2022-07

Development of the web-based data-driven university information management system (UIMS) for inter-university council for east Africa

Nshimiye, Abel

NM-AIST

DEVELOPMENT OF THE WEB-BASED DATA-DRIVEN UNIVERSITY INFORMATION MANAGEMENT SYSTEM (UIMS) FOR INTER-UNIVERSITY COUNCIL FOR EAST AFRICA (IUCEA)

Abel Nshimiye

A Project Report Submitted in Partial Fulfillment of the Requirements of the Award the Degree of Master of Science in Embedded and Mobile Systems of the Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

July 2022

ABSTRACT

A significant challenge faced by the Inter-University Council for East Africa (IUCEA) is the lack of a common Higher Education Information System (HEIS) for the East African Community (EAC) to harmonize the region's education and training system. The system has to manage data about the academic program, universities, research, and human capital respectively developed in different modules. Both EAC's citizens and IUCEA need statistics regarding this information and complex questions these data may answer. Traditional high education management is not only costly but also ineffective. In this work, a web-based data-driven University Information Management system (UIMS) for IUCEA is designed to effectively manage university information concerning the academic life cycle, assets, finance, and human resource. In addition, a university application portal to help high education institutions is provided to apply for being among all degrees awarding high education. This system is the second module of the East Africa Community High Education Information System (EAC HEIS); it has been developed using agile software development and web technology such as RESTful API, React, Django, and MySQL. UIMS for IUCEA will significantly impact the IUCEA's day-to-day operation and receive recognition from the surfer. It will improve the efficiency of many processes.

DECLARATION

I, Abel Nshimiye, do hereby declare to the Senate of the Nelson Mandela African Institution of Science and Technology that this project report is my original work and that it has neither been submitted nor being concurrently submitted for a degree award in any other institution.

Abel Nshimiye		14.07.2022
Name of Candidate	**Signature**	**Date**

The above declaration is confirmed by:

Dr. Mussa Dida Ally		
Name of Supervisor 1	**Signature**	**Date**

Dr. Silas Steven Mirau		5.4.2022
Name of Supervisor 2	**Signature**	**Date**

COPYRIGHT

CERTIFICATION

The undersigned certify that they have read and hereby recommend for acceptance by The Nelson Mandela African Institution of Science and Technology, a project report titled "**Development of the Web-Based Data-Driven University Information Management System (UIMS) For Inter-University Council for East Africa (IUCEA)**" in partial fulfillment of the requirements for the degree of Master of Science in Embedded and Mobile Systems, Mobile Systems specialty of the Nelson Mandela African Institution of Science and Technology.

Dr. Mussa Dida Ally

Name of Supervisor 1	**Signature**	**Date**

Dr. Silas Steven Mirau  5.4.2022

Name of Supervisor 2	**Signature**	**Date**

ACKNOWLEDGEMENTS

I would want to express my gratitude to God for providing me with the wisdom, bravery, and strength that I need during the course and final project work to accomplish this Master's degree. Sincerely, I thank the Center of Excellence for Information and Communications Technology (ICT) in East Africa (CENIT @ EA) for funding my entire Master's program, including my final project.

I would also want to express my gratitude to my supervisors, Dr. Mussa Dida Ally and Dr. Silas Steven Mirau, for their dedication and unflinching support in writing and assessing the final project.

Last but not least, my thanks go to the industrial supervisor, Mr. Ben Ruhinda, and the entire team of the Inter-University Council for East Africa (IUCEA) for believing in me and encouraging me spiritually, emotionally, and professionally. Finally, I thank my classmates at NM-AIST.

DEDICATION

This work is dedicated to God, my creator, and my source of inspiration, knowledge, comprehension, and wisdom. This work is also dedicated to my father, siblings, and other friends. I will always appreciate all they have done.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS AND SYMBOLS

API	Application Programming Interface
ARIS	Academic Records Information System
ASIS	Assets Information System
AUC	African Union Commission
CRUD	Create Read Update Delete
CSS	Cascading style sheet
CUE	Commission for University Education
DOM	Document Object Model
EACAPIMS	East Africa Community academic program Information Management System
EACHCIMS	East Africa Community Human Capital Information Management System
EACHEIS	East Africa Community high education information system
EACRIMS	East Africa Community Research Information Management System
EACUIMS	East Africa Community-University Information Management System
EAQFHE	East African Qualifications Framework for Higher Education
FINIS	Financial Information System
GCC	Gulf Cooperation Council
gRPC	Google Remote Procedure Call
HEC	Higher Education Council
HTML	HyperText Markup Language
HTTP	Hypertext Transfer Protocol
HURIS	Human Resource Information System
ICT	Information and Communications Technology
IM	Instant Message

IRC	Internet Relay Chat
IUCEA	Inter-University Council for East Africa
MVC	Model Control View
NCHE	National Council for Higher Education
NIS	National Innovation Systems
NPM	Node Package Manager
ORM	Object-Relational mapper
P2P	Peer-to-Peer
RDBMS	Relational Database Management System
RDBMS	Relational Database Management System
RDI	Research Development and Innovation
REST	Representational State Transfer
RPC	Remote Procedure Call
SIMS	Student Information Management System
SMTP	Simple Mail Transfer Protocol
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
TCU	Tanzania Commission for Universities
UIMS	University Information Management System
UML	Unified Modeling Language
URL	Uniform Resource Locator
WWW	World Wide Web

CHAPTER ONE

INTRODUCTION

1.1 Background of the Problem

The EAC is an African region that brings Burundi, Kenya, Rwanda, South Sudan, Tanzania, and Uganda into variant modes of economic partnership. The eventual dream is to achieve a political alliance (Baariu *et al.,* 2019). The EAC as a union of Uganda, Kenya, and Tanzania was established in 1967. The partnership did not take long since it collapsed in 1977, but after a few years, in 1999, it was re-established. Burundi and Rwanda joined the community in 2007, and South Sudan followed in 2016 (Murage *et al.,* 2019).

However, since 1981, when the Arusha Convention treaty was signed, attempts have been made to establish a typical higher education space in East Africa; unfortunately, the collapse of the EAC in 1977 hampered his progress (Ogachi, 2018). Therefore, upon the re-establishment of the EAC, East Africa's political leaders decided to resurrect the IUCEA, which would be entrusted with investigating methods for greater higher education convergence in the region. In addition, the EAC works in concordance with other provincial monetary networks in Africa to advance the plan of the African Union Commission (AUC) advanced education methodology.

In addition, EAC aims to provide elite education to United Nations Member States. The contribution to the development of higher education, professional and affordable training, elimination of equitable gender and wealth gaps, and the achievement of universal access to high-quality higher education is provided by EAC through the IUCEA in the region. The EAC member states signed the treaty to expressly pursue deliberate steps to encourage collaboration in education and training in the community in a cooperative effort to foster the development of higher education. The specific agreement concerns harmonizing education and training systems for comparability and compatibility among partner nations (O'leary, 2017). Through the IUCEA, the EAC developed the East African Qualifications Framework for Higher Education (EAQFHE) in response to the above development. A qualification framework was defined by (O'leary, 2017) as a tool for bringing the training and education systems closer together. The IUCEA, a leading EAC institution for an exemplary common higher education space for a prosperous and sustainable EAC, has been chosen to implement the system.

Furthermore, the IUCEA must develop a tool to harmonize the education and training system in the EAC region. According to (Ng'ingo, 2019), the tool will be a framework that will stipulate the easy mobility of learners and labour in the community. According to the final report on the EACHEIS submitted to the IUCEA by individual consultant Prof. Venansius BARYAMUREEBA in July 2020, The EACHEIS shall integrate four modules:

(i) EAC Academic Programs Information Management System (EACAPIMS); will be used by universities and university colleges to submit curricula for review or approval before being operational and publishing approved curricula.

(ii) EAC Universities Information Management System (EACUIMS) manages information for accredited or licensed Universities and University Colleges.

(iii) The EAC Human Capital Information Management System (EACHCIMS) should derive its information from the National Human Capital Information Management Systems.

(iv) EAC Research Information Management System (EACRIMS) should derive its information from the National Research Information Management Systems.

The EACHEIS uses data from six countries to involve legal analysis on data sharing in the region. It was established that the Institutional or University legal framework should conform to the National legal framework, which should work with the EAC legal framework. The EAC legal framework shall ensure that Partner States have a shared enabling environment to operate. Likewise, the national legal framework shall ensure that the Universities or Institutions in each Partner State have a shared enabling environment to work. The institutional, legal framework shall address the specific needs of each institution without contradicting neither the national legal framework nor the EAC legal framework.

Given the above legal analysis, the final report on the Status of the EAC HEIS provides two Frameworks about EAC HEIS, respectively, presented in the following figures.

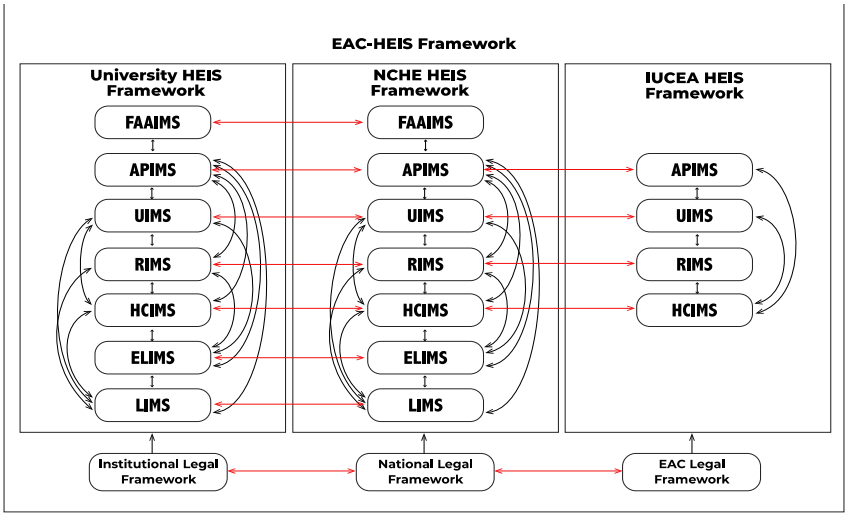


Figure 1: East Africa community high education information system framework – where EAC-HEIS derives data or information from NCHE HEIS

The framework (Fig. 1) ensures data is validated at the national level by the national council or commission for higher education before sharing it with IUCEA for the EAC-HEIS.

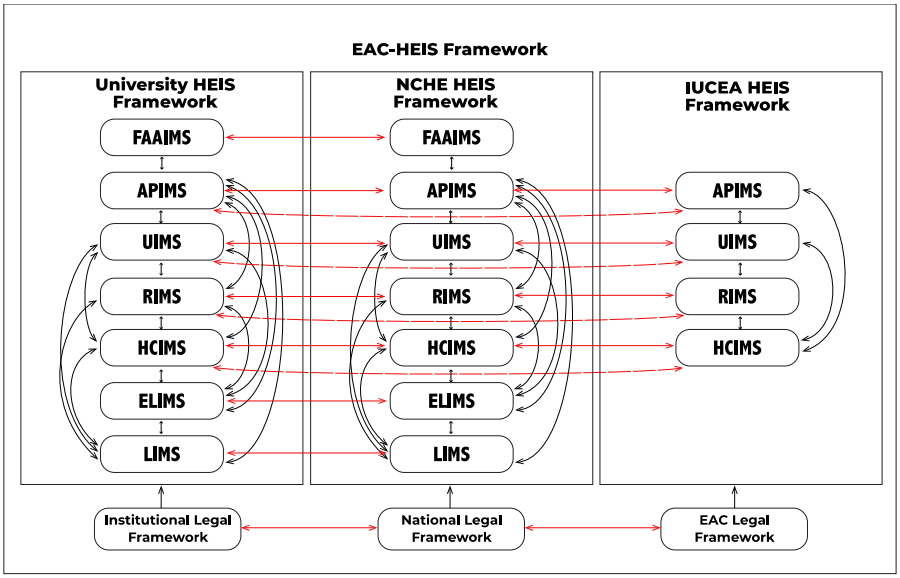


Figure 2: East Africa community high education information system framework – Where EAC-HEIS derives data/information from NCHE HEIS or/and University HEIS

The framework (Fig. 2) provides flexibility to IUCEA to either collect data from national councils or commissions for higher education or from universities and university colleges, or both.

The model technology has been adopted to manage the universities' education and the national council or commission for higher education within the EAC partner states. According to the final report on the Status of the EAC HEIS, as described in Table 1, four commissions or councils for higher education had a University Information Management system (UIMS) that manages information for accredited and licensed Universities operating among six in a Partner States.

Table 1: National council or commission for higher education available in the EAC region

	NCHE	UIMS
Burundi	National Commission for Higher Education (NCHE)	0
Kenya	Commission for University Education (CUE)	1
Rwanda	Higher Education Council (HEC)	1
South Sudan	Directorate of higher and tertiary education, Ministry of Higher Education, Science and Technology	0
Tanzania	Tanzania Commission for Universities (TCU)	1
Uganda	National Council for Higher Education (NCHE)	1
EAC	(Total)	4

The same report recommends that EACHEIS have the UIMS that manages the academic life cycle, human resources, assets, and finances. The academic life cycle shall include admission data, student enrolment, graduates, and drop-outs. Human resources shall consist of academic, administrative, and support staff. Assets shall contain information about facilities and infrastructure. Finances shall collect information about revenue and expenditure data. The system should also include University governance organs (structure, membership, powers, and functions), University policies, statutes, rules, and procedures. Also, UIMS should have a module that captures all information required to assess an application for license or accreditation or charter for proposed or new universities. That information needed for the UIMS comes from the four subsystems the HEIS institution should have, which are the Human

Resource Information System (HURIS), Academic Records Information System (ARIS), Assets Information System (ASIS), and Financial Information System (FINIS) (Tatnall *et al.*, 2011). The table below shows the Status of existing information systems within 49 institutions among 133 degrees awarding higher education in the survey done within five countries over six countries of the East African Community.

Table 2: Existing information systems within institutions will contribute to implementing the EAC UIMS

S/N	Country	ARIS	HURIS	FINIS	ASIS	Nº Institutions Surveyed
1	Burundi	3	3	3	2	8
2	Kenya	11	11	11	8	11
3	Rwanda	7	7	7	6	7
4	Tanzania	10	8	5	3	10
5	Uganda	13	12	12	6	13
6	EAC	44	40	38	25	49

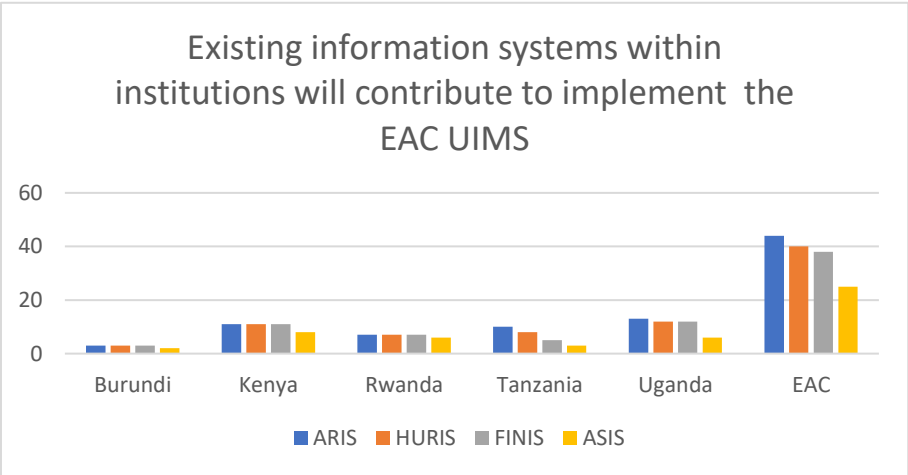


Figure 3: Existing information systems within institutions will contribute to implementing the EAC UIMS

The analysis has allowed us to design and implement the web-based data-driven university information management system for IUCEA, adopting the second framework that collects data from national councils or commissions for higher education and all degrees-awarding high education. In the final report about EACHEIS, this system has been called the East Africa Community University Information Management System (EACUIMS).

1.2 Statement of the Problem

The world's most valuable resource is no longer oil but data (Economist, 2017). Moreover, the advances in technology create new demands. As a result, Universities are constantly searching for new insights from data to generate strategies to meet these new demands (Daniel, 2017).

However, in higher education in the East African Region, there is a need for data-driven information systems to accommodate new economic, regional, national and international demands. Moreover, since the higher education system in East Africa has been undergoing revolution from the colonial to when African countries signed the Arusha convention to facilitate staff and student mobility through the continent, and the emergence of the European Higher Education Area (EHEA) through the Bologna process.

The East Africa Community needs a straightforward way to gather information about the academic life cycle, human resources, assets, and finance from all degree-awarding higher education within the EAC region. Then the EAC through the IUCEA will use that information to make decisions concerning the evolution of high education in the EAC region. That will be possible by developing a web-based data-driven University Information Management System (UIMS) for IUCEA, the second module of EACHEIMS.

1.3 Rationale of the Study

The conviction that education is one of the most powerful and proven engines for sustainable development is reaffirmed by achieving inclusive and high-quality education for all. The purpose of education is to achieve the success of a society which we can define as measuring how successfully the community meets the needs of its members (Bass, 1997).

However, there is a lack of a Common Higher Education Area (HEA) for the East African Community (EAC) in awareness of the need to harmonize the region's education and training

systems. A tool needed by the Inter-University Council for East Africa (IUCEA) as the leading EAC institution for an exemplary common higher education space.

This project aims to develop the web-based data-driven University Information Management System (UIMS) for the IUCEA, the second module of the East African community High Education Information System (EAC HEIS).

1.4 Objectives

1.4.1 Main Objective

To develop the web-based data-driven University Information Management System (UIMS) for IUCEA.

1.4.2 Specific Objectives

(i) To collect requirements for developing the proposed system.

(ii) To develop the proposed web application and integrate it with the first module's database.

(iii) To test and validate the performance of the developed system.

1.5 Research Questions

This project aims to respond to the questions below:

(i) What is required to develop the web-based data-driven University Information Management System (UIMS) for IUCEA and integrate it with the first module of EAC HEIS?

(ii) What are the features for developing the proposed web application system to manage a massive volume of data?

(iii) Is the developed system working as anticipated?

1.6 Significance of the Study

This project will contribute to the development of the EACHEIS, which will be all about electronic resource planning, integrated education management information system, and

knowledge management in the EAC region. Furthermore, it will help address suitable solutions to the high education questions in the EAC region.

1.7 Delineation of the Study

Developing the UIMS for the East African Community involves writing code and a lot of work and deep research about its implementation. Even if the report concerns that the proposed system has already specified types of data the system needs to handle, the project's scope has remained huge. First, to decide the framework to collect data among the two proposed by the report. To determine if the data collected will be stored in aggregated or disaggregated format. All decisions should be upon a study done in the region since the report did not decide; instead, the possible ways have been proposed.

Resource limitations such as lack of time, small budget, delay of the budget, and lack of experience have been the leading cause of not putting live the developed system. The period of six months and the Corona-Virus disease 2019 (COVID-19) situation in Kampala, Uganda, has been the leading cause of not realizing all required on the system analysis stage. Furthermore, the project's budget was small and has delayed for two months which caused the delay of many activities. Finally, a lack of experience working on big projects has been time-consuming since much time has been spent learning instead of working on the project.

The project's background has been explained in Chapter One. The project objectives and questions have been identified, and the value of such a project argued. The limitations of the project have also been discussed. Chapter Two reviews the existing literature to determine the current system and the critical value our system will have. Chapter three discusses the materials and methods needed to foster the proposed system. Finally, the results and discussion are reported and discussed in Chapter four.

CHAPTER TWO

LITERATURE REVIEW

2.1 Overview of High Education in Africa

Education is vital for the economic development of any country. No country will guarantee endogenous and property development while not adequate education and analysis establishments provide an important mass of mean and educated individuals (Astakhova *et al.,* 2016). Therefore, every economy needs vital education, research, and development.

Education changes depend on time and the environment; it is a socialization process that integrates individuals within a social context. Africa has started the model education and has created universities that date back to the colonial era. The recent ones have been started under the patronage of former colonial powers (Portugal, France, Belgium, and the United Kingdom) (Pellaud *et al.,* 2019). The University of London, for example, has branches in Sierra Leone's Fourah Bay College, the Nigeria's University of Ibadan, the United Kingdom's University of Salisbury, Ghana (the contemporary University of Harare), and Rhodesia's College of Rhodesia (now College of Zimbabwe). University of Yaoundé, Dakar, Brazzaville, and Abidjan are among the French institutions with "African campuses (Alemu, 2011).

With technology advancement, the current trend of university education is characterized by internationalization and globalization of higher education, leading to increased student and staff flows across borders. Domination of English as popular research and instruction, setting up offshore campuses and leveraging advanced technologies to provide education (Ogachi, 2009). The marketplace for university education is highly complex and calls for innovativeness and proper strategic direction.

With this complexity, East African Partner states have come together and identified the need for harmonizing education and training systems in East Africa to promote the free movement of labour within the region. To achieve this, IUCEA was identified as the institution to develop mechanisms and frameworks for ensuring the harmonization agenda. Information at the regional level is required for IUCEA to promote the harmonization of education and training institutions.

2.1.1 Status of High Education in EAC

(i) Burundi

Burundi's population is about 11 million people (Yaya *et al.,* 2020). It has a gross enrolment rate of approximately 4% (World Bank, 2021). The oldest University in Burundi is the University of Burundi (and presently, Burundi has two public Universities and nine private universities) (Wikipedia, 2021).

(ii) Kenya

Kenya's population is about 52 million people (National Council for Population and Development [NCPD], 2020). It has a tertiary, gross enrolment rate of approximately 11% (World Bank, 2021). The oldest University in Kenya is the University of Nairobi which was earlier the University of East Africa (Oketch *et al.,* 2008). Presently, Kenya has 74 institutions, as shown in Table 1 as extracted from Commission for Universities-Kenya.

Table 3: Growth trend in the number of universities licensed in Kenya

University Category	2016	2017	2018
Public Chartered Universities	23	30	31
Public Universities Constituent Colleges	8	4	6
Private Chartered Universities	17	18	18
Private Universities Constituent Colleges	5	5	5
Private Universities with LIA	12	12	14
Registered Private University	1	0	0
Total	**66**	**68**	**74**

(iii) Rwanda

Rwanda's population is about 12 million people, with a tertiary enrolment of about 6.2 % (World Bank, 2021). The University of Rwanda is the oldest in Rwanda. Rwanda has 26 private

universities and two public universities, i.e., the University of Rwanda and the Institute of Legal practice and development (Higher Education Council [HEC], 2021)

(iv) South Sudan

South Sudan's population stands at about 11 million people (World Bank, 2021), with no data publicly available for tertiary enrolment. Presently South Sudan has five public universities and 13 private universities, but only four are recognized (Kuyok, 2017).

(v) Tanzania

Tanzania's population is approximately 58 million people, with a tertiary enrolment rate of approximately 3% (World Bank, 2021). The oldest university in Tanzania is the University of Dar es Salaam, an East African college. Currently, there are 34 fully-fledged universities, 15 university colleges, and 11 university campuses, centers, and institutes (Tanzania Commission for Universities [TCU], 2021).

(vi) Uganda

Uganda's population is approximately 44 million people (World Bank, 2019), with a tertiary enrolment of about 4% (World Bank, 2014). Makerere University in Uganda is the oldest university in Uganda. Currently, Uganda has about 39 private universities and 11 public Universities (Ochwa-Echel, 2016).

2.1.2 Status of Internet Development in EAC

Internet use in the East African Community (EAC) region is without challenges, such as the high cost of broadband (Ayoo, 2009). As a result, East Africa is cut off from global economic and information possibilities provided by the Internet. Simultaneously, the majority of the earth is linked to many underwater fiber optic cables, giving the most cost-effective way to connect. Unfortunately, East Africa is one of the only places in the world without a fiber optic link, forcing it to rely on satellite communication at high costs (Goldstein, 2008)

There have been efforts to bridge this gap. One of them is SEACOM, a Mauritius-based business that has committed $ 650 million to link Djibouti, Kenya, Tanzania, and Mozambique to Madagascar, South Africa, across a distance of 15 000 km. Furthermore, the East African undersea cable system will extend 9900 km with a capacity of 20 gigabits per second, while

the East African maritime system will cover 4887 km from the United Arab Emirates. Mombasa, Kenya is a place where people come together (Ayoo, 2009). Also, internet usage (percentage of the population) among East African Community countries has increased steadily for the past ten years, as shown in Table 4.

Table 4: Internet usage for the past 10 years in EAC

Country	Internet Usage in 2009 (%)	Internet Usage in 2019 (%)
Burundi	0.9	2.661
Kenya	6.1	22.565
Rwanda	7.7	21.768
South Sudan	3.83 (2013 data)	7.977
Tanzania	2.4	25
Uganda	9.78	23.707

2.2 Information and Communication Technology for High Education

Clarkson (2002) categorizes Information and Communications Technology (ICT) as the impact on what is learned, how, when and where students learn. In terms of what is learned, the World Bank recognizes the widespread belief that ICT can empower teachers and students, transforming teaching and learning methods from strong teacher dominance to student-centered. Students will benefit from this transition because it will allow them to enhance their creativity, problem-solving abilities, informational reasoning skills, communication skills, and other thinking skills (World Bank, 2008).

2.2.1 Benefits of using ICT in High Education

Technology has drastically changed higher education; there is now the existence of virtual classrooms and the possibility for eLearning systems in terms of Massive Open Online Courses. In terms of education management, Information Communication and Technology enables people to lay out their educational and professional achievements to provide transparency on qualifications, rewards, and regulated experiences to members of the public

and employers (Qualification Register [QR], 2021).

Information Communication and Technology also promotes students' mobility; someone can email a prospectus university abroad and receive instant communication. In addition, massive Online courses and degree programs are now delivered online without physically needing to be at the university.

2.2.2 Challenges of using ICT in High Education

Barriers to using ICT in University education are extrinsic or intrinsic barriers (Ertmer & Hruskocy, 1999). Access, time, support, resources, and training are examples of extrinsic hurdles, while attitudes, beliefs, behaviors, and resistance are examples of intrinsic barriers (Al-Alwani, 2005). Extrinsic obstacles, on the other hand, are those that are tied to organizations rather than people, whereas intrinsic barriers are those that are related to instructors, administrators, and individuals.

In the work of other scholars, they divided the obstacles into two general categories: Barriers at the school and the level of the teacher. For example, Salehi-Isfahani (2012) categorized the obstacles into two categories: those affecting the person (teacher-level obstacles) and those affecting the establishment (school-level barriers). A teacher's lack of self-confidence is a teacher-level barrier, whereas a school's lack of practical training in tackling technical difficulties is a school-level obstacle.

2.3 Existing System

The study made by Richardson *et al.* (1998), examines the influence of policy environments, system design, and leadership on the performance of state higher education systems, using case studies of seven large and diverse state higher education systems: California, Florida, Georgia, Illinois, Michigan, New York, and Texas. Starting with an overview of the evolution of state higher education governance structures. This study presents and illustrates the differences in system design and policy environments among states. The study presents the state context; characteristics and history of higher education; the design of the higher education-state government interface; and work processes which include the information management, the budget process, program planning and review, and system articulation. This study has reported that the difference between the performance observed across the seven systems is related to system design and policy environments. The lack of direct communication between the system

and universities is the main problem with the functionality of the system. The system got different information that was supposed to do the same thing.

The study made by Cranfield and Taylor (2008) at seven UK higher education institutions focus on knowledge management. Even with the complexity of Higher Education Institutions within the United Kingdom, the institution has insisted the universities in today's knowledge economy with the dichotomy of priorities that aims to provide quality teaching and research activity. This study does not involve the use of ICT in the process of knowledge management which involve the waste of time. The lack of ICT in the process causes hard work in the evaluation of the process since the record is not well-tracked.

Gulf Cooperation Council (GCC) nations have integrated the use of ICT into their education systems, according to the research made by Wiseman and Anderson (2012). Internationally, ICT-based education has been a driver for the development of national innovation systems (NIS) and RDI capacities. The study shows that the GCC countries have the institutional capacity in ICT which is not used for knowledge development because of limitations imposed by the cultural contexts for education in the Gulf. Finally, this study recommends ways that GCC teachers can use ICT to create national innovation systems and research capacity.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Scope of the Project

The project's scope is to create a web-based platform for IUCEA called *"East African Community University Information Management System (EAC UIMS)"* to manage the information from all degree-awarding higher education in the EAC region. The project will collect and manage the academic life cycle, human resources, finance, and assets of all degree-awarding high education in the EAC region. In addition, the project will allow the application of new universities to work with IUCEA. On top of that, the project will allow EAC Citizens to access public information from universities. Last but not least, the project will work as a tool for harmonization and decision support in the EAC region.

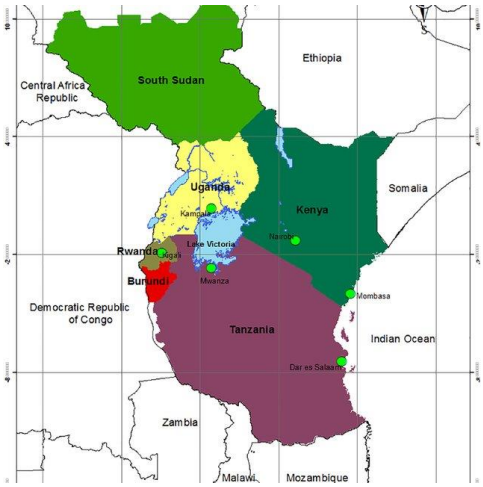


Figure 4: Map of the East African community region

3.2 Data Collection Methods

Methods to collect data may be quantitatively dealing with count data or qualitative methods that consider factors other than numerical values. Both data collection methods, qualitative and quantitative, have been conceded (Allen-Meares & Lane, 1990). During the industrial outreach and internship, data were collected within five weeks, from June to July 2021. Data were available in the final report on the Status of the EAC HEIS submitted by individual consultant Prof. Venansius BARYAMUREEBA in July 2020 to the IUCEA used for the quantitative data collection method. Other techniques used in this study were document analysis, interviews, Joint Application Design (JAD), Brainstorming, and Prototyping as described:

3.2.1 Document Analysis

According to the International Classification of Qualitative Data (ICQD), document analysis is one of the qualitative data gathering methodologies (Bowen, 2009). The three primary types of documents analysis, as defined by Zina (2021) have been used to support decision-making when choosing the type of data EAC UIMS will handle:

(i) Like an open report about the education of deferent national commissions in Africa and worldwide, public records have been used to categorize the data those institutions request from high education institutions.

(ii) Personal documents like newspapers, journals, blogs, and emails have saved costs and time during the development stage.

(iii) Physical evidence about EAC HEIS within the IUCEA, including the final report on the Status of the EAC HEIS submitted by individual consultant Prof. Venansius BARYAMUREEBA in July 2020 to the IUCEA, has been analyzed and used to support the data capturing decision-making stage.

3.2.2 Interviews

An interview defined as a face-to-face conversation between two individuals can be semi-structured, structured, and unstructured (Gill *et al.,* 2008). Interview technic has been used to support the need for new ideas about EAC UIMS; unstructured interviews have been organized, involving different IUCEA employees from different departments. The main objective was to understand how IUCEA works and add value to the developed system.

3.2.3 Joint Application Design (JAD)

Chuck Morris and Tony Crawford from IBM came up with the Joint Application Design (JAD). Collaborative workshops, known as JAD sessions, are used to design and create applications with input from the client or end-user (Davidson, 1999).

Joint application design sessions have been organized involving several employees within IUCEA and other experts in education from outside of the institution. In addition, the opening session has been organized between the ICT department and other facilitator departments of the IUCEA. Furthermore, the opening session has organized several meetings between the

IUCEA's ICT department and each National or Commission for High Education within EAC. However, unfortunately, all the meetings did not happen because of the COVID-19 pandemic.

Weekly online JAD session has been organized between the ICT department and other education experts. During those JAD sessions, the ICT department presented the progress of the EAC UIMS to the Executive secretary of IUCEA, who also contribute to the analysis stage of the system.

3.2.4 Brainstorming

Brainstorming is defined as generating ideas and sharing knowledge (Rawlinson, 2017). This method has been involved in all JAD sessions organized to develop the specific designs and requirements for the system.

3.2.5 Prototyping

Prototyping defines as an experimental method wherein design teams put into effect thoughts into tangible bureaucracy from paper to digital (Budde *et al.*, 1992). During the prototyping stage, Adobe XD has been used for the startup design. Figure 5 shows the startup design of the public dashboard.

Figure 5: Prototype view of the public dashboard (Adobe XD product)

3.3 Requirements Analysis

The requirement engineering process describes that the next stage is requirement analysis after requirements elicitation or gathering. Both functional and non-functional requirements have been analyzed to understand the requirement of the proposed system.

3.3.1 Functional Requirements

Functional requirements describe the developed system's performance task, operation, and service behaviors from the user's perspective (Glinz, 2007). Graphic User Interface (GUI), log in and logout functionalities, and right to Create Read Update Delete (CRUD) functionalities on the database of different sub-modules of the system by different types of users have been used to categorize the functional requirements for the proposed system.

Table 5: Functional requirements of the proposed system

S/N	Requirements	Description
1	Login and logout	All degree-awarding or NCHE and system admin have a different portal to access the database. Therefore, the system should authenticate them to access the data.
2	Create Read Update Delete (CRUD) information about the academic program, academic life cycle, asset, human resource, and finance submodules.	All degree-awarding or HCHE in each EAC's national will apply CRUD functions to the academic program, academic life cycle, asset, human resource, and finance information submitted by them.
3	View information about the academic life cycle, human resource, and finance submodules	In comparison and statistic view, anyone will view public information about universities concerning academic life cycle, human resources, and finance.
4	View information about the assets and other private information from different system submodules.	IUCEA can view a deep analysis of all information submitted, including assets information.
5	Apply to be among all degree-awarding	New universities or existing high education students will be among all degree-awarding working with IUCEA.

3.3.2 Non-Functional Requirements

Non-functional requirements set the criteria and define the performance attribute to the system according to how the system has to function; it defines how the system is supposed to be (Chung *et al.*, 2012). Finally, the table describes all quality attributes the developed system will have.

Table 6: Non-functional requirements of the proposed system

No	Requirements	Description
1	Security	The system should permit only authenticated universities to access and modifier information submitted by these universities. Each university should view and modify the information concerning their universities.
2	Robustness	The system has been developed to reduce the impact of operational mistakes, hardware errors, and erroneous input data.
3	Testability	The system has been developed upon different frameworks which accept the modularity and Structuredness approach. Therefore, the testability of the developed system depends on it.
4	Usability	The proposed system should be usable to IUCEA and let the institution automate many actions in high education. In addition, IUCEA should enjoy the easy way to operate the system should bring.
5	Learnability	The system should be easy to use and follow all rules of user experience design. In addition, the system should be designed to understand it and know what it has been built for without reading many things.
6	Reusability	The proposed system should fulfill all functions defined in the final report about the high education information system.
7	Performance	The system will support numerous statuses concurrently and handle different users without disappointment.
8	Maintainability	The system shall be suitable for debugging, the extension of functionality, and modification. In addition, the proposed system should allow the integration of new technology and easily update the functionality.
9	Extensibility	The system shall allow required modifications at the appropriate locations to be made without undesirable side effects.

3.4 Agile Software Development

During the development of this project, we will use the agile development methodology for a continuous iteration of the development and testing of the variable prototype. There will be a continuous improvement at each system stage (Abrahamsson *et al.*, 2017). Among the different approaches to agile software development such as scrum, crystal methods, adaptive software development, extreme programming, and learning development.

3.4.1 Scrum Agile Methodology

During the development of this project, we will use the Scrum agile methodology approach to have good management of time since this methodology evaluates the progress of the project day by day.

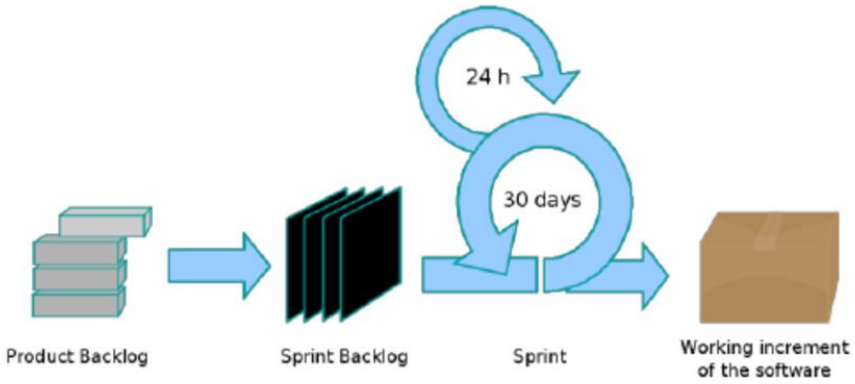


Figure 6: Diagram of Scrum agile method (Way *et al.*, 2009)

3.5 Architectural Design

This section explores the detailed design of the proposed system. It will present the conceptual design, use case diagram, data flow diagrams, and Entity Relationship diagram of the system.

3.5.1 Conceptual Design

The assessment and analysis of functional and non-functional requirements have bought to the next step, the conceptual design. The conceptual design is defined by Andreasen *et al.* (2015) as an umbrella term given to all forms of non-aesthetic design management displaces. The main lines of software function and form are established at this step of the design process. The design of encounters, experiences, processes, and tactics follows. Finally, it entails comprehending

people's wants and figuring out how to answer them via goods, services, and business processes: design artifacts, conceptual drawings, and models.

The proposed web-based university information management system will manage four types of users; the overall admin, Universities, National or Commission for High Education (NCHE), and Surfer. All those users will be able to access the information of universities concerning academic life cycle, human resources, assets, and finance with different rights on it.

The authentication is required for the overall admin to manage all information concerning universities. The Overall admin will manage all modules EAC HEIS contain. Universities will be able to apply for being among all degree-awarding higher education within EAC. The authentication will be required for all Degree Awarding High Education and each national commission or council for high education within the East African community to feed data needed by the system.

Suffer will be able to access the public information of universities within the East African community concerned academic life cycle, finance, and staff. Figure 7 presents the conceptual diagram developed to give the visual shape to a proposed system.

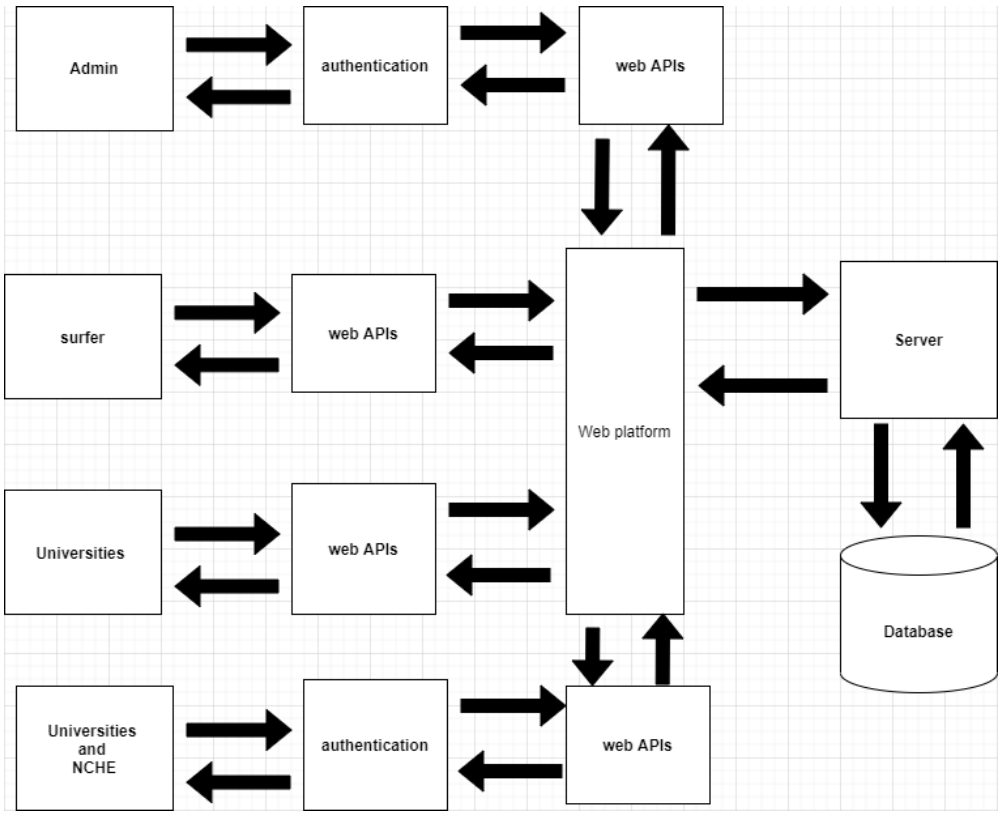


Figure 7: Conceptual diagram of the proposed system

3.5.2 Use case Diagram

After analyzing the system features and requirements, a case diagram was designed through a function and non-functional requirement. The use case diagram is among the tools used in software requirement specification (SRS) documentation to gather the system's requirements.

However, the use case diagram is one diagram among the behavioral diagrams of Unified Modeling Language (UML). UML has more than 14 diagrams to model application structures, software solutions, business processes, and system behavior. Use cases help structure the user's needs and the corresponding goals of a system and focus on expressing the system's requirements. The use cases identify the system's users, also called actors, and their interactions. They make it possible to classify the actors and structure the system's objectives. An actor represents a role played by a person who interacts with the system. When an external person, process, or item interacts with a system, they perform the role of an actor (Gomaa, 2011).

During the proposed system development, a use case diagram has been used to support the Scrum methodology for requirement definition. In addition, a brief study of the system to be put in place allowed us to identify the players and the corresponding use cases. Figure 8 shows the use cases designed for the proposed system.

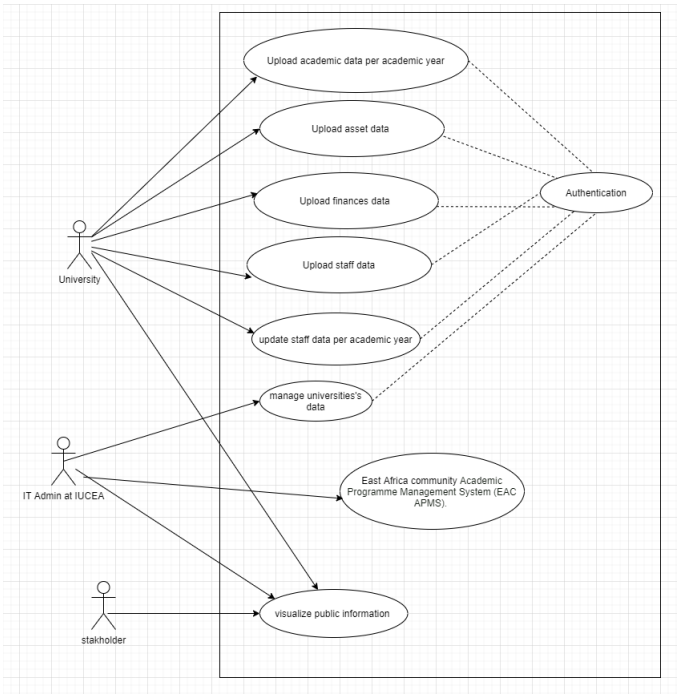


Figure 8: Use case diagram of the proposed system

3.5.3 Entity-Relationship (ER) Diagram

During system design ER diagram as a structure, diagram has been used to display the relationship between entities. In addition, ER diagram has been used to support the Scrum methodology to explain the logical structure of databases. Figure 9 shows the entity-relationship diagram of the proposed system, which displays the entail system's entities, attributes, and relationships.

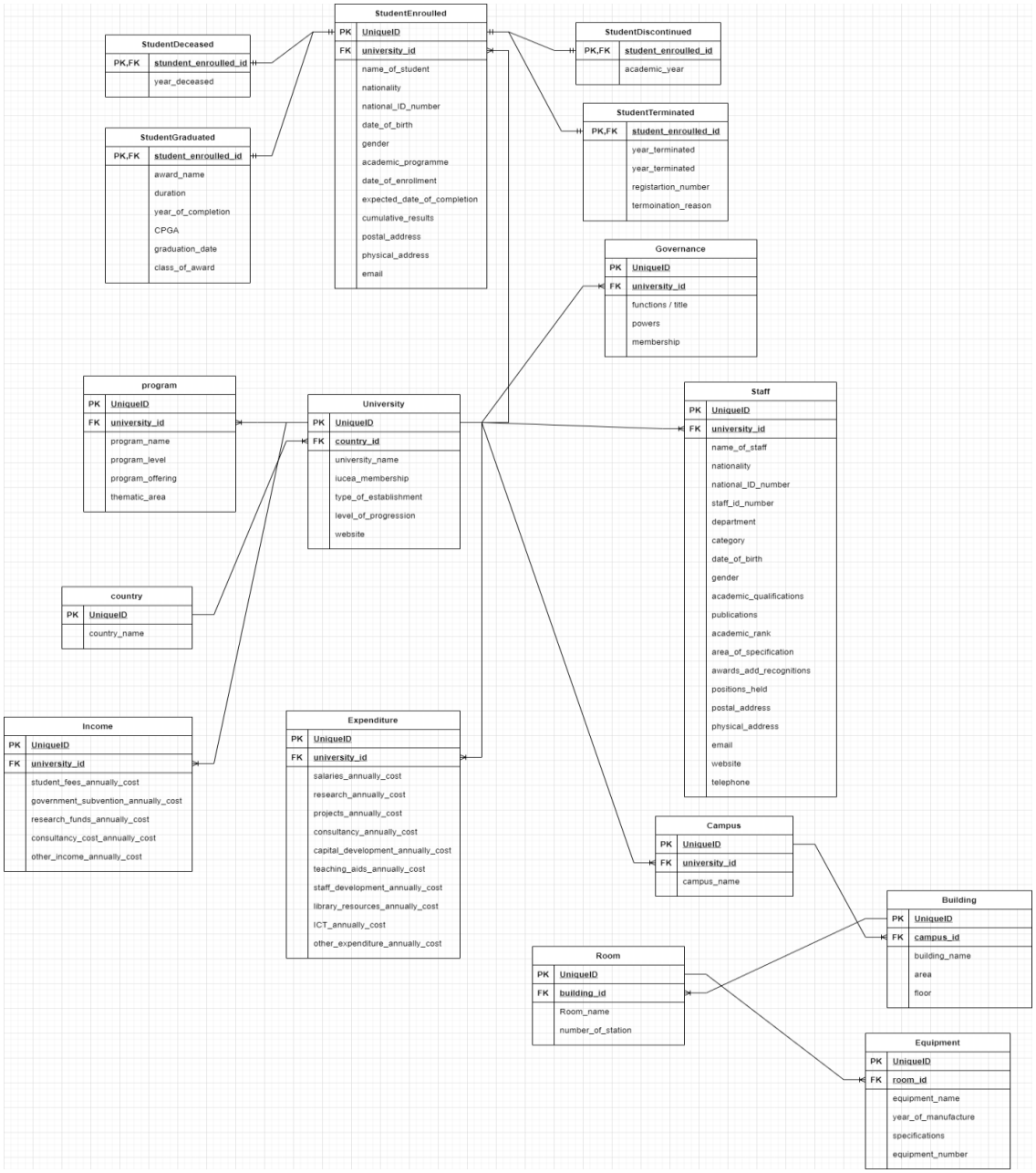


Figure 9: Entity-Relationship (ER) Diagram of the proposed system

3.6 Tools and Technologies in System Design and Implementation

This section explores the tools and technologies used to develop the proposed system. It starts by describing different technologies from their history and the current state of those technologies, and finally, this section will describe the tools used to develop the proposed system.

3.6.1 Web Technology

In a nutshell, web technologies are related to the markup languages that computers and other devices use to interact with one another via the internet. A hypertext markup language (HTML) is used to communicate over the web and to create, transport, or manage online content (HTML) (Berners-Lee *et al.,* 1994).

There is a big difference between the world wide web (WWW) and the Internet. The Internet is a global network that connects computers all over the world. Many protocols enable computers connected to the Internet to communicate with one another, including Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP), Internet Relay Chat (IRC), Instant Message (IM), Telnet, and Peer-to-Peer (P2P). Interconnected hypertext pages and applications may be found on the WWW, which can be accessed over the Internet, usually using HTTP (Hitzler *et al.,* 2009; Gralla, 1998).

The web has a long history behind it. In 1969 the ancestor of the Internet called Arpanet was created. It was a military network that wanted to be decentralized (with no central command Center). The network then evolved into a place of academic exchange before gradually becoming mainstream under the name of the Internet. In 1972 emails to exchange messages appeared. Moreover, the web was created in 1991 to display information on pages (Meier-Brügger, 2010).

When we talk about web technology, we have two types of computers called Server or Backend and Client or Frontend, which must communicate through the Internet. The backend is the part that runs on a web server and is responsible for data processing and validating business rules. The frontend is the part that is loaded inside a web browser on a client machine; it is the part that the user sees and interacts with. Hypertext Transfer Protocol (HTTP) governs this flow of information. Servers and clients may interact with each other using it. The client makes an

HTTP request to the server and gets a response in the form of an HTTP message (Luck *et al.*, 2004).

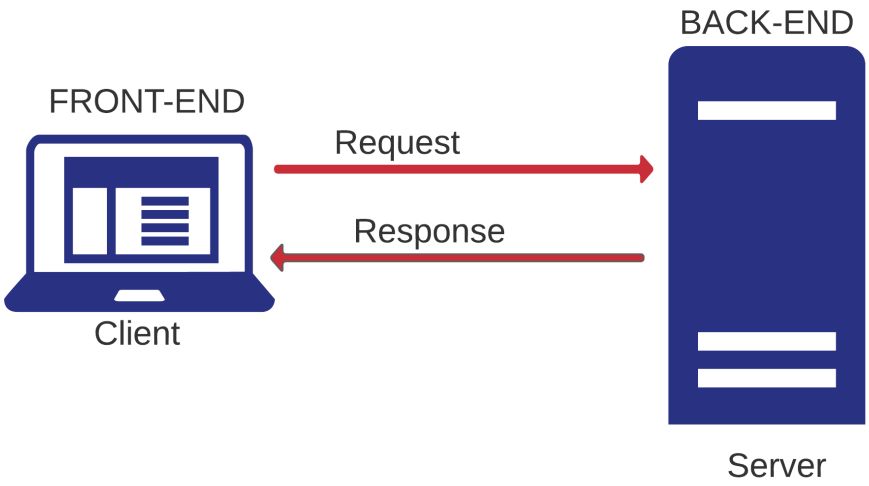


Figure 10: General concept of web technology

To build the backend for the website, we need to decide how we will respond to clients. The server might produce the requested page and send it back to the client. To do this, our team uses HyperText Markup Language (HTML). In HTML, web pages and their content are expressed using a basic markup language. Technically, the server generates the page and returns an HTML document to the client.

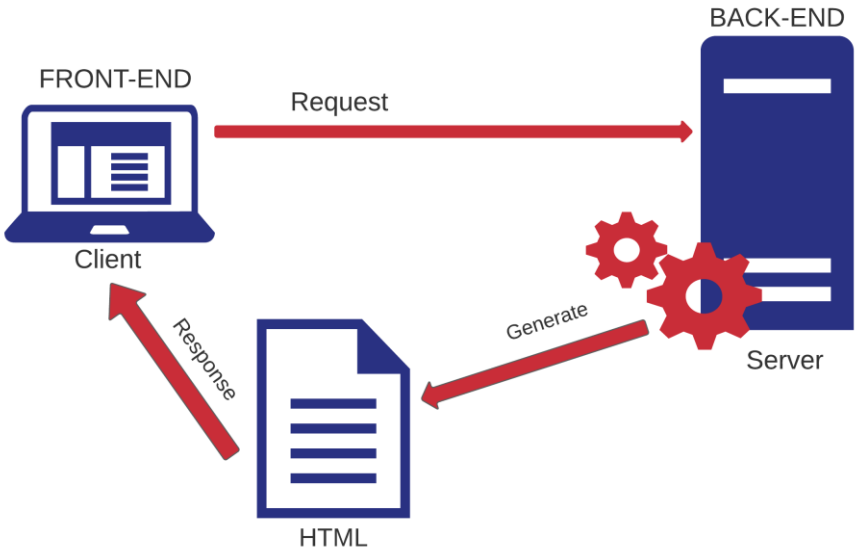


Figure 11: First approach of responding to the client by generating an HTML file

Returning just the information required to create the requested page and letting the client do the rest is the second option you have. Instead of putting a complete page or a complete HTML document in an HTTP response, we only return the data. This approach frees up the server, enables it to serve more clients, and makes it more scalable. The server becomes a gateway to the data. On the server, we can provide endpoints that the client can talk to get or save various pieces of data. Together, these endpoints represent the clients' interface to talk to the server. The server provides an API or an application programming interface to the client in technical terms.

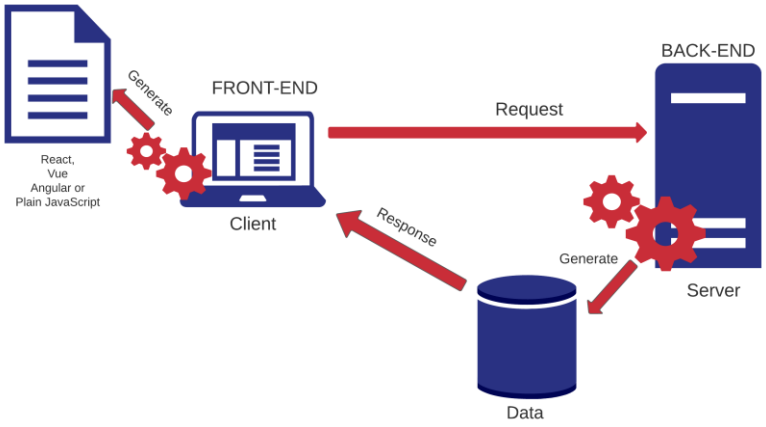


Figure 12: The server generates data, and the client generates a file to present data

To develop the client-side (Frontend) of the proposed system, react has been used for generating web pages on the client, and the server-side (Backend) has been developed using Django.

3.6.2 Web APIs

Application Programming Interface (API) is an interface for software, a point where two programs may meet and interact. The term API is often used to name the whole software product, including the API and its implementation (Robillard, 2009).

We have two main API types: hardware and software APIs. The software APIs are also divided into remote APIs and local APIs. Web APIs are remote APIs that may be used with the HTTP protocol. The web APIs can be public APIs or personal APIs. Public APIs are designed as the product or service by others; they do not construct, run, or install them; however, they use them. Public APIs are provided to anyone who needs them and is willing to accept the terms and conditions of the third-party supplier. Depending on the business model of the API providers,

such APIs can be free or paid for, just like any other software product. On the other hand, a private API is built for a company or institution; only applications developed by the company or teams inside the company use it. In this case, they are API providers and API consumers (Henning, 2007).

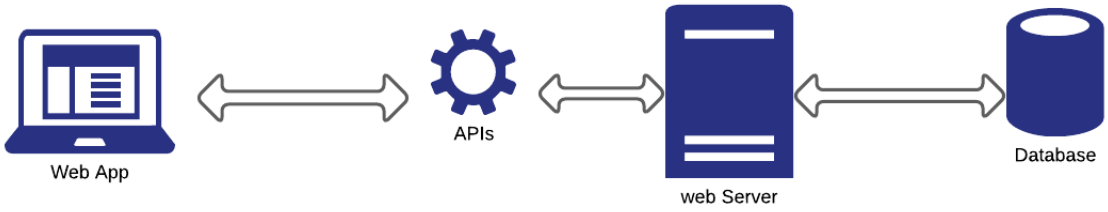


Figure 13: Illustration of the web APIs programming

The whole point of using an API is to make it easy for people to achieve their goals, regardless of the programming part. There have been, and there will be, many different ways of exposing data and capabilities through software; also, as different ways of enabling software to communicate over a network. For example, RPC (Remote Procedure Call), SOAP (Simple Object Access Protocol), REST (Representational State Transfer), gRPC (Google Remote Procedure Call), or GraphQL are different technologies used to create web APIs. Some are architectural styles, and others are protocols or query languages (Paternò & Santoro, 2019).

The REST style has been used to develop the private web APIs for the proposed system. REST is an architecture first proposed in 2000 by Roy Fielding in his PhD dissertation thesis. It is an approach to building APIs that relies on HTTP protocol. Software architecture must comply with the following six constraints in others to become RESTful as described by Patni (2017):

(i) Client and Server separation: client and server must be separated and allow communication between them. Once a web application and its API server are integrated, it should be possible to clearly distinguish between the two.

(ii) Statelessness: The request itself contains all of the information needed to carry out a task. During consultation between requests, no consumer context is kept on the server.

(iii) Cache ability: Requested information must describe what can be saved and how long it will be saved, so the client doesn't have to repeat the process.

(iv) Layered system: When a user interacts with a sensor, the server is the only thing they can see. There is only one layer of the system that is visible to the end user.

(v) Code on demand: Servers may deliver executable code to clients (such as JavaScript); this constraint is optional.

(vi) Uniform interface: All interactions must be driven by the idea of resources that have been recognized and may be managed via representations of resources in various stages of use. Information on the resources and their uses must be provided as part of standard method interactions.

3.6.3 React

JavaScript's React library allows developers to quickly and interactively create user interfaces. The most widely used JavaScript library for developing user interfaces was created by Facebook in 2011. React is the most used library and framework for creating user interfaces, according to Google trends. Angular and vue.js are the other two contenders (Gackenheimer & Paul, 2015).

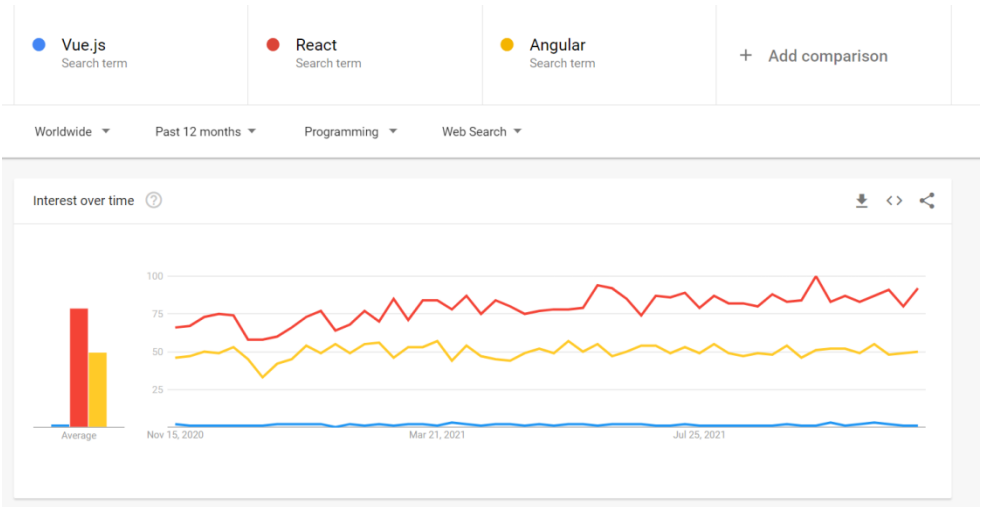


Figure 14: Trend between Angular, vue.js and React

As a user interface component, a component is at the center of all react activities. When using React to develop applications, we generate a large number of small, independent apps and then combine them to make larger, more complicated apps. The root component is a need for any React project. This component serves as a container for all of the other components that make up the application as a whole (Gackenheimer & Paul, 2015).

A JavaScript class including a render function and some state is used to design and build a component. When a component is drawn, the render function describes how the User Interface should appear based on the data in the component's state. As a result of the rendering technique, a JavaScript object that responds to a document object model item is generated (DOM). It's only a JavaScript object that represents it in memory, not an essential part of the DOM. The virtual DOM is React's lightweight version of the DOM in memory. When it comes to creating a virtual DOM, it's a lot easier than creating an actual one. For each component that we alter, we receive a React element, which is compared to the previous one to determine what has changed; subsequently, React updates a portion of the real-world DOM to match the virtual one. For this reason, we don't require the DOM API in browsers when building apps with react, unlike vanilla JavaScript and Query. That is to say, no code in Query or DOM manipulation is required, nor are DOM elements or event handlers required to be attached. As a substitute, we can just make changes to the state of our components, and React will take care of updating the DOM accordingly. In addition, this library is named React because react responds to the state change and changes the DOM (Jensen *et al.*, 2009).

React and angular are similar in component-based architecture; however, angular is a framework or a complete solution, while React is a library. React renders the view and makes sure the view is in sync with the state. For this very reason, react has a minimal API to learn. When building applications with React, we need to use other libraries or things like routing or calling HTTP services, and many more. Nevertheless, these are not necessarily bad because we get to choose the libraries we prefer instead of being fixed with what Angular gives us, which often breaks from one version to another (Saks, 2019). React library has been used to develop and implement the frontend side of the proposed system.

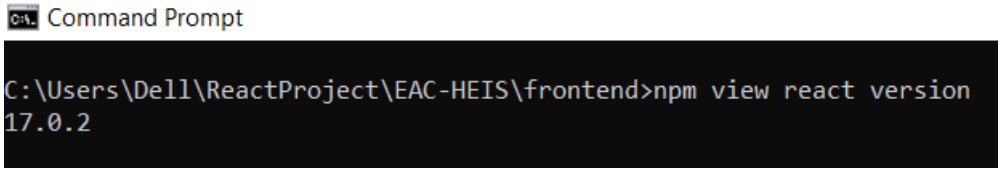


Figure 15: React version used to develop the proposed system

3.6.4 Node JS

Node.js is one of the prerequisites to work with React. It is used to manage the project. Figure 16 shows the node JavaScript version V14.17.1 downloaded on NodeJS.org and installed to manage the proposed system.

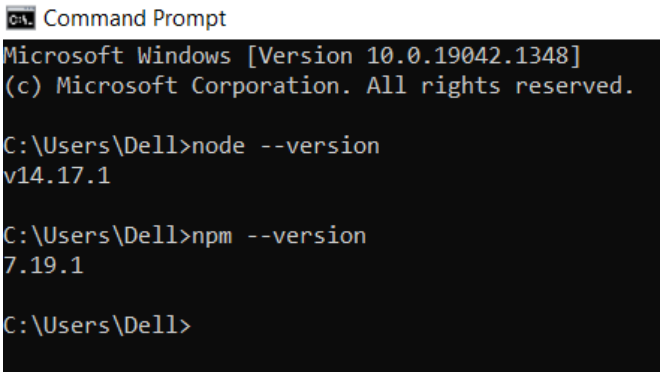


Figure 16: Widows, Node JavaScript runtime, and Node Package Manager (NPM) versions used to develop the proposed system

Node JS installation allows us to run the node JavaScript runtime and the Node Package Manager (NPM), an invaluable tool for installing the packages for the project.

3.6.5 Django Framework

Django is among the open-source frameworks for building web apps with python. It is maintained by the Django software foundation and was initially released on 21 July 2005 (Forcier *et al.,* 2008). A web framework may be defined as a collection of modular tools that abstract the difficulty and repetition inherent in web development. For example, most websites need the same basic functionality. These include setting Uniform Resource Locator (URL) routes, connecting to a database, handling security properly, displaying content on a page, and many more features. Rather than recreate all of this from scratch, programmers have created web frameworks in all the major programming languages (Curie *et al.*, 2019).

Django is not only the web framework for python; there are many and many more; in comparison, Django is the most popular one and old, which helps build a website in less time with few lines of code (Chouhan *et al.,* 2020). Figure 17 compares Django and other web frameworks such as Flask, Tornado, Bottle, Falcon, and Hug.

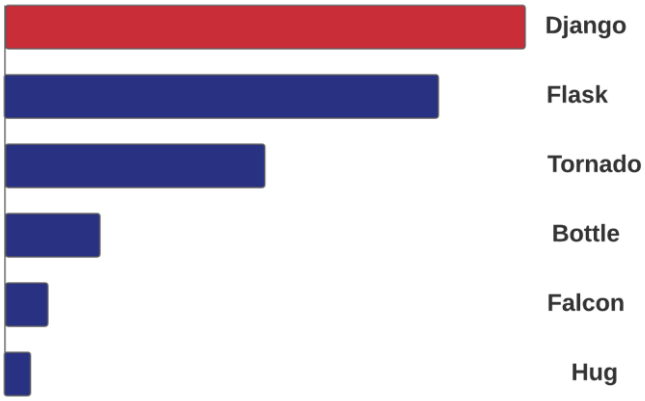


Figure 17: Comparison of Web framework built with python by popularity

Django is what we called batteries included, which means it comes with many features:

(i) Admin site gives us an admin interface for managing data which is a huge time saver.

(ii) Object-Relational mapper (ORM) allows the machine generates database tables based on the application's classes and abstract database. We can query or process data without writing a lot of Structured Query Language (SQL) code.

(iii) Authentication package for identifying users.

(iv) It also has a package for caching data.

The current Django version is 3.2.7, released on 1 September 2021. Figure 17 represents the Django released cycle from 2017 to 2026 as reported by the (***https://jefftriplett.com/django-release-cycle***) website visited on 2 December 2021.

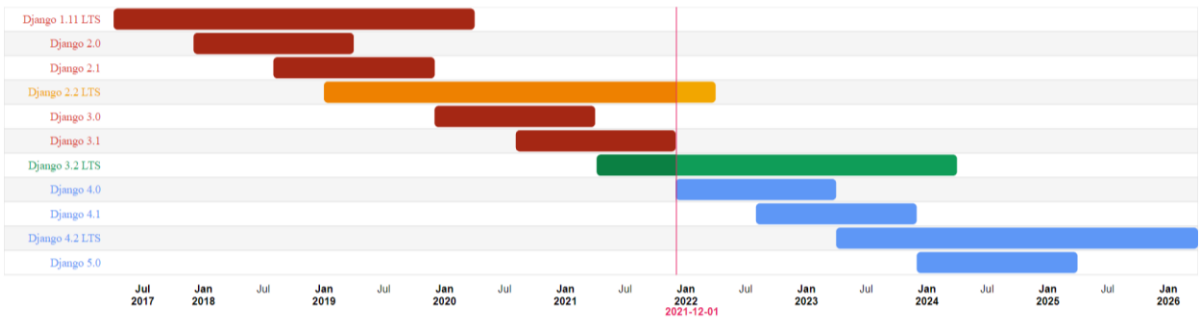


Figure 18: Django release cycle on https://jefftriplett.com/django-release-cycle/

The proposed system has used the current Django framework to develop the system's backend. Django combined with Django REST Framework library has been used to develop all web APIs the system will need to interact with the frontend side.

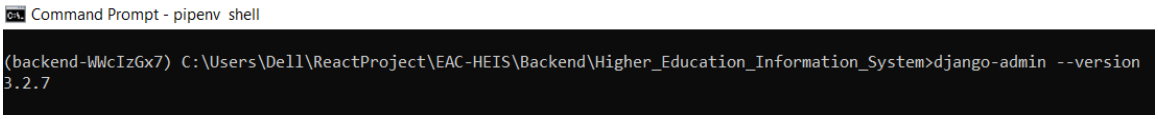


Figure 19: Django framework used to develop the proposed system

3.6.6 MySQL

MySQL is an open-source relational database management system (RDBMS) that is developed, distributed, and supported by oracle corporations (Christudas, 2019) (Malecha *et al.,* 2010). MySQL's name is a combination of "My," one of the names of the co-founder Michael Widenius's daughter, and "SQL," the abbreviation for the Structured Query Language (SQL). MySQL has been chosen for the proposed system among many relational database management systems because it is free and popular, and many companies use it to store information.

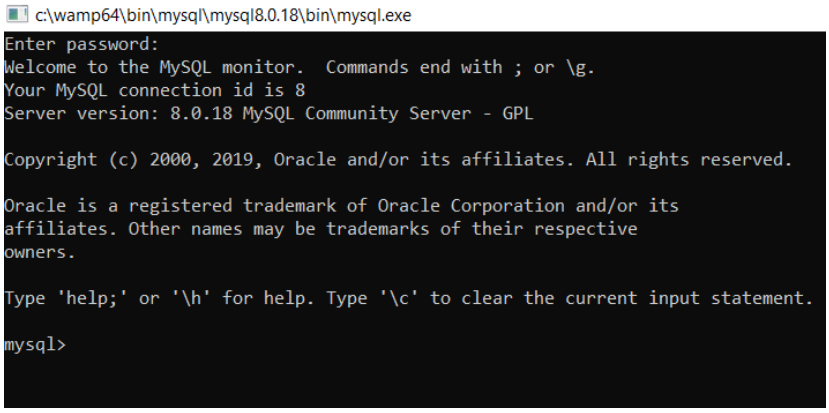


Figure 20: The version of MySQL server installed for the development of the proposed system

3.6.7 Visual Studio code

The visual Studio code editor has been used to develop the proposed system. It is an extensible code editor free and developed by Microsoft for Windows, Linux, and Mac OS. It offers various features, including debugging assistance, integrated Git, intelligent code completion, syntax highlighting, snippets, and refactoring of existing code.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Results from Data Collection

Different IUCEA members deal with direct questions about high education responded that: *"There is a need of East African Community University Information Management System able to manage academic life cycle, human resource, asset, and finance data to help IUCEA in knowledge Management (KM) and electronic resource planning (ERP) within EAC region. Moreover, the system should be able to manage information and advice or help IUCEA in decision making."*

The IUCEA was tasked with operationalizing the EAC Common Higher Education Area. For IUCEA to effectively implement the EAC Common Higher Education Area, there is a need to establish a one-stop live repository for information and data on higher education status and dynamics in the region. IUCEA can generate regional reports for the Partner States and also member universities.

4.2 Result of the Backend Design and Implementation

The backend consisted of database models, URLs, and views that interacted with the frontend templates of HTML, Cascading style sheet (CSS), and JavaScript that controlled the presentational layout of each web page if using the MVC (Model Control View) approach. In addition, the frontend will interact with the backend via URLs called endpoint or web APIs. This section will present database and web APIs designed to build the backend of the proposed system.

4.2.1 Database Design Section

The database has been developed based on the existing database for the first module of EACHEIS. In addition, different tables to store aggregated data from all degree-awarding about the academic life cycle, assets, governance, finance, and human resource have been added to the existing database. Further, the database has been implemented on a MySQL server to store and retrieve relational data using the MyISAM database engine. Figure 21 presents some of the 59 tables and views of the whole database of the developed system.

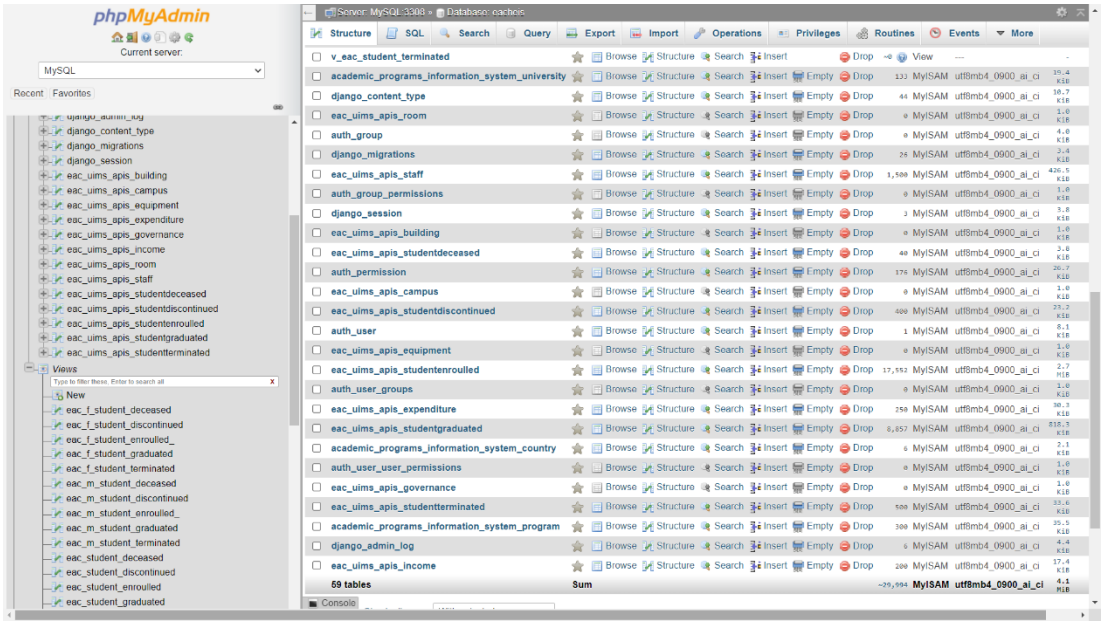


Figure 21: Database of the proposed system viewed through phpMyAdmin

4.2.2 Web APIs Design Section

Different web APIs to communicate with the developed database have been designed for the proposed system. Further, all Web RESTful APIs for the proposed system have been developed using Django and Django RESTful framework libraries. Django RESTful framework provides the visualization by default. Figure 22 shows the endpoint which GET and POST the student information.

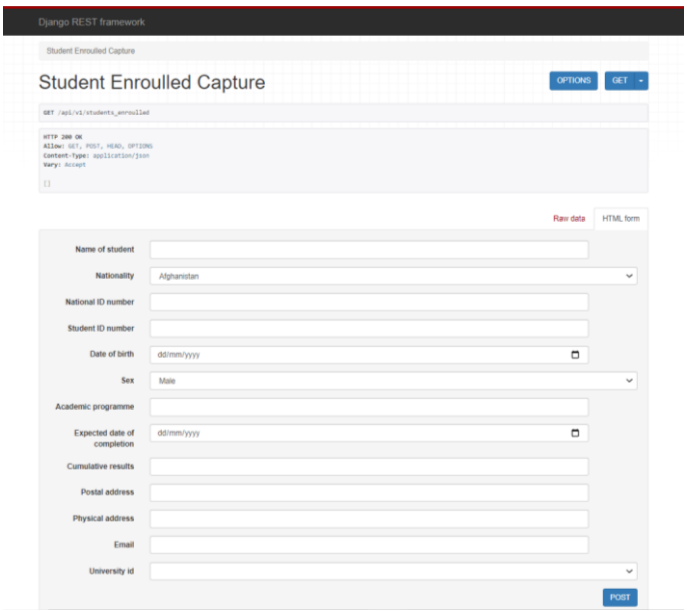


Figure 22: Web API to manage students enrolled

Fig. 23 shows the endpoint to manage some hypertext transfer protocol (HTTP) verbs which GET, POST and DELETE the student deceased information. This endpoint helps the specific high education to upload, update and retrieve information about student deceased in a specific academic year.

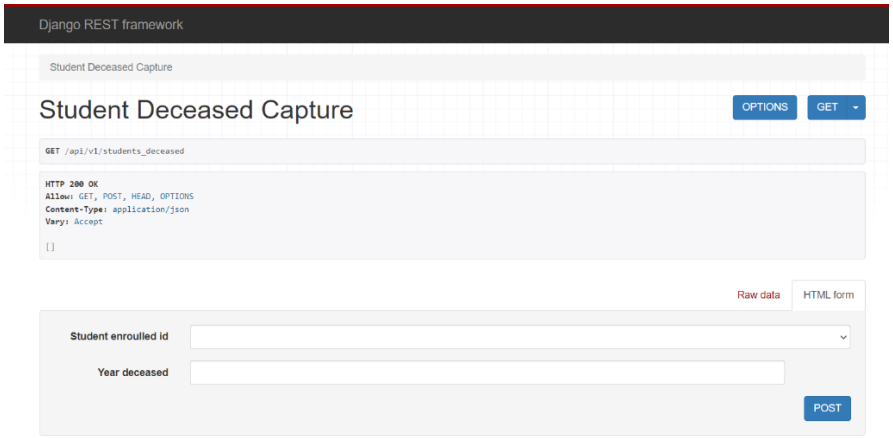


Figure 23: Web API to manage students' deceased

Fig. 24 shows the endpoint which will retrieve, update and register information about graduation. It corresponds to GET, POST and UPDATE in HTTP verbs. Students graduated should be registered by providing their registration number, award name, duration, completion year, their GPA, their award's class and their graduation date.

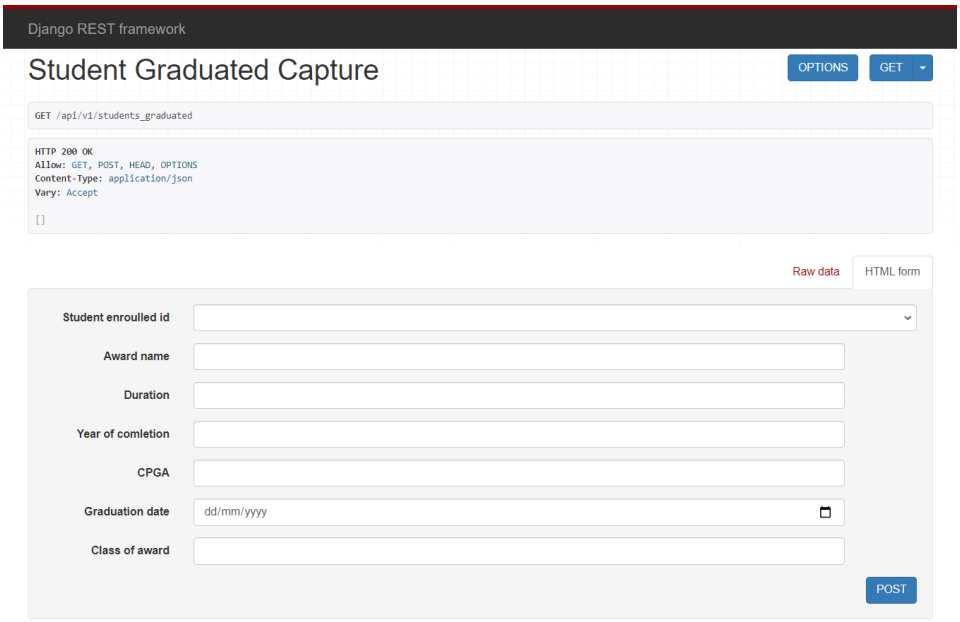


Figure 24: Web API to manage students who graduated

Fig. 25 shows the endpoint to capture the information about discontinued student. it helps to register, delete and update student discontinued in a specific institution. The endpoint will manage HTTP verbs like POST, GET and PATCH.

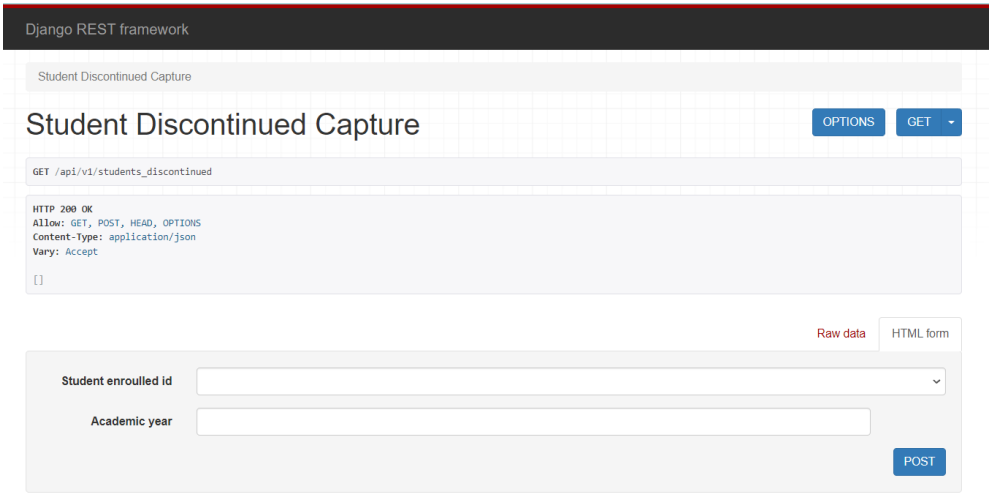


Figure 25: Web API to manage students discontinued by academic year

Fig. 26 shows the endpoint which manage the student terminated in a specific high education. It all the registration, retrieve and modification functions on student terminated. The HTTP verbs like GET, POST and DELETE are used to manage that information.

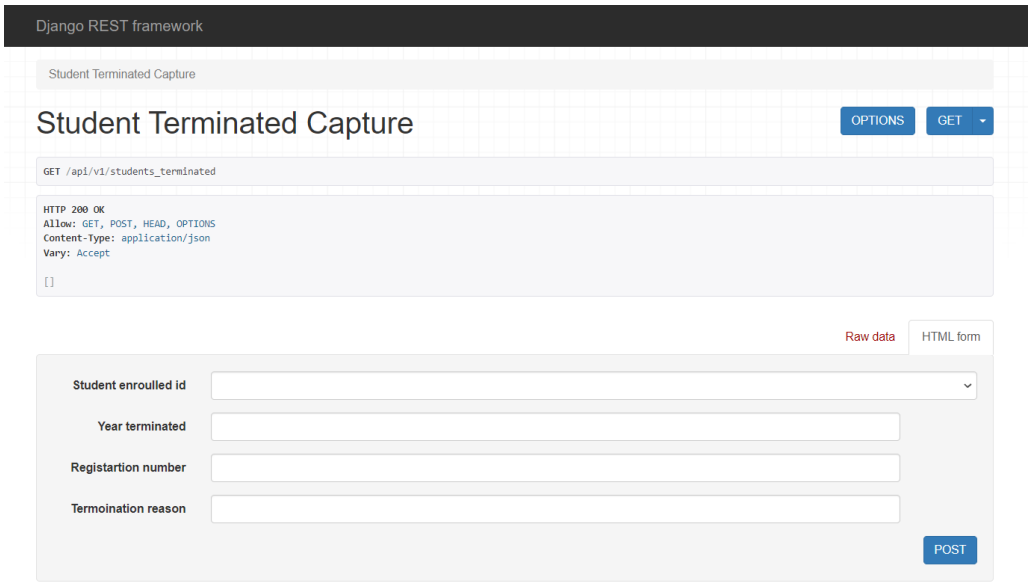


Figure 26: Web API to manage students terminated by academic year

Figures 27 to 31 present different Web APIs developed for assets. Figure 30 and Fig. 31 for human resources and governance.

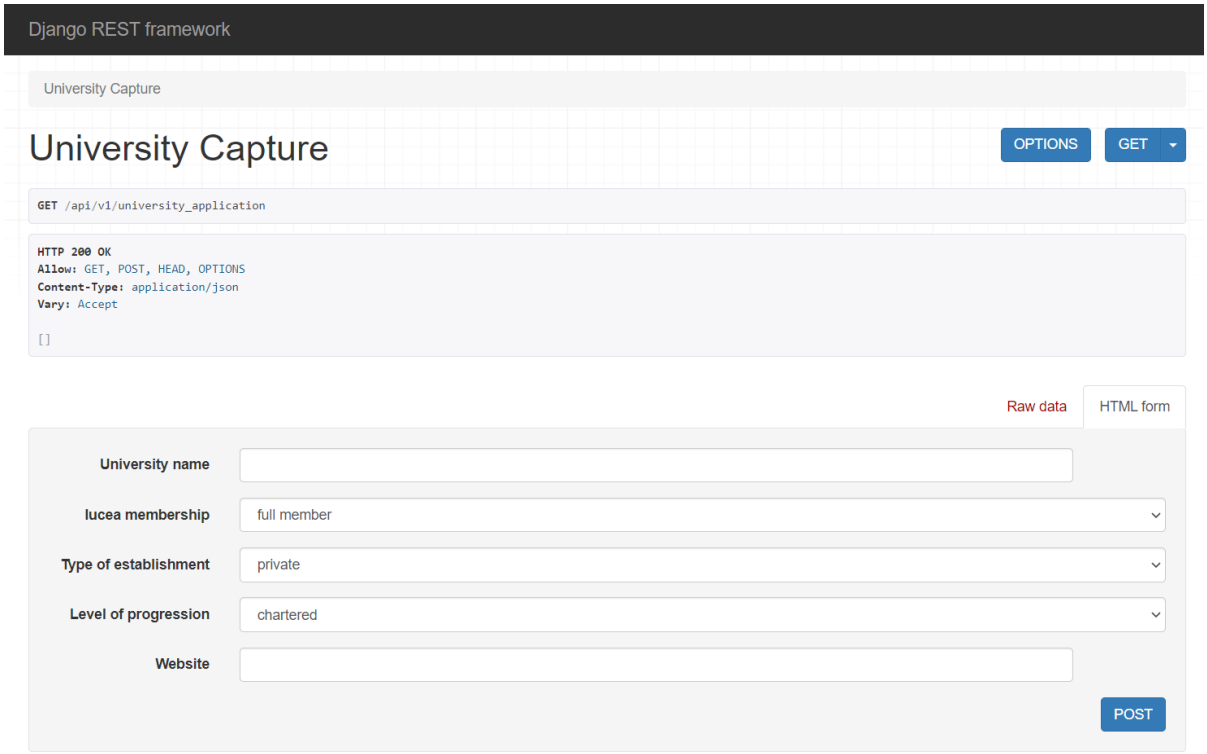


Figure 27: Web API to register universities among the APIs used to build the application portal

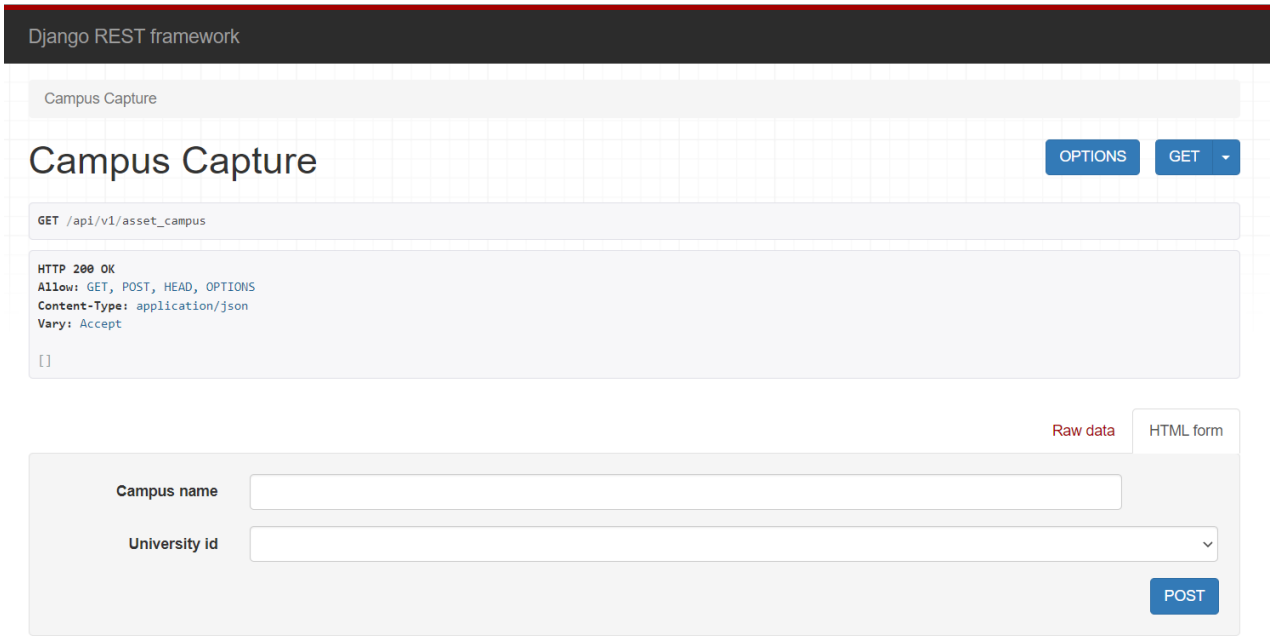


Figure 28: Web API build to capture campus within one university

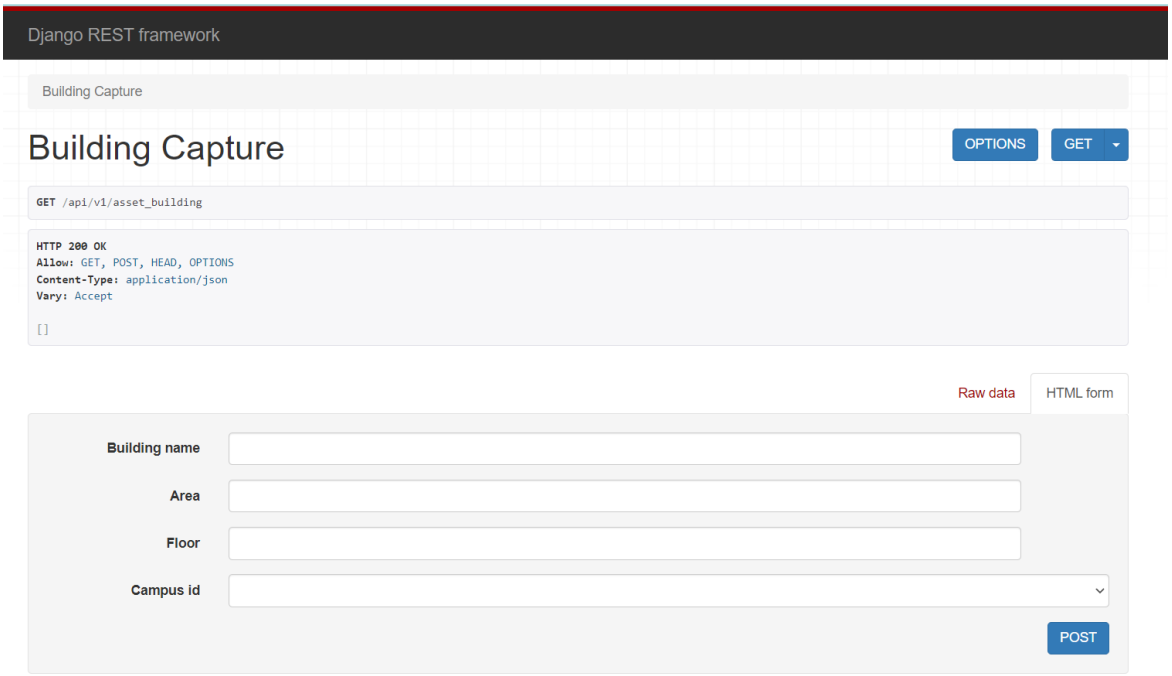


Figure 29: Web API to manage building within one Campus

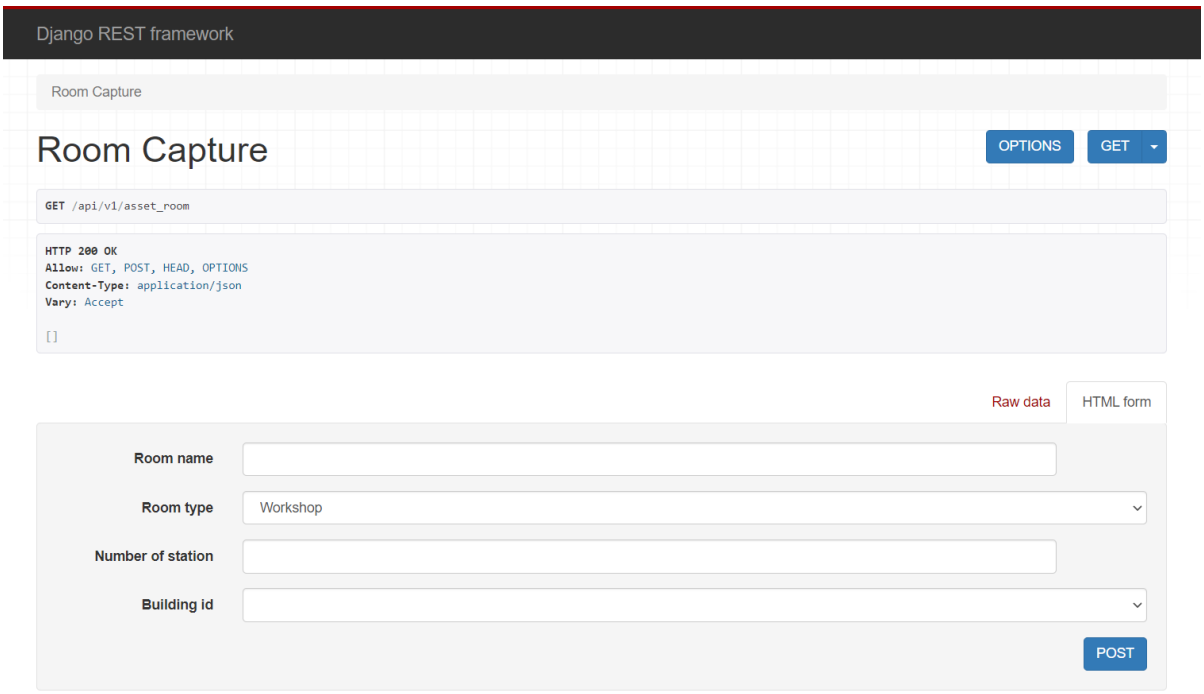


Figure 30: Web API to manage rooms within the building

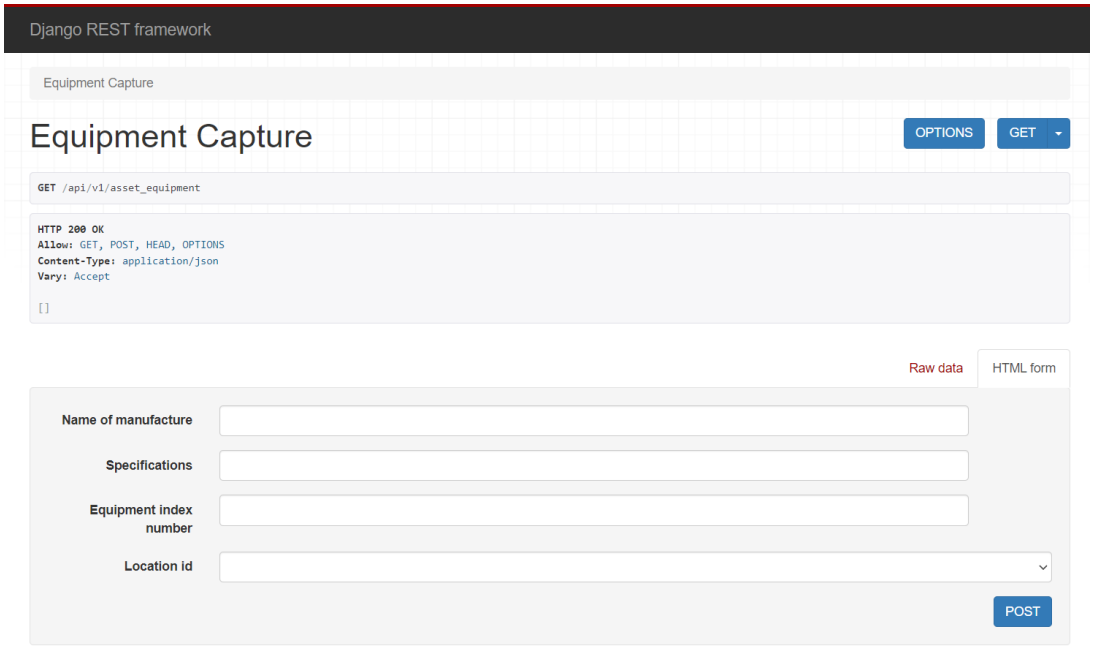


Figure 31: Web APIs to manage equipment within one room within a building

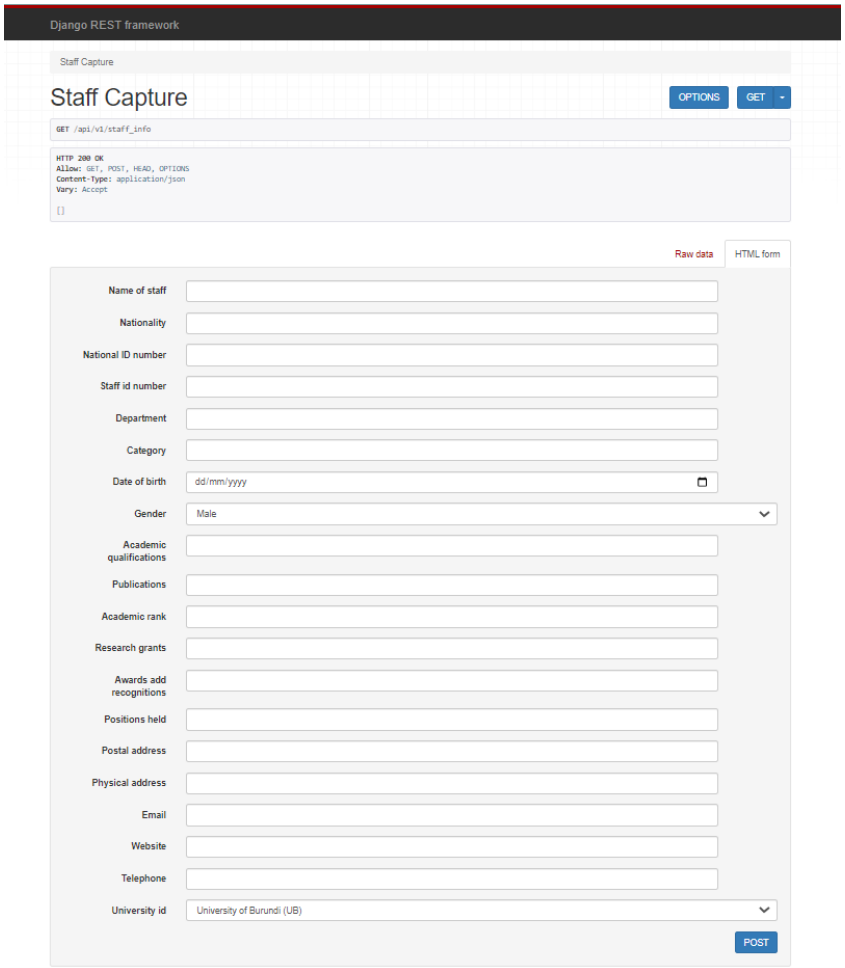


Figure 32: Web API to manage staff information

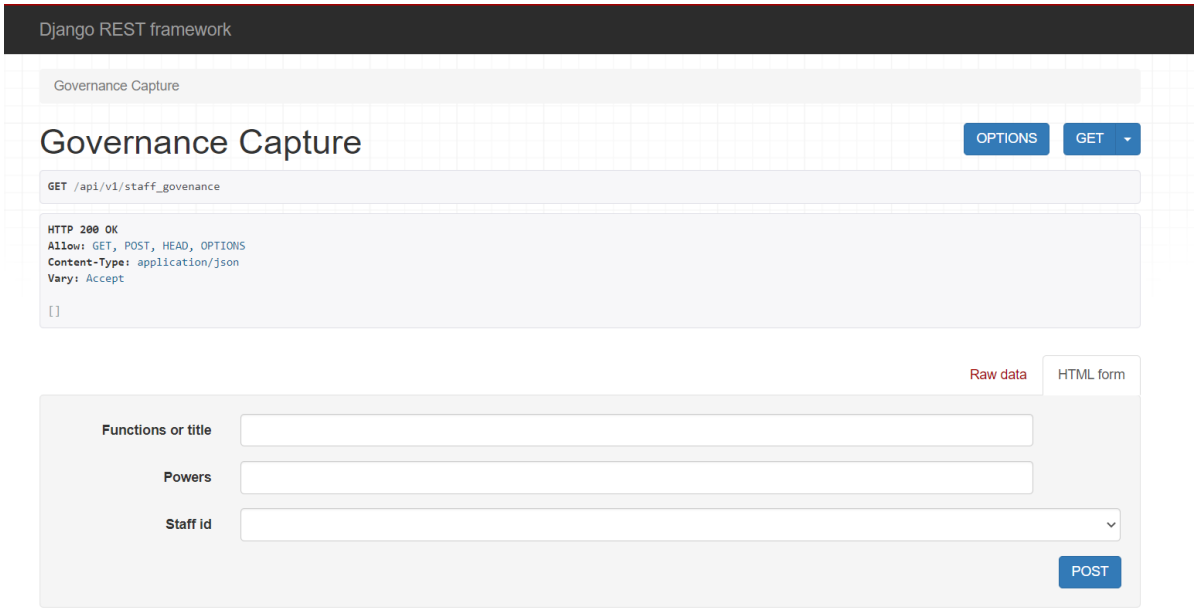


Figure 33: Web API to manage university's governance information

Figure 26 and Fig. 27 show how to manage the finance within universities.

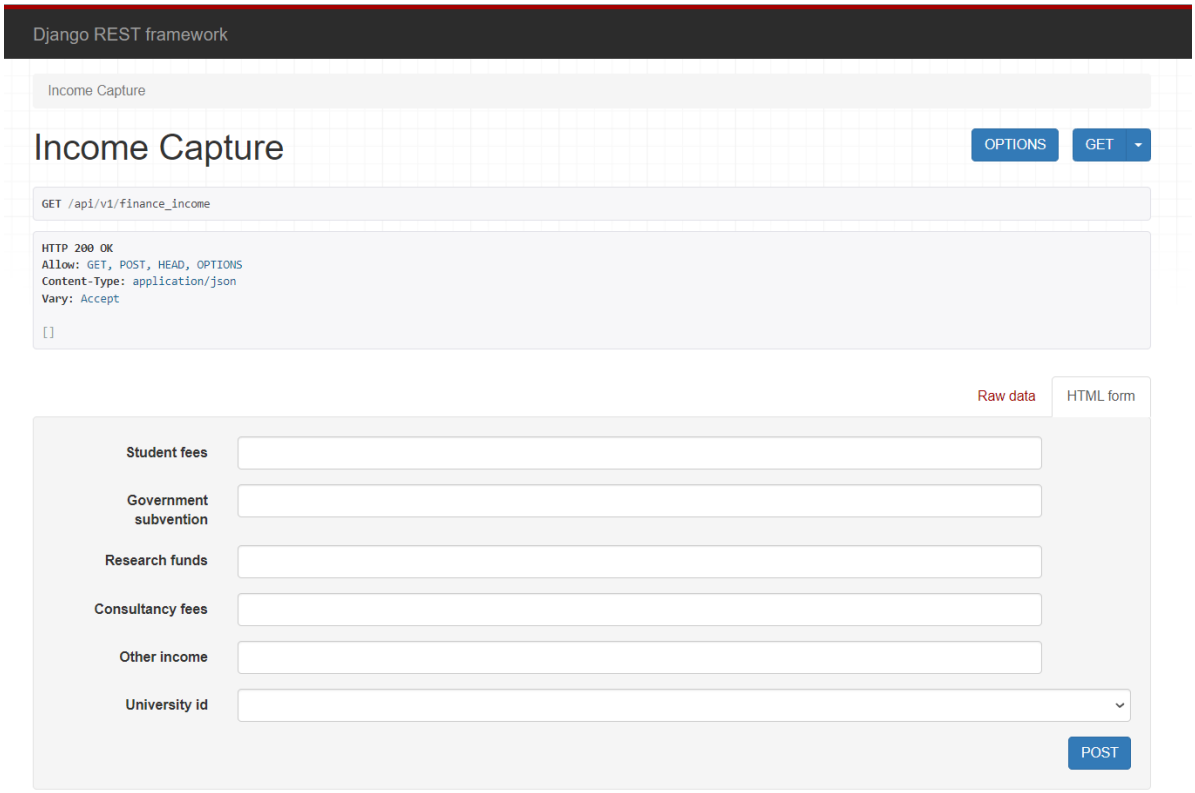


Figure 34: Web API to manage income within the university

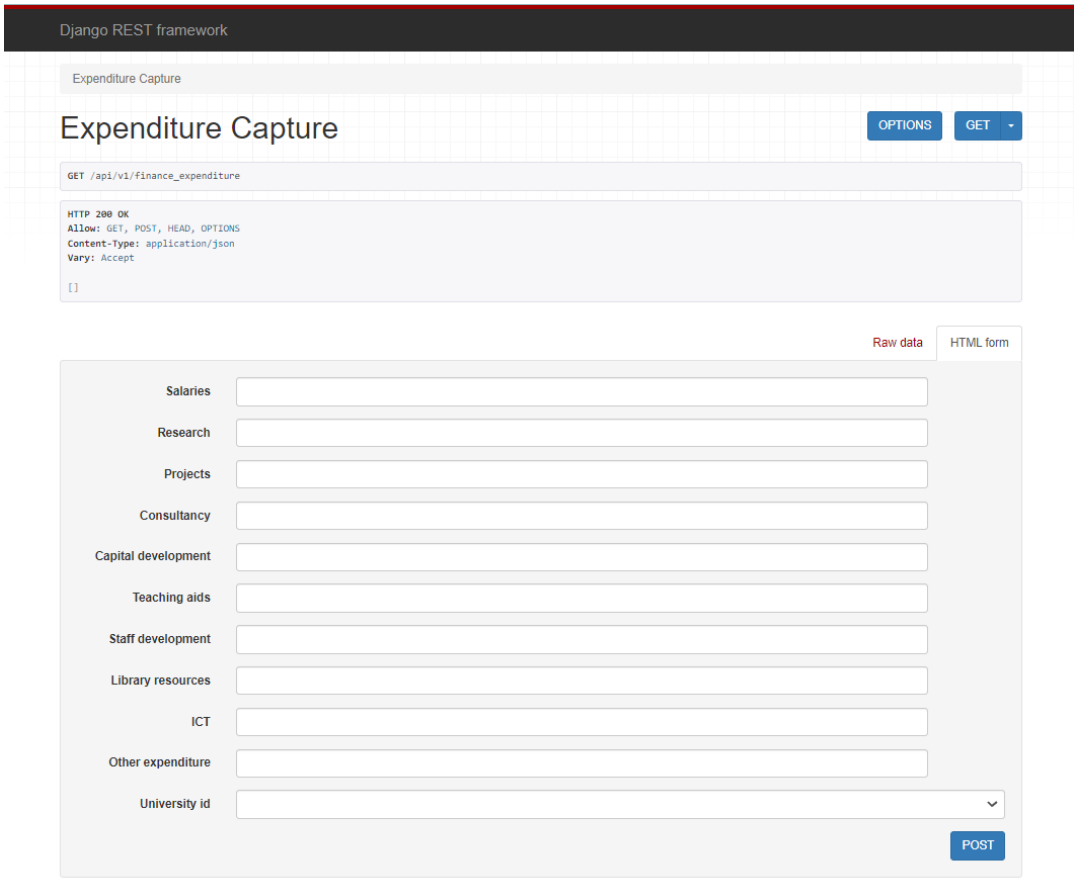


Figure 35: Web API to manage expenditure within the university

4.3 Result of the Frontend Design and Implementation

This section will present different portals of different system users developed for the proposed system. Those portals will contain labels and buttons, more of them equal to APIs.

4.3.1 Public Dashboard Design Section

The public dashboard displays public information about universities within the East African Community region to anyone who wants to know. That information is subdivided into three submodules: academic life cycle, cost of study, and human resources. This section summarizes the public dashboard design with some interface and explains the public information the universities within the EAC region will display. The following figure displays the home page of the public dashboard.

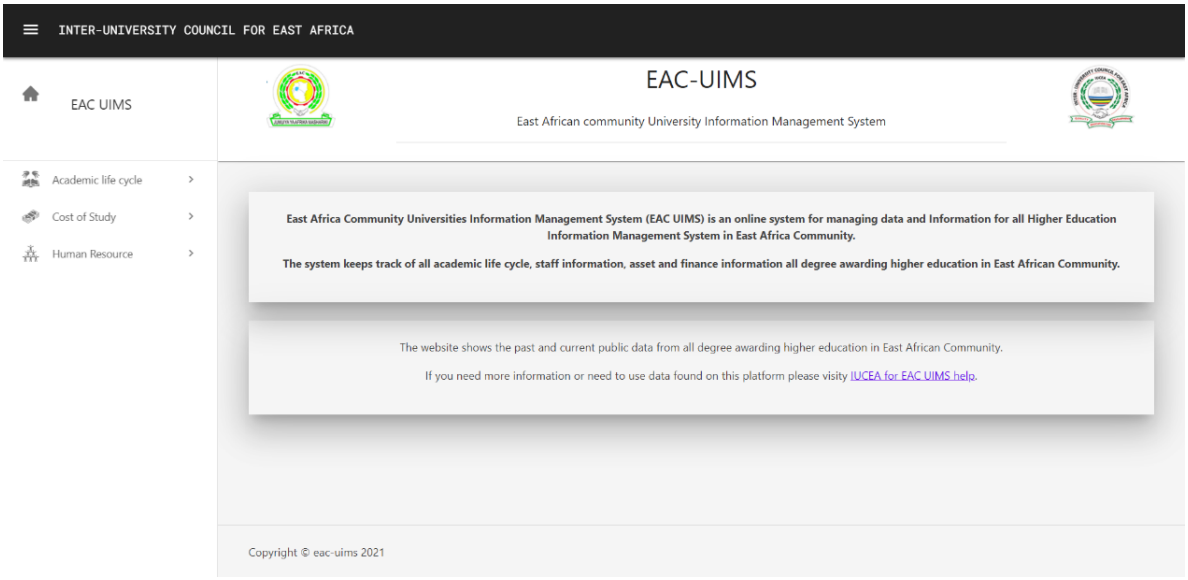


Figure 36: Homepage of the public dashboard

The academic life cycle section displays information about students Enrolled, students who graduated, students who discontinued, and schools abandoned per academic year for each country of the East African community by gender, nationality, and thematic area. For example, the following figures display students enrolled in Kenya by academic year from 2010 to 2021 by gender, nationality, and thematic area.

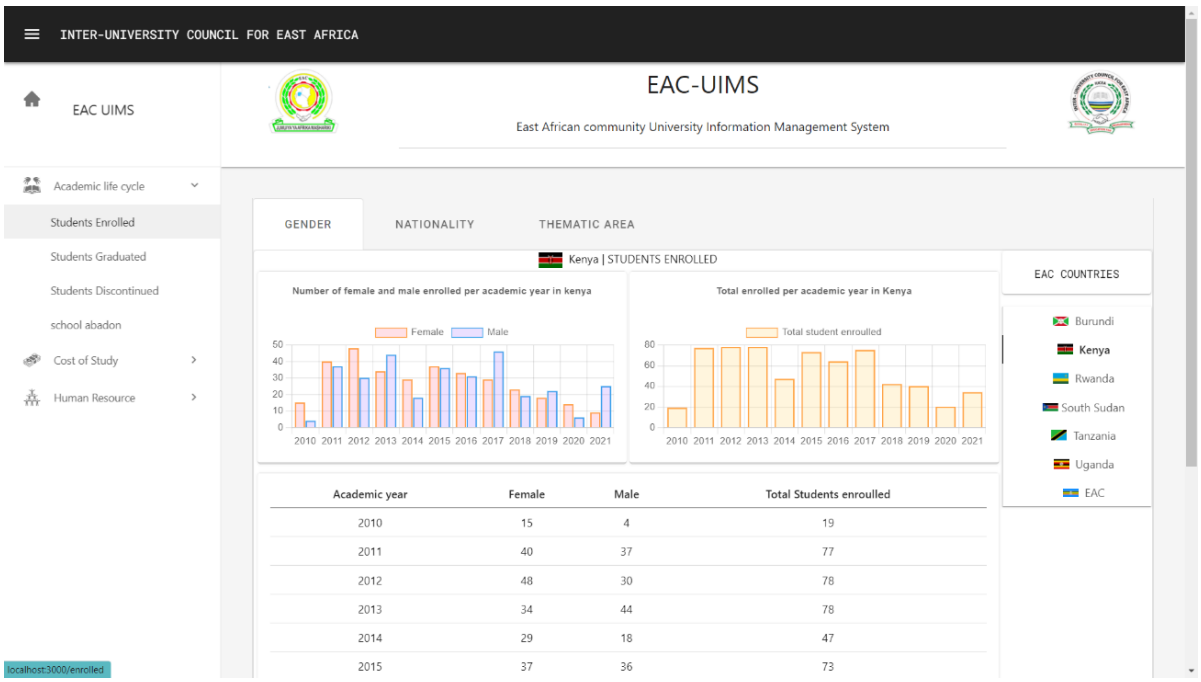


Figure 37: Students enrolled in Kenya from 2010 to 2021 academic year by gender

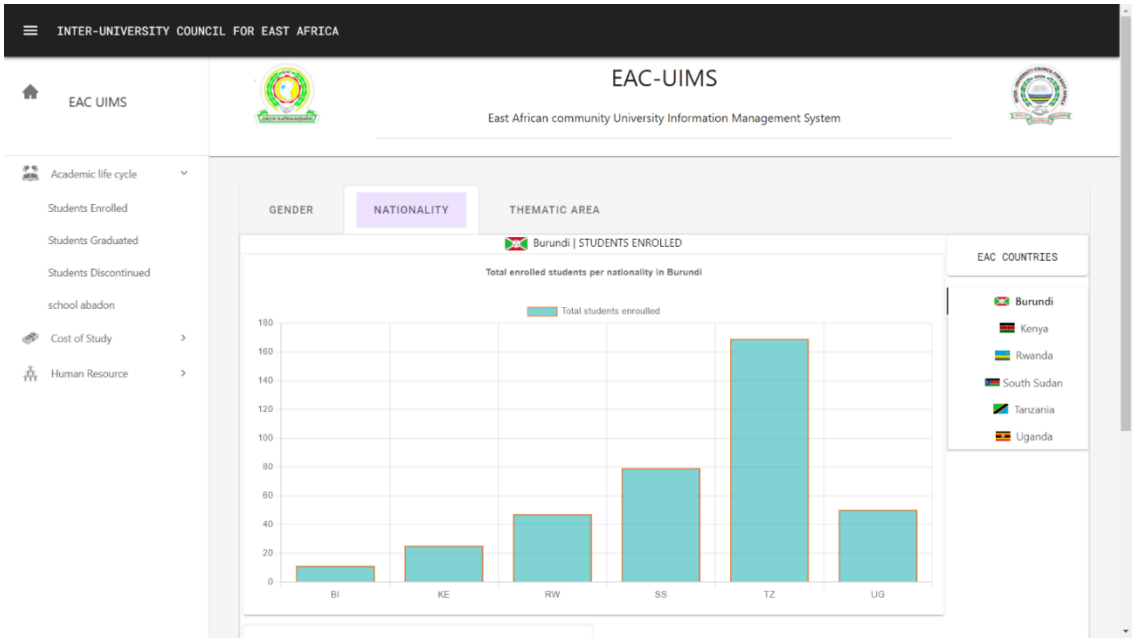


Figure 38: Students enrolled in Kenya from 2010 to 2021 academic year by nationality

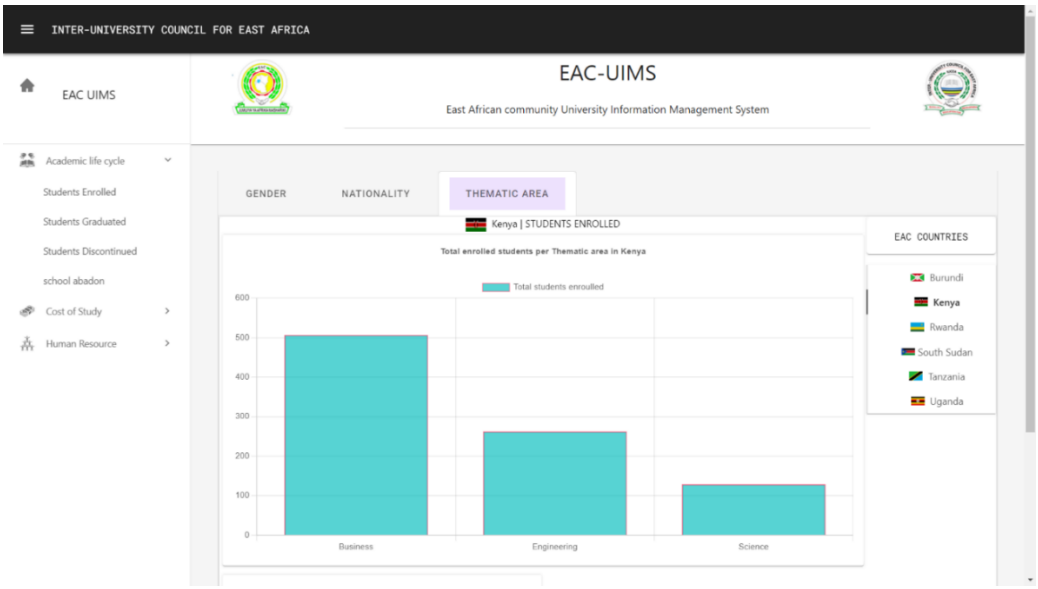


Figure 39: Students enrolled in Kenya from 2010 to 2021 academic year by thematic area

The cost of study section displays information about finance in different high education program levels such as bachelor's, diploma, Master's, and PhD. This information gives a picture of the cost of the study within the East African Community region since it displays the average cost of income and expenditure by each country by program level. For example, the following pictures show the average university income and expenditure of diploma level in Rwanda and EAC.

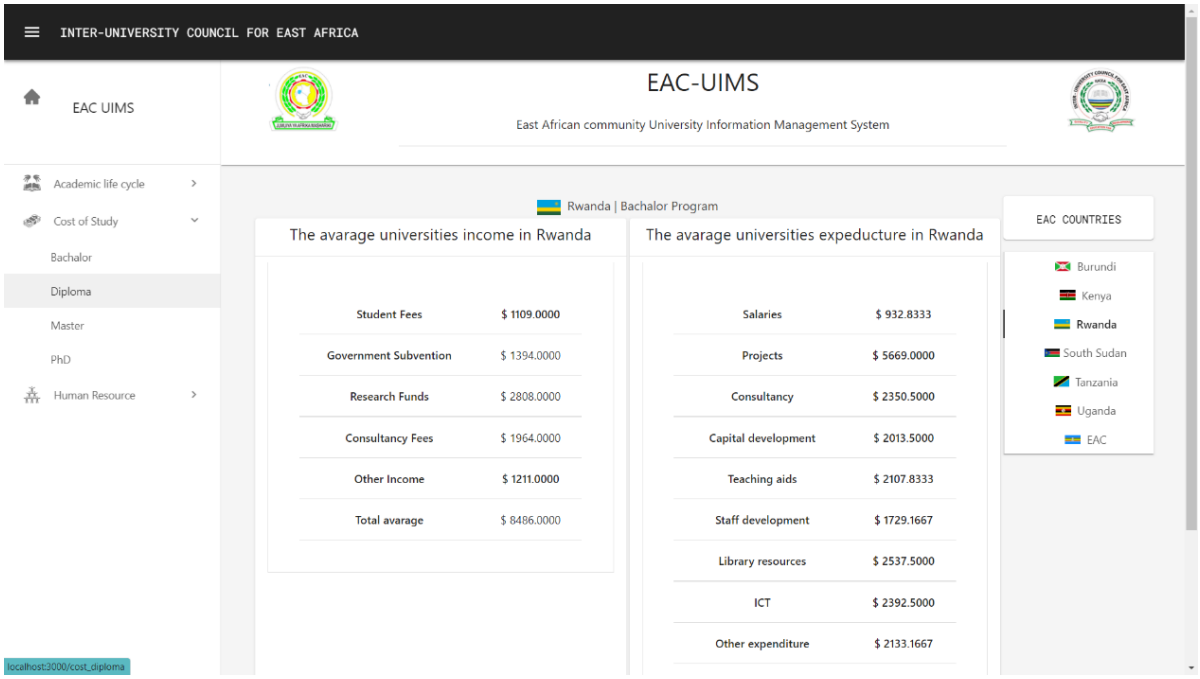


Figure 40: Cost study of diploma level in Rwanda

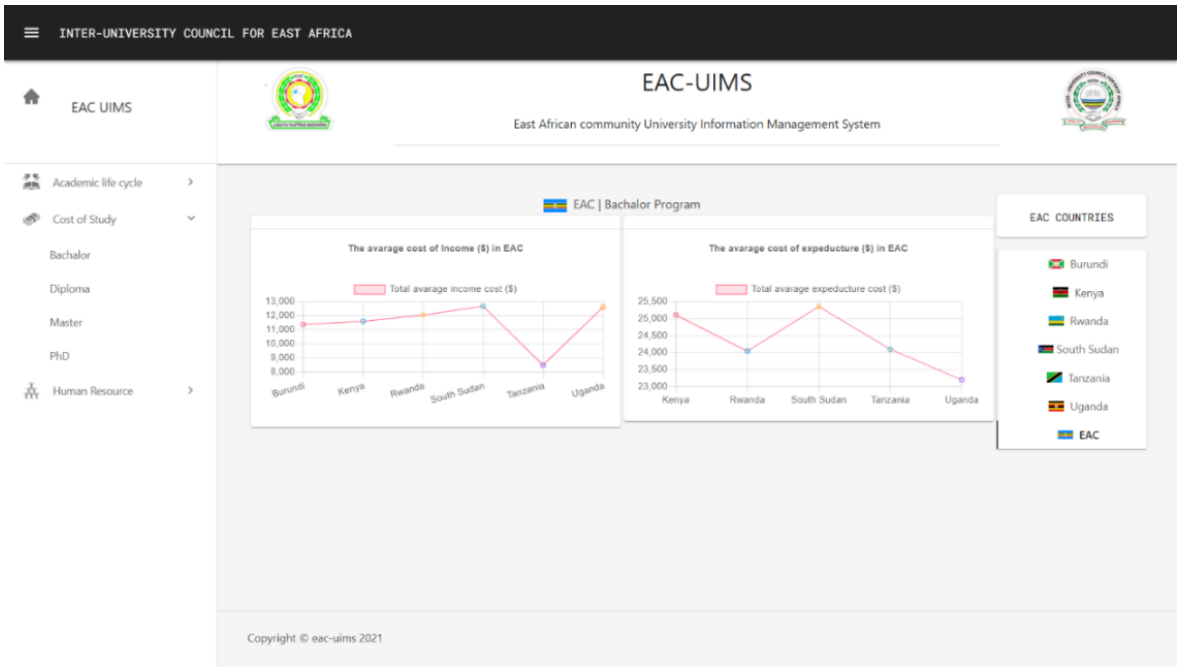


Figure 41: Comparison of the cost of the diploma level within EAC

The human resource section displays staff information on high education for each country of the EAC region by gender, area of specialization, and age group. The following figures display staff information in high education by gender, area of specialization, and age group.

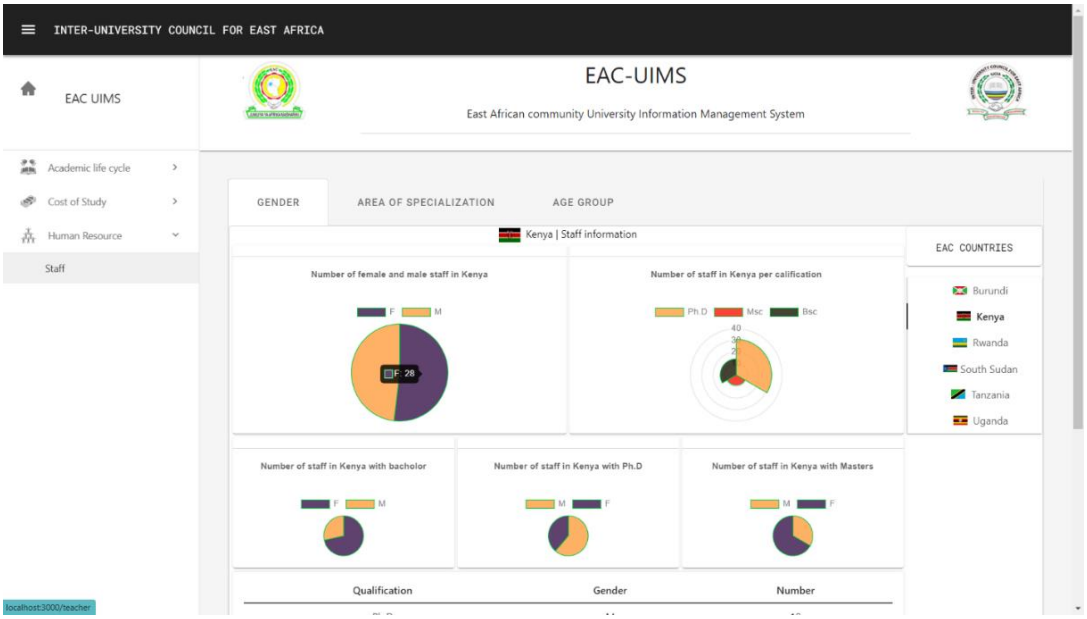


Figure 42: Staff information in high education in Kenya by gender

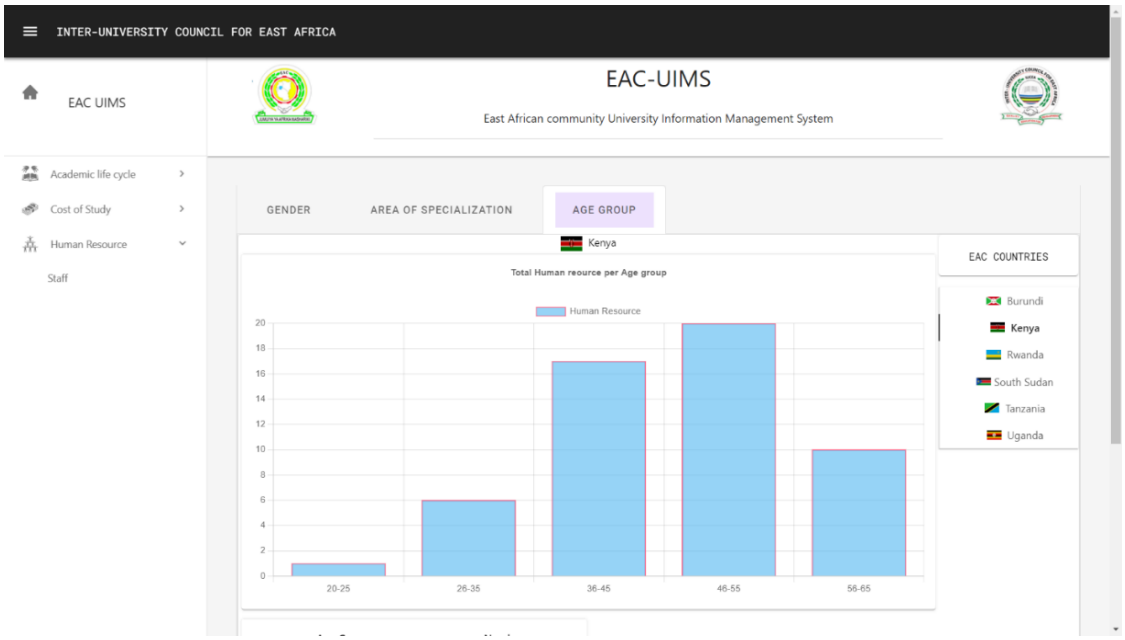


Figure 43: Staff information in high education in Kenya by age group

4.3.2 Overall Admin Design Section

The overall admin has all rights to the whole system and can do many things. Since EAC UIMS is the second module, the overall admin will be able to access even the first module, EAC PIMS, and other modules will come in the future to complete EAC HEIS. Moreover, an overall admin must be known as a superuser and have limited rights to different information. For example, the overall admin should have the right to create, read, delete, update and generate a

report about high education in East African Community region. Figure 43 to Fig. 46 present different interfaces and functionality of the overall admin portal.

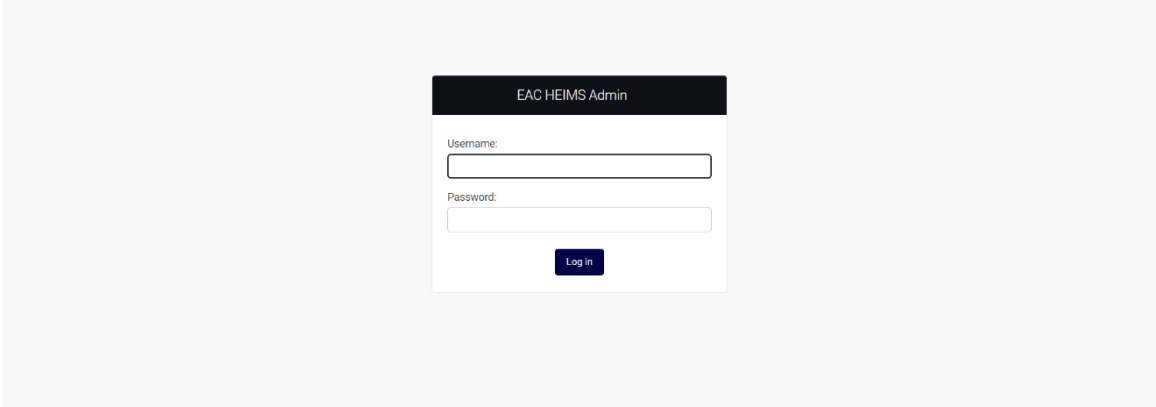


Figure 44: Authentication page

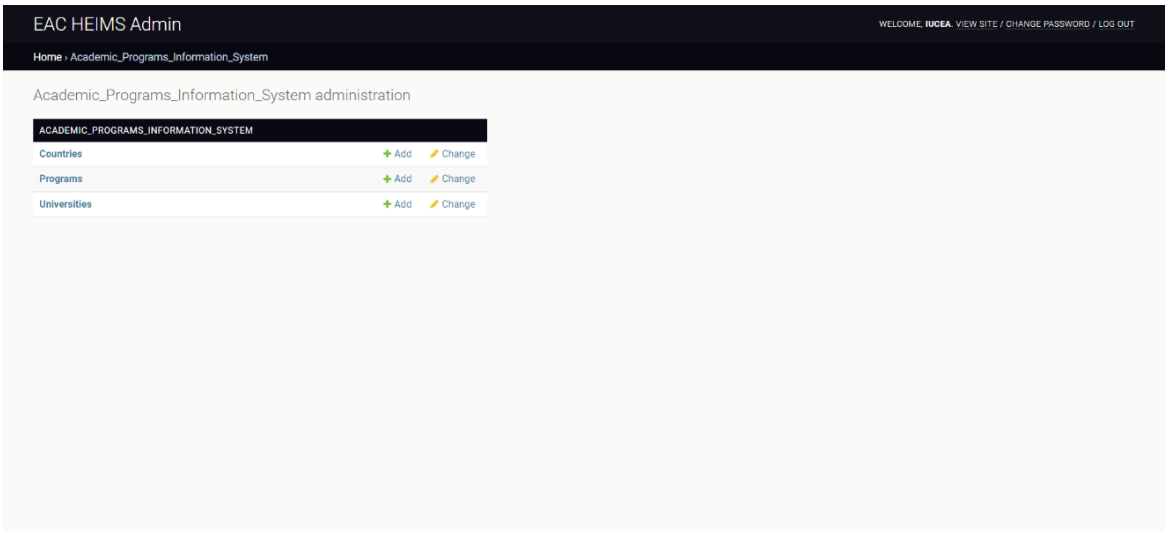


Figure 45: East Africa community APIS view page

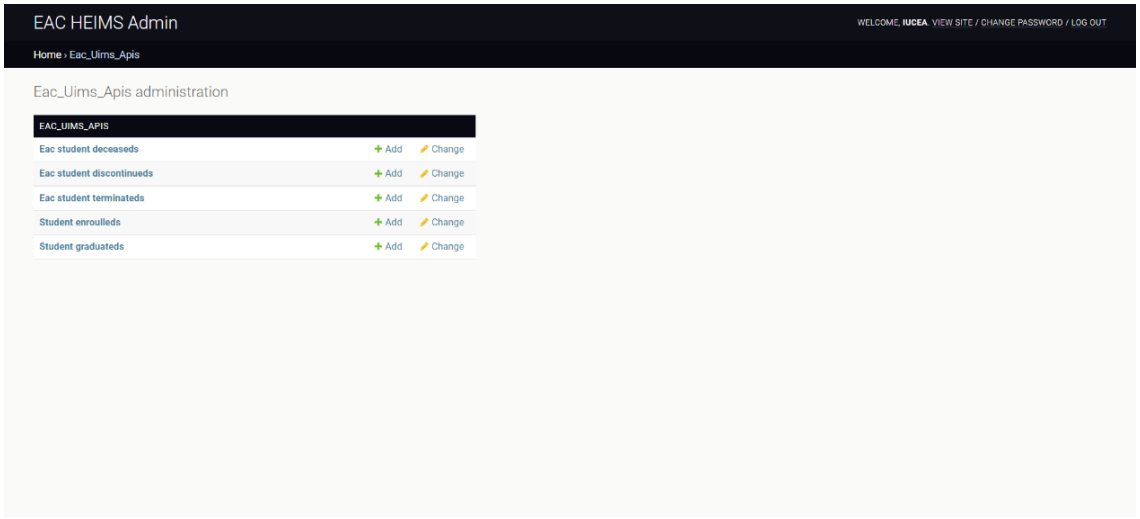


Figure 46: East Africa community UIMS view page

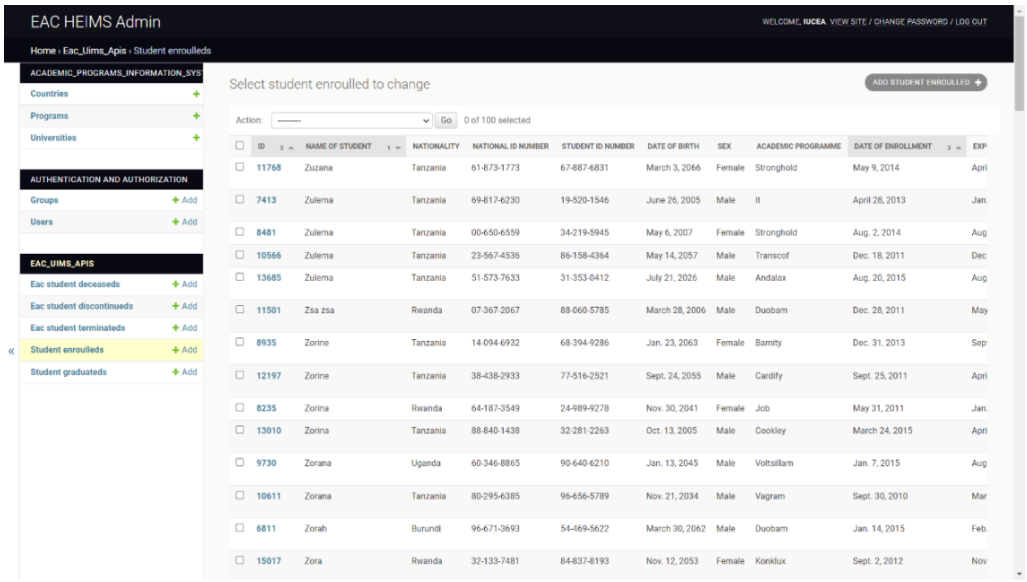


Figure 47: Students enrolled view page

4.3.3 University Application Portal Design Section

The Status of high education report reported that IUCEA works with 133 institutions. However, more than 133 institutions are in the EAC region. Since all data matter in the management of high education in the region, the EACUIMS provides a portal for the universities to apply for being among all degree-awarding higher education. This portal will receive all information about universities. Figure 40 presents the application portal.

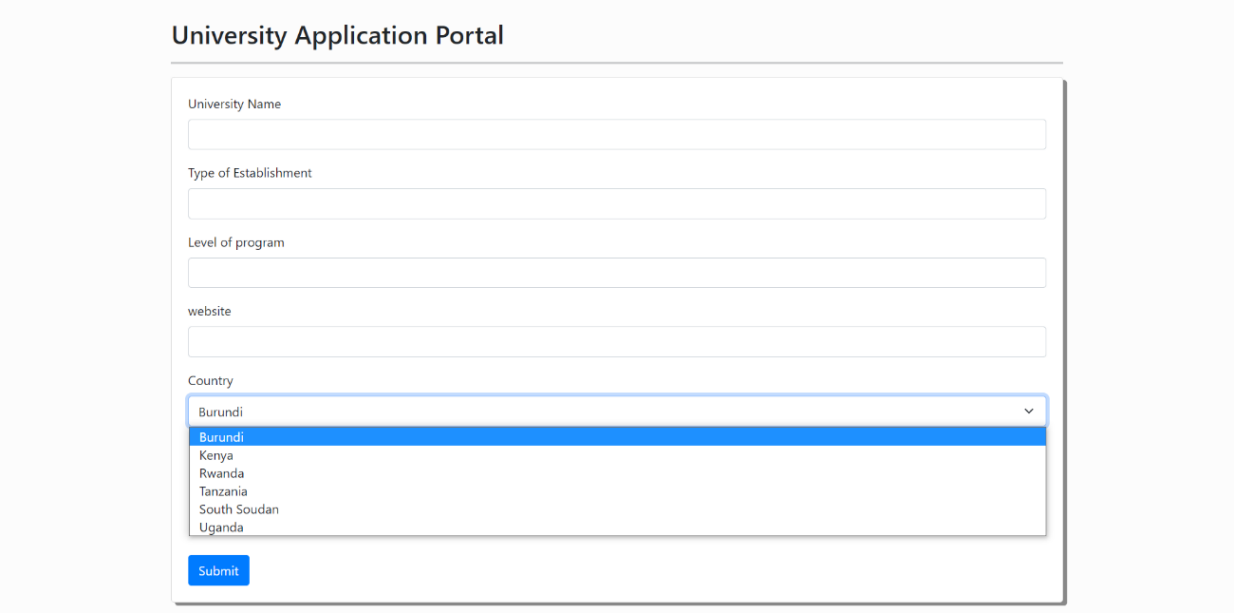


Figure 48: University application portal

4.3.4 University Portal Design Section

All degree-awarding high education should feed the system by submitting information about assets, academic life cycle, finance, and human resource. Of course, that information can also come from NCHE within each country. However, suppose that all that information will come directly from the universities; this section will present the university portal to submit all that information, one of the solutions proposed to resolve data sources issues for the proposed system. Otherwise, many propositions have been given of how the system will get valid information.

Start by university login page; all universities must be known before submitting the information to EAC UIMS. The proposed portal will allow all degree-awarding institutions to submit information about assets, academic life cycle, finance, and human resource. Figure 48 to Fig. 51 present different interfaces of the proposed system.

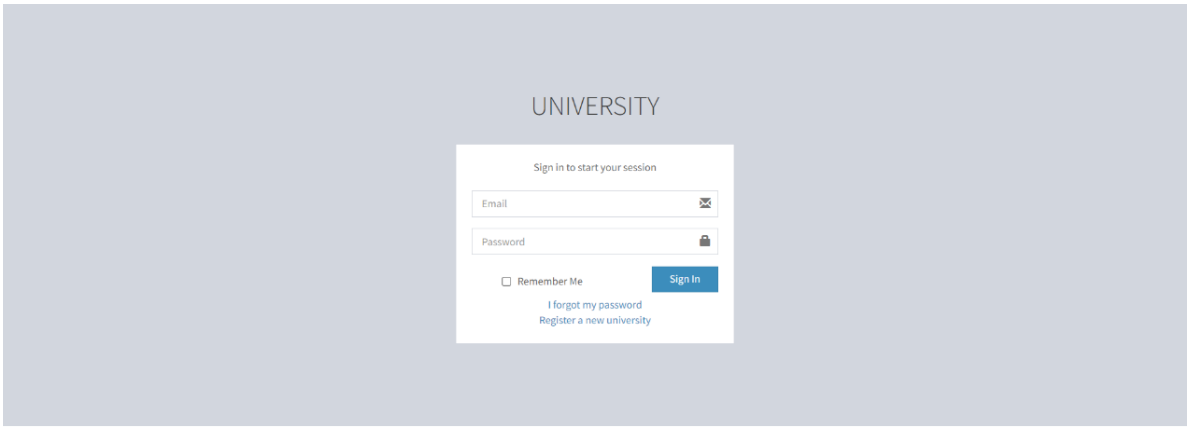


Figure 49: University login portal

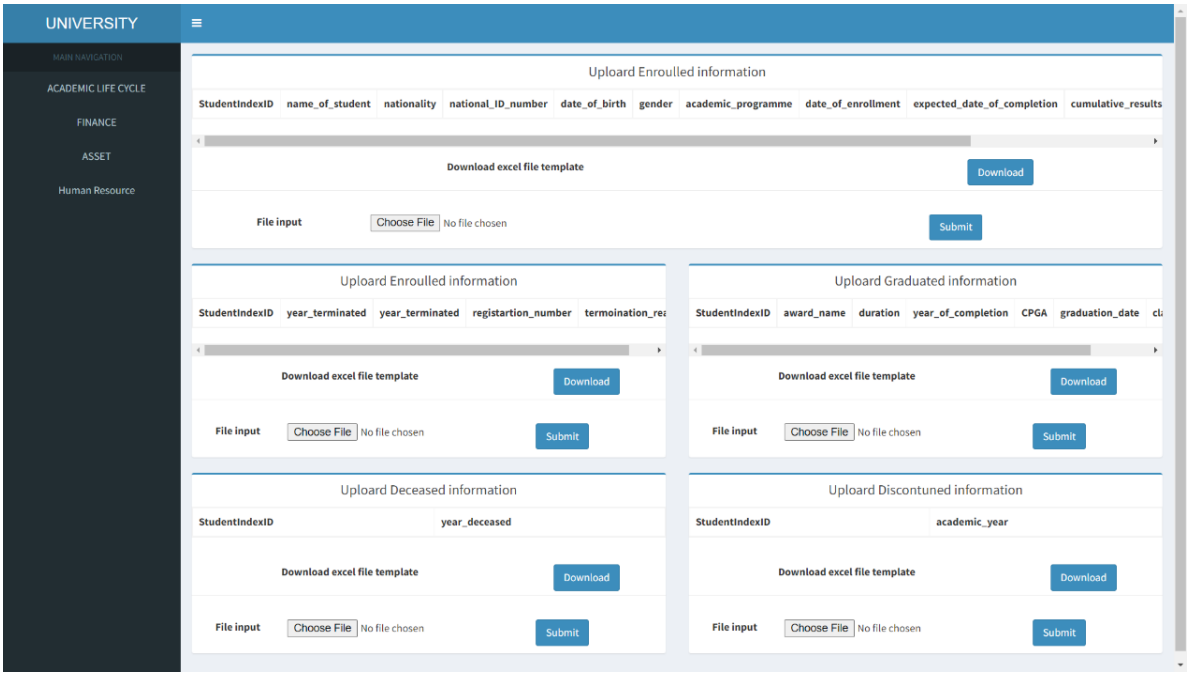


Figure 50: University's academic life cycle portal

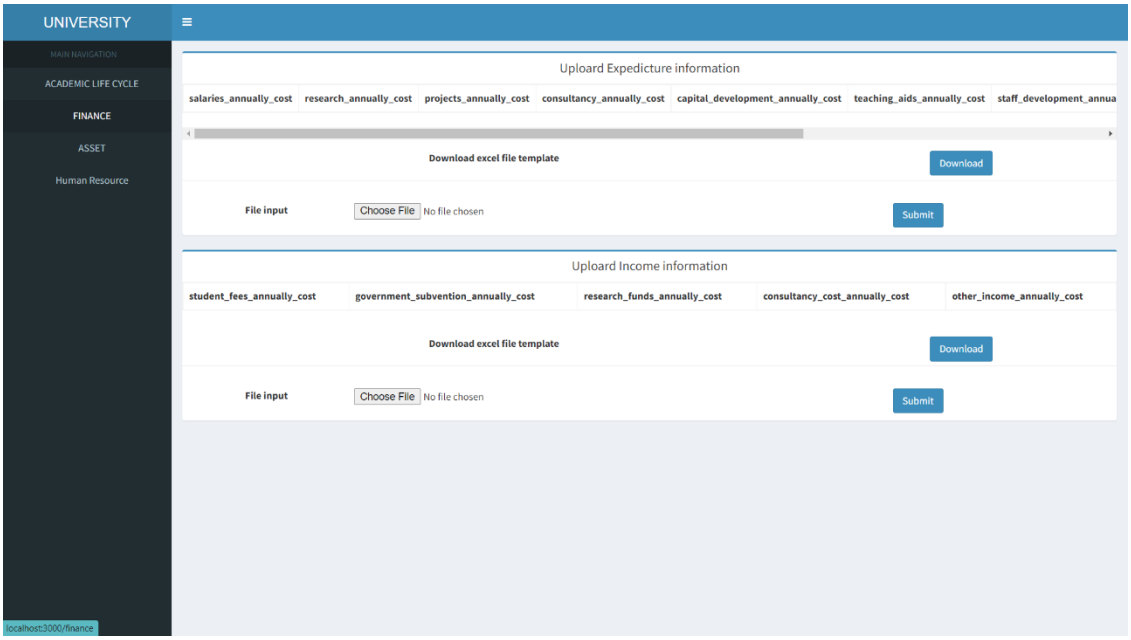


Figure 51: University's finance portal

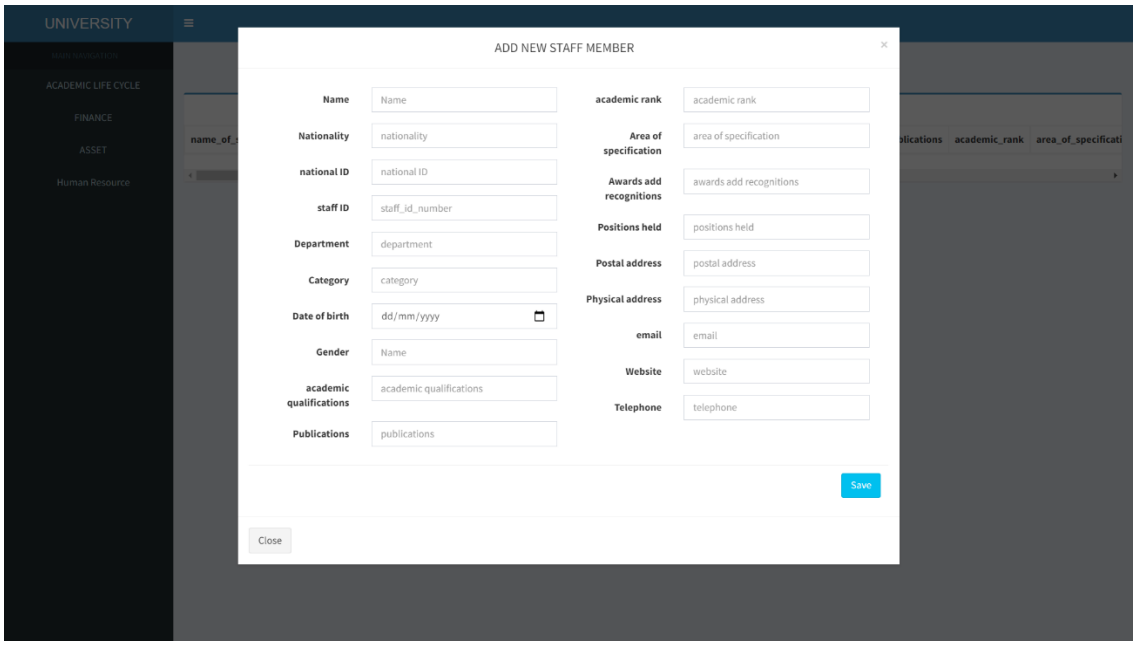


Figure 52: University's human resource (staff) management

4.4 System Requirement Evaluation

Table 7 describes, in brief, the accomplishment of the functional requirements in the developed system.

Table 7: System requirements verification

S/N	Requirements	System verification
1	The developed system should gather the university's academic life cycle, assets, finance, and human resource information.	MySQL database has been developed and connected to the RESTful API to communicate with the different clients.
2	The developed system must manipulate the received data and produce the analyzed report format.	The overall admin should generate a report about the East African Community region universities. A surfer should visualize data in graphic format.
3	The developed system must protect data successfully.	The database developed will store or retrieve data using web APIs. This approach will secure the system since the information will be accessed on different endpoints.
4	The developed system should be flexible and user-friendly.	The developed system has adopted the APIs development approach. In addition, the user experience design principle has been used to develop user-friendly interfaces.

4.5 Testing

Software verification processes include activities such as reviews, walkthroughs, and inspections. For example, the system verification exercise was performed by performing a walkthrough often performed by the ICT team at IUCEA to verify whether the developed software met the standards. Then, a validation involving several test phases was carried out to verify if the software requirements were met.

4.5.1 Unit Testing

Unit testing involves checking individual units or small components to check if they perform as expected (Runeson, 2006). Unit testing has been done on different components in this study, including the public dashboard, admin, and university application portal. Some examples are

the home page, filter, login to the admin dashboard, university application, university portal login, and other components.

4.5.2 Integration Testing

Integration testing as the ability to check the system's compatibility as a whole has been done on the developed system.

4.5.3 System Testing

System testing validates the fully integrated product against the requirements established for the system (Agarwal *et al.,* 2010). According to the specified requirements, the developed system was validated. The results of the validation process are summarized in Table 8.

Table 8: Table showing the results for system testing

Requirement	Description	Test Score
Management of the System	The Administrator shall manage the overall system, including the first module (EACAPIMS).	Passed
Filter	The public dashboard will allow the users to filter information about the academic life cycle, finance, and staff by East African Community's country and EAC.	Passed
View information about the academic life cycle, finance, and staff	The system shall display information about the academic life cycle, finance, and all degree-awarding high education staff.	Passed
University application	The system shall provide a portal for universities to apply for being among all degree-awarding high education worked with IUCEA.	Passed

4.5.4 User Acceptance Testing Results

User acceptance testing involves testing the developed system with dummy data to inspect that the system meets the requirements defined by the final report about EACHEIS. Under this, different levels of management had to test the developed system to determine whether or not the developed system is ready to be accepted for use and deployed in the client's environment. The respondents found the system impressive and could yield positive results. In addition, the system offered outstanding contributions such as a better management information system and better decision-making support.

The higher education experts of the IUCEA were clustered into 3 (three) clusters i.e. All Staff, ICT staff, and Quality assurance staff, the purpose of this clustering was to collect domain-specific data. For this study, we got 15 responses who are the staff of IUCEA and are fully involved in the harmonization of education and training systems in East Africa. At the time of data collection, the Quality Assurance Unit had one staff who was female, the ICT unit had two staff, and the questionnaire for all staff was filled by 12 respondents were by 41.7% were male, and 58.3 % were female.

Table 9 contains user acceptance testing results done by different users on the public dashboard interface, which helps the surfer know different information about universities in the EAC region.

Table 9: User acceptance testing results for the public dashboard

	Respondents			
	Strong agree (%)	**Agree (%)**	**Disagree (%)**	**Not sure (%)**
Web application is easy to use	75	25	0	0
The appearance of the web application is user friendly	100	0	0	0
The interfaces are well organized	75	25	0	0
The purpose of the system is well implemented	70	30	0	0
The system is comfortable to use	65	30	0	5

4.6 Discussion

The developed UIMS has fulfilled all requirements needed as reported by Prof. Venansis BARYAMUREEBA within the final report on the Status of the EACHEIS. The system is suitable to support higher education within the EAC region. Its suitability is the required goal for higher education, where IUCEA will use information about higher education from the EAC region. The developed system is the continuous development of EAC HEIS, which will help the IUCEA support higher education activities in the EAC region. The EAC HEIS as one stops live repository for information and data will allow IUCEA to generate regional reports for partner states and member universities of higher education status and dynamics. Moreover, EAC UIMS as a second module has successfully been developed to produce public and private data visualization reports.

However, the developed system is limited by the legal and policy since before implementation of the system, EAC and IUCEA should develop a legal framework. As reported, EAC should implement the EAC access to information and data law, and IUCEA should implement the IUCEA policy framework and EAC regional policy for higher education information and data management.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The project aimed to improve higher education in the EAC region by designing the second module of EACHEIS called EAC UIMS. Since EAC PIMS has been designed as the first module to deal with higher education programs in the EAC region, the EAC UIMS has been designed to manage the academic life cycle, finance, assets, and human resource within the universities in EAC region.

However, data needed by the developed system will come from NCHE and universities. The NCHE will help validate data from national universities, and all degrees awarding high education will share data needed by the system that are not in the NCHE databases. During the development, we did not contact a survey to know more about the different data-managed systems of different NCHE. Therefore, we assumed that all data needed by the system would come directly from all degree-awarding high education.

The developed system will help IUCEA manage higher education in the EAC region and all users to know the universities' public information. Considering the system's data where the universities upload the information about academics, finance, and human resource, the system allowed the public to visualize the academic life cycle, the cost of the study, and staff information statistically.

5.2 Recommendations

Web technology has been developed from web 0.0, known as the developing Internet, up to the current web 5.0, known as the Emotional web (Chakraborty *et al.*, 2014). IUCEA should put more effort into embracing technology and its IT-related supporting tools. By supporting the EACHEIS, there will be a more remarkable improvement in information management within the institution. The institution should also offer their employees training about the new ICT technology and related tools and software.

More features can be added to implement the functionality of the developed system. Therefore, the following can be conceded to improve the system design and quality:

(i) communication module between all degree-awarding high education and IUCEA may be integrated to support direct messages, group messages, or calls.

(ii) Machine learning algorithms can be implemented to allow the system to answer complex questions (Portugal *et al.*, 2018).

(iii) Since more users use mobile devices for different consumer services online (Schnauber-Stockmann & Karnowski, 2020), a mobile application version should be developed.

Due to the limitation of time, the developed system is still not yet implemented to the public's view, and dummy data still populate the database. IUCEA and other institutions in which the subject matter should push the implementation of all needed to start using the developed system.

As reported by the final report on the Status of the EAC HEIS the implementation of the EAC HEIS should follow rules and regulations regarding data sharing in the East African community region. The future work should consider deep analysis between two modules already developed deploy them and test the performance, design architecture, and scalability of the system.

REFERENCES

Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2017). Agile software development methods: Review and analysis. CoRR, abs/1709.08439. https://doi.org/10.48550/arXiv.1709.08439

Agarwal, B. B., Tayal, S. P., & Gupta, M. (2010). Software Engineering and Testing. *Computer Science Series*.

Al-Alwani, A. E. S. (2005). Barriers to integrating information technology in Saudi Arabia's science education. *The University of Kansas*.

Alemu, S. K. (2011). Analysis of Higher Education Cost Sharing Implementation in Ethiopia (Doctoral dissertation, SK Alemu).

Allen-Meares, P., & Lane, B. A. (1990). Social work practice: Integrating qualitative and quantitative data collection techniques. *Social Work*, *35*(5), 452-458. https://doi.org/10.1093/sw/35.5.452

Andreasen, M. M., Hansen, C. T., & Cash, P. (2015). Conceptual design. *Cham: Springer International Publishing*.

Astakhova, K. V, Korobeev, A. I., Prokhorova, V. V, Kolupaev, A. A., Vorotnoy, M. V, & Kucheryavaya, E. R. (2016). The role of education in the economic and social development of the country. *International Review of Management and Marketing*, *6*(1S), 53-58.

Ayoo, P. O. (2009). Reflections on the digital divide and its implications for the internationalization of higher education in a developing region: The case of East Africa. *Higher Education Policy*, *22*(3), 303–318.

Baariu, S. N., Mulaku, G. C., & Siriba, D. N. (2019). State of Cartographic Services among the East African Community Member States. *Journal of Geographic Information System*, *11*(01), 56. https://doi.org/10.4236/jgis.2019.111005

Bank, W. (2019). *Global Financial Development Report 2019/2020: Bank Regulation and Supervision a Decade after the Global Financial Crisis*. The World Bank.

Bass, B. M. (1997). Does the Transactional - Transformational Leadership Paradigm Transcend Organizational and National Boundaries? *American Psychologist*, *52*(2), 130. https://doi.org/10.1037/0003-066X.52.2.130

Berners-Lee, T., Cailliau, R., Luotonen, A., Nielsen, H. F., & Secret, A. (1994). The worldwide web. *Communications of the Association for Computing Machinery*, *37*(8), 76–82.

Bowen, G. A. (2009). Document analysis as a qualitative research method. *Qualitative Research Journal*, *9*(2), 27-40. https://doi.org/10.3316/QRJ0902027

Budde, R., Kautz, K., Kuhlenkamp, K., & Züllighoven, H. (1992). What is prototyping? *Information Technology & People*, 6(2), 89-95.

Chakraborty, A. K., Dey, S., & Dasgupta, G. (2014). Significance of Web 0.0-4.0, semantic web, cloud computing, and drupal in concomitant to library/information/knowledge centres management. *International Journal of Digital Library Systems*, *4*(2), 1–15.

Chouhan, L., Ghimire, S., Subrahmanyam, C., Miyasaka, T., & Biju, V. (2020). Synthesis, optoelectronic properties, and applications of halide perovskites. *Chemical Society Reviews*, *49*(10), 2869–2885.

Christudas, B. (2019). MySQL. *Practical Microservices Architectural Patterns*, 877–884. Springer.

Chung, L., Nixon, B. A., Yu, E., & Mylopoulos, J. (2012). Non-functional requirements in software engineering. *Springer Science & Business Media*, *5*, 476.

Clarkson, B., & Oliver, R. (2002). A Typology for Identifying Teachers' Progress in ICT uptake. *EdMedia+ Innovate Learning*, 317-322. *Association for the Advancement of Computing in Education.*

Cranfield, D. J., & Taylor, J. (2008). Knowledge management and higher education: A UK case study. *Electronic Journal of Knowledge Management*, *6*(2). https://cronfa.swan.ac.uk/Record/cronfa15052

Curie, D. H., Jaison, J., Yadav, J., & Fiona, J. R. (2019). Analysis of Web Frameworks. *Journal of Physics: Conference Series*, *1362*(1), 12114.

Daniel, B. K. (2017). *Big data in higher education: The big picture*. Springer, 19–28.

Davidson, E. J. (1999). Joint application design in practice. *Journal of Systems and Software, 45*(3), 215–223.

Economist, T. (2017). The world's most valuable resource is no longer oil, but data. *The Economist: New York, NY, USA*.

Ertmer, P. A., & Hruskocy, C. (1999). Impacts of a university-elementary school partnership designed to support technology integration. *Educational Technology Research and Development, 47*(1), 81–96.

Forcier, J., Bissex, P., & Chun, W. J. (2008). Python web development with Django. *Addison-Wesley Professional*.

Gackenheimer, C., & Paul, A. (2015). *Introduction to React, 52*. Springer.

Gill, P., Stewart, K., Treasure, E., & Chadwick, B. (2008). Methods of data collection in qualitative research: interviews and focus groups. *British Dental Journal, 204*(6), 291–295.

Glinz, M. (2007). On non-functional requirements. *15th IEEE International Requirements Engineering Conference (RE 2007)*, 21–26. https://doi.org/10.1109/RE.2007.45

Goldstein, J. (2008). Embracing "Open Access" in East Africa: A common Internet infrastructure policy agenda for human security and economic development. *Journal of Public and International Affairs, 19*, 139–150.

Gomaa, H. (2011). *Software modeling and design: UML, use cases, patterns, and software architectures*. Cambridge University Press.

Gralla, P. (1998). *How the Internet works*. Que Publishing.

Henning, M. (2007). API: Design Matters: Why changing APIs might become a criminal offense. *Queue, 5*(4), 24–36.

Hitzler, P., Krotzsch, M., & Rudolph, S. (2009). *Foundations of semantic web technologies*. Chapman and Hall/CRC. https://doi.org/10.1201/9781420090512

Jensen, S. H., Møller, A., & Thiemann, P. (2009). Type analysis for JavaScript. *International Static Analysis Symposium*, *5673*, 238–255. Springer, Berlin, Heidelberg

Kuyok, K. A. (2017). Higher Education in South Sudan: Living with Challenges. *International Higher Education*, (89), 16–18. http://dx.doi/org/10.6017/ihe.2017.89.9766

Luck, M., McBurney, P., & Preist, C. (2004). A manifesto for agent technology: Towards next-generation computing. *Autonomous Agents and Multi-Agent Systems*, *9*(3), 203–252.

Magara, E. (2006). *A framework for an integrated student information management system for higher education in Uganda*, *68*(05).

Malecha, G., Morrisett, G., Shinnar, A., & Wisnesky, R. (2010). Toward a verified relational database management system. *Proceedings of the 37th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, 237–248. https://doi.org/10.1145/1706299.1706329

Meier-Brügger, M. (2010). *Indogermanische Sprachwissenschaft*. Berlin, New York: De Gruyter. https://doi.org/10.1515/9783110251449

Murage, D., Mwangi, M., Kaijage, E., & Ochieng, D. E. (2019). Tax incentives and foreign direct investment relationship in the East African Community partner states. *Database Administrator Africa Management Review*, *9*(3), 57-67.

Ng'ingo, G. A. (2019). *Assessment of the Implementation of EAC Common Market Protocol on Free Movement of Labor in Tanzania*. Mzumbe University.

O'leary, Z. (2017). *The essential guide to doing your research project*. Sage.

Ochwa-Echel, J. (2016). Private universities in Uganda: Issues and challenges. *International Journal of Education and Social Science, 3*(3), 7-18.

Ogachi, I. O. (2018). The East African higher education area: A global or regional higher education space? *FIRE: Forum for International Research in Education*, *4*(3), 56-76. https://doi.org/10.32865/fire2018439

Ogachi, O. (2009). Internationalization vs regionalization of higher education in East Africa and the challenges of quality assurance and knowledge production. *Higher Education*

Policy, *22*(3), 331–347.

Oketch, M., Mutisya, M., Ngware, M., Ezeh, A. C., & Epari, C. (2008). Pupil school mobility in Urban Kenya. *Nairobi: African Health Research Center Working Paper, 38*

Paternò, F., & Santoro, C. (2019). End-user development for personalizing applications, things, and robots. *International Journal of Human-Computer Studies*, *131*, 120–130. https://doi.org/10.1016/j.ijhcs.2019.06.002

Patni, S. (2017). *Pro RESTful APIs*. Springer.

Pellaud, F., Ndiaye, A., Khushik, F., & Diemer, A. (2019). Environmental education to education for sustainable development: challenges and issues. *International Journal of Humanities and Social Science*, *9*(1), 1–14.

Portugal, I., Alencar, P., & Cowan, D. (2018). The use of machine learning algorithms in recommender systems: A systematic review. *Expert Systems with Applications*, *97*, 205–227. https://doi.org/10.1016/j.eswa.2017.12.020

Rawlinson, J. G. (2017). *Creative thinking and brainstorming*. Routledge. https://doi.org/10.4324/9781315259000

Richardson, R. C., Bracco, K. R., Callan, P. M., & Finney, J. E. (1998). *Designing State Higher Education Systems for a New Century. Series on Higher Education.* Oryx Press, PO Box 33889, Phoenix, AZ 85067-3889.

Robillard, M. P. (2009). What makes APIs hard to learn? Answers from developers. *IEEE Software*, *26*(6), 27–34. https://doi.org/10.1109/MS.2009.193

Runeson, P. (2006). A survey of unit testing practices. *IEEE Software*, *23*(4), 22–29. https://doi.org/10.1109/MS.2006.91

Saks, E. (2019). *JavaScript Frameworks: Angular vs React vs Vue.*

Salehi-Isfahani, D. (2012). Education, jobs, and equity in the Middle East and North Africa. *Comparative Economic Studies*, *54*(4), 843–861.

Schnauber-Stockmann, A., & Karnowski, V. (2020). Mobile devices as tools for media and communication research: A scoping review on collecting self-report data in repeated

measurement designs. *Communication Methods and Measures*, *14*(3), 145–164. https://doi.org/10.1080/19312458.2020.1784402

Tatnall, A., Kereteletswe, O., & Visscher, A. (2011). *Information technology and managing quality education*. Springer.

Way, T., Chandrasekhar, S., & Murthy, A. (2009). The Agile Research Penultimatum. *Software Engineering Research and Practice*, 530–536.

Wiseman, A. W., & Anderson, E. (2012). ICT-integrated education and national innovation systems in the Gulf Cooperation Council (GCC) countries. *Computers and Education*, *59*(2), 607-618. https://doi.org/10.1016/j.compedu.2012.02.006

World Bank, W. B. (2014). *LAC Poverty and Labor Brief, February 2014: Social Gains in the Balance-A Fiscal Policy Challenge for Latin America and the Caribbean*. The World Bank.

Yaya, S., Zegeye, B., Idriss-Wheeler, D., & Shibre, G. (2020). Inequalities in cesarean section in Burundi: evidence from the Burundi demographic and health surveys (2010–2016). *BMC Health Services Research*, *20*(1), 1–8.

Zina, O. (2021). *The essential guide to doing your research project*. Sage.

APPENDICES

Appendix 1: Sample interview questions

1. Does the system go to exchange data with the existing HEIS of institutions?
2. Does the data format managed by the system aggregated or disaggregated?
3. Did the existing HEC go to exchange data with the system?
4. Does the system go to manage structured data or unstructured data?
5. Which data collection method was used by the consultant to collect data used to write the report about the system?
6. Does the developed module (first module of EACHEIS, EAC APIMS developed by Alban MANISHIMWE) online or live for the use of IUCEA? If not, why?

Appendix 2: Sample questions for user acceptance validation

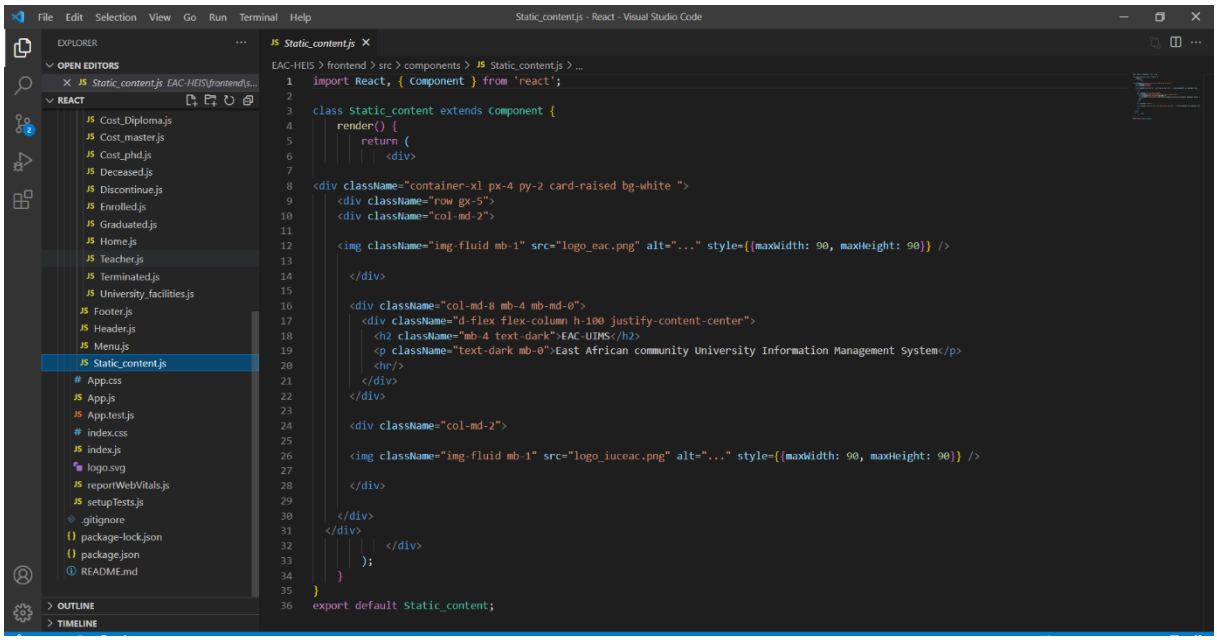
	Respondents			
	Strong agree (%)	**Agree (%)**	**Disagree (%)**	**Not sure (%)**
The web application is easy to use				
The appearance of the web application is user friendly				
The interfaces are well organized				
The purpose of the system is well implemented.				
The system is comfortable to use.				

Appendix 3: Sample questions for system testing results

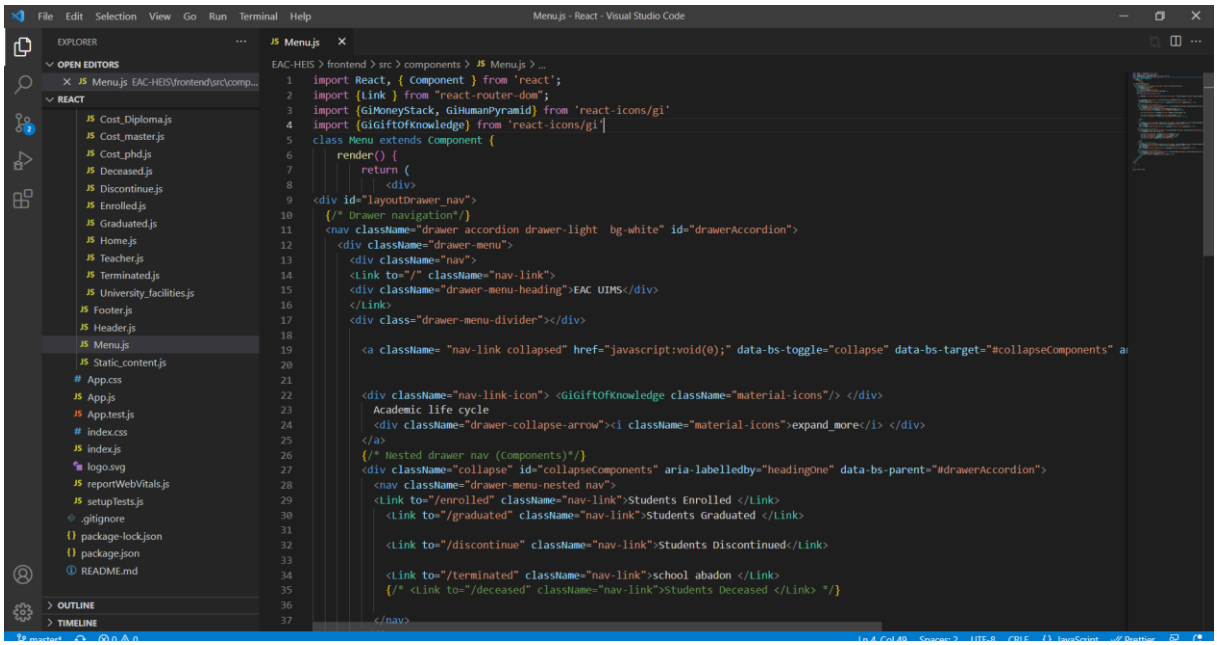
Requirement	Description	Test Score
Management of the System	The Administrator shall manage the overall system, including the first module (EACAPIMS).	
Filter	The public dashboard will allow the users to filter information about the academic life cycle, finance, and staff by East African Community's country and EAC.	
View information about the academic life cycle, finance, and staff	The system shall display information about the academic life cycle, finance, and all degree-awarding high education staff.	
University application	The system shall provide a portal for universities to apply for being among all degree-awarding high education worked with IUCEA.	

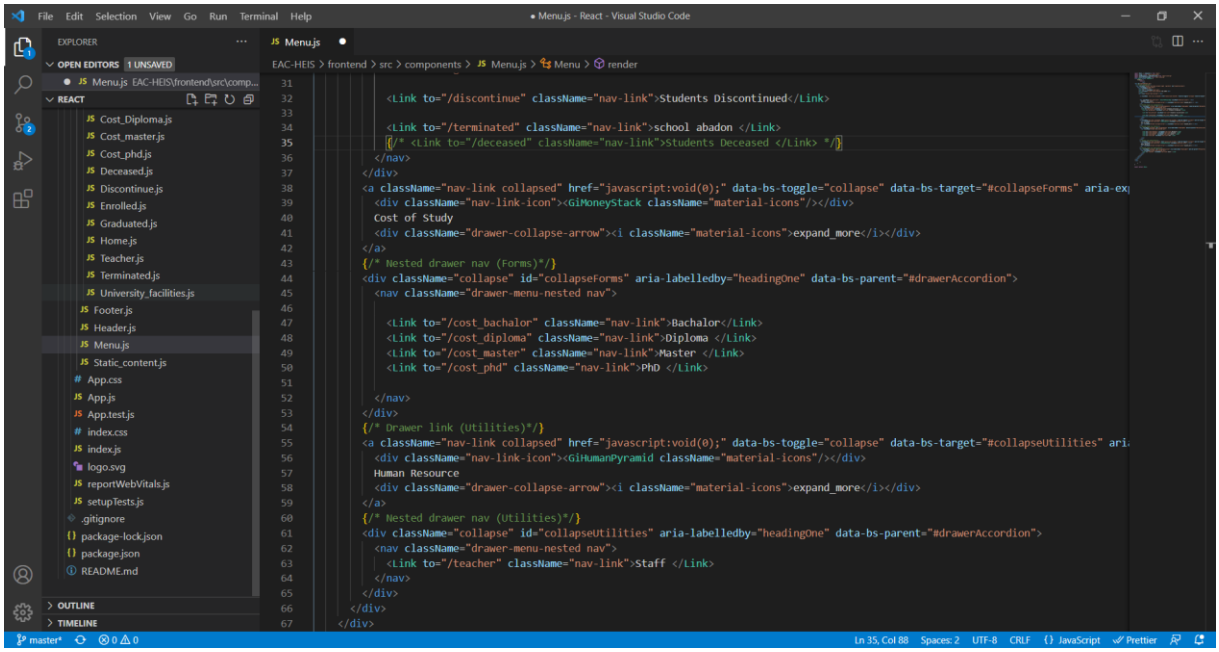
Appendix 4: Code of different components of the homepage of the public dashboard

a. Static Content

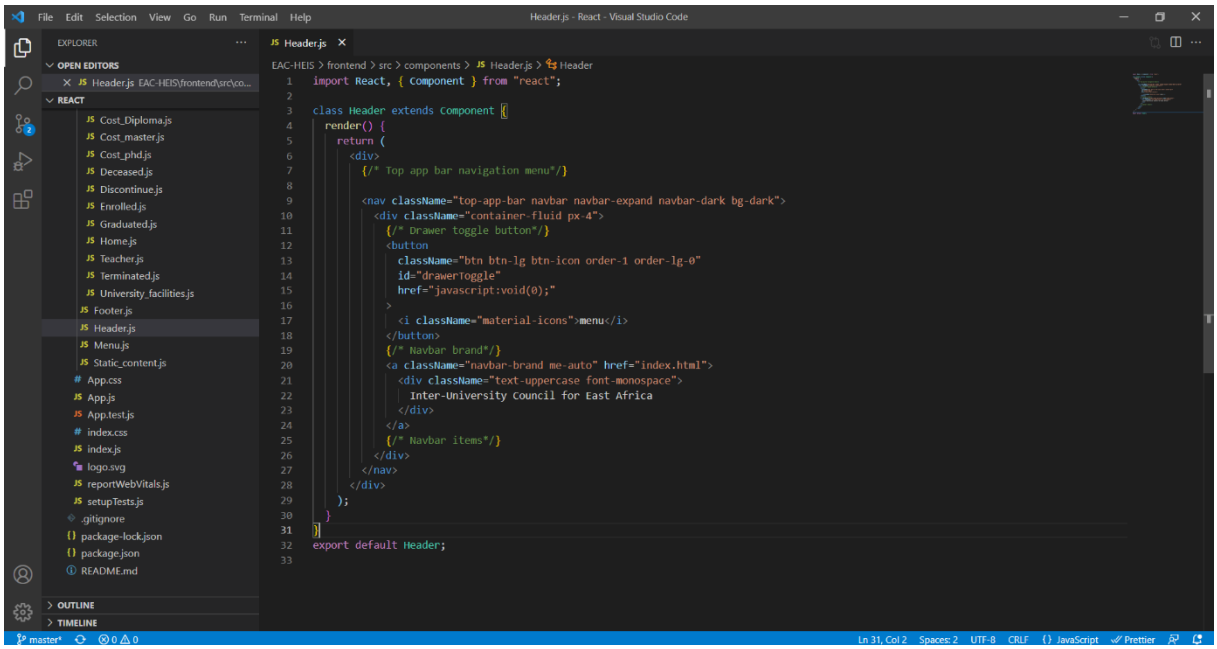


b. Menu

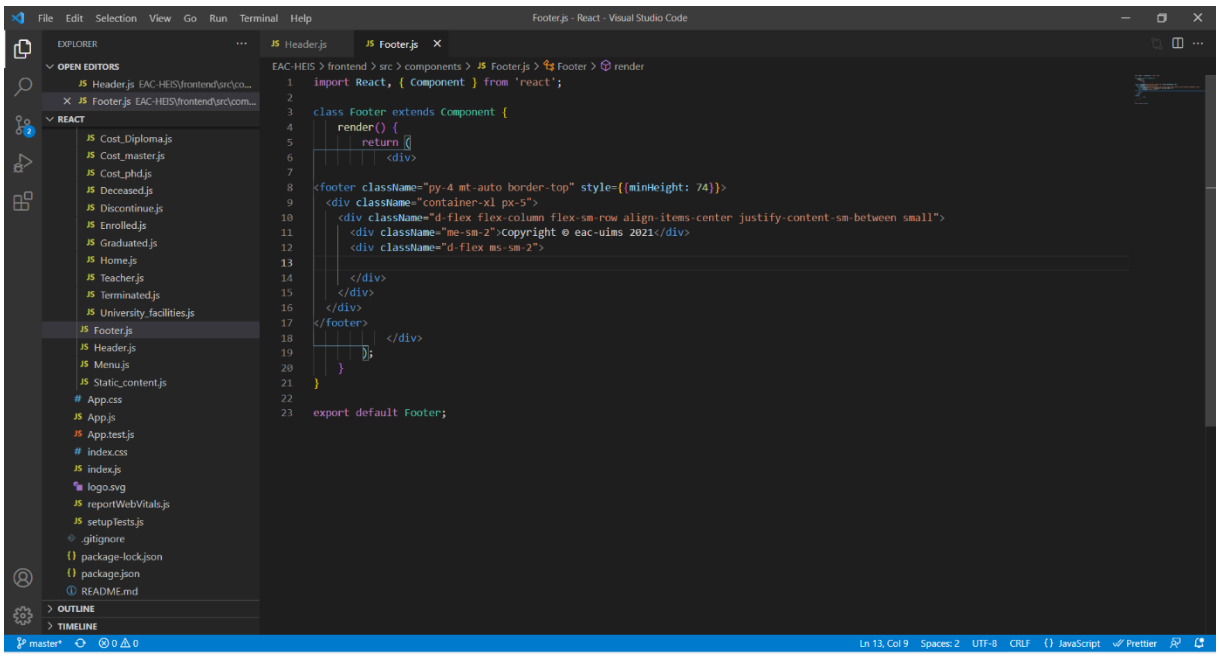




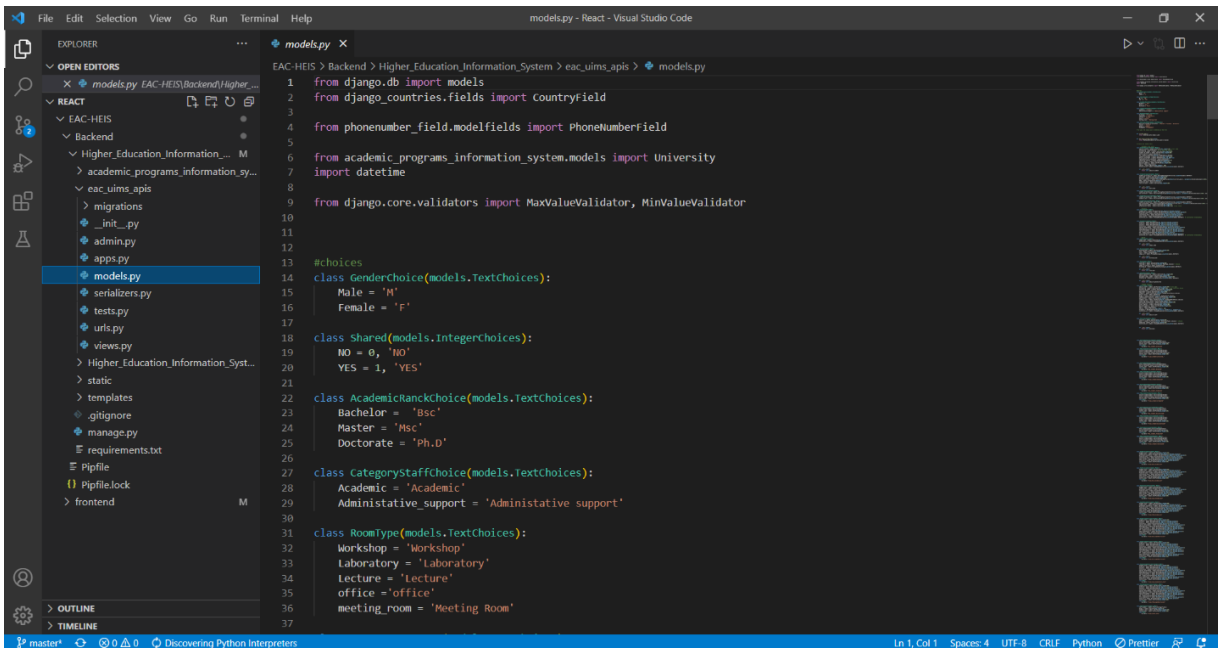
c. Header Component



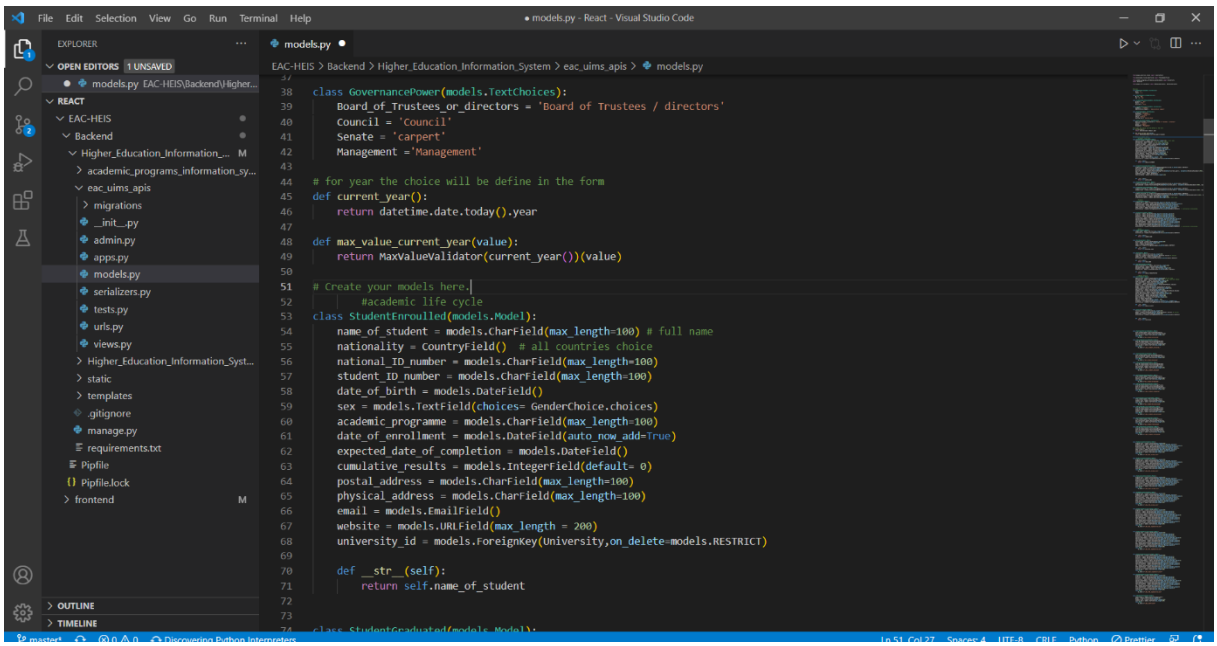
d. Footer component



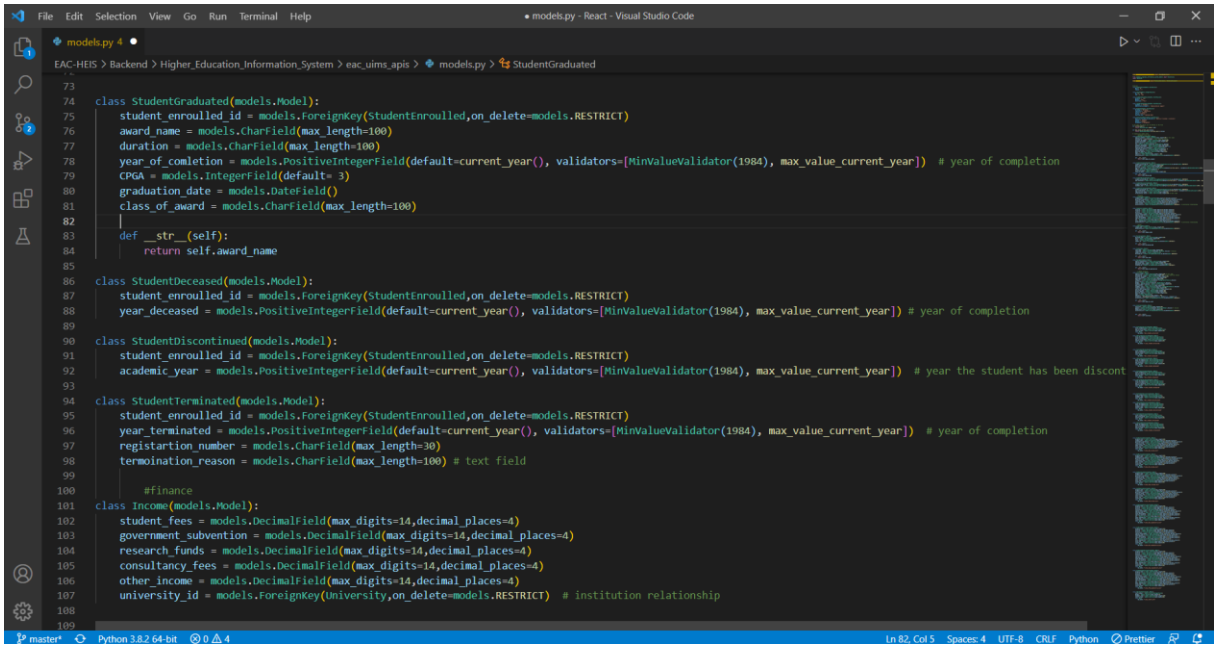
Appendix 5: Code of Models which generates a database using ORM technology



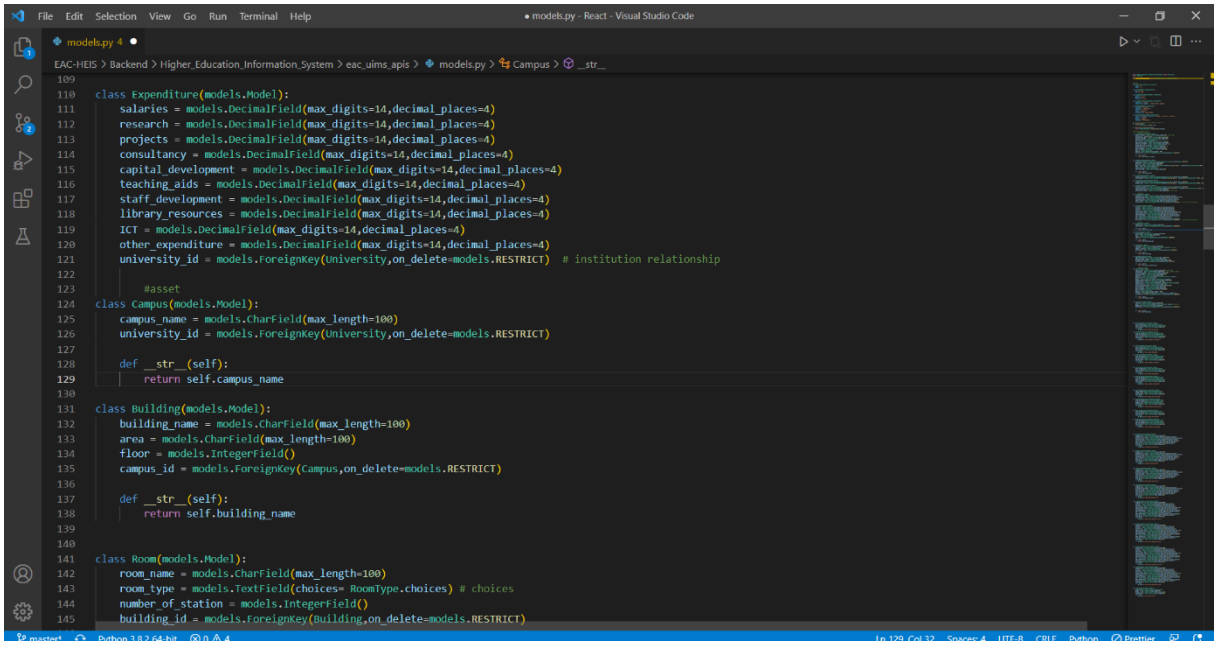
```python
from django.db import models
from django_countries.fields import CountryField

from phonenumber_field.modelfields import PhoneNumberField

from academic_programs_information_system.models import University
import datetime

from django.core.validators import MaxValueValidator, MinValueValidator



#choices
class GenderChoice(models.TextChoices):
    Male = 'M'
    Female = 'F'

class Shared(models.IntegerChoices):
    NO = 0, 'NO'
    YES = 1, 'YES'

class AcademicRanckChoice(models.TextChoices):
    Bachelor = 'Bsc'
    Master = 'Msc'
    Doctorate = 'Ph.D'

class CategoryStaffChoice(models.TextChoices):
    Academic = 'Academic'
    Administative_support = 'Administative support'

class RoomType(models.TextChoices):
    Workshop = 'Workshop'
    Laboratory = 'Laboratory'
    Lecture = 'Lecture'
    office ='office'
    meeting_room = 'Meeting Room'
```



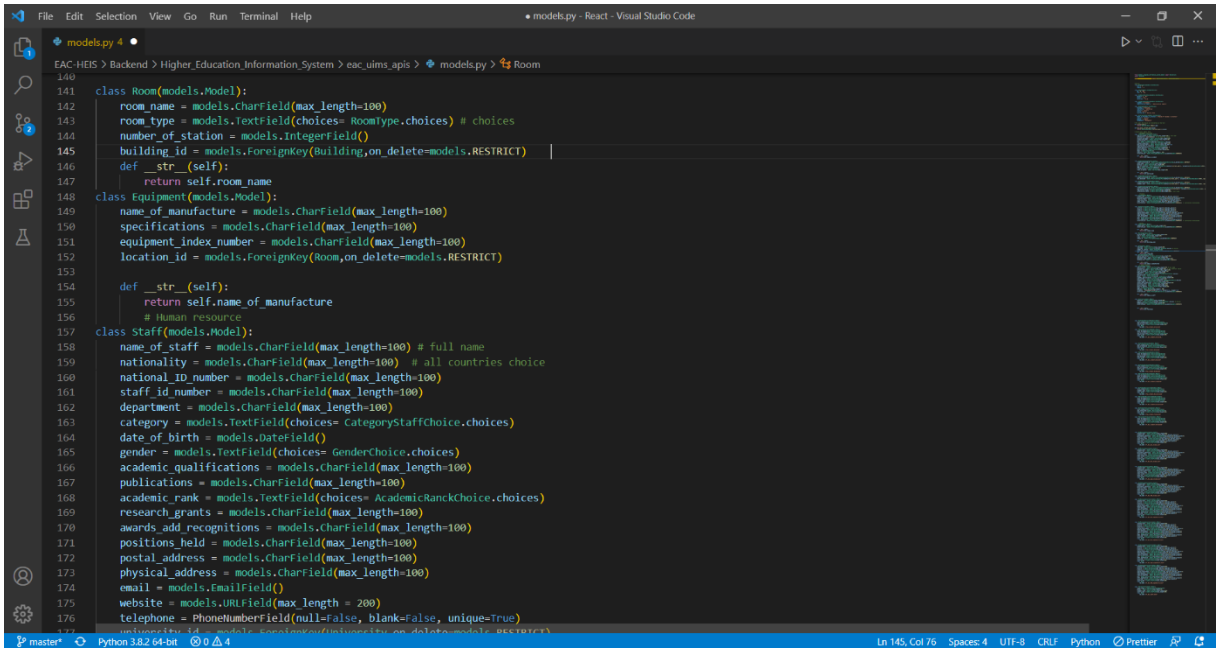
```python
class GovernancePower(models.TextChoices):
    Board_of_Trustees_or_directors = 'Board of Trustees / directors'
    Council = 'Council'
    Senate = 'carpert'
    Management ='Management'

# for year the choice will be define in the form
def current_year():
    return datetime.date.today().year

def max_value_current_year(value):
    return MaxValueValidator(current_year())(value)

# Create your models here.
        #academic life cycle
class StudentEnroulled(models.Model):
    name_of_student = models.CharField(max_length=100) # full name
    nationality = CountryField()  # all countries choice
    national_ID_number = models.CharField(max_length=100)
    student_ID_number = models.CharField(max_length=100)
    date_of_birth = models.DateField()
    sex = models.TextField(choices= GenderChoice.choices)
    academic_programme = models.CharField(max_length=100)
    date_of_enrollment = models.DateField(auto_now_add=True)
    expected_date_of_completion = models.DateField()
    cumulative_results = models.IntegerField(default= 0)
    postal_address = models.CharField(max_length=100)
    physical_address = models.CharField(max_length=100)
    email = models.EmailField()
    website = models.URLField(max_length = 200)
    university_id = models.ForeignKey(University,on_delete=models.RESTRICT)

    def __str__(self):
        return self.name_of_student
```



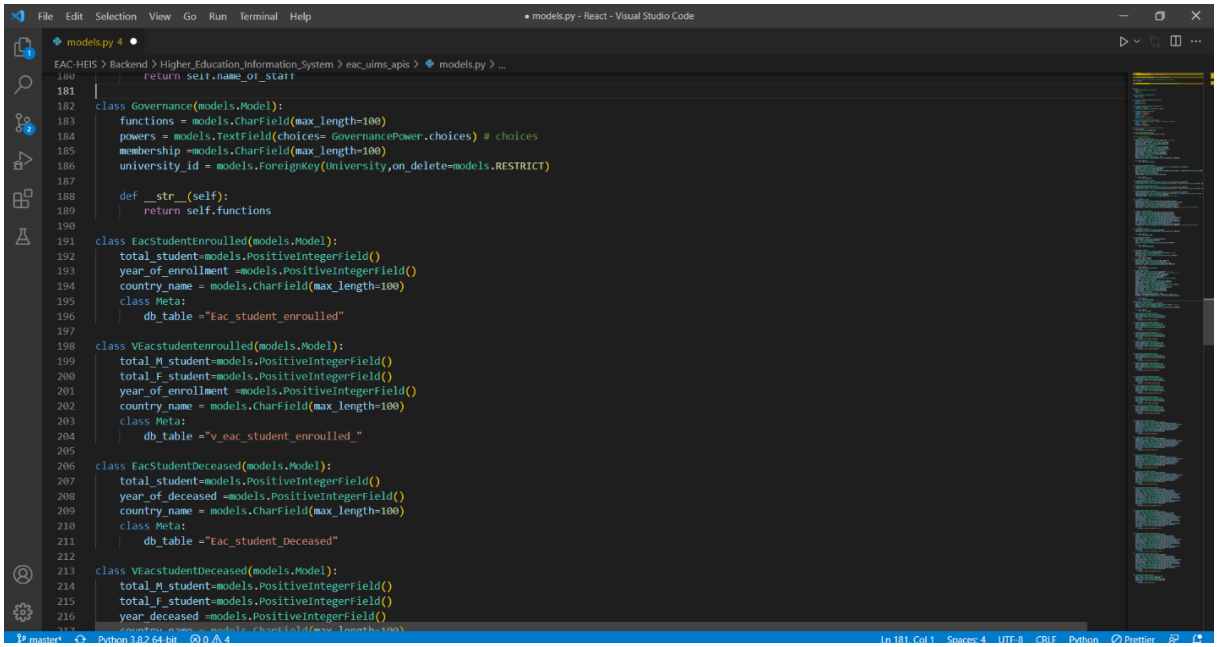
```
class StudentGraduated(models.Model):
    student_enroulled_id = models.ForeignKey(StudentEnroulled,on_delete=models.RESTRICT)
    award_name = models.CharField(max_length=100)
    duration = models.CharField(max_length=100)
    year_of_comletion = models.PositiveIntegerField(default=current_year(), validators=[MinValueValidator(1984), max_value_current_year])  # year of completion
    CPGA = models.IntegerField(default= 3)
    graduation_date = models.DateField()
    class_of_award = models.CharField(max_length=100)

    def __str__(self):
        return self.award_name

class StudentDeceased(models.Model):
    student_enroulled_id = models.ForeignKey(StudentEnroulled,on_delete=models.RESTRICT)
    year_deceased = models.PositiveIntegerField(default=current_year(), validators=[MinValueValidator(1984), max_value_current_year]) # year of completion

class StudentDiscontinued(models.Model):
    student_enroulled_id = models.ForeignKey(StudentEnroulled,on_delete=models.RESTRICT)
    academic_year = models.PositiveIntegerField(default=current_year(), validators=[MinValueValidator(1984), max_value_current_year])  # year the student has been discont

class StudentTerminated(models.Model):
    student_enroulled_id = models.ForeignKey(StudentEnroulled,on_delete=models.RESTRICT)
    year_terminated = models.PositiveIntegerField(default=current_year(), validators=[MinValueValidator(1984), max_value_current_year])  # year of completion
    registartion_number = models.CharField(max_length=30)
    termoination_reason = models.CharField(max_length=100) # text field

        #finance
class Income(models.Model):
    student_fees = models.DecimalField(max_digits=14,decimal_places=4)
    government_subvention = models.DecimalField(max_digits=14,decimal_places=4)
    research_funds = models.DecimalField(max_digits=14,decimal_places=4)
    consultancy_fees = models.DecimalField(max_digits=14,decimal_places=4)
    other_income = models.DecimalField(max_digits=14,decimal_places=4)
    university_id = models.ForeignKey(University,on_delete=models.RESTRICT)  # institution relationship
```



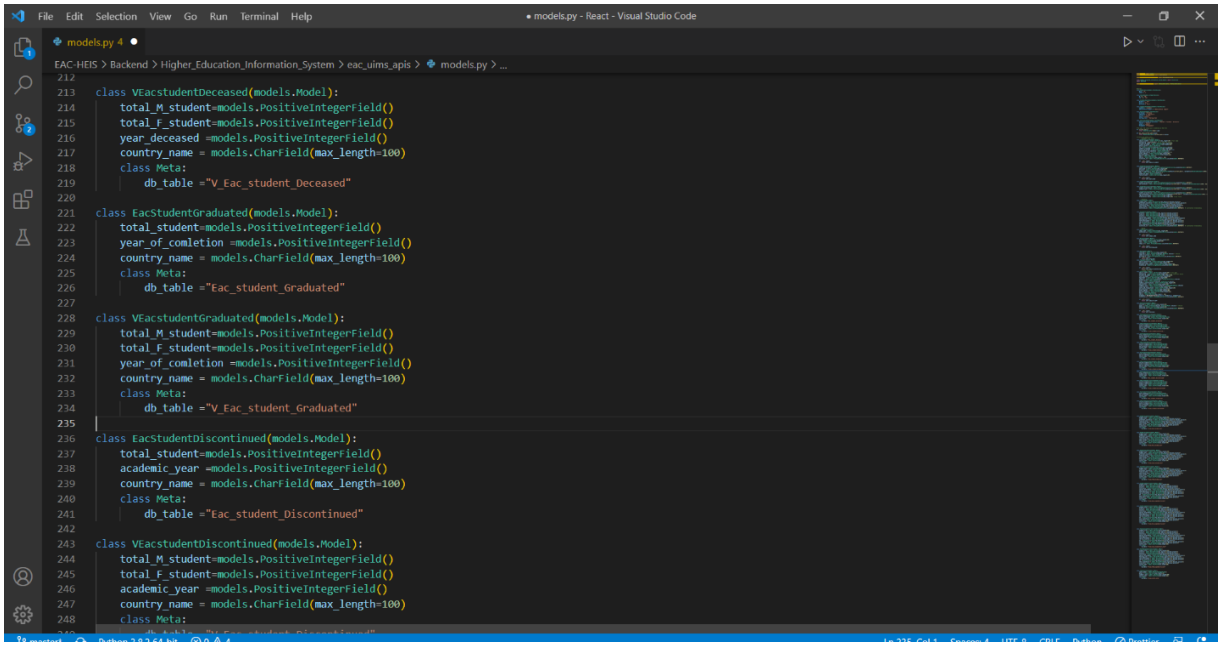
```
class Expenditure(models.Model):
    salaries = models.DecimalField(max_digits=14,decimal_places=4)
    research = models.DecimalField(max_digits=14,decimal_places=4)
    projects = models.DecimalField(max_digits=14,decimal_places=4)
    consultancy = models.DecimalField(max_digits=14,decimal_places=4)
    capital_development = models.DecimalField(max_digits=14,decimal_places=4)
    teaching_aids = models.DecimalField(max_digits=14,decimal_places=4)
    staff_development = models.DecimalField(max_digits=14,decimal_places=4)
    library_resources = models.DecimalField(max_digits=14,decimal_places=4)
    ICT = models.DecimalField(max_digits=14,decimal_places=4)
    other_expenditure = models.DecimalField(max_digits=14,decimal_places=4)
    university_id = models.ForeignKey(University,on_delete=models.RESTRICT)  # institution relationship

        #asset
class Campus(models.Model):
    campus_name = models.CharField(max_length=100)
    university_id = models.ForeignKey(University,on_delete=models.RESTRICT)

    def __str__(self):
        return self.campus_name

class Building(models.Model):
    building_name = models.CharField(max_length=100)
    area = models.CharField(max_length=100)
    floor = models.IntegerField()
    campus_id = models.ForeignKey(Campus,on_delete=models.RESTRICT)

    def __str__(self):
        return self.building_name


class Room(models.Model):
    room_name = models.CharField(max_length=100)
    room_type = models.TextField(choices= RoomType.choices) # choices
    number_of_station = models.IntegerField()
    building_id = models.ForeignKey(Building,on_delete=models.RESTRICT)
```

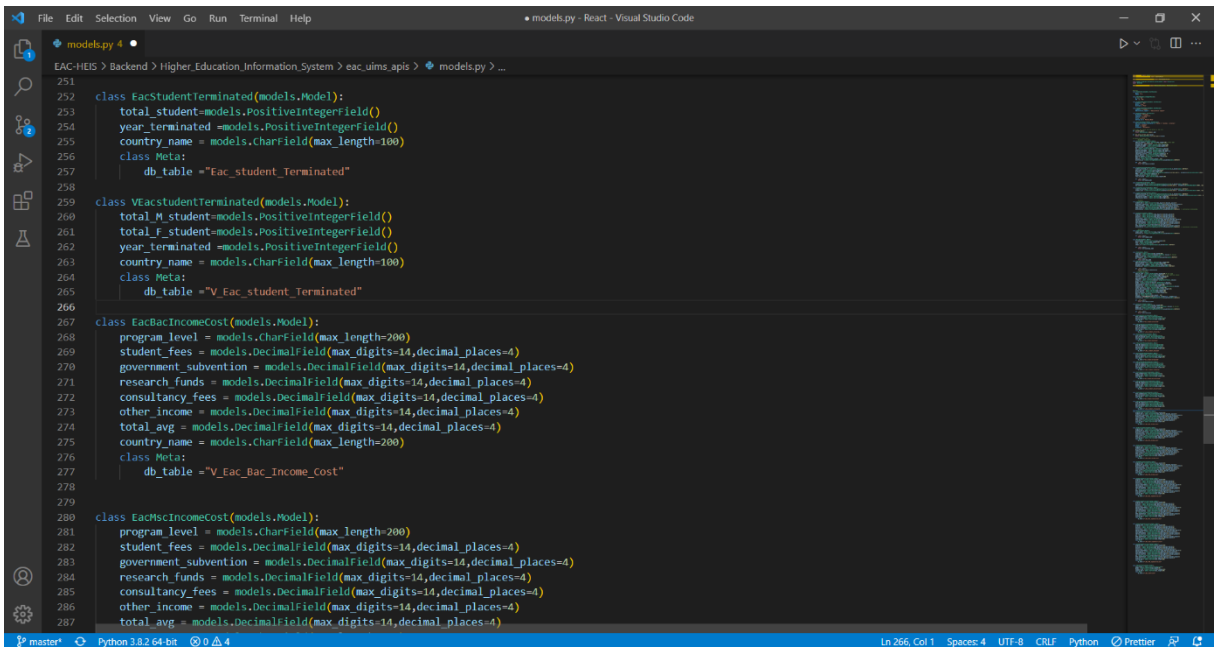


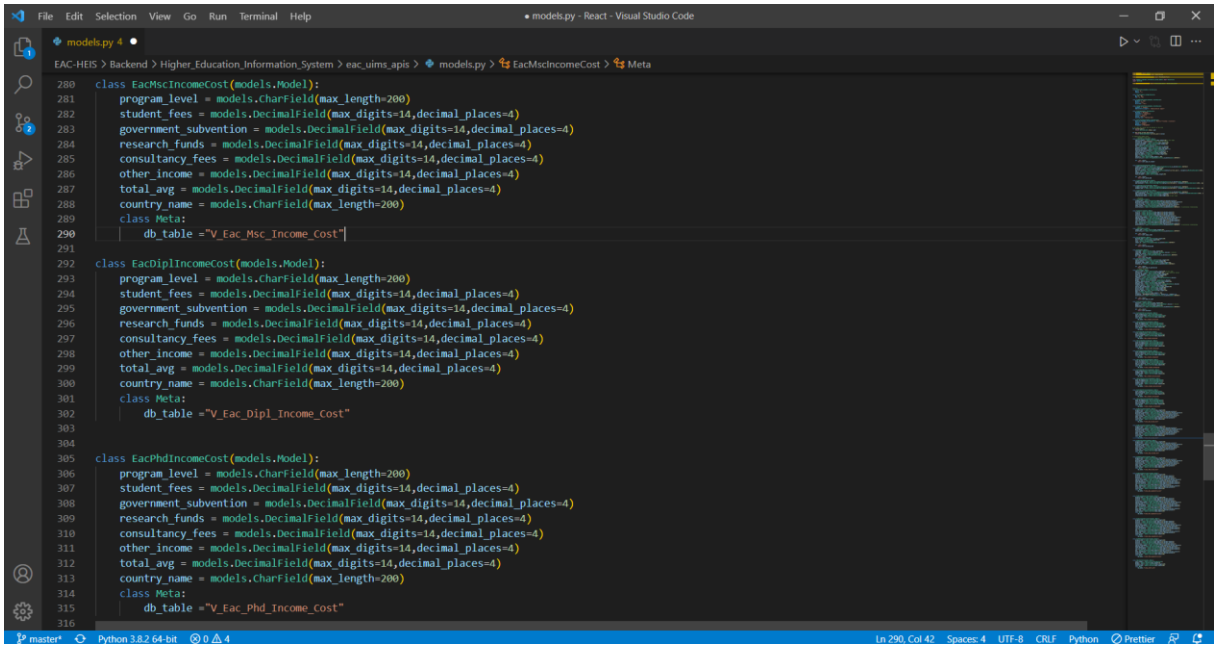
```
class Room(models.Model):
    room_name = models.CharField(max_length=100)
    room_type = models.TextField(choices= RoomType.choices) # choices
    number_of_station = models.IntegerField()
    building_id = models.ForeignKey(Building,on_delete=models.RESTRICT)
    def __str__(self):
        return self.room_name
class Equipment(models.Model):
    name_of_manufacture = models.CharField(max_length=100)
    specifications = models.CharField(max_length=100)
    equipment_index_number = models.CharField(max_length=100)
    location_id = models.ForeignKey(Room,on_delete=models.RESTRICT)

    def __str__(self):
        return self.name_of_manufacture
        # Human resource
class Staff(models.Model):
    name_of_staff = models.CharField(max_length=100) # full name
    nationality = models.CharField(max_length=100)  # all countries choice
    national_ID_number = models.CharField(max_length=100)
    staff_id_number = models.CharField(max_length=100)
    department = models.CharField(max_length=100)
    category = models.TextField(choices= CategoryStaffChoice.choices)
    date_of_birth = models.DateField()
    gender = models.TextField(choices= GenderChoice.choices)
    academic_qualifications = models.CharField(max_length=100)
    publications = models.CharField(max_length=100)
    academic_rank = models.TextField(choices= AcademicRanckChoice.choices)
    research_grants = models.CharField(max_length=100)
    awards_add_recognitions = models.CharField(max_length=100)
    positions_held = models.CharField(max_length=100)
    postal_address = models.CharField(max_length=100)
    physical_address = models.CharField(max_length=100)
    email = models.EmailField()
    website = models.URLField(max_length = 200)
    telephone = PhoneNumberField(null=False, blank=False, unique=True)
```



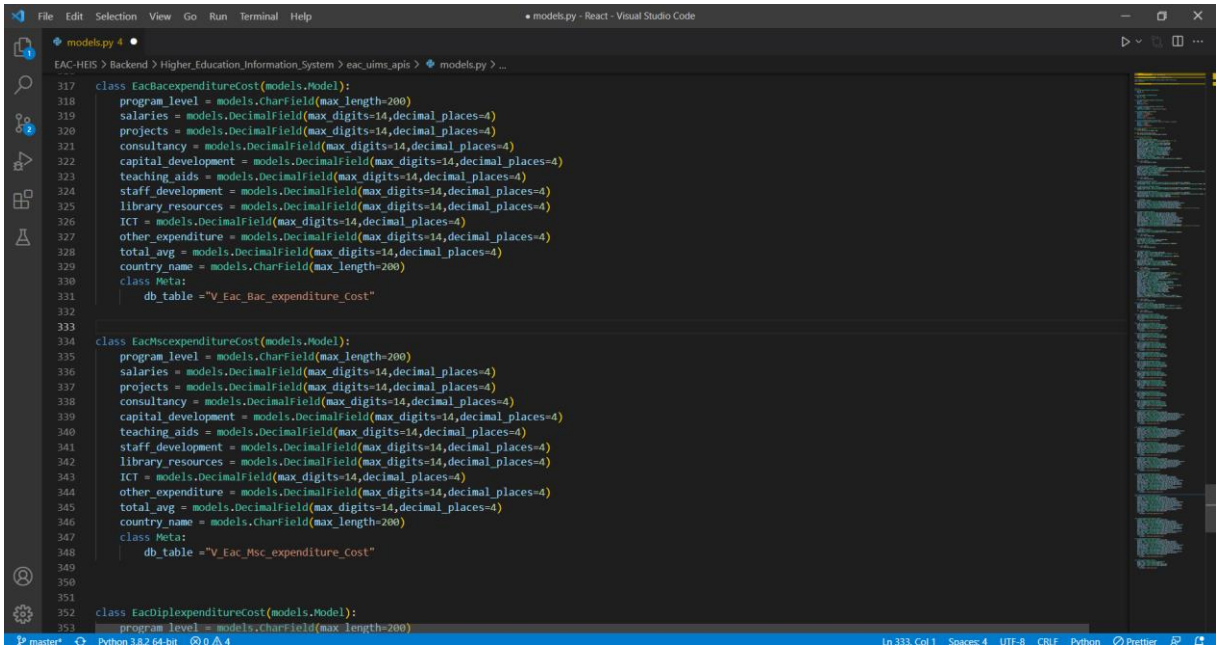
```
        return self.name_of_staff

class Governance(models.Model):
    functions = models.CharField(max_length=100)
    powers = models.TextField(choices= GovernancePower.choices) # choices
    membership =models.CharField(max_length=100)
    university_id = models.ForeignKey(University,on_delete=models.RESTRICT)

    def __str__(self):
        return self.functions

class EacStudentEnroulled(models.Model):
    total_student=models.PositiveIntegerField()
    year_of_enrollment =models.PositiveIntegerField()
    country_name = models.CharField(max_length=100)
    class Meta:
        db_table ="Eac_student_enroulled"

class VEacstudentenroulled(models.Model):
    total_M_student=models.PositiveIntegerField()
    total_F_student=models.PositiveIntegerField()
    year_of_enrollment =models.PositiveIntegerField()
    country_name = models.CharField(max_length=100)
    class Meta:
        db_table ="v_eac_student_enroulled_"

class EacStudentDeceased(models.Model):
    total_student=models.PositiveIntegerField()
    year_of_deceased =models.PositiveIntegerField()
    country_name = models.CharField(max_length=100)
    class Meta:
        db_table ="Eac_student_Deceased"

class VEacstudentDeceased(models.Model):
    total_M_student=models.PositiveIntegerField()
    total_F_student=models.PositiveIntegerField()
    year_deceased =models.PositiveIntegerField()
```


File Edit Selection View Go Run Terminal Help
• models.py - React - Visual Studio Code
models.py 4
EAC-HEIS > Backend > Higher_Education_Information_System > eac_uims_apis > models.py > ...
class VEacstudentDeceased(models.Model):
total_M_student=models.PositiveIntegerField()
total_F_student=models.PositiveIntegerField()
year_deceased =models.PositiveIntegerField()
country_name = models.CharField(max_length=100)
class Meta:
db_table ="V_Eac_student_Deceased"
class EacStudentGraduated(models.Model):
total_student=models.PositiveIntegerField()
year_of_comletion =models.PositiveIntegerField()
country_name = models.CharField(max_length=100)
class Meta:
db_table ="Eac_student_Graduated"
class VEacstudentGraduated(models.Model):
total_M_student=models.PositiveIntegerField()
total_F_student=models.PositiveIntegerField()
year_of_comletion =models.PositiveIntegerField()
country_name = models.CharField(max_length=100)
class Meta:
db_table ="V_Eac_student_Graduated"
class EacStudentDiscontinued(models.Model):
total_student=models.PositiveIntegerField()
academic_year =models.PositiveIntegerField()
country_name = models.CharField(max_length=100)
class Meta:
db_table ="Eac_student_Discontinued"
class VEacstudentDiscontinued(models.Model):
total_M_student=models.PositiveIntegerField()
total_F_student=models.PositiveIntegerField()
academic_year =models.PositiveIntegerField()
country_name = models.CharField(max_length=100)
class Meta:

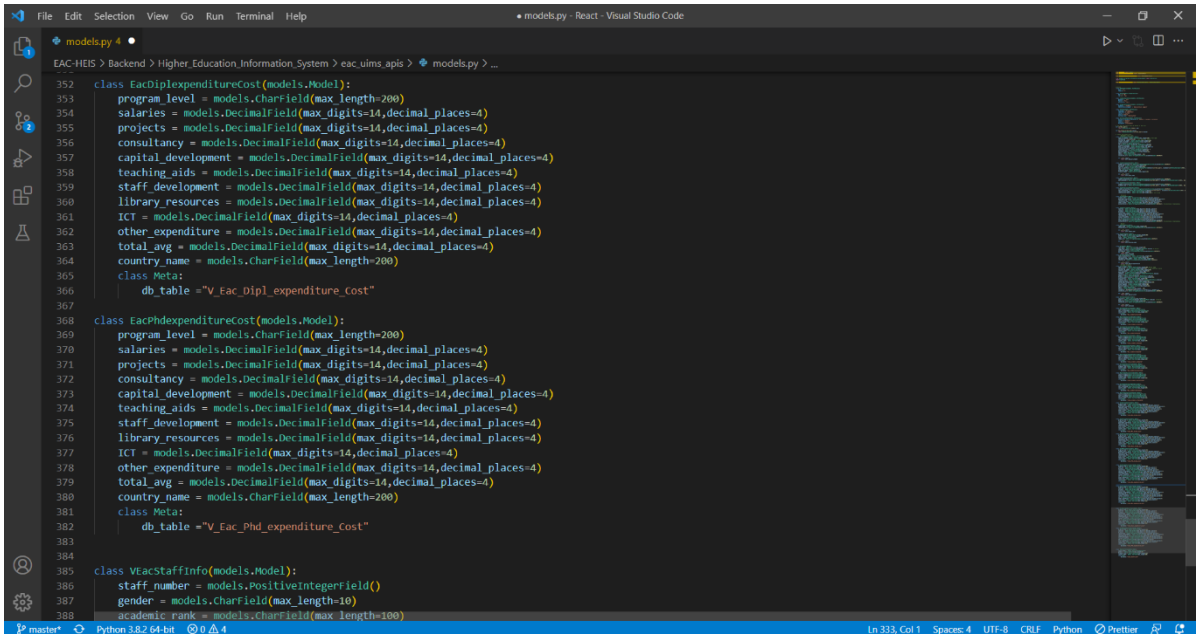

File Edit Selection View Go Run Terminal Help
• models.py - React - Visual Studio Code
models.py 4
EAC-HEIS > Backend > Higher_Education_Information_System > eac_uims_apis > models.py > ...
class EacStudentTerminated(models.Model):
total_student=models.PositiveIntegerField()
year_terminated =models.PositiveIntegerField()
country_name = models.CharField(max_length=100)
class Meta:
db_table ="Eac_student_Terminated"
class VEacstudentTerminated(models.Model):
total_M_student=models.PositiveIntegerField()
total_F_student=models.PositiveIntegerField()
year_terminated =models.PositiveIntegerField()
country_name = models.CharField(max_length=100)
class Meta:
db_table ="V_Eac_student_Terminated"
class EacBacIncomeCost(models.Model):
program_level = models.CharField(max_length=200)
student_fees = models.DecimalField(max_digits=14,decimal_places=4)
government_subvention = models.DecimalField(max_digits=14,decimal_places=4)
research_funds = models.DecimalField(max_digits=14,decimal_places=4)
consultancy_fees = models.DecimalField(max_digits=14,decimal_places=4)
other_income = models.DecimalField(max_digits=14,decimal_places=4)
total_avg = models.DecimalField(max_digits=14,decimal_places=4)
country_name = models.CharField(max_length=200)
class Meta:
db_table ="V_Eac_Bac_Income_Cost"
class EacMscIncomeCost(models.Model):
program_level = models.CharField(max_length=200)
student_fees = models.DecimalField(max_digits=14,decimal_places=4)
government_subvention = models.DecimalField(max_digits=14,decimal_places=4)
research_funds = models.DecimalField(max_digits=14,decimal_places=4)
consultancy_fees = models.DecimalField(max_digits=14,decimal_places=4)
other_income = models.DecimalField(max_digits=14,decimal_places=4)
total_avg = models.DecimalField(max_digits=14,decimal_places=4)
master*
Python 3.8.2 64-bit
0 4
Ln 266, Col 1
Spaces: 4
UTF-8
CRLF
Python
Prettier



```
class EacMscIncomeCost(models.Model):
    program_level = models.CharField(max_length=200)
    student_fees = models.DecimalField(max_digits=14,decimal_places=4)
    government_subvention = models.DecimalField(max_digits=14,decimal_places=4)
    research_funds = models.DecimalField(max_digits=14,decimal_places=4)
    consultancy_fees = models.DecimalField(max_digits=14,decimal_places=4)
    other_income = models.DecimalField(max_digits=14,decimal_places=4)
    total_avg = models.DecimalField(max_digits=14,decimal_places=4)
    country_name = models.CharField(max_length=200)
    class Meta:
        db_table ="V_Eac_Msc_Income_Cost"

class EacDiplIncomeCost(models.Model):
    program_level = models.CharField(max_length=200)
    student_fees = models.DecimalField(max_digits=14,decimal_places=4)
    government_subvention = models.DecimalField(max_digits=14,decimal_places=4)
    research_funds = models.DecimalField(max_digits=14,decimal_places=4)
    consultancy_fees = models.DecimalField(max_digits=14,decimal_places=4)
    other_income = models.DecimalField(max_digits=14,decimal_places=4)
    total_avg = models.DecimalField(max_digits=14,decimal_places=4)
    country_name = models.CharField(max_length=200)
    class Meta:
        db_table ="V_Eac_Dipl_Income_Cost"


class EacPhdIncomeCost(models.Model):
    program_level = models.CharField(max_length=200)
    student_fees = models.DecimalField(max_digits=14,decimal_places=4)
    government_subvention = models.DecimalField(max_digits=14,decimal_places=4)
    research_funds = models.DecimalField(max_digits=14,decimal_places=4)
    consultancy_fees = models.DecimalField(max_digits=14,decimal_places=4)
    other_income = models.DecimalField(max_digits=14,decimal_places=4)
    total_avg = models.DecimalField(max_digits=14,decimal_places=4)
    country_name = models.CharField(max_length=200)
    class Meta:
        db_table ="V_Eac_Phd_Income_Cost"
```

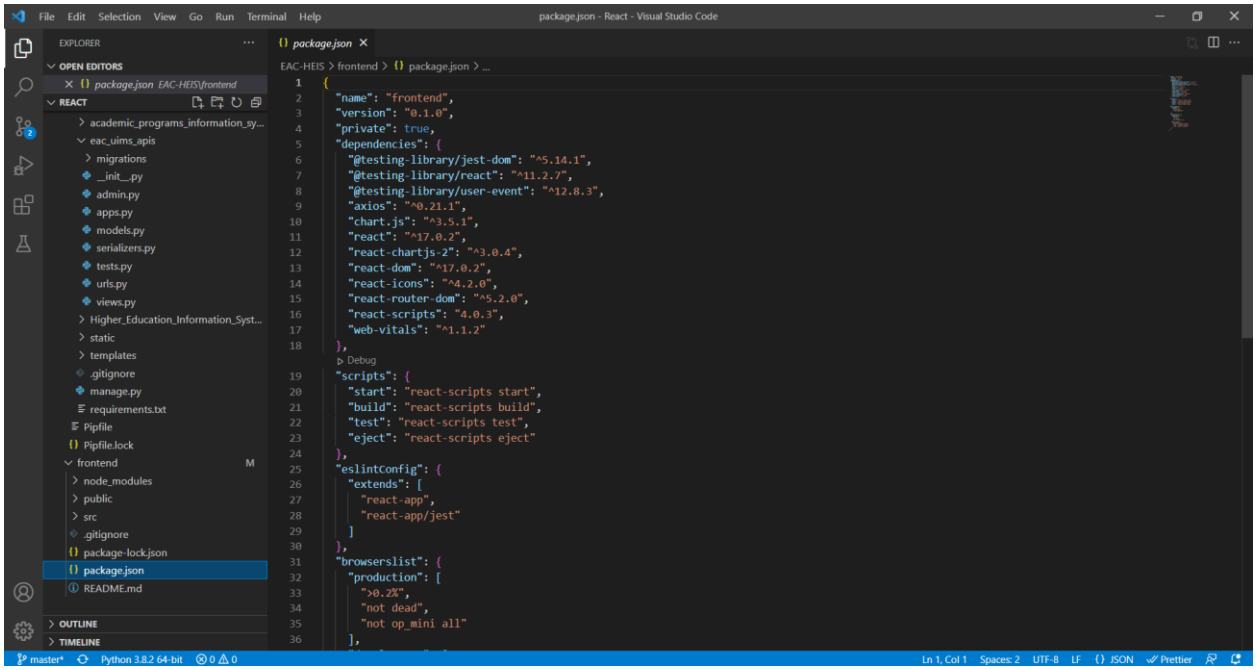


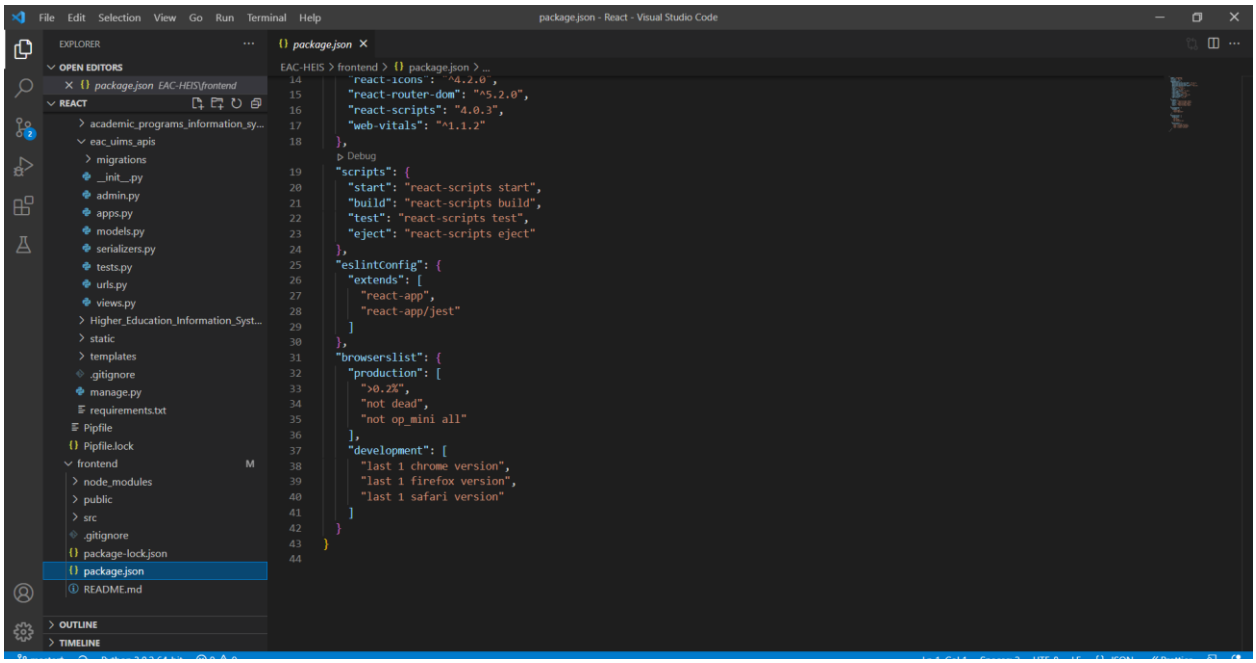
```
class EacBacexpenditureCost(models.Model):
    program_level = models.CharField(max_length=200)
    salaries = models.DecimalField(max_digits=14,decimal_places=4)
    projects = models.DecimalField(max_digits=14,decimal_places=4)
    consultancy = models.DecimalField(max_digits=14,decimal_places=4)
    capital_development = models.DecimalField(max_digits=14,decimal_places=4)
    teaching_aids = models.DecimalField(max_digits=14,decimal_places=4)
    staff_development = models.DecimalField(max_digits=14,decimal_places=4)
    library_resources = models.DecimalField(max_digits=14,decimal_places=4)
    ICT = models.DecimalField(max_digits=14,decimal_places=4)
    other_expenditure = models.DecimalField(max_digits=14,decimal_places=4)
    total_avg = models.DecimalField(max_digits=14,decimal_places=4)
    country_name = models.CharField(max_length=200)
    class Meta:
        db_table ="V_Eac_Bac_expenditure_Cost"


class EacMscexpenditureCost(models.Model):
    program_level = models.CharField(max_length=200)
    salaries = models.DecimalField(max_digits=14,decimal_places=4)
    projects = models.DecimalField(max_digits=14,decimal_places=4)
    consultancy = models.DecimalField(max_digits=14,decimal_places=4)
    capital_development = models.DecimalField(max_digits=14,decimal_places=4)
    teaching_aids = models.DecimalField(max_digits=14,decimal_places=4)
    staff_development = models.DecimalField(max_digits=14,decimal_places=4)
    library_resources = models.DecimalField(max_digits=14,decimal_places=4)
    ICT = models.DecimalField(max_digits=14,decimal_places=4)
    other_expenditure = models.DecimalField(max_digits=14,decimal_places=4)
    total_avg = models.DecimalField(max_digits=14,decimal_places=4)
    country_name = models.CharField(max_length=200)
    class Meta:
        db_table ="V_Eac_Msc_expenditure_Cost"


class EacDiplexpenditureCost(models.Model):
    program_level = models.CharField(max_length=200)
```


File Edit Selection View Go Run Terminal Help
• models.py - React - Visual Studio Code
models.py 4 •
EAC-HEIS > Backend > Higher_Education_Information_System > eac_uims_apis > models.py > ...
class EacDiplexpenditureCost(models.Model):
program_level = models.CharField(max_length=200)
salaries = models.DecimalField(max_digits=14,decimal_places=4)
projects = models.DecimalField(max_digits=14,decimal_places=4)
consultancy = models.DecimalField(max_digits=14,decimal_places=4)
capital_development = models.DecimalField(max_digits=14,decimal_places=4)
teaching_aids = models.DecimalField(max_digits=14,decimal_places=4)
staff_development = models.DecimalField(max_digits=14,decimal_places=4)
library_resources = models.DecimalField(max_digits=14,decimal_places=4)
ICT = models.DecimalField(max_digits=14,decimal_places=4)
other_expenditure = models.DecimalField(max_digits=14,decimal_places=4)
total_avg = models.DecimalField(max_digits=14,decimal_places=4)
country_name = models.CharField(max_length=200)
class Meta:
db_table ="V_Eac_Dipl_expenditure_Cost"
class EacPhdexpenditureCost(models.Model):
program_level = models.CharField(max_length=200)
salaries = models.DecimalField(max_digits=14,decimal_places=4)
projects = models.DecimalField(max_digits=14,decimal_places=4)
consultancy = models.DecimalField(max_digits=14,decimal_places=4)
capital_development = models.DecimalField(max_digits=14,decimal_places=4)
teaching_aids = models.DecimalField(max_digits=14,decimal_places=4)
staff_development = models.DecimalField(max_digits=14,decimal_places=4)
library_resources = models.DecimalField(max_digits=14,decimal_places=4)
ICT = models.DecimalField(max_digits=14,decimal_places=4)
other_expenditure = models.DecimalField(max_digits=14,decimal_places=4)
total_avg = models.DecimalField(max_digits=14,decimal_places=4)
country_name = models.CharField(max_length=200)
class Meta:
db_table ="V_Eac_Phd_expenditure_Cost"
class VEacStaffInfo(models.Model):
staff_number = models.PositiveIntegerField()
gender = models.CharField(max_length=10)
academic_rank = models.CharField(max_length=100)
master*
Python 3.8.2 64-bit
0 4
Ln 333, Col 1
Spaces: 4
UTF-8
CRLF
Python
Prettier

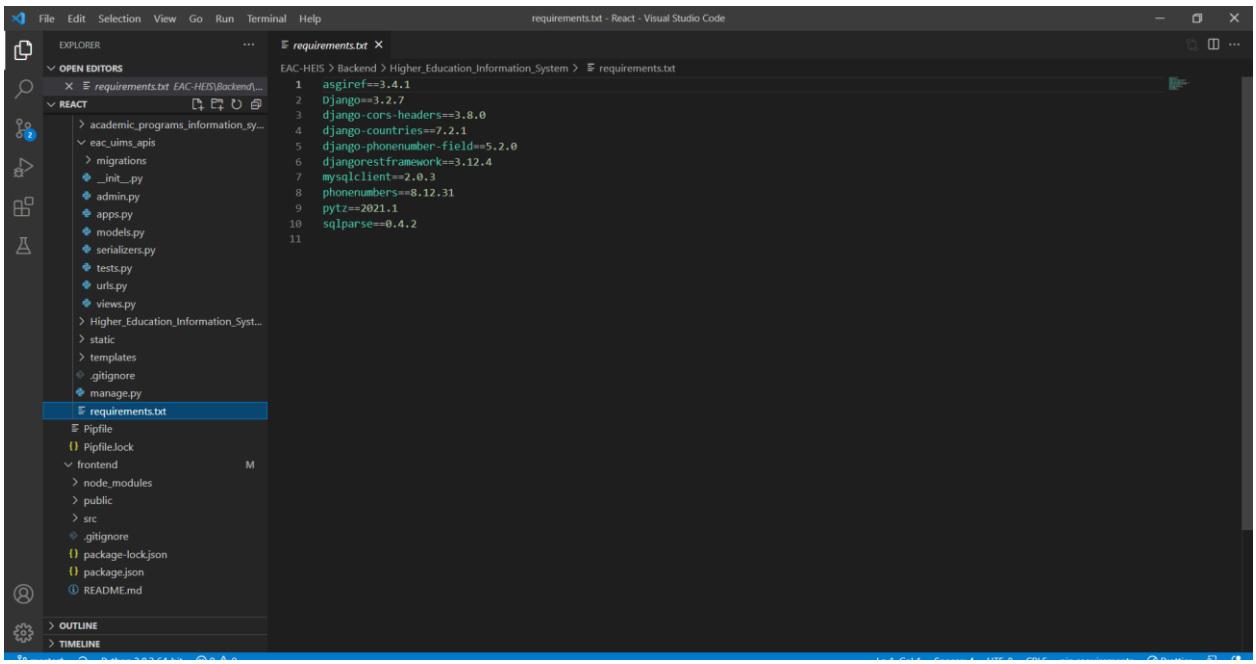
Appendix 6: Packages and libraries used to develop the system

a. React additional packages





b. Django libraries



Appendix 7: Poster Presentation

DEVELOPMENT OF THE WEB-BASED DATA-DRIVEN UNIVERSITY INFORMATION MANAGEMENT SYSTEM (UIMS) FOR INTER-UNIVERSITY COUNCIL FOR EAST AFRICA (IUCEA).

Abel NSHIMYE, Dr. Mussa Dida Ally, Dr. Silas Steven Mirau
Nelson Mandela African Institution of Science and Technology. P. O. Box 447, Arusha, Tanzania.
Emails: nshimiyea@nm-aist.ac.tz , mussa.ally@nm-aist.ac.tz , silas.mirau@nm-aist.ac.tz

Introduction

The conviction that education is one of the most powerful and proven engines for sustainable development is reaffirmed by achieving inclusive and high-quality education for all. The purpose of education is to achieve the success of a society which we can define as measuring how successfully the community meets the needs of its members (Bass, 1997).

However, there are is a lack of a Common Higher Education Area (HEA) for the East African Community (EAC) in awareness of the need to harmonize the region's education and training systems. A tool needed by the Inter-University Council for East Africa (IUCEA) as the leading EAC institution for an exemplary common higher education space.

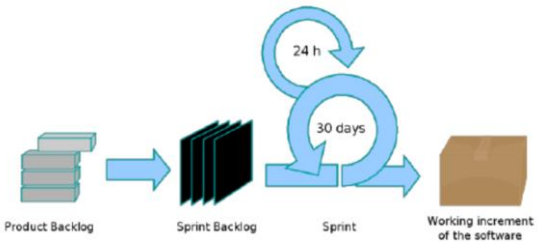
This project aims to develop the web-based data-driven University Information Management System (UIMS) for the IUCEA, the second module of the East African community High Education Information System (EAC HEIS).

Research question

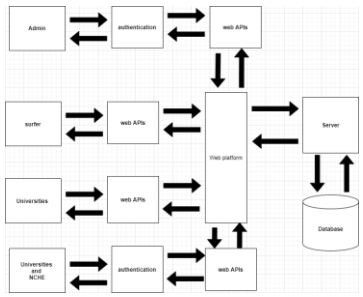
What are required to develop the web-based data-driven University Information Management System (UIMS) for IUCEA and integrate it with the first module of EAC HEIS? What are the features for developing the proposed web application system to manage a massive volume of data? Is the developed system working as anticipated?

Methodology

During the development of this project, we will use the SCRUM agile methodology approach to have good management of time since this methodology evaluates the progress of the project day by day.

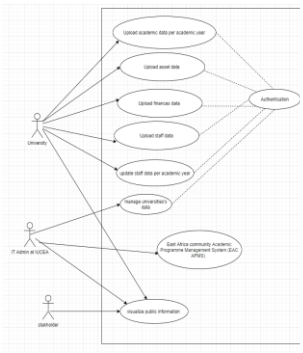


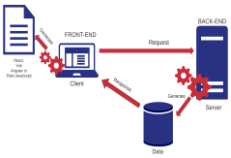
Conceptual diagram of the proposed system



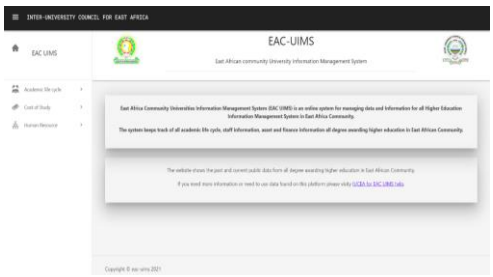
Developed solution

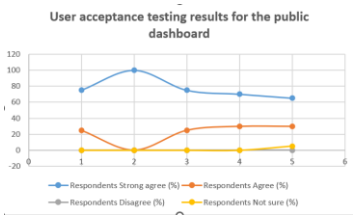
The developed system will help IUCEA manage higher education in the EAC region and all users to know the universities' public information. Considering the system's data where the universities upload the information about academic, finance, and human resource, the system allowed the public to visualize the academic life cycle, the cost of the study, and staff information statistically





System design





Significance:

This project contribute to the development of the EACHEIS, which will be all about electronic resource planning, integrated education management information system, and knowledge management in the EAC region. Furthermore, it will help address suitable solutions to the high education questions in the EAC region.