An Implementation of the Past Question Paper Archive System Model (A case study of Baze University)

An Undergraduate Thesis Presented to

The Department of Computing and Applied Sciences Baze University, Abuja.



In Partial Fulfilment of the Requirements for the Degree of B.Sc. in Computer Science

by

**Mamman,** Bilkisu Tahir BU/17C/IT/2827

September, 2020

DECLARATION

This is to certify that this Thesis entitled “An implementation of Past Question Paper Archive System Model (A case study of Baze University)”, which is submitted by Bilkisu Tahir Mamman in partial fulfilment of the requirement for the award of degree for B.Sc. in Information Technology to the Department of Computer Science, Baze University Abuja, Nigeria, comprises of only my original work and due acknowledgement has been made in the text to all other materials used.

Date: Day August 2020 Name of Student: Bilkisu Tahir Mamman

**APPROVED BY** ………………….………

#### Head

Department of Computer Science

CERTIFICATION

This is to certify that this Thesis entitled “An implementation of Past Question Paper Archive System Model (A case study of Baze University)”, which is submitted by Bilkisu Tahir Mamman in partial fulfilment of the requirement for the award of degree for B.Sc. in Computer Science to the Department of Computer Science, Baze University Abuja, Nigeria is a record of the candidate’s own work carried out by the candidate under my/our supervision. The matter embodied in this thesis is original and has not been submitted for the award of any other degree.

Date: Supervisor:

APPROVAL

This is to certify that the research work, Dental Management System and the subsequent preparation by [Your Name] with [Student ID] has been approved by the Department of Computer Science, Faculty of Computing and Applied Science, Baze University, Abuja, Nigeria.

By

Dr. Rislan Kanya Date/Sign

1st Supervisor

Dr. Amit Mishra Date/Sign

2nd Supervisor

Dr. C. V. Uppin Date/Sign

Head of Department

Prof. Hammawa Mohammed Baba

Dean, Faculty of Computing and Applied Science

Date/Sign

Prof. Ahmed Baita Garko Date/Sign

External Examiner

DEDICATION

This thesis is dedicated to everyone who played a role in furthering my education. First and foremost, my parents, who have given me this opportunity and without whom none of this would be possible. To my Mom for raising me to believe that I can do anything I put my mind to and my Dad who has guided me every step of the way.

To my family, brothers, sisters and friends, who have encouraged and supported me throughout the journey.

To my teachers, who have guided, taught and advised me in all that I know now. I will forever be grateful for all that each and everyone one of you have done for me.

To my mentor at Baze University, I am thankful for all the help that Dr. Rislan has given me. Thank you for giving me an opportunity to grow, learn and become a more skilled individual in my field.

# ABSTRACT

The proposed thesis is designed to be an online Baze University past question archiving system. The use of technologies has become critically important in the era of the Coronavirus pandemic which has required off-campus distance learning by students of Baze university. With no access to staff and on-campus facilities, the availability of easily accessible online study material has become critical. This project aims to provide students with the necessary materials needed to aid with learning and studying efficiently. It seeks to aid in distance learning by providing online access to past questions which have previously only been accessible as hard paper copies at the Baze university libraries. This web application has been developed using the Python programming language, its supporting flask framework and SQLite for the database. The application was successfully tested and produced the desired results.

# Table of Contents

[ABSTRACT vi](#_TOC_250069)

[List of Tables x](#_TOC_250068)

[List of Figures xi](#_TOC_250067)

[Chapter 1: Introduction 1](#_TOC_250066)

* 1. [Overview 1](#_TOC_250065)
  2. [Background and Motivation 2](#_TOC_250064)
  3. [Statement of the Problem 2](#_TOC_250063)
  4. [Aim and Objectives 3](#_TOC_250062)
  5. [Significance of the Project 4](#_TOC_250061)
  6. [Project Risk Assessment 5](#_TOC_250060)
  7. [Scope/Project Organization 5](#_TOC_250059)

[Chapter 2: Literature Review 6](#_TOC_250058)

* 1. [Introduction 6](#_TOC_250057)
  2. [Historical Overview 6](#_TOC_250056)
  3. [Related Works 7](#_TOC_250055)
     1. [Proposed Improvements 8](#_TOC_250054)
  4. [Summary 9](#_TOC_250053)

[Chapter 3: Requirements Analysis 10](#_TOC_250052)

[3.1 Overview 10](#_TOC_250051)

[3.2.1 Proposed Methodology 10](#_TOC_250050)

* + 1. [Waterfall model 10](#_TOC_250049)
    2. [Agile development methodology 12](#_TOC_250048)
    3. [Spiral Development Methodology 14](#_TOC_250047)
  1. [Approach to Chosen Methodology/Methods 16](#_TOC_250046)
  2. [Tools and Techniques 17](#_TOC_250045)
  3. [Ethical Consideration 18](#_TOC_250044)
  4. [Requirement Analysis 18](#_TOC_250043)
  5. [Requirements Specifications 19](#_TOC_250042)
     1. [Functional Requirement Specifications 19](#_TOC_250041)
     2. [Non-Functional Requirement Specifications 21](#_TOC_250040)
  6. [System Design 22](#_TOC_250039)
     1. [Application Architecture 22](#_TOC_250038)
     2. [Use Case 23](#_TOC_250037)
     3. [Activity Diagram 24](#_TOC_250036)
     4. [Data Design Diagram 25](#_TOC_250035)
     5. [Entity Relationship Diagram 26](#_TOC_250034)
     6. [User Interface Design 26](#_TOC_250033)
  7. [Summary 31](#_TOC_250032)

[Chapter 4: Implementation and Testing 33](#_TOC_250031)

* 1. [Overview 33](#_TOC_250030)
  2. [Main Features 33](#_TOC_250029)
     1. [Role-Based Authorization and Authentication 33](#_TOC_250028)
     2. [Search engine for the past questions 34](#_TOC_250027)
     3. [Uploading the past questions 35](#_TOC_250026)
     4. [Viewing and/or downloading past questions 36](#_TOC_250025)
  3. [Implementation Problems and Overcoming them 37](#_TOC_250024)
  4. [Testing 39](#_TOC_250023)
     1. [Tests Plans (for Unit Testing, Integration Testing, and System Testing) 39](#_TOC_250022)

[Approvals 41](#_TOC_250021)

* + 1. [Test Suite (for Unit Testing, Integration Testing, and System Testing) 42](#_TOC_250020)
    2. Test Traceability Matrix (for Unit Testing, Integration Testing, and System Testing) 50
    3. Test Report Summary (for Unit Testing, Integration Testing, and System Testing) 51
    4. [Error Reports and Corrections 52](#_TOC_250019)
  1. [Use Guide 52](#_TOC_250018)
  2. [Summary 53](#_TOC_250017)

[Chapter 5: Discussion 53](#_TOC_250016)

* 1. [Overview 54](#_TOC_250015)
  2. [Objective Assessment 54](#_TOC_250014)
  3. [Limitations and Challenges 54](#_TOC_250013)
  4. [Future Enhancements 55](#_TOC_250012)
  5. [Recommendations 56](#_TOC_250011)
  6. [Summary 56](#_TOC_250010)

[REFERENCES 57](#_TOC_250009)

[APPENDICES 58](#_TOC_250008)

[Appendix B – Interview 58](#_TOC_250007)

[Baze IT department 58](#_TOC_250006)

[Baze students 58](#_TOC_250005)

[Appendix C - Source Codes 59](#_TOC_250004)

[Appendix D – Test Cases 65](#_TOC_250003)

[Appendix E – User Guide/Manual 67](#_TOC_250002)

[Students User Guide 67](#_TOC_250001)

[Admins User Guide 67](#_TOC_250000)

# List of Tables

Table 3.2 Non-functional requirements 21

Table 4.1 Test Suite Performed for Login 42

Table 4.2 Test Suite Performed for Logout 43

Table 4.3 Test Suite Performed for Search engine 44

Table 4.4 Test Suite Performed for database update 45

Table 4.5 Test Suite Performed for viewing/download 46

Table 4.6 Test Suite Performed for upload 47

Table 4.7 Test Suite Performed for confirmation alert 48

Table 4.8 Test Suite Performed for home page 49

Table 4.9 Test traceability matrix 50

Table 4.10 Test report summary 51

Table 4.11 Error reports and correction 52

Table 4.7 Test Suite Performed for confirmation alert 65

Table 4.8 Test Suite Performed for home page 66

# List of Figures

Figure 3.1 Waterfall model (Tutorials Point, 2020) 11

Figure 3.2 Agile model (Anurina, 2019) 13

Figure 3.3 Spiral model (papertyari, 2015) 15

Figure 3.8.1 Application architecture 22

Figure 3.8.2 Use Case Diagram 23

Figure 3.8.3 Activity Diagram 24

Figure 3.8.4 Data Design Diagram 25

Figure 3.8.5 Entity Relationship Diagram 26

Figure 3.8.6 Landing page 27

Figure 3.8.7 Login page 27

Figure 3.8.8 Home page 28

Figure 3.8.9 Upload page 29

Figure 3.8.9 Admin Registration page 30

Figure 3.8.10 Search result page 31

Figure 3.8.11 Viewing page 31

Figure 4.2.2.1 Search Engine 34

Figure 4.2.2.2 Search Engine Result 35

Figure 4.2.3.1 Upload page 36

Figure 4.2.3.3 Upload database view 36

Figure 4.2.4.1 Viewing and download page 37

# Chapter 1: Introduction

## Overview

In the past few decades, technology has become a major ally in the education sector. The role of online teaching and learning material availability in quality of learning has become universally acknowledged and supported by decades of research. The importance of this is made even more apparent in developing countries like Nigeria. (Read, 2016)

In spite of the availability of decades of research that shows the importance of ensuring learning materials are adequately provided to students at all levels of education, Universities are yet to take full advantage of the technology that is available in 2020.

Past exam question papers have always been a good resource for learning and specifically for exam preparations. Past question papers give students a general idea of what a certain course is all about, familiarizes them with the exam format and most importantly, it provides an idea of what their expected exam questions may look like. (Docsity, 2013)

This project walks through the importance of having online learning materials for students, specifically the past question papers. The practices used at the university and how this project aims to help improve the current system will be analysed. The implementation and testing process will also be discussed in detail to aid in determining whether or not the goal of the project has been reached.

## Background and Motivation

As a final year student at Baze University, I have observed through the years how much of a hassle the search for past question papers is for the students. Currently, some papers are stored at the school library which is where they can be accessed. The problem with the current system is how unorganized it is.

The process of looking for a paper involves searching, one after the other, through all the past papers ever taken to the library since the inception of Baze University. That is hundreds of past papers to sift through without the promise of finding what is being looked for because some papers are never stored there or are taken out and never returned by some students.

If not found in the library, the students turn to their lecturers hoping they are kind enough to share the papers. This is a long process for something that can be automated using the available technology to provide these resources at the click of a button.

As a computer science student, I am fully equipped with the resources and knowledge needed to help in improving the lives of students at Baze University by ensuring that they are able to use their time efficiently on the things that matter like learning and studying for exams.

## Statement of the Problem

Using the technology that is widely available to improve the lives of the students for the better is a goal that all Universities are trying to reach. Baze University should not be excluded from this.

Students are having to go to the library and search through hundreds of unorganized past question papers with the hopes of finding what they are looking for. This process takes hours out of their studying time without the promise of success because not all past question papers are stored in the library as expected or because they have been illegally taken out of the library and never returned by other students.

If not found in the library, they turn to their lecturers hoping they are kind enough to share the papers. Because of this long process, a lot of students are discouraged from taking on the daunting task that is looking for past questions at Baze University. This hinders the learning and studying for students, and undoubtedly affects their educational performance.

The second problem is that, due to the Corona Virus pandemic, Baze University is adopting a permanent online school system for those who wish to take part in distance learning. These students deserve to be equipped with all the resources that their on- campus counterpart have access to.

## Aim and Objectives

This project proposes a system “Past Question Paper Archive System”, which is a web application system that aims to aid in distance learning by providing online access to past questions which have previously only been accessible as hard paper copies at the Baze university libraries. It will serve as an archive for all past question papers ever written at the institution.

The objective of this project will be to provide all past questions at Baze University, on the student portal, to be downloaded or studied online by the students. The question papers will be organized and indexed to enable easy access and also allow the users to search and retrieve information based on keywords.

The application will also enable the admins (which can be lecturers or other university authorities) to upload new past questions to be accessed by the students. The system will be hosted on the university server as a part of the student and staff portal through which the resources can be accessed by the students.

The objective of the project is to create a fully functioning past question paper archive that can easily be accessed by the students. With that goal in mind, the scope of the project will lie between the following challenges:

* + 1. The ability to collect and upload the past question papers.
    2. The ability for the users to be able to search, open and download and resource.

## Significance of the Project

The successful implementation of this project will bring relief for the students and staff of Baze University by enabling them to upload and access all the past examination resources at the comfort of their computers and on the same platform of which they are already proficient.

There will no longer be a need to be physically present at the university to be able to fully access and take advantage of its resources. This is an importance aid for the newly formed Baze University online school. When the system is successfully implemented, using it will be as easy as searching for any past paper by its name or course code after which they can be downloaded or studying online.

## Project Risk Assessment

Table 1. 1: Risk Assessment Table

Risks Risk alleviation techniques

Corona virus pandemic may restrict resource gathering

Late delivery of software due to exams

Contact university and try to get as many papers as possible sent online.

Create a timeline for project completion and stick to it.

Loss of work due to computer failure

Backup the application after every update.

Malware and virus attack Ensure antivirus is up to date

University staff not cooperating Reach out to different school

authorities for cooperation

## Scope/Project Organization

The remaining chapter of this project include chapter 2 “Literature Review” which contains all a review off work relating to this project, chapter 3 contains details on the functional and non-functional requirements, tools and methodologies utilized for the project, chapter 4 contains details on the implementation and testing processes including

any problems encountered and chapter 5 which contains discussions on the concluded project and any final notes.

# Chapter 2: Literature Review

## Introduction

This chapter will discuss previous research that has been done relating to the use of technology to provide online resources for students and online question archiving. Section 2.2 contains history on the use of technology in making learning resources available. Section 2.3 will consider other projects and software similar to this and discuss some of the features that make them ideal and otherwise. Lastly, the discussed topics in the chapter will be summarized in section 2.4.

## Historical Overview

Though it may seem like online and distance learning had its beginning at the dawn of technological and internet development, distance learning actually began in England during the 19th century. Lecturers and students would share studying materials and assignments through mail without ever having to physically be in school.

Due to modern day technology, distance/online learning has become more advanced and accessible to the average person. Many educational institutions around the now provide online courses, resources, classes and degrees that have made the idea of online education popular.

A few significant advancements have shaped and pushed distance learning forward since the late 1800s. In 1873 the first official correspondence education program, called the “Society to Encourage Home Studies”, was established in Boston, Massachusetts by Ana Eliot Ticknor. The University of Queensland in Australia founded its Department of

Correspondence Studies in 1911, which also relied on Australia's postal system. The University of South Africa, today known as one of the world's open distance learning mega colleges, became a champion and innovator of distance learning when it reshaped its mission and focus in 1946. (Visual Academy, 2009)

The next major inventions to revolutionize distance education were the personal computer and the personal web. Long before online classes and resource provision was considered in Nigeria, the University of Phoenix was the first University to offer full online bachelor and masters programs in 1989. In 1996, entrepreneurs Glen Jones and Bernand Luskin launched Jones International

University, which became the first accredited and fully web-based university. Since these fully online institutions were created, distance learning has continued to make leaps that have been aided by modern technology.

In the recent years in Nigeria, more universities are starting to adopt online learning as a staple in their institutions. Open University is one of the well-known institutions that primarily offer online learning. More recently, in response to the Corona Virus pandemic, many more private institutions like Baze University and Nile University are striving to move in that direction. Baze University will soon be offering full time online and distance learning opportunities for students who are unable to be present on campus.

The growth of distance learning programs has many important side-effects on higher education. For example, the profile of a typical undergraduate student has changed significantly. More women and older students are enrolled in online classes than ever before. As technology improves and online programs become respected, education experts predict that the distance education will continue to expand and become more complex in the future.

## Related Works

A web application similar to this project was once created by a student at Baze University which organised some past question papers according to course code. Students could view

the papers on the app and download them. It was located at [http://pastque.com/.](http://pastque.com/) The website is no longer in operation and the domain is up for sale. Having had the opportunity to view it while it was still running, a few issues that were noted about the web application was the lack of proper organization to make it easy to search through the paper. User were having to click and open multiple links to search for any paper without the promise of eventually finding it when a simple search engine could have been utilized. The chaotic nature of the user interface made it less than pleasant to use. There was also no cooperation between the unknown creator and Baze University to give them much needed credibility.

Other web applications that are somewhat similar to this project are a series of websites that include [www.exampastquestions.com](http://www.exampastquestions.com/) and “samphina.com.ng” which are online past question archiving website for universities in Nigeria that specialize in post UTME exam question. These, however, do not serve the purpose of this Baze University exam past questions archiving system as they do not have any resources that could be useful for students who are already enrolled at the University. Rather, they are useful for those who are trying to gain admission into the institution. It is also clear that these websites barely contain any of the resources that they claim to have.

#### Proposed Improvements

On [http://pastque.com/,](http://pastque.com/) the first feature noticed by any user of a web application is the user interface and how pleasant it is to use. This can really make or break a website but this website was not very good to look at or use. It left the user to chaotically figure it out instead of being straight to the point. This project aims to provide a very simple and pleasant user interface that usable to even the most inexperienced computer users at Baze University. It is easy to assume that most people are able to find their way around software in 2020, this is not the case. Having worked in IT student services during an internship, I observed that there are much more student that aren’t proficient in the use of computers, especially the older students. It is the aim of this project to ensure that even those students can use the system effectively.

A feature that all the works that have been reviewed above lack is the ability for the user to simply search for keywords and easily find a resource. This cuts down on so much time wasted on searching through a whole website just to one past paper. This will be a central feature in this project for the ease of use that is brings to a system.

This project also aims to collaborate with Baze University in collecting original copied of past question papers so that the system is credible and trusted. It will also provide a much larger database of all the papers that none of the discussed related works were able to.

## Summary

Having researched and reviewed all works related to this project, good attempts have been made in the effort to execute this idea but none of the software has been able to nail it and fully incorporate the necessary features to make them great software systems.

The only other past question paper archiving system specifically for Baze University was severely underdeveloped in its prime and is no longer functioning. This has left a gap that needs to be filled, especially in the wake of the Corona Virus pandemic which has made the requirement for online resource at Baze University a necessity.

# Chapter 3: Requirements Analysis

## 3.1 Overview

In this chapter, the requirements analysis and design will be discussed including all data gathering tools used in designing the system. The Data has been intentionally examined and Requirement Gathering Techniques utilized in building the system, the space for improvements and updates have been recognized, planned and information systems to actualize them and furthermore build up an idea for the new system. A designed working information system was implemented and also the development of a prototype for the new system.

System design was initiated, system design is a technical blueprint that either describes the new system or describes how the current system will be altered. It ultimately answers the question “How will the system do what it must do to obtain a solution to a problem?” It also details system outputs, inputs and user interfaces; specifies software, databases, telecommunication, personnel and procedures; and show how these components are related.

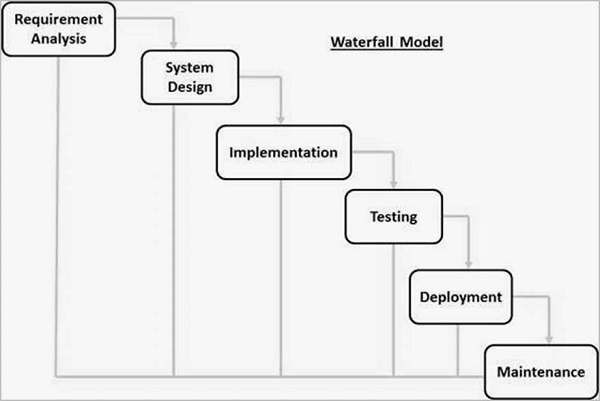
## Proposed Methodology

#### Waterfall model

The waterfall model which has been chosen for use in this project, also referred to as a linear-sequential life cycle model, is considered to be the most traditional approach to software development. It is a sequential approach in which each phase in the development process is carried out linearly. It is the earliest software development methodology ever used.

The waterfall was chosen due to the limited scope of this project. The features and requirements are clear, concise and somewhat small which means there are no expected changes required to be made in the future. This makes it the most practical option to be used in developing this project.

The following illustration is a representation of the various phases of a waterfall model.



*Figure 3.1 Waterfall model (Tutorials Point, 2020)*

#### Strengths of Waterfall model

This approach makes it easy to control and manage the software development process. Development moves from concept, through design, implementation, testing, installation, troubleshooting, and ends up at operation and maintenance. Each phase of development proceeds in strict order. It makes it easier to meet deadlines since each phase is completed

and processed before moving to the next phase. Other advantage for the waterfall model include:

* + - * + Clearly defined stages.
        + Ease of understanding and use.
        + Works well for smaller project were requirements are clear and well understood.

#### Weaknesses of Waterfall model

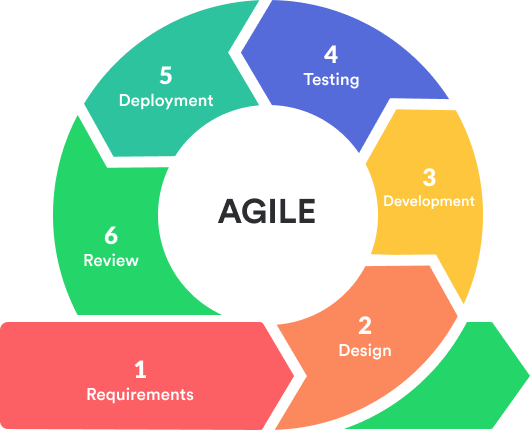
This method does not leave much room for changes during implementation of the software since the requirements, analysis and design phases are all completed in the initial phases of the model. Once the software is in the implementation or testing phase, it is very difficult to make changes to features that weren’t properly documented or additional features. Other disadvantages for the waterfall model include:

* + - * + Cannot accommodate changing requirements.
        + Poor model for big projects.
        + No space for changing requirements.
        + High amounts of risk.

#### Agile development methodology

Agile development methodology is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like planning, design, coding etc (Tutorials point, 2016). This methodology was developed as a response to growing dissatisfaction with the existing rigid and highly structured methodologies like the waterfall model. The

following is a simplified representation of the processes involved in the agile development method:



*Figure 3.2 Agile model (Anurina, 2019)*

#### Strengths of Agile

A main advantage is that developers are able to roll out a working software at every phase leading to a high level of costumer satisfaction. Agile methodology focuses on the users and interactions rather than the processes or phases of development. Other advantages of agile include:

* + - * + Suitable for projects with changing requirments.
        + Delivers early partial working software.
        + Easy to manage.
        + Provides flexibility for developers.

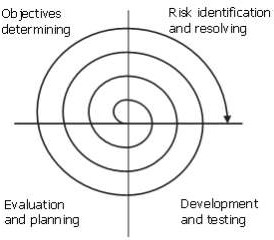
#### Weaknesses of Agile

Like all other software development methodologies, agile has its drawbacks. One of them is that its principles require close collaboration and extensive user involvement. Though it is an engaging and rewarding system, it demands a big commitment for the entirety of the project to ensure success. Clients must go through training to aid in product development. Any lack of client participation will impact software quality and success. It also reflects poorly on the developers. Other disadvantages of agile include:

* + - * + Lack of necessary documentation due to high individual dependencies.
        + Not capable of handling complex dependencies.
        + More risk of sustainability, maintainability and extensibility.
        + Project easily falls off track.

#### Spiral Development Methodology

The spiral development methodology is a combination of the control aspect of the waterfall model and the systematic nature of the iterative model. This model consists of four phases which are all carried out repeatedly and in a spiral manner from start to finish. The following is a representation of the four phases involved in a spiral model:



*Figure 3.3 Spiral model (papertyari, 2015)*

#### Strengths of Spiral model

A major advantage of the spiral model is that is allows new features to be added to the software at any phase. This ensures that there is no conflict with previous requirements and designs. This method is consistent with approaches that have multiple software builds and releases which allows making an orderly transition to a maintenance activity. Another positive aspect of this method is that the spiral model forces an early user involvement in the system development effort. Other advantages include:

* + - * + Delivers early partial working software.
        + Changing requirements can be accommodated.
        + Provides a more accurate representation of the requirements.
        + Development can be divided into smaller parts and parts with higher risks can be developed earlier which helps in better risk management.

#### Weaknesses of Spiral model

The successful completion of the project is very much dependent on Risk Analysis. Without very highly experienced expertise, it is going to be a failure to develop a project using this model. Other disadvantages include:

* + - * + It is expensive, which makes it unsuitable for smaller projects.
        + Management is more complex.
        + Spiral may go on indefinitely which is a problem for time constrained projects.
        + Large number of intermediate stages requires excessive documentation.

## 3.3 Approach to Chosen Methodology/Methods

In this section, the approach used for implementing the chosen waterfall methodology will be discussed. The waterfall methodology is broken down into multiple phases that include: requirements, design, implementation, verification and maintenance.

During the requirements phase, multiple interview sessions were carried out with the client of this project which is the Baze University IT department. Through meeting and working closely with them, we were able to gather information and fully understand the client’s perspective and their recommendations for the requirements of the system as well as all the features needed on the past question paper archiving system. Students at Baze University were also interviewed, as the users of the systems, to gather information on what they would like to be part of the system. All recommendations were then cross checked with the clients to verify the requirements.

In the design phase, with collaboration from the client at Baze University and the students, a comprehensive idea of the system design and user interface was derived in the form of a design template. This template was then used as a basic design template for the system on which changes were made until a completed template was derive. The derived template was then used as a basis for implementation phase.

Having gathered all the necessary functional and non-function requirements, as well as having an idea of what the user interface should look like, the next step would be the implementation phase. This is where all the coding was done. Having completed the previous phases fully, the program development and user interface design development came together with only a few hitches over a couple months.

Then came the testing phase, after the completion of the implementation, the project was properly with help from the clients and students at Baze university to ensure that it satisfies all the requirements. A sample size of some of the past question papers were also acquired to make the system realistic. During testing, all errors or bugs encountered were then isolated from the complete system after which the code for that particular section was revised and fixed.

Finally, during the maintenance phase, errors and bugs are monitored and fixed along the way along with updates to the system to ensure that it is always up to date. There will also be code optimization revision overtime to ensure that the system is operating at its optimum level as the amount of data and users increase.

## 3.4 Tools and Techniques

The projects features were developed using the python programming language with its supporting flask framework which is a micro framework developed to support python for the creation of web applications. This was chosen because it is one of the simplest ways of creating software using python. The front end of the web application was developed using HTML, CSS, JavaScript, JQuery and some bootstrap code.

The web application and database is intended to be hosted on the Baze University server which makes use of the windows server. The database was created using SQLite which is a popular choice as embedded database software for local/client storage in application software such as web browsers. The supporting SQLAlchemy and Whoosh Alchemy were also used for querying and searching purposes respectively.

## 3.5 Ethical Consideration

There are certain issues that have to be considered in every project with regards to how it affects others ethically. These issues include:

* + - * + Confidentiality: the confidentiality of the data provided the clients and users of a system have to be ensured. An understanding of trust between the clients and the programmer needs to be reached to ensure complete honesty.
        + Risk of harm: The security of all individuals involved in the project needs to be ensured. No harm may come to anyone due to their involvement in this project. Due to the nature of this project, it is extremely unlikely that any individual could get hurt in anyway.
        + Validity: the conclusions of the study must correlate to the questions posed and the outcome. Likewise, research about morals requests that the techniques utilized must relate explicitly to the research questions.

## 3.6 Requirement Analysis

The requirements of the past question paper archiving system were determined by analysing information gathered from the interviews with the clients and users at Baze University. From this, it was derived that the students complained about only being able to find past questions at the library and sometimes not at all. They complained about having to waste a lot of time just looking for the question papers with no promise of actually finding them. They also complained about the existing software (existed at the time of the interview) not serving the purpose that is was meant to. All of the user’s issues will be solved in this project by providing easy access to all the resources from the comfort of their computers. Instead of spending hours searching at the library, they can gain access within seconds.

The IT director expressed that this is a system he has wanted to implement for Baze University for a long time. The system will allow him as well as other Baze University staff to be able to access administrative features for the system.

## 3.7 Requirements Specifications

This section will provide a detailed description the systems as well as its functional and non-functional requirements. It will contain all necessary features required to ensure that the system is complete.

#### Functional Requirement Specifications

The functional specifications describe the features and behaviours that the system must have in accordance with the information derived while interviewing the clientele. It lists out the specific features that must be part of the system for it to be complete.

TABLE 3.1 FUNCTIONAL REQUIREMENTS

## Reg No. Description Type

FR-101 The system should be able to be viewed on any browser.

Configuration

FR-102 The system should allow admin/ lecturers to log into a separate portal

Functional

FR-103 The system should allow users to log in Functional

FR-104 The system should allow users to search for past question

Functional

FR-105 The system should allow users to open past question and study on the website

Functional

FR-106 The system should allow users to download past questions

Functional

FR-107 The system should allow admin/ lecturer’s to upload past questions.

FR-108 The system should allow users to manually scroll through the data base of all past questions

Functional

Functional

#### Non-Functional Requirement Specifications

The non-functional specifications describe the features necessary for the system to operate the way it was intended. These are the properties that make the app safe, reliable, efficient and portable to use. The plan for implementing non-functional requirements is detailed in the system architecture because they are significant to the architecture requirement.

*Table 3.2 Non-functional requirements*

## Req No. Description Type

NFR-101 The system should run on both iOS and windows computers.

Configuration

NFR-102 The system should be user friendly. Usability

NFR-103 The system should be written in python programing language.

Configuration

NFR-104 The system should be able to load a page in seconds. Performance

NFR-105 The system should make use of passwords to verify users. Security

NFR-106 User passwords should be encrypted before saving in database

Security

NFR-107 In case of an error, the system should be able to return to the state it was in before the error

Reliability

NFR-108 The user interface design should be consistent throughout the system.

Usability

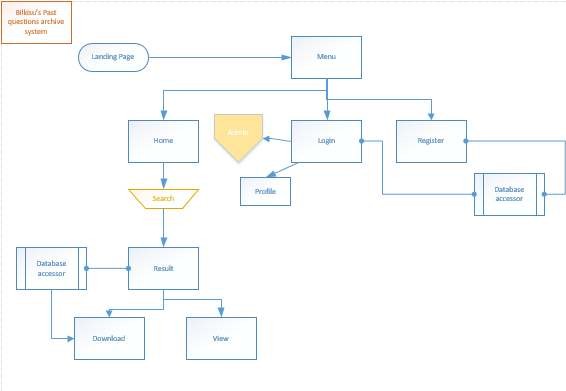
NFR-109 The system shall run on a windows server Configuration

## 3.8 System Design

This section contains a detailed description of the system, its functionality and the plan which they will be implemented. It includes the application architecture, database model, user, interface and UML diagrams.

#### Application Architecture

Below is the application architecture which shows a graphical representation of the application, views, and database. It also shows the connections and interactions that can be carried out in the system.



*Figure 3.8.1 Application architecture*

#### Use Case

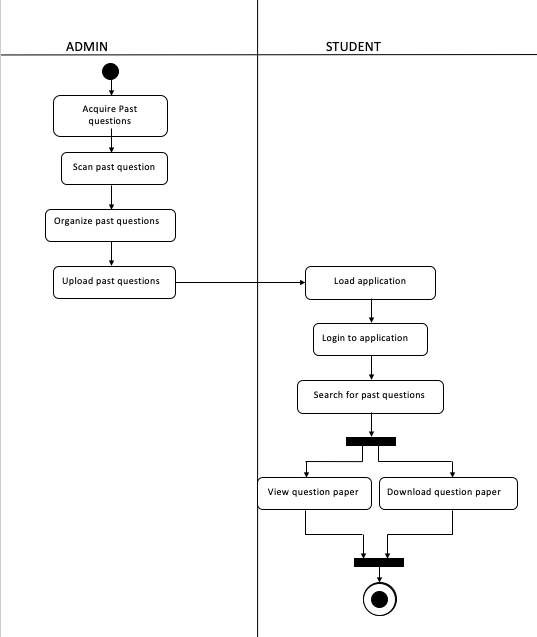
Below is the Use Case diagram which depicts the interactions between the users and the software system. It shows the functions of the system in chronological order and identifies the specific users that interact with them.



*Figure 3.8.2 Use Case Diagram*

#### Activity Diagram

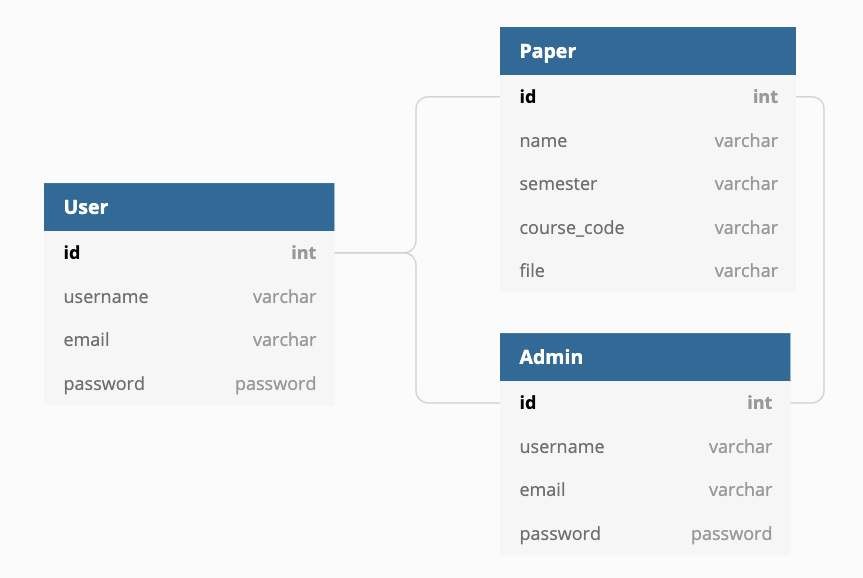
Below is an activity diagram that shows the sequential flow of activities within the system which also specifies the users who carry out such activities.



*Figure 3.8.3 Activity Diagram*

#### Data Design Diagram

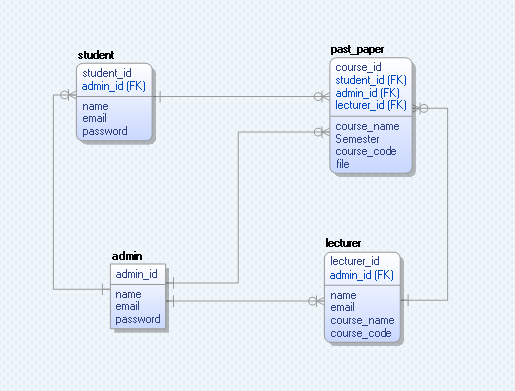
The diagram below shows a simple representation of the relationship between the entities within the system and their respective attributes.



*Figure 3.8.4 Data Design Diagram*

#### Entity Relationship Diagram

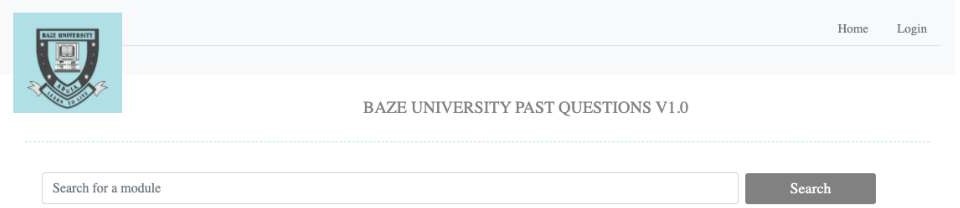
The below diagram shows a representation of database mode which includes the entities within the model and their respective attributes. It also shows the relationships and cardinalities between the tables in the diagram.



*Figure 3.8.5 Entity Relationship Diagram*

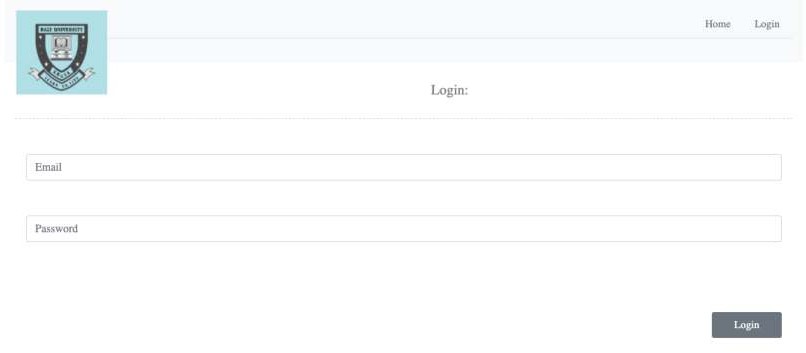
#### User Interface Design

The user interface design serves as the link through which the users interact with the system. Because the past question paper archiving system is created to be a part of the Baze University student portal, the clients placed importance on the system blending into the design of the already existing portal.

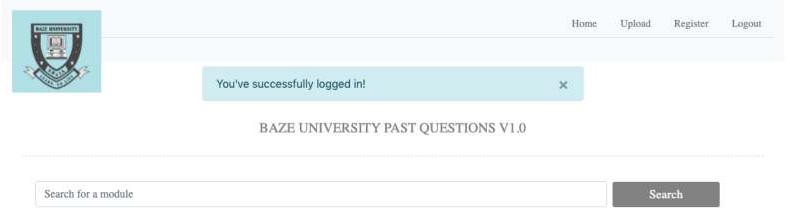


*Figure 3.8.6 Landing page*

As can be seen above, the system follows the minimal design colour pallet of the Baze University student portal as well also containing the Baze logo.

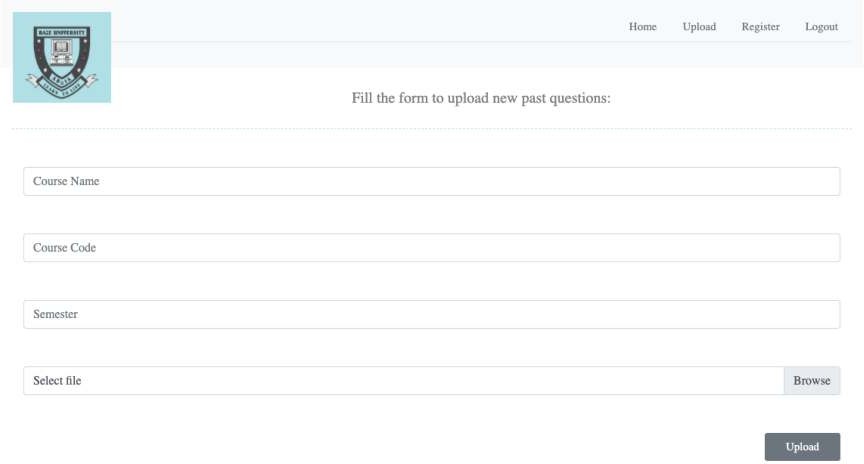


*Figure 3.8.7 Login page*



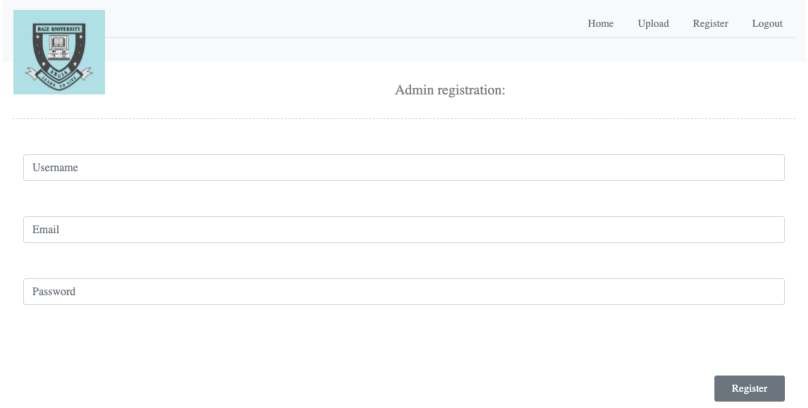
*Figure 3.8.8 Home page*

The above diagram is the home page after logging in as an admin. It is important to note the difference between the landing page and the home page. Only the admin has access to uploading new past question papers and registering new users.



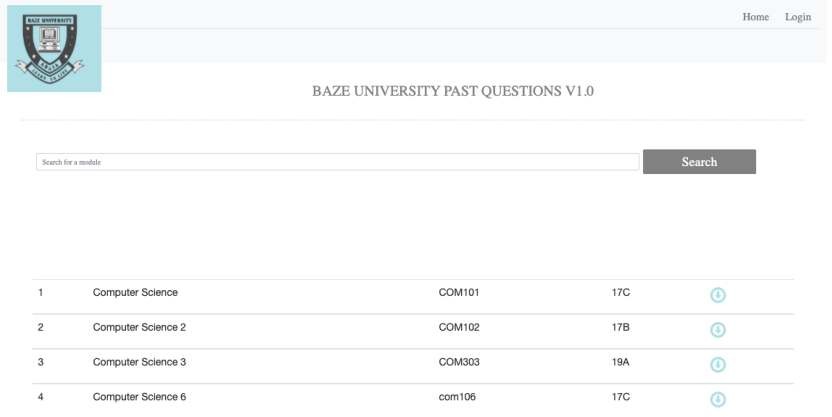
*Figure 3.8.9 Upload page*

The above diagram is the upload page through which all new past questions will be uploaded. It requires all the necessary details for proper documentation of the question papers.



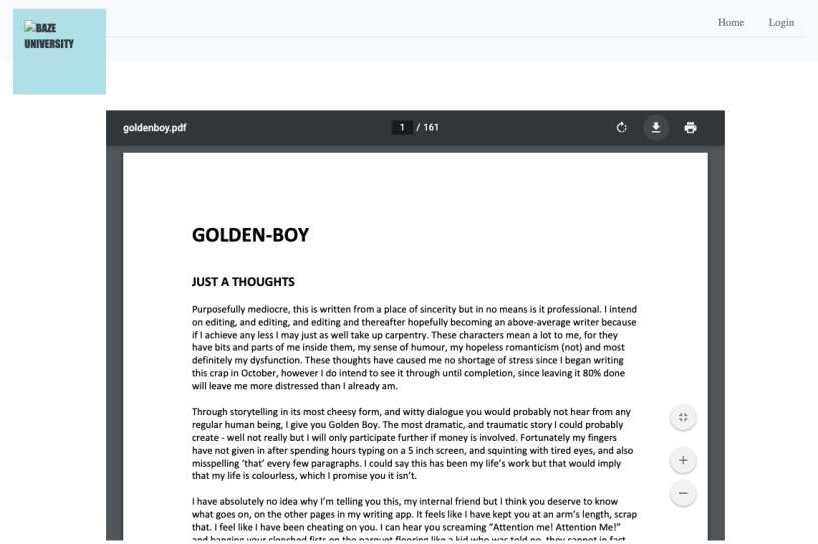
*Figure 3.8.9 Admin Registration page*

The image above is the admin registration page. In order to secure the system, new admins can only be created by other admins when logged in.



*Figure 3.8.10 Search result page*

The image above is the result of a “Computer Science” search. It shows the name of the courses, course code and semester as they’ve been inputted by the admin. It also has a button on the side of each past question paper through which they can be viewed or downloaded.



*Figure 3.8.11 Viewing page*

The image above is a what the view looks like when the open button on the previous page is clicked. The clicked document is open and can be left open and studied that way or the user may choose to download the document by clicking on the download button seen above.

## 3.9 Summary

In this chapter, the systems functional and non-functional requirements were discussed along with some of the development methodologies considered like the waterfall

methodology, agile methodology and the spiral methodology. The waterfall methodology was chosen specifically as the approach used in developing this software system as proven when the strengths and weaknesses of each methodology was discussed.

The tools and techniques used in the development of the system were also discussed as well as showing the user interface design that was created specifically to blend with the already existing Baze University student portal user interface.

Finally, this chapter also contains multiple graphical representations of the system, its users, and the interactions between them that were mostly created using UML (Unified Modelling Language).

# Chapter 4: Implementation and Testing

## Overview

This chapter will explore two main part of the project. The implementation part will be discussed which involves the source code and how each feature of the was created. Any errors gotten during the implementation of this system will also be explored along with the solutions used for those errors.

The second main part that will be discussed is the testing process. The system will be put through a series of test cycles until all errors are resolved and the systems functionalities works as intended. Close attention will also be paid to the performance of the system to ensure efficiency. And finally, a user guide will be specified to help in directing users through the system.

## Main Features

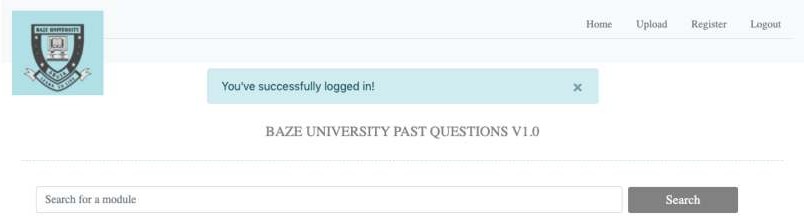
In this section, the main features and functionalities of the system will be discussed as well as an in-depth explanation of how they are implemented.

#### Role-Based Authorization and Authentication

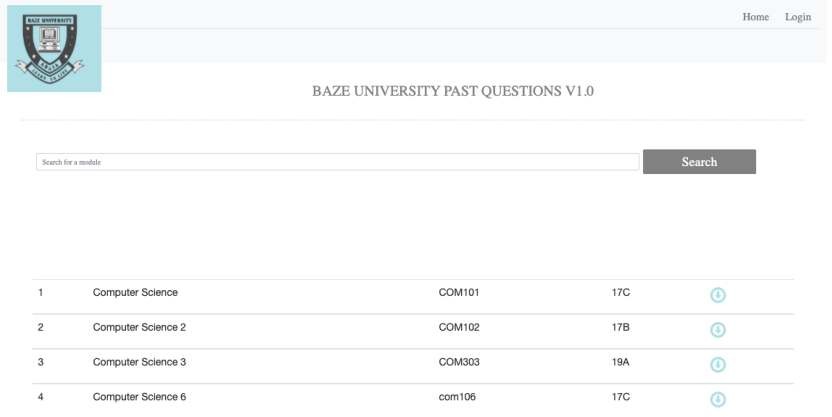
The past question archiving system has more than one user base that serve different purposes in the system. Therefore, it is necessary to reflect the users’ different roles within the system. The students use is mainly to be able to access the past question papers while logged in. While the admins use is mainly to be able to provide the documents to the students. This difference is reflected in the system and there are differences in the views that each user sees depending on their role within the system.

#### Search engine for the past questions

During the requirements gathering process, students of Baze University were interviewed. An issue that most of them had was the level of difficulty involved in getting past questions. Initially, the system was designed to have the documents sorted by course code or department such that the user will have to search through each sub-link to find them. However, that seemed like it would only cut down on the efficiency and increase the amount of time spent searching. This led to the decision to create a search engine such that users can simply input a key-word like the course name, course code or semester for their specific needs. This decision gives the system a more organized and refine feel as well as increasing efficiency and ease of use. Below is a view of the search engine which is the first page to be seen after logging in.



*Figure 4.2.2.1 Search Engine*



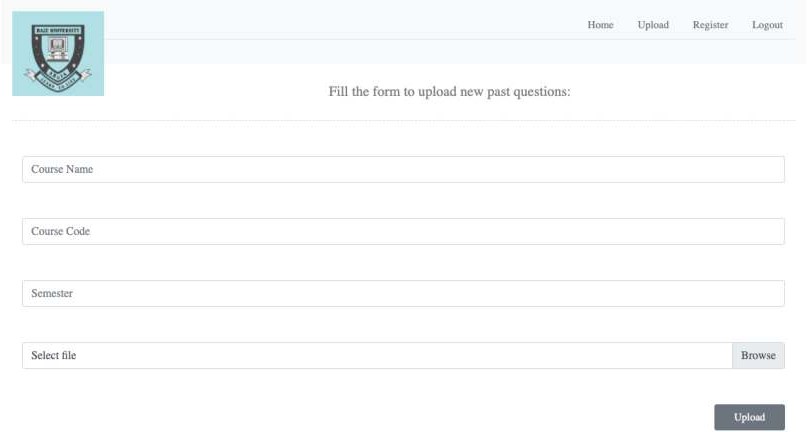
*Figure 4.2.2.2 Search Engine Result*

Above is the search engine after a “computer science” search.

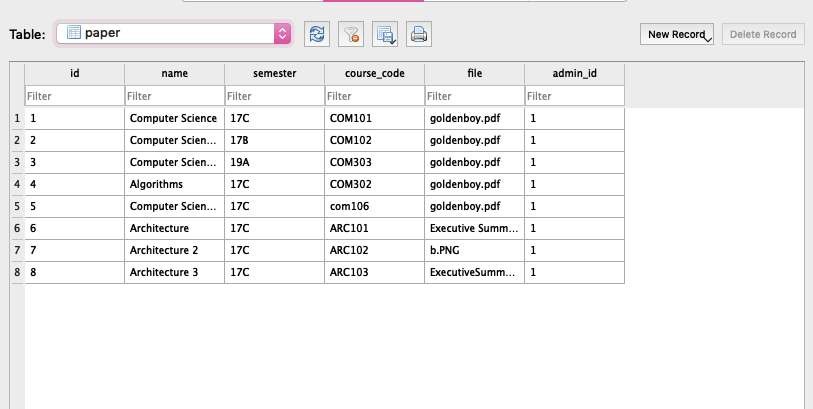
#### Uploading the past questions

During implementation, a decision needed to be made what method to use in getting the past question paper into the system for use. The decision was between acquiring and uploading all the past question papers myself or enabling admins or lecturers at Baze University to do so. The former was initially chosen but later decided against it because, as a final year student, I will soon no longer be at Baze University and there will be no one dedicated enough to do it. The latter decision was chosen to divide the work load to be done my multiple admins or have each lecturer upload theirs after each exam.

The upload page was created to allow only admins access to inputting any documents into the system.



*Figure 4.2.3.1 Upload page*

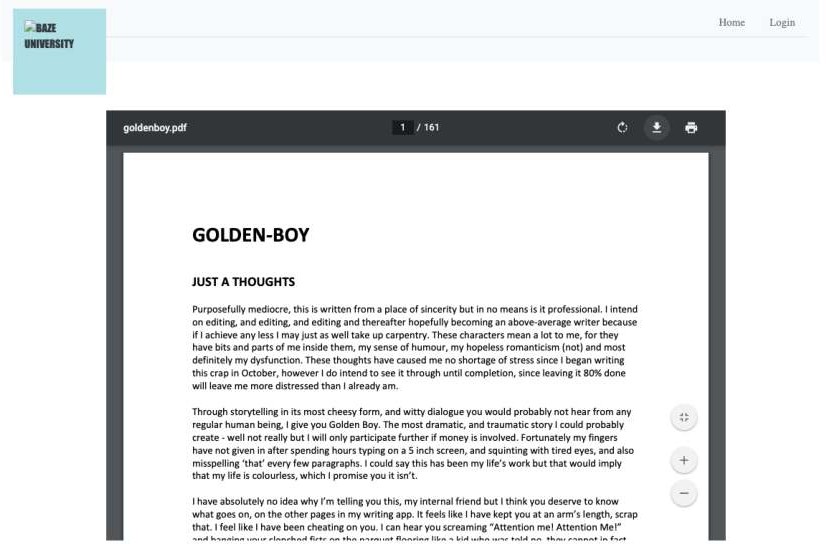


*Figure 4.2.3.3 Upload database view*

#### Viewing and/or downloading past questions

The main reason for this project is to provide students of Baze University with an alternative to going to the library for past question paper. Therefore, a necessary feature

of this system is for them to be able to access the documents on the website and have the option of downloading them if necessary. The system not only allows the students to study directly on the website but it enable them to download the documents for personal use. Below can be found an image of the view for studying and downloading the past questions.



*Figure 4.2.4.1 Viewing and download page*

## Implementation Problems and Overcoming them

As expected, many issues were encountered during the implementation process because it was also a process of learning how to use flask as a beginner. However, most of the problems encountered were resolved in the end with help from my supervisor when needed.

The first and biggest problems encountered was with learning how to use the Flask Framework. Flask is relatively easy to use but is difficult when finding solutions for

complicated problems. For instance, during the development process, I encountered issues with importing resources into the project that had to do with SQLAlchemy and the database because some of the database resources Couldn’t be used in Flask. Finding a way around this was difficult but after many hour and a couple days of research, a hack was found on GitHub which aided in fixing the issue. Websites like GitHub and Stack Overflow became a favourite for finding solutions to a lot of problems. If the internet was searched extensively to no avail, my supervisor would most likely be able to direct me on what to do.

The second instance of a problem encountered was with creating the database. After it was created, some issues arose such that it needed to be updated, it kept showing the old database because the project had stored it in a cache that was nowhere to be found. After searching for a solution extensively and consulting my supervisor, it was decided that the best cause is to delete he old database and initiate a new one with a specified location which solved the problem.

Another issue that was encountered is in regards to using JQuery and Ajax for live updates to the system and well as getting and posting values to the system. The main issue with this was my previous lack of experience using them which made me really confused. After a few days of learning and trying out different methods of implementing this, I finally had enough knowledge to be able to implement that successfully.

The next issue encountered was with trying to implement the past question viewing and downloading page. Initially, I had attempted to make it be on the same page as the search which proved to not work. A separate page was then created specifically for that purpose and it still didn’t work. For some reason, the documents weren’t being displayed at all. After consulting my supervisor for a fresh opinion on the issue, it was found that the route for the documents wasn’t coded correctly. After making changes to the route, everything worked perfectly.

Finally, some other issues that were encountered are during the:

* Implementation of the authentication and authorization.
* Getting and posting values to/from the system
* Posting values into the database

## Testing

Testing is one of the most important aspects of developing a good software. It helps to validate and verify whether all of functional requirements are met or not. Testing is important for making sure that the software quality is assured. This was carried out by running test data through the system to ensure that it works the way it’s supposed to. It was also used to sniff out any vulnerabilities in the system. All vulnerabilities are then isolated and reviewed from the rest of the system. This section will discuss all testing procedures carried out for the components of the system and a resulting test report will be included.

#### Tests Plans (for Unit Testing, Integration Testing, and System Testing)

Below is the test plan for the “Past Question Paper Archiving System”.

#### Test Identifier:

TEST LEVEL: Master Test Plan AUTHOR’S NAME: Bilkisu Tahir Mamman

AUTHOR’S CONTACT: [bilkisu2827@bazeuniversity.edu.ng](mailto:bilkisu2827@bazeuniversity.edu.ng)

#### Reference

* + - * + Past question paper archive
        + Work plane
        + Detailed project documentation
        + Test summary

#### Introduction

This is the master test plan for the past question paper archiving system. This will help to verify and validate the system requirements that were discussed in the third chapter of this document. Both white box and black box testing will be carried out for this project.

#### Features To Be Tested

The following aspects of the system were focused on during the testing of this system:

* + - * + Home page
        + Register page
        + Search engine
        + Log-in page
        + Log-out button
        + Database update
        + Viewing/download page
        + Upload page
        + Confirmation alerts

#### Approach

The built-in debugger for my computer will be used along with the browser inspection feature that enables inspection of the system as it runs to see how each function is operating and how values are exchanged between the system, program and database which will help in isolating any issues.

#### Deliverables

The deliverables for this test plan are as follows:

* + - * + Test cases
        + Test report
        + Traceability matrix
        + Test results
        + Error report

#### Approvals

Bilkisu Tahir Mamman Baze IT official

#### Test Suite (for Unit Testing, Integration Testing, and System Testing)

**Test case TC-001(Admin Login)**

*Table 4.1 Test Suite Performed for Login*

Test suite ID R-116

Test case ID TC-001

Test case summary Ensuring that admins can login

Related requirements R-116

Prerequisite - The system must be up running

* Admins must be registered
* Uninterrupted internet connection

Test procedure - Open website in browser

* Navigate to login page
* Login

Test data - Email

* Password

Expected result User should be able to login successfully

Actual result The user logged in successfully

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

**Test case TC-002(Log out)**

### Table 4.2 Test Suite Performed for Logout

Test suite ID R-117

Test case ID TC-002

Test case summary Ensuring users can log out

Related requirements R-117

Prerequisite - Website is running

* + Uninterrupted internet connection
  + Users are already logged in

Test procedure - Login

* + Click log out

Test data Log out

Expected result Users should be logged out

Actual result User was logged out

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

**Test case TC-003(Search engine)**

### Table 4.3 Test Suite Performed for Search engine

Test suite ID R-118

Test case ID TC-003

Test case summary Ensuring the search engine works and produces the desired results

Related requirements R-118

Prerequisite - Website is running

* + Uninterrupted internet connection

Test procedure - Type keyword in search engine

* + Click “search button”

Test data - Search

* + Key word: “computer science”

Expected result List of results for search

Actual result List of result

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

**Test case TC-004(Database update)**

### Table 4.4 Test Suite Performed for database update

Test suite ID R-119

Test case ID TC-004

Test case summary Changes are being reflected in the database

Related requirements R-119

Prerequisite - Logged in to system

* Uninterrupted internet
* New user or document to upload

Test procedure - Fill a form

* Submit form

Test data - Past question paper

* Past question paper details

Expected result Update should be reflected in the database

Actual result Update can be found in the database

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser and DB Browser for SQLite

**Test case TC-005(Viewing/Downloading)**

### Table 4.5 Test Suite Performed for viewing/download

Test suite ID R-120

Test case ID TC-005

Test case summary User should be able to open and download any document Related requirements R-120

Prerequisite - Logged in

* Uninterrupted internet
* Availability of searched document

Test procedure - Search for a paper

* Click the open button
* Click download button is required

Test data - Key word:” Computer Science”

Expected result Document should be displayed and downloaded if required

Actual result Document was displayed and downloaded when the download button was clicked

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

**Test case TC-006(Uploading)**

### Table 4.6 Test Suite Performed for upload

Test suite ID R-121

Test case ID TC-006

Test case summary Admins should be able to upload new past questions

Related requirements R-121

Prerequisite - Logged in as admin

* + Uninterrupted internet
  + New past question paper

Test procedure - Log in

* + Navigate to “upload”
  + Fill form
  + Submit form

Test data - New past question paper

Expected result Document should be uploaded and an alert should notify the user of this

Actual result Document was uploaded and an alert was received to confirm this

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

**Test case TC-007(Confirmation alerts)**

### Table 4.7 Test Suite Performed for confirmation alert

Test suite ID R-122

Test case ID TC-007

Test case summary Verifying that alerts are received to confirm form submissions

Related requirements R-122 Prerequisite - Logged in

* Uninterrupted internet
* Form data

Test procedure - Log in

* Navigate to upload
* Fill form
* Submit form

Test data - Past question paper demo

* Past question paper details

Expected result Expected to receive an alert to verify the data was saved in the database

Actual result An alert was received

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

**Test case TC-008(Home page)**

### Table 4.8 Test Suite Performed for home page

Test suite ID R-123

Test case ID TC-008

Test case summary The home page is opened when the URL is searched

Related requirements R-123

Prerequisite - Uninterrupted internet

* Browser is running

Test procedure - Search the URL: http://127.0.0.1:5000/

Test data - URL: “http://127.0.0.1:5000/” Expected result Landing on the home page

Actual result The home page was opened

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

* + 1. **Test Traceability Matrix (for Unit Testing, Integration Testing, and System Testing)**

### Table 4.9 Test traceability matrix

|  |  |  |  |
| --- | --- | --- | --- |
| Reqt. # Description Priority | Test Case | Test Date | Test Result |
| R-116 5  Ensuring that admins can login | 1 | 08/08/2020 | Pass |
| R-117 2  Ensuring users can log out | 2 | 08/08/2020 | Pass |
| R-118 Ensuring the search engine works and 5 | 3 | 08/08/2020 | Pass |

produces the correct results

R-119

Changes are being reflected in the

database

4 4 08/08/2020 Pass

R-120

Users should be able to open and

download any document

4 5 08/08/2020 Pass

R-121

Admins should be able to upload new

past questions

4 6 08/08/2020 Pass

* 1. Verify that alerts are received to confirm form submissions

1 7 08/08/2020 Pass

* 1. The home page is opened when the URL is searched

2 8 08/08/2020 pass

* + 1. **Test Report Summary (for Unit Testing, Integration Testing, and System Testing)**

### Table 4.10 Test report summary

|  |  |
| --- | --- |
| SUMMARY OF TEST CARRIED OUT | RESULTS |
| Number of functions tested: | 8 |
| The number of functions not tested: | 2 |
| The number of tests passed: | 8 |
| The number of tests failed: | 0 |
| Percentage of tests passed: | 100% |
| Percentage of tests failed: | 0% |

#### Error Reports and Corrections

### Table 4.11 Error reports and correction

Error Correction

Logical error due to mistake in code Use the error report to isolate the issue

and fix it

Runtime error End execution and use the runtime error report to fix error

Database error Look for error in code or re-initiate database

## Use Guide

This user guide serves as a manual for new users to help them navigate the system easily. A new user should type the URL into a search engine and log in as an admin or a student. Next step is to make use of the search engine within the system to search for a past question paper by entering a keyword and clicking the search button. Then click the open button next to a specific paper to open the document and download it if required.

For admins, after logging in, next step is to navigate to “uploads”, fill the form and submit to add a new document or navigate to “register” to add a new admin by filling the form with the new users details and submit.

Finally, all users can simply click the “log out” button to log out.

## Summary

This chapter has given a detailed explanation of all the features implemented in the past question paper archiving system. Problems that arose during the implementation were also discussed as well as providing images and screenshot or the source code and the user interface of the system.

After the successful implementation of the system, the testing process was also explored. Test suites for each feature and functionality of the system were all tested multiple time to ensure that they work the way they were intended.

This chapter is very critical in the software development life cycle of a project because adequate implementation is important to provide a reliable and efficient system that meets the requirements of the organization.

# Chapter 5: Discussion

## Overview

In this final chapter of the project, processes that have been discussed from the beginning of the project to arrive at the completed system will be discussed. This project has been successful in implementing all features that were discussed in the requirements chapter as well as also taking care to not make the same mistakes as the systems discussed in the literature review.

An objective assessment of the implemented system will be discussed in section 5.2 along with the limitation and problems encountered, and future enhancements discussed in sections 5.3 and 5.4 respectively. And lastly, any recommendations for future improvements to the system and a summary of all aspects explored in this chapter will be contained in sections 5.4 and 5.5 respectively.

## Objective Assessment

The past question paper archiving system and all of its features and functionalities have been successfully implemented. The objective of the system which is to provide an archive for all past question papers to be used by the students of Baze University was reached.

The system was implemented in such a way that any future improvements and updates can easily be integrated without affecting its core functionalities. This system has the advantage of being scalable for even larger groups of people is necessary.

Finally, this completed system performs all functions that its meant to and meets the requirements as discussed in chapter 3 of this documentation. I is ready to be integrated into the Baze University student portal and server.

## Limitations and Challenges

Due to the COVID-19 pandemic, the implementation capacity of this project was limited. Not physically being in school has meant that access to resources and supervisors was restricted.

The use of the system is currently limited only to students and staff of base university. Within the system, certain aspects are also restricted to certain users for security reason. However, this should not affect any students use of the system.

The main challenge encountered during the making of this project was gaining access to the past question papers. As stated above, going to the library and scanning all the past question papers is not an option due to the pandemic. So, all communications have been done through calls and emails which the client is aware of and is also putting effort into getting the softcopy of the past question papers from the exam officer.

## Future Enhancements

This project has the great advantage of having a lot of room for improvement. Therefore, the end of this project doesn’t mean that improvement could not be made to the system. In fact, the clients have feature which they have reserved to be added in the future. The first feature is the ability to use the past questions to be able to make possible predictions on future examinations at baze university. The system can also allow students to study the past questions in a quiz format which will give them a grade as well as corrections at the end. This can be used to improve studying efficiency for the students and make them more likely to want to access their portals.

Another enhancement that will be made in the future is to the user interface. As time go by, UI design can start age. It is necessary to make sure that the user interface keeps up with the changing times as well as also keeping up with the changes made to the already existing student portal.

## Recommendations

From all the research carried out for this project, it is evident that there is a need for all Nigerian universities to pay more attention and invest in the development and facilitation of online systems for providing academic resources and enabling online education, especial during this pandemic period. With that goal in mind, the following recommendations should be considered:

* Universities should have an online system dedicated to providing online resources for its students, especially does that are not on campus.
* Universities should look into having online schooling being a core part of the opportunities they provide. This brings in more finances without the limitations that a classroom has.
* And finally, students should be encouraged to learn about computer systems and know the basics of software development even if they are not going into a technology related field because the 21st Century is run by technology. It is a huge disadvantage not to have this knowledge.

## Summary

With this section, the project documentation has been completed. The project has explored and discussed the importance of technology and online resources within the Nigerian educational system. Due to the COVID-19 pandemic, the vulnerability and reliance of Nigerian universities on on-campus education has been exposed. This project aimed to be an aid and improvement on the online systems at Baze University to be a step in the right direction to making online education possible.

# REFERENCES

Connections academy, 2016. *Connections academy.* [Online] Available at: https://[www.connectionsacademy.com/support/resources/article/five-main-](http://www.connectionsacademy.com/support/resources/article/five-main-) benefits-of-online-school

[Accessed 8 september 2020].

Docsity, 2013. *Docsity.* [Online] Available at: https://[www.docsity.com/en/news/exams-and-study/advantage-](http://www.docsity.com/en/news/exams-and-study/advantage-) exampapers/

[Accessed 24 August 2020].

exam past questions, n.d. *exampastquestions.com.* [Online] Available at: https://[www.exampastquestions.com/exams/paper/2539-BAZE-](http://www.exampastquestions.com/exams/paper/2539-BAZE-) UNIVERSITY-POST-UTME-EXAMS

[Accessed 25 August 2020].

North eastern university, 2020. *North eastern university.* [Online] Available at: https://[www.northeastern.edu/graduate/blog/benefits-of-online-learning/](http://www.northeastern.edu/graduate/blog/benefits-of-online-learning/) [Accessed 8 September 2020].

Read, N., 2016. *UNESDOC Digital Library.* [Online] Available at: https://unesdoc.unesco.org/ark:/48223/pf0000245569 [Accessed 24 August 2020].

samphina academy, 2013. *samphina* *academy.* [Online] Available at: https://samphina.com.ng/post-utme-questions-answers-baze-university/ Tutorials point, 2016. *tutorialspoint.* [Online] Available at: https://[www.tutorialspoint.com/sdlc/sdlc\_agile\_model.htm](http://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm) [Accessed 4 September 2020].

Visual Academy, 2009. *onlineschools.org.* [Online] Available at: https://[www.onlineschools.org/visual-academy/the-history-of-online-](http://www.onlineschools.org/visual-academy/the-history-of-online-) schooling/

[Accessed 25 August 2020].

# APPENDICES

## Appendix B – Interview

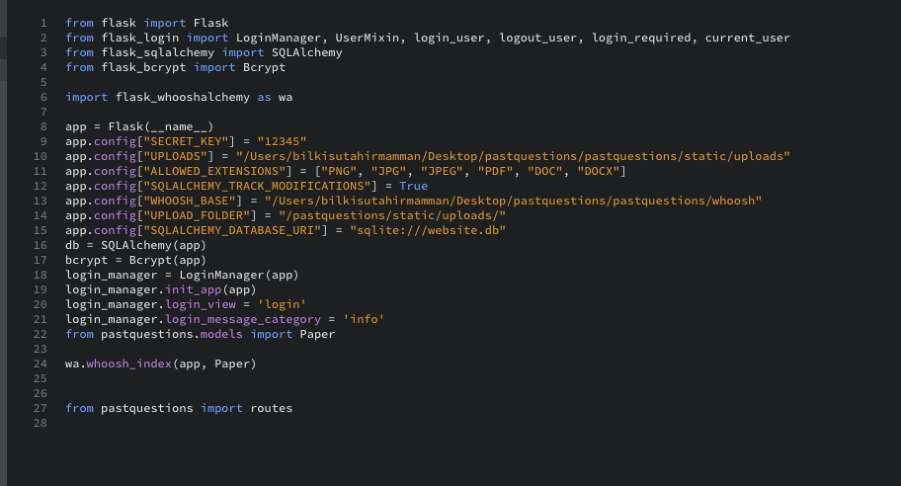
#### Baze IT department

* + 1. Describe your role in Baze university.
    2. What do you want from this system?
    3. How will the system improve the school?
    4. How do you think this affects the students?

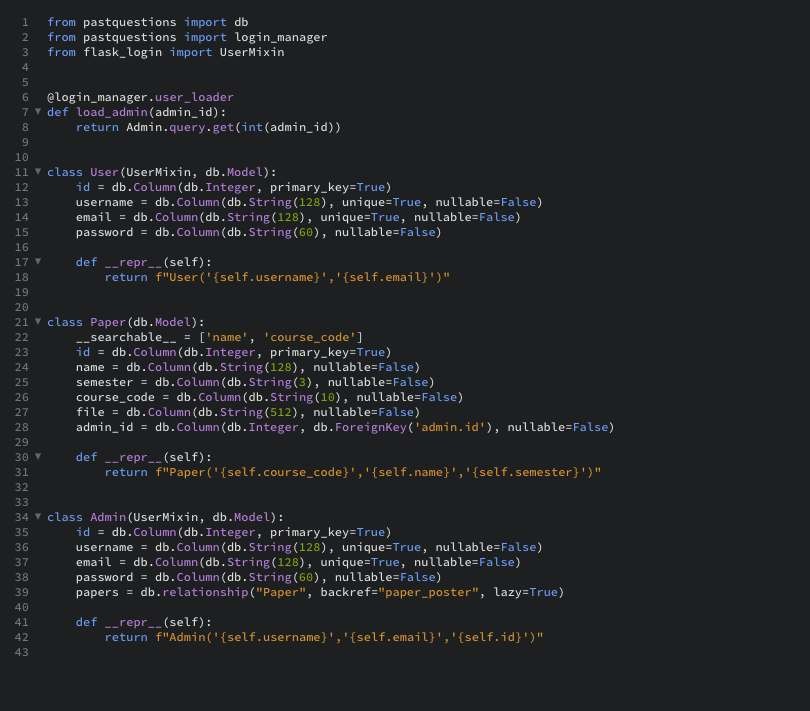
#### Baze students

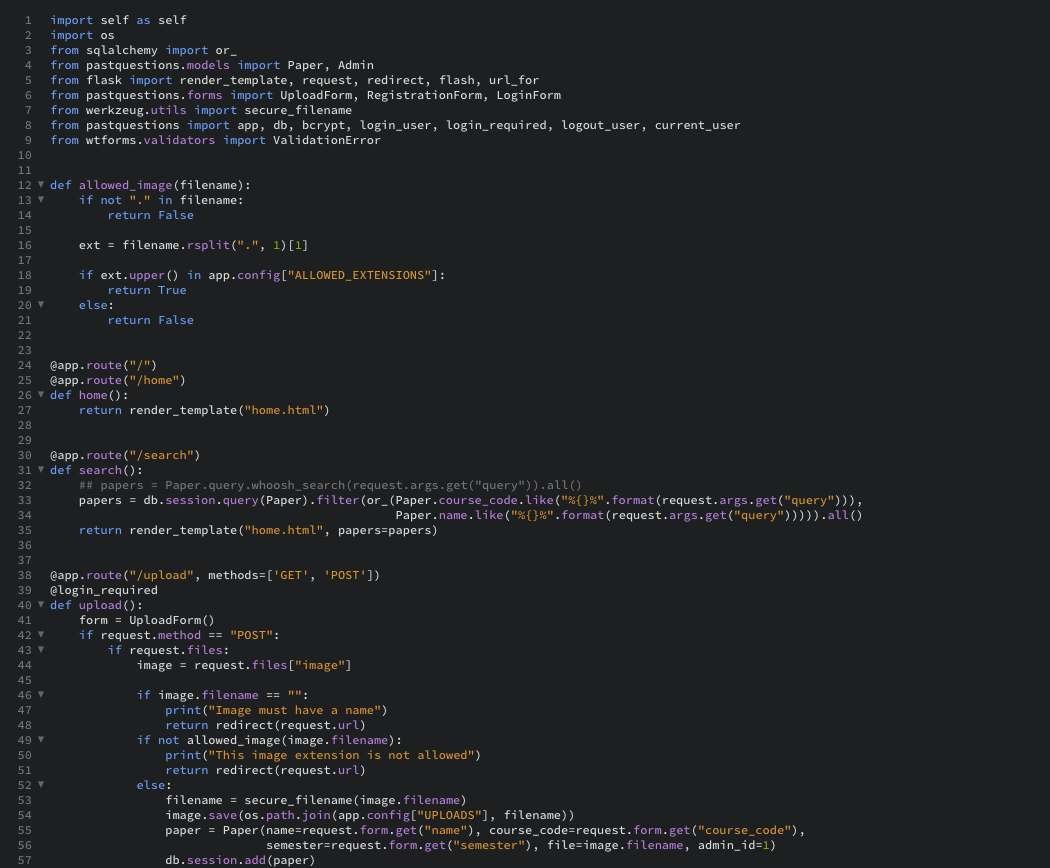
1. Can you explain how to get past question?
2. What issues do you have with the process?
3. How do you think the system can improve?
4. What would you like to be included in the online system?

## Appendix C - Source Codes

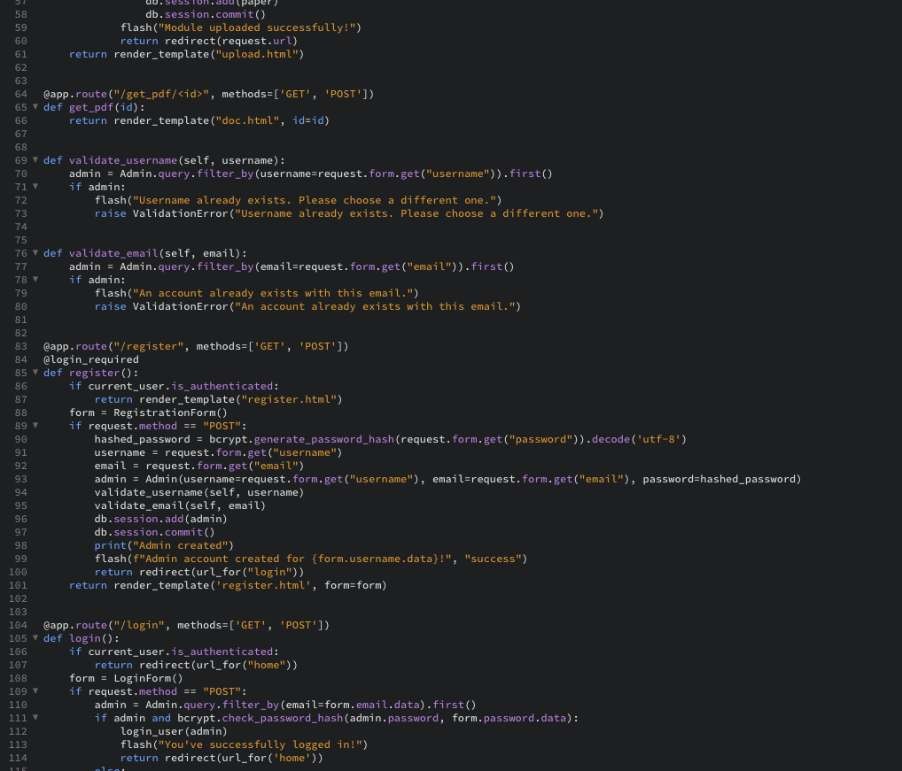
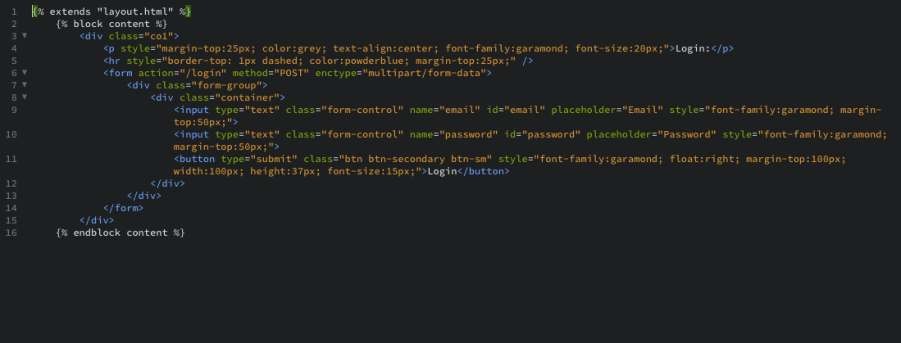




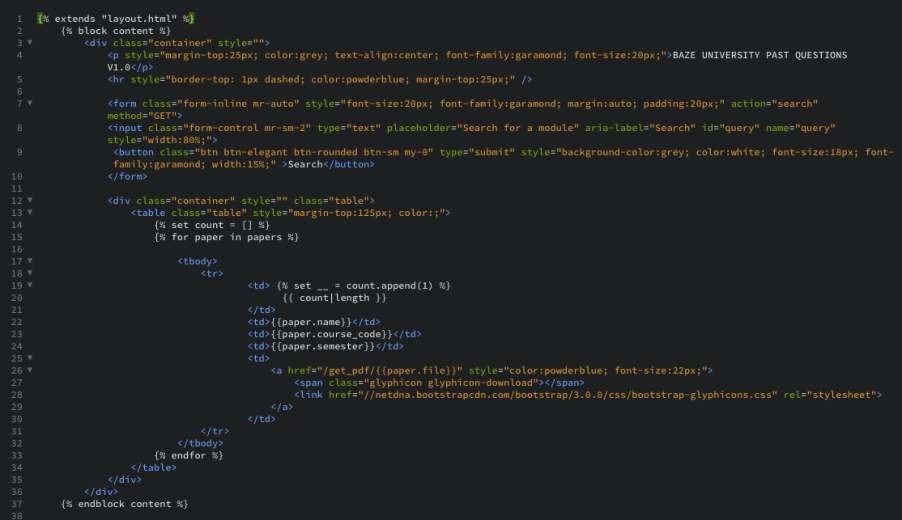


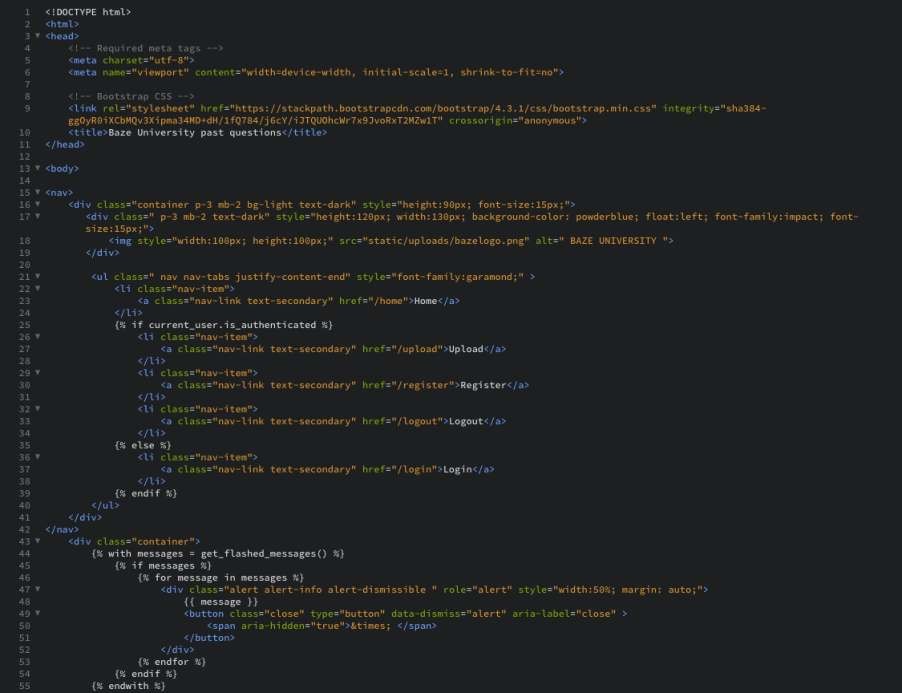












## Appendix D – Test Cases

**Test case TC-007(Confirmation alerts)**

*Table 4.7 Test Suite Performed for confirmation alert*

Test suite ID R-122

Test case ID TC-007

Test case summary Verifying that alerts are received to confirm for submissions

Related requirements R-122

Prerequisite - Logged in

* Uninterrupted internet
* Form data

Test procedure - Log in

* Navigate to upload
* Fill form
* Submit form

Test data - Past question paper demo

* Past question paper details

Expected result Expected to receive an alert to verify the data wa saved in the database

Actual result An alert was received

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

**Test case TC-008(Home page)**

### Table 4.8 Test Suite Performed for home page

Test suite ID R-123

Test case ID TC-008

Test case summary The home page is opened when the URL is searched

Related requirements R-123

Prerequisite - Uninterrupted internet

- Browser is running

Test procedure - Search the URL: http://127.0.0.1:5000/

Test data - URL: “http://127.0.0.1:5000/” Expected result Landing on the home page

Actual result The home page was opened

Status Test case passes

Remarks The test was carried out successfully

Created by Bilkisu Tahir Mamman

Date created 8th September, 2020

Executed by Bilkisu Tahir Mamman

Date of execution 8th September, 2020

Test environment Hardware: macbook

Software: google chrome browser

## Appendix E – User Guide/Manual

#### Students User Guide

1. Navigate to the past question paper archive via the school student portal
2. Next step is to make use of the search engine within the system to search for a past question paper by entering a keyword.
3. Click search button.
4. Click the open button next to a specific paper to open the document
5. Click the download icon to download it if required.

#### Admins User Guide

1. Log in as an admin
2. Navigate to “uploads”
3. Fill the form and submit to add a new document or
4. Navigate to “register” to add a new admin by filling the form with the new users details and submit.

-Finally, all users can simply click the “log out” button to log out.