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

1.1 Purpose

This document serves to describe the processes undertaken in the inception of the Dental Clinic Management system software prototype. 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.

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.Hard copy files stored in cabinets pose a security threat since it is possible for unauthorized personel to gain access because of negligence.Human error in the collection and capturing of data occurs when patients either fill in their details incorrectly or the receptionist captures the data wrongly.Paper files are hard to back up.

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

Define what the following sections do

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.

2 Software Requirements Specifications

2.1 Overall Description

t

2.1.1 Product perspective

2.2 Functionality

2.3 Usability

3 Project Design and Architecture

3.0.1 Architecture

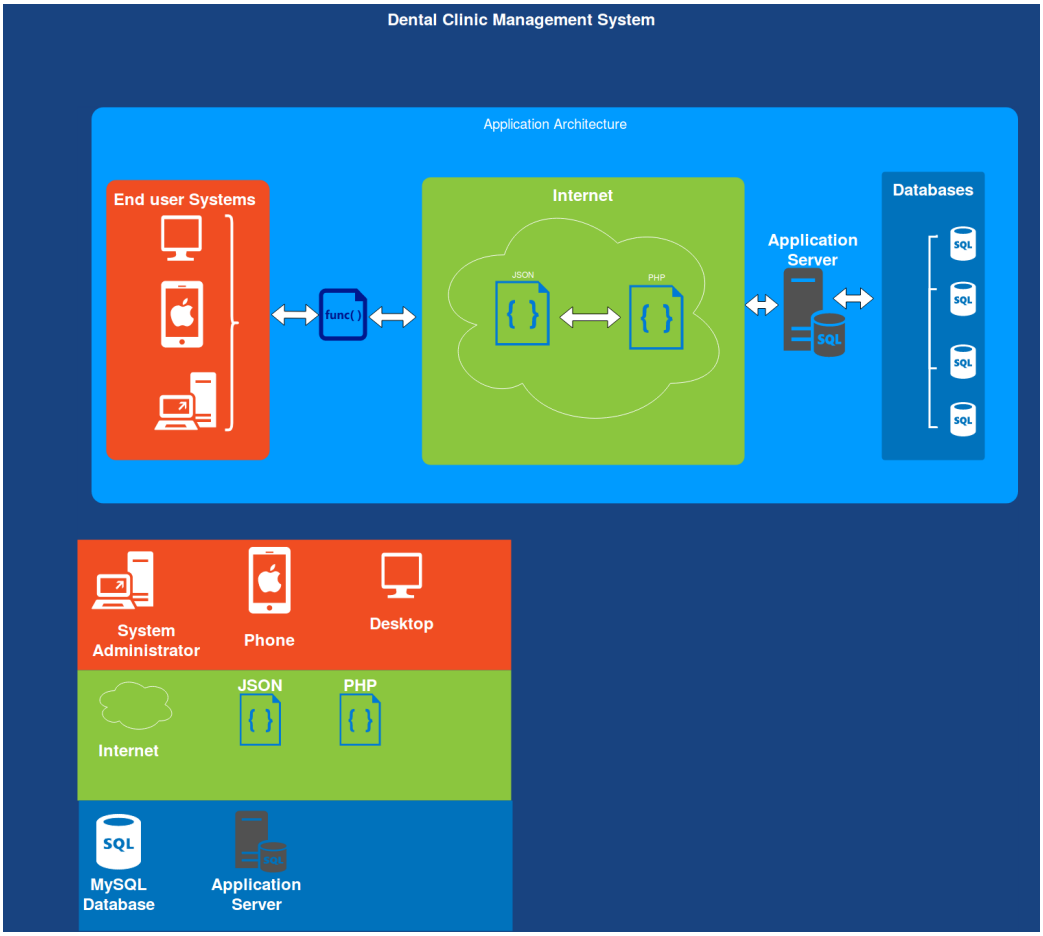


Figure 1: architecture

3.0.2 Implemented Database Tables

```
+----------------------+-------------+------+-----+---------+-------+
| 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 2: Patient Table

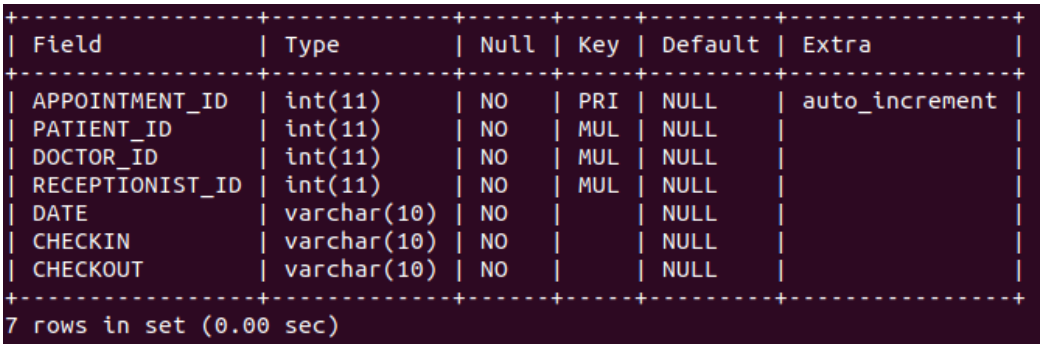
```
+----------------+-------------+------+-----+---------+----------------+
| Field          | Type        | Null | Key | Default | Extra          |
+----------------+-------------+------+-----+---------+----------------+
| DOCTOR_ID      | int(11)     | NO   | PRI | NULL    | auto_increment |
| NAME           | 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 3: Doctor Table

3.0.3 Entity Relationship Diagram

3.0.4 Software Tools

- Database Server: Microsoft SQL Server
- Client: Any web browser



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 4: Doctor Table

- Programming Language:Java
- Development Tools:Netbeans IDE 8.2

3.0.5 Hardware Requirements

The supported Operating Systems:

- **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

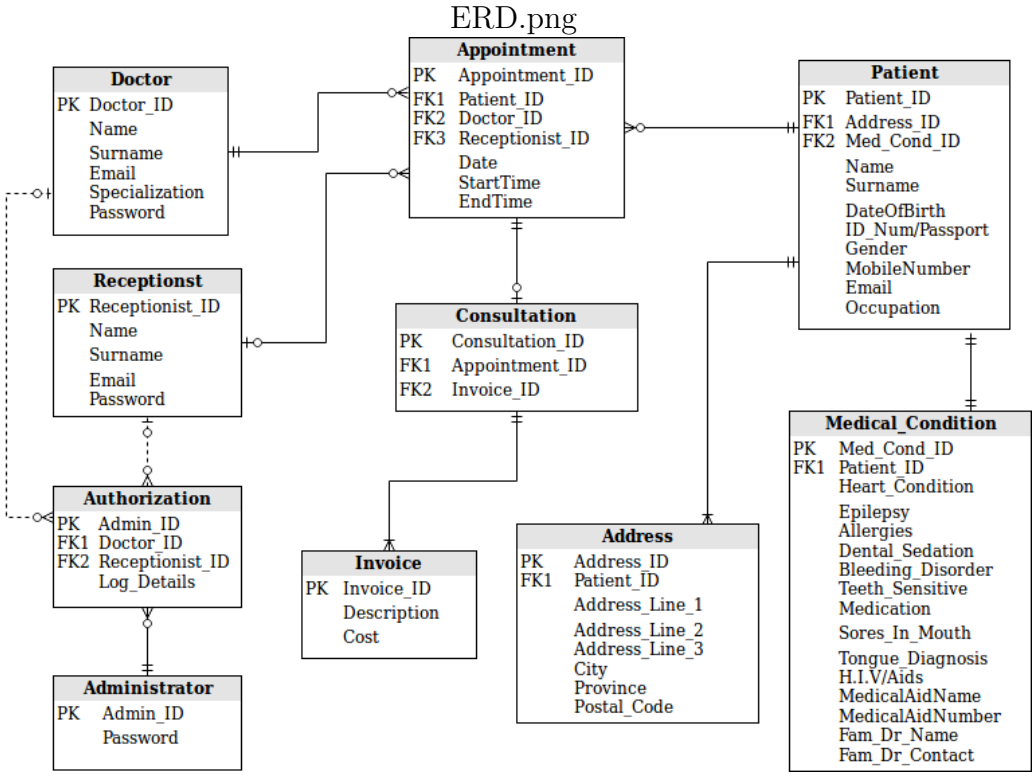


Figure 5: DCMS-ERD

  - Memory: 2 GB
  - Disk space: 650 MB of free disk space

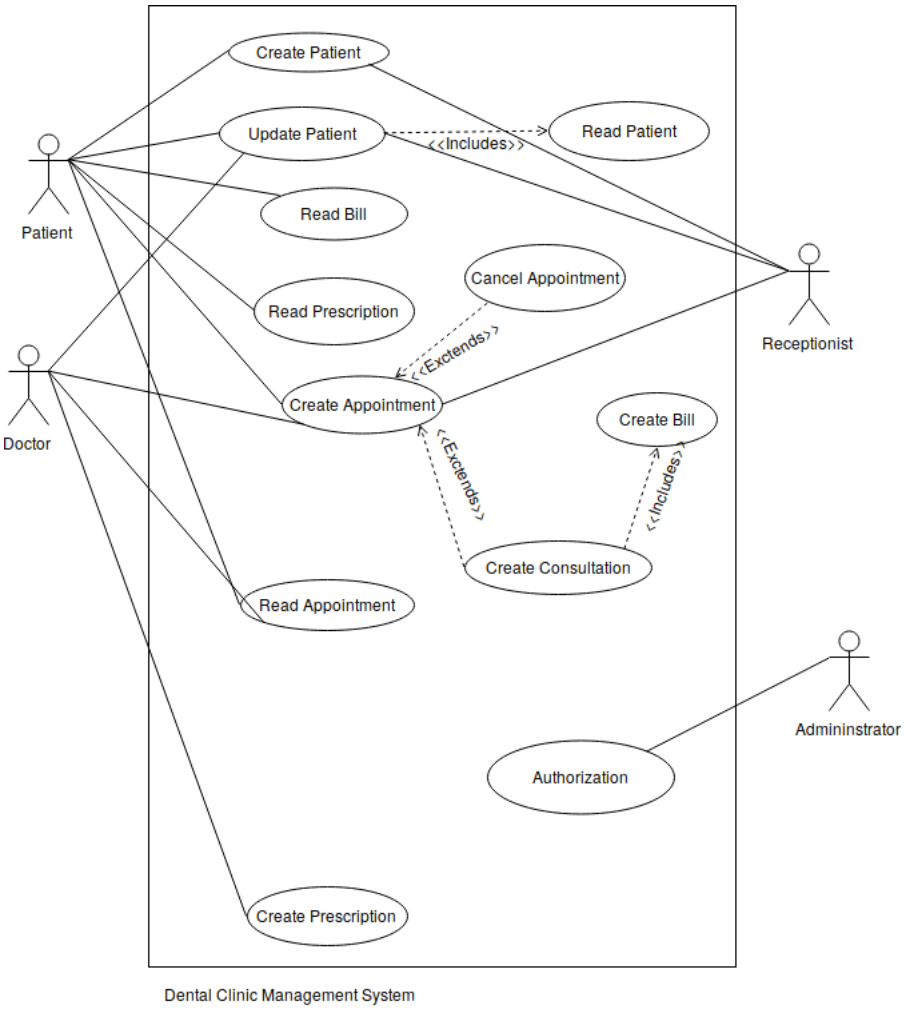
- **Smartphone Requirements:**
  - Android running OS 4.0+
  - iPhone running iOS 8+
  - Windows Phone 8.1+

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.

3.1.1 Use Case Diagram



Case Diagram.png

Figure 6: DCMS-ERD

3.2 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

4.2 Scrum Artifects

4.2.1 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.

4.2.2 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 7: Priority List

4.2.3 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 8: 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

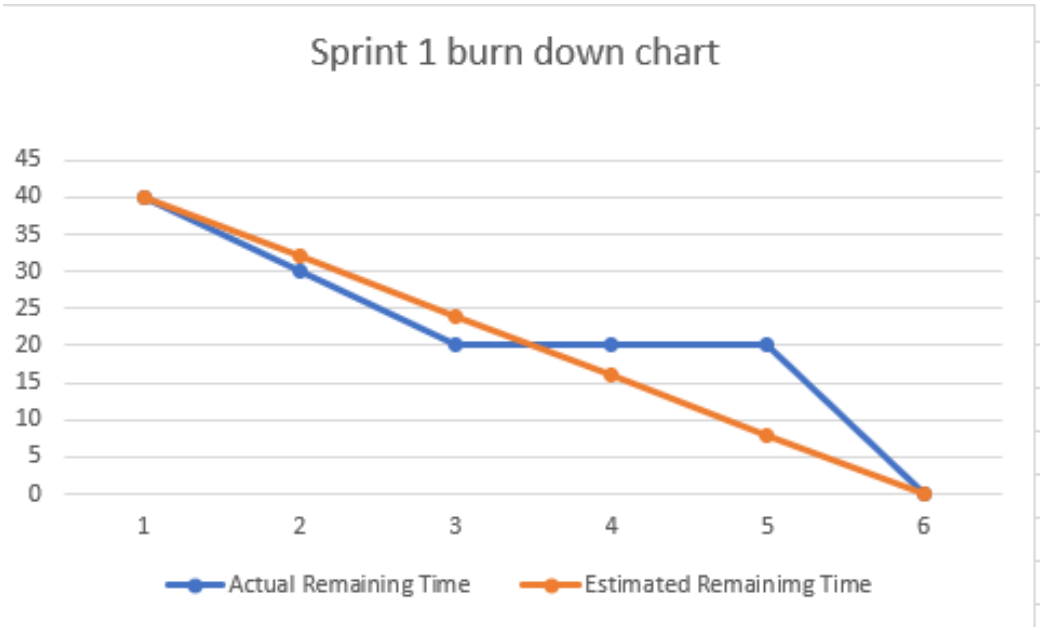


Figure 9: Sprint 1 burndown diagram

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 10: 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

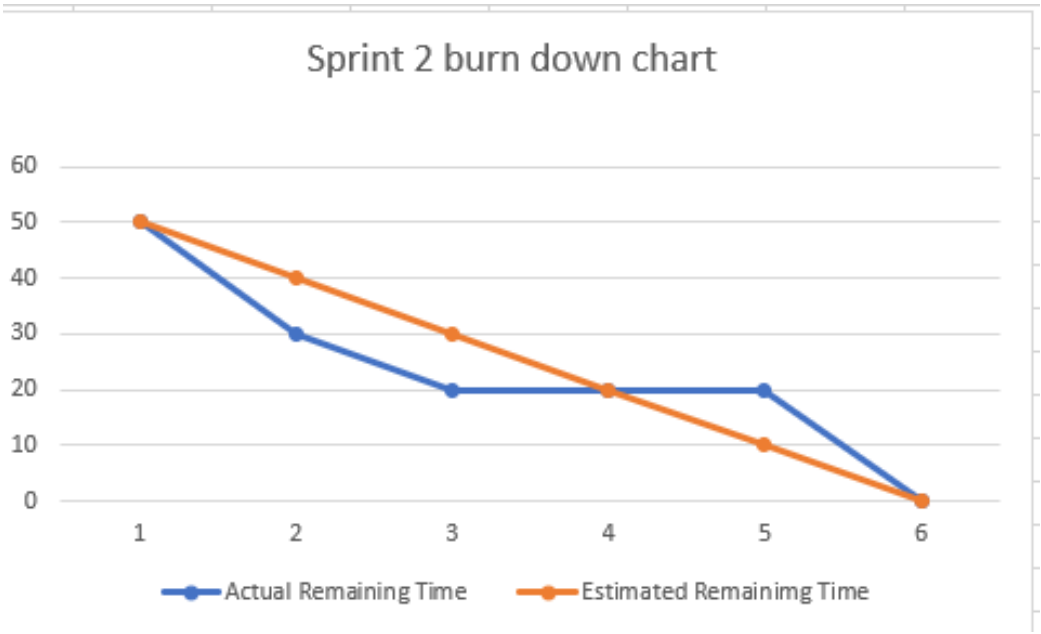


Figure 11: Burn down diagram

5 Module Descriptions and Demonstrations

5.1 Sceenshots

6 System development review method

6.1 Sprint retrospective

What went right

-

What went wrong

- Many tasks were underestimated
- Scrum meetings were not effective, we did not hold

What should we do differently

- k

7 System Testing

7.1 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 12: 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 13: Register testing