

**KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY  
COLLEGE OF SCIENCE  
FACULTY OF PHYSICAL AND COMPUTATIONAL SCIENCES**

**DEPARTMENT OF COMPUTER SCIENCE**



**PROJECT TOPIC:**

**COMPUTER SCIENCE STUDY MATE**

**BY**

**NAME: KWADWO KYERE ATTAKORAH & RICHMOND GYAU BAFFOUR**

**MENSAH**

**INDEX NUMBER: 4608718 & 4615518 (respectively)**

**SUPERVISOR: DR. LINDA AMOAKO BANNING.**

### **DECLARATION AND CERTIFICATION**

We hereby certify that this submission is our own work toward the BSc. Computer Science Program and hereby declare that the findings of this project are genuine information that has not been presented by any person or group of persons elsewhere towards the reward of any other degree of the University. Recognition has however been given to authors whose works have been cited. The project was undertaken at the Department of Computer Science, and we are fully responsible for all its shortcomings and imperfection.

ATTAKORAH KWADWO KYERE ..... ....

(Name of candidate) (Signature) (Date)

RICHMOND GYAU BAFFOUR MENSAH ..... ....

(Name of candidate) (Signature) (Date)

Dr. LINDA AMOAKO BANNING ..... ....

(Supervisor) (Signature) (Date)

Prof. JAMES BEN HAYFRON-ACQUAH ..... ....

(Head of Department) (Signature) (Date)

## **DEDICATION.**

We dedicate this project to Mr. Gilbert Addah, C.E.O of learnRite Ghana, who above all motivated us from identifying the problem that called for this web application development, which was successfully delivered on this project.

## **ACKNOWLEDGEMENT**

We would like to thank the Lord almighty for His mercy, grace, and strength throughout the period of this project work.

Our profound gratitude goes to our wonderful supervisor, Dr. Linda Amoako Banning for her enormous direction, guidance, time, patience, numerous contributions, and other supervisory roles toward the successful completion of this project work. God richly bless you, Madam.

Not forgetting Mr. Ofori Kuma who really helped us by giving us additional guidance on this project.

Finally, we would want to acknowledge Master Lemuel Anim-Boateng, a course mate for his support throughout this period of our project work.

## TABLE OF CONTENTS

DECLARATION AND CERTIFICATION .....	2
ACKNOWLEDGEMENT .....	3
ABSTRACT.....	9
CHAPTER ONE .....	10
GENERAL INTRODUCTION.....	10
1.1 INTRODUCTION.....	10
1.2 PROBLEM STATEMENT .....	11
1.3 PROJECT AIM .....	12
1.4 OBJECTIVES .....	12
1.5PROJECT IMPORTANCE AND JUSTIFICATION .....	13
1.6 PROJECT MOTIVATION. ....	14
1.7 PROJECT SCOPE .....	15
1.8 SIGNIFICANT PRIOR RESEARCH.....	15
1.9 METHODOLOGY .....	17
1.10 ETHICAL CONSIDERATION .....	19
1.11 PROJECT ANALYSIS .....	19
1.13 LIMITATIONS .....	20
1.14 BENEFICIARIES OF THE PROJECT. ....	20
1.15 STRUCTURE OF REPORT.....	20
1.16 CHAPTER SUMMARY.....	22
CHAPTER TWO .....	22
LITERATURE REVIEW .....	22
2.1 INTRODUCTION.....	22

2.2 DEFINITION OF CONCEPTS.....	23
2.2.1 I.C.T (Information and Communication Technology). .....	23
2.2.2 ONLINE PLATFORM.....	24
2.2.3 ONLINE LEARNING PLATFORM.....	24
2.3 EVOLUTION OF ONLINE PLATFORM. ....	24
2.4 EXISTING ONLINE EDUCATIONAL PLATFORM.....	26
2.4.1 QUORA.....	26
2.4.2 UDEMY.....	26
2.5 FEATURES OF AN EXISTING ONLINE EDUCATIONAL PLATFORM. ....	28
2.6 BENEFITS OF ONLINE EDUCATIONAL PLATFORMS .....	28
2.7 DISADVANTAGES OF E-LEARNING.....	29
2.8 ONLINE LEARNING IN GHANA.....	30
2.9 CHAPTER SUMMARY.....	31
CHAPTER THREE. ....	33
METHODOLOGY .....	33
3.1 INTRODUCTION.....	33
3.2 DEVELOPMENT METHODOLOGY. ....	33
3.2.1 WATERFALL MODEL.....	34
3.2.2 INCREMENTAL MODEL. ....	36
3.3 PROJECT METHOD ADAPTED AND JUSTIFICATION. ....	37
3.4 REQUIREMENT GATHERING PROCESS. ....	38
3.5 REQUIREMENT SPECIFICATION.....	38
3.5.1 FUNCTIONAL REQUIREMENT OF THE SYSTEM. ....	38

3.5.2 NON-FUNCTIONAL REQUIREMENT OF THE SYSTEM.....	39
3.6 UNIFIED MODELING LANGUAGE.....	40
3.6.1 USE CASE DIAGRAM.....	40
3.6.2 FLOWCHART .....	41
3.6.3 CONTEXT DIAGRAM.....	42
3.6.4 SYSTEM ARCHITECTURE.....	43
3.7 PROJECT DESIGN CONSIDERATION. (LOGICAL DESIGN).....	44
3.7.1 USER INTERFACE DESIGN .....	44
3.7.2 DATABASE DESIGN.....	45
3.8 DEVELOPMENT TOOLS AND ENVIRONMENT.....	47
3.8.1 DEVELOPMENT ENVIRONMENT.....	47
3.8.2 DEVELOPMENT SOFTWARE TOOLS.....	47
3.9 CHAPTER SUMMARY.....	47
CHAPTER FOUR.....	48
TESTING AND IMPLEMENTATION.....	48
4.1 INTRODUCTION.....	48
4.2 TESTING .....	48
4.2.1 UNIT TESTING.....	50
4.2.2. SYSTEM TESTING.....	51
4.2.3 USABILITY TESTING .....	52
4.2.4 TESTING PLAN .....	52
4.3 IMPLEMENTATION .....	52
4.3.1 THE HTML IMPLEMENTATION .....	52

4.3.2 THE CSS IMPLEMENTATION.....	52
4.3.3 DATABASE.....	53
4.4 MAPPING LOGICAL DESIGN ONTO PHYSICAL PLATFORM.....	53
4.5 CONSTRUCTION.....	53
4.5.1 DESIGN AND INTERFACE.....	54
CHAPTER 5 .....	65
SUMMARY, CONCLUSION, AND RECOMMENDATION.....	65
5.1 INTRODUCTION.....	65
5.2 SUMMARY .....	65
5.3 FINDINGS .....	66
5.4 LIMITATION.....	66
5.5 CONCLUSION.....	67
5.6 RECOMMENDATIONS .....	67
REFERENCES .....	68
APPENDIX.....	71
Appendix 1 .....	71
Appendix 2 .....	76
Appendix 3 .....	79
Appendix 4 .....	81
Appendix 5 .....	82
Appendix 6 .....	84



## **ABSTRACT**

**Computer Science Study Mate** is a web application that serves as a social networking platform for all computer science students across the country to share ideas and experiences. The network also aims at ensuring engaged learning and offers students and lecturers the opportunity to discuss several academic issues that will help clear some of the fears and challenges that have constituted barriers to the various courses offered as part of the computer science program. This social networking platform will provide features that will enable students and lecturers to sign up to the platform, enable students to search for study mates and send study mate requests which would be either accepted or declined, and allow students to download e computers books that will facilitate their discussions, send voices notes, pictures, videos, and any other attachment files.

## **CHAPTER ONE**

### **GENERAL INTRODUCTION**

#### **1.1 INTRODUCTION**

The creative application of Information and Communication Technologies (ICT) in a variety of fields, such as agriculture, health, education, and so on, is becoming a major emphasis for governments in developing nations, including Ghana. The use of ICT in government services (e-Government) aims to integrate these new technologies in all regions and in the hands of all citizens to provide the best possible web services (Alonso et al., 2004). As one of the main priority areas, the present government has digitized every sector of the economy, including agriculture, health, and education.

The Internet has evolved into a limitless marketplace for buying and selling products and services. Companies battle fiercely for each possible consumer while Web users seek for, inspect, and occasionally buy products and services on the Internet. Knowledge of potential client's demand and the capacity to create customized services that meet these needs are critical to winning this competitive race (Spiliopoulou, 2000). The key to optimizing the business value of e-business websites is to understand client behavior. Because of the lack of customer identity data, traditional Web server logs are useful for website performance analysis but not for user activity analysis.

Furthermore, due to the stateless connection style of the HTTP protocol, tracking an individual is challenging. Tonaton dot com is one of numerous online platforms in Ghana that facilitates the purchasing and selling of various things. However, there are just a few online sites dedicated to linking students to studying and sharing ideas.

## **1.2 PROBLEM STATEMENT**

The automated processes by which computer users and business organizations perform transactions over the internet, such as purchasing, selling, moving products, or even swapping goods and services, are known as electronic commerce or E-commerce systems (Alyoubi, 2015).

The entire global retail E-commerce sale was \$22.5 trillion at the end of 2015. In comparison to the \$21.3 trillion in 2014, this is more than a trillion dollars. It is expected to reach \$27.9 trillion by the end of 2019 (Emarketer, 2015). The data indicates that sales in E-commerce transactions are rapidly increasing. It has a huge impact on the worldwide economy.

Education as a social good is viewed as a vehicle for transformation. It is critical for societal advancement and development. This implies that education should continuously be changing to meet the needs of society (Serdyukov, 2017). Educational institutions have taken advantage of the evolution of technology such as online learning to improve instructional delivery.

E-learning is increasingly becoming a tool for promoting efficiency and quality in education (Alhassan, 2020). This is because E-commerce has had a significant impact on practically all enterprises, particularly in the essentials sector such as education (Terzi, 2014). It has simplified and influenced the introduction of new businesses without requiring large investments or capital outlays. Because of the low minimum investment requirement for physical infrastructure, venture capitalists are able to save a lot of money. A corporation with modest resources will have a global presence where goods and even services can be advertised. Another effect of E-commerce is the increased competition caused by adverts incorporated into its web pages. It also offers consumers low-cost goods while expanding the product line's options. Again, e-commerce in the educational sector ensures flexibility, convenience, and collaboration in curriculum design and instructional delivery (van Damme, 2019). It is one of the innovative ways to acquire knowledge and skills

acquisition (Campbell, 2019).

It is in this vein that this app is being developed to improve education in Ghana. This is because there have been in existence social media apps such as Facebook and Twitter that facilitate social interactions. However, this app fills in the gap by developing content that solely ensures learning online through sharing of ideas. This is because looking at our Ghanaian educational system, the standard of instruction undertaken in schools from kindergarten to the university comparatively are not equal. This inequity in educational standards in terms of resources leads to different educational outcomes and widens the gap between the rich and the poor. This app will bridge this gap by sharing teaching and learning resources such as books, PDFs, and journals that students can access. There will also be a section where students can interact and share ideas. Therefore, the Computer Study Mate web application would be a platform to solve the challenges or the problem identified and listed above.

### **1.3 PROJECT AIM**

To solve the problem, we've identified, the proposed project will enable computer science students to link up on a single platform having study mates and enable them to have discussions. This platform will help these study mates to share ideas, experiences, study materials, examinations, and quiz questions. The platform will enable the student to interact with other lecturers outside their universities to expand their ideas and knowledge.

### **1.4 OBJECTIVES**

The main objective of our project is as follows:

- For students to send, accept and decline study mates' requests.
- To be able to download computer science E-books.
- For students to have discussions with study mates either in real-time or not.
- For students/lecturers to send attachment files (pdf, notes, examination questions) to each other.
- For a student to send pictures of educational resources to aid their discussion.
- For a student to send voice notes to better explain themselves.
- For students to have discussions with lecturers when needed (to clarify doubts that would arise from their discussion).
- For both students and lecturers to send or share research topics or problems with colleagues to obtain desirable and excellent suggestions and contributions.

## **1.5PROJECT IMPORTANCE AND JUSTIFICATION**

The Computer science study mate web application has much importance, and they are as follows:

- To enrich the training of computer science students in Ghana.
- Healthy conversations and group discussions with study mates will generate intuitive ideas and help develop the analytic power of students.
- To create an ecosystem for computer science departments that involves lecturers and students across the country to connect and share knowledge, ideas, and experiences.
- To create and foster a good relationship between the various computer Science departments in the various universities as both students and lecturers come together to share research topics to get or obtain desirable and excellent suggestions and contributions.

Looking at the problem statement which calls for a solution to developing a platform like this has some driving force. And so, developing this project is so, to bridge the knowledge, idea, and experiences gap that exist between computer science students in various universities. Also, this platform is to expose computer science students nationwide as they connect to additional knowledge and experiences beyond the confines of the student's respective departments in their universities.

### **1.6 PROJECT MOTIVATION.**

Knowing Mr. Ampofo, a lecturer at the University of Cape Coast (UCC) who has been in practice for 30 years and is now retired has given us exposure to some ideas about the importance on peer to peer learning and likewise student to lecture learning. In one of our discussions, he made mention how interacting with our fellow students is helpful in learning. Mr. Ampofo said, peer through discussion sessions in the learning process, students sometimes understand their fellow students' misunderstandings and confusion whenever they are confused with what they are studying. Thus, giving them the chance to break down what looks complex to them from the lecturer to something simple

and easy to understand. He also made mentioned how students are limited to the knowledge and experiences of their lecturers in their university where the students find themselves studying. In the course of our discussion, he further said if only could students have access to other colleagues in other universities of the same study and also lecturers from other universities lecturing the same field of study it would ensure engaged learning thus clearing a lot of misunderstanding and confusion and offering students the opportunity to seek more knowledge and experience.

After this encounter with Mr. Ampofo, there was a hunger to solve this problem to help improve the learning experience thus the idea of Computer Science Study Mate.

## **1.7 PROJECT SCOPE**

The target audience includes all computer science lecturers and students in the listed universities below, to interact with each other in discussing and sharing ideas, knowledge, and experiences. The proposed solution which comes along with the development of a web application wishes to have a successful implementation in Ghana and expand its reach to other west African countries to connect their students and lecturers in their computer science departments in their respective universities, which will intend help open up the scope of knowledge sharing.

The universities are;

Public

universities;

1. Kwame Nkrumah university of science and technology (KNUST).
2. University of Ghana (Legon)
3. University of Development Studies

(UDS) Private universities;

1. Ashesi university.

## **1.8 SIGNIFICANT PRIOR RESEARCH**

Since the existence of the internet which has created a global community, it has been easy to access a lot of things globally. In terms of the reach of information, connecting and communicating with people out of physical reach, etc. Due to the creation of the internet, there have been several online social platforms for which most of which are for social networking and entertainment. Also, there are online platforms for education purposes of which our proposed

web application would be part, where the main aim is to create an environment online to host students and lecturers on this platform. Similarly, there has been the development of such an online platform to facilitate the learning of students. Some of them are Quora, Answers, Stack overflow, and Blurit. These platforms are community-based questions and answer web applications. Here users post questions on any topic and other users respond. The main purpose is to give users answers and allow other users to answer questions for others in subjects in which they are knowledgeable. These web applications are not real-time, it does not allow users to have direct and continuous contact with other users whom they are seeking extra knowledge from. Some other educational platforms are Udemy, Coursera, Edx, Khan Academy, and Masterclass. These listed educational web application platforms only allow instructors to build online courses on their preferred topics. Using the development tools of these web applications, videos, PowerPoint presentations, PDFs, audio, ZIP files, and live classes are uploaded to create courses. These web applications do not also allow users to have discussions on the uploaded content and to share knowledge with others beyond reach.

Looking at the problem statement stated in the problem statement section, the proposed Computer study mate will be a web application that will connect computer Science students and lecturers on a single platform where they would be able to share ideas, experiences, study materials, examinations, and quiz questions. The platform will enable students to interact with other lecturers outside their universities to expand their ideas and knowledge. This platform will not only allow users to just post questions and other users (students or lecturers) to respond to the questions where there is no continuous interaction occurring both in real-time and non-real-time. It will also not only allow the upload of educational content like some educational web application platforms but will allow the users to send, accept and decline study mate's request,

be able to download computer science E-books, and send voice notes to better explain themselves, to send or share research topics or problems with colleagues to obtain desirable and excellent suggestions and contributions.

## 1.9 METHODOLOGY

Looking at the problem statement, we are going to build a web application with tools to get computer science students in the various departments in the various universities connected to this platform. There would be a need for devices such as a desktop, laptop, mobile phone, or tablet for users to use to access the proposed web application.

With the development we will have to design an interactive user interface (front end) which we will have to use the following:

**Html** (HyperText Markup Language); A language that is used to structure a web page and its content.

**CSS** (Cascading Style Sheet); A stylesheet language that is used to describe the presentation of a webpage or a document written in HTML

After developing everything that users visually see first in their browser or application, we will go ahead to the development of the back end which focuses on how the web application will

work. For that **Php** (Laravel framework); A server-side scripting language that is embedded in HTML which allows for the creation of dynamic content and interacts with the database would be used.

Looking at the pending web application to be developed, there would be a need for a computer

server with good specifications as follows;

- A minimum RAM of 512 MB. The recommended amount can vary depending on the number of users connected, the number of websites hosted on the device, and other factors. Hence the minimum RAM or RAM above the minimum will allow more users to be connected to the web application and surf it.
- A minimum Intel® Celeron® Processor 847, of speed 1.10 GHz, or equivalent to help the computing processes be faster and more efficient.
- GB of available hard-disk space for installation; additional free space will be required during installation. Since we cannot install it on removable flash storage devices.
- A minimum storage capacity between 1.3 GB - 2.3 GB depending on the language version that will be used for the development

There will be a need for a minimum bandwidth of 50 KBps. This will help handle a specific amount of data during the computing process. A Latency under 150 ms will be needed as it is the time it takes for the computer, the internet, and everything in between, to respond to an action a user will take whiles using the proposed web application.

During our development, we will need a local host platform such as a Xamp server and a Wamp server. These local host platforms will allow us to test the web application that would be developed.

For users to be able to have access to the web application, there would be a need for browsers such as Microsoft Edge, Mozilla Firefox, Google Chrome, and Apple Safari, also their devices need to have a minimum processor of 1.9 gigahertz (GHz) x86 which will speed the computing process whiles surfing on the web application and preventing it from lagging. Also, the device will need a minimum RAM of 2GB to be able to hold enough data whiles using the web

application. A minimum display of Super VGA with a resolution of 1024 x 768 for a better display of the visual content, that is the UI (User Interface)

### **1.10 ETHICAL CONSIDERATION**

The proposed web application will be a platform that will ensure voluntary participation, where participants will have the right to join or redraw the activities on the platform. Consent of lecturers who are on the platform whose notes will be shared amongst students for their discussion purposes will be sought to avoid copyright infringement. Privacy will be of paramount importance as the proposed platform will be developed with high security to avoid non-users of the platform from accessing users' accounts. The Ghana data protection act will be implemented to protect the privacy and personal data of individuals or users. It will regulate the process of personal information acquired, kept, used, or disclosed.

### **1.11 PROJECT ANALYSIS**

In the analysis of this project, a use case diagram that forms part of the Unified Modelling Language (UML) would be used. The Use case diagram in this analysis would be used to visualize or put the proposed system into a graphical form which would help identify the various themes (actors/users), of this system. The Use case diagram would be used to capture the requirements and context of the system. This will provide functions of the system in a visual form performed by the actors and how they interact with the system to achieve their goals. A use case will help us design a system from the end user's perspective as also it is an effective technique for communicating system behavior in the user's terms by specifying all externally visible system behavior. Therefore, validating the system architecture, generating test cases, and

driving implementation.

### **1.13 LIMITATIONS**

Some of the major setbacks and limitations of the system are:

- Internet dependence -Without a reliable internet connection we cannot run the system or the web application
- Supported device – Without an internet enable device the web application cannot be run.
- Web issues – The application is entirely dependent on a website. This means that, if the website happens to undergo failure, then most likely the application will be failing too.
- Browser versions – As there are versions of a specific browser, some versions of browsers are outdated hence not being able to support the web application.

### **1.14 BENEFICIARIES OF THE PROJECT.**

The direct beneficiaries of this project are the users which include students and lecturers of computer science in various universities. The indirect beneficiaries also include book authors', parents, university administration, and the business community. This web application would help the academic performance of students and expose lecturers to better give assistance for students.

### **1.15 STRUCTURE OF REPORT.**

- Introduction

The project's purpose and scope are discussed in the introduction. It provides a broad overview of the issue and how the project will address it. The purpose, motivation, and objectives of the project are also included.

➤ Review of related works

In this chapter, we briefly discuss some related research concerning our research topic, features, advantage, and disadvantages.

➤ Methodology

The methodology is about the processes and details involved in the development of the project web app. It talks about the plan, and architecture to consider for our project. We also consider the functional and non-functional requirements, security, and overall design of the project.

➤ Implementation and Testing

This chapter documents the finishing of the web app and its testing, which was done according to a set of parameters, its plan, and results. This section proves our web app's usability, reliability, functionality, and efficiency.

➤ Conclusion

We conclude our project web application by discussing the challenges we faced, lessons learned, and any possible future works or additions to the project.

## **1.16 CHAPTER SUMMARY.**

Chapter one introduces the purpose of the proposed system to be developed. An explained reason that caused the development of such a platform in this project, along with the importance that comes along with the proposed system and the motivation behind the proposed system. This chapter further opens up briefly on prior research made on such platforms similar to the proposed system. The chapter goes on to highlight the methodology involving programming languages that will be used and the specifications of devices that will support the running of the proposed system. Users of this proposed system or platform were identified. Touching on ethical considerations made in order not to violate any ethical rules in the course of the development of the system and the beneficiaries of the system. Finally, the project analysis, limitations of the proposed system, and the structure of the report were outlined.

## **CHAPTER TWO**

### **LITERATURE REVIEW**

#### **2.1 INTRODUCTION**

The importance of information technology cannot be underestimated especially in the educational sector. Many initiatives about projects associated with technology, especially online education is now the center stage in many policy decisions across the globe. Online educational resources are emerging as one of the critical components of educational delivery, especially in the era of Covid 19.

This section of the project aims to do a literature review on educational platforms. Other issues discussed include a definition of concepts, the evolution of online platforms, the importance of

online platforms, and the challenges of the online platform.

## **2.2 DEFINITION OF CONCEPTS**

### **2.2.1 I.C.T (Information and Communication Technology).**

The importance of sophistication in technology and the advancement in economic and social growth has been recognized these past years of the technological era. As a result of personal studies and observation, scholars and people who are involved in making policies and policy decisions in the twenty-first century have focused their attention on the importance and the approach to technology, including specifically, within educational systems, information, and communication technology (ICT) in countries, both developed and developing.

ICT has been defined in various ways by different authors. For instance, Gokhe (2012) defines ICT as a technology that supports activities involving information. Data collection or gathering, processing, storage, and presentation are activities supported by ICT. Increasingly these activities also include collaboration and communication. Again, ICT is a set of tools that enable, facilitate, and reinforce educational transformation (Kreijnsa, van Acker, Vermeulend, and van Buurenm,2014). ICT is generally included in government education policies (Hernández-Ramos, Martnez-Abad, Pealvo, Garca, and Rodrguez-Conde (2014)). Despite the significance of ICT, it is widely accepted that ICT should be integrated into and made part of the educational system. Furthermore, an increased understanding of ICT within educational institutions may be beneficial in terms of students' existing ICT skill sets (Friedrich & Hron 2011). The use of ICT in the classroom has a direct impact on students' (Ghana, 2002) academic performance.

### **2.2.2 ONLINE PLATFORM.**

Online platform depending on the discipline has vast application. On the internet, an online platform describes a set of services available. These services include social media, application stores, search engines, communication services, and much more. An online platform is therefore defined as a digital service that helps facilitate interaction between two or more separate users through the service of the internet OECD (Organization for Economic Cooperation and Development) (2019). Again, the online platform is an internet-based website and web application that includes a social network, search engine, and many more for public-facing.

### **2.2.3 ONLINE LEARNING PLATFORM.**

When we talk about online learning, we notice that it encompasses a set of technologies which include www (world wide web), text, audio, video chat, and electronic mail which occur over computer networks. This set of technologies of online learning imparts the educational system in countries, both developed and developing. It allows the learners to learn at their speed and their leisure. In this teachers or lectures serve as transmitters of knowledge. Therefore, an online learning platform is defined as an online networked space where educational resources and learning content on a specific or different topic or knowledge discipline are found.

## **2.3 EVOLUTION OF ONLINE PLATFORM.**

In the twentieth century, businesses existing at that time were based on providing products and services to their customers or clients while they were either attempting to encapsulate the value chain or giving attention to a specific part of it. (Van Alstyne, Parker, & Choudary, 2016). Businesses operating in different many nations focused their efforts on developing better or more

cost-effective products, services, or commodities through economies of scale to eliminate competition. The emergence of the internet and its usage throughout the 1990s for a new business model began when the value was provided by developing a platform that provided opportunities for other businesses and even other sector disciplines (Carter, Vonno, & Singh, 2017; Still et al., 2017; Wang & Yin, 2017). The other sector discipline includes education.

The application of online platforms in the educational sector has given birth to what we know now as e-learning. Electronic learning (E-Learning), also known as computer-assisted learning, has been in existence since the 1960s, but its adoption and its widespread use began after the internet and the web gained popularity. E-learning has given the online platform a new term, which is an online learning platform or e-learning platform. As the Internet and its applications grew in popularity, the role of using computers in the learning process grew. E-learning is becoming more practical and accessible. E-learning can be seen as a natural progression of distance education. It has always taken advantage of new technology to design and adapt educational tool frameworks to shape education (Sangra, et al, 2012). But way before the internet and the use of the online platform or online learning platform, distance learning and sharing of ideas on knowledge were in existence. Isaac Pitman who was a teacher around the 1840's used the mail system to send his students' work to be done and also received completed assignments. With the introduction of the internet and computers in the twentieth century, delivery methods and as well as learning tools expanded as the use of the online platform used by businesses was adopted for distance learning. This online learning platform in the 21<sup>st</sup> century provides its users with facilities for easy communication and this helps the student to access educational content anywhere so far as there is an internet connection.

## **2.4 EXISTING ONLINE EDUCATIONAL PLATFORM.**

There are a lot of online educational platforms. Our survey spotlights the top and most popular online educational platform which has its focuses on improving the educational sector by allowing the student to enjoy education via computer network aside from the traditional way of learning or education. Some of these online educational platforms are up and running in most countries worldwide, of which some are profit-oriented and some are free and are usually sponsored by NGOs (Non-Government Organization), governments, and individuals. Some of the online educational platforms are discussed and reviewed.

### **2.4.1 QUORA.**

Quora is an online learning platform that has a mission of sharing and growing the world's knowledge. The reason for this platform is since knowledge even though can be written down, not all can be written and it remains in the head of people. Due to this reason, Quora is an online learning platform that connects people who have knowledge that other people need. It connects people of diverse perspectives to create room for clarification and understanding each other for everyone to share their idea or knowledge. Quora is an online community platform where people visit to ask questions for an online community of people to answer. With this platform, there is a collection of questions and answers which are created, edited, and organized by everyone that uses it, that is, its content is community-edited and organized and the content quality comes with voting showing the number of people who support and agree with the knowledge shared.

### **2.4.2 UDEMY.**

Udemy is an online educational platform or destination that helps governments, companies, and students to gain knowledge and skills that are needed in the 21<sup>st</sup>-century economy to reach their goals. Udemy platform is one of the biggest online peer-reviewed course providers taught by instructors around the world and has a series of learning experiences that are in the form of

module series. This platform has a multi-library that contains courses across a wide range of fields of study of which some are business and entrepreneurship, the arts, music, language, and a lot more. It provides educational lessons in such a way that it includes video, text notes, and assessment tests. For this reason, Udemy is one of the leading global marketplaces for teaching and learning thus connecting a lot of students to knowledge and practical skills.

Only two currently available online educational systems were examined. This was due to the findings of our research, which revealed that the majority of existing online educational platforms fall into two categories: online question-and-answer platforms and online course-learning platforms. Apart from Quora and Udemy, every other existing platform act in some way similar to them. Other online educational platforms include:

Table 2.4.1 Other Existing Online Educational Platforms.

<b>Questions and Answers online platform</b>	<b>Online courses learning platform</b>
**Quora	**Udemy
Answerbag	Coursera
Blur it	Skillshare
WikiAnswers	Edx
Askville	Khan Academy
And others	And others

Table 2.4.1 Other Existing Online Educational Platforms.

## **2.5 FEATURES OF AN EXISTING ONLINE EDUCATIONAL PLATFORM.**

Online educational platform has features that enable all users to have a full experience of the platform. The following are some features of the online educational platform

1. User Friendly – here users can log onto the platform and easily navigate through and around it.
2. Interactive – the platform provides forms where users relate with other users either students or instructors.
3. Accessibility – the platform is accessible to every user who wants to use it. Also being able to access the platform anywhere and across a variety of electronic devices.
4. Affordable – the platform provides online programs and knowledge which are cheaper to get as compared to the ones offered traditionally.

## **2.6 BENEFITS OF ONLINE EDUCATIONAL PLATFORMS**

There are a lot of advantages associated with E-learning. For example, e-learning offers far more flexible learning methods to attend classes with less travel time. Learners can have a stronger understanding of the knowledge by participating in classroom activities using an interactive video facility (Gautam and Tiwari, 2016). This allows students to respond quickly to activities.

Muruthy and Yamin (2017) studied how beneficial e-learning is for students in higher education institutions, as well as how it is used in the learning process. This research revealed several benefits. The first is flexibility, as e-learning enabled instructors and students to collaborate more effectively. It is also successful in improving institutional procedures that need learner participation. E-learning can also help to promote centralized learning, fast upgrades, a simplified learning process, and reduced costs.

According to Singh. V., & Thurman (2019), the rising number of online resources available over

the web, such as online journals and appropriate websites, provide a rich source for online learners. Even if students are thousands of miles apart, technology allows them to engage online on shared issues and build a sense of community.

According to Raba (2005), objectives can be met in the shortest length of time with the least amount of work with e-learning. Both learners and teachers can achieve and maintain improvement by gaining experience from a variety of experts in diverse sectors of knowledge. With E-learning, educational ethics are guaranteed. This is because e-learning environments are tolerant, making them an excellent approach to providing equal access to the information world regardless of the users' locations, ages, ethnic origins, and races. Because teachers are no longer the sole source of knowledge, the e-learning environment encourages learners or students to rely on themselves.

Instead, they become

## **2.7 DISADVANTAGES OF E-LEARNING**

Despite the numerous benefits of e-learning, students face several obstacles that eventually result in limited or unfavorable outcomes. In their study, Arkorful and Abaidoo (2015) found that e-learning is sometimes held through distance and reflection, resulting in a lack of student involvement. Because there is no face-to-face interaction with instructions or teachers, e-learning may be less successful than traditional schooling. Because assessments are typically held online under the e-learning technique, it reduces the possibility of regulating illegal actions such as cheating and plagiarism.

Technology is a platform that can be taken for granted when used in daily life, but it is not extensively used due to the lack of financial incentives for gaining access. Increased use of computers and other electronic devices among students is driving the worldwide knowledge

available on the internet (Talebian, Mohammadi, and, Rezvanfar, 2014). Another problem that online learners face is sustaining motivation in an online course. When compared to their peers, students who lack self-motivation and independence have lower success rates (Sarkar, 2012).

Learner concerns, educator issues, and content issues all provide barriers to online learning. Institutions face a difficult task in engaging students and getting them to participate in the teaching-learning process. Teachers face difficulties in transitioning from offline to online mode, adjusting their teaching approaches, and managing their time. It's difficult to create content that covers the curriculum while also engaging pupils (Kebritchi et al., 2017).

## **2.8 ONLINE LEARNING IN GHANA**

Ghana's tertiary education system is divided into two categories: state and private tertiary institutions. Both types of students are accepted into remote learning programs, though in different ways. The dual model is used by public universities, which rely primarily on print media, whereas private universities rely heavily on ICT. Ghana's remote education program's objective is to make quality education more accessible and relevant at all levels to satisfy the learning needs of Ghanaians to improve their performance and quality of life. It also aims to provide an alternative to old methods and ensure that physical and human resources are used wisely (Government of Ghana, 2002).

Because of its critical role in education, the Ghanaian government has committed numerous resources and implemented numerous measures to guarantee that ICT is available to all students in Ghana. The ICT for Accelerated Development (ICT4AD) policy, first implemented in 2003, defined a framework for transforming Ghana into information and knowledge-driven ICT literate nation (Ministry of Education, 2015).

ICT is widely assumed to be a priority for Ghana to fulfill global economic, social, and political

demands. As a result, Ghana's Ministry of Education implemented the "one laptop, one pupil" policy, distributing 1,000 computers to 30 schools (three in each region) across the country (Ernest Wesley Otoo, 2010). Another 60,000 laptops were purchased, with 2500 junior high schools receiving them (Naureen, 2012).

In Ghana, innovative learning takes numerous forms. Online learning, remote learning, sandwich learning, evening study, and weekend school are the most common. Through an Innovative Learning Environment, both state-owned and privately-owned institutions of higher learning provide distance/sandwich and e-learning academic programs (ILE). The University of Cape Coast, the University of Ghana, the University of Education, Winneba, and the University of Professional Studies are among the state-owned universities. Valley View University, Methodist University College, Central University, and Presbyterian University College are all privately owned universities. The University of Cape Coast's College of Distance Education, which began operations in 1997, was probably the forerunner of IL programs in Ghana (University of Cape Coast, 2019).

Mathematics, science and engineering, humanities, business studies, and education are just a few of the academic concentrations where these programs are available today. The primary goal of utilizing ILE to deliver such programs in Ghana is to provide learning chances for service members who are competent for higher education but do not favor face-to-face study (University of Ghana, 2014).

## **2.9 CHAPTER SUMMARY.**

Chapter two presents a summary of a literature review, which entails a brief introduction to the topic. This chapter explains or defines certain definitions such as information and communication

technology (ICT), Online Platforms, and Online Educational Platforms. In addition to the definition of concepts, the evolution of online platforms was also discussed. Furthermore, there was a brief description of existing online educational platforms such as Quora, an outline of some features of these existing online educational platforms, benefits of online educational platforms, disadvantages of e-learning, and finally online learning in Ghana.

## **CHAPTER THREE.**

### **METHODOLOGY**

#### **3.1 INTRODUCTION.**

In software engineering, a software development methodology is a division of software development work into distinct phases containing activities with the intent of better planning and management. The development of software projects in the past has been carried out based on different methodologies applied by software developers. Each methodology used is informed by the type of project, the organization, and the developers involved in seeing the execution of the project to completion. Some common methodologies include waterfall, iterative and incremental development, spiral development, rapid application development, extreme programming, etc.

According to software engineering, Project methodology is defined as a combination of logically related practices, methods, and processes that determine how best to plan, develop, control, and deliver a project throughout the continuous implementation process until successful completion and termination.

Project Methodology is a scientifically proven, systematic, and disciplined approach to software design, execution, and completion.

#### **3.2 DEVELOPMENT METHODOLOGY.**

The purpose of development methodology is to allow for controlling the entire software process through effective decision-making and problem-solving, while ensuring the success of specific processes, approaches, techniques methods, and technologies.

Project methodology defines various project management phases or processes: define, plan,

launch, manage, and close. Each phase addresses a specific aspect of managing a project or program from definition through close. Although these phases are described sequentially, in practice some of these phases may overlap or be applied concurrently during the lifetime of a project.

Some of the software engineering methodologies are further elaborated on below.

### **3.2.1 WATERFALL MODEL.**

This software development methodology is considered the traditional method of explaining the software development processes in software engineering. Waterfall Model happens to clarify the processes into a linear flow with a specified sequence to let the users understand that a further level is made progressively on completion of the previous one. Moreover, this methodology also talks about the fact that going back to deal with the changes is not possible.

The waterfall model undergoes the stages of requirement definition, system software design, implementation, and unit testing, integration and system testing, and operation and maintenance

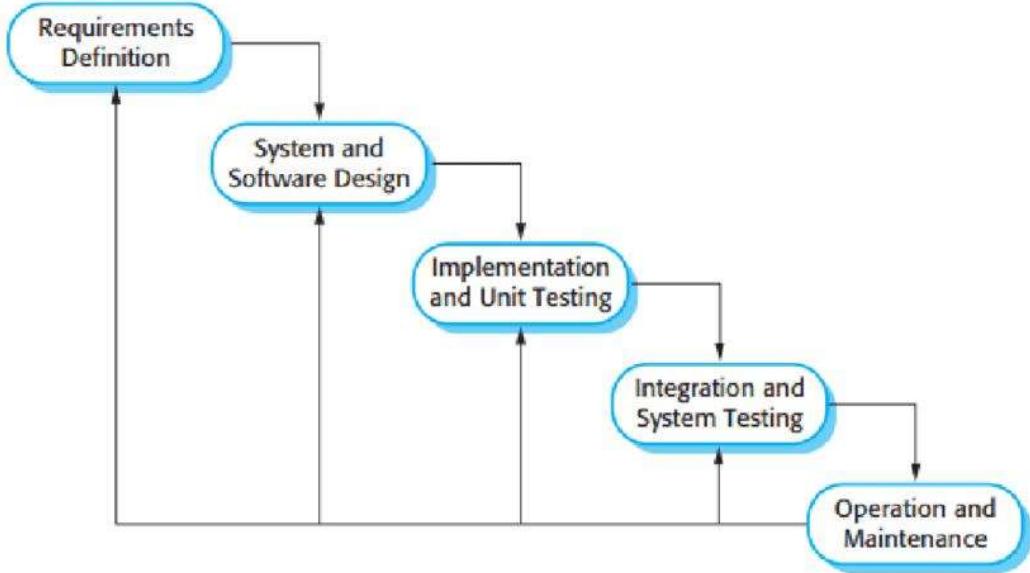


Figure 3.1 Waterfall Model Diagram

### **3.2.1.1 ADVANTAGES OF THE WATERFALL MODEL.**

- It uses a clear structure (defined set of steps).
- It determines the end goal early.
- Phases are processed and completed one at a time.
- Process and results are well documented

### **3.2.1.2 DISADVANTAGES OF THE WATERFALL MODEL.**

- Difficulty in making changes.
- Poor model for long and ongoing projects.
- Cannot accommodate changing requirements.

### **3.2.2 INCREMENTAL MODEL.**

The incremental model is a method of software development where the model is designed, implemented, and tested incrementally. It is based on the idea of developing an initial implementation, getting feedback from users and others, and evolving the software through several versions until the required system has been developed. Specification, development, and validation activities are interleaved rather than separate, with rapid feedback across activities (Sommerville, I., 2016).

In this model, requirements are divided into many standalone units, where each unit or module goes through the software activities and every release of a module adds functions to the previous release.

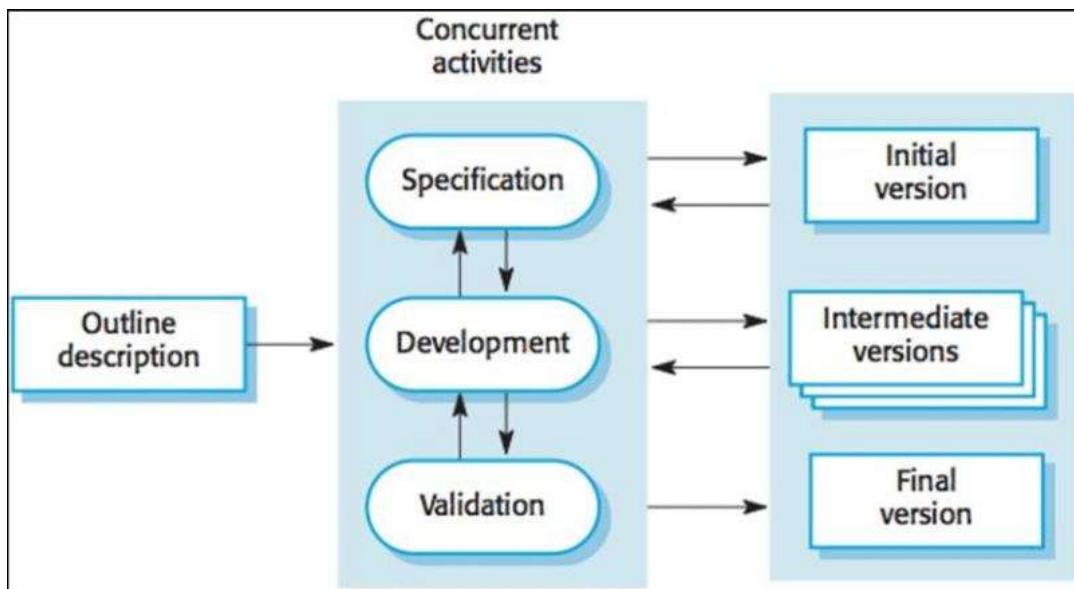


Figure 3.2 Incremental Model Diagram

### **3.2.2.1 ADVANTAGE OF INCREMENTAL MODEL.**

- It is more flexible making it less costly to change scope and requirements.
- It is easy to manage risk because of iteration.
- Version modification is easy
- It is easy for errors to be recognized.

### **3.2.2.2 DISADVANTAGES OF INCREMENTAL MODEL.**

- The total cost of using this model is high.
- It needs a clear, well-defined, and complete definition of the whole system before it can be broken down and built incrementally.

## **3.3 PROJECT METHOD ADAPTED AND JUSTIFICATION.**

In this project, the Agile method is the approach that will be used for the development of the web application. The Agile model is a type of incremental model that breaks tasks into smaller iterations. This project requires incremental implementation in smaller functionalities which will be put together at the end of the system and also require the user's involvement to have timely visible results. Unlike other approaches like the Waterfall model, the Agile model was adopted because of how easy it is to adapt to changes during the implementation stage. The Agile model shares some fundamental characteristics which are as follows:

- The client is satisfied with the rapid development of the system as working prototypes are released.
- The user and the developer are both actively involved in the development process.
- There is fast and timely delivery of development results.

- The direction of the project is influenced by the client.
- Functionalities are delivered at regular intervals.

### **3.4 REQUIREMENT GATHERING PROCESS.**

The requirement-gathering process, an essential part of system design involves data collection from the various stakeholders of the system to be developed. In our requirement process, we use the qualitative method of gathering information which involved conducting interviews with some students and lecturers and case studies on the academic performance of sampled students. It was during this gathering process we identified the requirements needed to solve the identified problem.

### **3.5 REQUIREMENT SPECIFICATION**

The requirements of a system are the total description of the services that a system provides which reflect the needs of the customer or the client of the system. The requirement can be classified into two categories, the user requirement, and the system requirement.

The user requirements are statements that are written for customers normally in the natural language of the services the system provides and its operational constraints while system requirements are structured documents that sets out detailed descriptions of the system's functions, services, and operational constraints.

The system requirement further has 2 types which are;

- Functional requirement of the system.
- Nonfunctional requirement of the system.

#### **3.5.1 FUNCTIONAL REQUIREMENT OF THE SYSTEM.**

The functional requirement of the system are statements of services the system should provide, how the system should react to certain inputs and how the system should behave in certain

situations that is what the system is or is not supposed to do. Some of the functional system requirements are as follows:

- The user shall be able to create an account by signing up with a username, password, email, and institution.
- The user shall be able to search for eBooks.
- The user shall be able to search for an online study mate.
- The user shall be able to send a request or follow a study mate.
- The user shall be able to send be able to chat with study mates.
- The user shall be able to send videos, voice notes, files, and much more

### **3.5.2 NON-FUNCTIONAL REQUIREMENT OF THE SYSTEM.**

Non-functional requirements are requirements that are not directly concerned with specific services delivered by the system to its users. This requirement is a global constraint on a software system and also addresses vital issues of quality for the software system relating to the emergent system properties. These requirements can be classified as performance constraints, operating constraints, interface constraints, economic constraints, and lifecycle constraints. Some of the non-functional requirements of the system are as follows:

- **Usability:** The system is focused on the user experience, in terms of the level of difficulty associated with the learning and operation of the system.
- **Security:** This focuses on protecting the system and data. How much time and effort it will take to break the system?
- **Reliability:** This looks at the question “how often will the system fail?” that is, the probability and percentage of the software performing without failure for a specific amount of time or a specific number of users.

- **Maintainability:** This requirement indicates the average time and ease and rapidity with which a system can be restored after a failure.

### **3.6 UNIFIED MODELING LANGUAGE.**

A Unified Modelling Language Diagram is a standardized modeling language consisting of a family of an integrated set of diagrams, developed to help system and software developers for specifying, visualizing, constructing, and documenting the artifacts of software systems in describing a software system (Martin Fowler,2003). There are several diagrams in the UML diagrams. These are class diagrams, activity diagrams, sequence diagrams, object diagrams, package diagrams, and use case diagrams.

#### **3.6.1 USE CASE DIAGRAM.**

A **use case diagram** is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures.



Figure 3.3 Use Case Diagram of Computer Science (CS) Study Mate System

### 3.6.2 FLOWCHART

A **flowchart** is a type of diagram that represents an algorithm, workflow, or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows (Sanghani, 2019). This diagrammatic representation illustrates a solution model to a given problem.

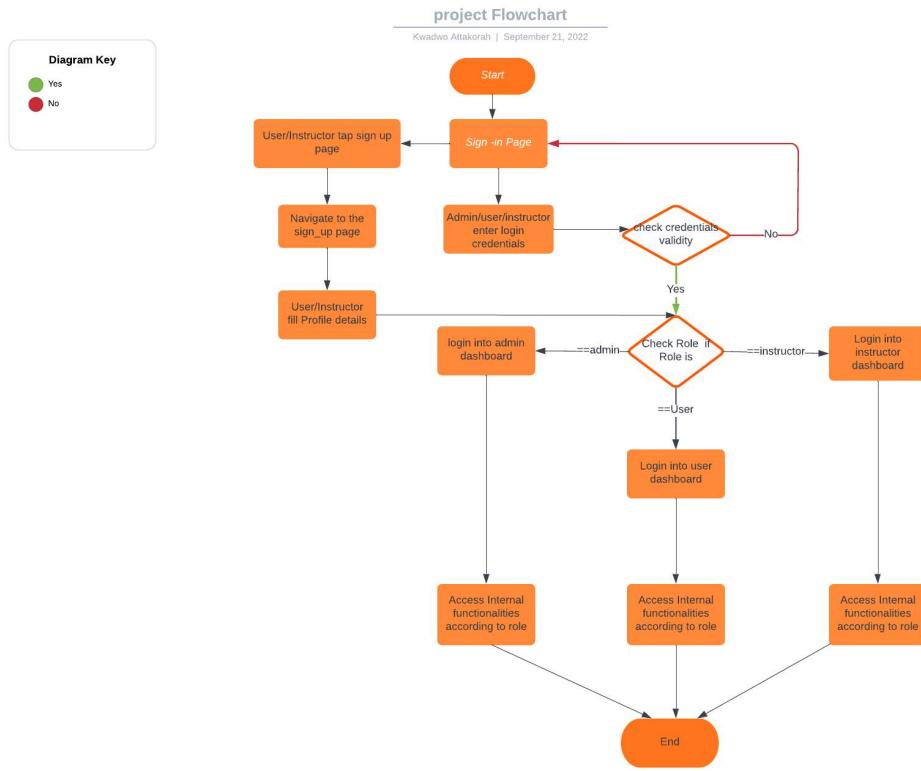


Figure 3.4 Flowchart diagram of the Computer Science (CS) StudyMate System

### 3.6.3 CONTEXT DIAGRAM.

A system context diagram (SCD) is a diagram that defines the border between the system, or part of a system, and its environment, illustrating the entities that interact with it. This diagram represents and gives a high-level perspective of a system.

It shows every external entity that might engage in communication with the system. This diagram portrays the system at the center, with no details of its interior structure, surrounded by all its interacting systems, environments, and activities.

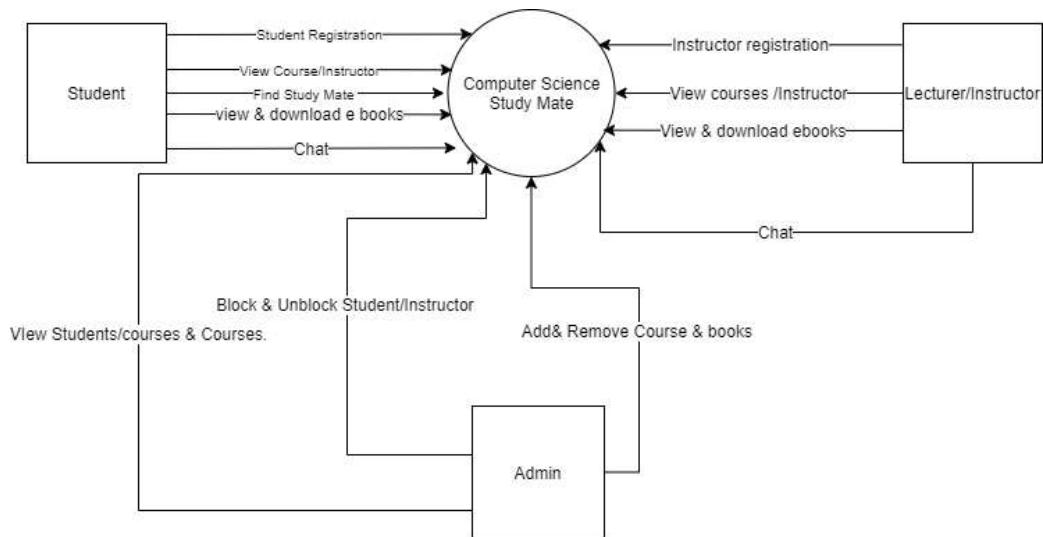


Figure 3.5 Context Diagram of the system

#### **3.6.4 SYSTEM ARCHITECTURE.**

The software architecture diagram is a graphic representation of every element that makes up a system, whether it be in its entirety or part. It is a portrayal of a group of concepts that form architecture. It serves as an abstract representation of the relationships, constraints, and basic layout of the software system.

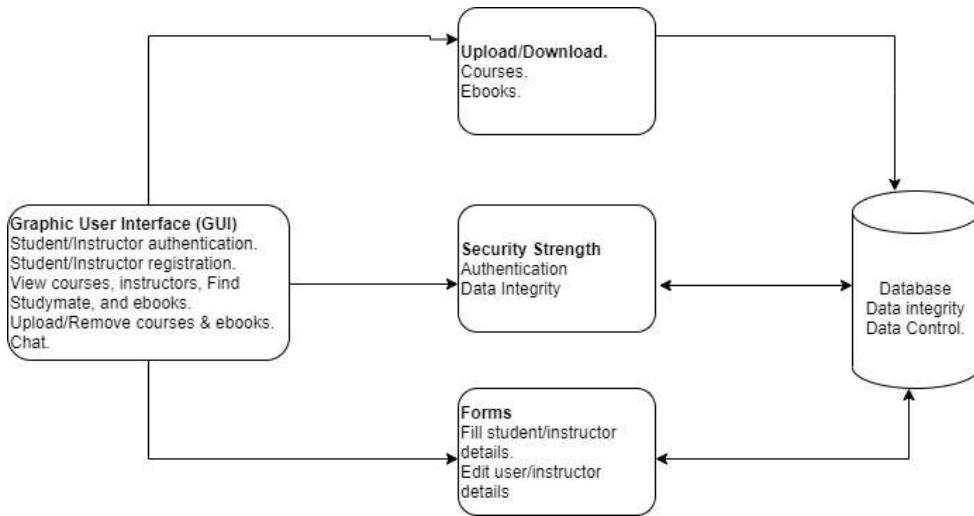


Figure 3.6 System Architecture diagram of the system

### **3.7 PROJECT DESIGN CONSIDERATION. (LOGICAL DESIGN).**

#### **3.7.1 USER INTERFACE DESIGN**

At this point, we would look at how the proposed system would look like, that is, how the proposed system should appear and interact after it is in operation.

The web application would have a sign-in page for students, instructors, and the administrator where there would be links that would lead to either the student or instructor sign-up page Which will enable them to register to join the platform. In the course sign-up onto the platform, the user is either the student or the instructor would be further directed to a fill profile page to complete the registration.

There would be a home page for the instructor and student and an administrator dashboard. The home page of the student would have a navigation bar would contain links to pages of the various courses on the platform, an e-book library where eBooks would be viewed and downloaded, and study mates where fellow students on the platform would be seen. The instructor home page would contain all as mentioned pertaining to the student except that of find mate. Both students and

instructors would have pages to enable them to chat as well.

The administrator dashboard would have sessions where the administrator can view students and instructors on the platform and either block or unblock them from the platform.

Lastly, the administration page would have pages where they would be to upload new courses and eBooks.

### **3.7.2 DATABASE DESIGN.**

A database would be needed for storage where data for the web application would be stored and retrieved.

In the design of the database there would table created such as the following;

- Courses
- Institutions
- Books.
- Course Instructors
- Users
- Instructors
- Request
- Studymate.

Below illustrates the database schema and the E-R diagram of the proposed system.

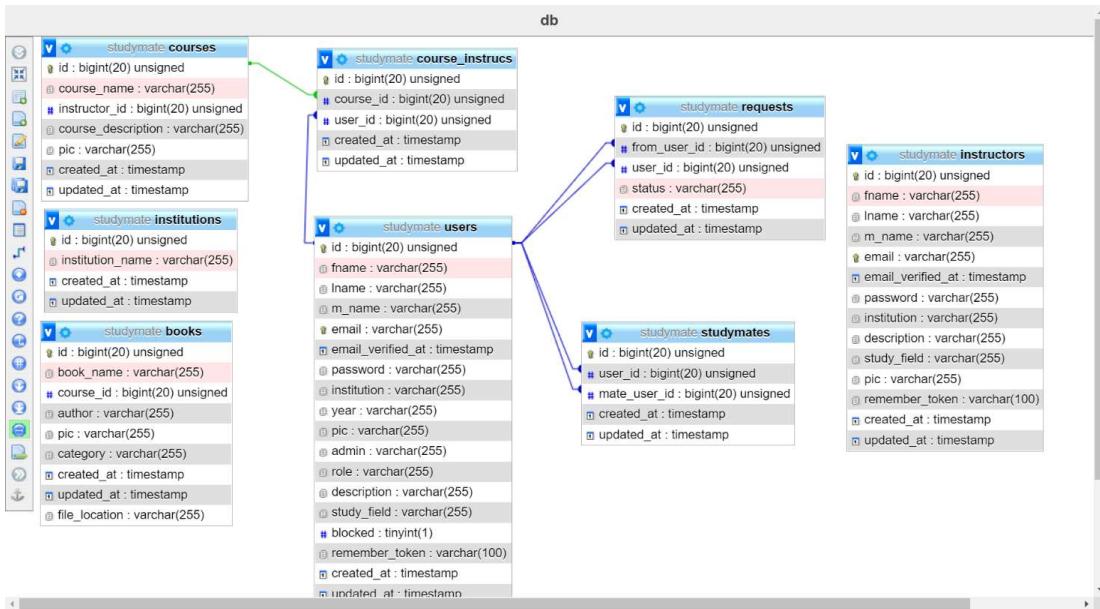


Figure 3.7 Database Schema of the system.

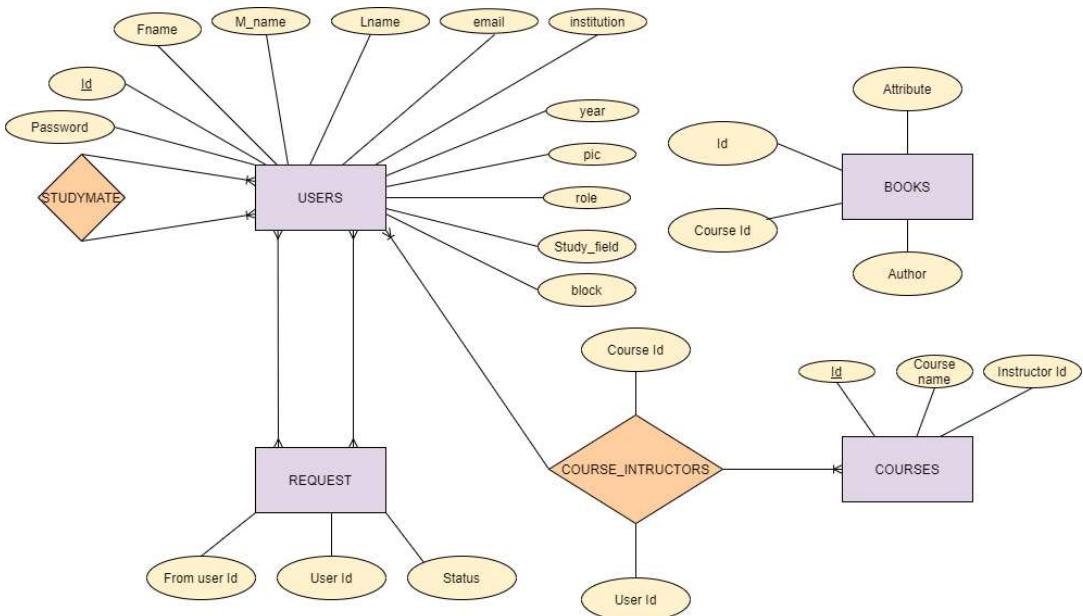


Figure 3.8 Database E-R diagram of the system

### **3.8 DEVELOPMENT TOOLS AND ENVIRONMENT.**

#### **3.8.1 DEVELOPMENT ENVIRONMENT.**

- Visual Studio code.

#### **3.8.2 DEVELOPMENT SOFTWARE TOOLS.**

User Interface (U.I)/ Front end

- Html
- CSS

Backend

- PHP (Laravel Framework)

### **3.9 CHAPTER SUMMARY.**

This chapter presents the system methodology, which entails a brief introduction to the topic, the various types and their advantages and disadvantages, UML diagrams, the flowchart, the context diagram, and the system architecture. This chapter further reveals the project design considered which involves the user interface and database design. (demonstrated using the database schema and E-R design). Finally, the development tools that would be used in the development of the system were listed along with the development environment.

## **CHAPTER FOUR**

### **TESTING AND IMPLEMENTATION.**

#### **4.1 INTRODUCTION.**

Here in this chapter is where proof of the many things said about the system is tested and settled or established. In here, a new system is developed from scratch and so much information is gathered from the existing ones. The focus here is narrowed on the people who will use the system and how best to train them to use and adapt to the new solution we have at hand in this project. Software testing is a process to evaluate the functionality of a software application with the intent to find whether the developed software met the specified requirements or not and to identify the defects to ensure that the product is defect free in order to produce a quality product. Implementation of the system is needful to ensure that the problem for which it was developed is solved. It involves the process of software design, programming, and mapping logical design onto the physical platform; this portion deals with the way in which the various logical modules are mapped unto the physical modules. The logical modules are implemented independently and they depict the physical system's independence of technical implementation and only illustrate the essence of the system. The physical modules, however, are implementation dependent and only illustrate the essence of the system.

#### **4.2 TESTING.**

Software testing is a process to evaluate the functionality of a software application with the intent to find whether the developed software met the specified requirements or not and to identify the defects to ensure that the software or product is free in order to produce the quality product.

Software testing can also give users a perspective on the software that enables them to see and comprehend the risks associated with software implementation. Software testing entails running a software or system component to assess one or more valuable properties. These characteristics typically show the degree to which the component or system being tested;

- Meets the requirement that guided the design and development.
- Responds correctly to all kinds of inputs.
- Performs its functions within an acceptable time.
- It is sufficiently usable.
- It achieves the general results desired by stakeholders.

Since there are almost countless ways to test even the simplest software component, all software testing employs a method to choose tests that can be completed in the time and with the resources that are available. So, in order to detect software bugs, software testing often tries to run a program or application (errors or other defects). Testing is an iterative process because when one fault is corrected, it sometimes reveals deeper bugs or even introduces new ones. Software testing can give users or sponsors unbiased, independent information on the caliber of the software and the risk of its failure.

As soon as executable software (even if it is only partially complete) is there, software testing can begin. Testing is frequently scheduled and carried out in accordance with the general approach taken to software development. The majority of testing happens simultaneously with system development in the incremental development methodology utilized to create this online attendance system.

Different levels of software testing exist. These tiers include usability testing, system testing, and unit testing.

#### **4.2.1 UNIT TESTING.**

Unit testing is also known as component testing each and every component of the program independently by the developer in the developer's environment, is done to ensure that the individual modules of the source code are functioning properly.

Furthermore, unit testing was done on individual modules, of the system. Unit testing is a testing strategy that is used to test the individual components of the system. The test is done to diagnose to know the working status of the various components of the system. Those components with errors are corrected and updated. Below is a table that shows some of the various units that were tested and how they were tested as well as the outcome.

ITEMS TO BE TESTED	TEST CASE	EXPECTED RESULTS	ACTUAL RESULTS
Sign-up unit	The unit was tested by entering required data like <ul style="list-style-type: none"><li>➤ Email</li><li>➤ Password.</li></ul> In the input field	The user (Student/Instructor) should be able to be successfully sign-up.	The signup process was successfully implemented.
Login unit	Valid username and password were entered in the input field	Successful authentication	Authentication successful
View &	Students/Instructors	E-book viewed and	The E-book was viewed

download the E-book.	select E-book.	successfully downloaded	and successfully downloaded.
Chat unit	Student/Instructor sends & receives messages and attachment files.	Student/Instructor successfully sends & receives messages and attachment files	Student/Instructor successfully sends & receives messages and attachment files

Table 4.2.1

The Computer Science Study Mate web application comes in three modules; the administrator module, the lecturer module, and the student module, each of these units have been evaluated on their functionalities, authorization, and authentication. Each module specified satisfies the software requirement set to build the system.

#### **4.2.2. SYSTEM TESTING.**

System testing (ST) is a stage in the software testing cycle that assesses how well the entire system complies with the requirements. Using this technique, testers assess the functionality of the product being tested without examining the internal code layout. comprehensive testing of each input in the application during system testing to ensure expected outcomes.

System testing is typically performed by a team that is separate from the development team in order to objectively assess the system's quality. Functional and non-functional testing are both parts of it. After putting together, the various modules implemented, the system was found to be working as expected fulfilling the requirements gathered.

#### **4.2.3 USABILITY TESTING.**

Usability testing is a method used in user-centered interaction design to assess a product by having users test it out. This provides direct feedback on how actual users interact with the system, making it a vital usability technique. Testing for usability focuses on determining how well a product created by humans can fulfill its intended function. The developed system was very easy to navigate through to the desired destination.

#### **4.2.4 TESTING PLAN**

In order to ensure that the developed system is fully tested, performs set objectives, and is approved, a test plan is conducted. This test plan has two key goals that it wants to achieve. First, check to see if the code is error-free and that the system's final output satisfies the user's need.

The Completed work was hosted locally throughout the testing and installation process.

### **4.3 IMPLEMENTATION.**

This chapter deals with the varied ways by which the various components of the system were made and how they were made. The implementation of these components in the system creates a single entity. The system was implemented using HTML, CSS, PHP (Laravel framework), and MySQL database.

#### **4.3.1 THE HTML IMPLEMENTATION.**

HTML is a markup language that was used to design the input forms of the system. It was also used to design the various navigational tabs of the system. These files are made with HTML tags. The HTML was coded in a simple visual studio code text editor. These files after creation were all put in a single location and are been accessed through hyperlinks.

#### **4.3.2 THE CSS IMPLEMENTATION.**

CSS stands for Cascading Style Sheets. It is the coding language that gives a website a nice and attractive look and layout pleasing to the users of the system (web application). Along with HTML,

CSS is fundamental to web design. Without it, the website would still be plain text on white backgrounds.

#### **4.3.3 DATABASE IMPLEMENTATION.**

The database of the system is a free version of MYSQL. It was aligned with the system from the XAMP local host environment. It contains all tables that are used to feed the system with data.

#### **4.4 MAPPING LOGICAL DESIGN ONTO PHYSICAL PLATFORM.**

Implementation is the stage in the project where the theoretical design is turned into a working system and gives confidence in the new system for the users to work efficiently and effectively. It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the changeover, and evaluation of changeover methods. Apart from planning major tasks of preparing the implementation, are education and training of users. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions are made regarding the equipment and resources and additional equipment has to be acquired to implement the new system. In a network backup system, no additional resources are needed. Implementation is the final and most important phase. The most critical stage in achieving a successful new system is giving the users confidence that the new system will work and be effective. The system can be implemented only after thorough testing is done and if it is found to be working according to the specification.

#### **4.5 CONSTRUCTION.**

This section shows the interfaces of the system showing how the system functions and snippet codes of the system.

#### **4.5.1 DESIGN AND INTERFACE.**



The login page features a header "Welcome to CS Study Mate" and a central illustration of a student and an instructor in a video conference setting. To the right is a "Login into CS Study Mate" form with fields for Email address and Password, and options for Remember me and Forgot password? A blue "Login" button is at the bottom.

Figure 4.1

A sign-in page for both users, instructor, and administrator



The sign-up page features a header "Welcome to CS Study Mate" and a central illustration of a student and an instructor in a video conference setting. To the right is a "Sign up for your account!" form with fields for Email address, Password, and Confirm Password, and a checkbox for terms of service. A blue "Sign Up" button is at the bottom.

Figure 4.2

Sign-up page for both user and instructor.

**Personal information of lecturers**

Lecturer first name \*

Lecturer middle name \*

Lecturer last name \*

**Education**

Name of Institution \*  select institution ..

Study Field \*

**Profile Pic \***  Choose File No file chosen

Figure 4.3

This page shows where the instructor fills in the profile details in the course of his or her registration.

**Personal information of students and lecturers**

Student first name \*

Student middle name

Student last name \*

**Education**

Name of Institution \*  select institution ..

Year of student \*

**Profile Pic \***  Choose File No file chosen

**Submit**

Figure 4.4

This page shows where the student fills in the profile details in the course of his or her registration.



Figure 4.5

Home page

A screenshot showing a section titled 'Most Popular Courses'. It features four cards with icons and counts: '3 Courses' (monitor icon), '5 Students' (two people icon), '3 Instructor' (graduation cap icon), and '3 Books' (book icon). Below this, there's a heading 'Most Popular Courses' with a sub-instruction 'Connect with lecturers on each course'. Four thumbnail images of a course page are shown, each featuring a yellow diamond icon.

Figure 4.6

This page is a section of the home page that allows either the student or instructor to view the number of courses, students, instructors, and books

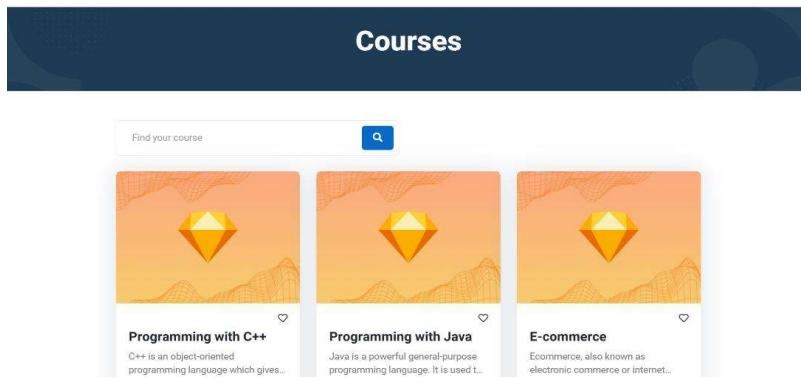


Figure 4.7

This is a section of the home page showing popular courses in computer science.

A screenshot of a course page for "E-commerce". At the top, there is a navigation bar with icons for a laptop, a pencil, and the word "STUDYMATE". Below the navigation bar, there are links for "Courses", "Book Library", "Instructors", and "Find Mate". On the right side of the header, there is a user profile picture. The main title of the page is "E-commerce". Below the title, there are two tabs: "Overview" and "Instructors", with "Instructors" being the active tab. A sub-section titled "Instructors to help with the course" lists one instructor: "Dr. George KNUST".

Figure 4.8

This is a page under the courses showing instructors under that particular course.

The screenshot shows the StudyMate website interface. At the top, there is a navigation bar with links for Courses, Book Library, Instructors, and Find Mate. A user profile icon is also present. Below the navigation bar, the title "E-commerce" is displayed in bold. Underneath the title, there are two tabs: "Overview" (which is selected) and "Instructors". The "Course Description" section follows, containing a brief definition of E-commerce. At the bottom of the page, there are links for "Cs Study Mate" and "Contact".

Figure 4.9

This page is a session of a particular course where it gives a brief description of that course.

The screenshot shows the StudyMate website interface from the perspective of an instructor named Dr Linda Banning. The top navigation bar includes a profile picture, the name "Dr Linda Banning", and a "Chat" button. On the left, a sidebar menu lists "Dashboard", "Edit Profile" (with a checked checkbox), "Courses" (which is selected and highlighted in yellow), "Delete Profile", and "Sign Out". The main content area is titled "Add Course" and contains a table with three rows. The columns are "Course Name" and "Action". The rows list "Programming with C++" (with a checked checkbox in the Action column), "Programming with Java" (with an unchecked checkbox in the Action column), and "E-commerce" (with a checked checkbox in the Action column).

Figure 4.10

This is a page where the instructor is able to add other courses to his/her field of study.

The screenshot shows the 'Edit Profile' interface. On the left is a sidebar with options: Dashboard, Edit Profile (which is selected and highlighted in blue), Courses, Delete Profile, and Sign Out. The main content area is titled 'Edit Profile'. It includes a 'Profile picture' field with a placeholder image of a person in orange, a 'Change' button, and a 'Full name' input field containing 'Dr Linda'. Below that are 'Username' (StudyMate.com) and 'Email id' (linda@gmail.com) fields. Further down are 'Institution' (Legon) and 'Study Field' (empty) fields. At the bottom right is a 'Save changes' button.

Figure 4.11

This page allows the instructor to edit his or her profile and update it.

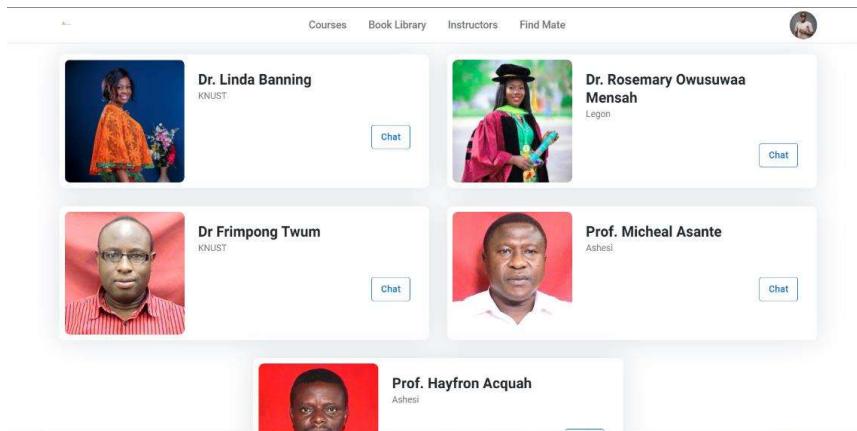


Figure 4.12

This page is the instructor page which displays the lecturers on the platform.

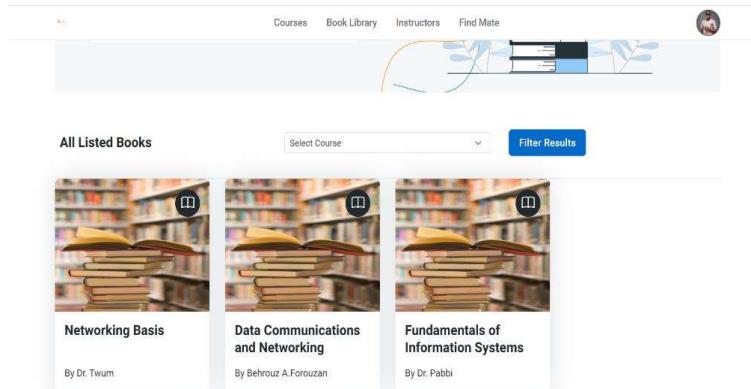


Figure 4.13

This page shows the e-books that have been uploaded onto the platform.

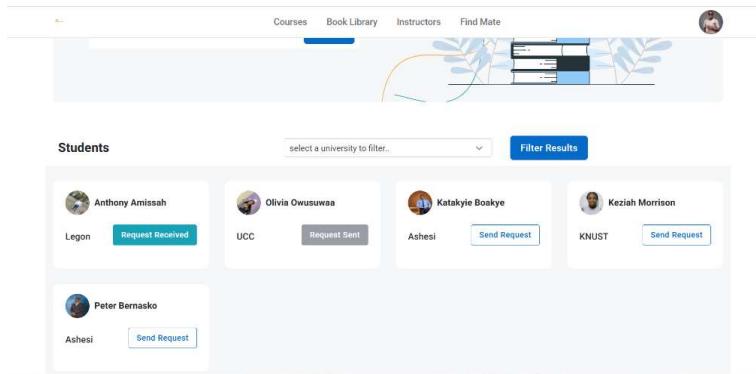


Figure 4.14

This is the Study mate page where a student is able to find a mate and send and accept a study mate request.

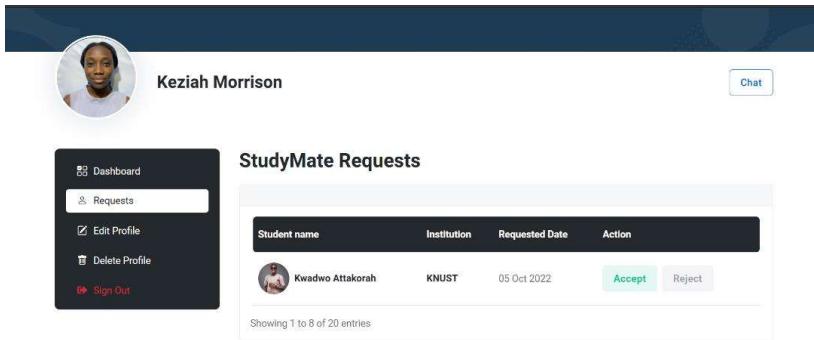


Figure 4.15

This is a page of a student showing the study mate request to either accept or reject.

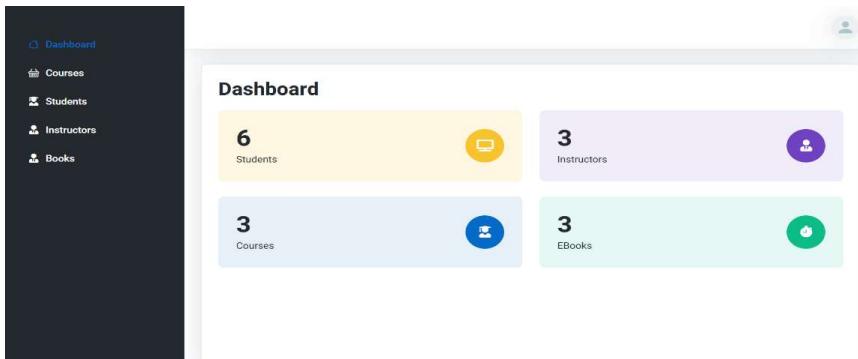


Figure 4.16

This is the administrator's dashboard where he or she is able to view the number of students, instructors, courses, and ebooks.

The screenshot shows a dark-themed administrative interface. On the left is a sidebar with navigation links: Dashboard, Courses (which is the active link, indicated by a blue underline), Students, Instructors, and Books. The main content area is titled "Courses". It displays a summary box with "Total Courses" and the number "3". Below this is a table listing three courses: "Programming with C++", "Programming with Java", and "E-commerce". Each row in the table includes a small thumbnail icon, the course name, and a "Remove" button. A blue "Create a Course" button is located in the top right corner of the main content area.

Figure 4.17

This page shows the administrator's course page where there is a button to add a new course or remove courses.

The screenshot shows a dark-themed administrative interface. On the left is a sidebar with navigation links: Dashboard, Courses (which is the active link, indicated by a blue underline), Students, Instructors, and Books. The main content area is titled "Submit a new Course". It has a section titled "Course details" containing two input fields: "Course title" (with placeholder "Enter course title") and "Short description" (with placeholder "Enter keywords"). A blue "Submit" button is located at the bottom right of the form.

Figure 4.18

This page shows the administrator adds a course page where the admin is able to add new courses.

The screenshot shows a web-based administrative interface. On the left is a dark sidebar with navigation links: Dashboard, Courses, Students (which is highlighted in blue), Instructors, and Books. The main content area has a white background and a title 'Students'. It features a search bar at the top. Below the search bar is a table with two columns: 'Student name' and 'Action'. The table lists five students with their names, profile pictures, and current status (e.g., Kwadwo Attakorah at KNUST). Each row has a red circular 'Action' button to the right.

Student name	Action
Kwadwo Attakorah 📍 KNUST	(Red Action button)
Peter Bernasko 📍 Legon	(Red Action button)
Keziah Morrison 📍 UCC	(Red Action button)
Olivia Owusuwa 📍 Ashesi	(Red Action button)
Katakyie Boakye	(Red Action button)

Figure 4.19

This show the administrator's student page where there is a button to block and unblock student user from the platform.

The screenshot shows a web-based administrative interface. On the left is a dark sidebar with navigation links: Dashboard, Courses, Students, Instructors, and Books (which is highlighted in blue). The main content area has a white background and a title 'Books'. It features a search bar at the top and a blue 'Add a Book' button on the right. Below the search bar is a table with two columns: 'Book Name' and 'Action'. The table lists three books with their names, authors, and current status (e.g., Networking Basis by Dr. Twum). Each row has a grey circular 'Action' button to the right.

Book Name	Action
Networking Basis 📍 Dr. Twum	(Grey Action button)
Data Communications and Networking 📍 Behrouz A. Forouzan	(Grey Action button)
Fundamentals of Information Systems 📍 Dr. Pabbi	(Grey Action button)

Figure 4.20

This show the administrator's book page where there is a button to add new books or remove books.

The screenshot shows a web-based administrative interface for adding a new book. On the left is a dark sidebar menu with white icons and text, listing: Dashboard, Courses, Students, Instructors, and Books. The Books option is highlighted with a blue icon. The main content area has a light gray header bar with the title "Add a New Book". Below this is a section titled "Book details" containing four input fields: "Book title" (with placeholder "Enter course title"), "Author" (with placeholder "Enter keywords"), "Category" (a dropdown menu with placeholder "Select Course"), and a file upload field with a blue camera icon and the placeholder "Upload image".

Figure 4.21

This show the administrator adds a book page where the admin is able to upload a new book.

## **CHAPTER 5**

### **SUMMARY, CONCLUSION, AND RECOMMENDATION.**

#### **5.1 INTRODUCTION.**

This chapter tends to make a summary of the entire project, discussing the findings, and limitations, drawing conclusions, and making recommendations.

#### **5.2 SUMMARY.**

After a view into the educational system as a whole and zooming into that of Ghana, we identified online education and online educational platform as interesting to us. We went ahead to make further research on the online educational platforms concerning computer science students and we realized there was a need to help computer science students improve upon the experiences they have. For this reason, we came out with a Computer Science Study Mate web application.

This project covered the scope of some public universities such as Kwame Nkrumah of the University of Science and Technology, the University of Ghana and the University of Development Studies, and a private university which was Ashesi university.

There were a lot of reviews on some existing online learning platforms and most of the reviewed platforms falls into two major categories according to our research. These categories were questioning and answering online learning platforms and online course learning platform

This project development involved the use of the Agile model which is a type of incremental model of the software engineering development process. The Project was developed using the Laravel framework for the front end and the use of PHP for the back end.

Explaining further the project development was the use of graphic representation to illustrate the functions and various components of the system. This graphic representation involved the use of

flowcharts, use case diagram, context diagram, and system architecture diagram.

The system was tested to check whether it met the stated objectives and the result was positive, meaning the system functioned as expected.

### **5.3 FINDINGS**

From our research topic, we identified the challenges that were faced by computer science students in the various universities across the country which called for a solution of developing the computer science study mate web application. This project met the objectives that were stated earlier in chapter one. The Project documentation gives a step by step guidelines on how to use the system developed which is easy to learn and navigate, thus meeting the system's non-functional requirements as stated earlier. The developed web application was successfully tested as the various components and features of the system were found to work as expected.

### **5.4 LIMITATION.**

After building this system with the aim of solving the identified problem in the problem statement, there was a realization of the limitation of the system, and they are as follows,

- Internet dependence -Without a reliable internet connection we cannot run the system or the web application
- Supported device – Without an internet enable device the web application cannot be run.
- Web issues – The application is entirely dependent on a website. This means that, if the website happens to undergo failure, then most likely the application will be failing too.
- Browser versions – As there are versions of a specific browser, some versions of browsers are outdated hence not being able to support the web application.

## **5.5 CONCLUSION.**

In conclusion, this research project picked pointed ways to improve the experiences in the learning of computer science, we found out that developing a fully working system is difficult but requires a lot of knowledge in database design and some programming languages such as PHP, Html, and CSS

The developed web application is very useful and can play a major role in the development of our Ghanaian tertiary education sector to be specific and the Ghanaian educational sector in general.

The computer science study mate web application is a platform for computer science students to connect and share ideas and experiences through peer-to-peer and student-to-lecturer online studies. The core aim of establishing a computer science study mate web application is to ensure engaged learning and offer computer science students and lecturers the opportunity to discuss several academic issues and misunderstandings that will help clear some of the fears and challenges that have constructed barriers to the learning of the various courses as part of the computer science program.

This web application enables students to send, accept and decline study mate request, download eBooks, have discussions with study mates and lecturers, and much more. Thus, the core aim of this web application has been implemented successfully.

## **5.6 RECOMMENDATIONS**

However, like any other system, more features can be added later to make the application even more interesting and fun to use. I will highly recommend for future researchers consider adding a page where student knowledge gained from the platform would be tested by posting challenging projects for students to come together and undertake with the supervision of the instructors on the platform. And also adding a mobile version of the software to the application development.

## REFERENCES

- Arkorful, V., and Abaidoo, N., (2015). *The role of e-learning, advantages, and disadvantages of its adoption in higher education*. International Journal of Instructional Technology and Distance Learning, 12(1), pp. 29-42.
- Alhassan, R. K. (2020). *Assessing the preparedness and feasibility of an e-learning pilot project for university-level health trainees in Ghana: a cross-sectional descriptive survey*. BMC Medical Education, 20(1), 1–10.5. Lwoga E. Making learning and Web 2.0 technologies work for higher.
- Campbell, L. (2019). *Teaching in an inspiring learning space: An investigation of the extent to which one school's innovative learning environment has impacted teachers' pedagogy and practice*. Research Papers in Education, 1-20
- Emarketer, (2015). *Worldwide retail e-commerce sales: Emarketer's updated estimates and forecast through 2019*. New York
- Ernest Wesley Otoo, (2010). *Education Sector Performance Report*. Accra: Ghana Ministry of Education.
- Friedrich, H. F., & Hron, A. (2011). *Factors affecting teachers' student-centered classroom computer use*. Educational Media International, 48(4), pp. 273-285.
- Gautam, S. S., and Tiwari, M. K., (2016). *Components and benefits of e-learning system*. International Research Journal of Computer Science (IRJCS), 3(1), pp. 14-17.
- Government of Ghana (2002). *Meeting the Challenges of Education in the Twenty-First Century. Report of the President's Committee on Review of Education Reforms in Ghana*. Accra: Government of Ghana.
- Hernández-Ramos, J. P., Martínez-Abad, F., Peñalvo, F. J. G., García, M.E. H., &

RodríguezConde, M. J. (2014). *Teachers' attitude regarding the use of ICT. A factor reliability*

*and validity study.* Computers in Human Behavior, 31, pp.509-516.

Kreijnsa, K., van Acker, F., Vermeulend, M., & van Buuren, H. (2014). *What stimulates teachers to integrate ICT into their pedagogical practices? The use of digital learning materials in education.* Computers in Human Behavior, 29(1), pp.217-225.

Martin Fowler, (2003). *What is the UML? UML Distilled: A Brief Guide to the Standard Object Modeling Language* (3rd Edition).

Megha Gokhe (2012). *Definition of Information and Communication Technology.* Information and Communication Technology, pp 1.

Ministry of Education. (2015). *ICT in education policy.* Accra.

Muruthy, A. E., and Yamin, F. M., (2017). *The perception and effectiveness of Learning Management System (LMS) usage among Higher Education students.* Journal of Technology and Operations Management, 12(1), pp.86-98.

Naureen, (2012). *Education Sector Performance Report.* Accra: Ghana Ministry of Education.

OECD (2019), *An Introduction to Online Platforms and Their Role in the Digital Transformation,* OECD Publishing, Paris, <https://doi.org/10.1787/53e5f593-en>.

Rabah, M. (2005). *E-learning*, Jordan: Dar Almnahej Publisher

Sanghani, P., Sanket, B., Rajas, T., Atharva, A., Manali, P. and Aatu, P., 2019. Data visualization. *Neurocritical Care Informatics: Translating Raw Data into Bedside Action*, 3, pp.135-147

Sarkar, S. (2012). *The role of information and communication technology (ICT) in higher*

- education for the 21st century.* Science, 1(1), pp. 30-41.
- Serdyukov, P. (2017). *Innovation in education: what works, what doesn't, and what to do about it?* Journal of Research in Innovative Teaching & Learning, 10(1), 4-33.
- Terzi, N. (2014). The impact of e-commerce on international trade and employment. 7th International Strategic.
- Singh, V., & Thuman, A. (2019) *How many ways can we define online learning? A systematic literature review of definitions of online learning (1988–2018).* American Journal of Distance Education, 33(4), pp.289–306.
- Sommerville, I., 2016. *Software Engineering 10th Edition (International Computer Science).* Essex, UK: Pearson Education, pp.1-808
- Talebian, S., Mohammad, H. M., and Rezvanfar, A., (2014). *Information and communication technology (ICT) in higher education: advantages, disadvantages, conveniences, and limitations of applying e-learning to agricultural students in Iran.* Procedia-Social and Behavioral Sciences, 152, pp. 300-305.
- The University of Cape Coast, (2019). *Distance education.* Retrieve from  
<https://ucc.edu.gh/admission-notices/sandwich-admissions-undergraduate> [accessed 9<sup>th</sup> August 2022]
- The University of Ghana. (2014). MA Sandwich program. Retrieved from  
<http://www.ug.edu.gh/esl/masandwich-program> [accessed 9<sup>th</sup> August 2022]
- Van Alstyne, M. W., Parker, G. G., & Choudary, S. P. (2016). *Pipelines, platforms, and the new rules of strategy.* Harvard Business Review, 94(4), pp. 54-62.

## APPENDIX.

### Appendix 1

Logic codes for the functions of the Administrator.

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

// use Illuminate\Http\Request;
use App\Models\Course;
use App\Models\Instructor;
use App\Models\User;
use App\Models\Book;

use Illuminate\Support\Facades\Auth;

class AdminController extends Controller
{
    //

    public function __construct(){
        $this->middleware('admin');
    }

    function index(){

        $students = User::where('role', '0')->count();
        $instructors = User::where('role', '2')->count();
        $books = Book::count();
        $courses = Course::count();

        // User::get();

        return view('admin.admin_dashboard')
            ->with(compact('students'))
            ->with(compact('instructors'))
            ->with(compact('books'))
            ->with(compact('courses'));
    }

    function show_courses(){
}
```

```

$courses = Course::paginate(3);
$courses_total = Course::count();
return view('admin.admin_courses')
    ->with(compact('courses'))
    ->with(compact('courses_total'));
}

function new_course(Request $request){
    // dd($request);
    $this->validate($request, [
        'course_title' => 'required'
    ]);

    Course::create([
        'course_name' => $request->course_title,
        'course_description' => $request->course_description,
        // 'pic' => $request->pic
    ]);

    return redirect()->route('admin_courses');
}

function show_instructors(){
    // $instructors = Instructor::all();
    $instructors = User::where('role', '2')->get();

    return view('admin.admin_instructors', compact('instructors'));
}

function show_users(){
    $users = User::where('role', '0')->paginate(10);

    return view('admin.admin_users', compact('users'));
}

function show_instructor_detail($id){
    $instructor = User::find($id);

    $courses = collect([]);
    // if($instructor->courses()){
    //     $courses = $instructor->courses()->paginate(4);
    // }
    return view('admin.admin_instructor_detail')
}

```

```

->with(compact('instructor'))
->with(compact('courses'));
}

function show_user_detail($id){

    $user = User::find($id);
    return view('admin.admin_user_detail', compact('user'));
}

function show_course_detail($id){
    $course = Course::find($id);
    $instructors = $course->instructors()->get();

    return view('admin.admin_course_detail')
        ->with(compact('course'))
        ->with(compact('instructors'));
}

function show_course_create(){
    return view('admin.admin_course_create');
}

function show_books(){

    $books = Book::paginate(4);
    return view('admin.admin_books', compact('books'));
}

function show_book_new(){
    return view('admin.admin_book_library_new_entry');
}

function new_book(Request $request){
    $this->validate($request, [
        'book_title' => 'required'
    ]);
    $filename = null;

    if($request->file('book_file')){
        // dd($request->file('book_file'));

        $file = $request->file('book_file');
        $filename = time().'.'.$file->getClientOriginalExtension();
        // $filename= date('YmdHi').$file->getClientOriginalName();
    }
}

```

```

// dd($filename);
$file->storeAs('public/books/test', $filename);

}

// dd('nope');

Book::create([
    'book_name' => $request->book_title,
    'author' => $request->author,
    'pic' => $request->pic,
    'file_location' => $filename,
    // 'file_no_loca' => $request->file('book_file')->getClientOriginalName()
]);

return redirect()->route('admin_books');
}

function sign_out(Request $request){
    Auth::logout();

    return redirect()->route('sign_in');
}

function block_user(Request $request){
    $user = User::find($request->user_id);

    // check if user is already
    if($user->blocked){
        $user->fill([
            'blocked' => null
        ]);

        $user->save();
    } else {
        $user->fill([
            'blocked' => true
        ]);
    }
}

```

```
    $user->save();
    // dd('the user has been blocked', $user);
}

return redirect('admin-users');

}

function delete_course(Request $request){
    Course::where('id', $request->course_id)->delete();

    return redirect()->route('admin_courses');
}

function delete_book(Request $request){
    Book::where('id', $request->book_id)->delete();

    return redirect('admin-books');
}
```

## Appendix 2.

Logic codes for Authentication.

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
use Illuminate\Support\Facades\Auth;
use App\Models\User;
use App\Models\Instructor;

class AuthController extends Controller
{

    function __construct(){
        $this->middleware('guest');
    }
    //
    function sign_up(){
        return view('signup');
    }
    function sign_in(){
        return view('signin');
    }

    function sign_up_user(Request $request){
        $this->validate($request, [
            'email' => 'required|email',
            'password' => 'required',
            'confirm_password' => 'required'
        ]);

        User::create([
            'email' => $request->email,
            'password' => Hash::make($request->password),
        ]);

        Auth::attempt($request->only('email', 'password'));

        return redirect()->route('user_details');
    }
}
```

```

function sign_in_user(Request $request){
    $this->validate($request, [
        'email' => 'required|email',
        'password' => 'required',
        // 'confirm_password' => 'required'
    ]);
}

// dd($request);
// User::create([
//     'email' => $request->email,
//     'password' => $request->password,
// ]);

$user_ = User::where('email', $request->email)->where('blocked', true)->count();
if($user_ > 0){
    return redirect()->route('sign_in');
}

Auth::attempt($request->only('email', 'password'), $request->remember);

return redirect()->route('home');
}

function get_details(){
    return view('user_details');
}

function forgot_password(){
    return view('forgot_password');
}

function instruc_register(){
    return view('instruc_register');
}

function register_instruc(Request $request){

$this->validate($request, [
    'email' => 'required|email',
    'password' => 'required',
    'confirm_password' => 'required'
]);
}

```

```
// Instructor::create([
//   'email' => $request->email,
//   'password' => Hash::make($request->password),
// ]);

User::create([
  'email' => $request->email,
  'password' => Hash::make($request->password),
  'role' => '2'
]);

Auth::attempt($request->only('email', 'password'));

return redirect()->route('instruc_details');

}

}
```

### Appendix 3.

Logic codes for book upload and download books.

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Storage;

use App\Models\Book;

class BookController extends Controller
{
    //

    function __construct(){
        $this->middleware('auth');
    }

    function index(){

        $books = Book::paginate(3);

        return view('book_library', compact('books'));
    }

    function filter(Request $request){

        if($request->category == ""){

            return redirect('/book-library');

        }

        // dd($request->category);
        $books = Book::where('category', $request->category)->paginate(3);

        return view('book_library', compact('books'));
    }

    function download($id){

        $book = Book::find($id);

        if(Storage::disk('public')->exists("books/test/".$book->file_locatin)){
            // dd('good');
        }
    }
}
```

```
}

// dd('bad');

$file_location = '';
$path = Storage::disk('public')->path("books/test/".$book->file_location);
// dd($path);
// dd(file_get_contents($path));
$content = file_get_contents($path);

return response($content)->withHeaders([
    'Content-Type' => mime_content_type($path)
]);
// return Storage::download($path);

// return redirect('/404');

}

}
```

## Appendix 4

Logic codes for adding and removing courses.

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\Course;

class CourseController extends Controller
{
    //
    function __construct(){
        $this->middleware('auth');
    }

    function index(){

        $courses = Course::paginate(3);

        return view('courses', compact('courses'));
    }

    function detail($id){

        $course = Course::find($id);

        $instructors = $course->instructors()->get();

        // dd($instructors);

        // dd($course);
        return view('course_detail')
            ->with(compact('course'))
            ->with(compact('instructors'));
    }
}
```

## Appendix 5.

Logic codes for the instructor page.

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\Instructor;
use App\Models\User;

class InstructorController extends Controller
{
    //

    function __construct(){
        $this->middleware('auth');
    }

    function index(){

        $instructors = Instructor::paginate(8);

        $instructors = User::where('role', '2')->paginate(8);

        return view('instructors', compact('instructors'));
    }

    function detail($id){

        $instructor = User::find($id);

        $courses = $instructor->courses()->paginate(4);

        // dd($courses);

        return view('instructor_detail')
            ->with(compact('instructor'))
            ->with(compact('courses'));
    }
}
```

```
function filter(Request $request){  
  
    if($request->institution == "blank"){  
        $instructors = User::where('role', '2')->paginate(8);  
        return view('instructors', compact('instructors'));  
    }  
  
    $instructors = User::where('role', '2')->where('institution', $request->institution)->paginate(8);  
    // dd($request->institution);  
  
    return view('instructors', compact('instructors'));  
}  
}
```

## Appendix 6.

Logic codes for study requests.

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use Illuminate\Support\Facades\Auth;

use App\Models\Request as RequestModel;
use App\Models\Studymate;

class RequestController extends Controller
{
    //
    function __construct()
    {
        $this->middleware('auth');
    }

    function index(){

    }

    function new_request(Request $request){

        // dd([$request, auth()->user()]);
        RequestModel::create([
            'user_id' => $request->user_id,
            'from_user_id' => auth()->user()->id,
            'status' => 'Pending'
        ]);

        return redirect()->route('find_mates');
    }

    function respond(Request $request){
        if($request->response == 'Accepted'){
            // RequestModel::find($request->request_id)->up
            RequestModel::where('id', $request->request_id)->update([
                'status' => 'Accepted'
            ]);
        }
    }
}
```

```
Studymate::create([
    'user_id' => auth()->user()->id,
    'mate_user_id' => $request->request_user_id
]);
Studymate::create([
    'user_id' => $request->request_user_id,
    'mate_user_id' => Auth::id()
]);
}elseif($request->response == 'Rejected'){
    RequestModel::where('id', $request->request_id)->update([
        'status' => 'Rejected'
    ]);
}
return redirect()->route('user_request');
```

## Appendix 7

Logic codes for the user of the system.

```
<?php

namespace App\Http\Controllers;
use Illuminate\Support\Facades\Auth;
use Illuminate\Support\Facades\Storage;
use Illuminate\Support\Facades\Validator;
use Image;

use Illuminate\Http\Request;

use App\Models\User;
use App\Models\Instructor;
use App\Models\Book;
use App\Models\Course;
use App\Models\CourseInstructor;
use App\Models\Request as RequestModel;

class UserController extends Controller
{
    public function __construct(){
        // $this->middleware(['auth']);

        $this->middleware('not admin');
    }

    //

    function home(){

        if(Auth::user()->role == '99'){
            return redirect()->route('admin_dashboard');
        }

        $students = User::where('role', '0')->where('blocked', null)->count();
        $instructors = User::where('role', '2')->count();
        // $instructors = Instructor::count();
        $books = Book::count();
        $courses = Course::count();
    }
}
```

```

        return view('home')
        ->with(compact('students'))
        ->with(compact('instructors'))
        ->with(compact('books'))
        ->with(compact('courses'));
    }

    function index(){
        // get data for number of studymates the user has
        // auth()->user()->mates()->count();
        $mates = auth()->user()->mates()->count();
        $courses = 0;
        if(auth()->user()->role == '2'){
            $courses = auth()->user()->courses()->count();

        }

        return view('user_dashboard')
        ->with(compact('mates'))
        ->with(compact('courses'));
    }

    function edit(){
        return view('user_edit');
    }

    function edit_user(Request $request){

        $user = User::find(Auth::id());

        // $image_name = Auth::id().'.'.time().'.'.$request->user_pic->extension();
        // $request->user_pic->move(public_path('users', $image_name));

        // dd($request->file('user_pic_d'));

        // this storage works
        // if($request->file('user_pic_d')){
        //     // dd($request->file('user_pic_d'));
        //     $file= $request->file('user_pic_d');
        //     $filename= date('YmdHi').$file->getClientOriginalName();
        //     // dd($filename);
        //     // $file->move(public_path('public/Image/test'), $filename);
        //     $file->storeAs('public/Image/test', $filename);
        //
    }

```

```

//    dd('done');
//    $data['image'] = $filename;
// }

if($request->hasFile('user_pic')){

    $avatar = $request->file('user_pic');
    $filename = time() . '-' . Auth::id() . '.' . $avatar->getClientOriginalExtension();
    // dd(public_path('\\uploads\\avatar'));

    $path = $request->file('user_pic')->storeAs('public/Image/test', $filename);

    Image::make('C:/Users/USER/Desktop/_project/_Project/studymate/storage/app/public/
Image/test/' . $filename)->resize(300, 300)->save();

    $user = User::find(Auth::id());
    $user->pic = $filename;
    $user->save();
}

// if($request->hasFile('user_pic_d')){

    // $avatar = $request->file('user_pic_d');
    // $filename = time() . '-' . $avatar->getClientOriginalExtension();
    // Image::make($avatar)->resize(300, 300)->save('C:/Users/USER/Desktop/_project/_Project/studymate/storage/app/public/Image/test/nice
' . $filename);
    // Image::make($avatar)->resize(300, 300)->save( public_path('\\uploads\\avatar\\' . $filename));

    // $user = User::find(Auth::id());
    // $user->pic = $filename;
    // $user->save();
}

$user->fill([
    'fname' => $request->fname,
    'lname' => $request->lname,
    'm_name' => $request->m_name,
    'email' => $request->email,
]

```

```
// 'pic' => $image_name,
'institution' => $request->institution,
'study_field' => $request->study_field,
// 'year' => $request->year,
// 'email' => $request->email

]);


// Save user to database
$user->save();

return redirect()->route('home');

}

function edit_user_image(Request $request){

    // User::creat([
    //     'email' => $request->email,
    //     'password' => Hash::$request->password,
    // ]);

    dd('thsi is fu');

    $user = User::find(Auth::id());

    $image_name = Auth::id().'_'.time().'.'.$request->user_pic->extension();
    $request->user_pic->move(public_path('users', $image_name));

    $user->fill([
        'pic' => $image_name,
    ]);

    // Save user to database
    $user->save();

    return redirect()->route('home');

}
```

```

function get_details(){
    return view('user_details');

}

function update_details(Request $request){

    $userId = Auth::id();
    $user = User::findOrFail($userId);
    $filename = null;

    // this storage works
    // if($request->file('user_pic')){
    //     dd($request->file('user_pic'));
    //     $file= $request->file('user_pic');
    //     $filename= date("YmdHi").$file->getClientOriginalName();
    //     dd($filename);
    //     $file->move(public_path('public/Image/test'), $filename);
    //     $file->store('public/Image/test');

    //     dd('done');
    //     $data['image']= $filename;
    // }

    if($request->hasFile('user_pic')){

        $avatar = $request->file('user_pic');
        $filename = time() . '-' . Auth::id() . '.' . $avatar->getClientOriginalExtension();
        // dd(public_path('\\uploads\\avatar'));

        $path = $request->file('user_pic')->storeAs('public/Image/test', $filename);
        // dd($path);
        // dd(storage_path($path));
        Image::make('C:/Users/USER/Desktop/_project/_Project/studymate/storage/app/public/
Image/test/' . $filename)->resize(300, 300)->save();

        // Image::make($avatar)->resize(300, 300)->store('public/Image/test');
        // Image::make($avatar)->resize(300, 300)-
>save('C:/Users/USER/Desktop/_project/_Project/studymate/storage/app/public/Image/test/nice
' . $filename);
    }
}

```

```

        // Image::make($avatar)->resize(300, 300)->save( public_path("\uploads\avatar\" . $filename));

        $user = User::find(Auth::id());
        $user->pic = $filename;
        $user->save();
    }

    // dd('end');

    // Fill user model
    $user->fill([
        'fname' => $request->fname,
        'lname' => $request->lname,
        'm_name' => $request->m_name,
        'institution' => $request->institution,
        'year' => $request->year,
        // 'pic' => $filename,
        'study_field' => $request->studyfield,
        // 'email' => $request->email
    ]);

    // Save user to database
    $user->save();

    return redirect()->route('home');
}

function bookmark(){
    return view('user_bookmark');
}
function setting(){
    return view('user_setting');
}

function delete(){
    return view('user_delete');
}

function details(){
    return view('user_details');
}

```

```

function sign_out(Request $request){
    Auth::logout();

    return redirect()->route('sign_in');
}

function find_mates(){

    // $users_with_request = RequestModel::all()->where('from_user_id', Auth::id());
    //these are the users that the current user has already send request to
    $already_requested = User::find(auth()->user()->id)->request_from()->where('status', 'Pending')->get();
    $already_accepted = User::find(auth()->user()->id)->request_from()->where('status', 'Accepted')->get();
    $already_rejected = User::find(auth()->user()->id)->request_from()->where('status', 'Rejected')->get();

    // these are the users that have sent request to the current user
    $has_requested = User::find(auth()->user()->id)->request()->where('status', 'Pending')->get();
    $accepted_request = User::find(auth()->user()->id)->request()->where('status', 'Accepted')->get();

    $users = User::whereNotIn('id', [Auth::id()])->where('role', '0')->paginate(8);

    return view('find_mates ')
        ->with(compact('already_requested'))
        ->with(compact('already_accepted'))
        ->with(compact('already_rejected'))
        ->with(compact('has_requested'))
        ->with(compact('accepted_request'))
        ->with(compact('users'));
}

function requests(){
    Requests = RequestModel::where('status', 'Pending')->where('user_id', auth()->user()->id)->paginate(6);

    return view('user_requests', compact('requests'));
}

function filter_mates(Request $request){

    if($request->institution == "blank"){

```

```

$users = User::where('role', '0')->paginate(8);
// return view('find_mates', compact('users'));
return redirect()->route('find_mates');
}

$already_requested = User::find(auth()->user()->id)->request_from()->where('status',
'Pending')->get();
$already_accepted = User::find(auth()->user()->id)->request_from()->where('status',
'Accepted')->get();
$already_rejected = User::find(auth()->user()->id)->request_from()->where('status',
'Rejected')->get();

// these are the users that have sent request to the current user
$has_requested = User::find(auth()->user()->id)->request()->where('status', 'Pending')-
>get();
$accepted_request = User::find(auth()->user()->id)->request()->where('status', 'Accepted')-
>get();

$users = User::whereNotIn('id', [Auth::id()])->where('role', '0')->where('institution', $request-
>institution)->paginate(8);

return view('find_mates')
->with(compact('already_requested'))
->with(compact('already_accepted'))
->with(compact('already_rejected'))
->with(compact('has_requested'))
->with(compact('accepted_request'))
->with(compact('users'));

}

function courses(){

if(auth()->user()->role != '2'){
    return redirect('/home');
}

// dd(auth()->user()->courses()->get());

$courses = auth()->user()->courses()->get();

```

```

        return view('user_courses', compact('courses'));
    }

function show_add_course(){

$lec_courses = auth()->user()->courses()->get();
// dd($lec_courses);
$courses = Course::all()->diff($lec_courses);

// dd($courses);

$courses = Course::paginate(8);

// dd($courses - $lec_courses);

}

return view('user_add_course')
->with(compact('lec_courses'))
->with(compact('courses'));
}

function add_course(Request $request){

// dd($request->keys());

foreach($request->keys() as $course_key){
    // dd($course_key->keys());
    if($course_key != '_token'){
        //add this course id to course_instruc with user id as user_id
        // dd($course_key);
        CourseInstructor::create([
            'course_id' => $course_key,
            'user_id' => auth()->id()
        ]);
    }
}

// $this->validate($request, [
//     'course' => 'required',
// ]);

}

```

```
        return redirect('/user-courses');
    }

function delete_account(Request $request){
    //block user
    $user = User::find(Auth::id());

    $user = $user->fill([
        'blocked' => true
    ]);

    $user->save();

    // sign the user out

    Auth::logout();

    return redirect()->route('sign_in');
}

function remove_course(Request $request){

    $request->validate([
        'course_id' => 'required'
    ]);
    // remove the entry from course_instru where user_id is user_id and course_id
    CourseInstructor::where('user_id', auth()->id())->where('course_id', $request->course_id)->delete();

    return redirect('/user-courses');
}
```