



University
of Dundee

Online Result Computation System

Author: Ibrahim Mohammed

Supervisor: Dr Keith Edwards

MSc Computing
School of Science and Engineering
University of Dundee

September, 2019



Computing
at the University of Dundee

EXECUTIVE SUMMARY

This project involves the design and development of a web-based result computation system for Niger State Polytechnic, Zungeru (System that allows 5 user access roles to replace the existing manual system). The primary aim of this project is to design and develop such system which functions to successfully input student records, courses records and exam scores by importing the data through an excel file in .cvs format, then generate a result in broadsheets including graphical analysis.

Initially, the system is designed to have three user roles (Coordinator, Students, and Teaching staff). It was decided to add two more roles (System Manager, and Exams & records), taking off some duties off the coordinator thereby reducing the duties of coordinators. Sketches were created through a drafted plan and discussion were held regarding the proposed function of the overall system and its features

Following the development of these designs, some features were created to get a realistic representation of the concept and the design layout. Early prototype of these features includes low technology implementation using materials like pen and paper to sketch the look and feel of the proposed system and high technology implementation using a prototyping software called indigo studio to develop a functional prototype of the proposed system, interaction, and dynamic features like navigational links and buttons were also used.

After evaluating the prototype, some ideas were introduced to add and modify some features in the system to improve its efficiency and overall functionality. Some of the ideas include creating a backup, adding visualisation to the analysis, using a light and less colourful interfaces and background colours.

In developing the final system, agile methodology was used, every piece of code was tested using Test-Driven Development (TDD) approach. Decision were made to improve the reliability, effectiveness and security. In the front end; user logs, user navigation menus control and generating blank .csv file format for importing records were added. In the back end the structure of the database was modified by adding more tables. More decisions were made, which include the use of persona to create better user experience, and the use of “use cases and misuse cases” to detect and mitigate possible threat or attack on the system. Similarly, to build the final system programming language and database connection language (PHP, CSS, JavaScript, and MYSQL) were used. The final testing was done with 5 participants working with system to perform all users’ tasks, responds to questions and make suggestion about the overall functionality of the system based on their experience.

The system was finally tested for accessibility and ease of use, using some online tools (AChecker, Survey Monkey and Nasa-TLX). These tests conducted on the final system yield a consistent result in terms of functionality. The result of the tests showed that it was not frustrating and less demanding in both mental and physical efforts required to perform the tasks of computing students result. In less than 5minutes they were able to import scores and generate results including the analysis.

As such, the performance of the system during formal assessment is expected to be successful, and of a similar time, providing no frustration throughout the use of the system.

DECLARATION

I declare that the special study described in this dissertation has been carried out and the dissertation composed by me, and that the dissertation has not been accepted in fulfilment of the requirements of any other degree or professional qualification.

Signed:

Ibrahim Mohammed

CERTIFICATE

I certify that (your full name) has satisfied the conditions of the Ordinance and Regulations and is qualified to submit this dissertation in application for the degree of Master of Science.

Signed:

Dr Keith Edwards

ACKNOWLEDGEMENTS

Praises be to Almighty Allah, for His endless blessings throughout my stay in Dundee and successfully completing my study.

I would like to express my deep and sincere gratitude to my Supervisor Dr. Keith Edwards for the continuous support throughout my course work and project; for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in times of development, research and writing of this project report. His dynamism, vision, sincerity and motivation have deeply inspired me. I could not have imagined having a better Supervisor and mentor for my project. I say gratitude for his friendship, empathy, and great sense of humor.

My sincere thanks also goes to Dr. Craig, and Dr. Sasa, for their professional and helpful support, sparing their precious time to attend to me in my times of need. Their professional support helped to correctly fix the use and misuse case diagram and usage of the diagram to figure out the security challenges in my work.

I am extremely grateful to my parents for their love, prayers, caring and sacrifices they may have made preparing me for my future. I am very much thankful to my wife for her love, understanding, prayers and continuing support throughout the time of this study. Also I express my thanks to my sisters, brothers, sister in law and brother in laws for their support and valuable prayers. My Special thanks goes to my friends; especially Mohammed Nura Adamu for the keen interest shown to complete this thesis successfully.

TABLE OF CONTENTS	
TITLE PAGE	i
EXECUTIVE SUMMARY	ii
DECLARATION	iii
CERTIFICATE	iv
ACKNOWLEDGEMENTS	v
TABLE OF CONTENTS	vi
LIST OF TABLES	ix
LIST OF FIGURES	x
CHAPTER ONE	1
1.0 Introduction	1
1.1 Objectives of the study	2
1.2 Organisation of the Report	2
CHAPTER TWO	3
2.0 Background	3
2.1 Review of relevant and related literatures	3
CHAPTER THREE	5
3.0 Requirement Specification	5
3.1 Problem Specification	5
3.2. Gathering Requirements	5
3.2.1 Online Review	7
3.2.2 Prototyping	7
3.2.3 Evaluation	7
3.2.4 Implementation	8
3.3 System Requirement Analysis and Specification	8
3.3.1 Departmental Registration:	8
3.3.2 Problematic student	9
3.3.3 System Requirements Specification	9
3.3.4 Functional Requirements of Registration	9
3.3.5 Major Requirement	10
3.3.6 Treatment of Special Students' Cases	10
3.4 Project plan	11
CHAPTER FOUR	12
4.0 Design methods	12
4.1 Personas	12
4.2 Use and misuse case diagram	12

4.3 Prototyping.....	13
4.3.1 Low Fidelity – Sketches	13
4.3.2 High Fidelity – Indigo studio.....	14
4.4 Evaluation of High-Fidelity Prototype.....	15
4.4.1 Participants	15
4.4.2 Analysis and result of evaluation	16
4.5 Database design	16
4.5.1 Requirement analysis.....	16
4.5.2 Conceptual Modelling.....	16
4.5.3 Logical modelling	17
4.5.4 Physical modelling.....	17
4.6 Design process	18
4.7 Technologies Used.....	19
4.7.1 Localhost Wamp Server	19
4.7.2 Server-side scripting language	19
4.7.3 Back end and front end.....	20
4.7.4 Visualisation / Charts	21
4.7.5 GitHub	21
4.7.6 Visual paradigm.....	22
4.7.7 Accessibility Checker (AChecker)	22
4.8 Design Decisions	22
CHAPTER FIVE	24
5.0 Implementation and testing.....	24
5.1 Software development	24
5.2 Security	26
5.2.1 Cross-Site Request Forgery (CSRF).....	26
5.2.2 Password Hashing	27
5.2.3 SQL Injections.....	27
5.2.4 Cross Site Scripting (XSS).....	28
5.2.4 Other Security Challenges.....	28
5.3 Testing.....	28
5.3.1 Agile Testing (TDD).....	29
4.3.2 Web Content Accessibility	29
5.4 Final product	31
5.4.1 Evaluation of final product.....	31
5.4.2 Result of Evaluation	31

5.4.3 Tasks worked on	32
5.4.4 NASA-TLX work load results (Average Scores)	32
5.5 Functionality and interfaces of the completed system	33
5.5.1 Login	33
5.5.2 User Dashboard	33
5.5.3 Users Navigation Menu.....	34
5.5.4 Profile Dropdown Menu	34
5.5.3 Settings.....	35
5.5.6 Teacher / Lecture Role	41
5.5.7 Exams and Record Role.....	41
5.5.8 Student Role.....	41
5.6 Appraisal	42
5.6.1 Implementation decision	42
5.6.2 Wider Context	42
5.6.3 Self-Appraisal	43
5.6.4 Evaluation	43
CHAPTER SIX	44
6.0 Summary, Conclusions and Recommendation	44
6.1 Summary	44
6.2 Conclusions.....	44
6.3 Recommendations for further work	44
REFERENCES	45
APPENDICES	47

LIST OF TABLES

Table 3. 1: Grading System for Polytechnics in Nigeria	6
Table 3. 2: Cumulative Grade Point Average	6
Table 3. 3: Project plan	11
Table 3. 4: Test summary Report.....	29

LIST OF FIGURES

Figure 4. 1: Use/Misuse Case diagram.....	13
Figure 4. 2: login page.....	14
Figure 4. 3: Home page with accessible menus.....	14
Figure 4. 4: student individual result, analysis page.....	14
Figure 4. 5: Indigo Studio Software Interface.....	15
Figure 4. 6: The Prototype of the Login Page.....	15
Figure 4. 7: Wamp Server Icon in Green.....	17
Figure 4. 8: PhpMyadmin URL.....	18
Figure 4. 9: Database Menu.....	18
Figure 4. 10: Database Name Field.....	18
Figure 4. 11: Wamp Server Index page.....	19
Figure 4. 12: Bootstrap Admin Dashboard Templates.....	20
Figure 5. 1: WAMP Menu.....	24
Figure 5. 2: GitHub Repository.....	25
Figure 5. 3: Login Page Code.....	25
Figure 5. 4: Confirm_page.php.....	26
Figure 5. 5: Save_Comfirm.php.....	26
Figure 5. 6: Bcrypt Password Hashing Function.....	27
Figure 5. 7: Bcrypt Hashing Verification Function.....	27
Figure 5. 8: Strip Unwanted Characters from User Input.....	28
Figure 5. 9: Validating Untrusted HTML Input.....	28
Figure 5. 10: Setting Default Language to English.....	30
Figure 5. 11: Nasa-TLX workload chart.....	32
Figure 5. 12: User Login Page.....	33
Figure 5. 13: System Manager Menu.....	34
Figure 5. 14: Coordinator Menu.....	34
Figure 5. 15: Student Menu.....	34
Figure 5. 16: Teacher Menu.....	34
Figure 5. 17: Exam & Records Menu.....	34
Figure 5. 18: Other Users Profile Dropdown Menu.....	34
Figure 5. 19: Users (System Manager) Profile Dropdown Menu.....	35
Figure 5. 20: Change Password Form.....	35
Figure 5. 21: System Manager Dashboard.....	35
Figure 5. 22: Coordinator Dashboard.....	36
Figure 5. 23: Student dashboard.....	36
Figure 5. 24: Teacher/Lecturer dashboard.....	37
Figure 5. 25: View Board Result Form.....	37
Figure 5. 26: Broadsheet Result.....	37
Figure 5. 27: Manage Score Menu and Submenu.....	38
Figure 5. 28: Exam Score Sheet Form.....	38
Figure 5. 29: Import Score Form.....	39
Figure 5. 30: File Import Form.....	39
Figure 5. 31: List of Students.....	39
Figure 5. 32: Edit Student Score Form.....	40
Figure 5. 33: Single Course Analysis Form.....	40

Figure 5. 34: Single Course Visual Analysis	40
Figure 5. 35: All courses Visual Result Analysis.....	41
Figure 5. 36: Semester Result Menu and Submenus.	41
Figure 5. 37: Student Semester Result.....	42

CHAPTER ONE

1.0 Introduction

Niger State Polytechnic, Zungeru is a tertiary institution located at Zungeru, Wushishi Local Government Area of Niger State, Nigeria in West Africa. The institution has 2 colleges; College of Science and Technology (CST) and College of Administrative and Business Studies (CABS). College of Science and Technology (CST) comprise of School of Environmental Studies (SES), School of Engineering Technology (SET), and School of Natural and Applied Sciences (SNAS) whereas College of Administrative and Business Studies (CABS) comprise of School of Administrative Studies (SAS), School of General Studies (SGS), and School of Business Studies (SBS). Each school has various departments and about 35 programmes comprising of Higher National Diploma (HND) and National Diploma (ND) from 17 departments in total. All programmes last for two years with each year known as academic session, in a session there are two semesters, totalled four semesters for each programme. The polytechnic is led by the Rector, Directors oversees the colleges, and the Departments head by Head of Departments.

There are about 770 teaching staff and 1300 non-teaching staff in the polytechnic, including principal officers, cleaners and security personnel. The average total number of students in the polytechnic from the year 2015-2019 is 2250. In each department there are average of about 100 students, and average of 60 students in each programme. (NSP, 2018).

In total the polytechnic has an average of 210 courses including general courses such as: Use of English, and Entrepreneurship Development. Each course has course Code, Course Title and Course Unit offered in different semesters. The Course unit is the number of credit hours allocated to a course in each programme which vary from one course to another, because of the differences that exist between the courses in terms of scope and need. Hence some courses are allocated greater course unit than others.

Academic Planning Unit (APU) manage student admission process, courses offered in the polytechnic, create time table for lectures and examination, including resumption and vacation dates. Result computation starts from the departmental level, the department collects the approved list of students and courses from the APU, all exam scores are submitted to the departmental coordinators by various course lecturers, then the scores are computed against each student and the corresponding courses they offered.

After computation, the result is printed on large paper format, usually in A3 (29.7cm by 42cm) sheet, with columns running down the whole sheet known as broad sheet result submitted to APU, to be presented in the academic board, the broad sheet comprises of analysis showing all courses offered, course code, unit, grades, total, mean, standard deviation and percentage pass, it also include the result summary, showing total number of students in class, number of students that passed, number of students who failed and will have to resit, number of students absent with Excuse, and number of students advised to withdraw.

Exams and record unit of the polytechnic keeps student's examination records and results, they are responsible for ensuring that the records of result presented to the academic board is collected and kept after the board decisions and amendment from the departments for records purposes.

The goal of the study is to develop a web-based system as an automation to the existing manual process of computing students' examination scores in Niger State Polytechnic, as the existing system is prone to errors and waste of time and resources like paper.

In view of the above, the new (automated) system will comprise of five user access roles as:

- System manager: This user controls all user accounts, manages personal records of students and courses (add, update and view)
- Department Coordinator: This user manages student scores (import/input/add student exam scores, consider/edit/update) score and allocate courses to a staff in a specific department
- Student: This user will only have access to view their individual results, all or on semester basis.
- Teaching staff: this user can import, and view student exam scores only of courses allocated to them by the departmental coordinator.
- Exams and Records: This user can only view student personal information, courses, and broad sheet result of all departments

All the users listed above will have a different view when given access to the system through a login page process.

1.1 Objectives of the study

- Gather information about user needs of a Result Computation System
- Design a wireframe / prototype
- Develop and design a web application in PHP, using MySQL, HTML, CSS and JavaScript, with five (5) user role each with different user view as described above.
- To keep track of up-to-date records of all the students in the polytechnic
- Support registration, data upload, queries to the system, generate broad sheet report and visual and graphical result analysis for academic board decision-making with user-friendly interfaces for easy interaction
- Test the application for any dysfunctionality.

1.2 Organisation of the Report

This report starts by introducing the online result computation system in chapter 1, chapter 2 covers a brief review of relevant literature to establish the context of the project. Chapter 3 gives a specification of the problem and an explanation of how the developer arrived at this specification. An initial work schedule including an overall project plan with time-scales, deliverables and resources. Chapter 4 presents the design, which includes the design method, design process & outcome as well as describing the design decisions. Chapter 5 explain further how the completed system is implemented, tested, finally evaluated and describes the functionality and interfaces which comprise the description of production, testing, debugging and demonstrating that the specification has been satisfied. Chapter 6 comprises of summary, conclusion and recommendations.

CHAPTER TWO

2.0 Background

Nigerian tertiary institutions are following the trend in Information and Communication Technologies in transforming their record keeping and information management operations. Result Management System (RMS) is currently an information system that readily attracts the attention of many tertiary institutions in Nigeria.

The review of relevant and related web applications (result computation system) is an important process to achieve a successful design and implementation of a more comprehensive result computation system.

Based on these reviews, it was obvious that majority of the systems do not have the combination of student profile, graphical and visual result analysis, and broad sheet generation. However, some of the system have the combination of two out of these functionalities. Many, though, only generate broad sheets, which is one of the major components of the system required by tertiary institutions. Moreover, only a few of them allow importation of scores from file e.g. excel (.csv) format. Finally, only few has more than two user access roles.

2.1 Review of relevant and related literatures

Creating and organizing information in a useful manner is known as data processing. (Udeze, 2017) Explain how the manual method of computing students' result produce lots of errors in most tertiary institutions in Nigeria, therefore, the use of computer systems to process information is imperative and desirable as it will enable computation of students' progress, better access to students' records (personal, and courses), updating student records, keeping track of passed and failed courses (performance). The traditional method suffers lots of setbacks; it takes a lot of time and is prone to errors, leading to late publishing of examination result, sometimes with inaccurate grades and GPAs generated from scores computed, these errors could lead to wrong decision in awarding students.

(Akpasam, 2017) says that most of the students' RMS are traditional and standalone with few or basic elements of a standard Result Management System (RMS). Most of the students RMS are not web based and others do not integrate well with other information system in their institutions. Therefore, the support for RMS with an automated platform to managing result for all categories of students in a friendly and interactive manner is required.

Ashwin et al (2017) describe student result analysis as an online application used to analyse, store and keep track of student data and compute the mark analysis process in an educational institution, it enables the view of individual student's result, automated computation of results from various departments. The use of computation systems that have capabilities beyond result computation is not so much in use. Tertiary institutions today compute student result analysis manually which consume lots of time and effort. System that does analysis of student result is required as it will aid decision-making in tertiary institutions.

Obasa, et al (2011) observed that several problems associated with student academic record management include improper course registration, late release of student results, errors due to manual and tedious calculation and retrieval difficulties/inefficiency. According to them, the development of database concept is the answer to these problems, where the amount of redundant data is reduced and the possibility that the data contained on a file might be inaccurate because they were never updated. Amar, (2009) indicated that publication of students' results in the manual system takes a very long time resulting in the students remaining idle for long. Sometimes the delay in declaration of result causes heavy losses to the students as generally they cannot join further studies or appear in competitive exams and jobs because of the unavailability of certificates on time.

According to E. O. Ukem et al (2012), result systems can be improved upon by using back-up policies and audit trails. Although his solution was built with java and MySQL database, it did not establish any evidence for processing results batched in files to increase data integrity and mitigate fraud. The research could not also address what happens to a student after he/she is on suspension, deferment of studies, medical treatment, supplementary exams or related cases. We have analysed these scenarios and developed an audit trail for references.

Just like the solution by Akinmosin (2014) in his study titled “The Architectural Design of an Integrated Virtual Classroom System” the software has a login form for authentication of users and Student Registration forms for registering students every semester after payment of dues, before results are uploaded using internet browsers. His solution was built with the Oracle Procedural Language/Structured Query Language (PL/SQL) and adopts forms and reports for its implementation. However, the interface widely uses “Grades form for inputting student grade and Grades Edit form for editing erroneously entered grades.” This is a security threat and a loophole in the design that could aid intruders in accessing the database. It’s a weakness that may not properly check the mutilation of grades by users who find a way to access those forms.

Bhatt et al (2016) proposed a Credit-based Grading Scheme (CBGS) in India. It is simply a PHP-MYSQL solution that can compile results of students. The reports of the system are generated in either excel or PDF format depending on the usage. Although the grading scheme differed from that of the Nigerian system, the formula for computing the Grade Point Average (GPA) is similar: the summation of the product of the credit hours and grade points divided by the summation of the credit hours. One major advantage of such automated systems over the manual procedure is the ease of searching and list generation.

In view of the above, the requirement to have an online Result computation system in Niger state Polytechnic, Zungeru, Nigeria, is important. In the institution, Students result are computed every semester manually from the summation of Continuous Assessment and Exam score of each course taken by students, allocating grades, Grade Point (GP), Cumulative Grade Point Average (CGPA) and remarks.

The major objective of this project is to allow teachers submit/input/import students examination scores, the system compute the scores, generate a broadsheet and provide a graphical/visual analysis of the results on a semester basis. The students would be able to view their result through their profiles. Hence, in this project, we considered issues of security breaches as utmost priority and have incorporated a log file to help monitor activities that go on in the database. This will make the system more credible and enable the management account for changes made at any time by a user. It also keeps backups of record in case of loss.

CHAPTER THREE

3.0 Requirement Specification

To commence any development, it is important for any project to consider the requirements of the end users. This is to ensure that the proposed system function the way users expect it to. Therefore, requirement specifications have been gathered from sources as will be described later in this chapter. The primary focus of system requirement analysis and specification is to understand in detail what the user(s) really want the system to do. For us to know what the user needs, requirement analysis was conducted with the sole aim of gathering desirable features for the system.

This phase of the project explains how we arrived at the specifications, including problem specification, methods of gathering requirements, online review, wireframe & prototyping, initial evaluation of the prototype, implementation, and project plan.

3.1 Problem Specification

With the current trend in technology, Niger state polytechnic as a tertiary institution should have an automated means of computing student results to meet certain educational standard and requirements as well as to eliminates errors in result computation and reduce miss-use of resources and bring ease.

Currently the school is operating a manual system, every department compute student result in different formats in an un-centralised manner, some departments uses format such as excel sheet while some are completely paper based.

In previous years, a partial automated system of result computation has been introduced by the school authority to various departments, though was used as a standalone system to enable departmental coordinators enter exam scores in text file then use the command line environment to manipulate the text files to generate a computed result, the system was not friendly, requires memorising commands which in turn takes a lot of time. The system was developed using FORTRAN. Yet with the introduction of the so-called automated system, every operation still seems manual, errors recorded are even more compared to the use of excel files, which has led to many departmental coordinators to dump it. Even after result is generated, analysis is done manually thereby leading to inaccurate results. It takes the coordinator so many days to compute a single class, they become stressed up and sometimes frustrated as so much time is required. This problem leads to delay in producing result for student and delay in decision making.

If these problems are not properly taken care of, it would lead to continuous inconsistency in result generation and educational degradation. Although, many institutions have similar issues or problems but they have often used different methods in solving these problems, but I feel developing a web based system that will be used by all department in a centralised manner, enabling result computation, adding and editing student exams score, generating broad sheet result with analysis, allowing individual student see their result on time from the comfort of their homes and also enable effective decision making will have more significant difference from the methods previously used.

Finally, I decided to develop the proposed system based on the problem statements stated above, leading to the requirement stated below.

3.2. Gathering Requirements

In order to gather the requirements for the web application a heuristic evaluation was conducted on a similar web application.

The Involvement of the researcher in the process of computing student results makes it easy to investigate the existing system and to obtain detailed information on the application area to be designed.

During our findings, several effective methods of information gathering, or data collection were employed which include: evaluation and inspection of relevant documents such as result broad sheet, raw score samples, and transcript formats. The data required for the development of the proposed system were from primary and secondary sources, using a survey/questionnaire and searching from the internet respectively.

With the detail idea on how the existing system work, most of the data/information required to build the proposed system is known and acquired through primary and secondary sources.

The data from primary source is the outcome of the prototype evaluation done by the participants and the secondary source include data about the new unified Grading System, that was adjusted by the National Board for Technical Education (NBTE) in 2016 for both HND (Higher National Diploma) and ND (National Diploma) available online for public view as shown in figure 3.1 and 3.2 below, result broad sheet and result analysis format all of which could be found online. See the samples for broadsheet and result analysis in [Appendix I](#). The data gathered is used to fulfil certain requirement in the development of the proposed system, most of the information from participants would be used for the purpose of:

1. Motivation (do the target audience want the proposed system?),
2. Usability (Can the target audience work out how to use it?) and
3. Content (Does the target audience understand what the proposed system is about?).

Marked Range	Letter Grade	Weighting (Grade Point Average)
75% above	A	4.00
70% - 74%	AB	3.50
65% - 69%	B	3.25
60% - 64%	BC	3.00
55% - 59%	C	2.75
50% - 54%	CD	2.50
45% - 49%	D	2.25
40% - 44%	E	2.00
Below 40%	F	0.00

Table 3. 1: Grading System for Polytechnics in Nigeria

CGPA	Class of Diploma
3.50 above	Distinction
3.00 – 3.49	Upper Credit
2.50 – 2.99	Lower Credit
2.00 – 2.49	Pass

Table 3. 2: Cumulative Grade Point Average

3.2.1 Online Review

An online review was done to understand how other similar systems work, to find out what is lacking and how the proposed system can be a better replacement of the existing system.

From the review, the idea of using agile methodology in the development process was thought of, though due to the issue of security involved in result computation, specialists were consulted for a professional advice on how to prevent any security breach by malicious users or hackers. The professionals recommended the use of “use and misuse case” diagram representing the user requirements and possible threats or attack to a system, this was helpful as it describes the flesh of the actual requirements. The focus of the research was mainly on the computation of scores, to produce results with analysis and preventing possible threats. As an insider it was easy to get the major information's required for the use and misuse case.

Another method used is an alternative to use case diagram known as Persona. Alan Cooper introduced Persona in 1998. Personas represent potential users of a product as a fictitious character. It was very beneficial to the design process, it helps the designers to better focus and understand the user experience as they use the product ^[1].

The use of persona in the study was because of its benefit and more realistic nature; for instance, personas can describe multiple type of users, while actors are limited to a single user type. Equally the researcher also uses the use case and misuse case diagram to clearly understand and fetch out the possible attack on the functionalities of the system and to develop preventive measures to the attacks (Scott, 2018).

Use and misuse case diagram, agile methodology and persona are detailed discussed in the next chapter.

3.2.2 Prototyping

Having known the requirements, wireframes were designed to lay out the functionalities and specific content on a page, it considers the needs of users and user journeys. It was an early approach in the development process to establish the basic structure of a page before adding the content and visual design. The next chapter explain prototyping in detail.

Indigo Studio was used as a prototyping tool to develop a prototype that mimic the proposed system and was used for pre-evaluation and to collect information from users about the proposed system, in order to understand the required functionalities and better user experience. Indigo Studio is an interaction design tool from Infragistics² that enables you to explore and create functional, animated User Interface (UI) prototypes. The tool is aimed at designers who lack development skills and developers without or with few design resources. It is an interaction design tool that allows you prototype user interfaces without writing code. Indigo Studio makes it easy for anyone to rapidly design functional, animated UI prototypes that maximize usability and appeal. It promotes design process best practices and enables teams to stay focused on the end users and their stories (Conard, 2012).

3.2.3 Evaluation

The design of the prototype was based on information from an observation and direct involvement of the researcher/developer as he is part of the examination committee and an instructor in the polytechnic. It is needful to have an extra opinion as it was helpful and significance in the development process.

¹ https://en.wikipedia.org/wiki/Alan_Cooper

² Infragistics “is a global software company, founded in 1989, that publishes user interface development tools and components for a range of developer applications, across all platforms. The company is also a provider of developer support, testing tools, and UI and User Experience training and consulting services”. [Wikipedia](#)

To answer these questions; (will the target audience find satisfaction in the use of the proposed system? can they use it? and do they find ease in using it?) leads to the design of the prototype for evaluation using survey question to be completed by participants recruited. The participant will fill the evaluation form responding to questions base on their experience and observation. Typically, 10 participants were individually given time to interact with the prototype to express their opinions, thoughts and preferences about the proposed system with no negative influence through responding to the survey question in the evaluation form. Example of the evaluation questions can be found in Appendix VI.

Focus group is an alternative method that would require less time to enable approximately five participants respond to the evaluation questions and the result could be negatively inflicted as few individuals could possibly be introverts while others take control of the debate and impact the result, or possibly even introduce bias. The researcher may also have trouble controlling participant's discussions and managing process.

The responds to the evaluation questions were analysed starting with the end in mind (check of the top research questions) using the cross tabulating and theme & explore approach to filter results, Interrogate the data, analyse the results then draw conclusions. Theme and explore is the process of selecting key quotations from each respondent illustrating the themes found. It was more like a literary analysis and is good for smaller sample sizes and more complex subjects. It is particularly helpful when comparing respondents with different views and understanding of the same issue. Data were systematically analysed to ensure credibility in result evaluation.

3.2.4 Implementation

The two most popular methodologies for web development (Agile method and Waterfall) were studied to choose the most appropriate for the design and implementation of the proposed system.

Waterfall basically is a sequential model where software development is segregated into a sequence of predefined phases including feasibility, planning, design, build, test, production, and support. On the other hand, agile development methodology follows a linear sequential approach while providing flexibility for changing project requirements, as they occur (Flatworld Solutions, 2019).

Agile principles arose in the 1990s, in part, to address perceived weaknesses of the traditional waterfall development process. Agile methodology has steadily gained credibility and adherence over the last two decades (Informatica, 2019).

In 2001, Agile codified several lightweight theories under one umbrella, which revolve around short, iterative changes and frequent builds in order to make the development process responsive and adaptive to changing requirements. Agile developers believe that making the customer/business happy is more important than executing a long-range plan.

In view of the above differences between the two methodologies, we choose to implement the proposed system based on the agile method to fully leverage it advantages.

3.3 System Requirement Analysis and Specification

3.3.1 Departmental Registration:

New Student

A student whose name appears clearly in the approved list of admitted students, dispatched from exams and records unit to the department, through the polytechnic authority should be allowed to proceed to the department for screening.

Upon a successful screening in the department, the student is permitted to register in the system by uploading his/her academic credentials and other important documents to the school registration portal, after obtaining matriculation number.

The Academic Planning Unit will export the records of successful registered students of all departments in excel format from the registration portal and then upload or import it into the result computation system by the department examination officer or coordinator, as the systems are design independent of one another.

All student would be allowed to access their accounts by login in using their matric number and a default password. Students will also be required to change their password in the first login before having full access to the menus in their profile.

Course Registration

In the polytechnic, student of the same department and the same year offer the same courses, the department register courses for each student.

Extra-year student

A student shall have a resit for all courses he/she failed to pass and the scores of such courses will be treated as a special case or a carry/spill over cases.

The system should automatically select all the courses failed in past session if any.

3.3.2 Problematic student

Exam Malpractice and Related Offenses

Student who is currently facing the penalty of suspension/rustication should not be allowed to access any feature of the system until he/she is restored.

A student whose session has been nullified or rusticated for a session should automatically join a like class, courses before rustication on return after rustication.

Health Challenges

Supplementary examination can be granted for the lost semester(s); which could be uploaded as replacement.

3.3.3 System Requirements Specification

The system should accept scores uploaded by the lecturers in excel (.csv) format in the process of data input; map the results to individual records in the database; Compute the grades, grade points, GPA/CGPA of each student; and generate students result in a broad sheet format then prepare analysis of the result in a visualise or graphical format.

Functional requirements based on type users include; Students, teacher/course lecturer, Department Examination officer/Coordinator, Examination and records unit, and Academic planning Unit (System manager).

3.3.4 Functional Requirements of Registration

At the beginning of a new session, prior to the addition of new students to the database, student on the new admission list would undergo screening in their respective department and the successful ones will be given matriculation number. In most cases, not all admitted students are cleared for studies; hence,

a list of cleared students with matriculation numbers is uploaded into the system. All students uploaded are assigned default password, and an account is created for each of them. It is at this point that they can change their password at initial login and view their profiles as well as their result at the end of the semester when result is uploaded.

3.3.5 Major Requirement

- Keep track of each student records.
- The system considers special cases like suspension, deferment of admission, expulsion, supplementary exams and leave on health grounds.
- The system can automatically fetch all the REPEAT courses of the previous session, for student with extra year or carry overs.
- Restrict any change or modification of some sensitive information such as student registration/matriculation number, name, and level, year of admission and mode of entry.
- The system would enable system manager to upload staff list, student list and course list, manage the list of staff and students and keep the course list up to date, grant access to records such as, student record, staff record and courses, also to manage backups and logs
- The system would enable coordinator to upload/import, edit student score, can view broad sheet result and visual analysis and change his/her password
- Student can view their result through their profiles and make changes to their password
- The system allows Exams & records unit to have access to the result broad sheet and visual/graphical analysis of the result through their profile and can also make changes to their password.

3.3.6 Treatment of Special Students' Cases

- **Offenses/Malpractice:** The system considers some offenses which may result in rustication, expulsion, suspension or nullification of some academic work.
- **Deferment of Studies:** The system provides facilities for a student who wishes to defer his studies.
- **Health Cases:** A student with approval of a medical report has such a case created on his account indicating suspension of studies until further notice.
- **Supplementary Examination:** Approval of supplementary examination is done by the HOD into the system after the Registrar must have granted it.
- **Reinstatement of Students:** On all these cases, the student is expected to report back to the department after which he is reactivated back into the system.

3.4 Project plan

The time management plan used in this project work is described as shown in the table below.

Activities	Time (Duration)
Defining the goals and collecting necessary information about the requirements	1 weeks
Prepare records and apply for ethical approval	1 week
Create wireframe and prototype for the web application	1 week
Wait for the approval from ethics committee	2 weeks
Planning the evaluation process <ul style="list-style-type: none">- Meet with participant (consent forms)- Ask them to respond to the evaluation questions, and observe them interact with the prototype	1 weeks
Proposed database structure <ul style="list-style-type: none">- Draw an ER diagram- Set up a localhost server (Wamp server)- Use the ER diagram to define the database schema on the server	1 week
Adopt a template (User interface design) <ul style="list-style-type: none">- Setting up GitHub- Refine the template to suit the proposed system	1 week
Create personas	1 week
Create use case diagram	
Implementation using agile method <ul style="list-style-type: none">- Create the product backlog- Create first sprint backlog	1 week
Start development (using PHP, HTML, CSS and JS) <ul style="list-style-type: none">- Test code- Create second sprint backlog	1 week
Continue development and testing code <ul style="list-style-type: none">- Create 3rd sprint backlog	1 week
Continue development and testing code <ul style="list-style-type: none">- Create 4th sprint backlog	1 week
Final Sprint review and retrospective	1 week
Final evaluation of the developed system	1 week
Writing report	2 weeks

Table 3. 3: Project plan

CHAPTER FOUR

4.0 Design methods

Research was conducted on similar online result computation system, where tertiary institutions manage their student result as mentioned in chapter 2, similar features of the systems that would be beneficial in the development of the proposed system were explored, during the initial designs, persona, use and misuse case diagram, and prototypes were created, the prototype was evaluated with participants.

4.1 Personas

Cooper and Reimann defined persona as the collection of genuine representative information which may include fictitious details for a more accurate characterization (Plinio, 2005). “Persona is a user-centered design method which sets up fictitious characters to represent the different user types within a targeted demographic group that might use a site or product”³.

Two important points can be identified from the above definitions; firstly, personas are created based on reality. It accurately represents users in the real world. Secondly, most of the attributes that make up a persona are imaginary. For instance; the name, social element and photo on persona could be totally fictitious.

Particularly, persona helps the developer in user interface design and to understand the requirements of the end user, setting up a strong figure for better, reliable and consistent understanding of end user groups. The personas are created based on five different users of the proposed system to enable users have best interaction with the system (See Appendix XX).

4.2 Use and misuse case diagram

Use case diagram is a behaviour diagram in Unified Modelling Language (UML)⁴. It is used in modelling system⁵ functionalities using use-cases⁶ and actors⁷ or used to gather the functional requirements of a system, to get an outside view of a system, identify the external and internal factors influencing the system and to show the interaction among the requirements of actors (SmartDraw, 2019).

Misuse cases⁸ are included in the diagram to show possible threat or abuse that may occur within the system. Misuse case is used to describe the process of executing a malicious act against a system. Figure 4.1 shows the use and miss use case diagram of the proposed system. Figure 4.1 below is the initial “use and misuse case” diagram before consulting Dr. Craig Ramsay (Use case diagram professional) who suggested that the diagram should be separated into 5 separate use case diagrams, with each actor in a separate diagram, followed by documentation describing each use case. See Appendix II for the decomposed diagrams and full documentation. The reason for decomposing the diagram is to make the diagram less cumbersome and easy to map each actors with functionalities, and threat to mitigation and functions. This decision was helpful as it made it easy to understand and easily provide description of the diagram. The documentation of the diagram helps to remove complexity in the diagram as it describe functionalities and features that cannot be presented using diagram.

³ Wikipedia, “Personas”, <http://en.wikipedia.org/wiki/Personas>

⁴ UML is a way of visualizing a software program using a collection of diagrams.

⁵ system is something being developed or operated, such as a web site

⁶ Use cases are a set of actions, services, and functions that the system needs to perform.

⁷ Actors are people or entities operating under defined roles within the system

⁸ The term Misuse Case or mis-use case is derived from and is the inverse of use case.

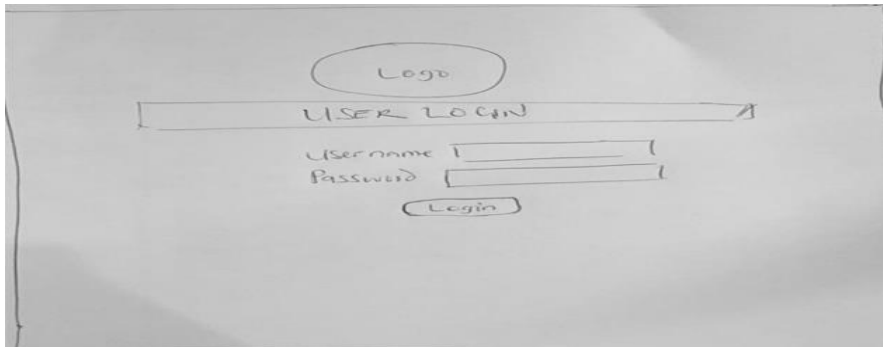


Figure 4. 2: login page

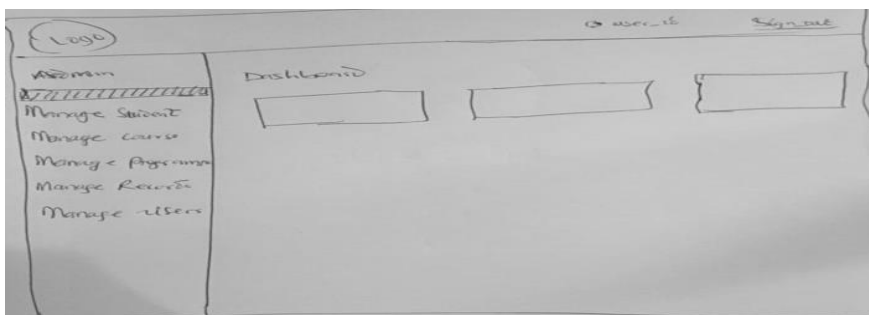


Figure 4. 3: Home Page with Accessible Menus

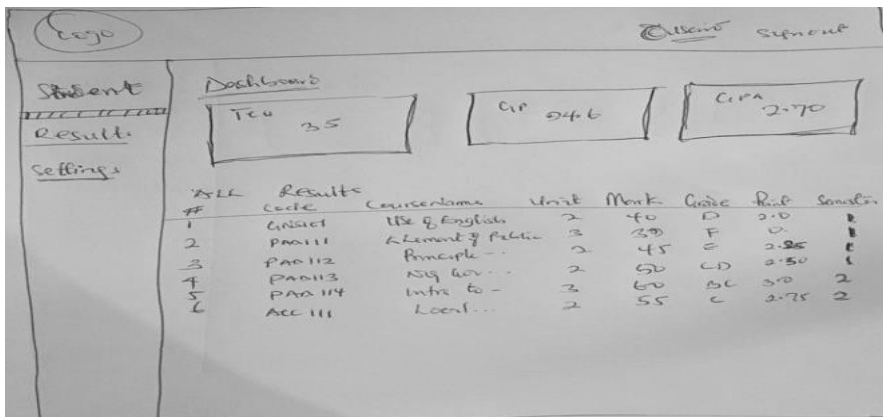


Figure 4. 4: Student individual Result, Analysis Page

4.3.2 High Fidelity – Indigo studio

To create a higher tech visual representation of the website. A high-fidelity prototyping tool called indigo studio was adopted, because it is suitable for producing a fully functional prototype of the proposed system, including interaction and dynamic features like hyperlinks and buttons. Figure 4.5 & 4.6 shows the Indigo studio software interface and the prototype of the login page. Other pages are included in [Appendix III](#)

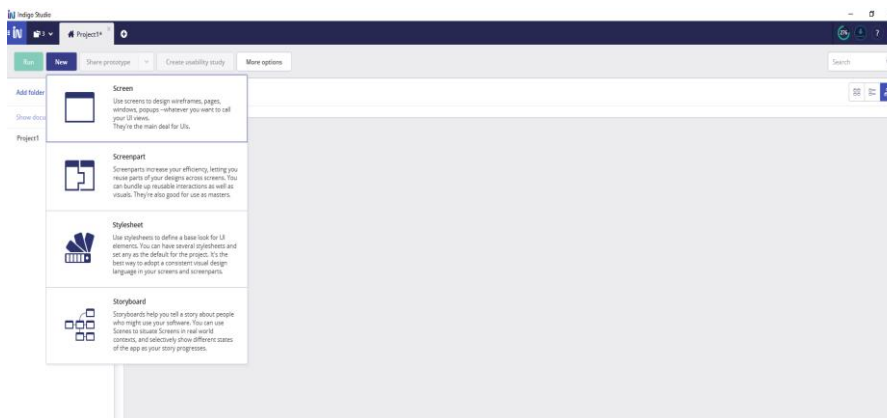


Figure 4. 5: Indigo Studio Software Interface

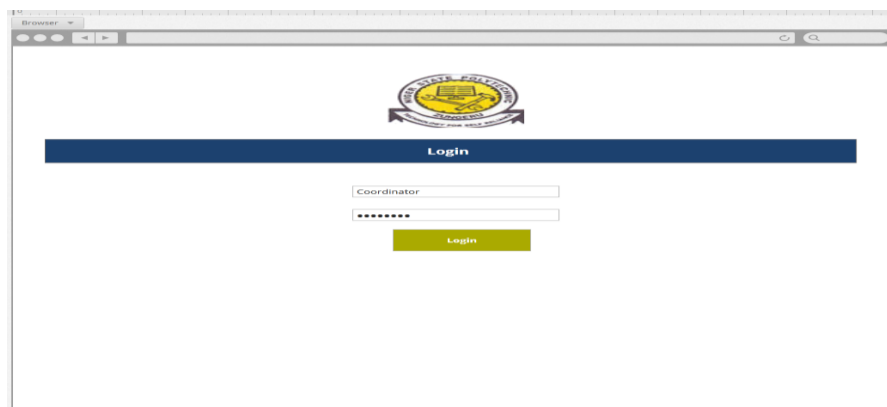


Figure 4. 6: The Prototype of the Login Page

There are many other alternative prototyping tools like; Proto.io, Axure RP and many more. Initially, Axure RP prototyping tool was the tool considered to be used for the high-fidelity prototyping. Axure was among the initial tools used by user experience professionals for interactive prototyping. It has most features available, but the negative aspect of this extensive feature is its complexity. At first glance it can be overwhelming with the depth of it options and menus. The features in Axure are still focussed on desktop experience, the built-in gestural support is limited to up and down swipes and tap without pinch to zoom or multi-finger gestures. It has a minimal built in animated transition support. Indigo Studio was identified by the developer to be easier compared to AxureRP based on the following reasons. First, the time-line based action of Indigo Studio is more intuitive compared to Axure RP's dynamic panels, especially now that the researcher is familiar with the timeline-based animation. Second, it felt more intrusive with indigo studio using it visual navigation through interactions explorer, it saves a lot of time when double-checking the linkage between screens.

Commented [MA(1): Do not understand this sentence

Commented [I2]:

4.4 Evaluation of High-Fidelity Prototype

4.4.1 Participants

In total 15 participants were invited, only 10 were able to turn up, they were given the prototype to interact with to give their feedback and opinions. Each participant was asked to read the participant

information sheet (Appendix IV) defining the nature of the task and then sign a consent form (Appendix V) indicating that they have agreed to participate in the study. Each participant was asked to respond to evaluation questions/survey (Appendix VI) based on their interaction with the prototype. The survey includes several questions and statements, where each participant uses a five-point Likert Scale or multiple-choice questions with options to select any choice of their opinions on each question or statement and open-ended question to describe based on their opinion.

The survey seeks their opinion on the aesthetics of the website. They were then asked to go through each of the main web pages of the prototype and were asked how reliable, easy and satisfied they were with each statement using the Likert Scale or multi choice options. Participants were given the option to provide any additional feedback at the end of the survey.

4.4.2 Analysis and result of evaluation

In total 10 participants evaluate the prototype, responded to the questions and the response to the questions were analysed (See Appendix XXI)

In view of the analysis, the design and implementation of the proposed system is worthwhile and feasible, since majority of the participant give a positive response on how reliable the system can be, how satisfied they are with it features, and the ease of use. Moreover, the comments, suggestions and observations of the respondents are motivating.

4.5 Database design

Database design is the organisation of data according to a database model. The designer determines what data must be stored and how the data elements interrelate. With this information, they can begin to fit the data to the database model (Teorey, 2009).

The proposed system was developed with a database as back end, such that all the data working with the system are retained, resulting to information and knowledge. In order to access and manipulate the database, a structured standard and organised computer language, (Structured Query Language (SQL)) was used. Using SQL is nearly the only choice available when developing the database because there are not known alternatives to SQL for connecting to relational databases (i.e. SQL as a protocol), but there are other alternatives ways of writing SQL in applications.

These alternatives are in form of frontends for working with relational databases, examples include; SchemeQL and CLSQL, which are probably the most flexible, owing to their Lisp heritage, but they also look like a lot more like SQL than other frontends. LINQ (in .Net), ScalaQL and ScalaQuery (in Scala), SqlStatement, ActiveRecord and many others in Ruby⁹.

The development of the database was successful following the steps of modelling data in a domain; starting with the requirement analysis, conceptual, logical then physical data modelling.

4.5.1 Requirement analysis

From the requirements gathered in the earlier chapter, we were able to identify what type of data to store, to retrieve, how to retrieve it, how the data would be displayed, the constraints on or between data items, and other requirements like speed

4.5.2 Conceptual Modelling

⁹ <https://stackoverflow.com/questions/2497227/what-are-good-alternatives-to-sql-the-language>

From what was deduced in the requirement stage, we were able to produce a conceptual model containing all the major elements. Conceptual Model was created in such a way that it can be understood by the product owner. It is a framework that describes the problem space and demonstrates correct interpretation of concepts between client and designer. It is represented with an Entity Relationship (ER) diagram comprising the main data objects (entities), properties of the entities (attributes), links between entities (relationships). In this stage no consideration was made to satisfy any database design, because conceptual model is valid for implementation in any database type.

4.5.3 Logical modelling

In this stage we start by considering the type of database (Relational Database), taking account of the architecture of the database. Relational database was the choice of database type made, because of its language choice (SQL),

In this model, detail of the tables, attributes, primary and foreign keys, linking tables and normalisation is described. This phase of modelling was more complex than the conceptual model. It is also represented with an ER diagram (see appendix VII).

4.5.4 Physical modelling

This is final and actual blue print of the database, the actual database (Relational Database) is specified, this phase represent how data were related and structured in a specific Database Management System (DBMS). This made it important considering the convention and restriction of the DBMS used during the design of the physical ER diagram, accurate use of data names, data type, of an entity and the use of reserved words were avoided, including additional constraint and integrity like primary keys, foreign keys were also specified. Visual paradigm was used as tool to create the ER diagram, this tool is customised to make it easy in creating the physical database. More detail on Visual paradigm is in 4.7.6 below.

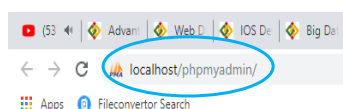
After installing Wamp Server¹⁰ the database was created finally on localhost Wamp Server from PhpMyAdmin on my PC. Wamp Server installation is detailed explain in [Appendix VIII](#). The following steps are followed to create database on localhost Wamp Server.

- First run the Wamp Server, as the Wamp Server icon turns green in the task bar menus, see the figure 4.7 below.



Figure 4. 7: Wamp Server Icon in Green

- Open the browser and type <http://localhost/phpmyadmin/>. See figure 4.8 below



¹⁰ WAMP, also known as WAMPserver is a free localhost server stack comprising Apache, MySQL and PHP for Windows. It is ideal for learning, testing and developing websites without having to use a remote web server.

Figure 4. 8: PhpMyadmin URL

- Select the database menu as marked in figure 4.9 below

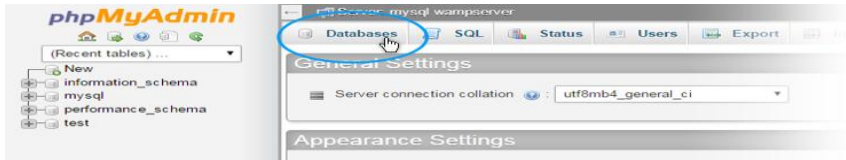


Figure 4. 9: Database Menu

- Type the database name, for example 'nipoly_consultdbsnw', then click 'create' button (by clicking on the button database is created). See figure 4.10

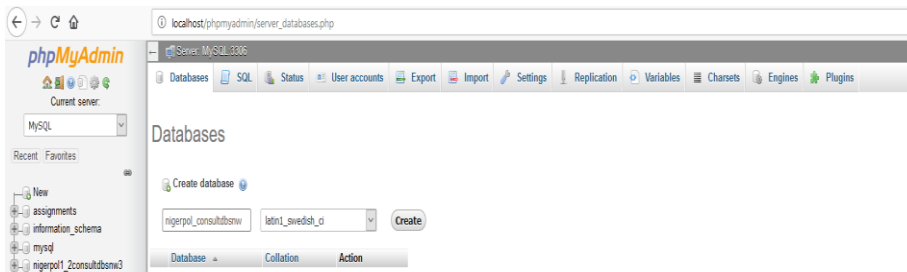


Figure 4. 10: Database Name Field

On the left side you can find the newly created database. To create table of database 'nipoly_consultdbsnw', Sql statement was written and saved with ".sql" extension, the file is then imported to create tables. See [Appendix IX](#) to view the SQL statement used to create the following tables; login table to hold user login credentials, colleges table to hold the descriptions of colleges, courses table to hold the courses offered by student, logs table to hold user logs, school table to hold description schools in a college, department table to hold the description of departments in a school, programmes table to hold the description of programmes in a department, student table to hold student data, staff table to hold staff data, user type table to hold the description of user types, and session table to record the academic session. See the database and table structure in [Appendix X](#).

4.6 Design process

The design process follows an agile approach as mentioned earlier in the report. These processes lead to interpreting the requirements as list of stories¹¹ prioritised in product backlog, then pick the stories with the highest priority from the product backlog into the sprint backlogs for implementation. An artefact of scrum¹² known as product backlog and sprint backlog from sprint¹³ planning is adopted, though we are not up to the required number to form a scrum team. We adopted these artefacts because of it benefit and importance in the development process of software. Product backlog consist of list of all things that need to be done within the project ("to do" list), it is created from the user story descriptions derived from requirements. Sprint backlogs are created from product backlog, sprint

¹¹ Story is a description of discrete functionality, it used to describe user goals.

¹² Scrum is a framework for developing and sustaining complex products, a collaborative...team that is delivering business value in a tightly coordinated fashion

¹³ Sprint is one time-boxed iteration of a continuous development cycle. Within a Sprint, planned amount of work must be completed by the team and made ready for review. The term is mainly used in Scrum Agile methodology.

backlog is a subset of product backlog, and it is more detailed than the product backlog. See product backlog and sprint backlog in [Appendix XI](#)

The total number of sprints in this project is 5, in each sprint we made different sprint backlogs until all the stories in the product backlog were exhausted. Though due to some unforeseen challenges some of the stories that we could not complete in any sprint is moved into the next sprint with the highest priority. Every sprint last for 1 week and at the end of every sprint we do retrospective and sprint review together with my supervisor to answer the 3w questions, what went well? What went wrong? What should be done next?

4.7 Technologies Used

The technologies used in this project are to enable the developer to develop the proposed system through frameworks/environments including APIs and scripting/programming languages. These technologies are explained as follows:

4.7.1 Localhost Wamp Server

Wamp is a technology used for developing and testing the website on a server locally hosted on user PC. Other alternatives include XAMP and LAMP. The purpose of using Wamp was because it allows experiment with code, design, plugins and frameworks safely (Darren, 2009). It could be dangerous to design and maintain tasks on a live website. There is more to lose depending on the size of the website. Therefore, it is better to use localhost to avoid nasty surprises. Figure 4.11 below shows the index page of Wamp Server

After WAMP is downloaded¹⁴ and installed in user PC, there will be no need to install MySQL database. Wamp have MySQL included and is accessed through PhpMyAdmin as shown above.

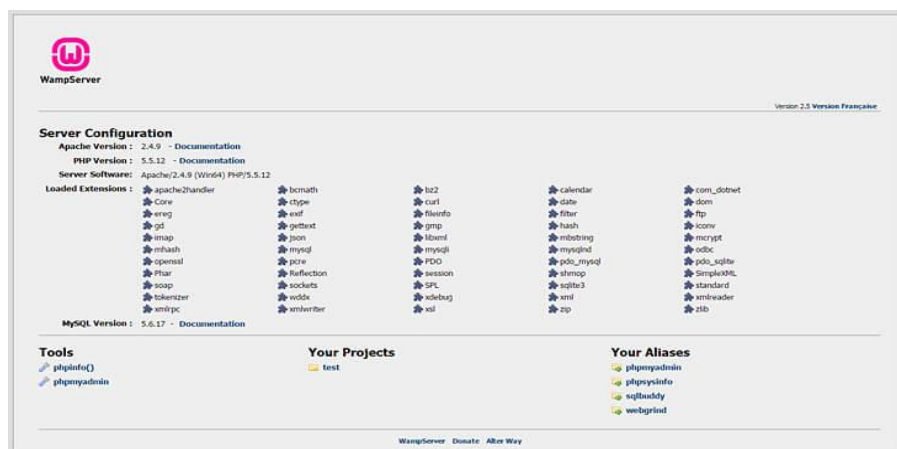


Figure 4. 11: Wamp Server Index page.

4.7.2 Server-side scripting language

The Server-side scripting language used in the development of the proposed system is PHP embedded in HTML. It was originally designed in 1994 by Rasmus Lerdorf, it is a general-purpose server-side

¹⁴ <http://www.wampserver.com/en/>

programming language used primarily for the purpose of developing websites. Initially known as Personal Home Page, now called Hypertext Pre-processor. One of the most popular social networking sites such as Facebook and a well-known educational institution are both based on PHP, therefore, making it popular and increasing its credibility. However, the use of PHP in the development is because it is simple and can be maintained, improved, updated with ease periodically¹⁵ and some of the advantages such as; cross platform¹⁶, easy database connection¹⁷, easy to use, and an open source¹⁸.

4.7.3 Back end and front end

The technology used for the back end is MySQL, provided by Oracle, it is an open source relational database management system (RDBMS) based on Structured Query Language (SQL) (Rose, 2018). MySQL can be virtually used in all platforms such as; Windows, UNIX, and Linux. It is usually associated with online and web application. MySQL is a significant component of LAMP, LAMP is a web development environment that uses Linux OS, a web server (Apache), relational database management system (MySQL) and PHP as the scripting language. Since the database choice in this project is Relational database, therefore SQL is considered the language choice as it is portable easy to learn and it uses an established standard adopted by American National Standards Institute (NSI & ISO).¹⁹

The technology used for the development of front-end is HTML (Hypertext Mark-up Language) and CSS (Cascading Style Sheets), these are two of the core technologies used for designing websites²⁰.

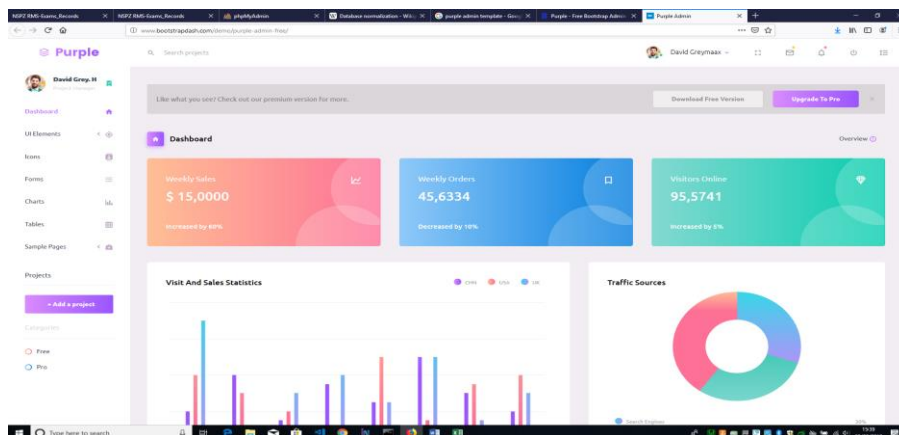


Figure 4. 12: Bootstrap Admin Dashboard Templates

In order to achieve a responsive design²¹, a purple-free-admin template was adopted. Purple-free-admin is designed by bootstrap Admin Dashboard Templates, built on Bootstrap; a popular JavaScript

¹⁵ <https://www.w3trainingschool.com/php-advantages>

¹⁶ PHP is supported by most Operating Systems such as; Windows, UNIX, and Linux.

¹⁷ There is a built-in module in PHP that helps it to connect easily with database, this makes it demanding in the field of web development to develop websites that are data driven.

¹⁸ To learn PHP no intensive manual or studying is required, it commands, and function can easily be understood and can easily figure out what is does from the name.

¹⁹ <https://www.ntirety.com/why-you-still-need-a-relational-database/>

²⁰ <https://www.w3.org/standards/webdesign/htmlcss>

²¹ Responsive Web Design is about using HTML and CSS to automatically resize, hide, shrink, or enlarge, a website, to make it look good on all devices (desktops, tablets, and phones)

framework for developing responsive pages. Using npm²², Gulp²³, Bower, Sass and Git, development was made easy, application are built faster.²⁴

However, the template was used by the developer, because of its usability features, having structures that accommodates all other structures introduced during the process of the development. The template came with pages that can be customised to fit developer need. In the process of the development the developer uses only the index page of the template including the attached CSS page to retain it look. Out of the hundreds of templates designed by bootstrap admin dashboard templates, the developer chooses the purple admin template, because of its simple touch, colour combinations, side bar navigational menu, most importantly its complete responsive nature. Figure 4.12 above is the adopted template.

In addition, JavaScript was used to add some dynamic features such as; event-driven, functional, and imperative (object-oriented) programming styles. JavaScript is one of the core technologies of the World Wide Web used alongside with HTML and CSS, it enables interactive web pages. To extend JavaScript features on the web application, jQuery was used.

4.7.4 Visualisation / Charts

The technology used for visualisation in this project is known as Google charts tools (API). Google Chart API is an interactive web service used in creating graphical charts from data supplied by users. It is conveniently embedded in the web page by simply using JavaScript and HTML tags (div or image tag) with an id in the web page. It is used by loading some google libraries, outline the data to visualise, customise the chart then finally create a chart object with an id used in any of the HTML tags

Google chart tools are powerful, simple to use, and free. It has a rich gallery of interactive charts and data tools like line charts, complex hierarchical tree maps. The gallery provides a variety of ready to use charts types²⁵. Google chart API was chosen due to its simplicity and standard visualisation feature.

4.7.5 GitHub

GitHub is an online tool that is used to store files, basically code on the cloud. It is not just a storage, but a full and advance version control system powered by Git. Linus Trovalds the founder of Linux, started it as an open source version control system, such as; Mercurial, CVS and Subversion.

Git is helpful to developers when they create applications; they make changes to the code constantly to release new versions of the application after official release. Git as a version control system keep these revisions, storing the changes in a central repository which allow developers to cooperate, enabling them to download new version of the application, make changes and upload the newest revision. All developers can see and have access to the new changes, downloads and make contributions (Brown, 2017).

However, GitHub is used in this project by the developer to keep records of all files including changes made during development process, keep track or get access to the files anywhere anytime. Among all version control system, Github is chosen because most developers preferred using it and it has many advantages compared to other version control systems. It keeps file changes more efficiently and better integrity is ensured. See explanation on how GitHub works.²⁶

²² <https://www.npmjs.com/>

²³ Gulp is a tool that helps you out with several tasks when it comes to web development. It's often used to do front end tasks

²⁴ <https://www.bootstrapdash.com/product/purple-free-admin-template/>

²⁵ <https://developers.google.com/chart/interactive/docs/>

²⁶ <http://git-scm.com/book/en/Getting-Started-Git-Basics>

4.7.6 Visual paradigm

Visual Paradigm is a tool used in this project to design the Entity Relationship Diagram used in the physical design of the proposed system. It is a software tool designed for software development teams to model and manage business information or intelligent system and development processes respectively. This tool supports many key industry modelling languages and standards such as Unified Modelling Language (UML), SysML, BPMN, XMI, SoaML and many more.

4.7.7 Accessibility Checker (AChecker)

AChecker is a free online open source web accessibility evaluation tool that was developed in University of Toronto in 2009 by the Inclusive Design Research Centre. On this tool, a user can upload a web page through its URL or by chosen uploading its HTML file locally from the user PC and can also select the guidelines in which the evaluation could follow such as; WCAG 2.0, WCAG 1.0, the HTML Validator, Stanca Act, Section 508, and BITV. (Mifsud, 2011)

However, there are many more accessibility checkers online, but the developer chose to use AChecker because it gives user different option of website submission methods, where user can either submit through the URL, upload the file or paste the row code and has more accessibility guideless reference than other known tools.

4.8 Design Decisions

During the design of the proposed system, decisions were made leading to changes that results to adding and correcting some features of the system. These decisions are geared towards a better development of the proposed system.

In the front-end design, the proposed system was meant to have three user access roles which are; Students, Coordinator/Examination officer and teacher/course lecturer. To have proper account and control of the system for security reasons more roles were added, dividing roles of the coordinator into; System Manager and Examination & records. System manager control the entire menus accessible by other users, add, view and edit student, courses, programs, and staff records. The decision of creating more roles was made to reduce the task on user and allow proper management and control of the system. Moreover, a decision was made to enable user to generate a blank worksheet in .csv format for entering courses, student and exam score data. This decision was as a result of an observation made about user importing incorrect file format and wrong data column order.

In the back-end design, the first change made was adding an entity “session” in the ER diagram, the entity was needed to keep track of academic sessions. Furthermore, the database initially comprises of fourteen tables, but due to a decision made to create menu control and user logs, four more tables (user logs, menu control, submenu control, and user type tables) were added. User logs table was created to keep logs and keep track of all changes made by users in the database, and this will control miss use of the system by its users, mitigating unwanted changes that a user might make. The decision to create user logs was thought of when a use and misuse case diagram was drawn and documented. Menu and sub menu control tables were created to enable the user (system manager) have full control of user menus, the system would be able to enable or disable user menu to deny them access to the functionality of the menu. User type table was created to ensure normalisation in the database to reduce data redundancy and improve data integrity by ensuring that dependencies are enforced properly²⁷.

²⁷ “Normalization entails organizing the columns (attributes) and tables (relations) of a database to ensure that their dependencies are properly enforced by database integrity constraints. It is accomplished by applying some formal rules either by a process of synthesis (creating a new database design) or decomposition (improving an existing database design)”

All the decision made caused a great improvement in the development process, hence, resulting to better outcomes in terms of the security of the back-end and interactive interfaces from the front-end.

CHAPTER FIVE

5.0 Implementation and testing

The development of the software follows an agile method as described in previous chapters, all the items in the sprint backlog are implemented, strictly following the stories based on their priorities in the backlog. The developer uses Test Driven Development (TDD) to detect early problems and get quicker feedbacks in form of regressive testing. This type of test is also known as test first programming/development. However, the developer work in a very short cycle adding failing test²⁸ and then making it work. This is in line with the assertion made by Beck, (2003) indicating that two rules are to be followed in testing; “Never write a single line of code unless you have a failing automated test” and eliminate duplicates.

In TDD every piece of code is tested in the following process; building an automated failing test before writing a single line of code, only required code is written for the test to arrive as fast as possible at a working code, rewrite the code as soon as the code works, then refactor²⁹ it to eliminate duplicate.

5.1 Software development

The development started with creating directory on the server in the www directory. The www directory is installed to C:\wamp\www\ by default. The directory is accessed via the WAMP menu in the system tray as shown in figure 5.1 below.

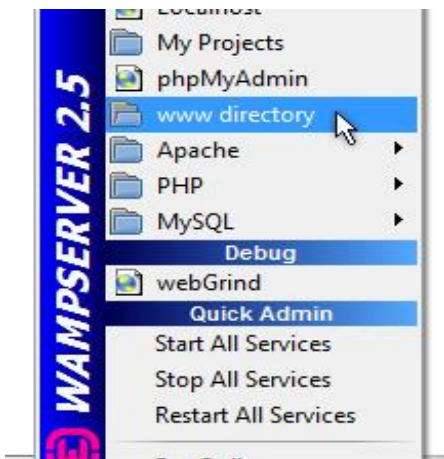


Figure 5. 1: WAMP Menu

A new folder named “compute” is created in the **www directory**. All other pages and folders that comprise the project reside in the folder “compute” on the server. A repository of GitHub was created to keep all related files and folders saved as shown in figure 5.2 below.

²⁸ Test represent requirement that must be fulfilled by the code. If there is no requirement (i.e no test), there is no need for implementing anything, therefore no need for any code.

²⁹ “Process of making changes to the code, such that it does not alter the external behaviour of the code yet improves its internal structure”. Martin Flower, “Refactoring”, 1999 www.refactoring.com

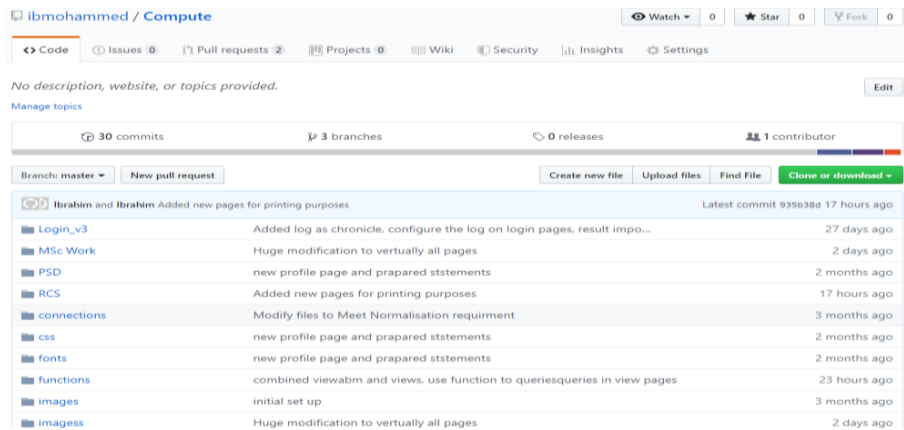


Figure 5. 2 GitHub Repository

Example of some commonly used Git command line instruction used to facilitate GitHub activities on PC, which will enable making changes and update to the repository are;

- \$ git status: List all new modified files to be committed
- \$ git add: Snapshots the file in preparation for versioning
- \$ git commit: Record file snapshots permanently in version history
- \$ git push: Uploads all local branch commits to GitHub

Following the list of requirements in the sprints, the sprint backlog has the list of tasks with estimated complexity, task, and estimated effort. The development starts with the task from top to bottom of the table. All task completed are coloured in green, the task in progress are coloured in orange while task not started are coloured red. See the sprint backlog in [Appendix XI](#).

The login page was developed immediately after adopting the template and was completely independent of the adopted template as it was fully coded by the developer as shown in figure 5.3 below.

```

1 <?php
2 if (!isset($_SESSION))
3 {
4     session_start();
5 }
6 error_reporting(-1);
7 ini_set('display_errors', 1);
8
9 require_once('Connections/logs.php');
10 require_once('Connections/connection.php');
11 require_once('Functions/queries.php');
12
13 $loginFormAction = $_SERVER['PHP_SELF'];
14
15 if (isset($_GET['accesscheck']))
16 {
17     $_SESSION['PrevURL'] = $_GET['accesscheck'];
18 }
19
20 if (isset($_POST['submit']))
21 {
22     $loginUsername = $_POST['username'];
23     $loginUsername = preg_replace("/[a-zA-Z0-9\%]/", "", $loginUsername);
24     $password = $_POST['password'];
25     $password = preg_replace("/[a-zA-Z0-9\%]/", "", $password);
26     $MM_redirectLoginsuccess = "index.php";
27     $MM_redirectLoginsuccess2 = "s_profile.php";
28     $MM_redirectLoginsuccess3 = "manage.php";
29     $MM_redirectLoginsuccess4 = "exams_record.php";
30     $MM_redirectLoggedIn = "logins.php";
31     $MM_redirectToReferrer = true;
32     $stmt = mysql_prepare($conn,
33         "SELECT id, username, password, prog, t_user, status
34         FROM loginb1 WHERE username = ?");
35     // bind parameters for markers '?'
36     mysql_stmt_bind_param($stmt, "s", $loginUsername);
37     // execute query
38     mysql_stmt_execute($stmt);
39     mysql_stmt_bind_result($stmt, $id, $uname, $pwd, $prog, $t_user, $number);
40     mysql_stmt_store_result($stmt);
41     // fetch value '?'
42     $deptcode = mysql_stmt_fetch($stmt);
43     $_SESSION['myid1'] = $id;
44     $_SESSION['uyp'] = "staff";
45     $_SESSION['t_user'] = $t_user;
46 }

```

Figure 5. 3: Login Page Code

The development of other pages continues following the list of stories in the sprint backlog

5.2 Security

Security is an integral part and important issue to any software especially when it involves storage of data or information that could be sensitive. Taking this into consideration the developer makes provisions for possible mitigating or preventive measures to tackle issues involving threat or attack from malicious users such as the following:

5.2.1 Cross-Site Request Forgery (CSRF)

This is the type of hacking that occurs when a hacker tricks a user to load into a webpage from a site on which they are currently authenticated. Therefore, it becomes advantageous to the hacker, due to the user's authenticated state. The developer uses onetime token (a single-use password or PIN passcode) associated with a session³⁰ in a post requests from a user to prevent CSFR attack. The token is generated and saved in a session on every form that send data to the server to be loaded in to a database, the token is received on the server before loading data into the database. This helps in preventing hackers access to load data from their end into the database, since they do not have or know the token to the server for authentication. This mitigating or preventive measure for this type of attack is important as tricking users has become so rampant. See example of the code in figure 5.4 and 5.5 below:

```
1. <?php
2. session_start();
3. $token= md5(uniqid());
4. $_SESSION['delete_score_token']= $token;
5. session_write_close();
6. ?>
7. <form method="post" action="save_confirm.php">
8. <input type="hidden" name="token" value="<?php echo $token; ?>" />
9. Do you really want to delete?
10. <input type="submit" value=" Yes " />
11. <input type="button" value=" No " onclick="history.go(-1);" />
12. </form>
```

Figure 5. 4: Comfirm_page.php

From figure 5.4 above, the code on line 2 initiate the session, line 3 generate a unique token using PHP unique id function and hashed with MD5 hashing algorithm, line 4 store the generated token in a session named "delete_score_token", in line 8 is a hidden field that hold the generated token as a value, and line 10 & 11 is the confirmation buttons "Yes" and "No". If user click on the Yes button, the form is submitted to the page named "save_comfirm.php" on the server specified on line 7. See the code of the page save_comfirm.php in figure 5.5 below.

```
1. <?php
2. session_start();
3. $token = $_SESSION['delete_score_token'];
4. unset($_SESSION['delete_score_token']);
5. session_write_close();
6. if ($token && $_POST['token']==$token) {
7. // delete the record
8. } else {
9. // log potential CSRF attack.
10. }
11. ?>
```

Figure 5. 5: Save_Comfirm.php

³⁰ A session begins when a user logs in to or accesses a computer, program or web page and ends when the user logs out of or shuts down the computer, closes the program or web page. A session can temporarily store information related to the activities of the user while connected.

In figure 5.5, on line 2 the session is initialised, the token stored in the session named `delete_score_token` in figure 5.4 above is stored in a variable named `token`, the code in line 4 delete the session (`delete_tscore_token`). In line 6, the stored token in the variable “`token`” is compared with the stored token passed from the form in figure 5.4 in line 8 above. If they are equivalent it performs a delete action otherwise it declines, meaning it’s a CSRF attack.

An alternative to this solution is for the user to use a framework like Django that has an in-built procedure for this purpose, this alternative was not used by the developer because the developer is not using any framework rather strictly writing the codes.

5.2.2 Password Hashing

The hashing function use in this project is the “bcrypt password hashing function designed by Niels Provos and David Mazieres, based on the Blowfish cipher, and presented at USENIX in 1999” (Provos, 2012). If this function is used on any phrase, it turns to a mathematically transformed random strings of characters, unreadable by human.

On the system whenever a new user is created, the passphrase submitted by the user is hashed using the bcrypt hashing function before submitting it to the database so that it cannot be misused. However, the developer choses to use bcrypt because it can be implemented in PHP and is a function that is adaptive, it gives room to increase it iteration count which makes it slower, therefore, remains resistant to brute-force searching attacks no matter the increase in computation power of an adversary. See example below.

```
1. $password = $_POST["password"];
2. $hash = password_hash($password, PASSWORD_BCRYPT, $options);
```

Figure 5. 6: Bcrypt Password Hashing Function

```
1. If(password_verify($existing_pass, $entered_pass)){
2. // login successful
3. }else{
4. //incorrect login detail
5. }
```

Figure 5. 7: Bcrypt Hashing Verification Function

In figure 5.6, line 1 is the user password entered through a form, and line 2 is the PHP hashing function that accept three arguments (`$password`, `PASSWORD_BCRYPT`, `$options`) as shown above. The first argument is the password of the user to be hashed, the second is the type of hashing function and the last is the salt³¹.

5.2.3 SQL Injections

SQL Injection is a common hacking technique that can destroy a database³². This usually occurs when a malicious user input SQL Statements into a text field in a form, for instance username or email field, instead of username or email which will be run on the database. This attack is prevented or mitigated by the developer ensuring that user inputs are validated, unwanted characters are stripped from users input and ensure that only valid data has been inputted by the user. For example, a form has a date input field and if a user attempts to insert an SQL statement instead of a valid date, the system will automatically prompt the user with a generic message like “Please enter correct data for all fields”.

³¹ “Salt is random data that is used as an additional input to a one-way function that “hashes” data, a password or passphrase”.

³² https://www.w3schools.com/sql/sql_injection.asp

An Alternative to this is to use web development framework that automatically validate user input. The developer decided to write the code to prevent this attack to have a complete control of the code and be able to make changes in the code in case of any behaviour as a result of the code. Since it is not easy to make changes to the code generated by the frame work. See example of the code in figure 5.8 below.

```
1. <?php
2. If(isset($_POST["Submit"]))
3. {
4.     $loginUsername = $_POST["username"];
5.     $password = $_POST["password"];
6.     $loginUsername = preg_replace("/[^a-zA-Z0-9|s|/]/", "", $loginUsername);
7.     $password = preg_replace("/[^a-zA-Z0-9|s|/]/", "", $password);
8. // other code continues
9. }
```

Figure 5. 8: Strip Unwanted Characters from User Input.

In figure 5.8 above, the code in line 2 check if user has submitted a form, on line 4 and 5 the username and password submitted by the user is stored in the variable “\$username” and “\$password” respectively, then on line 6, the PHP function “preg_replace()” strip all the characters inputted by the user as username except characters a-z, A-Z, 0-9, space and /. Also, on line 7, the function strips all the characters inputted by the user as username except characters a-z, A-Z, 0-9, space.

5.2.4 Cross Site Scripting (XSS)

Cross-Site Scripting occurs when the content user submit is not escaped or cleared by the website before it is rendered into HTML. An attack of this nature can lead to users disposing vital or sensitive information to malicious user or hacker unknowingly. This is prevented by the developer using Contextual output encoding/escaping of string and validating untrusted HTML input. See example in figure 5.9

```
$Studentsname = $_POST['studentname'];
$Studentsname = mysqli_real_escape_string($Studentsname);
```

Figure 5. 9: Validating Untrusted HTML Input

5.2.4 Other Security Challenges

Other security challenges where tackled, these challenges are misuse of user right, for instance when a user try to make amendment to students result without reasons or permission from the top management. A means of preventing this type of security challenge was developed by creating user logs to keep track of user activities on the system. The decision to include user log to mitigate this misuse was as a result of consulting Dr. Sasa Radomirovic (a professional in the field of information security). He also advised on the use of misuse cases together with use cases for easy detection of security threat and solutions to the threat such as the once mentioned above.

5.3 Testing

Agile method of testing known as Test-Driven Development (TDD) was used to test the smallest variable or every functionality in the system during development. After every functionality of the system was tested and debugged during development and the system was finally built, the next test carried out was the accessibility checked, this is described below.

Furthermore, the developer tested the system for browser compatibility using popular browsers such as Mozilla Firefox, Microsoft Edge, Google chrome, Safari on Mac. On each browser the interfaces were

rendered as expected with no noticeable difference shown. Hence, the system is compatible with the above-mentioned browsers.

5.3.1 Agile Testing (TDD)

The test carried out was recorded and the summary is as shown below (see the detail of the test report in [Appendix XII](#)).

Executed	Passed	41	43
	Failed		
	Total test Executed (Passed + Failed)		
Pending			2
In progress			0
Blocked			0
(Sub-total) test planned			43
(Pending + in progress + Test executed)			

Table 5.1: Test summary Report

The test report serves as a document ensuring that the current status of the project and quality of the product is informed, to enable corrective action to be taken if necessary, and finally determine whether the product is ready for release. The test report shows that the product has passed the test and is ready for delivery.

4.3.2 Web Content Accessibility

As previously stated in chapter 4 Accessibility Checker (AChecker) was used as a tool to test the web pages following the Web Content Accessibility Guidelines (WCAG) to improve Website Accessibility and the result is as follows:

Wednesday August 7, 2019 10:00:48

Source Title: NSPZ Student Profile

Accessibility Review (Guidelines: WCAG 2.0 (Level AA)) Report on known problems (2 found):

3.1 Readable: Make text content readable and understandable.

Success Criteria 3.1.1 Language of Page (A)

Check 48: Document language not identified.

Repair: For HTML documents add the lang attribute and a valid ISO-639-1 two letter language code to the opening HTML element. For XHTML documents add both the lang and xml:lang attributes with a valid ISO-639-1 two letter language code to the opening HTML element.

✖ Line 3, Column 1:

```
<html><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
<title>NSPZ Stude ...
```

Check 49: Document has invalid language code.

Repair: Add a valid 2 letter or 3 letter language code as defined in the ISO 639 specification to the HTML 'lang' attribute. For XHTML, both 'lang' and 'xml:lang' must be set.

✖ Line 3, Column 1:

```
<html><head><meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

```
<title>NSPZ Stude ...
```

Report on likely problems (0 found):

🎉 Congratulations! No likely problems.

See detailed accessibility test report in [Appendix XIII](#)

The tool indicated the above “known” problems and recommended that the document languages should be added and should follow the “valid ISO-639-1 two letter language code” to the opening HTML element. It was added as shown below. Though this is not really a problem that necessarily frustrate user in terms of accessibility and by default the language is set to “en” meaning English.

```
<html lang="en"> <head> <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

Figure 5. 10: Setting Default Language to English

5.4 Final product

5.4.1 Evaluation of final product

Survey Monkey

The final evaluation of the web application was conducted using an online questionnaire targeted mostly to initial participants in the evaluation of the high-fidelity prototype. The questions were transferred to an online survey tool (Survey Monkey) to have better layout arrangements and easily administered to the participants. The questionnaire was open to participants for one week. The questions and the analysis of participants' feedbacks in table and visual form can be seen in [Appendix XIV](#).

5.4.2 Result of Evaluation

The result from the analysis of the participants' feedback was positive since the participants respond as follows;

System Simplicity

All the participants rate the system's simplicity with at least 4 stars, this means that the navigation menu of the system is simple to use and can be accessed easily.

The participants found it easy allocating course(s) to staff, as not much is required to use the allocation form. The backup feature according to all the participants is very useful. The system is overall very good as rated by all participant.

System Performance

The performance of the system is also rated with at least 4 stars, meaning that the system is fast loading and displaying pages, the experience of user on the system is said to be satisfactory by all the participant. Generating .CSV file format for data entry is very helpful to all the participants. All the participants are satisfied with the output of the visual result analysis.

Personal Opinions and Comments

In summary the opinion of the participants is that, for now the system is working fine and can be improved on when observed in features as it continues to work. One of the participants commented that the system should be recommended for immediately and another participant said the passphrase of the user (students) should be sent to their mail upon the upload of their records while other three participants said no comment.

In view of the above analysis, the online result computation system meets user requirements and recommended for use

NASA-TLX

In addition, Nasa-TLX questionnaire (see [Appendix XV](#)) was used. The Nasa TLX was also targeted to the initial participants as in the prototype evaluation. This was used to measure the effort of users performing tasks on the system from beginning to the end. "NASA-TLX is a multi-dimensional scale designed to obtain workload estimates from one or more operators while performing a task or immediately afterwards" (Hart, 2006). It was created by Hart et al in 1988 and consists of six subscales to measure workload (Physical, Mental, frustration effort temporal demand and performance) which all involves weighing and rating.

- **Weighting part**
In this part 15 workload measures comparison cards are personally administered to participants using my PC. The participants compare 2 workload measures from each card and tell which one is more important in relation to their experience during the task. See [Appendix XV](#).
- **Rating part**

In this part, 6 workload subscales were administered to each participant requiring them to mark the appropriate point that matches their experience. Each line has two points from low on the left to high on the right. Subscale Performance has a scale of good on the left and bad on the right. The scale increments by 5 on each line. Scale is 0-100. This can be seen in [Appendix XV](#).

The weighting and rating are multiplied for each subscale to form an Adjusted Rating for that subscale. Then all Adjusted Ratings are summed up and the total is then divided by 15 to gain the Task Load Index score or rating for that task. The template containing the calculated scores can be seen in [Appendix XVI](#).

5.4.3 Tasks worked on

Five participants were recruited to conduct series of tasks on the website for Online Result Computation System. These tasks include:

- Login and logout as five different users of the system
- On all the user navigational menu perform the task in each menu
- Then respond to the question on the NASA-TLX platform and online questionnaire (SurveyMonkey).

5.4.4 NASA-TLX work load results (Average Scores)

The two parts of the NASA-TLX was completed by the participants after completing the tasks. The [figure 5.11](#) below gives graphical representation of the average score of tasks based on the experience of each participant.

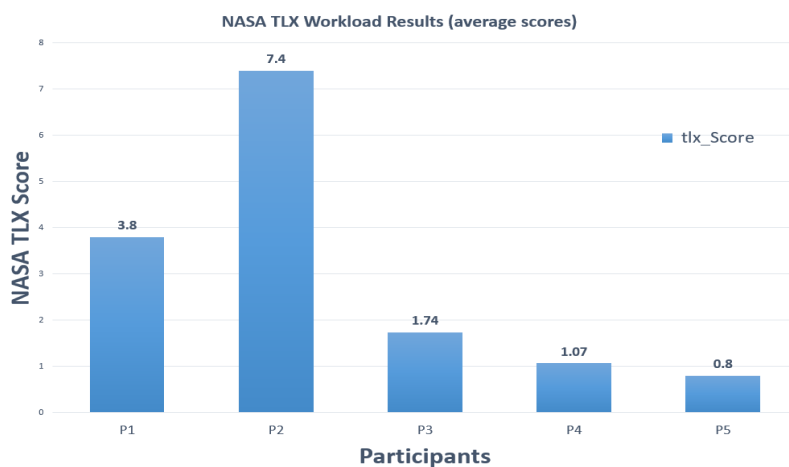


Figure 5. 11: Nasa-TLX workload chart

In this project, the researcher consider participants with the score below 8 to be acceptable. From the above graph in figure 5.11, it is shown that all participants scored below 8. The highest score was P2 with a score of 7.4 while the lowest score was P5 with a score of 0.5. Therefore, based on the experience of participants (using the system), the average scores from the graph falls below 8, hence the tasks performed were not frustrating and are easy to complete. From this result it can be concluded that the system is easy and intuitive to use with the fulfilment of all the system requirement specification specified.

5.5 Functionality and interfaces of the completed system

To better understand how the system works, all the menus will be explained according to user roles, using interfaces to aid explanation. To run the system on a local server (Localhost Wamp Server), a user is required to start the server as explained in the chapter above, on a browser, in the address bar type the URL: (http://localhost/The_RMS/SEET/) to access users (System Manager, Teacher, Coordinator and Student) login page.

Currently the Online Result Computation System is hosted on the Students server (ZENO) and can be accessed through the URL: https://zeno.computing.dundee.ac.uk/2018-msc/ibrahimmohammed/The_RMS/SEET/

5.5.1 Login

All users of the system are required to enter their login details on the login page to access their dashboard and menus, when the users enter their login detail the system authenticate and verify their detail, if the details are correct then user is given access to the system and a log is created indicating who logged in, and when they logged in, otherwise the user is prompted “incorrect login details” and denied access. See the login page in figure 5.12 below.

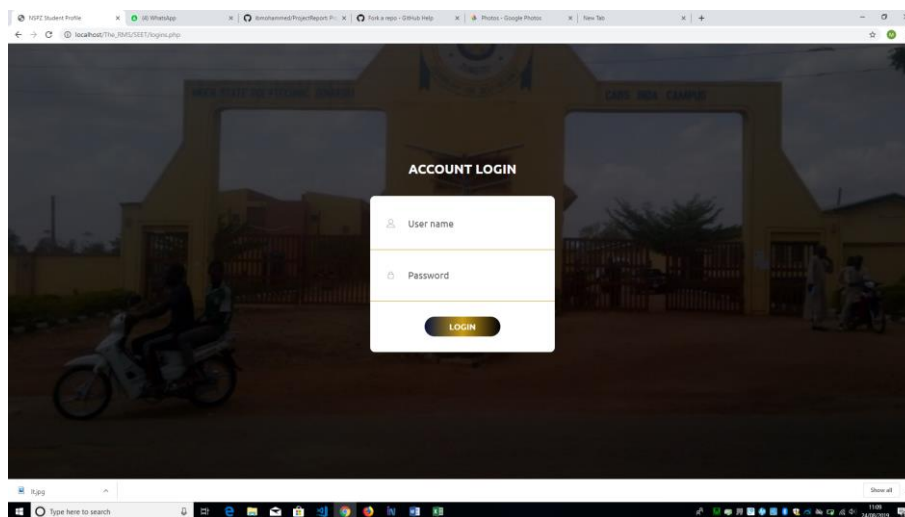


Figure 5. 12: User Login Page

5.5.2 User Dashboard

Every user when logged in successfully, will be presented with a dashboard, and navigation menus and also a logout icon, profile drop down menu at the top right-hand corner of the page, every user is presented with different dashboard, and navigation menus.

The dashboard of the users (system manger, and exams & records) shows a link for viewing colleges, schools and departments. For a coordinator as a user, the dashboard shows the name of the college, school and department in which the coordinator belongs, and a link to view the programmes in the department. The dashboard for a student as a user shows the course unit offered by the student, the cumulative grade point (CGP), the cumulative grade point average (CGPA) and the courses offered and result of each course. The dashboard for a teacher/instructor/lecturer as a user shows the name of the

college, school, department to which a teacher / lecturer belongs, and the courses allocated to user (teacher). See all the users' dashboards below in figure 5.20, 5.21, 5.22 & 5.23

5.5.3 Users Navigation Menu

The navigation menu of the user (System Manger) comprises of the menus; generate templates, manage backup, manage courses, manage programme, manage student and user access control (see figure 5.13).

The navigation menu of the user (Coordinator) consist of the menus; Allocation, Generate Template, Manage Result, Manage Score, Result Analysis and Settings (see figure 5.14).

The navigation menu of the user (Teacher) comprises of Exam score, generate template and setting (see figure 5.16).

The navigation menu of the user (Student) consist of menus; Semester Results, and Settings (see figure 5.15).

The navigation menu of the user (Exams and Records) consist of the menus; Courses, Result, Student data and Settings (see figure 5.17).

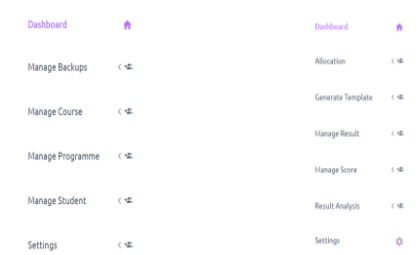


Figure 5.13: System Manager Menu

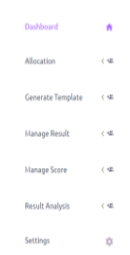


Figure 5.14: Coordinator Menu

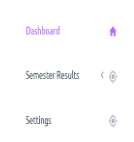


Figure 5.15: Student Menu

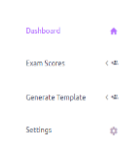


Figure 5.16: Teacher Menu

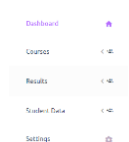


Figure 5.17: Exam & Records Menu

5.5.4 Profile Dropdown Menu

The “profile dropdown” menu shows the username/id of user and gives a submenu in dropdown when clicked, the users (Student, Teacher, exams & records) have the same submenu (logout) on the dropdown while the system manager have the submenu as Activity logs and logout.

The activity log sub menu enables the system manager to view user logs and the logout is to exit the user out and destroy all session attached to the user on login. Figure 5.17, & 5.18 shows the “profile dropdown”.

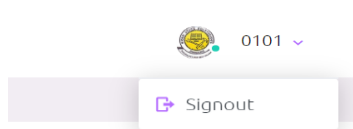


Figure 5.18: Other Users Profile Dropdown Menu

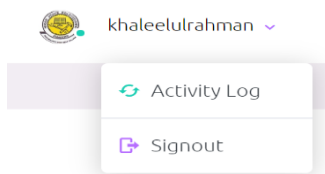


Figure 5. 19: Users (System Manager) Profile Dropdown Menu

5.5.3 Settings

Setting is a menu common to all the users (coordinator, Teacher, Student and exam & record), when the menu is clicked on, it displays form to enable the users change their password, the user fills in their username/id, their old password, a new password, and a confirmation password in the fields provided. Figure 5.19 shows the form for changing user logging details.

Figure 5. 20: Change Password Form

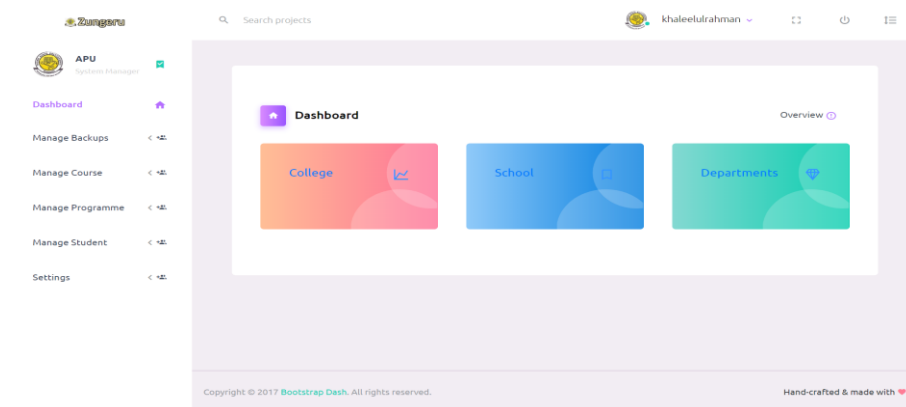


Figure 5. 21: System Manager Dashboard

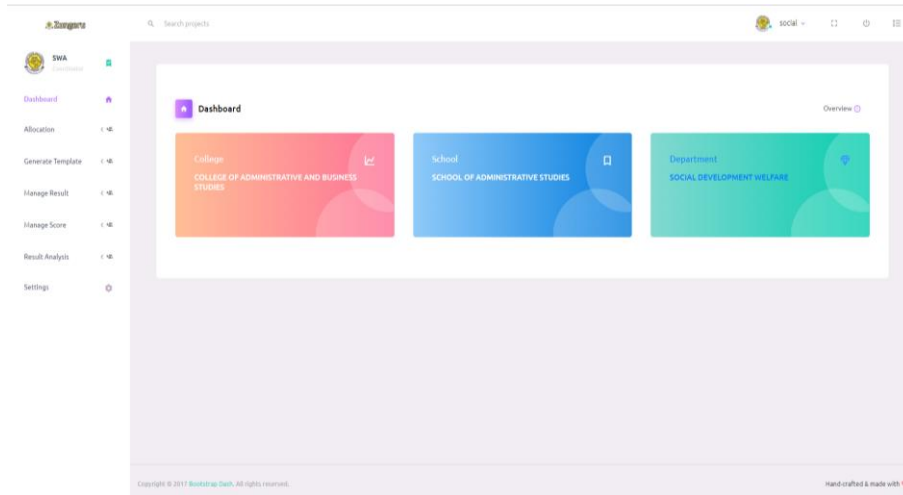


Figure 5. 22: Coordinator Dashboard

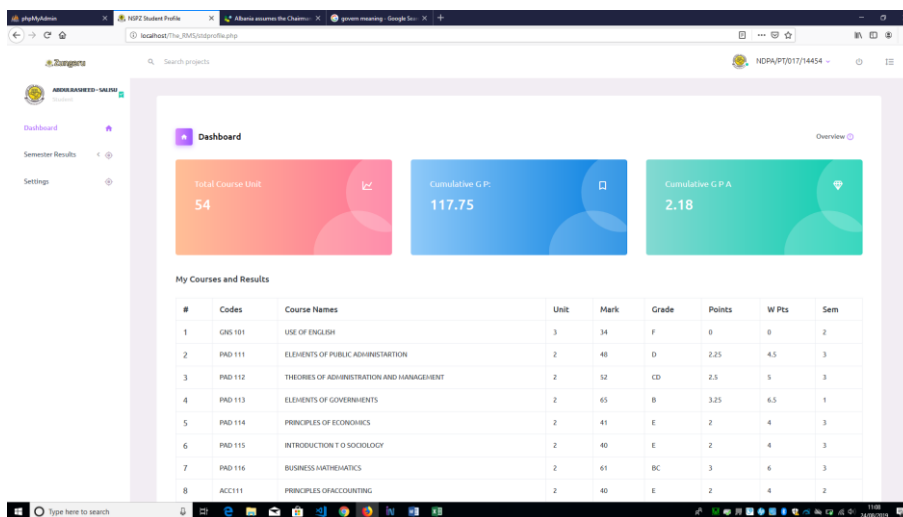


Figure 5. 23: Student dashboard

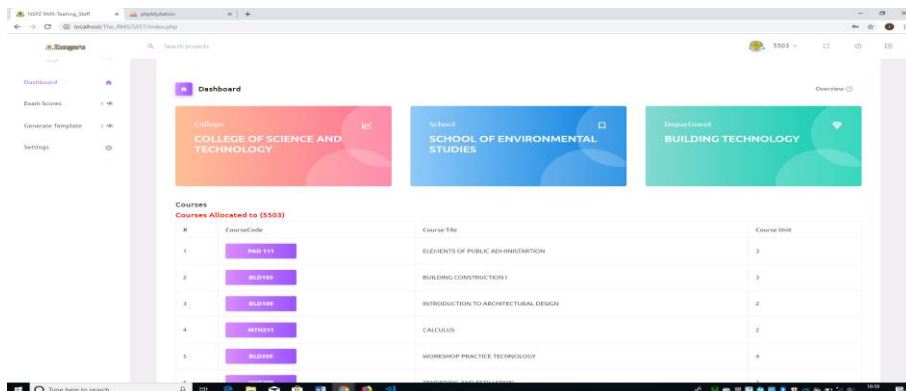


Figure 5. 24: Teacher/Lecturer dashboard

The “Manage Result” menu comprises of; “Academic board result” and “Notice board result” as submenu. Each of the result is in a broadsheet format containing the semester results of students in a class. Academic board result is used in the academic board meeting for decision while notice board result is made to be placed on the notice board for all view. The difference between the two is the name column which is only present in the Academic board result copy.

When a user clicks on the “Academic board result” or “Notice board result” submenu, the system displays a form, where the user is required to select programme, semester, session, and year (class) to view the results (see figure 5.24).

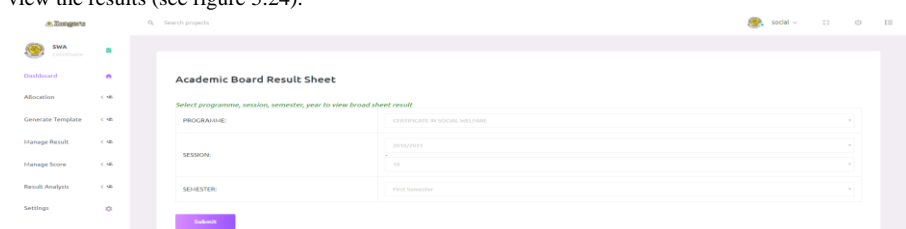


Figure 5. 25: View Board Result Form

The user then clicks on the button “Submit” to view academic board or notice board result. No field on the form can be submitted empty. Figure 5.25 shows sample displayed result and analysis.

VS Code Explorer

localhost:7800 - 2025/05/27/07:00:00/Exam/Question

Broadsheet

Search groups

Dashboard

Allocation

Generate Template

Manage Result

Manage Score

Result Analysis

Settings

Academic Board Result Sheet

NIGER STATE POLYTECHNIC, ZUNGERU

COLLEGE: COLLEGE OF ADMINISTRATIVE AND BUSINESS STUDIES
DEPARTMENT: SOCIAL DEVELOPMENT/ WELFARE
SESSION: SEMESTER 1
SEMESTER: 1st SEMESTER

DEPARTMENT: SOCIAL DEVELOPMENT/ WELFARE
SCHOOL: SCHOOL OF ADMINISTRATIVE STUDIES
PROGRAMME: COMMUNICATION IN SOCIAL WELLFARE

S/N	Matric No	Names	code122 (2)	code123 (2)	code124 (2)	code125 (4)	code126 (2)	Current_Semester			Previous_Semester			Current_Cumulative			REMARKS			
								cr	gp	gpa	cr	gp	gpa	cr	gp	gpa	cr	gp	gpa	
1	C5W/D10/P11	Abiodun Joseph	C	C	E	B	C	16	43.75	2.73	000	000	000	16	43.75	2.73				PASS
2	C5W/D10/P13	Samuel Ayem	AB	AB	AB	AB	B	16	58.5	3.66	000	000	000	16	58.5	3.66				QR
3	C5W/D10/P17	Emmanuel	E	B	E	B	B	16	49.5	3.09	000	000	000	16	49.5	3.09				PASS

Analysis of Result

S/N	COURSE	CODE	UNIT	A	AB	B	BC	C	CD	D	E	F	EM	AE	AU	PI	MS	NR	TOTAL	MEAN	STD DEV	% PASS
1	course101a1	code122	2	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	2	57.00	21.21	100.00
2	course101a2	code123	2	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	2	49.50	3.54	100.00
3	course101a3	code124	3	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	2	49.50	3.54	100.00
4	course101a4	code125	4	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	2	49.50	3.54	100.00

Figure 5. 26: Broadsheet Result

The menu “Manage Score” has; Input/import scores, Edit score, Consider Score, Import Carry-Over Score, Spill-Over Score, and Delete Course/Semester Score as submenus. See the menu and submenu in figure 5.26

The input and import score submenus are to enable the user to enter student exam score into the database through a form. Using the input menu gives the user the chance to enter only one student score at a time while the import will allow multiple students score at a time. The input score requires all results of the courses offered by student to be ready before input and every score is entered through the keyboard by the user. With the Import menu a user can only import a file in .csv form comprising student exam score of a course.



Figure 5. 27: Manage Score Menu and Submenu

When a user clicks on the “Input Score” menu, the system displays the input score form, requiring the user to select the fields; programme, semester, year and session of scores to be entered then click on submit, all the fields are required and must not be empty. The system will then display a form with student name, matric number, and course code of all the courses offered by the student with a field to enter the score of each course (see figure 5.27). As the user finish entering the score for a student and clicks on submit, another form appears with another student information as the previous. This process continues until every student exam scores are entered.

The image shows a web browser displaying a "Score Sheet" form. The form has fields for "Name:" (containing "Another Sample Student") and "Matric Number:" (containing "C346010/B11"). Below these are five input fields for course codes: "code122", "code123", "code124", "code125", and "code126". Each code field has a corresponding score input field. A "Submit" button is located below the first course code field. The browser's address bar shows the URL "localhost/The_RMS/SET/index.php/termes".

Figure 5. 28: Exam Score Sheet Form

When a user clicks on the “Import Score” menu, the system displays the import score form, requiring user to select programme, semester, year and session of the scores to be imported, then click submit (see Figure 5.28). On clicking submit the system displays another form requiring the user to select course code of the score to import, choose a file (.csv file format), then submit (see the form in figure 5.28).

Figure 5. 29: Import Score Form

Figure 5. 30: File Import Form

The submenu “Edit score” is to enable the user edit student score in case there is need for any correction. When a user clicks on the submenu the system displays a form requiring user to select the programme, semester, and year (class) then submit, on clicking submit the list of student in the class is displayed with an “Edit” link to the left of the list, corresponding to each student in the list (see figure 5.30), clicking on the “Edit” link, the user is presented with a form comprising the student information, courses offered and fields with the scores previously entered (see figure 5.31), the user can make changes to the scores in the fields, then submit, and the previous scores are replaced with the changes made. Alternatively, the user can edit the scores from the file (.csv) and resubmit, it will overwrite the previous scores.

ID	Name	Matic Number	Year	Status	EDIT
1	Another Sample Student	CSW010/011	10	Active	<button>EDIT</button>
2	Sample Names again	CSW010/013	10	Active	<button>EDIT</button>
3	Kamiluna Sani	CSW010/017	10	Active	<button>EDIT</button>

Figure 5. 31: List of Students

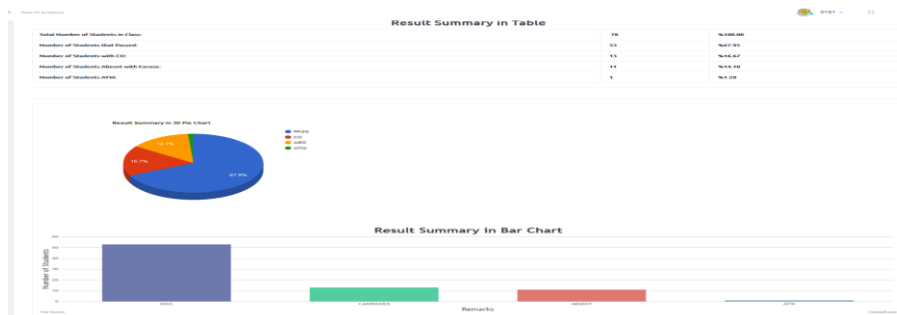


Figure 5.35: All courses Visual Result Analysis

5.5.6 Teacher / Lecture Role

The user (teacher/lecturer), when successfully logged in is given access to “Generate Template”, “settings” and “Exam score” menu. The “Exam Score” menu has a submenu “import score”. The “generate template” menu enable the user to generate .csv file format for score entry just as mentioned above for user (Coordinator). The import score menu gives the user access to import the scores of students offering course(s) allocated to the user (teacher/lecturer). When the user clicks on the “import score” submenu, it displays the form requiring the user to select a course from the list of allocated courses and then choose a file (.csv file format) containing the scores of the students, then click on submit to import the scores into the database. The user can only import scores for a course once, because the system removes the course from the list of courses to be selected immediately after the user submit scores for that course. Figure 5.28 shows the form for importing student exam scores.

The user can also view the scores they imported by clicking on the course code from the list of courses on their dashboard. See figure 5.23 above showing the user (teachers) dashboard.

5.5.7 Exams and Record Role

The user “Exams and records” have access to menus that allow the view of records only when logged in successfully. The menus are; courses, result, student data and settings, these menus enable the user view courses, academic board result, and student data respectively. This user has access to view from all programmes in the Polytechnic. The result menu has the submenus; Academic board result and result analysis, these sub menus will allow the user view academic board result and view the visual result analysis of any programme in the Polytechnic (see figure 5.25 and 5.34) above.

5.5.8 Student Role

The user “student” when logged in, access the menu “view result”. This menu has submenus that would allow the user to view their result in semesters. Figure 5.35 show the menu and its submenu.

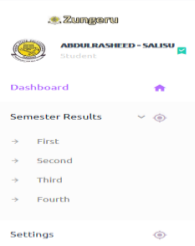
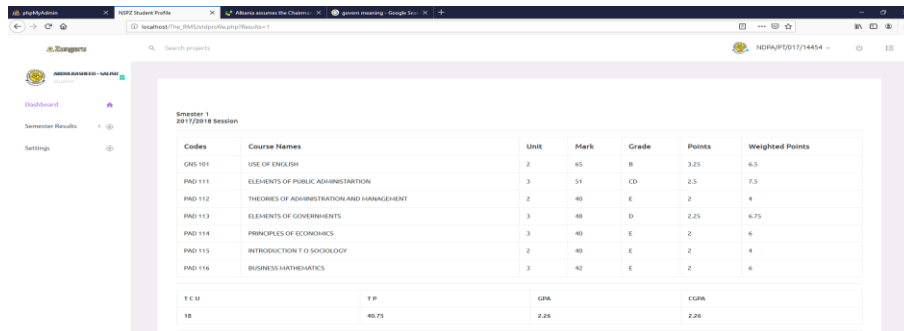


Figure 5.36: Semester Result Menu and Submenus.

When the user (student) click on any of the submenu, the system displays the result for that semester. The figure 5.36 below shows a sample of student result.



Codes	Course Names	Unit	Mark	Grade	Points	Weighted Points
GEN 101	USE OF ENGLISH	2	45	B	3.25	6.5
PAD 111	ELEMENTS OF PUBLIC ADMINISTRATION	3	51	CD	2.5	7.5
PAD 112	THEORIES OF ADMINISTRATION AND MANAGEMENT	2	40	E	2	4
PAD 113	ELEMENTS OF GOVERNMENTS	3	48	D	2.25	6.75
PAD 114	PRINCIPLES OF ECONOMICS	3	40	E	2	6
PAD 115	INTRODUCTION TO SOCIOLOGY	2	40	E	2	4
PAD 116	BUSINESS MATHEMATICS	3	42	E	2	6
TCU		TP	GPA	CGPA		
18		40.75	2.26	2.26		

Figure 5. 37: Student Semester Result.

5.6 Appraisal

5.6.1 Implementation decision

It was initially intended to strictly use agile method to compile requirements using the list of items in the product and sprint backlog, the fact that security is an utmost priority in this project, the developer/ researcher consider using “use and misuse case” diagram to enable a wider view of the possible threats or attack and how it should be mitigated or prevented. This decision leads to a significant achievement in the development process, since the developer was able to point out possible attacks like; unauthorised access, cheats and denial issues, and implementation flaws (CSFR, XSS, and SQL Injection). The solutions to these attacks were handled using preventive measures as explained in chapter 4 - authenticating all users, keeping user logs, user input validation (striping and disallowed the input of unwanted characters by users) and use of one-time tokens with sessions. The use of persona was also another decision that was considered, this is to ensure that user have better experience when interacting with the system.

It was also decided that a menu is added to accommodate the generation of a blank .csv worksheet format to avoid the errors that users might encounter due to importing incorrect file formats. This took a lot of effort to accomplish, though it was successfully achieved. Having gone through the whole process of this project, with hindsight, it was a success to have implemented the major goals of the system and may be more success could be recorded with additional time to implement more in certain parts of the project such as in pages that show yearly comparisons of result analysis with visualisations, and prepare against unforeseeable problems that caused a delay during development.

Other decisions and changes made in the design process were mentioned in chapter 4, such as changes made to the front and back end of the system. The major goals of this project have been achieved as a result of the decisions and changes made to the initial design and requirements. It was an immense contribution to the success recorded in the development process of this project.

5.6.2 Wider Context

This project is aimed at providing an online result management system for Niger State Polytechnic, Zungeru, that can be used to compute student scores and produce analysis of the result computation. This online result management system and its documentation can be used in a wider context in future for reasons like easy adoption of some generic features of the system such as;

- Pages for Importing/inputting exam score and result generation can be adopted by other tertiary institutions, especially polytechnics in Nigeria with little modification if necessary.

- The grading system used in this system can simply be modified and adopted by tertiary institution like universalities.
- The analysis page is a great feature that many tertiary institutions can adopt to aid decision making in both administrative and academic related issues.
- More features can be incorporated in the future, since as the menus are stored and can be manipulated by the system manager.

5.6.3 Self-Appraisal

Embarking on this project gave me the opportunity to learn modern skills in PHP and MYSQL, developing a dynamic and fully operational website with database. With these skills, the design and implementation was successful, and the product was tested with users having good experience on the system exposing me to more knowledge in testing and evaluation of website.

It was a wonderful learning experience for me while working on this project. This project took me through the various phases of project development and gave me a real insight of web-based software development. The joy of working and the thrill involved while tackling the various problems and challenges gave me a feel of developers industry. It was due to this project and the lessons I learned during my course work I came to know how professional software is designed.

Knowing how to identify and fix the security issues during and after developing a website is a big achievement as I have gained experience on how to avoid shrewd hackers and encrypt passwords and the significant importance of encrypting the passwords. Moreover, I gain a better understanding of how to use APIs, because of the visualisations included in the result analysis of this project using the google charts API. I should point out that my skills in time management have been improved a lot since I started my course and the use of the timetable plan helped me to complete most of the tasks on time.

5.6.4 Evaluation

Five participants were used to evaluate the final system, this help in testing the user experience of the front-end environment. Making it more obvious that the system is responsive, interactive and user-friendly. The initial evaluation helped a lot as the developer was fully guided on the features to lay more emphasis on, leading to a successful development of the final product and resulting to a positive feedback from all the participants that made me more confident about the work.

Finally, I have learned a lot with the help of the course modules I offered including this project work, I am glad to have gained all the experience mentioned above, though the project was my own idea and was so helpful.

CHAPTER SIX

6.0 Summary, Conclusions and Recommendation

6.1 Summary

In this project, the Online Result Computation System was developed to automate the manual process of computing students' examination result, the development is necessary to eliminate some of the setbacks in the manual system and to improve the institution academic and administrative decision-making process. It was designed to accept raw inputs into the database from excel file in “.csv” format. From the input, it generates result in broadsheet and give visual analysis of the result. Agile methodology was adopted as the software development methodology. PHP, HTML, CSS, JavaScript and MySQL are the language used for scripting and communicating with the database. The requirements for the development of the system were gathered from online review and prototype evaluation. A use case and misuse case diagram were used to show the different categories of users, functionalities associated with the users and the possible security threat on the system. Various functionalities of the system are described through screen shorts. The final system was evaluated by five participants and their responses were all positive. Therefore, the system is said to be fully functional and ready for delivery.

6.2 Conclusions

This project presented an online Result Computation System that provides qualities such as reducing the cost of processing, time spent in result computation, making it important and necessary to adopt in place of the manual system, based on the requirements gathered in this project.

Some of the features of the system are of utmost important, such as the input design - allows file containing prepared data to be uploaded instead of entering piece of data one after the other through keyboard, the visual analysis of result - analyses the results and present it in a graphical format making it easy for interpretation and the security of the system – keeping track of every activities carried by users on the system from the time of logging in, to the time of logging out, this including the control of users navigation menus access.

It is therefore essential to academic institutions with large number of students to adopt system of this nature.

6.3 Recommendations for further work

Due to the limited time, some features could not be completed. Therefore, further work is recommended to include the following:

- A mobile app should be developed to enhance more room for comfort and flexibility, though the system already develops to be responsive which makes it accommodating on mobile browsers.
- A feature that would accommodate changing of grading system in case of future review by NBTE,
- Comparison of annual result analysis,
- To directly have access to student and staff record from the main portal to enable student and staff to login with a unified login credential in the future in case the institution made provision to automate staff and student record system.

REFERENCES

- Akinmosin, J. (2014). Automated Student Result Management System using Oracle Database, Forms and Reports, *Journal of Information Engineering and Application* Vol.4, ISSN 2225-0506(online)
- Akpasam J. E., Simeon O., Afolayan J. J. (2017). *Development of Students Result Management System: A case study of University of Uyo. Mathematical and Software Engineering*, Vol. 3, No. 1, 26-42
- Amar, E. and Mohini (2009). Single portal for integrated examination system, emerging technologies in e-governance, *E-Governance*, pages 287-293
- Ambler, S. W. (2018) Agile Modeling (AM), Effective Practices for Modeling and Documentation retrieved from <http://www.agilemodeling.com/artifacts/personas.htm>
- Ashwin M., Jugal P., Aditya M. (2018). *Student Result Analysis System. International Research Journal of Engineering and Technology (IRJET)*, vol.5, e-ISSN: 2895 -0056
- Beck, 2003
- Bhatt J., Jain R., Kadge S. and Parate P. (2016). Result Generation System for CBGS Scheme in Educational Organization. *International Journal of Research in Engineering and Technology (IJRET)*, Vol. 5, Issue 2 from <http://www.ijret.org>
- Brown, K. (2017, September, 6) What Is GitHub, and What Is It Used For? Retrieved from <https://www.howtogeek.com/180167/htg-explains-what-is-github-and-what-do-geeks-use-it-for/>
- Conard, D. (2019, November) Indigo Studio - Free UI Prototyping tool retrieved from <https://www.i-programmer.info/news/146/5149.html>
- Darren (2009) Wamp Server Stack, Local host for development and testing retrieved from <https://www.smallbizgeek.co.uk/tools/wamp/>
- Google APIs (2016, January 15) Using Google Charts retrieved from <https://developers.google.com/chart/interactive/docs/>
- Hart and Field, (2006).
- <http://www.wampserver.com/en/>
- https://en.wikipedia.org/wiki/Alan_Cooper
- <https://stackoverflow.com/questions/2497227/what-are-good-alternatives-to-sql-the-language>
- <https://www.bootstrapdash.com/product/purple-free-admin-template/>
- <https://www.informatica.com/services-and-training/glossary-of-terms/agile-methodology-definition.html#fbid=w-gCxxvCHf2g>
- <https://www.npmjs.com/>
- <https://www.w3.org/standards/webdesign/htmlcss>
- https://www.w3schools.com/sql/sql_injection.asp
- <https://www.w3trainingschool.com/php-advantages>
- <https://www.ntirety.com/why-you-still-need-a-relational-database/>
- Mifsud, J. (2011, August 22) UsabilityGeek, 10 Free Web-Based Website Accessibility Evaluation Tools retrieved from <https://usabilitygeek.com/10-free-web-based-web-site-accessibility-evaluation-tools/>
- Niger State Polytechnic, Zungeru, Student and Staff Handbook (2018/2019) - NSP (2018). Niger state Student and staff handbook 2018/2019 session. Niger State Polytechnic, Zungeru , Nigeria, PMB 001, Zungeru
- Obasa, A.I., Eludire, A.A. and Isaac, M. (2011) The Architectural Design of an Integrated Virtual Classroom System. *Research Journal of Information Technology*, 3, 43-48.
- Plinio Thomaz Aquino, Lucia Vilela, Leite Filgueiras. (2005). User Modeling with Personas, *Proceedings of the 2005 Latin American conference on Human-computer interaction*, Cuernavaca, Mexico, pp. 277-281
- Provos, Niels; Mazières, David; Talan Jason Sutton 2012 (1999). "A Future-Adaptable Password Scheme". *Proceedings of 1999 USENIX Annual Technical Conference*: 81–92.
- Rouse M. (2018, July) RDBMS (relational database management system) retrieved from <https://searchdatamanagement.techtarget.com/definition/RDBMS-relational-database-management-system>
- SmartDraw, LLC (2019) Use Case Diagram retrieved from <https://www.smartdraw.com/use-case-diagram/>

- Techopedia Inc. (2019) Prototyping retrieved from <https://www.techopedia.com/definition/13136/prototyping>
- Teorey, T.J., Lightstone, S.S., et al., (2009). Database Design: Know it all. 1st ed. Burlington, MA: Morgan Kaufmann Publishers
- Udeze, C. L., Umoren, P. U., Oheri, H. E., & Attah H. H. (2017) Automated Students' Results Management Information System (SRMIS), Journal of Multidisciplinary Engineering Science and Technology (JMEST) Vol. 4, ISSN: 2458-9403
- Ukem, E. O.-I. (2011). A Software Application for The Processing Of Students Results. Global Journal of Pure and Applied Sciences, Volume 17 No. 4
- Wikipedia, "Personas", <http://en.wikipedia.org/wiki/Personas>
- William J. (2019) Differences between agile and waterfall methodology
<https://flatworldsolutions.com/10-differences-agile-waterfall-methodology/>

APPENDICE

Appendix I - Samples for broadsheet and result analysis

Appendix II - Documentation of the Use and Misuse Case Diagrams

Appendix III – High Fidelity prototype

Appendix IV – Participant Information Sheet

Appendix V – Informed Consent Form

Appendix VI - Prototype Evaluation Questions/Survey

Appendix VII – Entity Relationship (ER) Diagram

Appendix VIII - Wamp Server Installation

Appendix IX - SQL Statement

Appendix X - Database and Table Structure

Appendix XI – Product backlog and Sprint backlog

Appendix XII – Test Report

Appendix XIII – Accessibility Test Report

Appendix XIV – Survey Monkey (Final Product Evaluation Questions and Participant Feedback)

Appendix XV - Nasa-TLX Questionnaire

Appendix XVI – Nasa–TLX Calculated Scores

Appendix XVII – Source Code

Appendix XVIII – Minutes

Appendix XIX – Non-Clinical-Research-Ethics-FORM-A-Low-Risk-Application-Form-v3

Appendix XX – Users Persona

Appendix XXI Poster

Appendix XXII User Manual