LONDON METROPOLITAN UNIVERSITY

Islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CS6P05NI Final Year Project

Assessment Weightage & Type

25% FYP Interim Report

Semester

2020 Autumn

Student Name: Sagar Poudel

London Met ID: 18030043

College ID: np01cp4a180204

Internal Supervisor: Pritivi Maharjan

External Supervisor: Shubhankar Sharma

Assignment Due Date: 2020/12/25

Assignment Submission Date:2020/12/25

Word Count (Where Required):

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Acknowledgement

This interim report would not have completed successfully without encouragement and guidance of my supervisors Mr. Shubhankar Sharma and Mr. Prithivi Maharjan. I always feel encouraged and motivated after attending meeting with them. I am grateful for their guidance, support, supervision and encouragement. I would like to extend my sincere thanks to both of supervisors for their tremendous help and support. I would also like to thank my friends and the participants in my survey, who have voluntarily shared their valuable time for goodwill of this report.

Abstract

This report is like progress report of final year project **Institute Management System** which is a web-based application. This application is helpful for educational institute to maintain record of their students, send SMS to students regarding class, overview flow of students in institute via various reports. This application also includes Point of Sales system so that institute can sell study related books.

This report consists details regarding progress of final year report in the means of five chapters such as Introduction, Background/literature review, Development to Date, Analysis of Progress and Future Work.

Introduction includes short information of final year project i.e. **Institute Management System** whereas background/literature review holds detailed information about resources requirements , methodology used in developing project along with analysis and comparison of similar projects. Detailed information of development of project till date, progress analysis by comparing with Gantt chart and work that is to be done further are provided respectively in development to date, analysis of progress and future work section.

Table of Contents

Table of Figures

Table of tables

Institute Management System

1. Introduction

This system is a web application for educational institute specially for those institute which provides education to various types of students. As being a developing country, education system is also in the phase of development so that most of works are manually done in the field of education. There are various sources that provide education such as school, college, institute. This project is focused on educational institute that provides education to people who get enrolled to compete in exams of government jobs/ public service.

1.1. Problem statement

In educational institute, there are many students and their records are almost manual in most of education system. Educational institute have problem due to:

1. Manual record of students
2. Depend on paper / register/ brain to remember type of student
3. No proper track of students
4. Difficulty in sending SMS to students
5. Manual billing of fee payment

1.1. Project as a solution

This project will help people who will have educational institute by working as solution:

1. Digital record of students
2. Allocates students at the time of admission
3. Sending SMS to students to give information about class
4. Digital billing of fees
5. Generate manual/month/annual report
6. Digital billing of books related to study materials by using barcode.

1.2. Aim and Objectives

1.2.1. Aim

The main aim of project to make work digitalize more than manual in the field of educational institute.

1.2.2. Objectives

Some objectives of project are:

- To learn about scope of digitalization
- To manage manual works of institute.
- To learn about API programming.
- To make work easier for people involving in educational institute
- To get knowledge on programming concept

1.3. Report Structure

This part of report shows structure of remaining parts.

a) Background/ literature view
   i) Client Description and Requirements.
   ii) Resource Requirements
   iii) Methodology Used
   iv) Similar Projects
   v) Analysis of similar projects with my projects
b) Development to date
   i) System Architecture
   ii) Use case diagram
   iii) High level use case descriptions
   iv) Wireframes
   v) ER diagram
   vi) Web application development
c) Analysis of progress
d) Future work

e) References
f) Appendix A
g) Appendix B
h) Appendix C

2. Background/literature view

2.1. Client Description and Requirements

2.1.1. Client's Name

The Success Institute

2.1.2. Description

This institute is newly, fresh startup which gives tuition classes for courses such as staff nurse, A.N.M, AHW, *kharidar*, *nayab subba* etc. Our client "The Success Institute" is located in Nepalgunj, Banke. Founders of this institute are young and energetic so they were eager to listen about project IMS and assured to give essential requirements which can be helpful in making a good project.

2.1.3. Requirements

Requirements and suggestions provided by client are given below:

i) System should be secure enough so that no outsider can interfere or stalk on confidential data of institute.
ii) System must have different login system for both staff and owner so that staff couldn't misuse system.
iii) There is difficulty in sending message to every student so that system should have feature of sending SMS to student at once.
iv) System should be able to show/ track reports monthly/annually.
v) There should be feature of viewing and printing reports of fee.

Please refer Appendix C for Survey Findings

2.2. Resource requirements

2.2.1. Software Requirements

- PHP/ Laravel Framework
- MySQL for Database
- Html/CSS/JS
- Code editor i.e. Visual Studio / PHP storm
- API for SMS integration.
- Bar code reader app if scanner is not available

2.2.2. Hardware Requirements

- PC with windows 10
- Web cam/ phone with camera/ bar code scanner for scanning bar code of books.

2.3. Methodology

2.3.1. Considered Methodology

Waterfall Methodology

Waterfall methodology is also one of the methodologies which is commonly used at the time of software development. It is a sequential development model. Different phases of waterfall model life cycle are:

i) Analysis
ii) Design
iii) Development
iv) Testing
v) Implementing
vi) Maintenance

In this methodology, all phases are carried out in a linear sequence i.e. to go to next phase previous phase must be complete. All phases are done without overlapping to one another.

Each step is frozen before initialization of next step. Requirement phases should be frozen before starting design and for coding phase design phase should be completed.
This method was not selected because of its linear sequence. As this project is web application and as design part and other logical part are always iterative model need to be flexible and support changes in any part of phases in order to catch up with modern days. In this project, requirement can be changed as per guidance and requirement of client and this changing in requirement is not supported by waterfall methodology. (S.Balaji & Murugaiyan, 2012)

2.3.2. Selected Methodology

Rational Unified Process

There are many methodologies for software development. Some of them are; waterfall, agile, scrum, spiral, joint application, rational unified process etc. We are using Rational Unified Process (RUP) for development of this project.

Rational Unified Process (RUP) is one of the techniques used for software development. RUP provides a disciplined approach to assigning tasks and responsibility within a development organization. Its main goal is to ensure the production of high- quality software that meet the need of end-users within a predictable schedule and budget. (Kruchten, 2004) There are four phases of RUP methodology. They are:

- Inception:

In this phase, requirement analysis is done. This is the phase which describes schedule, estimation of overall system and risk assessment of project. It also describes scope and goals of the system along with identifying interaction with systems (t4tutorials, 2020).

- Elaboration:

Architecture of system is evaluated in this phase of rational unified process. This phase is considered important as it includes analysis of problem domain. Use case diagrams are made in this phase (Christensson, 2006).

- Construction:

In this phase, major coding is done along with extensive testing. In big projects, construction phase can be repeated or there may be several constructions phases in order to divide use cases into segments so that it will be smaller and manageable. At the end of this stage, the project team should have user manuals and a beta version of the system ready to be evaluated. (ANWAR, 2014)

- Transition:

Transition is the last phase of rational unified process in which analysis of system along with its maintenance is done. This is the phase where product is released. (Scott, 2001) .

2.4. Similar Projects

2.4.1. Project 1 : Web Based Student Information Management System

Authors: S.R.Bharamagoudar , Geeta R.B. , S.G.Totad

This Student Information Management System helps in maintenance of student information. This system can be used by educational institutes to maintain records of students. Student information system deals with all kind of student details, academic related reports, college details, course details, curriculum, batch details, placement details and other resource related details too. User interface of this system is designed and implemented to replace current paper records. The main purpose of this system is to maintain information of students up to date so that it can help in improvement of efficiency of institutes/colleges. (S.R.Bharamagoudar, et al., 2013)

Figure 1 Web Based Student Information Management system

2.4.2. Project 2: Student Record Management System

Author : Fabian Ros

This Student Record Management System is PHP based system which is made in order to maintain records of students. This project run on virtual serve i.e. XAMPP. This system allows user to add student records such as name, contact number, photo etc. In this system, admin plays vital role by updating, removing students from system. (Ros, 2019)

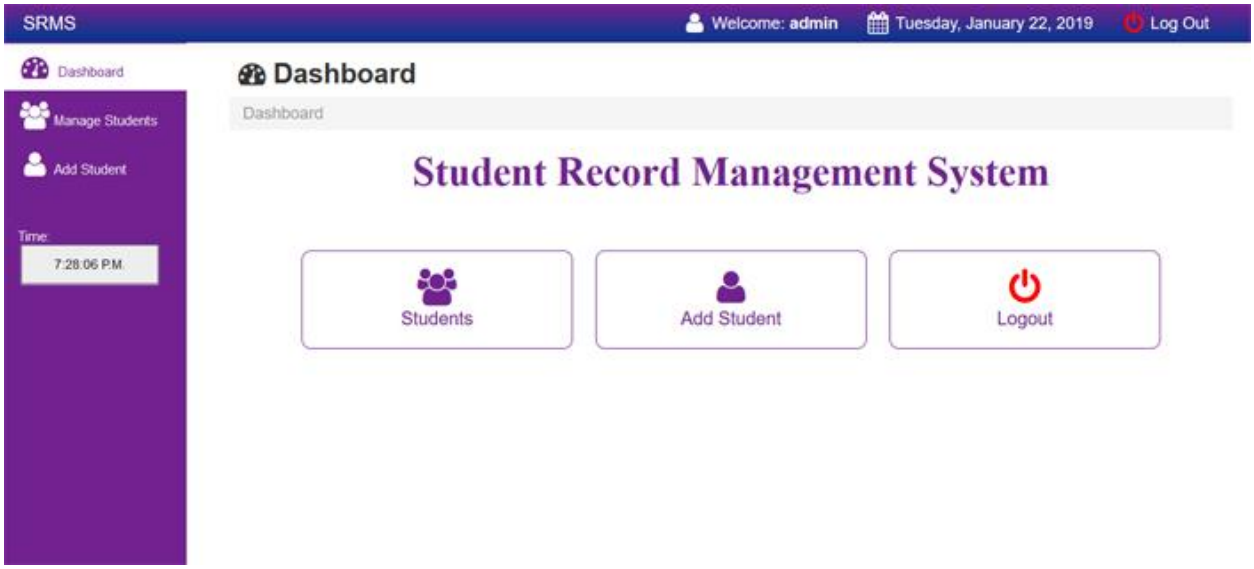

Figure 2 Student Record Management System

2.4.3. Project 3: Student Management System

Authors: Sangamesh K, Akash Samanekar, Ningappa T Pujar

This Student Management System is a web-based system which will help to manage everyday tasks of institution. The main aim of this project is to manage records of students, attendance , accounts, admissions etc. and keep track of information digitally as many works are carried out of paper based in most of academic institution. (K, et al., 2018)

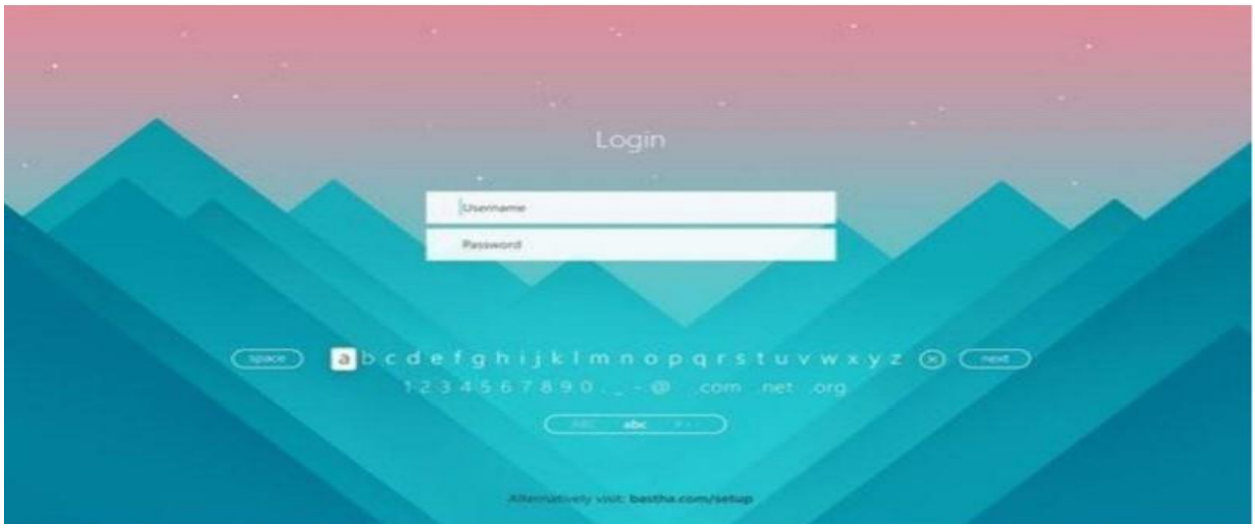

Figure 3 Student Management System

2.5. Analysis of similar projects with my projects

Project Comparison Table

S.N	Features	Project 1	Project 2	Project 3	This project
1	Technology Used	HTML, CSS, JS, PHP, SQL	PHP, CSS, JavaScript, and Bootstrap.	Java frameworks such as Hibernate, Spring-Data-JPA, Angular	HTML, CSS, JS, PHP/ Laravel Framework, MySQL
2	Fee Management	✓	✗	✗	✓
3	Student Type/ Course Type Management	✓	✗	✓	✓
4	Easy to use	✗	✓	✗	✓
5	SMS sending	✗	✗	✓	✓

Table 1 Project Comparison Table

3. Development to date

3.1. System architecture

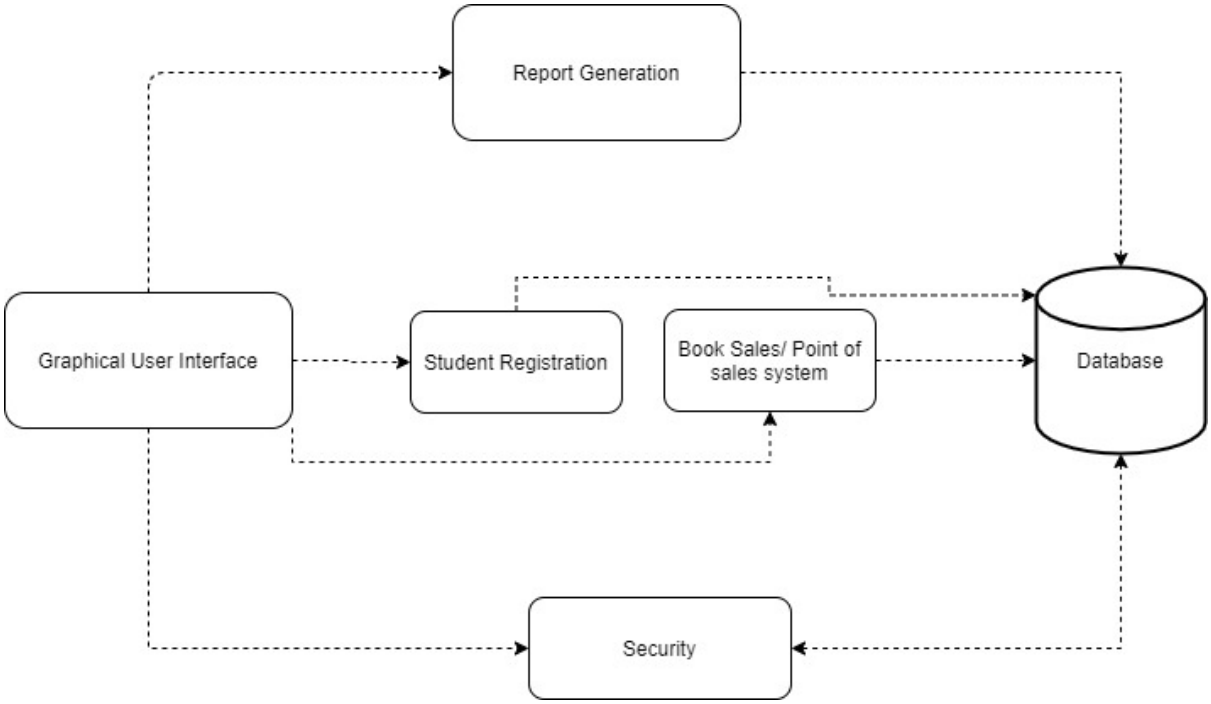

Figure 4 System Architecture

3.2. Use case diagram

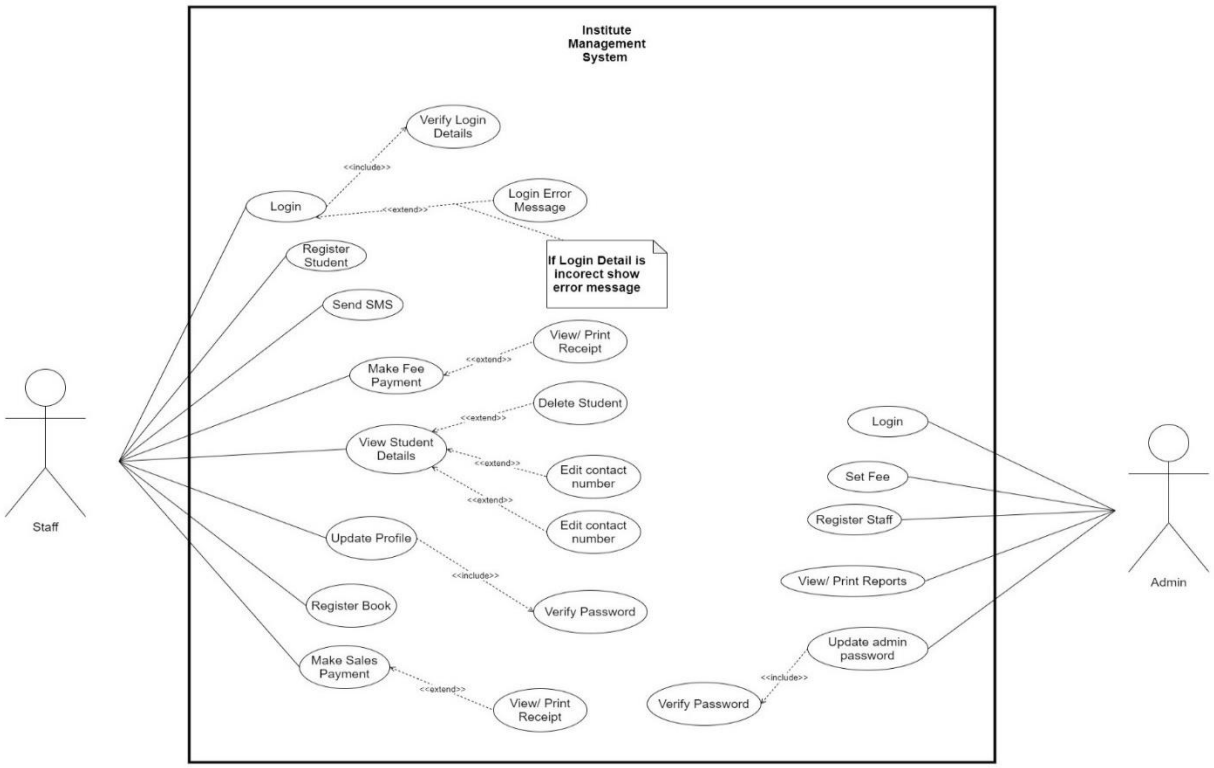

Figure 5 Use case diagram of Institute Management System

3.3. High level use case descriptions

High level use case description are given below:

Use Case	**Login**
Actor	Staff
Description	Staff logins to application to access other functions.

Table 2 High level use care description of login

Use Case	**Register student**
Actor	Staff
Description	Staff register student in application

Table 3 High level Use case description of register student

Use Case	**Send SMS**
Actor	Staff
Description	Staff send SMS to student choosing from database

Table 4 High level Use case description of send SMS

Use Case	**Fee Payment**
Actor	Staff
Description	Staff makes payment of fee and give receipt to student

Table 5 High level Use case description of fee payment

Use Case	View Student Details
Actor	Staff
Description	Staff can view student details along with editing it.

Table 6 High level Use case description of view student details

Use Case	Update Profile
Actor	Staff
Description	Staff can update his/her profile

Table 7 High level Use case description of update profile

Use Case	Register Book
Actor	Staff
Description	Staff register book that it new to institute .

Table 8High level Use case description of register book

Use Case	**Make Sales Payment**
Actor	Staff
Description	Staff make payment of book that he/she sells.

Table 9 High level Use case description of make sales payment

Use Case	**Login**
Actor	Admin
Description	Admin logins to application to access other functions.

Table 10 High level Use case description of login admin

Use Case	**Set fee**
Actor	Admin
Description	Admin can set fees of various fields that are taught in institute.

Table 11 High level Use case description of set fee

Use Case	**Register Staff**
Actor	Admin
Description	Admin registers staff

Table 12 High level Use case description of register staff

Use Case	**View/ print Reports**
Actor	Admin
Description	Admin can print overall reports.

Table 13 High level Use case description of view/print report

Use Case	**Update Admin Password**
Actor	Admin
Description	Admin update his/her password.

Table 14 High level Use case description of update admin password

3.4. Wireframes

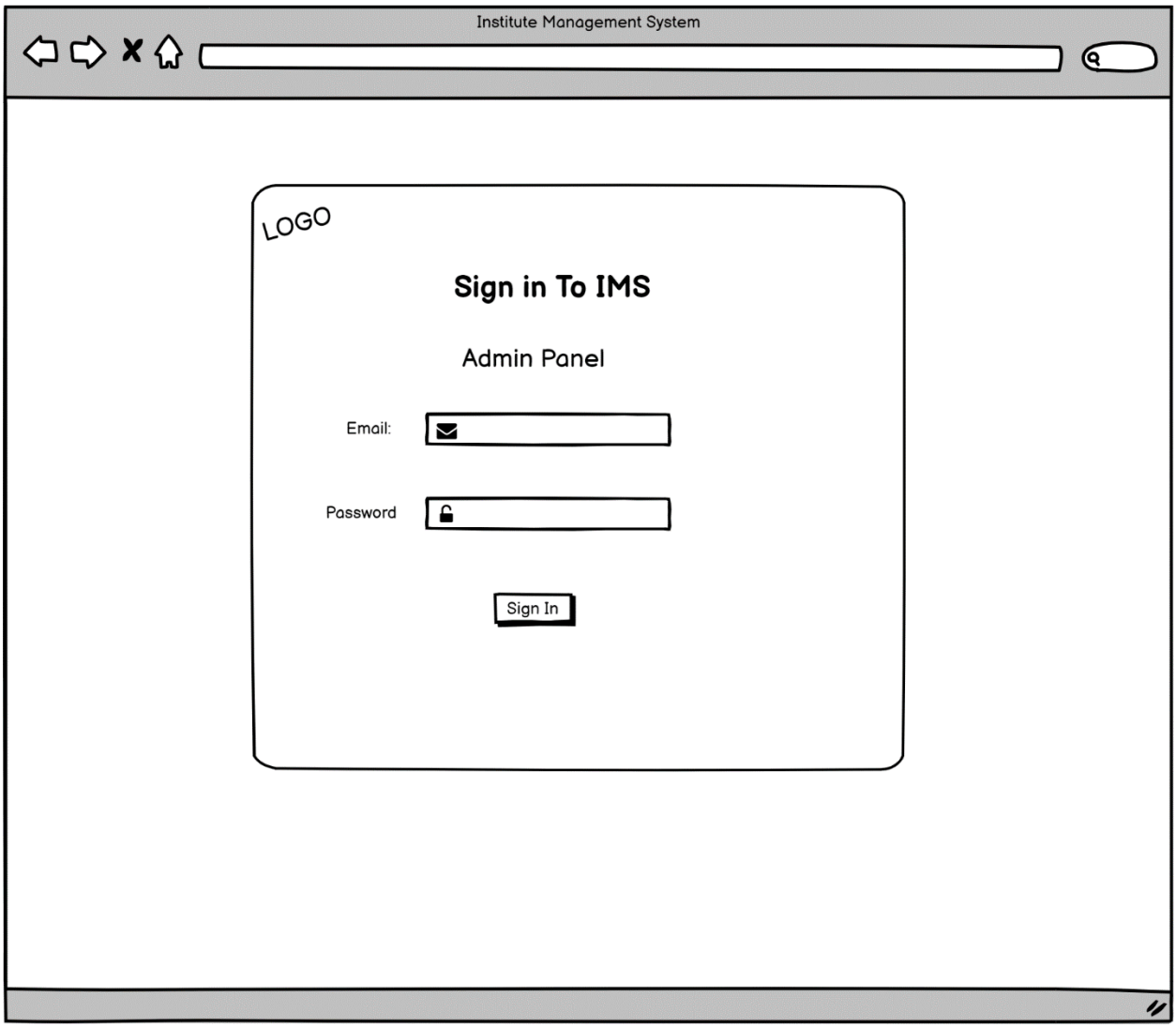

Figure 6 Admin Panel Login Page

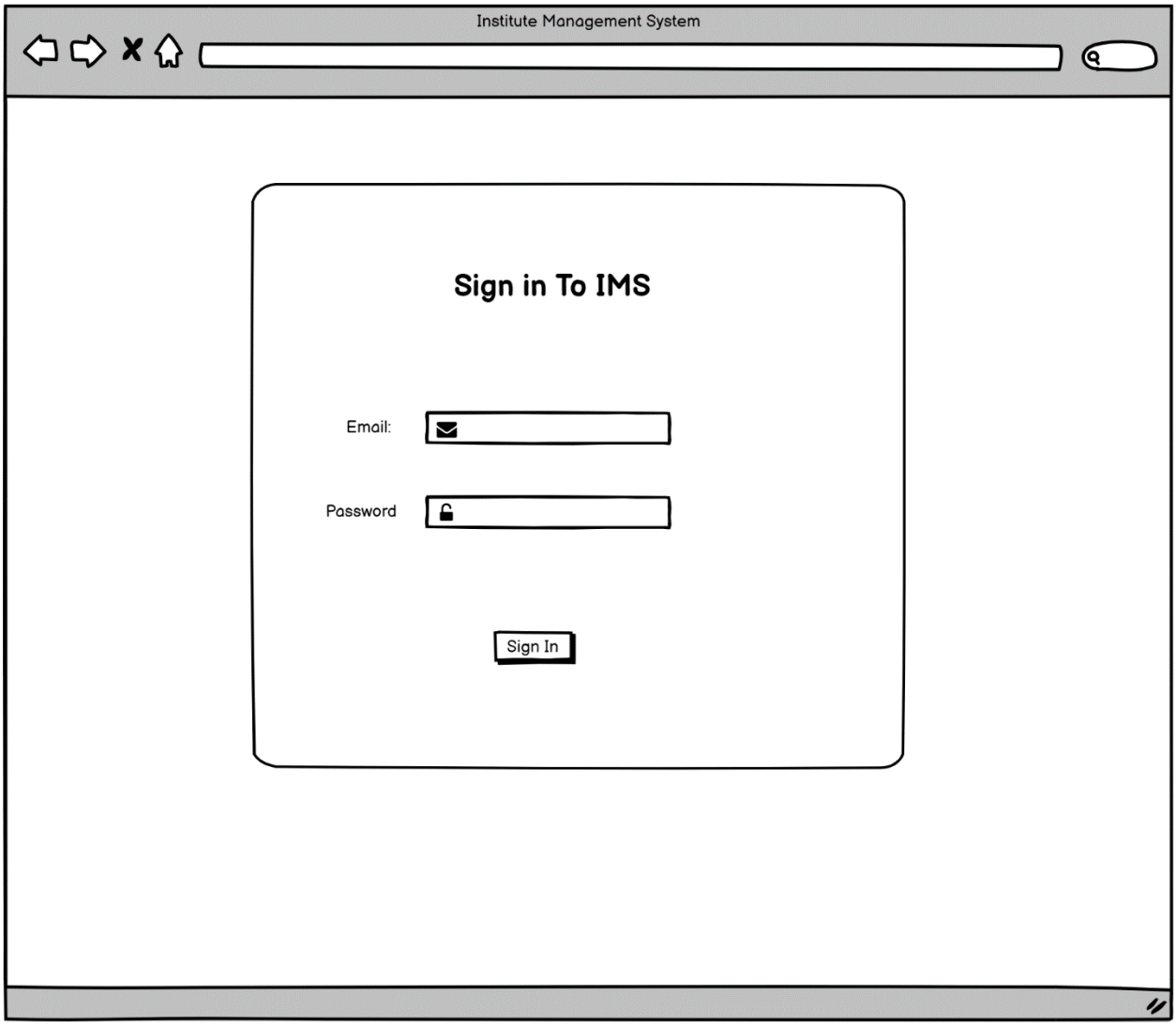

Figure 7 Login for Staff Page

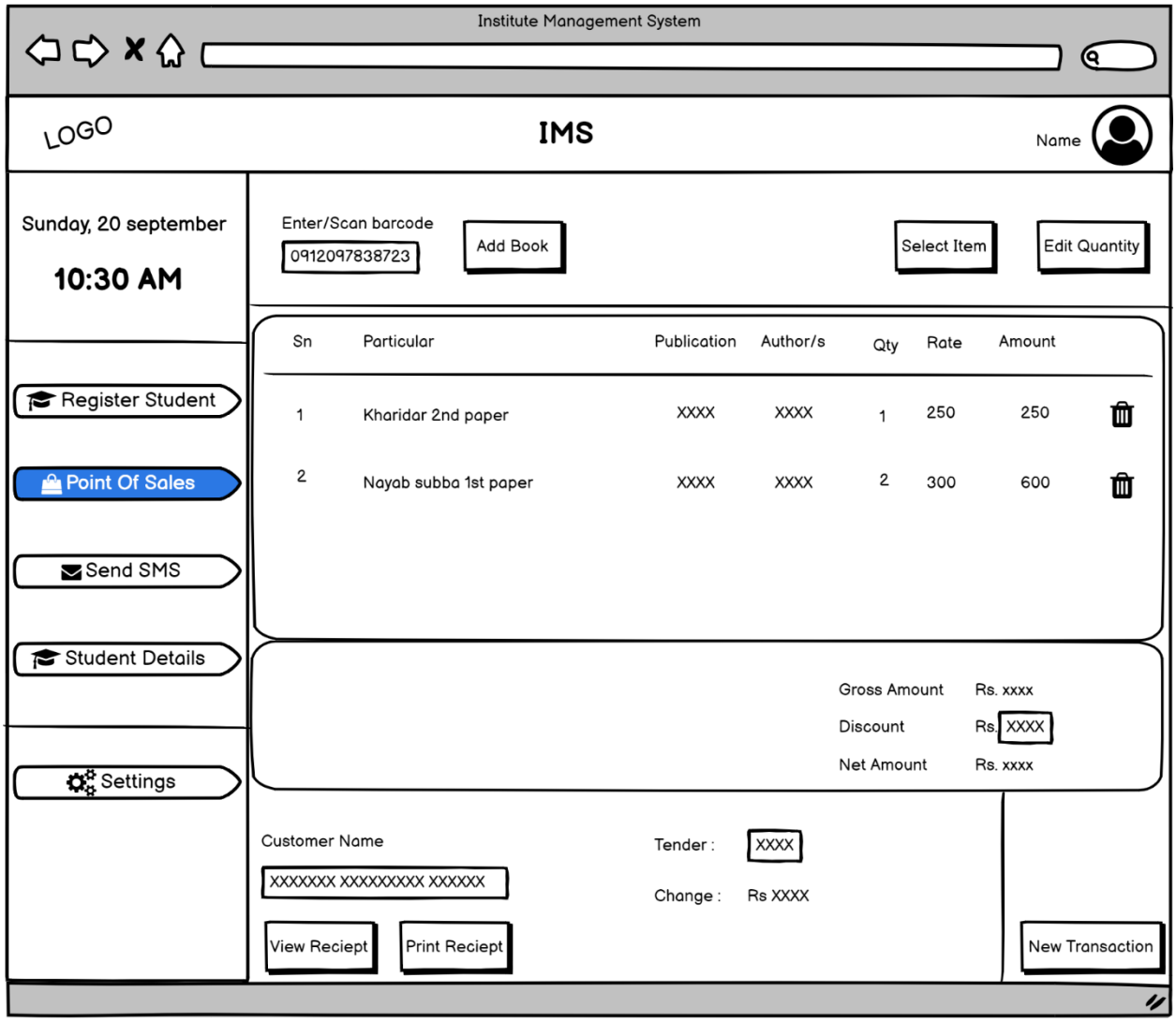

Figure 8 Point of sales / landing page of web application

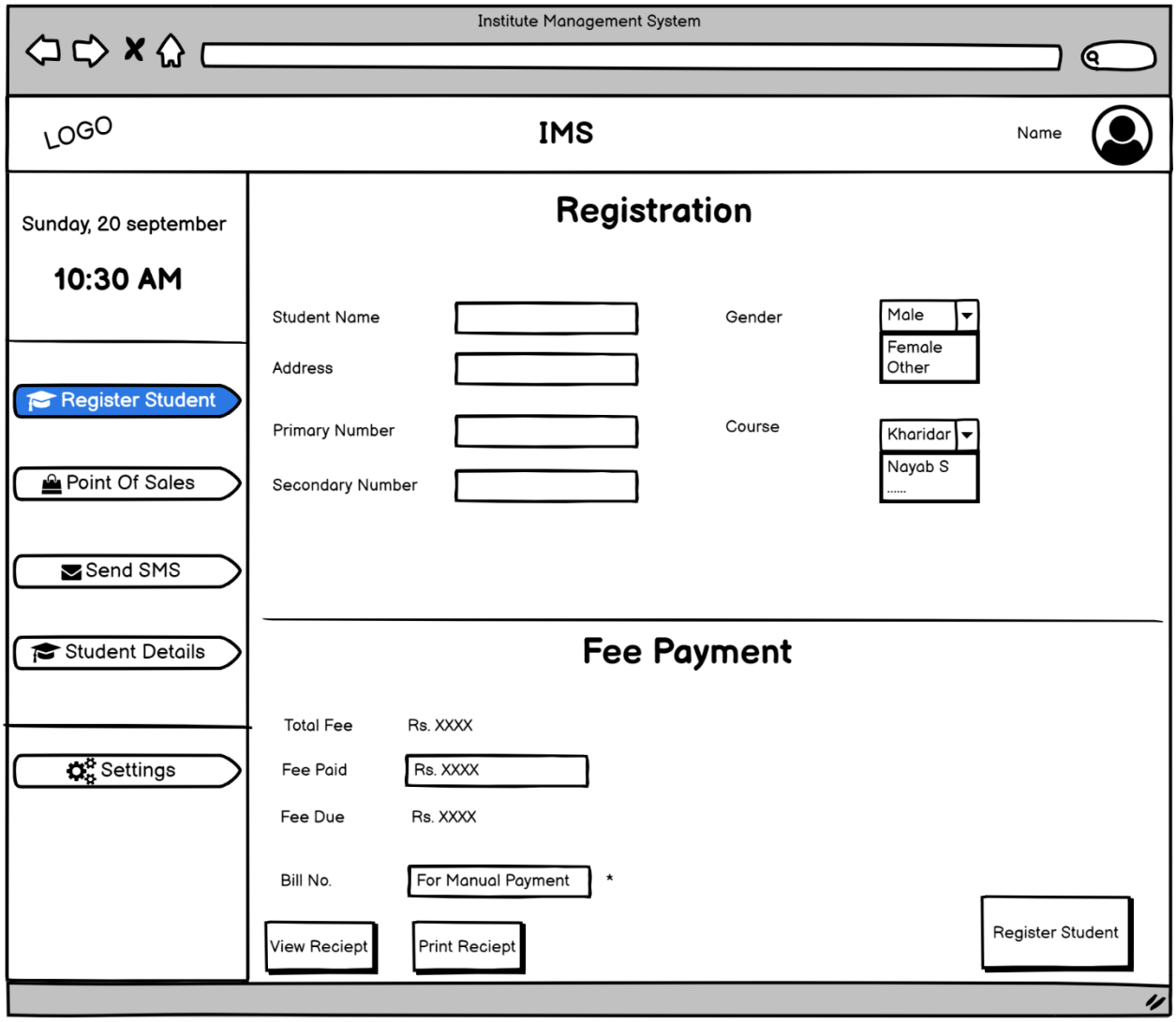

Figure 9 Student Registration page

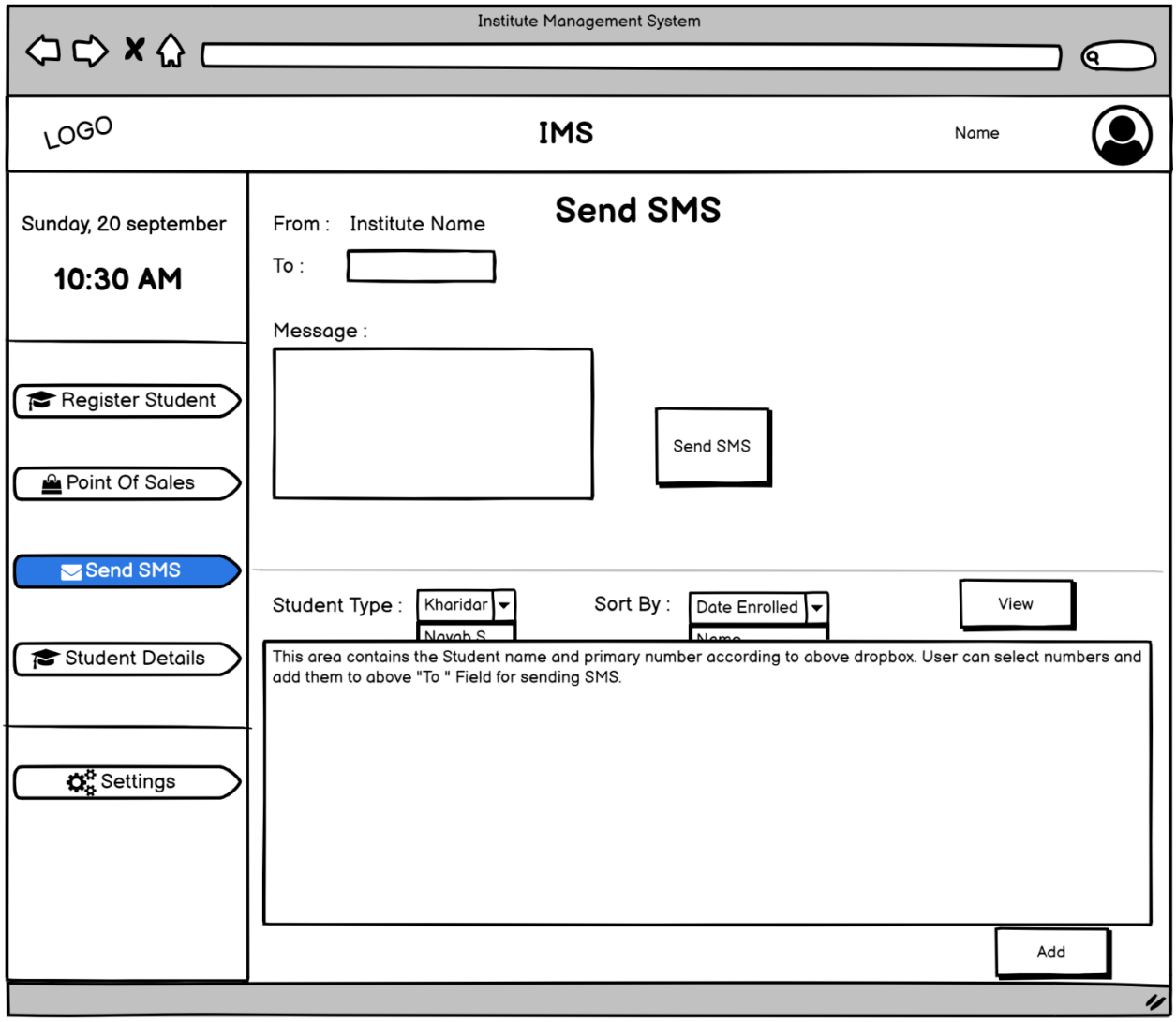

Figure 10 Send SMS page

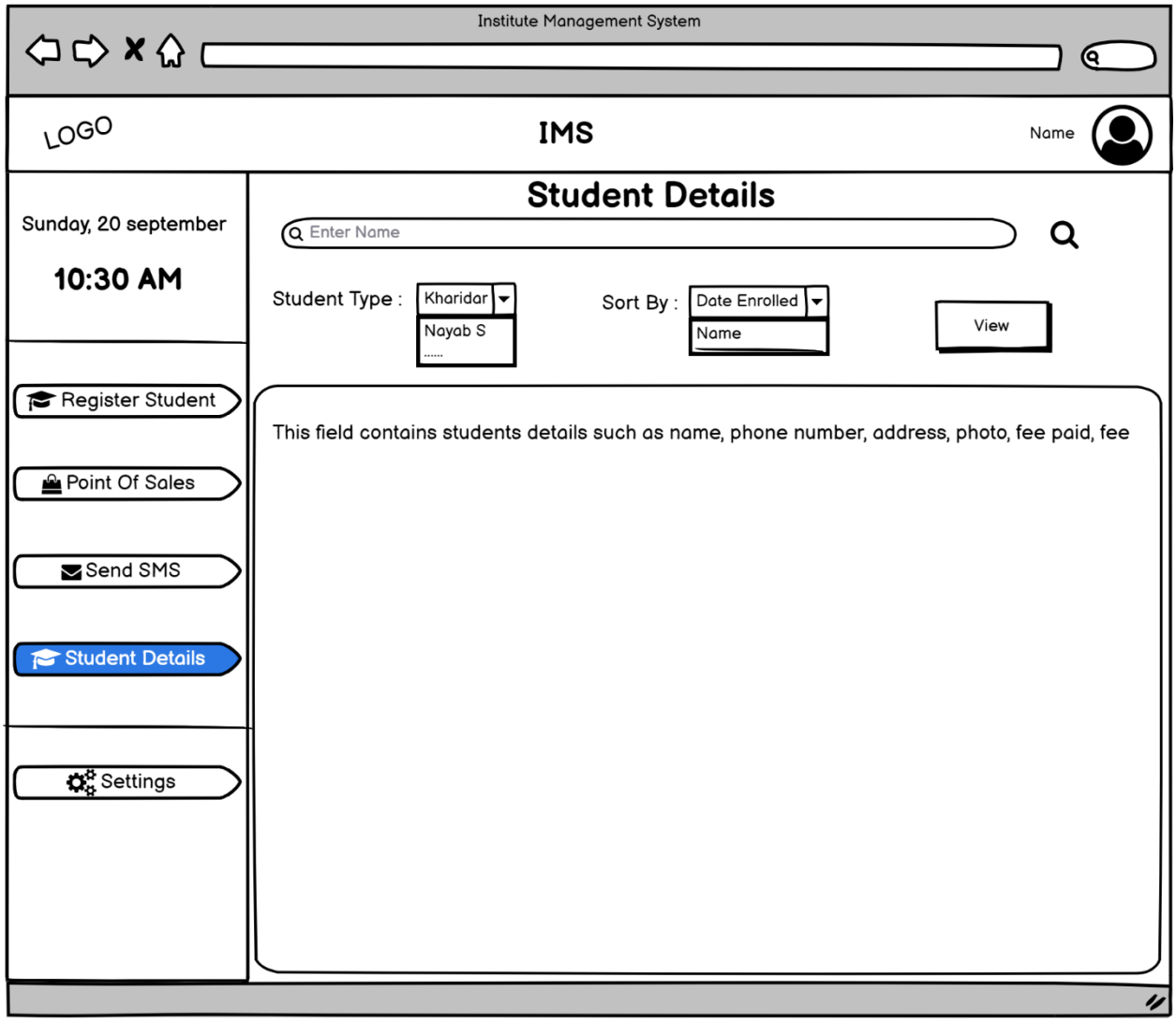

Figure 11 Student Details Page

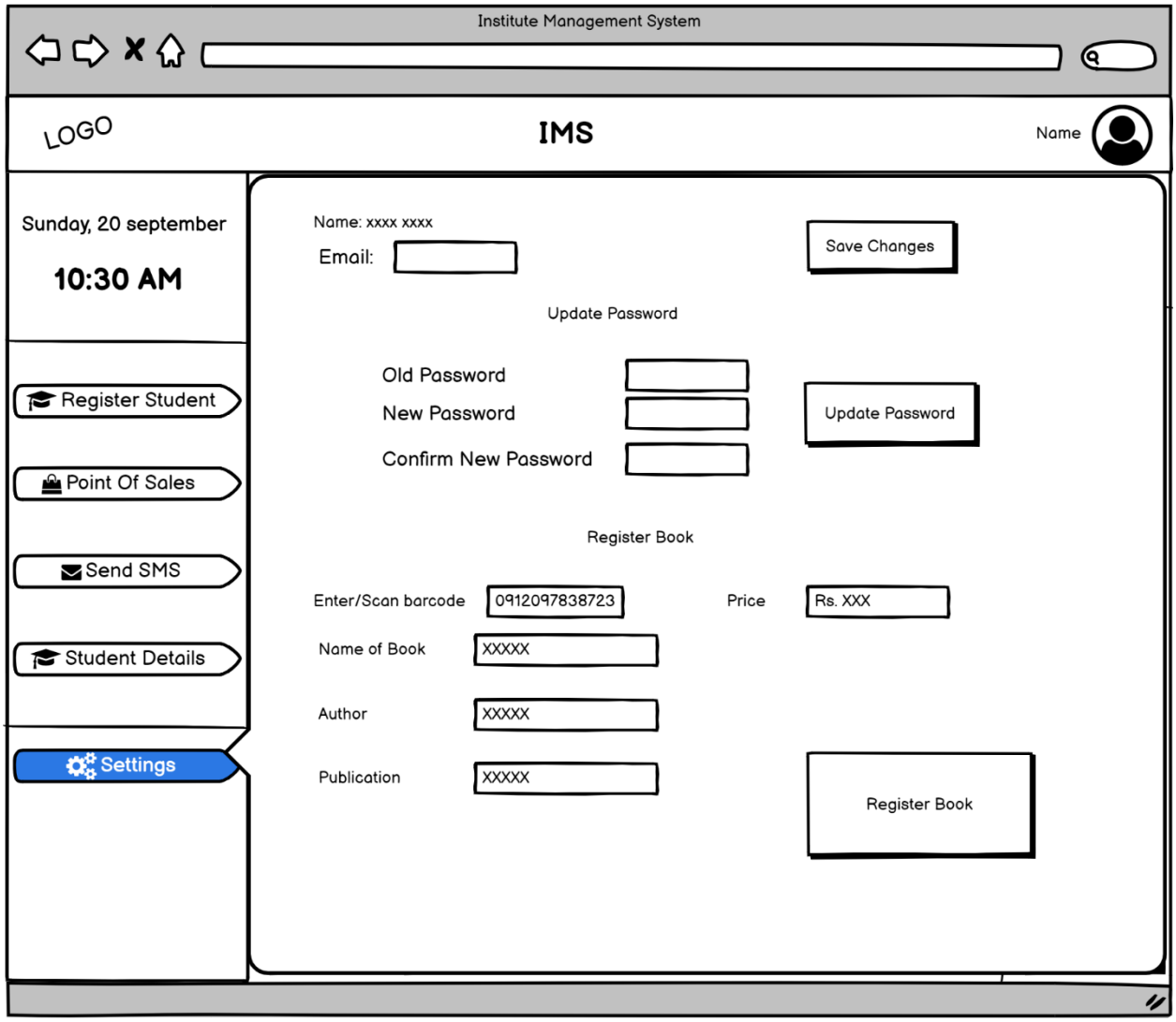

Figure 12 Account Setting / Staff Setting page

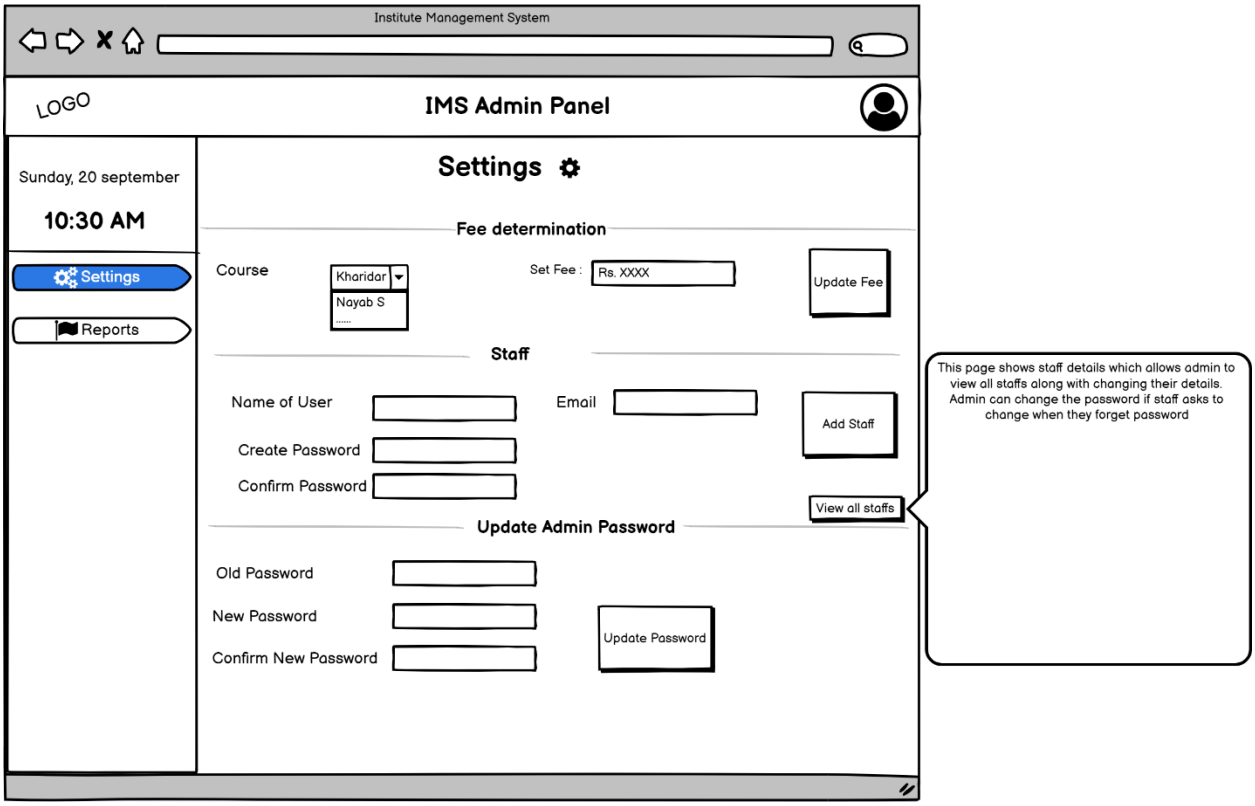

Figure 13 Admin Setting Page

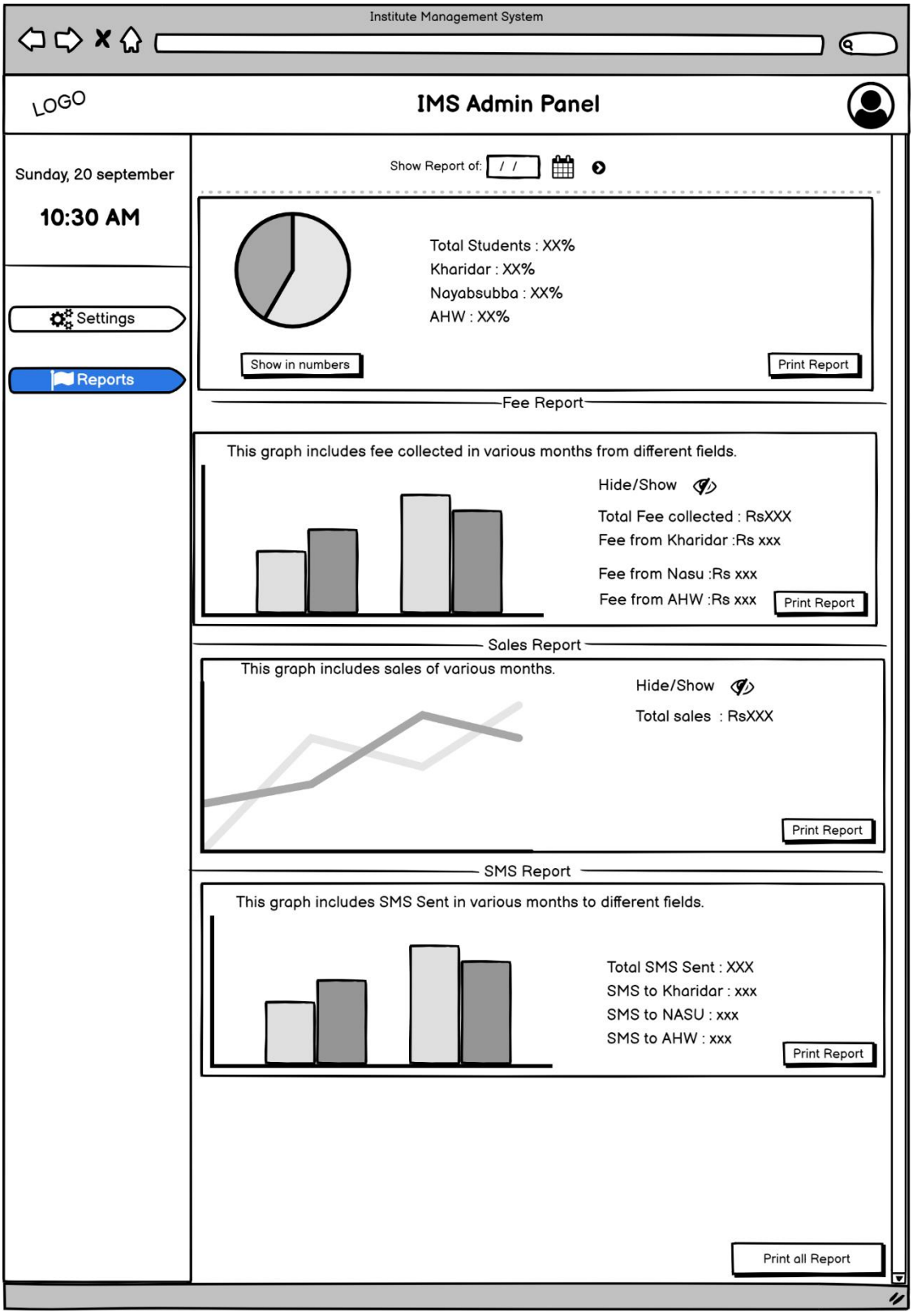

Figure 14 Report Page

3.5. ER diagram

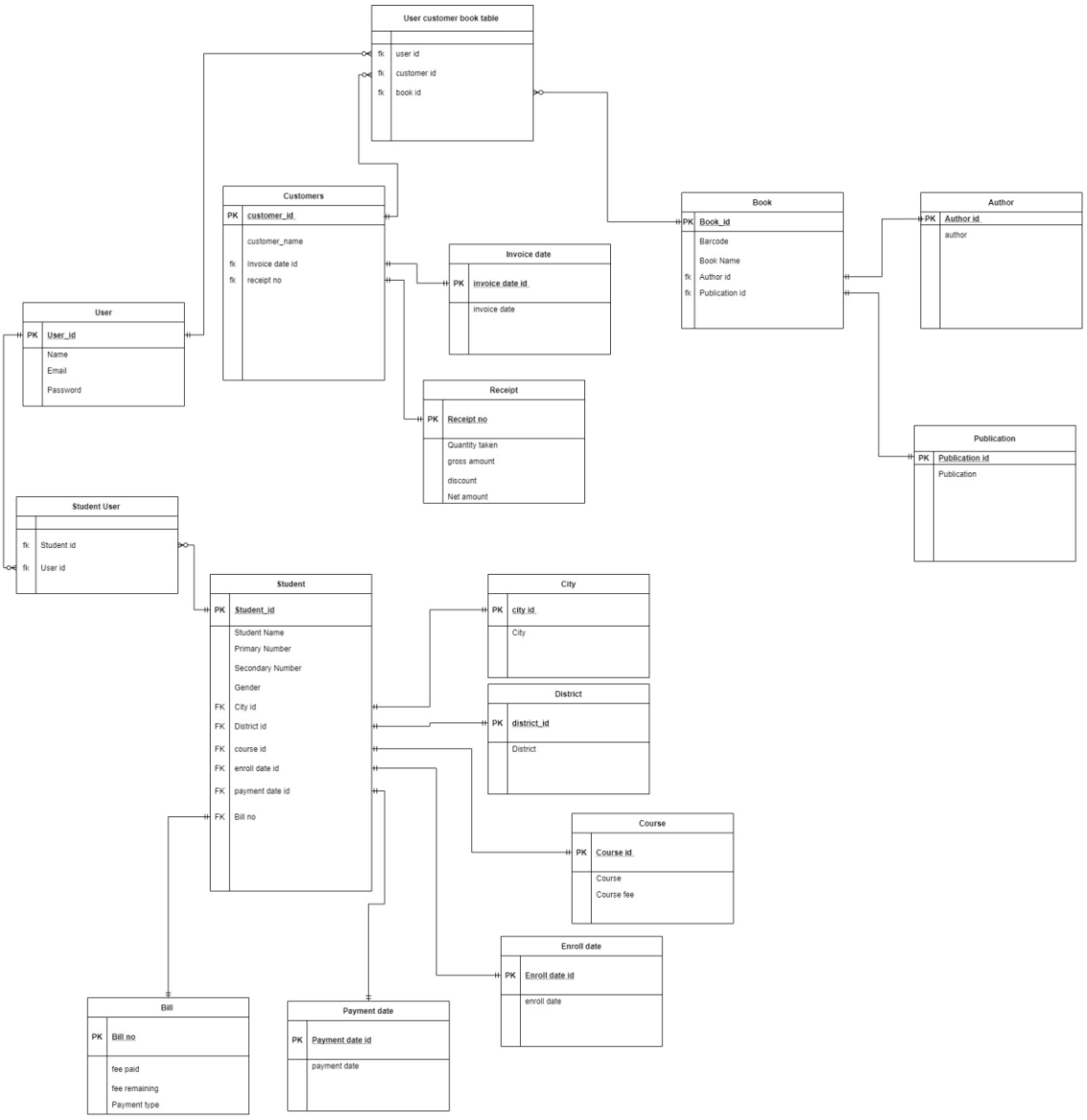

Figure 15 ER Diagram

3.6. Web application development

3.6.1. Staff Login Page

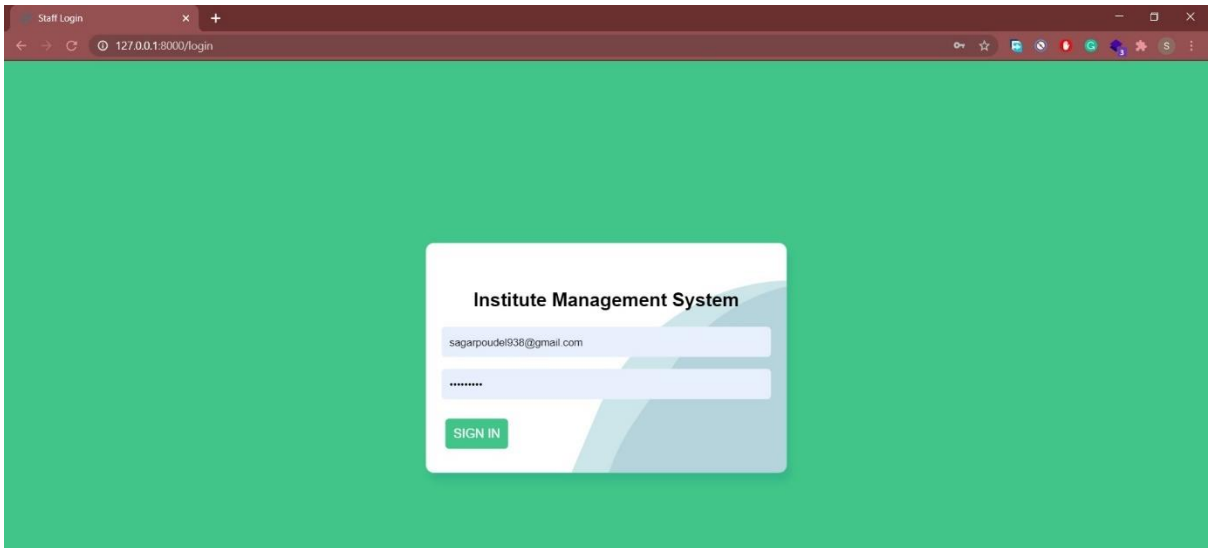

Figure 16 Staff Login Page

3.6.2. Admin Login Page

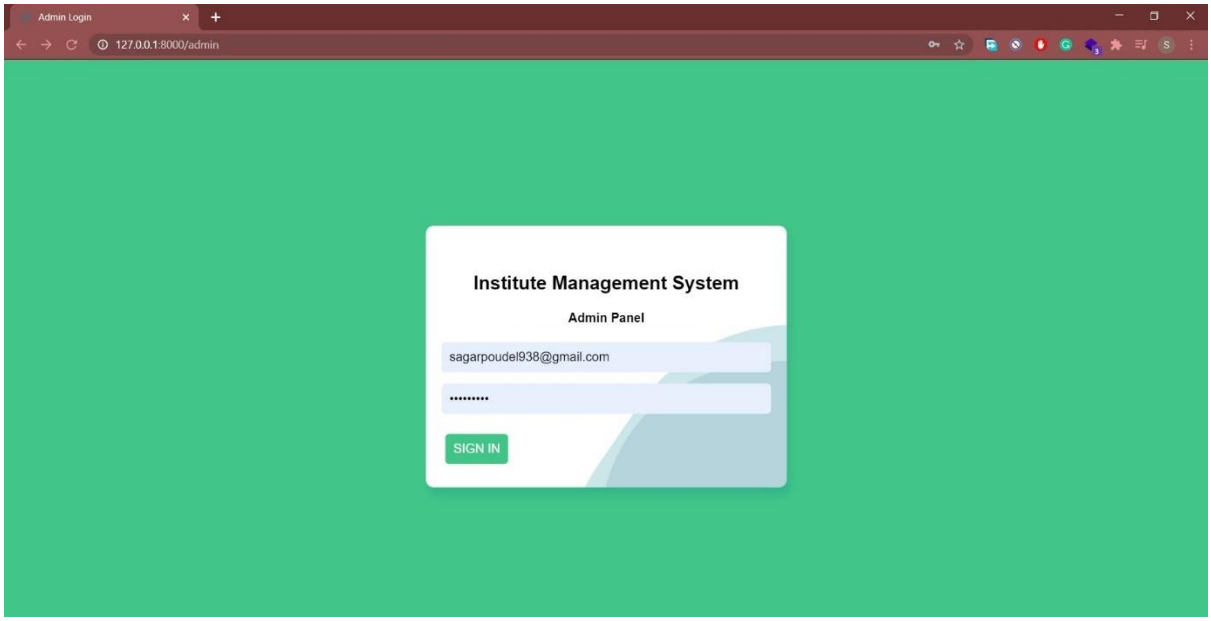

Figure 17 Admin Login Page

Please refer Appendix A for Source Code.

3.7. Gantt Chart

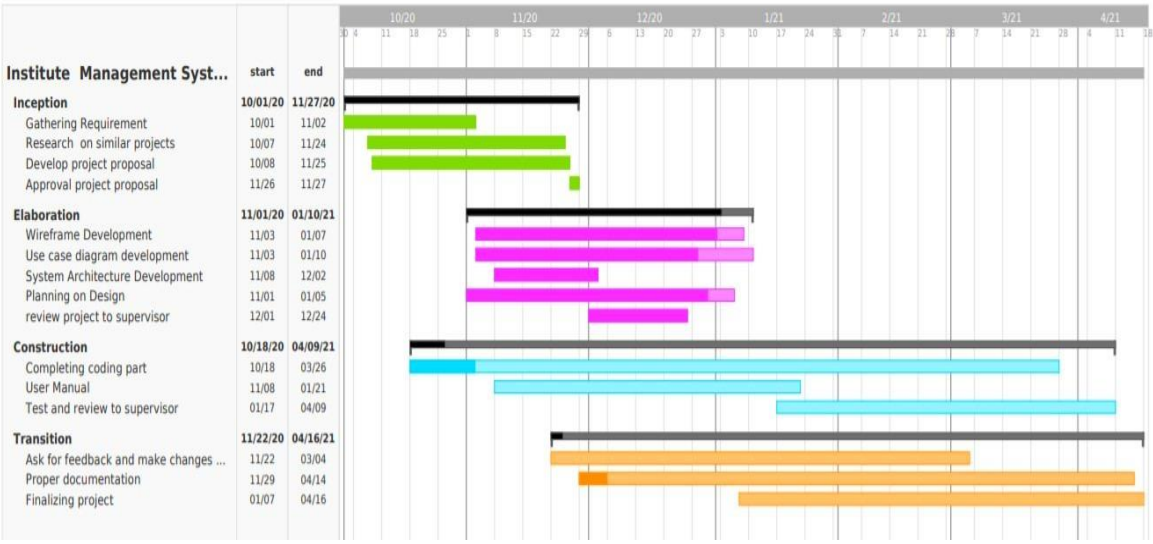

Figure 18 Gantt Chart

Tabular form of Gantt chart is provided below:

Activities	Start	End	Days
Inception	**10/1/2020**	**11/27/2020**	
Gathering Requirement	1-Oct	2-Nov	32
Research on Similar Projects	7-Oct	24-Nov	48
Develop Project Proposal	8-Oct	25-Nov	48
Approval Project Proposal	26-Nov	27-Nov	1
Elaboration	**11/1/2020**	**1/10/2021**	
Wireframe Development	3-Nov	7-Jan	65
Use case Diagram Development	3-Nov	10-Jan	68
System Architecture Development	8-Nov	2-Dec	24
Planning on Design	1-Nov	05-Jan	65
Review Project To supervisor	1-Dec	24-Dec	23
Construction	**10/18/2020**	**4/9/2021**	
Completing Coding Part	18-Oct	26-Mar	159
User Manual	8-Nov	21-Jan	74
Test And Review to Supervisor	17-Jan	9-Apr	82
Transition	**11/22/2020**	**4/16/2021**	
Ask for Feedback and Make changes	22-Nov	4-Mar	102
Proper Documentation	29-Nov	14-Apr	136
Finalizing Project	7-Jan	16-Apr	99

Table 15 Gantt chart

4. Analysis of progress

4.1. Progress table

S.N.	Tasks	Status	Progress(%)
1	Gathering Requirement	Completed	100
2	Research on Similar Projects	Completed	100
3	Develop Project Proposal	Completed	100
4	Approval Project Proposal	Completed	100
5	Wireframe Development	Almost completed	90
6	Use case Diagram Development	Almost completed	80
7	System Architecture Development	Completed	100
8	Planning on Design	Almost completed	90
9	Review Project To supervisor	Completed	100
10	Completing Coding Part	Initialized	10
11	User Manual	Incomplete	0
12	Test and Review to Supervisor	Incomplete	0
13	Ask for Feedback and Make changes	Incomplete	0
14	Proper Documentation	Initialized	5
15	Finalizing Project	Incomplete	0

Table 16 Progress Table

Please refer to Appendix B for Milestone Review

4.2. Progress Review and Action Plan

As in above progress table, requirement gathering, research on similar projects, proposal development, system architecture development is completed in time as planned in Gantt chart submitted in project proposal. Design part such as wireframe development, use case development, planning on design are delayed but almost completed. Things that didn't go as per plan is coding part. Coding part is not completed as it should be because of spending little more time in wireframe and use case development.

From now onwards, tasks will be carried out according to new Gantt chart which is submitted in interim report. Coding part will be given more priority as other tasks are going according to plan.

5. Future work

This part of report includes remaining tasks that are to be completed in future time. As per Gantt chart following things are future tasks:

i) Finalizing Wireframe Development
ii) Completing Use case development and design planning
iii) Preparation of User Manual
iv) Working on barcode system
v) Working on SMS API integration
vi) Completing Coding part
vii) Finalizing documentation
viii) Submission of Final year Project Report

6. References

ANWAR, A., 2014. A Review of RUP (Rational Unified Process). *International Journal of Software Engineering (IJSE),* 5(2), p. 13.

Christensson, P., 2006. *techterms.* [Online] Available at: https://techterms.com/definition/rup [Accessed 14 october 2020].

Kruchten, P., 2004. *The Rational Unified Process: An Introduction.* 3rd ed. s.l.:Addison-Wesley Professional.

K, S., Samanekar, A. & Pujar, N. T., 2018. Student Management System. *INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY,* VI(15), pp. 1-4.

Ros, F., 2019. *code-projects..* [Online] Available at: https://code-projects.org/student-record-management-system-in-php-with-source-code/ [Accessed 23 12 2020].

S.Balaji & Murugaiyan, D., 2012. WATEERFALLVs V-MODEL Vs AGILE: A COMPARATIVE STUDY ON SDLC. *International Journal of Information Technology and Business Management,* II(1), pp. 26-30.

S.R.Bharamagoudar, R.B., G. & S.G.Totad, 2013. Web Based Student Information Management. *International Journal of Advanced Research in Computer and Communication Engineering,* II(6), pp. 2342-2348.

Scott, K., 2001. *informit.* [Online] Available at: https://www.informit.com/articles/article.aspx?p=24671&seqNum=7 [Accessed 14 october 2020].

t4tutorials, 2020. *t4tutorials.* [Online] Available at: https://t4tutorials.com/rational-unified-process-rup-phases-of-rup-inception-elaboration-construction-transition-software-engineering/#What_is_the_elaboration_phase [Accessed 14 october 2020].

7. Appendix A

StaffLogin.blade.php

resources > views > StaffLogin.blade.php > html > body > form.login

```
<!DOCTYPE html>
<html>
<head>
    <title>Staff Login</title>
    <link rel="stylesheet" href="/css/login.css">

</head>
<body>
    <form class="login">
        <center><h2>Institute Management System</h2></center>

        <input type="text" name="email" placeholder="Email">

        <input type="password" name="password" placeholder="Password">
        <button type="submit">Sign In</button>
    </form>
</body>
</html>
```

Figure 19 Staff Login Source Code

AdminLogin.blade.php

resources > views > AdminLogin.blade.php > html > body > form.login

```
<!DOCTYPE html>
<html>
<head>
    <title>Admin Login</title>
    <link rel="stylesheet" href="/css/login.css">

</head>
<body>
    <form class="login">
        <center><h2>Institute Management System</h2></center>
        <center><h4>Admin Panel</h4></center>

    <input type="text" name="email" placeholder="Email">


    <input type="password" name="password" placeholder="Password">
    <button type="submit">Sign In</button>
    </form>
</body>
</html>
```

Figure 20 Admin Login Source Code

Login.scss

resources > sass > login.scss > .login

```scss
@import "bourbon";

// Colors
$greenSeaweed: rgba(2, 128, 144, 1);
$blueQueen: rgba(69, 105, 144, 1);
$redFire: rgb(66, 197, 136);

// Fonts
$fontAsap: 'Asap', sans-serif;

body {
  background-color: $redFire;
  font-family: $fontAsap;
}

.login {
  overflow: hidden;
  background-color: white;
  padding: 40px 30px 30px 30px;
  box-shadow: 5px 10px 10px rgba($greenSeaweed, 0.2);

  &::before, &::after {
    content: '';
    position: absolute;
    width: 600px;
    height: 600px;
    border-top-left-radius: 40%;
    border-top-right-radius: 45%;
    border-bottom-left-radius: 35%;
    border-bottom-right-radius: 40%;
```

Attaches one or more drop-shadows to the box. The property is a comma-separated list of shadows, each specified by 2-4 length values, an optional color, and an optional 'inset' keyword. Omitted lengths are 0; omitted colors are a user agent chosen color.

Syntax: none | <shadow>#

MDN Reference

```
    z-index: -1;
  }

  &::before {
    left: 50%;
    bottom: -130%;
    background-color: rgba($blueQueen, 0.15);
    animation: wawes 6s infinite linear;
  }

  &::after {
    left: 40%;
    bottom: -125%;
    background-color: rgba($greenSeaweed, 0.2);
    animation: wawes 7s infinite;
  }

  > input {
    font-family: $fontAsap;
    display: block;
    border-radius: 5px;
    font-size: 16px;
    background: white;
    width: 100%;
    border: 0;
    padding: 10px 10px;
    margin: 15px -10px;
  }

  > button {
    font-family: $fontAsap;
    cursor: pointer;
    color: #fff;
    font-size: 16px;
    text-transform: uppercase;
    width: 80px;
    border: 0;
```

```
    padding: 10px 0;
    margin-top: 10px;
    margin-left: -5px;
    border-radius: 5px;
    background-color: $redFire;
     transition:background-color 300ms;

    &:hover {
      background-color: darken($redFire, 50%);
    }
  }
}

  @keyframes  wawes {
  from { transform:rotate(0); }
  to { transform:rotate(360deg); }
}
```

Figure 21 login.scss code

8. Appendix B

Milestone review is provided below:

- **Milestone 1:** Proposal Submission
  Status: Completed

- **Milestone 2:** Interim Report Submission
  Status: Only Submission remaining

- **Milestone 3:** UI design and wireframe
  Status: Almost Completed

- **Milestone 4:** Develop Use Case
  Status: Almost Completed

- **Milestone 5:** Development of Web application
  Status: Initialized

- **Milestone 6:** Testing
  Status: Incomplete

- **Milestone 7:** Documentation
  Status: Incomplete

- **Milestone 8:** Submit final Year project
  Status: Incomplete

9. Appendix C

Survey Findings

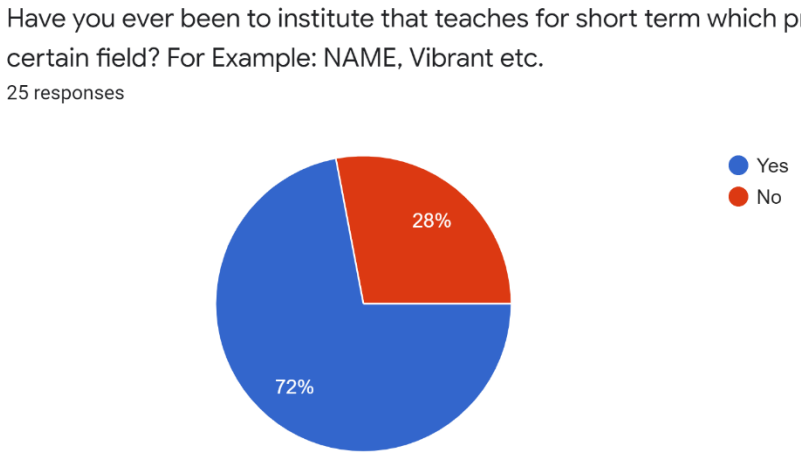

Figure 22 Survey Question 1

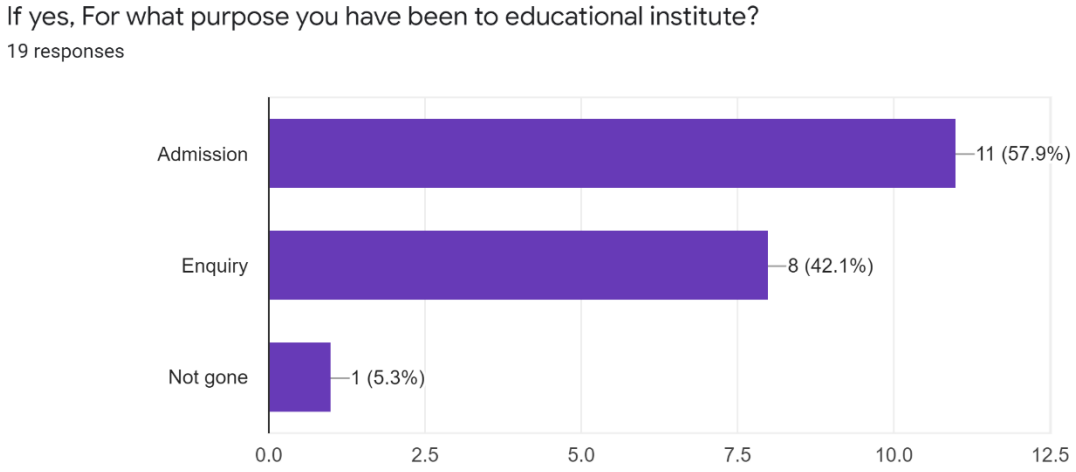

Figure 23 Survey Question 2

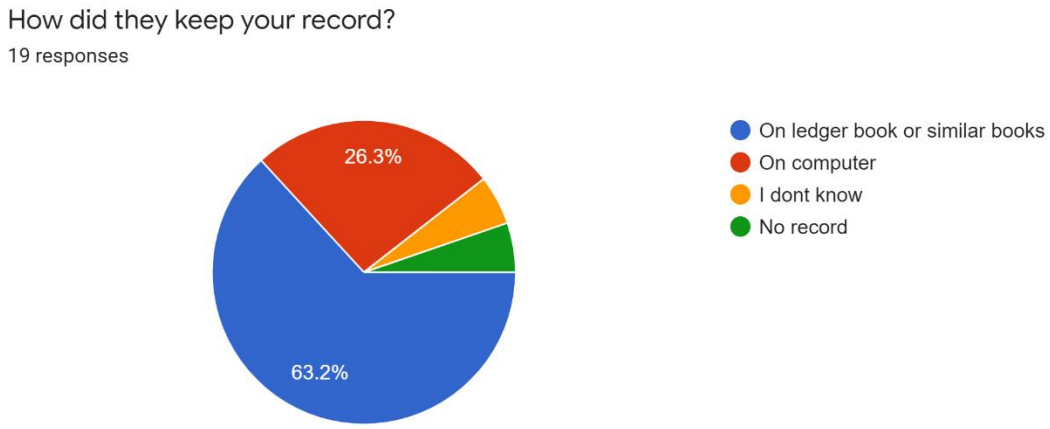

Figure 24 Survey Question 3

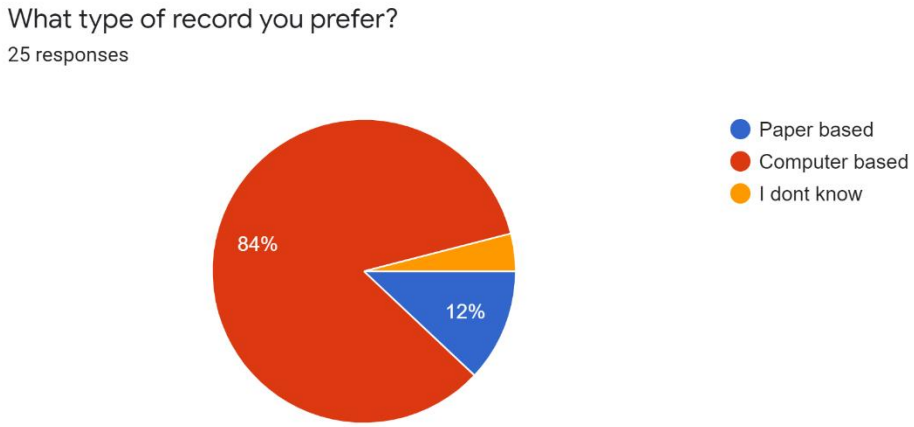

Figure 25 Survey Question 4

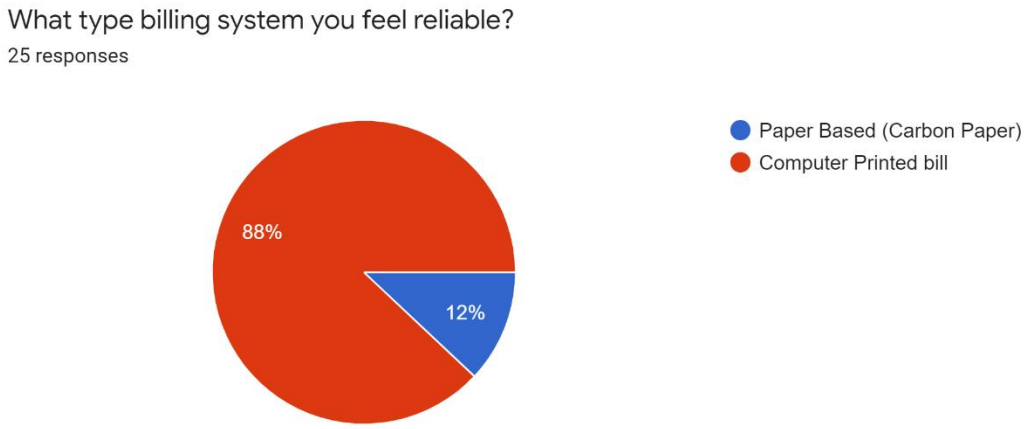

Figure 26 Survey Question 5

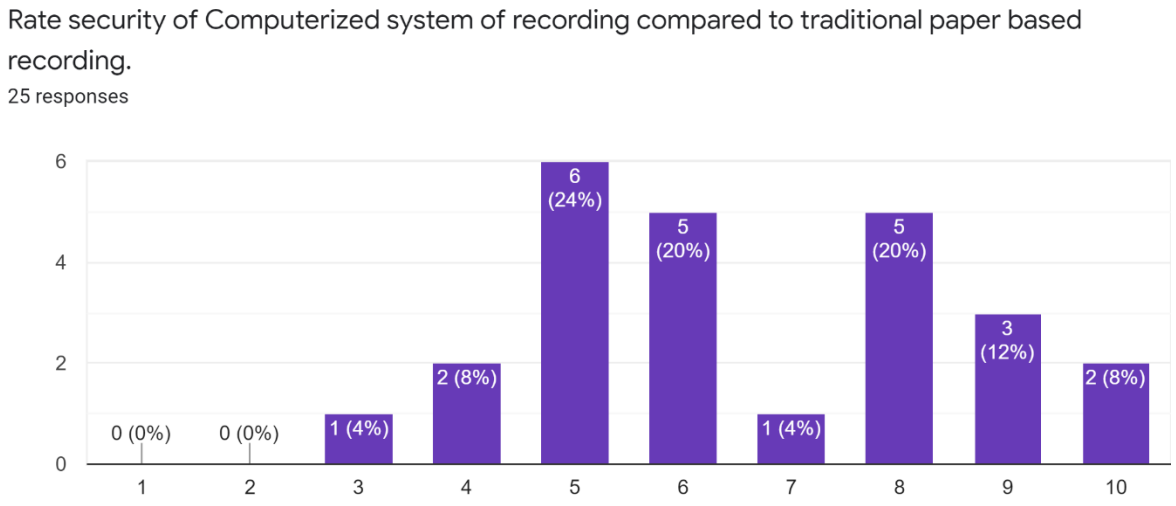

Figure 27 Survey Question 6

Give suggestion on increasing security for computerized system.

25 responses

Having Password Protection

Add Firewall

strong passwords.?

Password system; user login system

Strong passwords

Having strong password and asking for permission for any editing job

Keep password

Token authentication on each bill, secure encryption

Consulting a cyber security specialist while building the system

Install and Use Antivirus and Anti-spyware Software

Not to share pin and passwords.

Having strong password

Data privacy, authentic software or system

increase physical security as well as use of different protocol

Anything.

To improve security in computerized system at first network security should be established .

Use of cards for access should be minimized Access management should focus more on fingerprints and passcode access

Having OTP while login

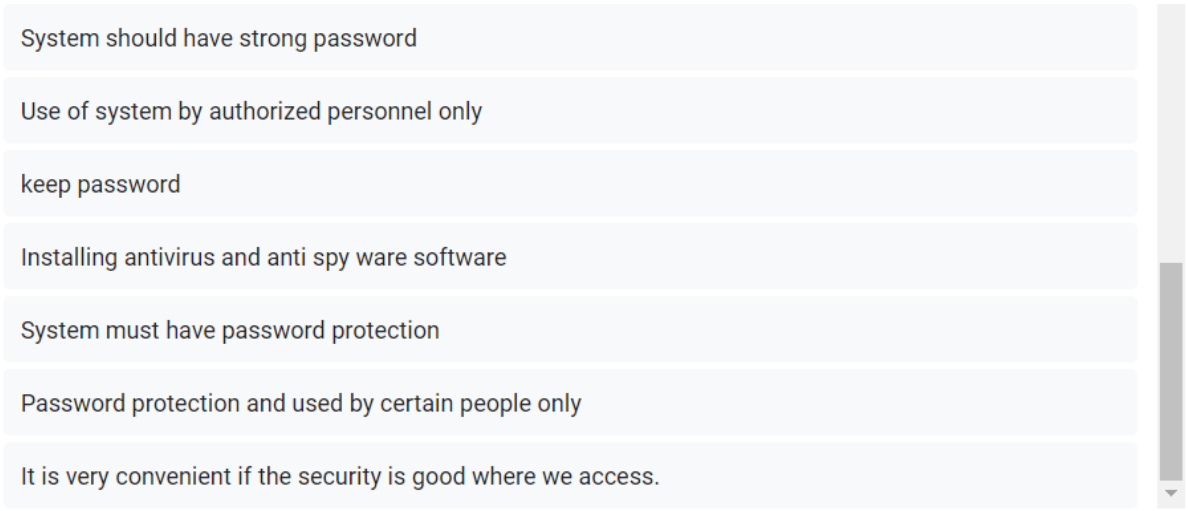

Figure 28 Survey Question 7

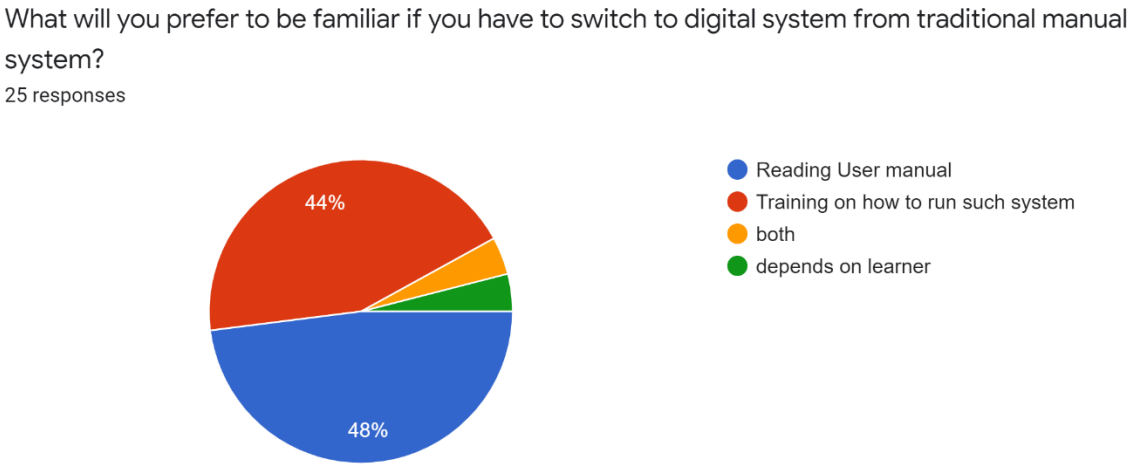

Figure 29 Survey Question 8