PHP Project Documentation Report

Development of School of Computing and Information Technology Website

Version Number - 1.0

29th April 2023

Location: https://github.com/dbosire/MSc_webtechnologies-Exam

Branch: Master

KenyaWeb Technologies

OVERVIEW

The PHP Project Documentation Report provides a comprehensive report on the development of the School of Computing and Information Technology website for KenyaWeb Technologies.

Target Audience

The target audience for this document includes the project manager, the development team, and the client, KenyaWeb Technologies.

Project Team Members

- Dennis Bosire - Lead Developer / Database Administrator
- Paul Bett – Designer / Tester

Version Control History

Version	Primary Author(s)	Description of Version	Date Completed
1.0	1. Paul Bett **(**SCT313-2351/2022**)** 2. Dennis Bosire **(**SCT313-1832/2022**)**	Initial version	29th April 2023

TABLE OF CONTENTS

SECTION A: SOFTWARE PROJECT MANAGEMENT PLAN

1 INTRODUCTION:

1.1 Project Overview:

KenyaWeb Technologies has been contracted to develop a website for the School of Computing and Information Technology. The website will contain the following pages: Home page, academics page, programmes page, department page, login page, staff page, and apply now page. The website must be developed using the Laravel framework and MySQL database. The website must be database-driven, allowing the web administrator to update and modify content as needed. The website must also include the logo, menu bar, and footer on every page.

1.2 Project Deliverables:

- Fully functional website for the School of Computing and Information Technology with the following pages: home page, academics page, programmes page, login page, staff page, and apply now page.
- Database architecture used to store and retrieve data for the website.
- User manual detailing all features and content on the website.

2 PROJECT ORGANIZATION

2.1 Software Process Model

The software development process model used for this project is the Agile methodology. This methodology allows for flexibility in the development process, with an emphasis on collaboration between the development team and the client. The Agile methodology also allows for iterative development, where the product is continuously refined and improved upon throughout the development process.

2.2 Roles and Responsibilities

The following are the roles and responsibilities for the project:

- Project Manager: responsible for overseeing the entire project, ensuring that all tasks are completed on time and within the constraints set
- Lead Developer: responsible for developing the front-end and back-end of the website, ensuring that it meets the client's requirements, and communicating with the project manager.

- Database Administrator: responsible for designing and managing the database architecture for the website.
- Quality Assurance Tester: responsible for testing the website to ensure that it functions properly and meets the client's requirements.
- Web Administrator: responsible for updating the website with new information as needed.

2.3 Tools and Techniques

The following tools and techniques will be used for the project:

- Laravel Framework: a PHP framework for web application development.
- MySQL: a relational database management system for storing data.
- Git: a version control system for tracking changes to the codebase.
- Trello: a project management tool for tracking tasks and progress.
- Slack: a communication tool for the development team to stay in touch.

3 PROJECT MANAGEMENT PLAN

3.1 Tasks

Task-1: Requirements Gathering Description
In this task, the team will gather requirements from the assignment paper containing details for the website for the School of Computing and Information Technology. The team also analyzed the provided images and texts for the website.

Deliverables and Milestones: Requirements specification document

Duration: 1 day

Task-2: Designing the Front-end Description
In this task, the team will design the front-end of the website. This includes designing the Home Page, Academics, Department Page, Programmes, Login, Staff, and Apply Now pages as specified by the client. The team will use Laravel Framework for PHP and MySQL for database management.

Deliverables and Milestones: Front-end design document, website prototype.

Duration: 5 days

Task-3: Designing the Back-end Description
In this task, the team will design the back-end of the website. This includes designing the database architecture and implementing it in MySQL. The team will also develop the forms and functionality required by the client for updating the website.

Deliverables and Milestones: Back-end design document, database architecture document, website with functional back-end

Dependencies and Constraints: Front-end design document, compatibility issues with Laravel Framework and MySQL Risks and Contingencies: Delay in back-end design, errors in database architecture

Duration: 5 days

Task-4: Integration and Testing Description
In this task, the team will integrate the front-end and back-end of the website and test the website for functionality and compatibility issues. The team will also perform user acceptance testing to ensure that the website meets the client's requirements.

Deliverables and Milestones: Integrated website, testing report, user acceptance report Resources

Dependencies and Constraints: Completion of front-end and back-end design, availability of resources for testing Risks and Contingencies: Compatibility issues with different web browsers, errors in integration

Duration: 2 days

3.2 Assignments

Task Name	**Assigned To**	**Estimated Duration (days)**
Requirements Gathering	Paul Bett	1
Designing the Front-end	Dennis Bosire	5
Designing the Back-end	Paul Bett & Dennis Bosire	5
Integration and Testing	Paul Bett & Dennis Bosire	2

3.3 Timetable

Task Name	**Start Date**	**End Date**	**Duration (days)**
Requirements Gathering	2023-04-14	2023-04-14	1
Designing the Front-end	2023-04-15	2023-04-19	5
Designing the Back-end	2023-04-19	2023-04-24	5
Integration and Testing	2023-05-25	2023-05-26	2

SECTION B: SOFTWARE REQUIREMENTS SPECIFICATIONS

1 INTRODUCTION

1.1 Product Overview

The School of Computing and Information Technology website is a database-driven web application developed using Laravel framework and MySQL database. The website provides information on the school's academics, programmes, staff, payment methods, and online application. The website has five main pages, namely Home, Academics, Departments,, Programmes, Login, Staff, and Apply Now. The website features a logo, menu bar, and footer on every page. The back-end of the website allows the web administrator to update information related to the website areas that require updating.

2 SPECIFIC REQUIREMENTS

2.1 External Interface Requirements

2.1.1 User Interfaces

The user interface should be user-friendly, consistent across all pages, and visually appealing. The website must have a logo, menu bar, and footer on all pages. The website must have the following pages:

- Home Page: Provides information about the current activities of the School, how to apply with a link to Apply Now, message from the director, etc.
- Academics: Provides information regarding various academic activities including and not limited to boot camps, Cisco academies, and related information as well as Microsoft-related courses that students undertake to enrich the programs they are taking. The Academics page should also state the department of IT.
- Departments: This page should have information on the available departments and the courses offered per department organizing all details as per department.
- Programmes: This page should enlist the various programmes of the school at large and allows searching per level of study.
- Login: This page should provide the user with the ability to log in to the system. Based on the credentials provided the user will be taken to the front end and admin logins will be directed to the back-end pages.

2.1.2 Hardware Interfaces

The website should be compatible with all major browsers and devices, including mobile phones and tablets. Testing on all these devices is key to ensure it runs seamlessly for all users.

2.1.3 Software Interfaces
The website should be built using Laravel framework and MySQL for database management.

2.1.4 Communications Protocols
The website should use HTTP and HTTPS protocols for communication between the server and the client.

2.2 Software Product Features

The website should have the following features:

- Home Page: The Home Page should provide information about the current activities of the School, how to apply with a link to Apply Now, message from the director, etc.
- Academics: The Academics page should provide information regarding various academic activities including and not limited to boot camps, Cisco academies, and related information as well as Microsoft-related courses that students undertake to enrich the programs they are taking. The Academics page should also state the department of IT.
- Departments: This page should provide information on the available departments and the courses offered per department organizing all details as per department
- Programmes: This page should enlist the various programmes of the department and the school at large.
- Login: This page should provide the user with the ability to log in to the system.
- Staff: This page should provide information about the staff of the School of Computing and Information Technology.
- Apply Now: This page should provide the user with the ability to apply to the School of Computing and Information Technology.

2.3 Software System Attributes

The software system should have the following attributes:

2.3.1 Reliability
The website should be reliable and available 24/7. The website should have a backup system to ensure that data is not lost in case of system failure.

2.3.2 Availability
The website should be available 24/7, with minimal downtime for maintenance and upgrades.

2.3.3 Security
The website should be secure, with appropriate measures to prevent unauthorized access and data breaches. The website should have a login system with appropriate authentication and authorization mechanisms.

2.3.4 Maintainability

The website should be easy to maintain and update. The website should have an administrator panel for updating the content of the website.

2.3.5 Portability

The website should be portable and should run on various web servers without any issues. It should also be compatible with different web browsers and mobile devices.

2.3.6 Performance

The website should be fast and responsive. It should load quickly and handle multiple requests simultaneously without any lag.

2.4 Database Requirements

The website should be database-driven and use MySQL for database management. The database schema should be well-designed and normalized to avoid data redundancy and improve data integrity. The website should use proper database indexing to improve query performance.

SECTION C: SOFTWARE DESIGN DESCRIPTION

1 INTRODUCTION

This Software Design Description (SDD) document provides a detailed description of the design and architecture of the School of Computing and Information Technology website. The purpose of this document is to provide guidance to developers, project managers, and stakeholders about the software design and its implementation. This document describes the system's architecture, components, user interface design, and additional material.

1.1 Design Overview

The School of Computing and Information Technology website is designed to provide information about the school's programs, academics, Departments, staff, and login for students. The website is divided into several pages, including Home, Academics, Departments, Programmes, Login, Staff, and Apply Now. The website's frontend is developed using PHP and Laravel framework, and the backend is developed using MySQL for database management.

1.2 Requirements Traceability Matrix

The following table presents the mapping between the system requirements and their corresponding design components.

Requirements	**Components**
Home Page	3.1
Academics	3.2
Departments	3.3
Programmes	3.4
Login	3.5
Staff	3.6
Apply now	3.7

2 SYSTEM ARCHITECTURAL DESIGN:

2.1 Chosen System Architecture

The chosen system architecture for the School of Computing and Information Technology website is a three-tier architecture consisting of presentation, application, and database tiers. The presentation tier contains the user interface, the application tier handles the business logic, and the database tier manages the data storage.

2.2 Discussion of Alternative Designs

Alternative designs considered for the School of Computing and Information Technology website include the two-tier architecture and the n-tier architecture. The two-tier architecture is not suitable as it combines the presentation and application tiers, making the system less flexible and harder to maintain. The n-tier architecture was not chosen as it would add unnecessary complexity to the system.

2.3 System Interface Description

The system interface for the School of Computing and Information Technology website includes the following components:

- Presentation Layer: This layer consists of the user interface that users interact with to access the system's functionality.
- Application Layer: This layer handles the business logic and includes the PHP and Laravel framework.
- Data Access Layer: This layer manages the data storage and includes the MySQL database.

3 DETAILED DESCRIPTION OF COMPONENTS

In this section, we provide a detailed description of each of the components of the website, including their functionality and implementation details.

3.1 Component-1: Home Page

The Home Page is the main page of the website, which provides information about the current activities of the School, how to apply with a link to Apply Now, message from the director, and other relevant information. It is implemented using the HomePageController and HomePage model. The HomePage model is responsible for retrieving the information about the school and its activities from the database.

When a user visits the Home Page, the HomePageController retrieves the information about the school and its activities from the HomePage model and passes it to the Home view. The Home view then displays the information in a user-friendly format.

The Home Page also includes a form that allows users to apply to the school. When the user submits the form, the ApplyNowController processes the application and updates the database accordingly.

3.2 Component-2: Academics Page

The Academics Page provides information regarding various academic activities including and not limited to boot camps, Cisco academies, and related information as well as Microsoft-related courses that students undertake to enrich the programs they are taking. It is implemented using the AcademicsPageController and AcademicsPage model. The AcademicsPage model is responsible for retrieving the academic information from the database.

When a user visits the Academics Page, the AcademicsPageController retrieves the academic information from the AcademicsPage model and passes it to the Academics view. The view then displays the academic information in a user-friendly format.

The Academics Page also includes information about the department of IT, which is stored in the Departments table of the database.

3.3 Component-3: Departments Page

The Departments Page allows users to view the details of each department within the School of Computing and Information Technology. The page provides an ecosystem for viewing all the details per the department selected.

The Departments Page is implemented using the DepartmentsPageController. The page displays a dropdown menu that lists all the available departments within the School of Computing and Information Technology.

Once the user selects a department from the menu, the page displays the details of that department, including the department name, head of department, and other relevant information. The details are retrieved from the database and displayed using the view file.

The Departments Page also provides links to other pages related to the selected department, such as the department's academic programs and staff members. The page provides a comprehensive and user-friendly way for users to view the details of each department within the School of Computing and Information Technology.

3.4 Component-4: Programmes Page

The Programmes Page lists the various programs of the School, which are stored in the Programs table of the database. It is implemented using the ProgrammesPageController and

ProgrammesPage model. The ProgrammesPage model is responsible for retrieving the program information from the database.

When a user visits the Programmes Page, the ProgrammesPageController retrieves the program information from the ProgrammesPage model and passes it to the Programmes view. The view then displays the program information in a user-friendly format.

3.5 Component-5: Login Page

The Login Page allows registered users to log in to the website. It is implemented using the LoginController and Login model. The Login model is responsible for verifying the user's credentials and granting access to the website if the credentials are correct.

When a user visits the Login Page, the LoginController displays the Login view, which includes a form for the user to enter their username and password. When the user submits the form, the Login model verifies the credentials and redirects the user to the appropriate page based on their access level.

3.6 Component-6: Staff Page

The Staff Page lists the staff members of the School, which are stored in the Users table of the database. It is implemented using the StaffPageController and the staff.blade.php view file. The StaffPageController retrieves the necessary data from the database and passes it to the view file for rendering.

The view file displays the staff members in a table format, with columns for their names, roles, and email addresses. The table is sortable by clicking on the column headers. Pagination is also implemented to display a limited number of staff members per page.

The StaffPageController also handles the search functionality on the staff page. Users can search for staff members by their names or roles. The controller retrieves the search query from the user and queries the database for staff members that match the query. The search results are then passed to the view file for rendering.

The staff page also includes a link to a detailed profile page for each staff member. Clicking on a staff member's name in the table takes the user to their profile page. The profile page is implemented using the StaffProfileController and the staff_profile.blade.php view file. The StaffProfileController retrieves the necessary data from the database for the selected staff member and passes it to the view file for rendering.

The staff profile page displays detailed information about the staff member, including their name, role, email address, and a brief bio. The page also displays any courses or programs that the staff member is involved in, as well as any research or publications they have contributed to. The profile page includes a back button that takes the user back to the staff page.

3.7 Component-7: Apply now Page

The Apply Now page is where prospective students can apply to join the School of Computing and Information Technology. The page is implemented using the ApplyNowController and the apply_now.blade.php view file.

The ApplyNowController retrieves the necessary data from the database, such as the list of available programs and any application requirements, and passes it to the view file for rendering. The view file displays the available programs in a dropdown menu, along with any required documents or information that the user needs to provide.

The user fills out the application form with their personal information, academic background, and any other relevant details. The form includes fields for the user to upload any required documents, such as transcripts or recommendation letters.

Once the user submits the application, the ApplyNowController processes the data and stores it in the database. The user is then redirected to a confirmation page that confirms their application has been received and provides them with any additional instructions or next steps.

4 USER INTERFACE DESIGN

4.1 Description of the User Interface

The user interface for the website has been designed to be simple and easy to use. Each page contains a header, footer, and menu bar, as specified in the project requirements. The header contains the school logo and a navigation menu, while the footer contains contact information and links to social media pages.

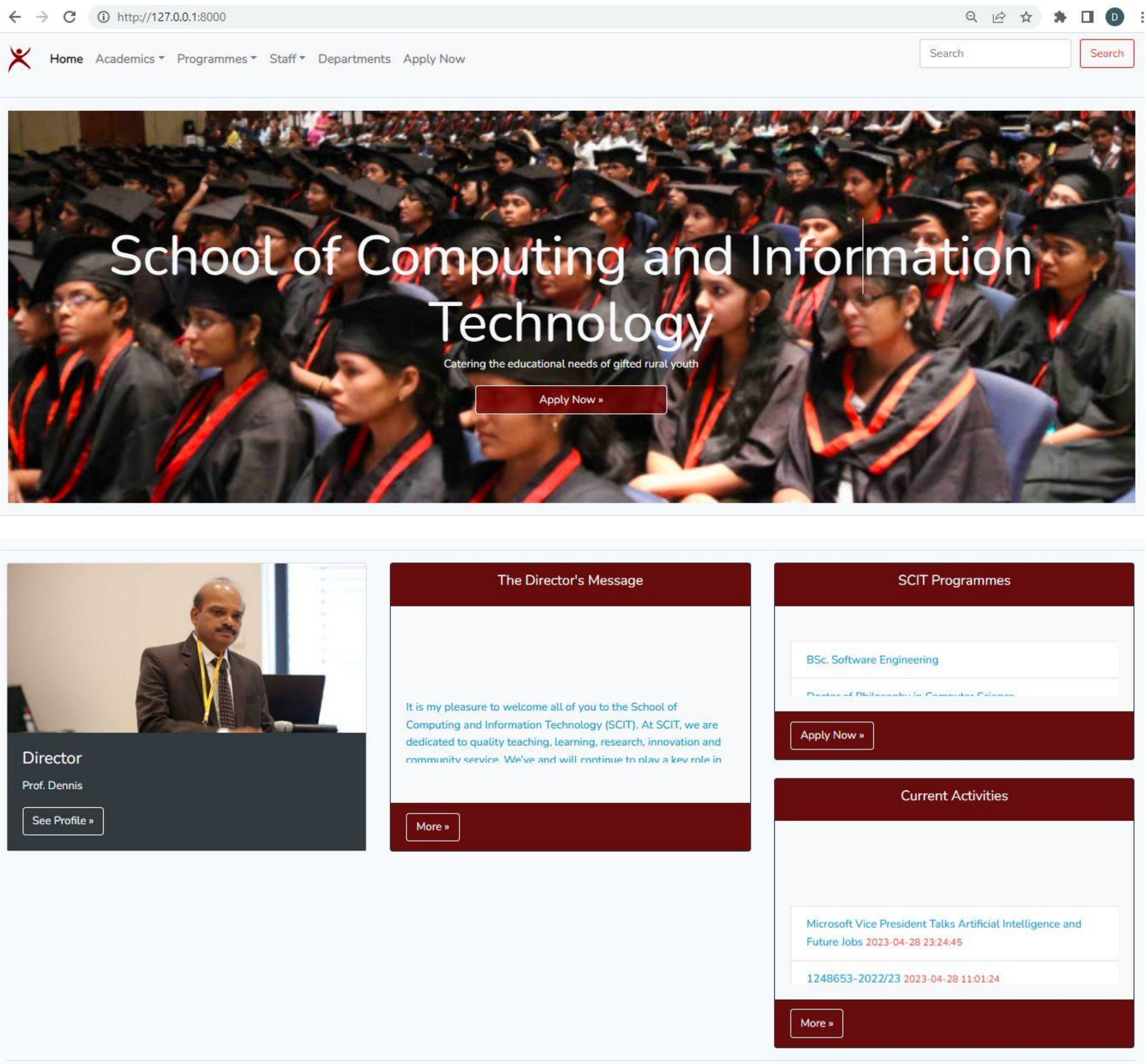
Each page has a unique layout, with appropriate images and text, as specified in the project requirements. The pages are designed using responsive design principles, ensuring that they look good on all devices, including desktops, laptops, tablets, and smartphones.

4.1.1 Screen Images

The following are screen images of the different pages of the website:

Front End

- Home Page



School of Computing and Information Technology

Catering the educational needs of gifted rural youth

The Director's Message

It is my pleasure to welcome all of you to the School of Computing and Information Technology (SCIT). At SCIT, we are dedicated to quality teaching, learning, research, innovation and community service. We've and will continue to play a key role in

SCIT Programmes

BSc. Software Engineering

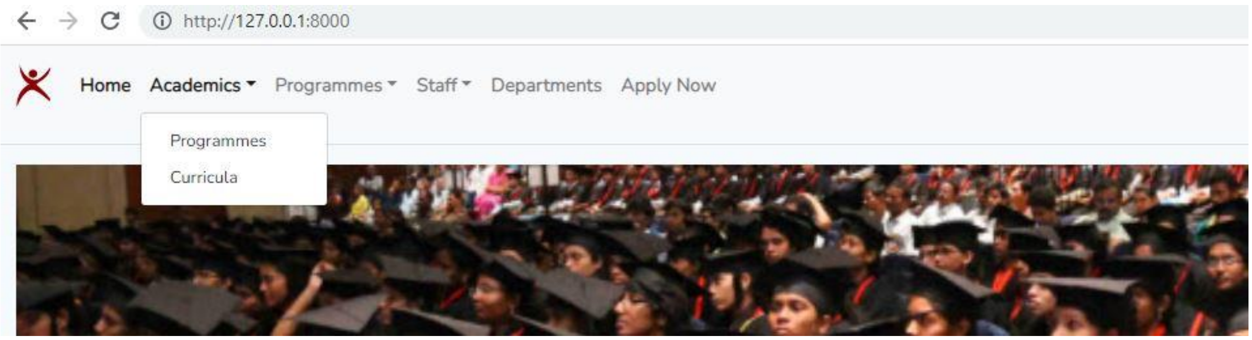
Doctor of Philosophy in Computer Science

Current Activities

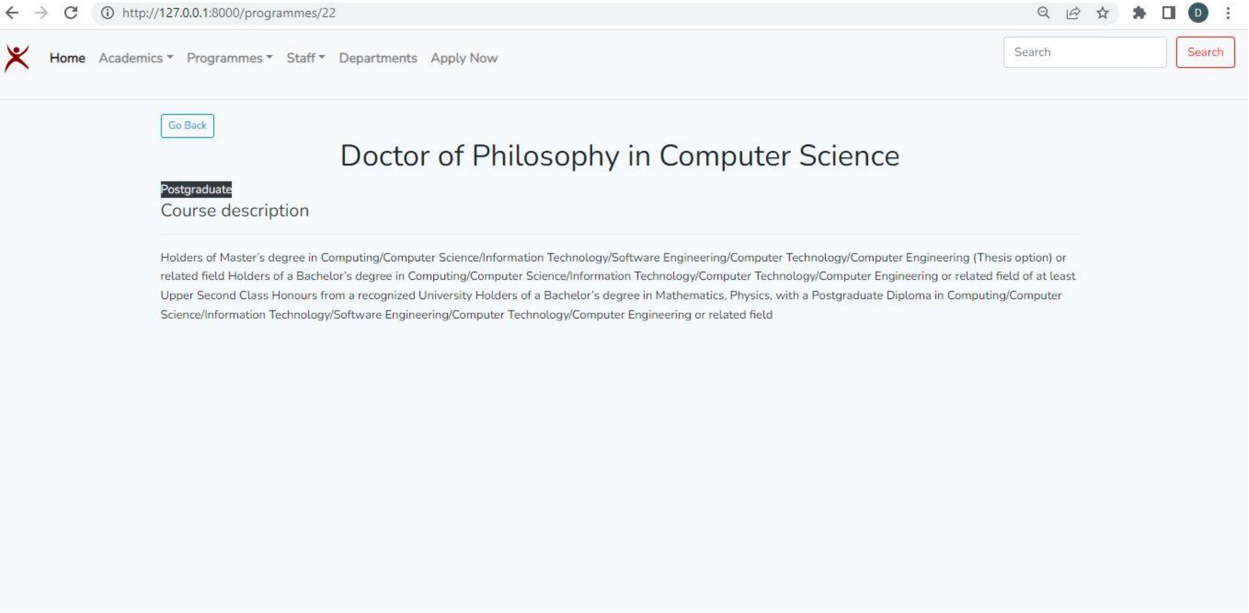
Microsoft Vice President Talks Artificial Intelligence and Future Jobs 2023-04-28 23:24:45

1248653-2022/23 2023-04-28 11:01:24

- Academics Page



- Departments Page



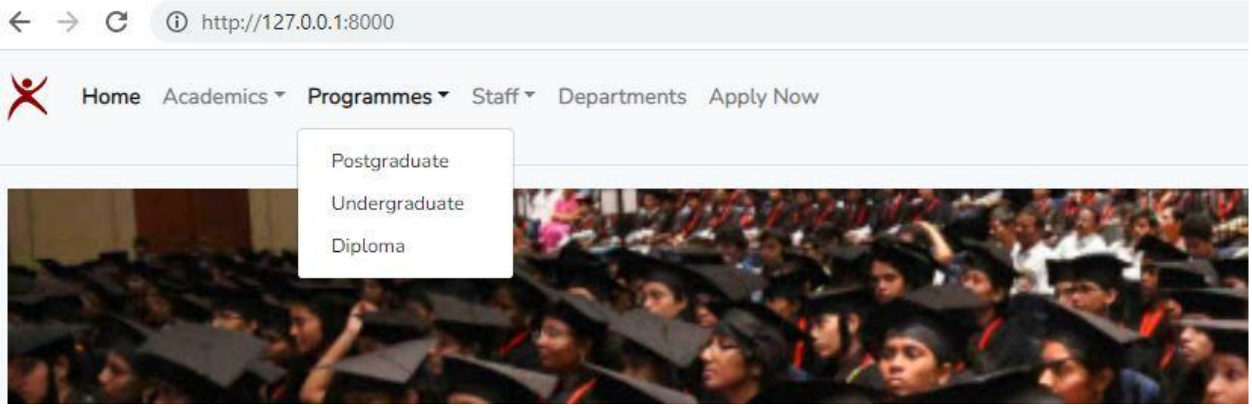
Doctor of Philosophy in Computer Science

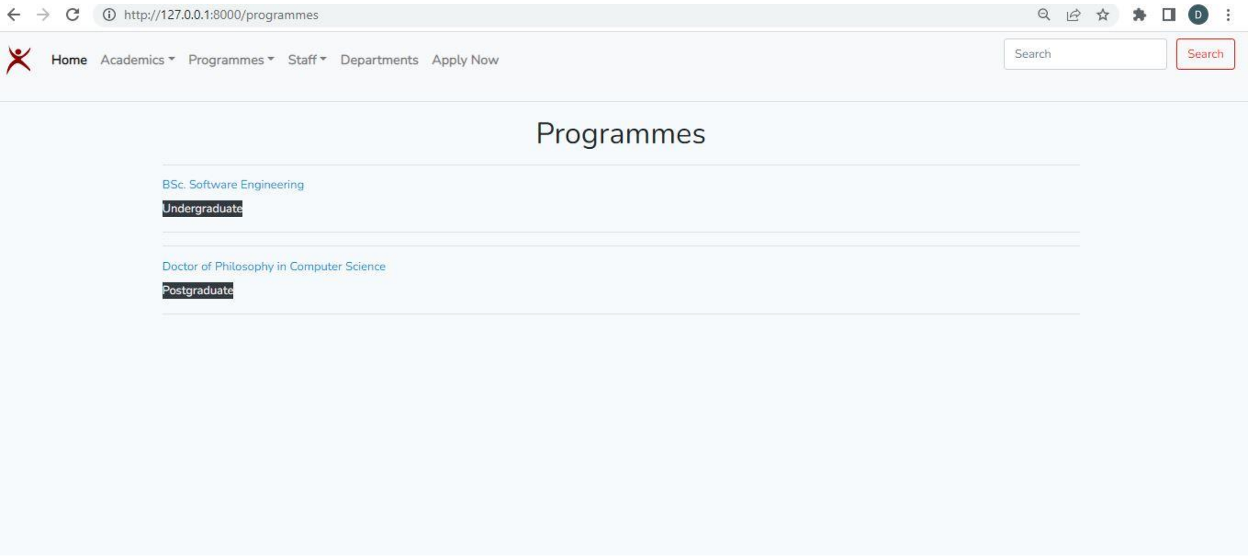
Postgraduate

Course description

Holders of Master's degree in Computing/Computer Science/Information Technology/Software Engineering/Computer Technology/Computer Engineering (Thesis option) or related field Holders of a Bachelor's degree in Computing/Computer Science/Information Technology/Computer Technology/Computer Engineering or related field of at least Upper Second Class Honours from a recognized University Holders of a Bachelor's degree in Mathematics, Physics, with a Postgraduate Diploma in Computing/Computer Science/Information Technology/Software Engineering/Computer Technology/Computer Engineering or related field

- Programs Page [insert image of the Programs Page]





Programmes

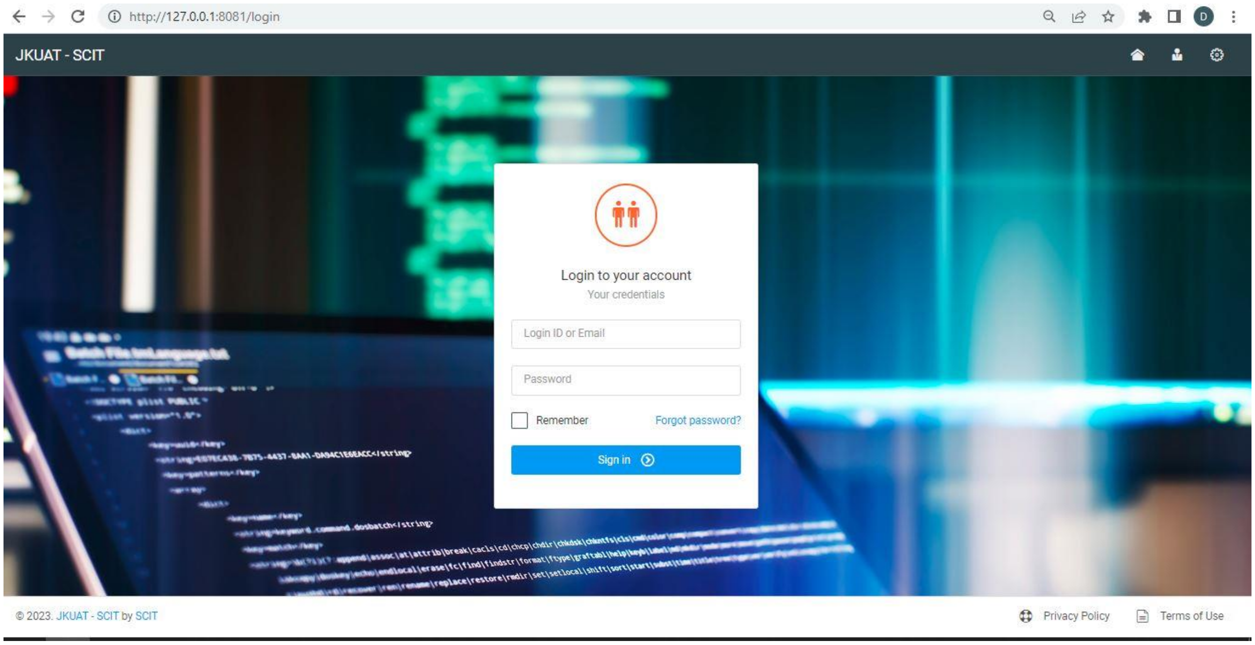
BSc. Software Engineering

Undergraduate

Doctor of Philosophy in Computer Science

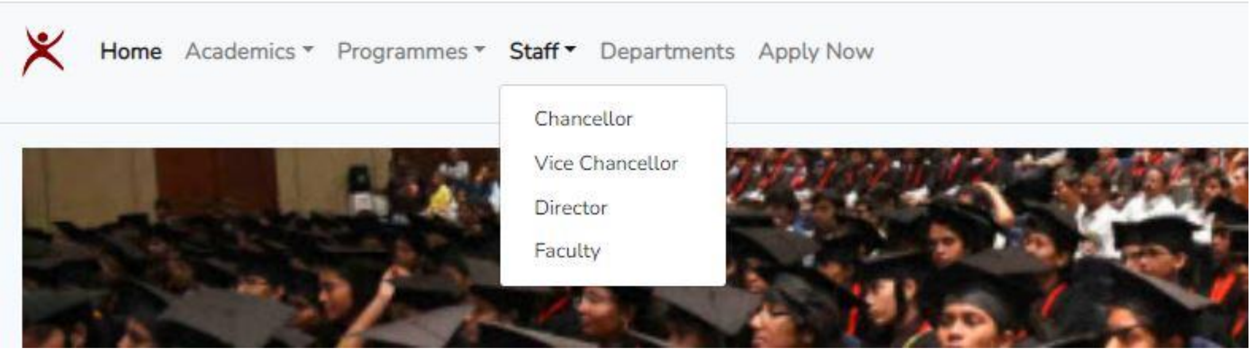
Postgraduate

- Login Page

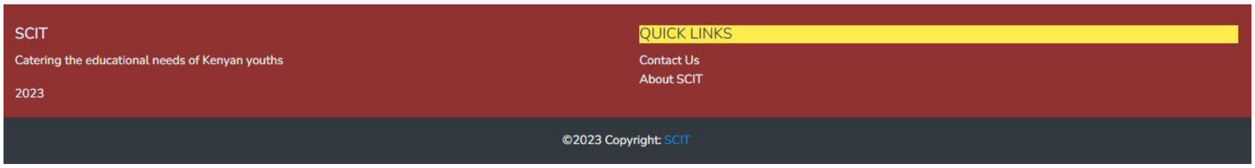


© 2023. JKUAT - SCIT by SCIT

- Staff Page



- Footer

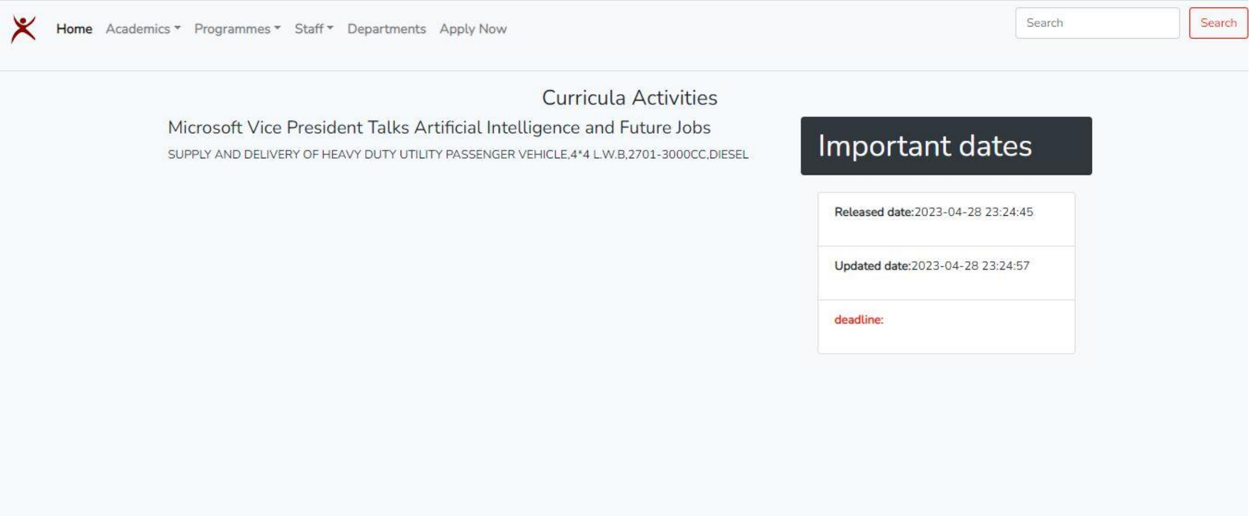


SCIT

Catering the educational needs of Kenyan youths

2023

©2023 Copyright: SCIT



Curricula Activities

Microsoft Vice President Talks Artificial Intelligence and Future Jobs

SUPPLY AND DELIVERY OF HEAVY DUTY UTILITY PASSENGER VEHICLE,4*4 L.W.B,2701-3000CC,DIESEL

Important dates

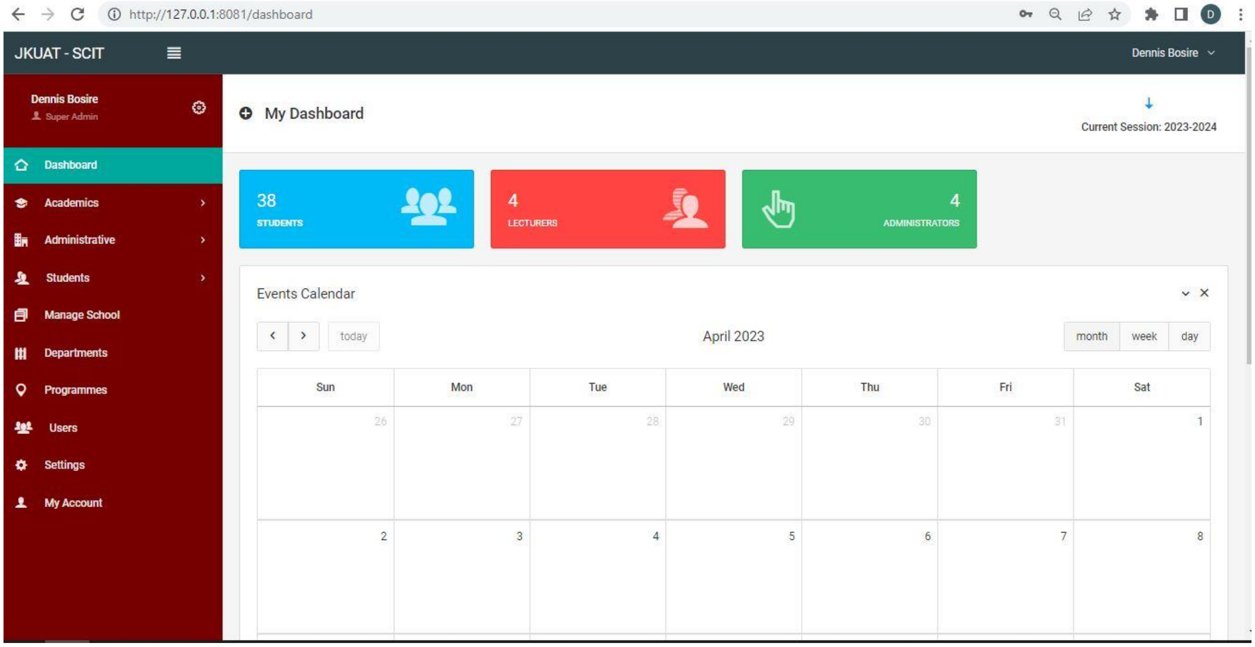
Released date:2023-04-28 23:24:45

Updated date:2023-04-28 23:24:57

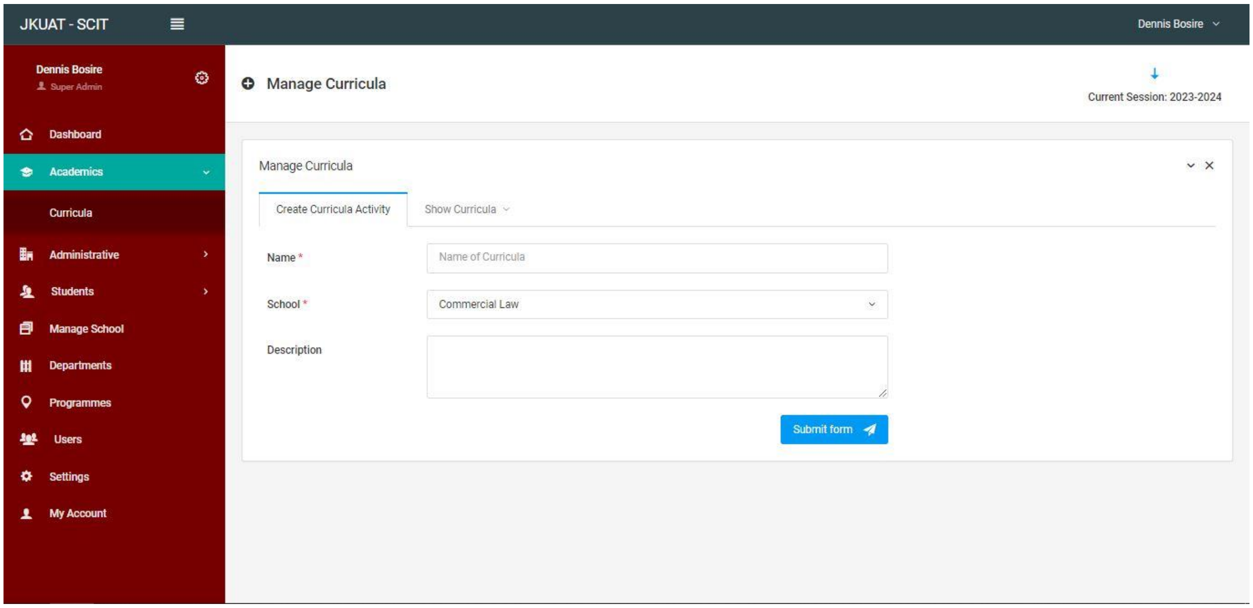
deadline:

Back End

Administrator dashboard for administrative edits to the system will have the features as shown in the screen captures below.



Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	31	1
2	3	4	5	6	7	8

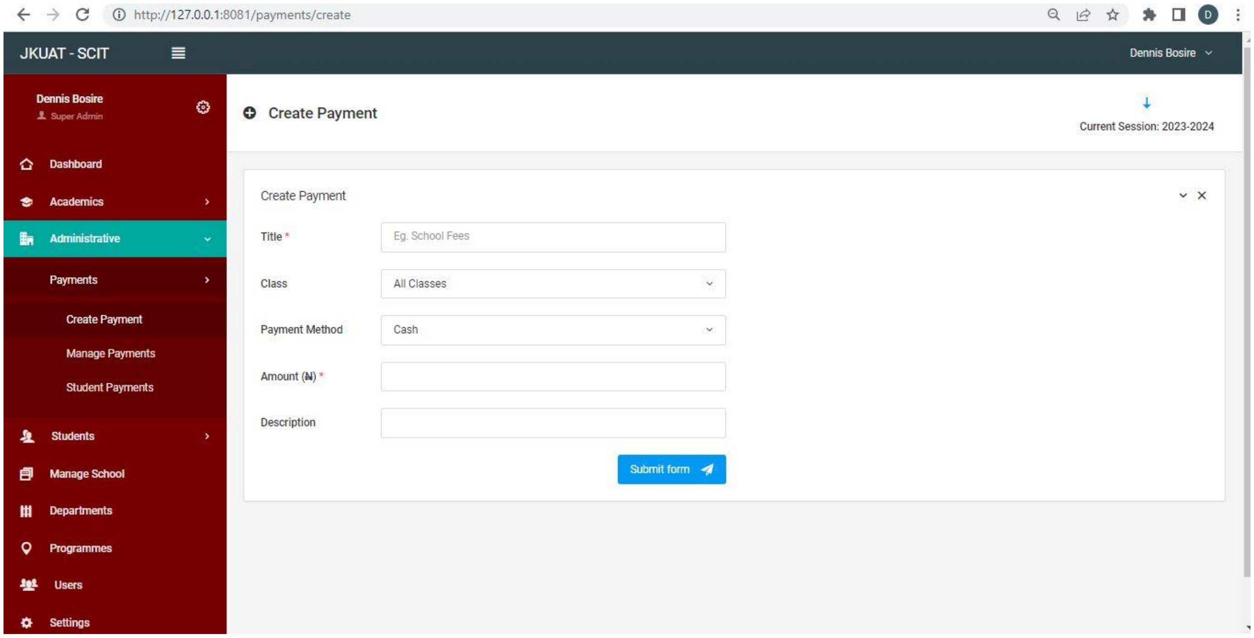


Manage Curricula

Current Session: 2023-2024

Manage Curricula

Name *	Name of Curricula
School *	Commercial Law
Description	

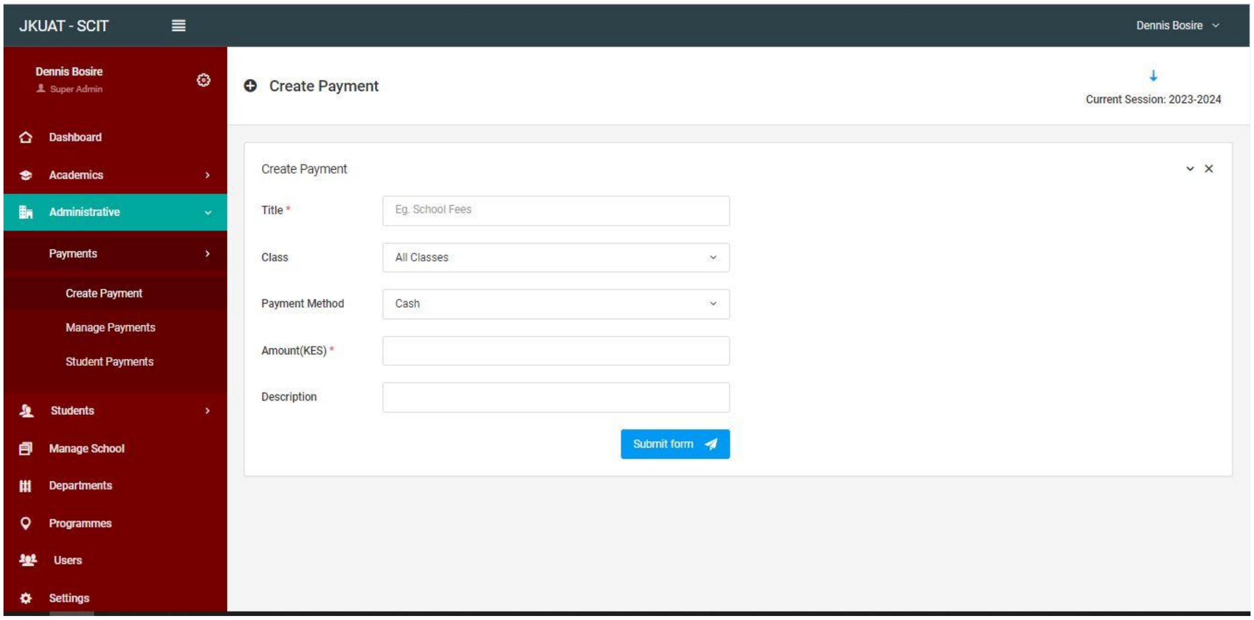


Create Payment

Current Session: 2023-2024

Create Payment

Title *	Eg. School Fees
Class	All Classes
Payment Method	Cash
Amount (₦) *	
Description	



Create Payment

Current Session: 2023-2024

Create Payment

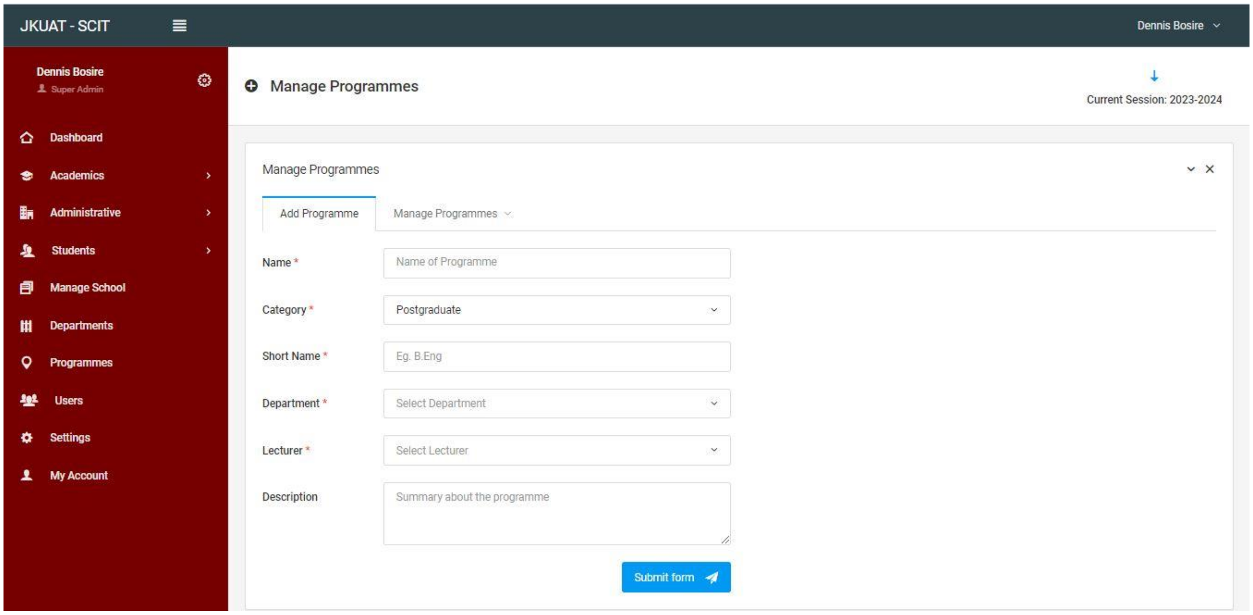
Title * Eg. School Fees

Class All Classes

Payment Method Cash

Amount(KES) *

Description



Manage Programmes

Current Session: 2023-2024

Manage Programmes

Name * Name of Programme

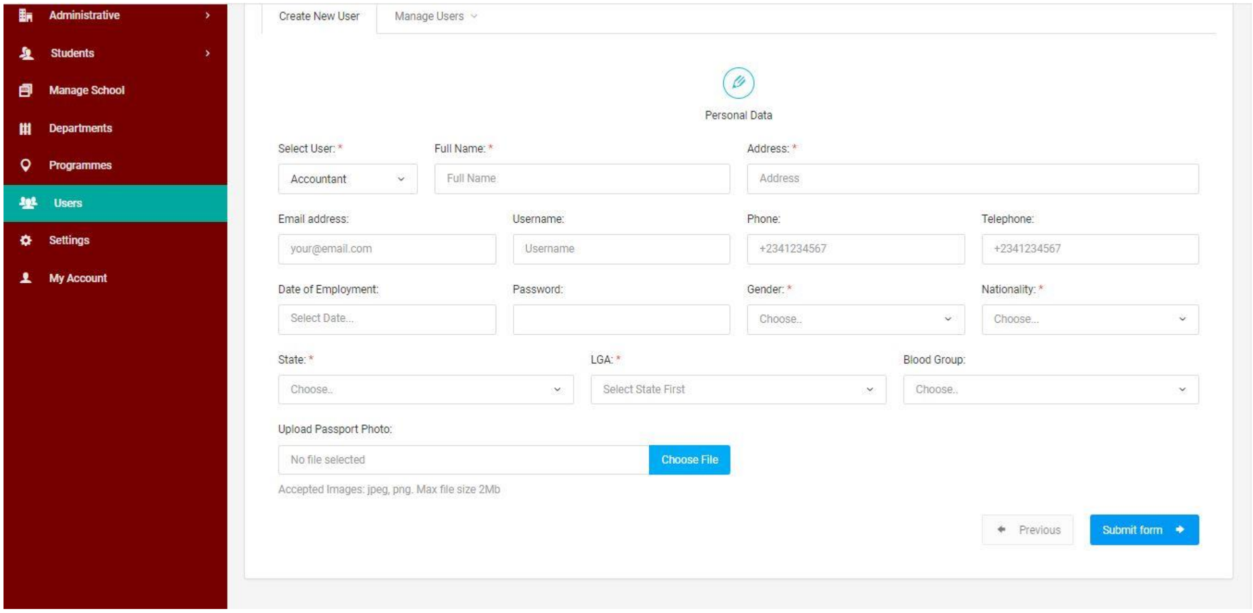
Category * Postgraduate

Short Name * Eg. B.Eng

Department * Select Department

Lecturer * Select Lecturer

Description Summary about the programme



Select User: * Accountant

Full Name: * Full Name

Address: * Address

Email address: your@email.com

Username: Username

Phone: +2341234567

Telephone: +2341234567

Date of Employment: Select Date...

Password:

Gender: * Choose..

Nationality: * Choose...

State: * Choose..

LGA: * Select State First

Blood Group: Choose..

Upload Passport Photo: No file selected

Accepted Images: jpeg, png. Max file size 2Mb

4.1.2 Objects and Actions

The following objects and actions are used in the user interface design:

- Navigation menu: allows users to navigate between different pages of the website.
- Links: allow users to access additional information, such as application forms or contact information.
- Forms: allow users to submit data to the website, such as login credentials or payment information.
- Buttons: used to trigger actions, such as submitting a form or navigating to a new page.
- Images: used to display relevant images, such as the school logo or pictures of the campus.

5 DATABASE ARCHITECTURE

The website uses a MySQL database to store information about the school, its programs, and payment methods. The database consists of the following tables:

- users: stores user information, such as usernames and passwords.
- payment_methods: stores information about payment methods, such as bank account numbers and M-PESA numbers etc.
- programs: stores information about the school's programs, such as program names and descriptions.
- academics: stores information about the school's academic activities, such as boot camps and Cisco academies.
- departments: stores information about the school's departments, such as the department of IT.

The database architecture is designed to be efficient and scalable, allowing for future expansion of the website as needed.

SECTION D: ADDITIONAL MATERIAL

GLOSSARY

Term	**Definition**
Laravel	A free, open-source PHP web application framework used for web development
MySQL	An open-source relational database management system used for storing and managing data
Front-end	The part of a website that the user interacts with, including the user interface and visual design
Back-end	The part of a website that is responsible for server-side processing and database management
PHP	Hypertext Preprocessor
UI	User Interface
UX	User Experience
Software requirements specification	A document that outlines the functional and non-functional requirements for a software project
Software design document	A document that describes the software architecture and design of a software project
User manual	A document that provides instructions for using a software application
Quality assurance	The process of ensuring that a software project meets its intended quality standards
Project management	The process of planning, organizing, and controlling resources to achieve specific goals within a specified timeframe

REFERENCES

1. ISO/IEC/IEEE 26515:2012, Systems and software engineering -- Developing user documentation in an agile environment. https://www.iso.org/standard/53054.html

2. Brammer, C. & Ervin, N. (1999). Bridging the Gap: A Case Study of Engineering Students, Teachers, and Practitioners. In Proceedings of the 1999 Professional Communication Conference. IEEE.

3. Laravel. (2021). Laravel - The PHP Framework for Web Artisans. Retrieved from https://laravel.com/

4. MySQL. (2021). MySQL :: MySQL Documentation. Retrieved from https://dev.mysql.com/doc/