**CHAPTER FOUR**

**IMPLEMENTATION AND EVALUATION/TESTING**

**4.0 INTRODUCTION**

System implementation is the process of defining the user requirements and designing a system to meet such requirements. This is the stage of systems development in which hardware and software are acquired, developed and installed. This section provides details of the various modules in the platform, how this modules work and how they can be tested (Grady, 1994).

**4.1 IMPLEMENTATION OF THE PROPOSED SYSTEM**

This section describes how the proposed platform works and how it is structured. The platform is structured in such a way that each module in the system is loosely coupled to each other. The platform can be grouped into two parts generally which are the backend module and client side module.

**Backend Module**

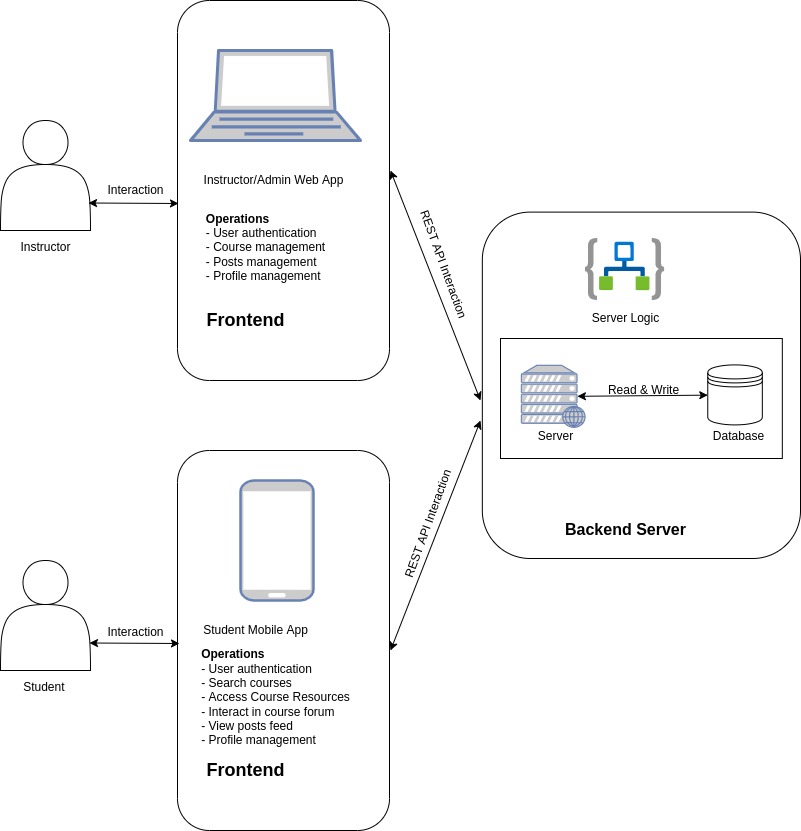
The server component of the platform is responsible for handling all business login that makes it possible for users to interact with the platform. The server component of the platform is hosted on an online server accessible via the internet. This module provides several functions from user authentication to course management functionalities.

**Frontend Module**

This frontend module of the platform is responsible for providing an interface to users making it possible for them to access functionalities. The frontend module of the platform  is accessible via a website and a mobile application; the website can be accessed on https://tollgator.vercel.app/ and the mobile application can be downloaded on https://play.google.com/store/apps/details?id=com.devcircus.teogate

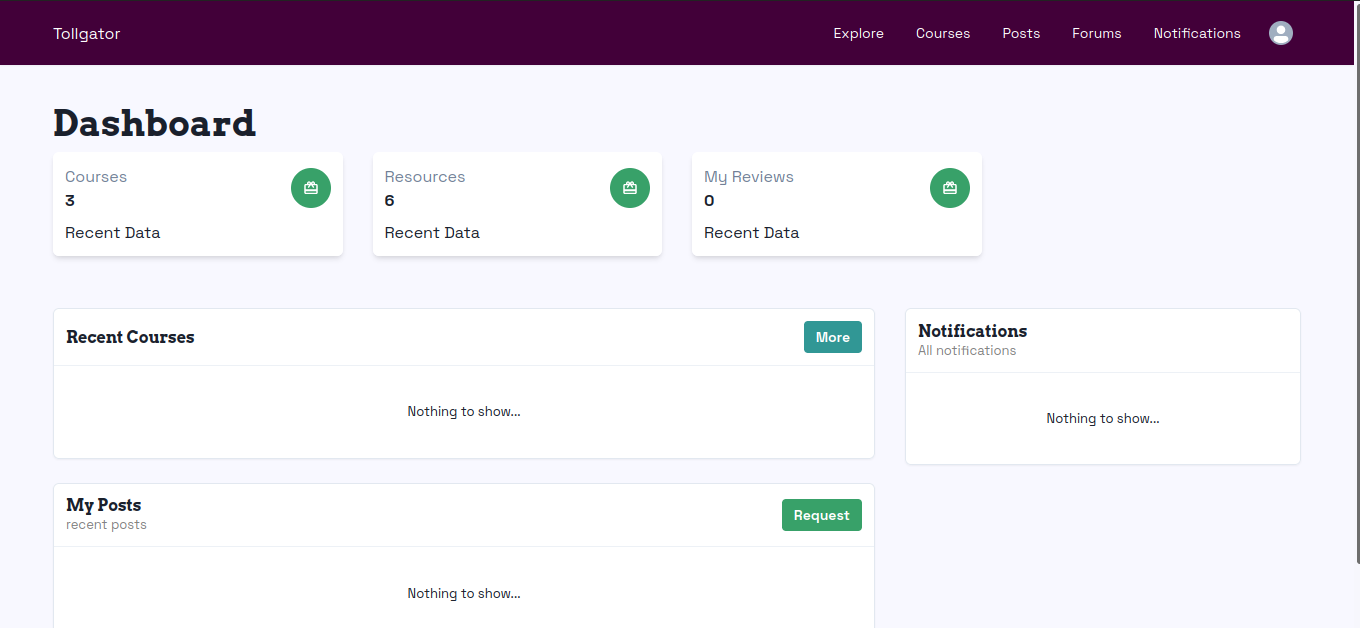
**4.1.1 HARDWARE COMPONENT**

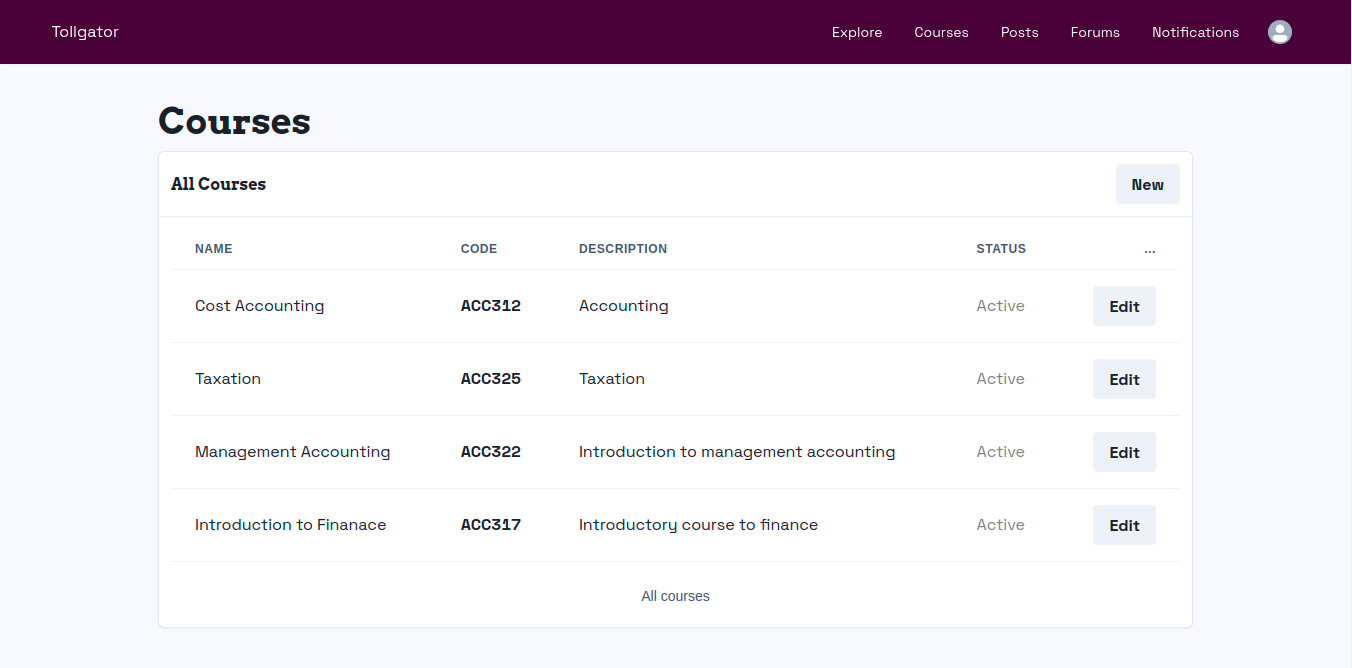
The proposed system runs primarily on computer devices with access to the internet. The backend server which is the major module of is based on a nodeJs server and hosted on a heroku dyno server which can be accessed on the internet. Users of the platform make use of computers and mobile phones to access its functionalities.

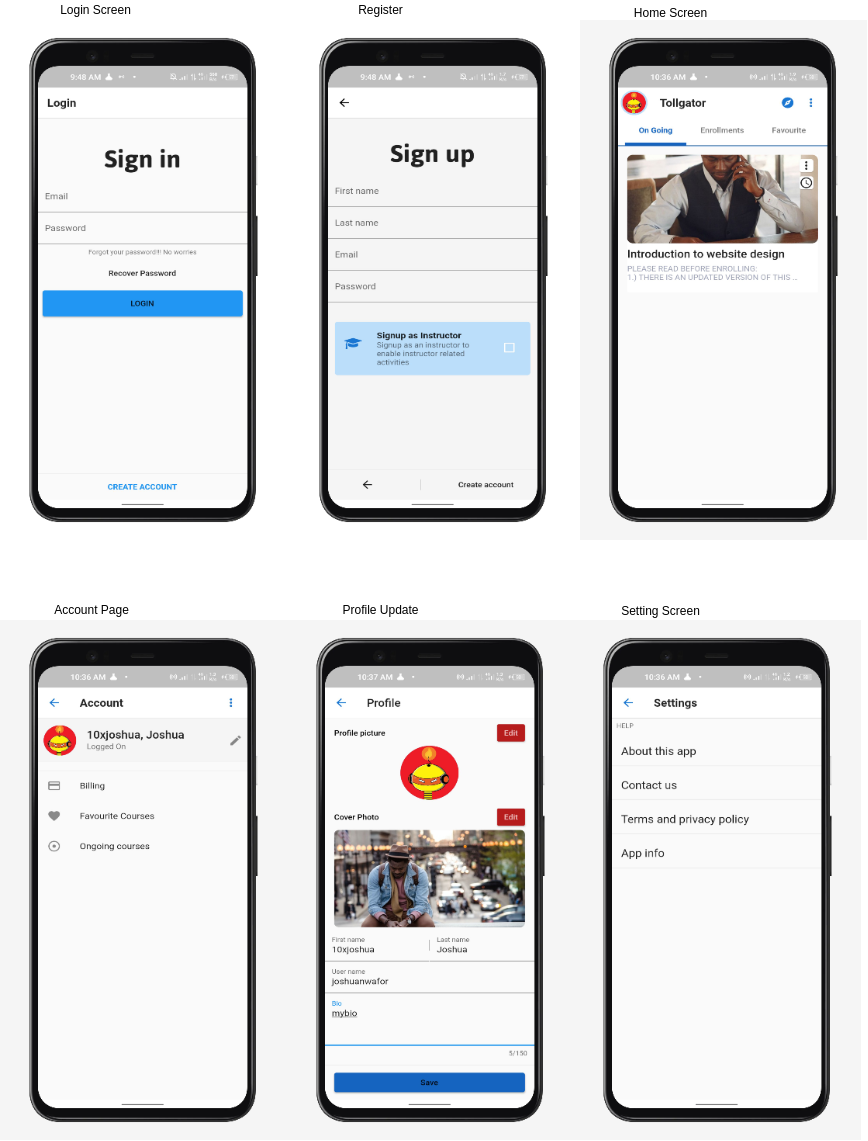
*Figure 4.1: User interaction with Virtual Library Learning Platform*

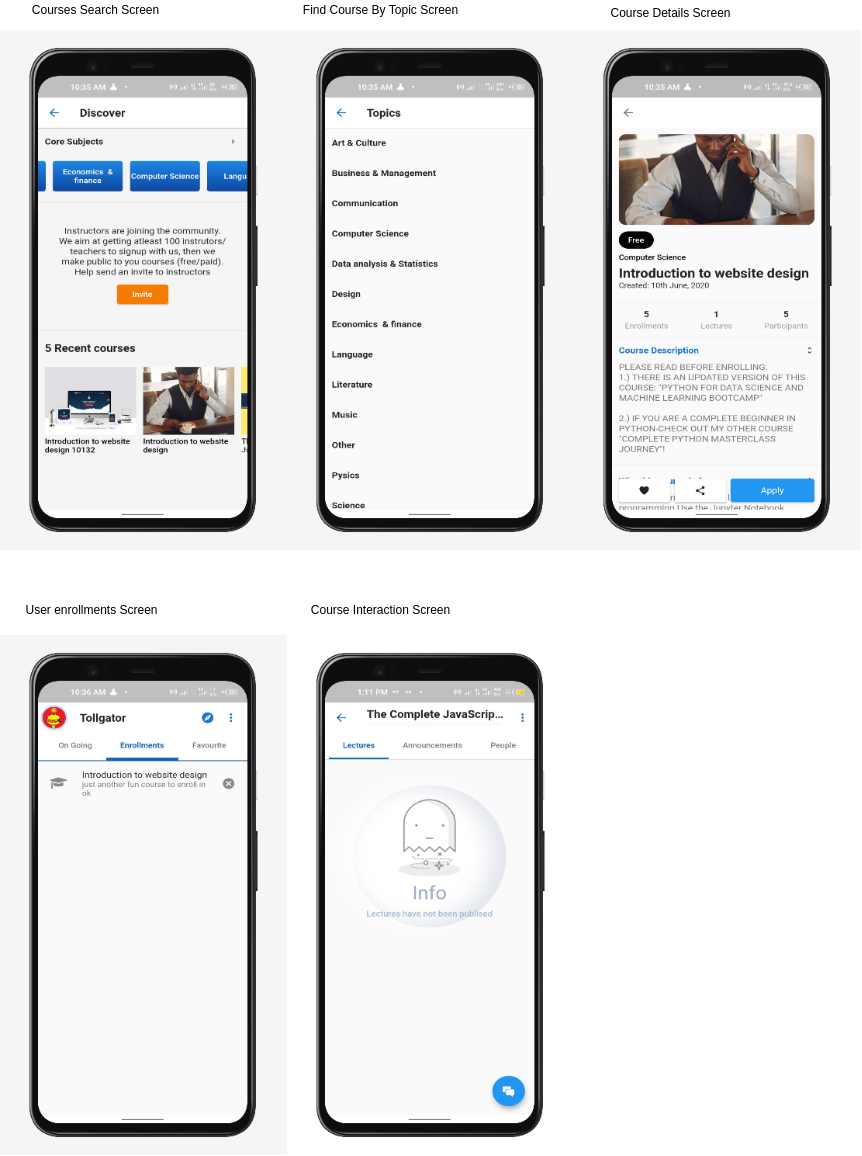
**4.1.2 SOFTWARE COMPONENT**

The proposed platform runs on a backend remote server and exposed REST endpoints making it possible for clients to access several functionalities from authentication to course resource management. The software component of the platform is divided into two parts; the backend software and client side software which is made available to the users of the system. The client side software can be accessed via a website and mobile application.

*Figure 4.2: Instructor/Admin Dashboard*

*Figure 4.3: Instructor/Admin Course Management Screen*

*Figure 4.4: Student Authentication and Account Management Screens*

*Figure 4.5: Other relevant screens on Student Mobile Application*

**4.2 HARDWARE REQUIREMENTS**

Below are the hardware requirements of the platform

* RAM: 512 MB Heroku ephemeral storage
* Heroku Web Server
* Operating System Disk
* Processor
* Mobile or Computer Device

**4.3 SOFTWARE REQUIREMENTS**

Below are the software requirements of the platform

* Android, Windows, or Linux Operating System
* NodeJs Server Software
* Mobile application software
* Web Browser
* HTML
* CSS
* JavaScript Programming Language
* React - Website development framework
* React Native - Mobile application development framework

**4.4 OTHER NEEDED REQUIREMENTS**

Below are other requirements needed  to effectively use to platform

* Data Bundle
* Internet Connectivity

**4.5 SYSTEM TESTING**

The main importance of system testing is to ensure that the program performance is as intended and meets the defined objectives of the project. The researcher chose to test the software first to discover all the faults and defects as well as the general functionalities in the software.

**4.5.1 INSTALLATION TESTING**

Most software systems have installation procedures that are needed before they can be used for their main purpose. Testing these procedures to achieve an installed software system that may be used is known as installation testing (Myers, 2004).  So far, the various modules have been tested to meet the requirements of the platform. The installation test ensures that when every module in the system is integrated they produce the desired result. According to Kaner &  Pettichord  (2001), Installation testing may also be considered as an activity-based approach to how to test something. For example, install the software in the various ways.. Check which files are added or changed on disk. Does the installed software work? What happens when you uninstall?

**Below are outlines test cases to be followed for successful installation and testing of the Tollgator mobile application;**

**Application Installation**

* Users should be able to search and install the mobile application from Google Playstore
* Application should be able to install successfully without any error.
* Application should be able to create their icon properly into menu tray & on Home screen.
* Users should be able to create and login successfully after installation
* Learners should be manage profile settings
* Learners should be able to browse and access course resources

**Application Uninstallation**

* Application should be able to uninstall successfully without any error.
* Application icon should not be present on the device where you would expect to see the application icon after successful uninstallation.
* After application uninstallation, correlated files & folders should be able to remove properly from the Phone ROM.

**4.5.2 FUNCTIONALITY TESTING**

Functional testing is the process through which QAs determine if a piece of software is acting in accordance with predetermined requirements. It uses black-box testing techniques, in which the tester has no knowledge of the internal system logic. Functional testing is only concerned with validating if a system works as intended. (Browserstack, 2021).

**Authentication and Profile management Test**

**Description:** User should be able to successfully create an account

**Preconditions:** The user should not have an already existing account on the platform and must have an active email address

**Steps:**

1. User should create an account using mobile application
2. User should update profile details including profile picture
3. User should be able to browse courses available on the mobile application

**Expected result:** On successful registration, the user should be redirected to the mobile application dashboard and have access to a list of courses and navigation options. On successfully update of profile picture, the user should see the recent profile picture update on the profile page.

**Course Accessibility Test**

**Description:** User should be able to successfully search and subscribe to a course

**Preconditions:** The user should have an account already created on the platform

**Steps:**

1. User should login existing account.
2. User should browser for any course of interest.
3. User should be able to view the details of a course and enroll for the course in question.
4. Users should be able to view course resources and forum.

**Expected result:** On successful login, user should be presented with a list of courses from which he can view the details of any of the courses listed. On successful enrollment to a course, user should gain access to course resources.

**4.5.3 USABILITY TESTING**

Usability testing is a technique used in user-centered interaction design to evaluate a product by testing it on users. This can be seen as an irreplaceable usability practice, since it gives direct input on how real users use the system. (Nielsen, 1994).

**Mobile Application Usability Test Script**

User is expected to download and install the mobile application from Google Playstore and perform the tasks outlined below:

**Tasks**

1. Begin by downloading the app to your device.
2. Create and login to your account
3. Update account profile on the profile screen
4. Update profile picture on the profile screen
5. Browse through courses in the courses screen
6. View course details of any course of your choice by clicking on the course card
7. User should enroll to a course by clicking on the enroll button on the course details page
8. On successful enrollment, user should access course resources on the course resource screen

**Survey Questions**

1. What was the worst thing about your experience?
2. What other aspects of the experience could be improved?
3. What did you like about the mobile application?
4. What other comments do you have on the mobile application?

**4.6 CONCLUSION**

This chapter provided information on how the platform is being implemented, how it is being structured, the various components required to effectively use the platform and how it is tested to ensure it meets the specified requirements.

**CHAPTER FIVE**

**5.0 SUMMARY**

**CONCLUSION AND FUTURE WORK**

The project was aimed to develop a platform that makes it possible for instructors and learners to communicate seamlessly via the internet, this was done in consideration of how course resources are being communicated to learners. The platform provides a mobile application which could be used by learners to easily browse and access course resources. The system was developed using the React.js (Javascript library for creating user interfaces), Firebase development toolkit, Heroku server, Node.js (run-time environment for JavaScript) and MongoDB as the database management software of the system. The project requirement analysis was conducted in a systematic manner using the fact-finding techniques described in chapter three to fully understand the current system.

Further developments that could be made to the platform are real time video lesson sessions between students  and instructors, payments for course enrollments and sell of PDF resources on the platform.

Chapter one of this project focused on giving the reader an overview of Virtual learning applications, how they work, and some of they components that make up a Virtual learning application. The project went further to state the goal of developming a Virtual Learning Platform for the Faculture of Management Science 300 Level students, pointed out the current problem in the method of teaching used by lecurers and stated the objectives of the project.

Chapter two of this project provided more details on Virtual Learning Applications, the current Learning System currently used by students of the Faculty of Management Science, and the Effectiveness of the proposed system to both instructors and students.

Chaper three of the project described the structure and components of the application, programming languages used in development and justification for the use of those languages. The Virtual Library Learning Platform is built to make it possible for both instructors and learners to communicae and share information virtually. It also stated the requirements and system design of the platform.

Chapter four of the project provided information on how the platform is being implemented, how it is structured, the various components required to effectively use the platform and how it is tested to ensure it meets the specified requirements.

**5.1 CONCLUSION**

Generally, this research is based on the need for course resources to be made available on the internet and the need for effective interaction between learners and instructors. The developed system can help  university instructors communicate course resources and materials effectively and speedily to students through an easy-to-use interface.

I was able to build a platform that makes it easy for instructors and students to share information, interact and access information easily.

The platforms mobile application was built and customised for students primarily making it possible for them to easily access resources publised by instructors from their mobile devices.

Instructors interact with the platform using a customised and easily-to-use dashboard making it possible for them to manage various courses, share resources for their courses and publish other resources which is made availble to students.

**References**

BrowserStack. (2022). *Functional Testing : A Detailed Guide.* Available at: <https://www.browserstack.com/guide/functional-testing>.

Kaner, C., Bach, J. & Pettichord, B. (2001). *Lessons Learned in Software Testing: A Context-Driven Approach*. Wiley. p. 41.

Grady, J.O. 1994. *System integration.* Boca Raton, FL, USA: CRC Press, Inc.

Myers, G., Sandler, C., Badgett, T. & Thomas, M. (2004). *The Art of Software Testing (2ed.)*. Wiley.

Nielsen, J. (1994). *Usability Engineering*. Academic Press Inc.