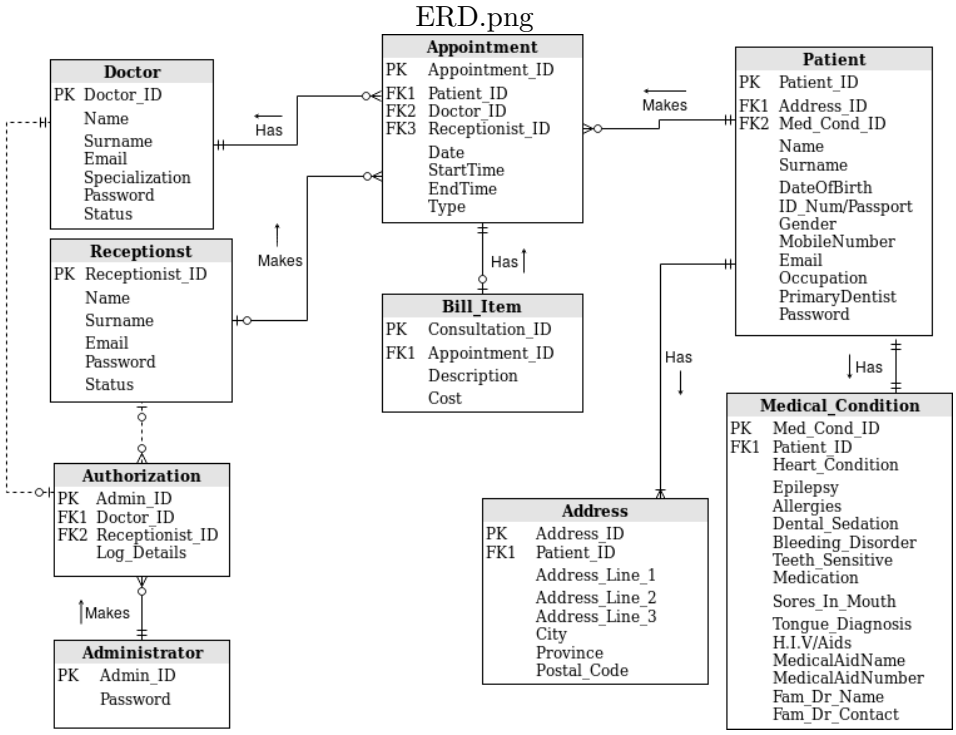


Figure 7: DCMS-ERD

  - New Dentists and Receptionist's require an Administrators authorization to be registered on the system.
  - A new patient is required to enter their personal and medical details.

- An Appointment must be booked by the patient. They have the options of doing so telephonically(Where the receptionist will be the one capturing the appointment) or engaging directly with the system. Booking of an appointment requires viewing the relevant dentist's schedule to identify available slots.

- A Dentist can view their schedule. This means viewing all the appointments that have been booked for the doctor and displayed as of their requirement either Daily,Weekly or Monthly schedule calendar

view.

- A consultation is created by a dentist. This follows the arrival of a patient for their appointment and discussions or dental procedures are conducted and recorded. A consultation can also be recorded for a patients failure to arrive for an appointment without cancelling. This consultation type is labelled as missed appointment.

- Generating Bill follows a consultation, this is where all the costs of the medical procedure are recorded. This may also include the recording of a missed appointment charge.

- Authorization is done by an administrator. This is required when new a receptionist or dentist is created. Similarly so when it will be updated or deleted.

3.2.1 Use Cases

Actor	**Description**
Receptionist	May assist patient with registration and booking, should they require assistance.
Administrator	Administrator is responsible for Doctors registration and other issues that directly related to the system like update or archive if necessary.
Patient	Patient may directly interact with the system during registration or booking process, depending on the patient's level of of computer literacy
Doctor	May set appointment with the patient, depending on patient's problem

Use Case	Description	Related Use case and Relationships
Create Patient	Patient or the Receptionist will interact with this use case. Step involved in this use case is entering demographic data.	
Read Patient	This use case will be used when accessing a patient's data.This includes when making appointment bookings and generating bills	Invoked by the Update Patient use case. <<include>> relationship.
Update Patient	The Receptionist or Patient will mainly interact with this use case. It will be accessed to update a Patient's demographic data	This use case invokes the Read Patient use case. <<include>> relationship
Create Administrator	An Administrator will interact with this use case. In order for Administrator to have an access to the system, an already existing Administrator should capture relevant data of new Administrator	
Create Appointment	The Patient,Receptionist or Doctor will interact with this use case. This use case will be triggered when a user wants to make an appointment.	This use case invokes read doctor and read patient

Read Administrator	The Administrator will interact with this use case. It will be triggered when Administrator request to view Administrator's profile.	This use case invokes the Update Administrator use case. <<include>> relationship.
Update administrator	An Administrator will interact with this use case. It will be triggered when there is a change in the demographic data of the Administrator.	This use case invokes the Read Administrator use case. <<include>> relationship.
Archive Administrator	Administrator will interact with this use case. it will be triggered by the other Administrator to archive an Administrator who no longer has an access to the system due to end employment contract or other reasons.	
Create Doctor	An Administrator will interact with this use case. It will capture Doctor's demographic data.	
Read Doctor	This use case is used when a doctors profile will need to be accessed. This will include when booking appointments, recording consultations and generating bill. It will be triggered when a user requests to view Doctor's details	

Create Bill	The Doctor will interact with this use case. This Involves capturing all charges of operations done on a patient during a consultation.	This use case invokes the Read Doctor,Read Patient use case. <<include>> relationship
Read Bill	The Doctor,Patient or Receptionist will interact with this use case. This Involves viewing and existing bill.	This use case invokes the Read Doctor,Read Patient use case. <<include>> relationship
Update Doctor	Administrator will interact with this use case. It will be triggered when an Administrator wants to modify Doctor's details	This use case invokes the Read Doctor use case. <<include>> relationship
Archive Doctor	Administrator will interact with this use case. It will be triggered when the Doctor no longer granted access to the system due to end of employment contract or other reason	
Generate Report	The Project Owner will interact with this use case. It will be accessed when the Project Owner wants to assess the effectiveness of the system.	

3.3 Fully Dressed Use Cases

3.3.1 Create patient use case

Use case name:	Create Patient
Scope:	Dental Clinical Management System for better health.
Triggering Event:	User request to create patient.
Brief description:	user request to create a new Patient profile. Either the Patient themselves via mobile phone, desktop, self-service terminal or Receptionist on behalf of the Patient.A form is displayed and prompt for the completion of all relevant data, including: The Patient's first-name, lastname, ID number, date of birth and email(if applicable). A prompt to confirm and save the profile is displayed. The user can double-check the entered data and confirm the creation of the profile. The profile is then created by creating a new record in the Patient table in the data store. Login details are generated and sent to the patient.
Actor(s):	Patient (Primary), Receptionist (Primary)
Related use cases:	N/A
Stakeholders and interests:	1. **Patient** - wants all their demographic data (first-name, lastname, ID number, date of birth, mobile number and email address(optional)) to be accurately captured to ensure the completion of their profiles. 2. **Receptionist** - wants to accurately capture Patient's demographic data (firstname, lastname, ID number, date of birth, mobile number and email address(optional)) on behalf of a computer illiterate Patient.
Pre-condition:	N/A
Post-condition:	1. Created Patient's profile recorded in the Patient's data store. 2. Login details are generated and sent to the Patient.

3.3.2 Create appointment use case

Use case name:	Create Appointment
Scope:	Dental Clinical Management System for better health.
Triggering event:	user request to create Appointment.
Brief description:	user request to create new Appointment.This involves Doctor's schedule where patient can select date and time available in the slot. Receptionist may also create appointment on behalf of patient. in case of emergency or serious problem depending on the condition of a patient a Doctor may also create appointment. Patient ID must be read first then time and date has to be selected on the Doctor's schedule and after successful booking confirmation message has to be generated and sent to the patient via email.
Actor(s):	Patient (primary), Doctor (primary), Receptionist (primary).
Related use cases:	Read Patient (includes), create consultation (extends).
Stakeholders and interests:	**Patient**- wants to make sure that appointment is done accordingly. 2. **Receptionist**- wants to accurately make appointment on behalf of those patients who have lack of computer skills. **Doctor**- wants to make appointment that is urgently and need serious attention.
Pre-condition	Patient must exist in the database.
Post-condition	Confirmation message must be send via email.

3.3.3 Use Case Diagram

Case Diagram.png

Figure 8: DCMS-ERD

3.4 Project Constraints

- DCMS must run on any platform that supports Java.
- Data captured should be stored on a cloud database.

- The user needs to be connected to the internet.

4 Agile Approach : SCRUM

4.1 Scrum Roles

- Product owner - Represents the customer/users. He Provides the specifications or requirements of the product, along with their priorities. This prioritized list of features is the product backlog.
- Scrum master - Enacts scrum values and practices. They Remove impediments, which are the obstacles that disrupt progress.
- Scrum team -perform analysis, design, program, test, document, and so forth

5 Agile Approach:SCRUM

5.1 Scrum Roles

- Product owner - Represents the customer/users. He Provides the specifications or requirements of the product, along with their priorities. This prioritized list of features is the product backlog.
- Scrum master - Enacts scrum values and practices. They Remove impediments, which are the obstacles that disrupt progress.
- Scrum team -perform analysis, design, program, test, document, and so forth

5.2 Scrum Artifects

5.2.1 User Stories

Patient

- As a patient, I want to be able to register on the system, so that I can have credentials to use to access the system
- As a patient, I want to be able to log in the system, so that I can access my portal on the system
- As a Patient, I want to be able to book an appointment, so that I can have a time reserved for me
- As a Patient, I want to be able to view my appointments, so that I can stay informed of the time and date.
- As a Patient, I want to be able cancel an appointment, so that I can change it's details without being charged a missed appintment fee.
- As a Patient, I want to be able view my bill, so that I can know all charges I have been charged.

Dentist

- As a Dentist, I want to be able to register on the system, so that I can have credentials to use to access the system
- As a Dentist, I want to be able to log in the system, so that I can access my portal on the system
- As a Dentist, I want to be able to view my schedule , so that I can stay informed.
- As a Dentist, I want to be able create a Consultation/Bill, so tha I can record all conducted procedures.

Receptionist

- As a Receptionist, I want to be able to register on the system, so that I can have credentials to use to access the system
- As a Receptionist, I want to be able to log in the system, so that I can access my portal on the system

- As a Receptionist, I want to be able to book an appointment, so that I can have a time reserved for a requesting patient
- As a receptionist, i want to be able to cancel an appointment, so that cancelled appointments are shown as such

Administrator:

- As an Administrator, I want to be able to authorize the creation of a new Doctor/Receptionist so that I can be able to ensure all relevent users are legitimate.

5.2.2 Product Backlog

This is a list of prioritized features. The product backlog of DCMS is given below.

Priority Rank	Item	Description
1	Register	Capturing the data of a new user
2	Authorize new Dr/Receptionist	Administrator authorizes the creation of a new Dr/Receptionist
3	Log in	User access the system using username and password
4	Create Appointment	User needs to be able to create an appointment slot reservation
5	View Appointment	User needs to be able to view their upcoming appointments.
6	Create Bill	Generate an invoice
7	Veiw Bill	View an invoice
8	Create e-prescription	Dentist generates e-prescription for patient
9	View e-rescription	Patient view e-prescription made by dentist

Figure 9: Priority List

5.2.3 Sprint Backlog

During the first sprint plan meeting, the product backlog was used to develop sprint backlogs.The first sprint comprises of 4 tasks/items. The tasks and

their description are given below.

Item	Description	
Register	Front End	Back End
	Develop a screen that allows user to select user type to register, and input required data	Run SQL query to insert data into database and varify that the same user does not already exist
Authorize new Dr/Receptionist	The administrator should be able authorize new Drs or new receptionists. They should be presented with a list of all users awaiting authrization where they can drill down into the individual user before authorization.	The status of a new dr/receptionist is set to pending authorization. It is under this condition where the admin will make the decision to authorize or not, where the user status will be updated to either active(Approved) or rejected
Login	Any user should be able to select their user type and use their correct username and password to log in. Should either be incorrect, an appropriate error message should be shown	The user type selected indicates the table in the tdatabase where the username and password should be checked in.
Create Appointment	Develop a screen that allows user to create a new appointment. This is done be selecting the doctor and the time and date of the appointment	After the user has selected the doctor and date. The query then uses these 2 field to return the doctors schedule for the day so that appointments do not clash.

Figure 10: Sprint 1

Each task was estimated to take at most 10 hrs.The first sprints ran for 5 days.Daily scrum meetings were conducted to check the progress of each team member and to unblock any impediments.At the end of each sprint, sprint review meetings were conducted to test and demostrate the functionality of the product.A sprint burn down diagram which shows the progress of the first sprint is given below

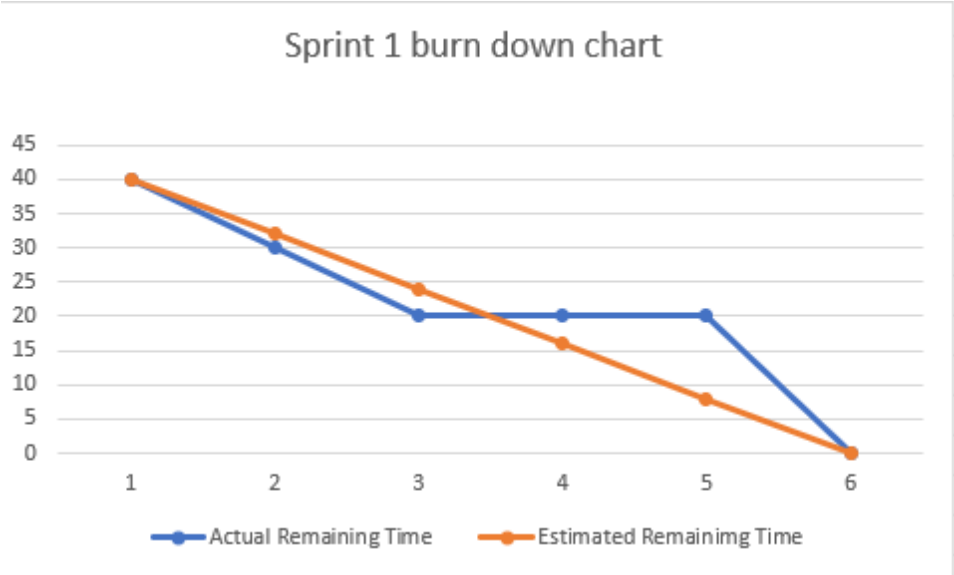


Figure 11: Sprint 1 burndown diagram

During the second sprint plan meeting, the product backlog was used to develop sprint backlogs.The second sprint comprises of 6 tasks/items. The tasks and their description are given below.

Task	Description	
View Appointment	A patient or dentist should be able view their schedule for a desired week.	Query involves patient or dentist ID with date or week of appointment.
Create Bill	A dentist creates a bill for a patient by selecting all the items the patient has been treated with.	A bill requires the doctor and patient ID to be created. It also includes all the items and costs of the procedures conducted.
View bill	A patient should be able to view their bill after a consultation has taken place showing the total costs of procedures	llPatient ID and date is used to view a patients bill
Create e-prescription	Develop a screen that allows user to create a new appointment. This is done be selecting the doctor and the time and date of the appointment	After the user has selected the doctor and date. The query then uses these 2 field to return the doctors schedule for the day so that appointments do not clash.
View e-prescription	A patient or dentist should be able view their schedule for a desired week.	Query involves patient or dentist ID with date or week of appointment.

Figure 12: Sprint 2

A second sprint of 5 tasks was formed from the product backlog.Each tasks was estimated to take at most 10 hrs and the sprint ran for 9 days.The tasks and their description are given below.A sprint burn down diagram which shows the progress of the second sprint is given below

5.3 Sprint planning documents

Sprint is timeboxed incremental iterations, each aims to produce a potentially shippable increment (PSI).
Product backlog(priority list)-¿after meeting sprint backlog breaking user story to tasks
Sprint 1 - User stories + Fixes for any outstanding bugs

Daily scrums - whats been done, sprint review meeting followed by retrospective meeting

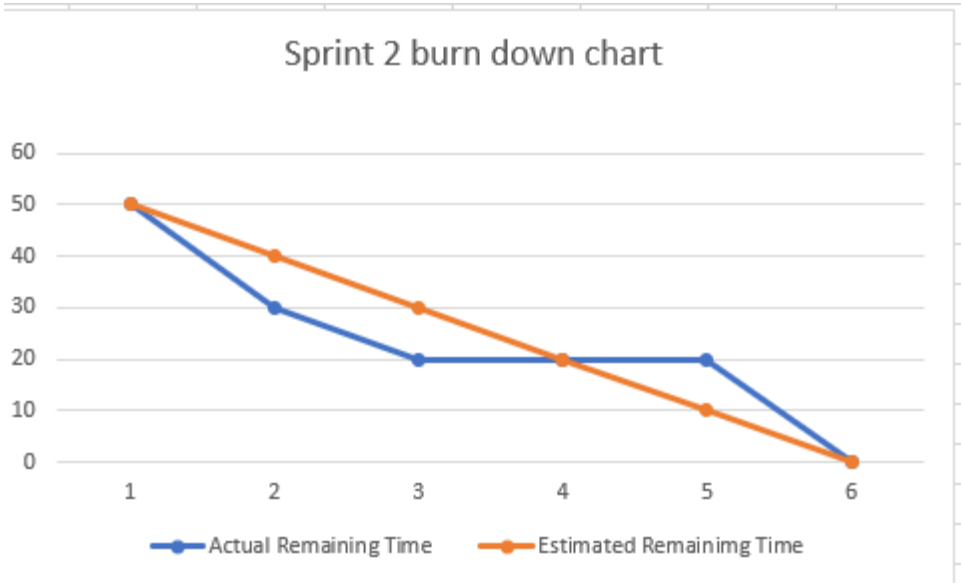


Figure 13: Burn down diagram

6 Module Descriptions and Demonstrations

6.1 Sceenshots

Figure 14: DCMS login

All users aere requierd to login before they can begin using the software's main capabilities. The user begins by selecting the user type which they are (Patient, Dentist, Receptionist or administrator). The entered username is then checked in the appropriate database table to see if the password entered matches the one stored.
The doctor's username is taken as the doctor's practice number, all other users use their email addresses as usernames. Data validation is done on the screen to show appropriate error messages if anything is incorrect(refer to System Testing) Should the user be new and without a username and password, they are required to register as a new user. Based on the input of the combo box, the correct new user portal is loaded. The above the User Interface shows the registration of a new patient.

The above user interface shows the registration of a new dentist. The button on the top left (Authorize), is only clickable from the administrator

patient.png

Figure 15: Create patient

point of view. After the dentist has entered all their required data, in order to be able to use the created credentials to log in, the new dentist must first be authorized(approved to be legitimate) by the administrator.

The creation of a new appointment can be done by either the patient or the receptionist. The details recorded include the patient making the booking and the doctor being booked for. Each patient has a prefered dentist, however one has the option of booking to see a different dentist who may have a different specialization for the needed appointments.
The appointment has a date and time selection, the dates available are from the schedule of the selected doctor for the given week. This is to ensure that appointments are not double booked for doctors.

The illustration above shows how a new appointment is sctructure and stored. The php admin

dentist.png

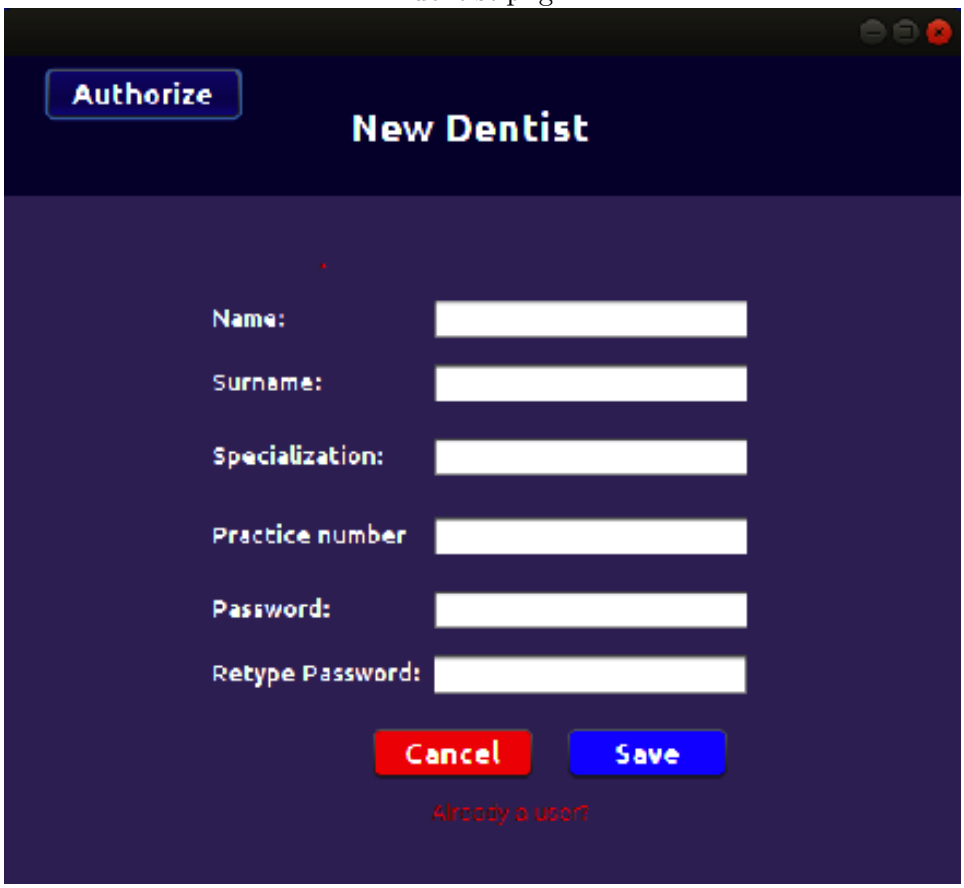


Figure 16: Create patient

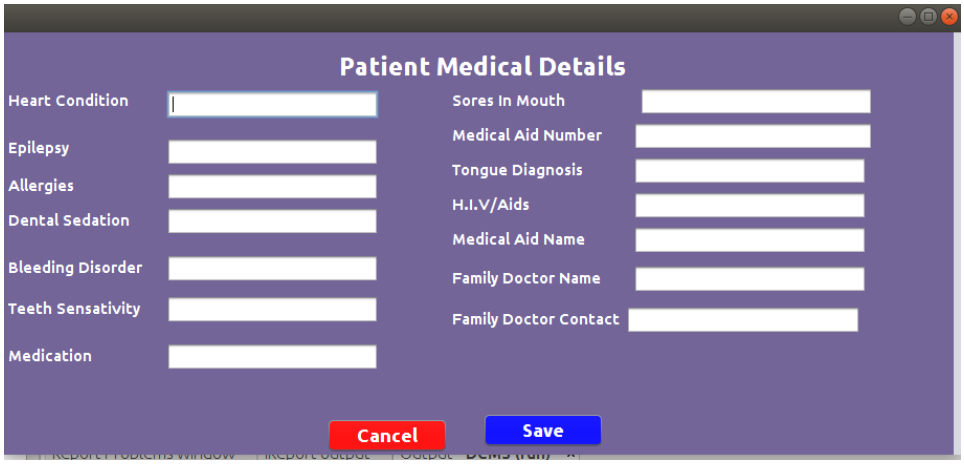


Figure 17: Create patient

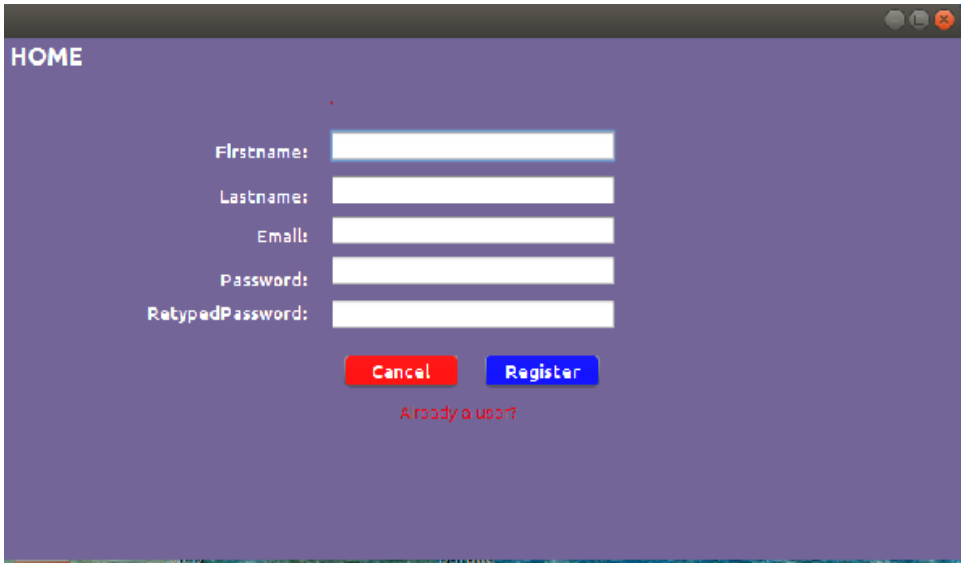


Figure 18: Create patient

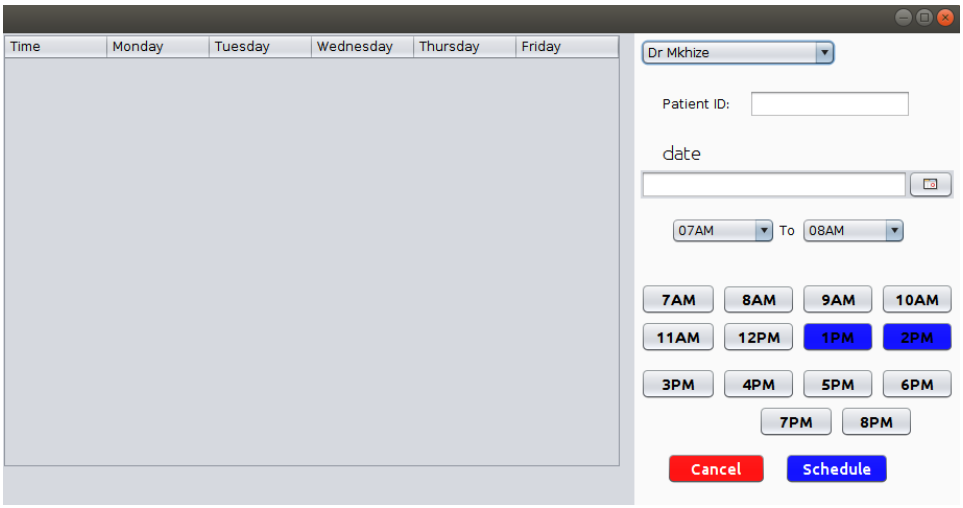


Figure 19: Create Appointment

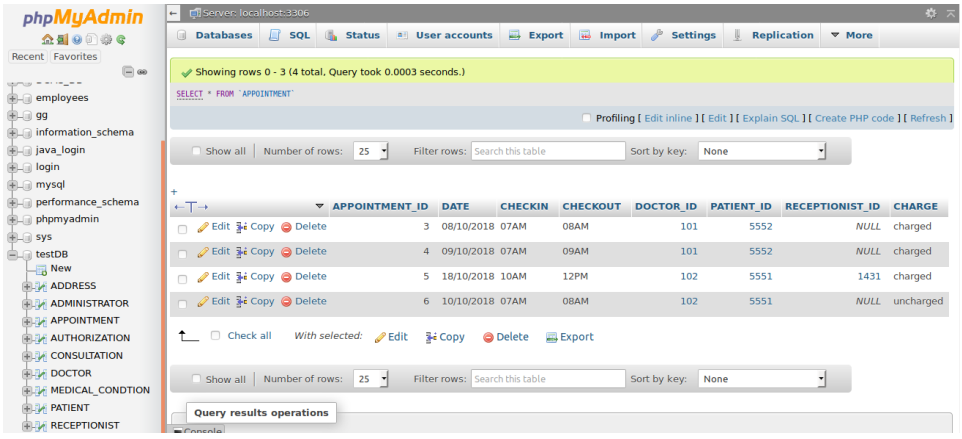


Figure 20: Create Appointment

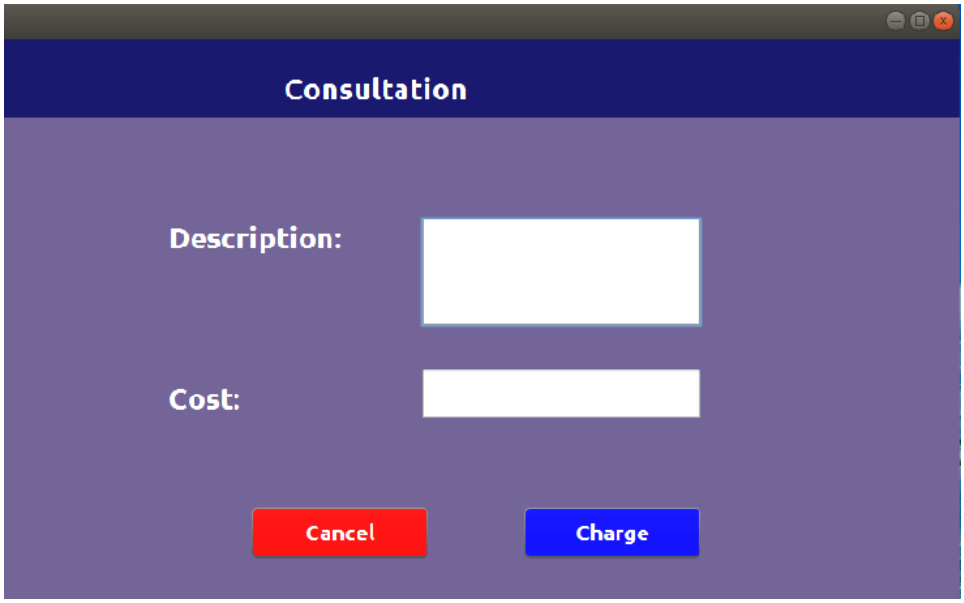


Figure 21: Create patient

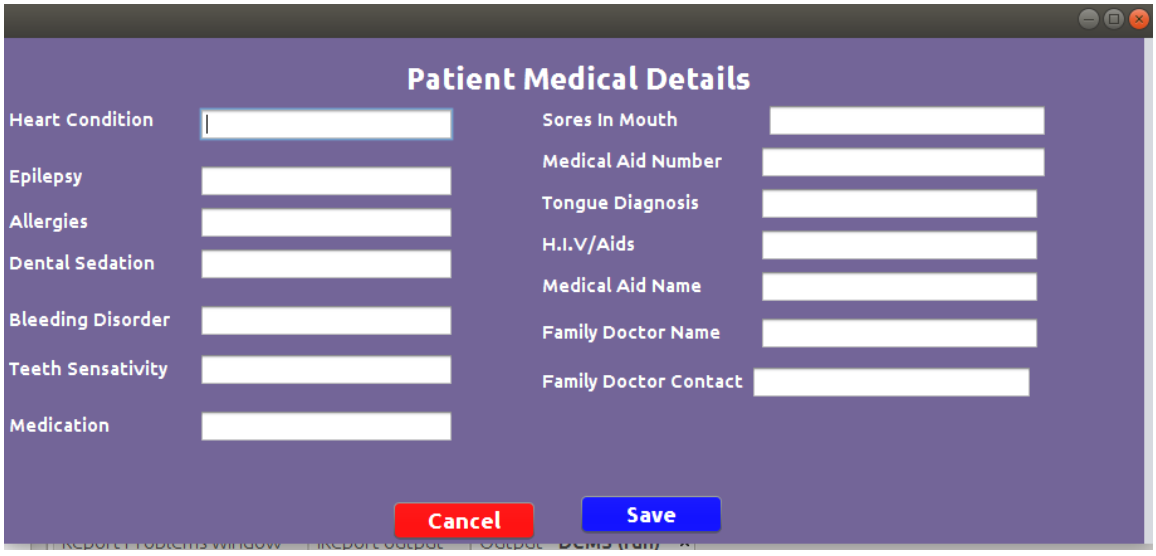


Figure 22: Create patient

7 System development review method

7.1 Sprint retrospective

What went right be

- j

What went wrong

- h

What should we do differently

- k

8 System Testing

8.1 Unit Tests

Unit testing also called component testing was performed on standalone modules to check whether they were developed correctly. The following standalone modules were tested

- **Login**

What was tested	Expected	obtained	verdict
Response to selecting user type from combo box	Correct portal opened for relevent user type selected, either login or creating new user.	If username and password valid, and valid user type chosen, user is directed to correct user portal	pass
Response to valid login user name and password	Accepts the user credentials	Accepts the user credentials	pass
Response to invalid login username or password	Error message" incorrect user name or password "	Error message" incorrect user name or password invalid"	pass
Response to Empty login	Message" please enter user name"	Message " please enter user name"	pass
Response to clicking the login button	Takes you to the relevant user home screen if credentials are valid	Takes you to the relevant user home screen if credentials are valid	pass
Response to clicking sign up button	Takes you to the registering page, depending on the user type selected	Takes you to the registering page, depending on the user type selected	pass

Figure 23: login testing

- **Register**

What was tested	Expected	Obtained	verdict
Resposnse to empty required text field(first name,last name,email,password, retype password)	Message " enter required field"	Message "enter required field"	pass
Response to mismatching password	Error message"password does not match"	Error message" password does not match"	pass
Response to valid inputs	Message" data captured succefully"	Message"data captured succesfully"	pass

Figure 24: Register testing