

**Senior Secondary School Management System**

**A case study of the Khomastura Secondary School**

**By**

**Ntinda Wellington T**

**202062775**

**Submitted in partial fulfilment of the requirements for the degree**

**BACHELOR OF SCIENCE IN SOFTWARE DEVELOPMENT**

**In the**

**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY**

**At the**

**INTERNATIONAL UNIVERSITY OF MANAGEMENT**

**Supervisor: Mrs. G Rukanda**

**Date of submission**

**28 October 2022**

# **DECLARATION**

I Wellington Ntinda hereby declare that the project work entitled ‘ Is to develop a senior secondary school management system’, ” submitted to the faculty of information and technology at the International University of Management (Dorado Campus), is a record of original work done by myself under the guidance and mentorship of Mrs. G. Rukanda Faculty Member of the International University of Management, and this project work has not performed the basis for the award of any degree/ fellowship and similar project if any.

|  |  |  |
| --- | --- | --- |
| **Student’s name** | **Signature** | **Date** |
| Ntinda W.T | ……….…………… | …………… |
| **Supervisor’s name** | **Signature** | **Date** |
| Mrs. G Rkanda | ……..……………… | ……….….. |

# **ABSTRACT**

This project work automates school management system. In the system one applications is developed, Web based (thin client).

The windows application takes most of the activities such as offline student registering, transcript and report card generation and producing the timetable. The web application facilitates attendance recording by the homeroom teachers, to view status of students by their parents and to view reports.

Our solution of the timetable is very simple. In the high school considered for the project there are ten subjects for both grade nine and grade ten. Loads are assigned to each subject teacher and a code is given for each teacher-subject combination. A simple search technique has been used during allocation of each teacher-subject code to a time slot. A database has been used to enforce constraints and to store data.

# **ACKNOWLEDGEMENTS**

I would like to express my gratitude to my advisor Mrs. G Rukanda for his guidance, support and his continuous enthusiasm and encouragement throughout the project. I am also very grateful and extend my sincere thanks to the principals and staff members of the department of Science at Khomastura Secondary School for their cooperation by sharing the load that I was teaching to make me have time to work on this project and throughout my study.

Finally, many thanks to friends, who have helped and given me suggestions, supports and corrections throughout the project.

**TABLE OF CONTENT**

[**DECLARATION** i](#_Toc117769147)

[**ABSTRACT** ii](#_Toc117769148)

[**ACKNOWLEDGEMENTS** ii](#_Toc117769149)

[**LIST OF TABLES** vi](#_Toc117769150)

[**LIST OF FIGURES** vi](#_Toc117769151)

[**CHAPTER ONE:** 1](#_Toc117769152)

[**INTRODUCTION AND BACKGROUND OF THE STUDY** 1](#_Toc117769153)

[**1.1** **INTRODUCTION** 1](#_Toc117769154)

[**1.2** **BACKGROUND STUDY** 1](#_Toc117769155)

[**1.3** **PROBLEM STATEMENT** 2](#_Toc117769156)

[2 1.2 Problem statement 2](#_Toc117769157)

[**2.1** **OBJECTIVES AND AIMS OF THE STUDY** 3](#_Toc117769158)

[**2.2** **DELIMITATIONS OF THE STUDY** 3](#_Toc117769159)

[**2.3** **PROJECT CONSTRAINTS/ LIMITATIONS OF THE STUDY** 4](#_Toc117769160)

[**2.4** **ETHICAL CONSIDERATIONS** 4](#_Toc117769161)

[**2.5** **CONCLUSION** 4](#_Toc117769162)

[**CHAPTER TWO:** 5](#_Toc117769163)

[**LITERATURE REVIEW** 5](#_Toc117769164)

[**3.0** **INTRODUCTION** 5](#_Toc117769166)

[4 Observed Products 5](#_Toc117769167)

[5 Manual Timetabling 6](#_Toc117769168)

[6 6](#_Toc117769169)

[7 Drawbacks of the Reviewed Systems 6](#_Toc117769170)

[**7.3** **CONCLUSION** 8](#_Toc117769171)

[**CHAPTER THREE:** 9](#_Toc117769172)

[**ANALYSIS AND DESIGN** 9](#_Toc117769173)

[**3.** **INTRODUCTION** 9](#_Toc117769175)

[**3.2** **SOFTWARE DEVELOPMENT METHOD** 9](#_Toc117769176)

[**3.3** **REQUIREMENT SPECIFICATION** 10](#_Toc117769177)

[**3.4** **SYSTEM DESIGN** 12](#_Toc117769178)

[**3.5** **USER INTERFACES** 14](#_Toc117769179)

[3.5.1 Welcome/Index Page (login) 15](#_Toc117769180)

[3.5.2 Welcome/dashboard Page (users/students) 15](#_Toc117769181)

[3.5.3 User interface parents 16](#_Toc117769182)

[**3.6** **Database Design** 18](#_Toc117769183)

[3.6.1 ER Diagram of database 18](#_Toc117769184)

[3.6.2 Table 1. Admin Table 18](#_Toc117769185)

[3.6.3 Table 2. Student table 19](#_Toc117769186)

[3.6.4 Table 3. Parents table 19](#_Toc117769187)

[3.6.5 Table payements 20](#_Toc117769188)

[**3.7** **Hardware and software Requirements** 21](#_Toc117769189)

[3.7.1 **Hardware requirements** 21](#_Toc117769190)

[**3.7.2** **Software requirements** 21](#_Toc117769191)

[3.8 CONCLUSION 21](#_Toc117769192)

[**CHAPTER FOUR:** 22](#_Toc117769193)

[**SYSTEM IMPLEMENTATION** 22](#_Toc117769194)

[**4.2** **INTRODUCTION** 22](#_Toc117769196)

[**4.3** **ALGORITHM** 22](#_Toc117769197)

[**4.4** **CODES** 23](#_Toc117769198)

[**4.4.1** **Login Page** 23](#_Toc117769199)

[**4.5** **Conclusion** 23](#_Toc117769200)

[**CHAPTER FIVE** 24](#_Toc117769201)

[**SYSTEM TESTIMG** 24](#_Toc117769202)

[**INTRODUCTION** 24](#_Toc117769204)

[**5.1** 24](#_Toc117769205)

[**5.2.1** **Login Test** 24](#_Toc117769206)

[**CHAPTER SIX:** 24](#_Toc117769207)

[**CONCLUSION AND FUTURE WORK** 24](#_Toc117769208)

[**6.1 CONCLUSION** 24](#_Toc117769209)

[**6.2 FUTURE WORK** 24](#_Toc117769210)

# **LIST OF TABLES**

[3.6.2 Table 1. Admin Table 18](#_Toc117769211)

[3.6.3 Table 2. Student table 19](#_Toc117769212)

[3.6.4 Table 3. Parents table 19](#_Toc117769213)

[3.6.2 Table 1. Admin Table 18](#_Toc117769214)

[3.6.3 Table 2. Student table 19](#_Toc117769215)

[3.6.4 Table 3. Parents table 19](#_Toc117769216)

**CHAPTER** **ONE:**

**INTRODUCTION AND BACKGROUND OF THE STUDY**

## **INTRODUCTION**

This chapter focused on the background of the study, motivation, problem statement, objectives of the study, research questions of the study, delimitations of the study and the limitations of the study.

## **BACKGROUND STUDY**

Education system forms the backbone of every nation. And hence it is important to provide a strong educational foundation to the young generation to ensure the development of open-minded global citizens securing the future for everyone. Advanced technology available today can play a crucial role in streamlining education-related processes to promote solidarity among students, teachers, parents and the school staff.

Education is central to development. It is one of the most powerful instruments for reducing poverty and inequality and lays a foundation for sustained economic growth. With this aim currently our government has given special emphasis to the educational sector and school improvement activities such as continuous professional development for teachers, training and upgrading teachers and capacitating schools with manpower and materials are among the major actions which have been taken in both primary and secondary schools. In order to facilitate and simplify these actions one of the major tool is to have automated school management system.

School Management System (SMS) consists of tasks such as registering students, attendance record keeping to control absentees, producing report cards, producing official transcript, preparing timetable and producing different reports for teachers, parents, officials for Khomastura Secondary School and other stakeholders.

Automation is the utilization of technology to replace human with a machine that can perform more quickly and more continuously. By automating SMS documents that took up many large storage rooms can be stored on few disks. Transcript images can be annotated. It reduces the time to retrieve old transcripts from hours to seconds. However, most of the school system in the government schools of Namibia is not automated and the record officers generate transcripts and reports manually and the school administrators use their experienced knowledge of miss and hit approaches to prepare timetables.

## **PROBLEM STATEMENT**

# 1.2 Problem statement

In recent times, the statistic by the ministry of Education has shown that there had been problems of admission that it can only be done within a limited time compared to online system which was seen or discovered to be a solution to all difficulties of recent systems which are still being used by the ministry of education at large and all schools which saves time and can be done in the comfort of your own (Davis, 2021).

Due to the inefficiency of the current manual system, the need arises to automate SMS in order to efficiently handle students’ attendance, to produce transcript, report cards and the various reports satisfying users and customers and to produce timetable which can schedule courses for teachers and classes of students.

Past years up to date the entirely ministry at different circuits and regions, all schools without mentioning, have been stressing using pen and a paper known as early method of recording applicants to save their information also known as traditional or file-based system (FBS,2019). With the use of this old system, a maximum time is required, and documents and records can be easily misplaced or even get lost (E. Marten). An online portal with the database system will reduce working hours and data lost and redundancy on the ministry information systems.

Teachers may want to associate a student with his parent or emergency persons for disciplinary measures which need searching of the students record in the record office. It has been difficult to search a record from thousands of such records and observed that students can take any person claiming that he/she is their parent or emergency person which creates problem in control of students.

Waste of materials like forms and other sources is one of the biggest issues during registration as papers or forms are wasted or torn when wrong information was recorded. Introduction for an online application or admission system to schools in the Khomas region will be both beneficial for the region at large, the ministry and the applicants. This helps with keeping up records, the number of applicants per school and the required number of admissions per school, unlike the manual system where papers may be lost. The online will have proper systems in place to save the whole application and admission process and have the back up if data were deleted somehow. It will also save people from standing in long lines at schools or education offices as people will now be able to do it from their homes.

Some existing systems that were discovered, e-learning management system, moodle which seem not to be available to senior secondary schools Shikesho (2020), and the big blue button, Banner SIS which is not user friendly (Ellucian, 2021).

.

## **OBJECTIVES AND AIMS OF THE STUDY**

The purpose of the study is to analyse the current management system and to design and develop a new online student management system.

* To map and establish the need of the study
* To facilitate attendance record keeping,
* To facilitate various report generation,
* Access to personal information (timetable, upcoming events, marks, report cards and financial statement) online.
* To reduce the time taken by teachers during registration on the current systems
* Discussions, chats, and forums
* To facilitate attendance record keeping
* To facilitate various report generation
* To allow teachers, parents, school community and Education officials to view reports on students, ̇

## **DELIMITATIONS OF THE STUDY**

* Blacklisting and blockage of users with bad or poor interaction with the system.
* The system will only be operating in English language, but not a multilingual system

## **PROJECT CONSTRAINTS/ LIMITATIONS OF THE STUDY**

The system cannot manage the controlling of how many times people may log in the system.

Network failure could be a problem since the system is only accessible when there is internet, and if the networks is very poor the connection might be slow and it might affect the system. The developer cannot control how users uses the system since the developer only have positive energy about the system operation

## **ETHICAL CONSIDERATIONS**

* Honesty- Honestly report data and results, will not fabricate or falsify data.
* Objectivity- Avoid bias or self-deception.
* Accountability- Will take responsibility for any inconvenience caused and try to justify why particular actions might be taken.
* Intellectual property- The researcher will not pass of other researchers work as his own

## **CONCLUSION**

In this proposal the system developer discussed the project’s background, problem statement, aims and objectives, methodology, delimitation, limitation, and research ethics. With the above proposed project, the researcher proposes to develop a management information system for senior secondary schools.

**CHAPTER TWO:**

**LITERATURE REVIEW**



## **INTRODUCTION**

Automated management system plays a great role in simplifying the job of employees at the school and satisfying the need of customers and stakeholders of the school. Even though no documentation is found in Namibia to be reviewed, products have been observed at some schools to help understand the problem of managing schools and handling school data. This chapter reviews these products.

# Observed Products

In the recent years, the Khomas region was very much interested to have automated school management system to get uniform and quick access to the students’ data for administrative purpose on promoting the students’ achievement and related issues. The Khomastura high school has been selected for the pilot test. At the time the school principals together with officials from Khomastura signed a contractual agreement with some software developer company. The developers installed their first version of the product which can register a student offline and generate official transcript with some level of difficulty. As the system is not fully automated, it does not support management of attendance, does not support generating report cards and other important functions such as generating school timetable and a web-based report for parents. Due to the lack of follow up by the government officials at Khomastura, the company was unable to complete the project. The school currently is unable to use the partially developed system because of lack of trained person and lack of hardware and software maintenance.

Another product that is in use is transcript generator system. The transcript generator system at Khomastura Secondary School generates official transcript of students. To generate transcript, the record officer enters the student information along with the grade marks for the grades completed per year and per semester. Then the system generates the required official transcript. Currently the school is using the system to generate official transcript even though the data entry format has unnecessarily many fields which are not applicable for the record office but can be used for continuous assessment by the course teacher.

# Manual Timetabling

# 

Manual timetables are prepared by dedicated teachers. In manual timetabling, it is common to proceed in an iterative fashion where each iteration selects and schedules a lesson. Scheduling a lesson requires to choose a classroom (fixed for each section of students) and a time slot such that the commitment to the choice will not violate any constraint. In school timetabling, we are required to schedule a given set of meetings such that the resulting timetables are feasible and acceptable to all people involved. Humans can prepare the timetable using some hit/miss approach. So, it is possible to automate the timetable based on a simulation of the human way of solving the problem. Such techniques, that we call direct heuristics, were based on a successive augmentation. That is, a partial timetable is extended, lecture by lecture, until all lectures have been scheduled. The underlying idea of all approaches is to schedule the most constrained lecture first.

Usually, some responsible teachers are assigned to schedule subjects and teachers. The number of teachers available per each subject is predefined and the load that each teacher has is calculated. With these data the timetable constructor assigns each teacher-subject association to the appropriate classes with the available time slots.

The manual solution of the timetabling problem usually requires many person-days of work. In addition, the solution obtained may be unsatisfactory. The lessons should be fairly distributed satisfying the identified constraints.

# Drawbacks of the Reviewed Systems

The reviews described have the following problems:

• Generate official transcript with some level of difficulty,

• Do not sufficiently produce the required reports to allow parents to view status of their children and reports for officials to help them participate in decision making,

• Do not generate timetable for the schools

• Do not facilitate attendance record keeping by the homeroom teachers

This project work tries to fill the gap by automating the various activities at schools. It tries to satisfy customers need and simplify the works of administrators, record officer and teachers. With an automated school management system parents can easily interact with the school community to follow up their children’s achievement and play their role in the school development processes.

## **CONCLUSION**

This chapter looked at the literature review of other African School management systems and how they manage their online learning systems. It includes different existing system and consider how to run a school management system with the example from the existing system.

**CHAPTER THREE:**

**ANALYSIS AND DESIGN**



## **INTRODUCTION**

To develop the best system to the school, there are some stages necessary for developing of new system include below features Gather information, design, implementation, coding, final testing and so on Within these few sections, deferent tactics will be adopted so that we can design a system that can maintain high usability and accessibility

3.1 **REQUIREMENT GATHERING**

During requirement gathering the researcher investigate the tasks and goals of targeted audiences to establish functional and usability objectives for the system or website. Although there are several methods of data collection, the researcher chooses to use two sources of data collection in carrying out their study, they are:

1. Interview

2. Questionnaire

The interview refers to the source of collecting data whereby the participant asks questions and the other provides the answers. Here the researcher approach one of the participants and ask them face to face. The questionnaire source of data of this research will be obtained by handing over question paper to some few participants for some hours for them to answer and provide feedback. This research includes the population of the Khomastura high school to be conducted, whereby random teachers and learner’s will be interviewed and given questionnaires.

## **SOFTWARE DEVELOPMENT METHOD**

In this research the researcher applies the prototyping method under the software development life cycle (SDLC) to develop the management system for the school. It used by system developers to produce information systems or software. Prototyping method is an experimental process where design teams implement ideas into tangible forms from paper to digital. It provides functionality and interactions between the system and the developer. It allows the developer to trace the faulty at early stage unlike other methods where you have to finish developing the project and detect the faults in the testing stage. It enhances collaboration within the team members that are involved in the development of the project. Sometimes moving back to the previous stage is necessary due to failure that occurs in the current stage. Even though the researcher chose to use prototyping method, it does not mean it is the best among all, prototyping method have some drawbacks one of them is that there is insufficient analysis which can make the developer to focus on limited resources and analyzing. Using prototyping method is quite expensive and time consuming.

Advantages of Prototype model:

* Users are actively involved in the development
* Since in this methodology a working model of the system is provided, the users get a better understanding of the system being developed.
* Errors can be detected much earlier.
* Quicker user feedback is available leading to better solutions.
* Missing functionality can be identified easily

Disadvantages of Prototype model:

* Leads to implementing and then repairing way of building systems.
* Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans.

## **REQUIREMENT SPECIFICATION**

**The proposed system has some of the features such as:**

* Functionality
* Usability
* Reliability
* Performance

**Functionality**

Firstly, allowing the end user to interact with the system through a rich interface provide a much more enjoyable user experience, particularly for the non-technical users which ~~will~~ account for the majority of the system’s users. In addition, this isolation layer also protects the integrity of the database by preventing users from taking any action outside those which the system is designed to handle. Because of this design pattern, it is essential to enumerate exactly which functions a user has presented. As the goal of the system is to make learning management as simple as possible for the teachers, parents and care givers, the functionality provided through the school management system is restricted to that which most pertinent to accomplish the desired task. All of the functions outlined above, with the exceptions of account creation and management, have been used a user logged in. By not including extraneous functions, I am moving towards my goal of simplifying the management system.

**Reliability**

The School management system is available only to Khomastura environment at the moment as the name suggests, allow them to manage the work load, fee management and others, that is displayed to users of the school management system. The functions afforded by the school management system provide user with the ability to, using a graphical interface like to add a new/ update /delete from the database.

**Usability**

Users of the school management system interact with the application through a series of simple menus. The menu includes bars with different options which can be clicked when then user wish to interact with the system depending on what activity they would want to perform. Each category of action has its own form associated with it which presents a drop-down menu for choosing which specific item from the category should be added to the selection, and a series of check boxes and radio buttons for selecting which options are to be included. Adding a query item is accomplished by a single button click.

**Performance**

User interaction with the system will be very simple. The application can automatically fetch new orders from the database at regular intervals and display the

## **SYSTEM DESIGN**

3.4.1 This system included some of the system design

Use case

Sequence diagram

Dataflow (DFD) Database interfaces

In this project, a use case diagram was used since it is the simplest representation of user’s interaction with the system, in this project it shows the relation between the restaurant and the customers and other additional users involved in online ordering system.

**Use case diagram**

**Student management system use diagram**

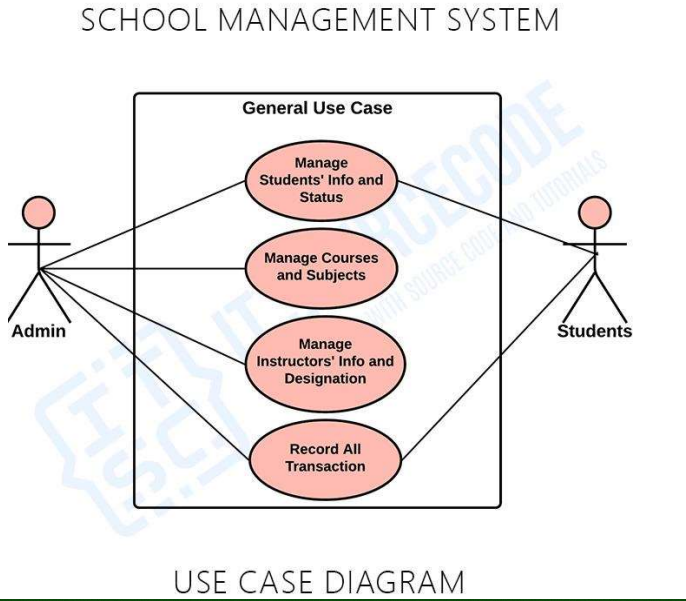


Figure 1. sms use-case diagram

Sequence Diagram

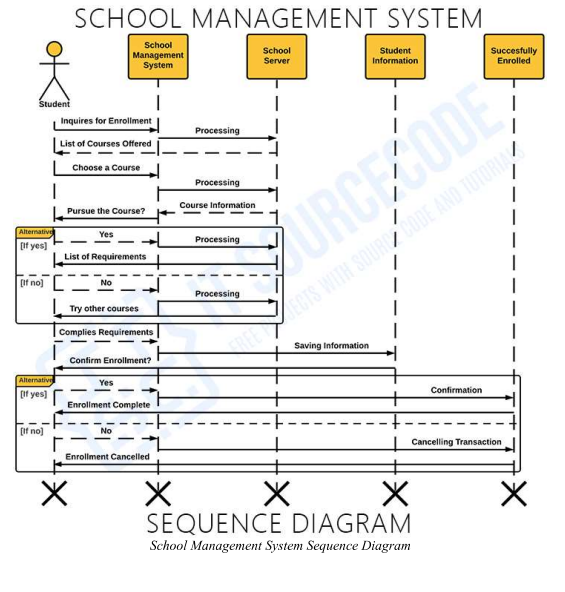


Figure 2. Sequence Diagram

## **USER INTERFACES**

### Welcome/Index Page (login)

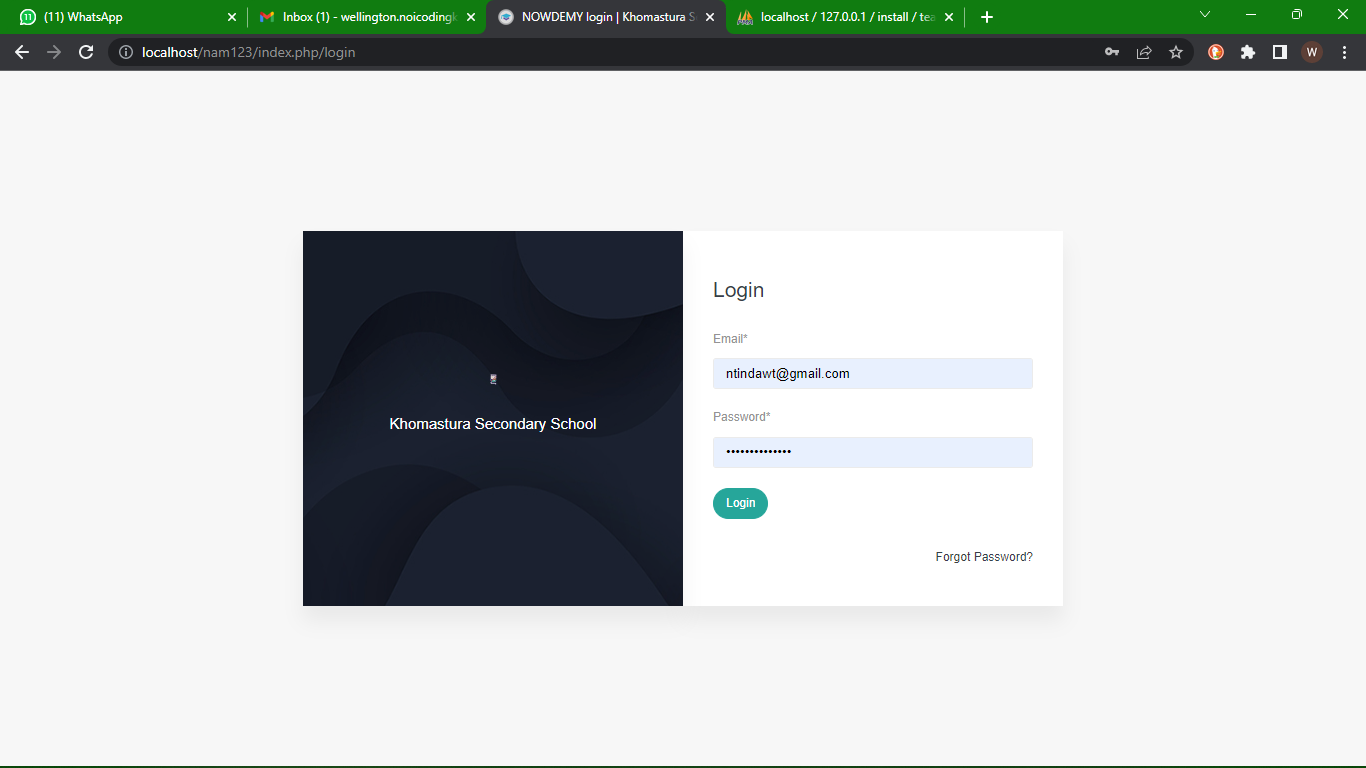


Figure 3. Welcome/ Index Page for admin

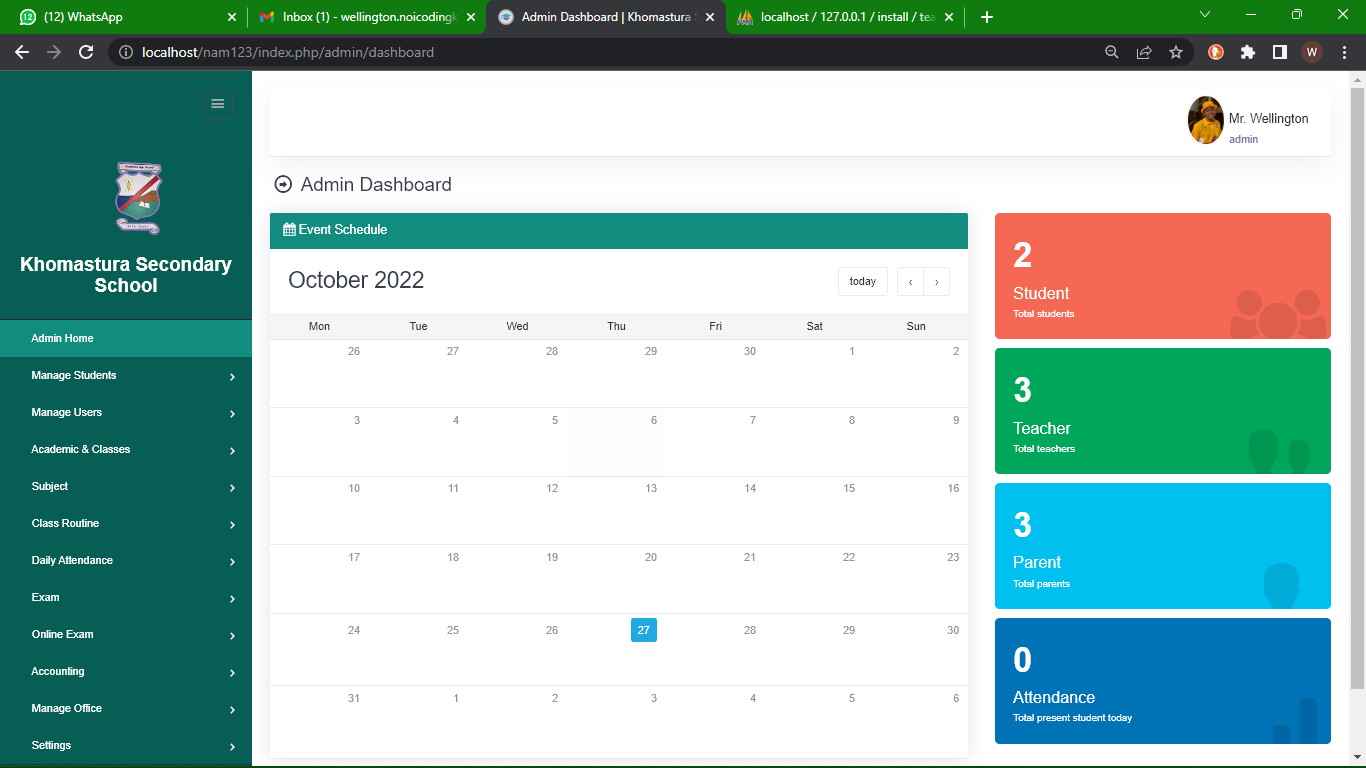


Figure 4. Welcome/ dashboard Page for admin

### Welcome/dashboard Page (users/students)

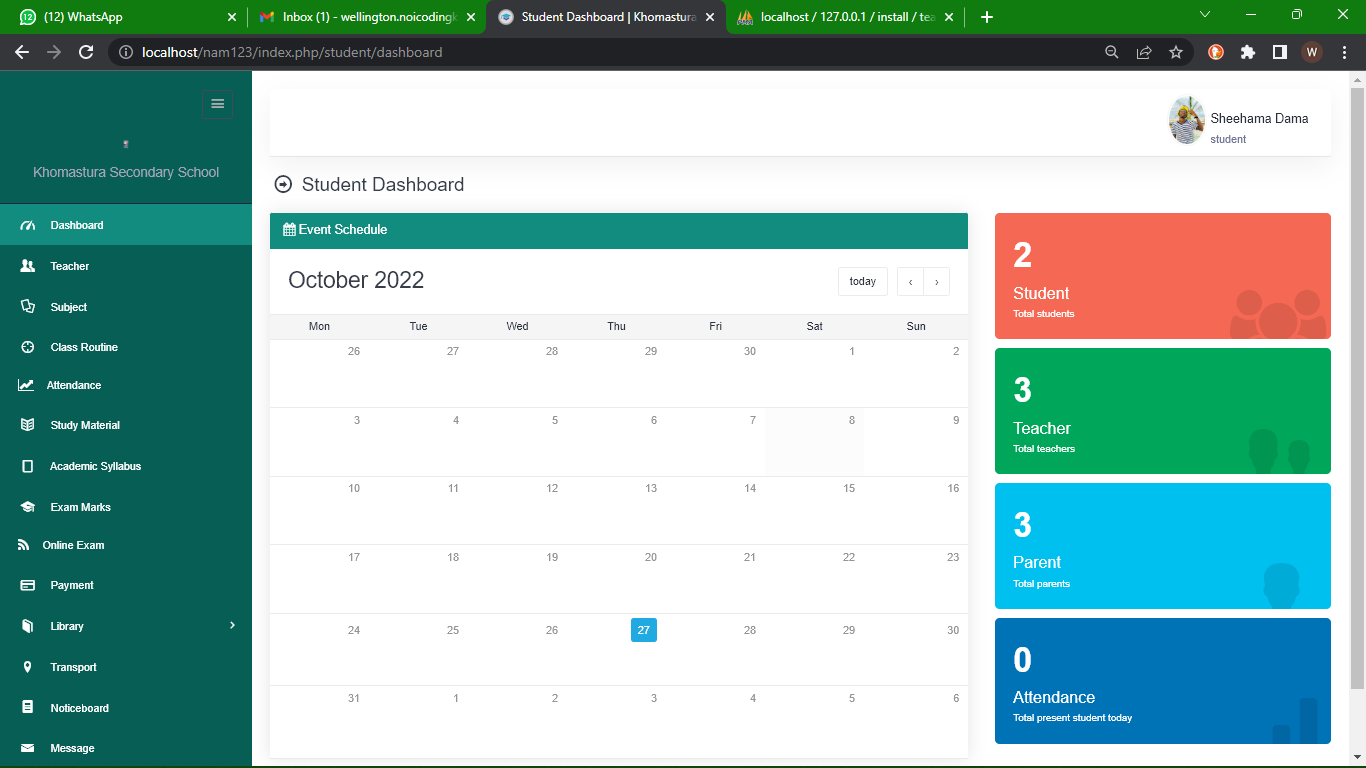


Figure 4. user student dashboard

### User interface parents

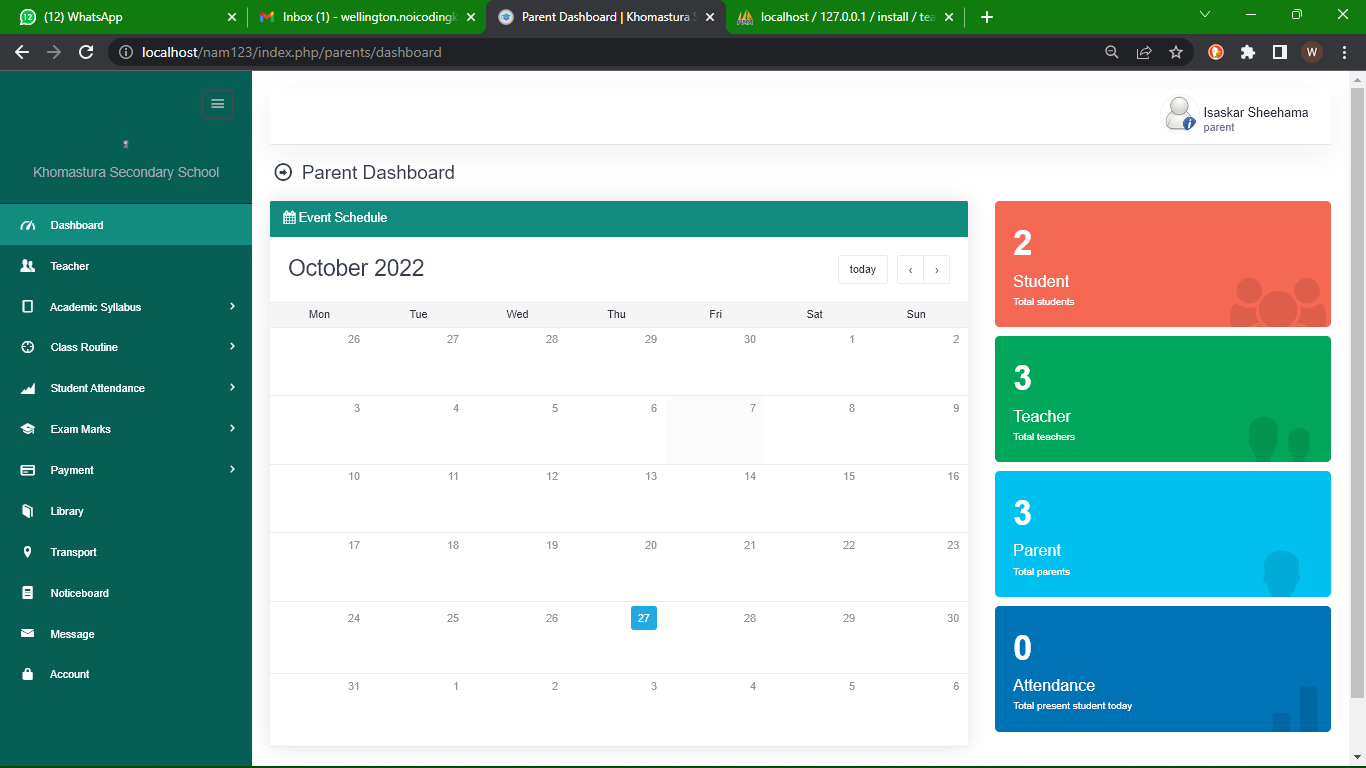
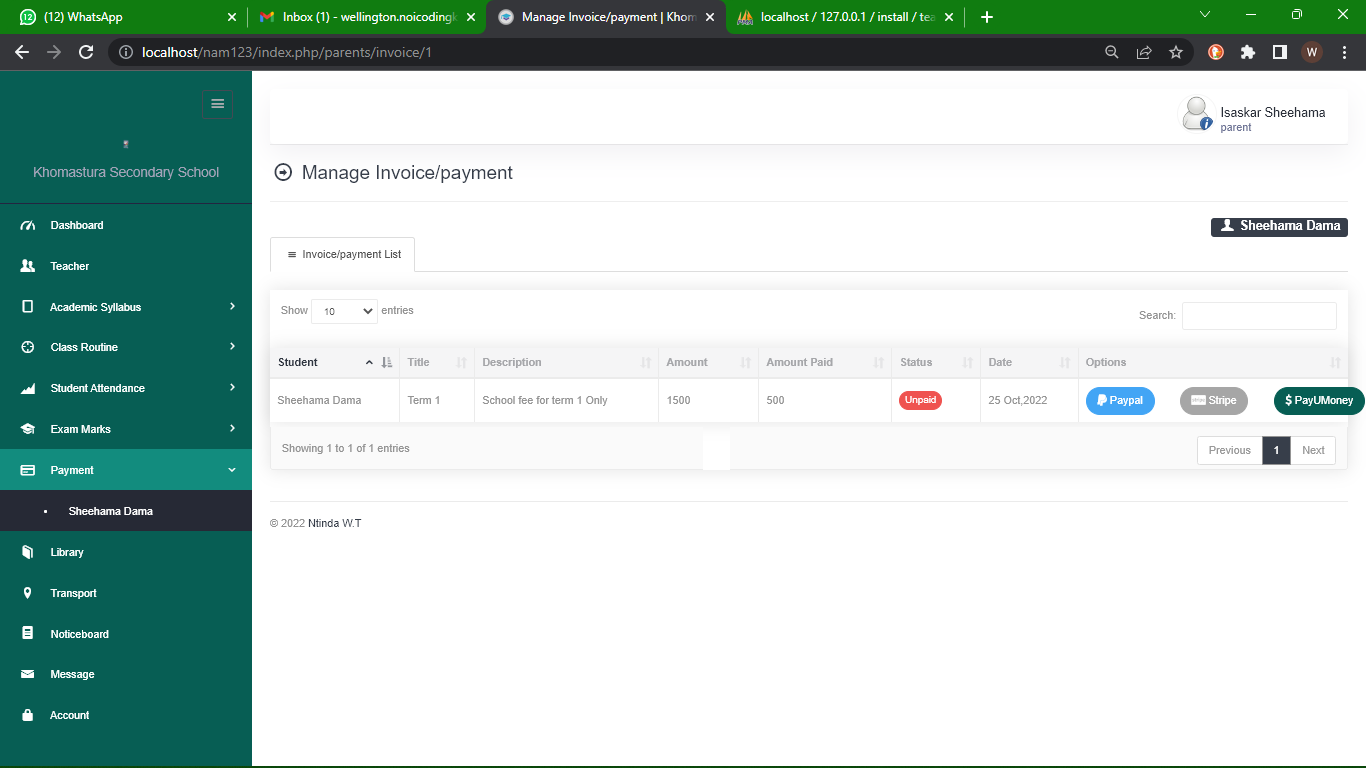
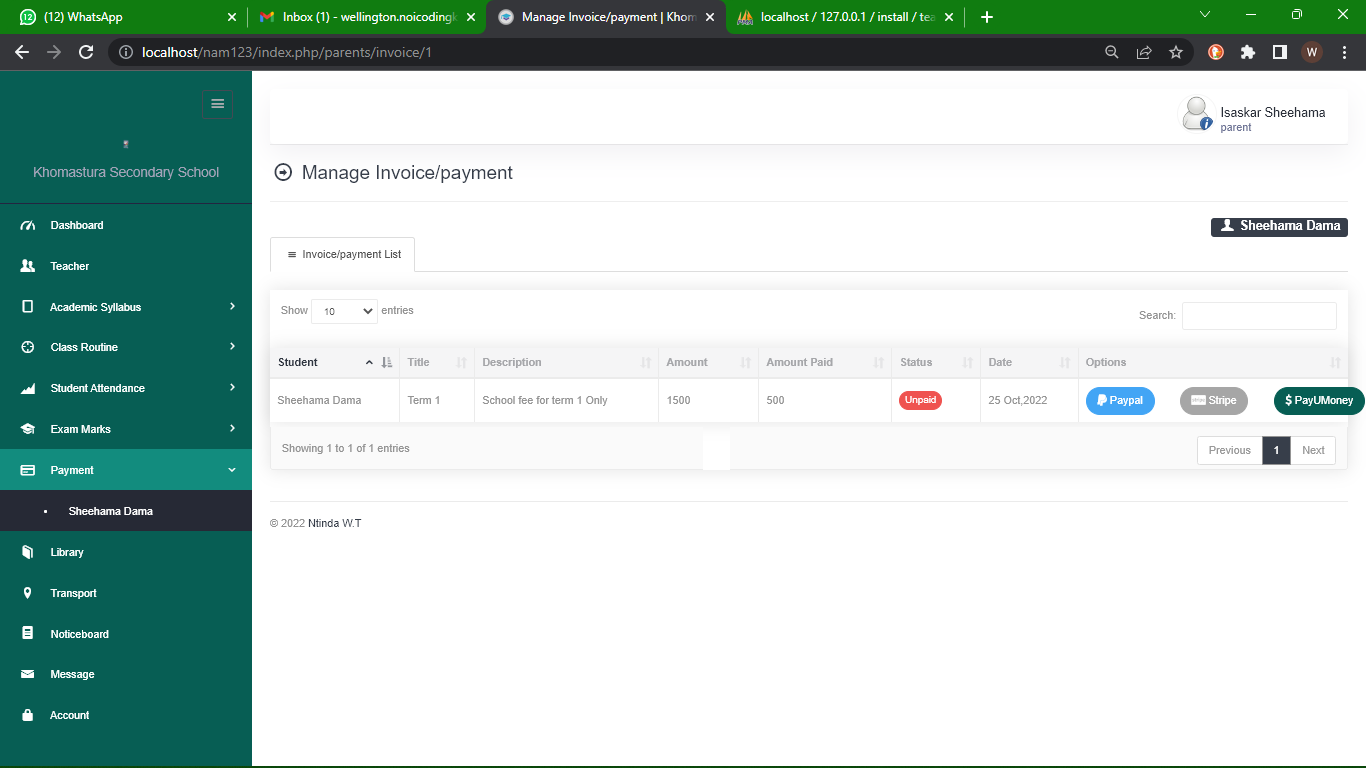


Figure 5. parents dashboard

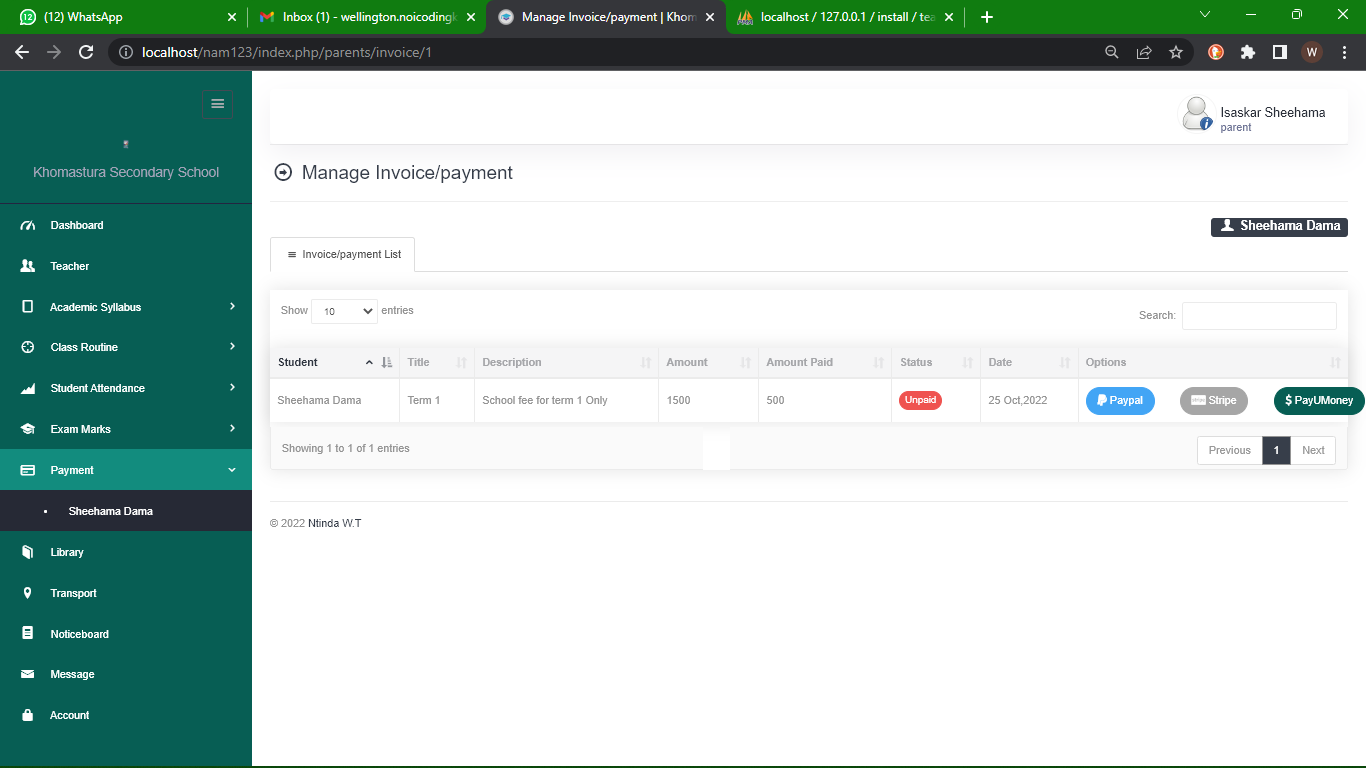


*Figure 5. parent making payements*

 A screenshot of a computer

Description automatically generated with medium confidence

*Figure 7. payment options*



*Figure.8 Payement successfully made*

## **Database Design**

### ER Diagram of database

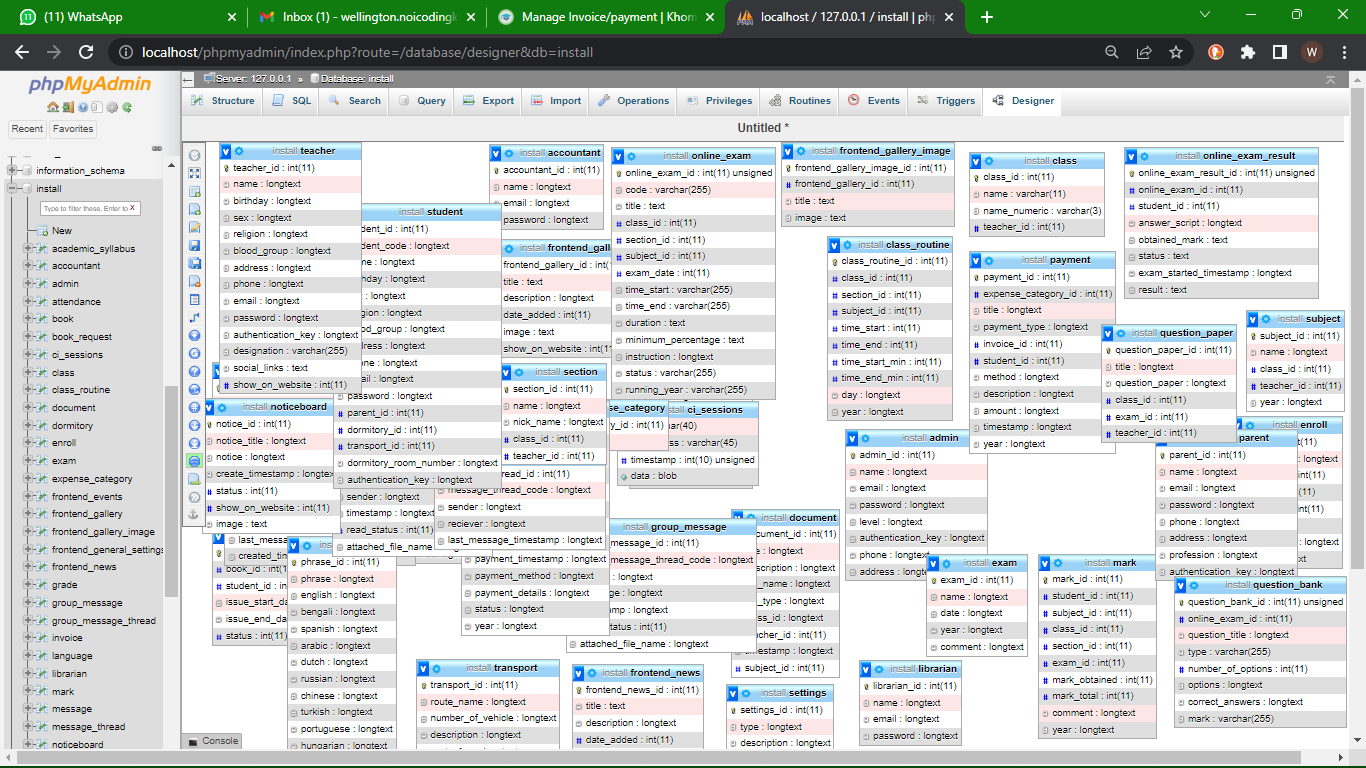


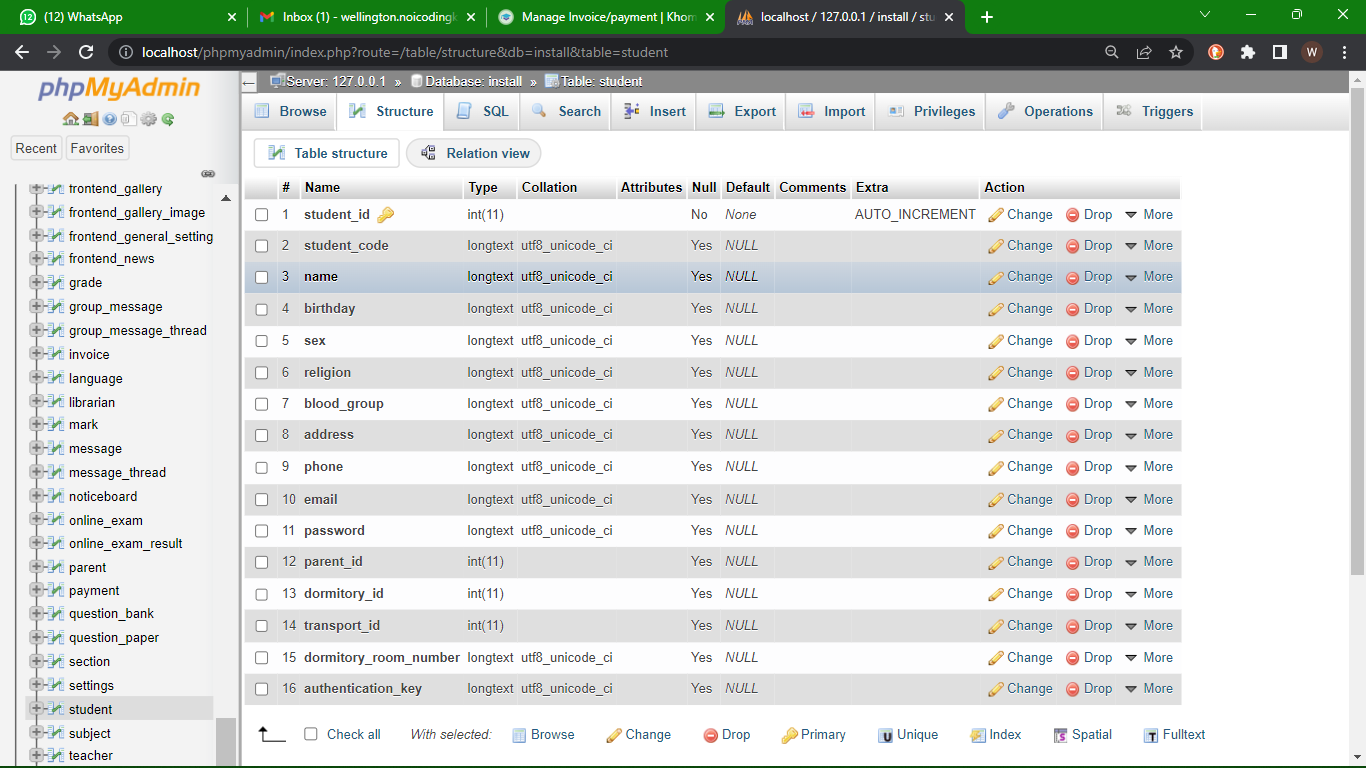
Figure 9. ER Diagram of database

### Table 1. Admin Table

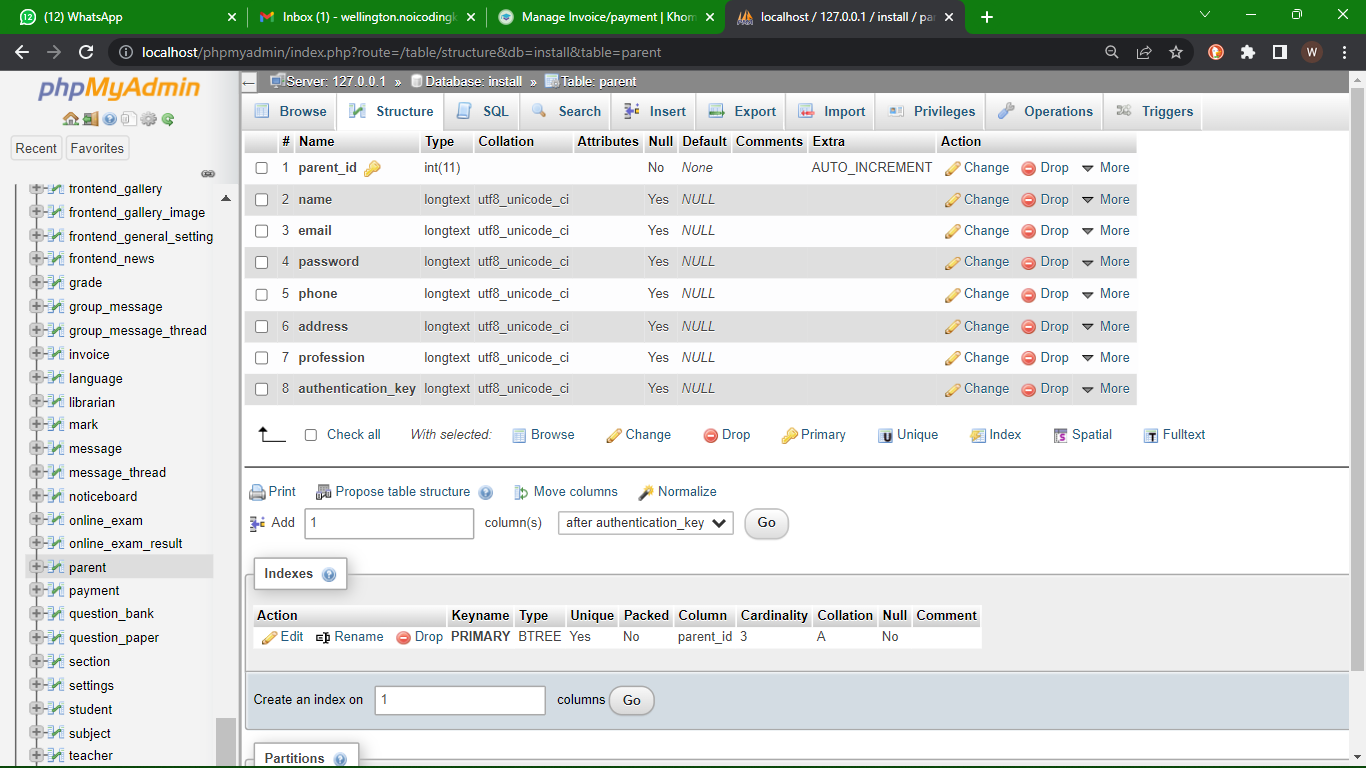
Graphical user interface, text, application

Description automatically generated

### Table 2. Student table



### Table 3. Parents table



### Table payements

A screenshot of a computer

Description automatically generated

## **Hardware and software Requirements**

### **Hardware requirements**

* Operating system: windows 10 pro
* Hard disk: 64 GB
* RAM: 4GB

### **Software requirements**

* HTML
* CSS, bootstrap, CodeIgniter
* JavaScript
* vs code editor
* PHP, Ajax, Js, jQuery
* MySQL database
* Xampp server

## CONCLUSION

This chapter the researcher looked at the suitable requirement gathering techniques to use for the system which are questionnaires and interviews, the software development method, which is prototyping, the different diagrams and user interfaces showing how the system will operate and the entities of the system, the systems architecture, hardware and software requirements of the system.

**CHAPTER FOUR****:**

**SYSTEM IMPLEMENTATION**



## **INTRODUCTION**

The implementation phase involves putting the project plan into action. In this chapter the developer provides codes that implement the main functions of the system and a few system algorithms.

## **ALGORITHM**

An algorithm is as process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer.

Admin

1. *Start*
2. *Admin login into system with valid credentials.*
3. *Admin information is stored in the database,*
4. *Display the admin interface*
5. *The admin manages user category (add, delete or update)*
6. *The admin manages the classes (add, delete or update)*
7. *The admin can see the number of users registered*
8. *Admin is able to manage attendance, exams, messages*
9. *Can make payments*
10. *Sign out.*

User (student)

1. *Start*
2. *user login into system with valid credentials.*
3. *User information is stored in the database,*
4. *Users go through the Dashboard*
5. *Do tasks e.g., take a test, exam, view notification,*
6. *Can view or add payements*
7. *Logout;*

User (Parent)

1. *Start*
2. *user login into system with valid credentials.*
3. *User information is stored in the database,*
4. *Users go through the Dashboard*
5. *Can view notification, payments history*
6. *Can add payments*
7. *Logout;*

## **CODES**

### **Login Page**

1. <?php
2. $system\_name  = $this->db->get\_where('settings', array('type' => 'system\_name'))->row()->description;
3. $header\_logo  = $this->frontend\_model->get\_frontend\_general\_settings('header\_logo');
4. ?>
5. <!DOCTYPE html>
6. <html lang="en">
7. <head>
8. <meta charset="utf-8">
9. <meta http-equiv="X-UA-Compatible" content="IE=edge">
10. <meta name="description" content="<?php echo get\_settings('system\_name'); ?>" />
11. <meta name="viewport" content="width=device-width, initial-scale=1, maximum-scale=1" />
12. <link name="favicon" type="image/x-icon" href="<?php echo base\_url();?>uploads/frontend/<?php echo $header\_logo;?>" rel="shortcut icon" />
13. <title>NOWDEMY login | <?php echo get\_settings('system\_name'); ?></title>
15. <!-- Favicon -->
16. <link rel="shortcut icon" href="<?php echo base\_url();?>uploads/frontend/<?php echo $header\_logo;?>" />
18. <!-- font -->
19. <link  rel="stylesheet" href="https://fonts.googleapis.com/css?family=Montserrat:300,300i,400,500,500i,600,700,800,900|Poppins:200,300,300i,400,400i,500,500i,600,600i,700,700i,800,800i,900">
21. <link rel="stylesheet" type="text/css" href="<?php echo base\_url('assets/login/plugins-css.css'); ?>" />
22. <link rel="stylesheet" type="text/css" href="<?php echo base\_url('assets/login/typography.css'); ?>" />
23. <link rel="stylesheet" type="text/css" href="<?php echo base\_url('assets/login/style.css'); ?>" />
24. <link rel="stylesheet" type="text/css" href="<?php echo base\_url('assets/login/responsive.css'); ?>" />
26. <link rel="stylesheet" href="<?php echo base\_url('assets/css/neon-forms.css'); ?>">
27. <link rel="stylesheet" href="<?php echo base\_url('assets/css/font-icons/entypo/css/entypo.css'); ?>">
28. <link rel="stylesheet" href="<?php echo base\_url('assets/css/custom.css'); ?>">
30. </head>
32. <body>
34. <div class="wrapper">
36. <section class="gray-bg height-100vh d-flex align-items-center page-section-ptb ">
37. <div class="container">
38. <div class="row no-gutters justify-content-center">
39. <div class="col-lg-4 col-md-6 login-fancy-bg bg-overlay-black-10" style="background: url(<?php echo base\_url();?>assets/login/login\_bg.jpg);">
40. <div class="login-fancy pos-r">
41. <div class="text-center">
42. <div style="padding: 55px;" class="d-none d-md-block"></div>
43. <img src="<?php echo base\_url().'uploads/logo.png'; ?>" height="25" />
44. <h2 class="text-white mb-20"></h2>
45. <h4 class="text-white mb-20"><?php echo get\_settings('system\_name');?></h4>
46. </div>
47. </div>
48. </div>
49. <div class="col-lg-4 col-md-6 white-bg">
50. <div class="login-fancy pb-40 clearfix" id = "login\_area">
51. <h3 class="mb-30"><?php echo get\_phrase('login'); ?></h3>
52. <form action="<?php echo site\_url('login/validate\_login'); ?>" method="post">
53. <div class="section-field mb-20">
54. <label class="mb-10" for="name"><?php echo get\_phrase('email'); ?>\* </label>
55. <input id="email" class="web form-control" type="email" placeholder="<?php echo get\_phrase('email'); ?>" name="email" required>
56. </div>
57. <div class="section-field mb-20">
58. <label class="mb-10" for="Password"><?php echo get\_phrase('password'); ?>\* </label>
59. <input id="Password" class="Password form-control" type="password" placeholder="<?php echo get\_phrase('password'); ?>" name="password" required>
60. </div>
61. <button type="submit" class="btn btn-primary"><?php echo get\_phrase('login'); ?></button>
62. </form>
64. <div class="section-field">
65. <div class="remember-checkbox mb-30">
66. <a href="#" class="float-right" id = "forgot\_password\_button" onclick="toggleView(this)"><?php echo get\_phrase('forgot\_password'); ?>?</a>

69. </div>
70. </div>
71. </div>
73. <div class="login-fancy pb-40 clearfix" id = "forgot\_password\_area" style="display: none;">
74. <h3 class="mb-30"><?php echo get\_phrase('forgot\_password'); ?></h3>
75. <form class="" action="<?php echo site\_url('login/reset\_password');?>" method="post">
76. <div class="section-field mb-20">
77. <label class="mb-10" for="name"><?php echo get\_phrase('email'); ?>\* </label>
78. <input id="forgot\_password\_email" class="web form-control" type="email" placeholder="<?php echo get\_phrase('email'); ?>" name="email" required>
79. </div>
80. <button type="submit" class="btn btn-primary"><?php echo get\_phrase('send\_mail'); ?></button>
81. </form>
83. <div class="section-field">
84. <div class="remember-checkbox mb-30">
85. <a href="#" class="float-right" id = "login\_button" onclick="toggleView(this)"><?php echo get\_phrase('back\_to\_login'); ?>?</a>
86. </div>
87. </div>
88. </div>
89. </div>
90. </div>
91. </div>
92. </section>
93. </div>
95. <!-- jquery -->
96. <script src="<?php echo base\_url('assets/login/jquery-3.3.1.min.js'); ?>"></script>
97. <script src="<?php echo base\_url('assets/js/toastr.js'); ?>"></script>
98. <script type="text/javascript">
99. function toggleView(elem) {
100. if (elem.id === 'forgot\_password\_button') {
101. $('#login\_area').hide();
102. $('#forgot\_password\_area').show();
103. }else if (elem.id === 'login\_button') {
104. $('#login\_area').show();
105. $('#forgot\_password\_area').hide();
106. }
107. }
108. </script>
109. <!-- SHOW TOASTR NOTIFIVATION -->
110. <?php if ($this->session->flashdata('flash\_message') != ""):?>
112. <script type="text/javascript">
113. toastr.success('<?php echo $this->session->flashdata("flash\_message");?>');
114. </script>
116. <?php endif;?>
118. <?php if ($this->session->flashdata('error\_message') != ""):?>
120. <script type="text/javascript">
121. toastr.error('<?php echo $this->session->flashdata("error\_message");?>');
122. </script>
124. <?php endif;?>
125. </body>
126. </html>

up codes continuing…

#### Forgot password

Codes for when forget password

1. <!doctype html>
2. <?php
3. //$system\_title = $this->db->get\_where('settings', array('type' => 'system\_title'))->row()->description;
4. $system\_name  = $this->db->get\_where('settings', array('type' => 'system\_name'))->row()->description;
5. ?>
7. <html class="no-js" lang="">
8. <head>
9. <meta charset="utf-8">
10. <meta http-equiv="x-ua-compatible" content="ie=edge">
11. <title>
12. <?php echo get\_phrase('forgot\_password'); ?> | <?php echo $system\_name; ?>
13. </title>
14. <meta name="description" content="">
15. <meta name="viewport" content="width=device-width, initial-scale=1">
17. <link rel="shortcut icon" href="<?php echo base\_url('assets/login\_page/img/favicon.png');?>">
18. <link rel="stylesheet" href="<?php echo base\_url('assets/css/bootstrap.css');?>">
19. <link rel="stylesheet" href="<?php echo base\_url('assets/login\_page/css/font-awesome.min.css');?>">
20. <link rel="stylesheet" href="<?php echo base\_url('assets/login\_page/css/normalize.css');?>">
21. <link rel="stylesheet" href="<?php echo base\_url('assets/login\_page/css/main.css');?>">
22. <link rel="stylesheet" href="<?php echo base\_url('assets/login\_page/css/style.css');?>">
23. <script src="<?php echo base\_url('assets/login\_page/js/vendor/modernizr-2.8.3.min.js');?>"></script>
24. <link href="https://fonts.googleapis.com/css?family=Quicksand:300,400,500,700" rel="stylesheet">
26. </head>
27. <body>
28. <div class="main-content-wrapper">
29. <div class="login-area">
30. <div class="login-header">
31. <a href="<?php echo site\_url('login');?>" class="logo">
32. <img src="<?php echo base\_url('assets/login\_page/img/logo.png');?>" height="60" alt="">
33. </a>
34. <h2 class="title"><?php echo $system\_name; ?></h2>
35. </div>
36. <div class="login-content">
37. <form method="post" role="form" id="form\_login"
38. action="<?php echo site\_url('login/reset\_password');?>">
39. <div class="form-group">
40. <input type="text" class="input-field" name="email" placeholder="<?php echo get\_phrase('email').'/'.get\_phrase('username');?>"
41. required autocomplete="off">
42. </div>
43. <button type="submit" class="btn btn-primary"><?php echo get\_phrase('reset\_password'); ?><i class="fa fa-unlock"></i></button>
44. </form>
46. <div class="login-bottom-links">
47. <a href="<?php echo site\_url('login');?>" class="link"><i class="fa fa-lock"></i><?php echo get\_phrase('return\_to\_login\_page'); ?></a>
48. </div>
49. </div>
50. </div>
51. <div class="image-area forgot-pass"></div>
52. </div>
54. <script src="<?php echo base\_url('assets/login\_page/js/vendor/jquery-1.12.0.min.js');?>"></script>
55. <script src="<?php echo base\_url('assets/js/bootstrap-notify.js');?>"></script>
57. <?php if ($this->session->flashdata('reset\_error') != '') { ?>
58. <script type="text/javascript">
59. $.notify({
60. // options
61. title: '<strong><?php echo get\_phrase('error');?>!!</strong>',
62. message: '<?php echo $this->session->flashdata('reset\_error');?>'
63. },{
64. // settings
65. type: 'danger'
66. });
67. </script>
68. <?php } ?>
69. <?php if ($this->session->flashdata('reset\_success') != '') { ?>
70. <script type="text/javascript">
71. $.notify({
72. // options
73. title: '<strong><?php echo get\_phrase('success');?>!!</strong>',
74. message: '<?php echo $this->session->flashdata('reset\_success');?>'
75. },{
76. // settings
77. type: 'success'
78. });
79. </script>
80. <?php } ?>
82. </body>
83. </html

Figure16: forgot password

Figure17: Admin Page

1. <a href="javascript:;" onclick="showAjaxModal('<?php echo site\_url('modal/popup/modal\_admin\_add');?>');"
2. class="btn btn-primary pull-right">
3. <i class="entypo-plus-circled"></i>
4. <?php echo get\_phrase('add\_new\_admin');?>
5. </a>
6. <br><br>
8. <table class="table table-bordered datatable" id="table\_export">
9. <thead>
10. <tr>
11. <th style="width: 60px;">#</th>
12. <th><div><?php echo get\_phrase('name');?></div></th>
13. <th><div><?php echo get\_phrase('email').'/'.get\_phrase('username');?></div></th>
14. <th><div><?php echo get\_phrase('phone');?></div></th>
15. <th><div><?php echo get\_phrase('address');?></div></th>
16. <th><div><?php echo get\_phrase('options');?></div></th>
17. </tr>
18. </thead>
19. <tbody>
20. <?php
21. $count = 1;
22. $admins   =   $this->db->get('admin')->result\_array();
23. foreach($admins as $row): ?>
24. <tr>
25. <td><?php echo $count++;?></td>
26. <td><?php echo $row['name'];?></td>
27. <td><?php echo $row['email'];?></td>
28. <td><?php echo $row['phone'];?></td>
29. <td><?php echo $row['address'];?></td>
30. <td>
32. <div class="btn-group">
33. <button type="button" class="btn btn-default btn-sm dropdown-toggle" data-toggle="dropdown">
34. Action <span class="caret"></span>
35. </button>
36. <ul class="dropdown-menu dropdown-default pull-right" role="menu">
38. <li>
39. <a href="#" onclick="showAjaxModal('<?php echo site\_url('modal/popup/modal\_admin\_edit/' . $row['admin\_id']);?>')">
40. <i class="entypo-pencil"></i>
41. <?php echo get\_phrase('edit');?>
42. </a>
43. </li>
44. <li class="divider"></li>
46. <li>
47. <a href="#" onclick="confirm\_modal('<?php echo site\_url('admin/admin/delete/' . $row['admin\_id']);?>');">
48. <i class="entypo-trash"></i>
49. <?php echo get\_phrase('delete');?>
50. </a>
51. </li>
52. </ul>
53. </div>
55. </td>
56. </tr>
57. <?php endforeach;?>
58. </tbody>
59. </table>
61. <!-----  DATA TABLE EXPORT CONFIGURATIONS ---->
62. <script type="text/javascript">
64. jQuery(document).ready(function($)
65. {
66. $('#table\_export').dataTable();
67. });
69. </script>

Figure16: Admin page

## **Conclusion**

In this chapter developer has provided the most important codes that are functional and contributed more functions to the system. Among these where the codes for the login forms and logout pages for both admin and users among others.

# **CHAPTER FIVE**

# **SYSTEM TESTIMG**



**INTRODUCTION**

This chapter demonstrates test results of the system testing. Various test carried out include the sign test, log in test, adding students, editing, deleting.

**5.1 UNIT TESTING**

A type of testing in which individual units or functions of [software testing](https://www.xenonstack.com/blog/software-testing-best-practices). Its primary purpose is to test each unit or function. A unit is the smallest testable part of an application. It mainly has one or a few inputs and produces a single output.

### **Login Test**

Graphical user interface, application

Description automatically generated

Figure login test

Calendar

Description automatically generated

Figure admin dash

**Admin login success**

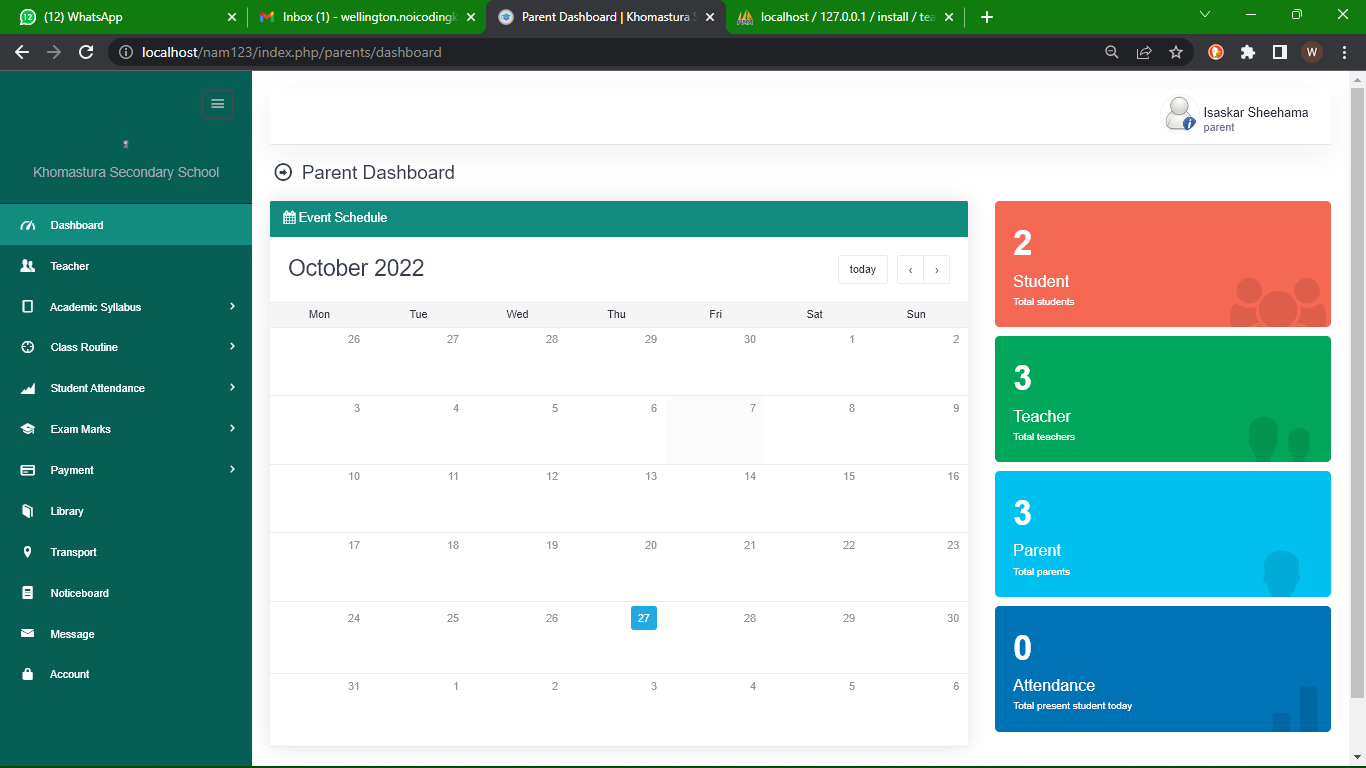


Figure parent dash board success

**Parent signin success**

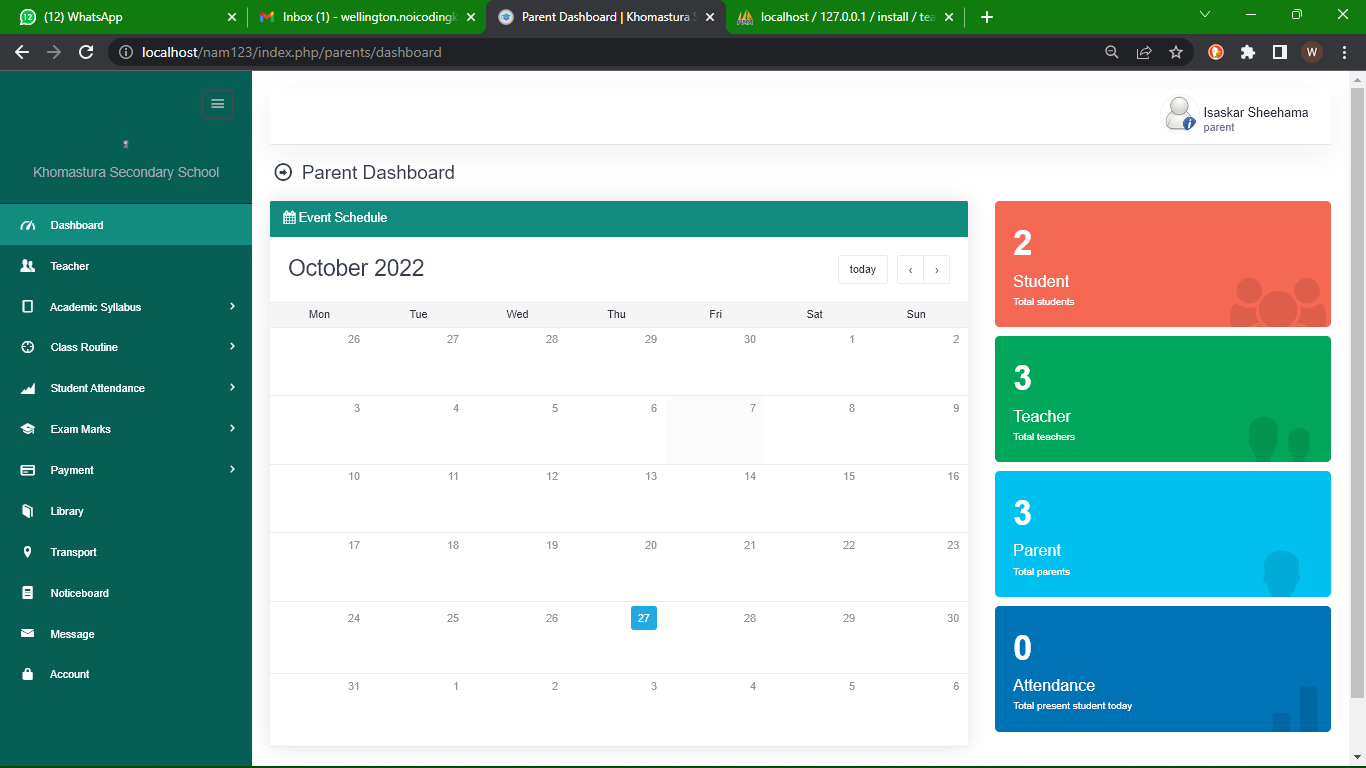


Figure parrent signin success

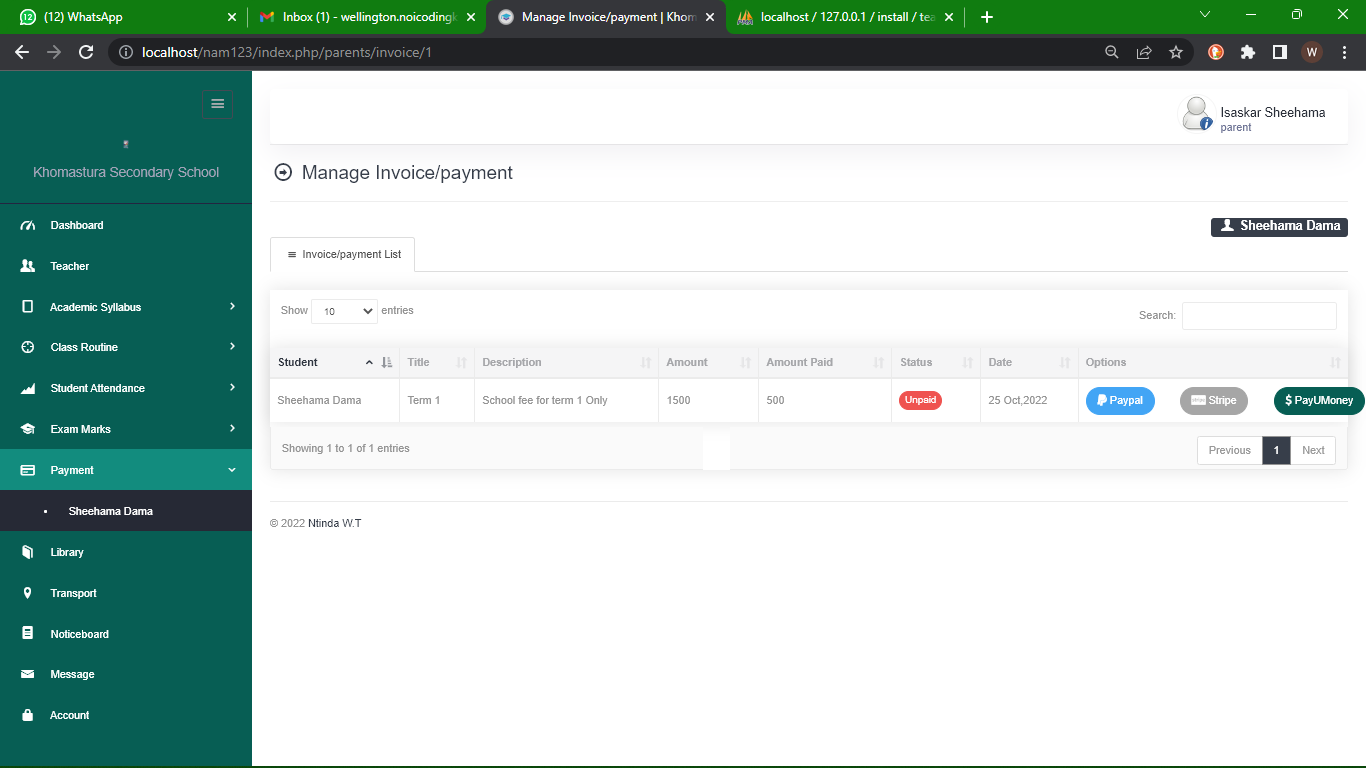


Figure parents payments test

A screenshot of a computer

Description automatically generated with medium confidence

Figure payement options

# **CHAPTER SIX:**

# **CONCLUSION AND FUTURE WORK**

## **6.1 CONCLUSION**

Although any changes that may come can be looked at, the school management system is a complete system it provides what was asked of it to do. The students can be able to register for classes, view the results and scroll to see what is available in the notification. The admin is able to add, edit and delete from db.

## **6.2 FUTURE WORK**

As mentioned, room for improvement is always open for the future the developer would like to improve the performance of the system. The developer would like to transform the developed system into a bigger system in future.

# Reference

Ellucian (2021). Banner Reviews. In Higher Education Student Information Systems. Retrieved from:

https://www.gartner.com/reviews/market/higher-education-studentinformationsystems/vendor/ellucian/product/banner/reviews

Kadhilongo, E. (2018, 09 04). The cause of school drop outs. (F. Johannes, Interviewer)

Kleopas, S. (2021, 03 20). How about the accademic perfomance on past two years. (E. Haifete, Interviewer)

Maria, N. (2021, 03 29). How about the school enloring system. (W. Ntinda, Interviewer)

Monique, H. (2021, 04 14). How stressful or stressfree is the manual registration process. (N. W.T, Interviewer)

Ngesheya, S. (2021, 04 19). How many learners can a school register per year. (N. Naeman, Interviewer)

Nghishitelwa, B. (2019, 12 26). What causes the movement of learners between schools. (N. D. Cleo L, Interviewer)

Shikushu, N. (2021, 04 20). Why learners keep traveling long distance seeking for schools inkhomas. (N. Wellington, Interviewer)

system, E. m. (2018, june). *Education statistics in Namibia.* Retrieved from EMIS Education statistics :

http://www.moe.gov.na/files/downloads/d15\_EMIS%20Education%20Statistics%202017%20%20web%20quality.pdf

**7. APPENDICES**

**7.1 Appendix A- User manual**

The student café online ordering system include the user page and admin page. Both pages have different task which can be performed there.

**Admin manual**

1. *Start*
2. *Admin login into system with valid credentials.*
3. *Admin information is stored in the database,*
4. *Display the admin interface*
5. *The admin manage user category (add, delete or update)*
6. *The admin manage the classes(add, delete or update)*
7. *The admin can see the number of users registered*
8. *Admin is able to manage attendance*
9. *Can make payement*
10. *Sign out;*

User (student)

1. *Start*
2. *user login into system with valid credentials.*
3. *User information is stored in the database,*
4. *User go through the Dashboard*
5. *Logout;*

User (Parent)

1. *Start*
2. *user login into system with valid credentials.*
3. *User information is stored in the database,*
4. *Users go through the Dashboard*
5. *Can view notification, payments history*
6. *Can add payments*
7. *Logout;*

**7.2 Appendix B- screenshots found in chapter**

**7.3 Appendix C**

Appendix is found on the research proposal, and the document of research proposal is attached to this document