ADDIS ABABA UNIVERSITY

COLLEGE OF NATURAL AND COMPUTATIONAL SCIENCE

SCHOOL OF INFORMATION SCIENCE

Askuala Teaching Learning Support System

Prepared by:

No.	Name	ID
1.	Selamawit Lemma	UGR/0426/12
2.	Mintamir Agegnehu	UGR/5712/12
3.	Soliyana Yalewdeg	UGR/2202/12
4.	Tsion Asdegdid	UGR/3653/12
5.	Sofonias Miftah	UGR/3809/12
6.	Temesgen G/abzgi	UGR/4330/12

Submitted to: School of Information Science

Examination Board

Advisor _____

Name: _____

Signature: _____

Date:

Examiner _____

Name: _____

Signature:

Date: _____

Examiner _____

Name: _____

Signature:

ACKNOLEDGMENT

We would like to express our special thanks of gratitude to our adviser Ms. Lemlem Hagos, who has invested her time and effort in guiding the team in achieving our goal. Throughout thecomposition of the paper, we are grateful for her unwavering patience, unwavering encouragement, and unreserved and important advice. With acute intellect and understanding,she provided us with assistance for all of the obstacles we had, from creating the study proposalthrough the completion of this phase of the project.

Furthermore, we would also like to acknowledge and appreciate the crucial role of the staff of Addis Ababa university staff who gave us the permission to get all the information required forthe completion of this project.

Finally, and most importantly, we want to thank our families and friends for their unfailing support and encouragement throughout our academic careers. Finally, we'd want to thank everyone who helped us complete this project on schedule, whether directly or indirectly.

Table of Contents

Acronyms

AAU – Addis Ababa

UniversityCSS –

Cascading Style Sheet

HTML – Hyper Text Markup Language
ICT – Information Communication

TechnologyID – Identification

IT – Information

TechnologyJS – Java

script

MVC – Model View Controller

MySQL – My Structured Query

Language NoSQL – Not only Structured

Query Language

PIECES - Performance, Information and Data, Economics, Control and Security,

Efficiency,and service

PHP – Hypertext Processor

SDLC – System Development

Life CycleSQL – Structured

Query Language

UI – User Interface

UML – Unified Modeling

LanguageUX – User

Experience

Sysadmin- system administrator

CHAPTER ONE

1. Introduction

A teaching learning support system (TLSS) is a computer-based learning and teaching system that provides an effective way to supplement the educational instruction provided in the classroom. It helps to provide additional help and resources to learners in the form of simulations, digital books, videos and other digital content. TLSS also focuses on motivating and engaging learners, as well as assessing the understanding and progress of learners. A Teaching-Learning Support System (TLSS) refers to a set of tools and resources that are designed to support the teaching and learning process.

Teaching and learning support systems are designed to enhance and support the teaching and learning process. The requirements for a teaching and learning support system can vary depending on the specific needs of the educational institution and the learners, but some common requirements include accessibility, user-friendly interface, multimodal delivery, content management, interactivity, assessment and feedback, security and privacy, compatibility, technical support. By meeting these requirements, teaching and learning support systems can provide a more effective and engaging learning experience for students, and better support for instructors in their teaching efforts.

Using technology in the teaching learning provides students with access to countless online resources, encouraging them to carry out research and therefore become more independent. It also simplifies learning by making concepts more digestible.

The name “Askuala” indicates that old incorrect calling of modern education or school. We use this title for the purpose of making the system familiar to the user.

1.1. Overview

In Our proposed System the system will enable students to get relevant academic books, recorded lecture videos, tutorial videos, and collection of previous exams alongside with their answers, Student forum Question Answering platform, where students

interact with other students through questioning and getting a response to their questions.

1.2. Background of the organization

Modern higher education in Ethiopia began with the founding of the Addis Ababa University on March 20, 1950. When formal lectures started in the university on December 11, 1950, theFaculty of Science, one of the only two Faculties then, had only two departments or sections, known as Section A and Section B. In Section A, students were given basic training in Engineering, which would enable them to go abroad to specialize in one of the many branchesof Engineering, whereas those in Section B were prepared for Medical School as well as for further studies in Biology and allied fields.

When the Haile Selassie I University was established in 1961, the Faculty of Science was reorganized into five teaching departments, all offering B.Sc. degree programs. These were the departments of: Biology, Chemistry, Geology, Mathematics, and Physics. A Forestry Department, the Natural History Museum and the National Her barium were established a littlelater, while a statistical training center was opened in the department of Mathematics, center that developed into a full-fledged department of Statistics in the early 1970s.

College of Natural and Computational Sciences is one of the natural science colleges from Addis Ababa University. The College of Natural and Computational Sciences in its more than50 years of establishment has gone through various stages of development. Currently, the College comprises eight departments, two schools, two institutes, and three multidisciplinary programs offering undergraduate and postgraduate degrees. (cns, 2023)

1.1. Statement of the problem

Although traditional chalk and board educational system is widely used all over the world, acting like the main teaching and learning system, it is faces several challenges that makes it unfortunate to provide a successful knowledge exchange between teachers and students. Problems like a limitation to access educational resources due to geographical restrictions, lack of technology and high cost of learning equipment, material may not up to date with the latest development and technologies in the industry. Student learns differently, some prefer to learn slowly and explore different learning methods, while others use huge resource and references to understand their syllabubs, helping all of them fully retain information is our priority. In traditional chalk and board educational system, students must adapt to the pace of the class or be left behind. Students learn better andfeel more comfortable learning in an environment of their preference. This is not always possible in a classroom with a physical space restriction. After the classroom teaching process, student need support in terms of accessing learning materials like reference books, exam questions and discussion with their instructors without time and place restriction.

1.3. Objectives of the Project

Project objectives state the aim of a project. Some projects have only one project objective, while others own several objectives that could be completed at various times during the project.Project objectives are a practical tool. The objectives of a project are often written up into an objective statement. A project's objective statement should contain all the objectives for the project.

The following are the general and specific objectives of Askuala teaching learning support system.

1.3.1. General objective

To develop a web based teaching learning support system that used by students and teachers to help in teaching and learning system of Addis Ababa University in College of Natural and Computational Science.

1.3.2. Specific objectives

To sufficiently meet the general objective of the project, the following specific objectives havebeen identified and must be addressed: -

- Depth understanding of the problem.

- Formulate a project plan which outlines the activities to be carried out in the development process of the main deliverable of the project
- Conduct required feasibility studies;
- Conduct interviews with the relevant users of the system and college deans;
- Formulate requirements for the system to be built according to the information gatheredanalyzed throughout the information gathering process;
- Analyze requirements gathered and model the respective diagrams using chosen tools;
- Create intermediate prototypes;
- Design system modules, functionality and features;
- Implement the system designed;
- Test the system implemented.

1.4. Feasibility Study

A feasibility study is an assessment of the practicality of a proposed plan or project. A feasibility study analyzes the viability of a project to determine whether the project or venture is likely to succeed. Feasibility studies can also provide a company's management team with crucial information that could prevent them from entering into a risky business venture.

We assessed 4 feasibility studies which can describe about how the proposed project planned to be executed and weather it is technically, operationally, economically and legally feasible.

Feasibility study A study that determines if the proposed information system makes sense for the organization from an economic and operational standpoint. (F.George, 2021)

1.4.1. Technical Feasibility

The purpose of assessing technical feasibility is to gain an understanding of the organization's ability

to construct the proposed system. This analysis should include an assessment of the development group's understanding of the possible target hardware, software, and operating environments to be used, as well as system size, complexity, and the group's experience with similar systems. (F.George, 2021)

- Is it possible to develop the product with the available technology in the company?
- Is the organization equipped with the necessary technology for project completion?
- Is it possible to develop the product with the available technology in the company?
- Is the organization equipped with the necessary technology for project completion?
- Are there technically strong employees who can deliver the product on time and withinbudget using the available technology?
- Is there scope in the company's budget to add more technical resources?
- Is the available technology the right choice to help the product team save time andcomplete development within budget?

When we assess the technical feasibility of our system, we consider what technologies we used to implement the work, and the technical skills the project team has. For Developing the system we use the following technologies:-

- ✓ HTML (Hypertext Mark-up Language)
- ✓ CSS (Cascading Style Sheets) and Bootstrap
- ✓ JavaScript
- ✓ React JS
- ✓ Node JS
- ✓ Mongo DB
- ✓ Microsoft Word
- ✓ Microsoft Visio

- ✓ Adobe XD for building UI design
- ✓ Visual studio
- ✓ UML modeling tools :- lucid chart

Since the development being carried out using freely available technologies and the technical skills required are manageable, time limitations of the platform development and the ease of implementing using these technologies are synchronized. For the technical skills of the project team, since we are familiar with the mentioned development tools and softwares we can conclude that the project team is capable for building the proposed system. So, the proposed project is technically feasible.

1.4.2. Operational Feasibility

Operational feasibility relates to examining the likelihood that the project will attain its desired objectives. Its purpose is to gain an understanding of the degree to which the proposed system will likely solve the business problems (F.George J. S., 2021) The desired goal of our system is to support the teaching learning process by integrating services delivered to students different platform in to one i.e. like assignments are given to representative and then the representative share it ,courses are given sometimes by email, sometimes by telegram or sometimes by Google classroom for such problems we develop a website that enable teacher to create a class and meet their student through it and also the teacher can create a virtual class to conduct tutorial class.

Therefore, the team assesses the operational feasibility of the project and manages the systemdevelopment activities in the system life cycle in order to satisfy the operational feasibility of the project.

The proposed system is acceptable to users. Our project is operationally feasible as the new system:

- It satisfies users need or requirements
- Provides users with timely, accurate, reliable, flexible, and usefully formatted information.
- It facilitate the teaching learning process by making it interactive two way communication.

1.4.3. Economic Feasibility

The purpose of assessing economic feasibility is to identify the financial benefits and costs associated with the development project (Laplante, 2006,1994). Economic feasibility is often referred to as cost–benefit analysis. (F.George J. S., 2021) Here we describe typical benefits and costs resulting from the development of an information system and provide several useful worksheets for recording costs and benefits. In order

to correctly calculate the economic feasibility of the project, the team can consider different issues that have a direct impact on the cost of the proposed system (like: hardware and software cost) and running (operational) cost to use the proposed system.

Benefit

An information system can provide many benefits to an organization. For example, a new or renovated information system can automate monotonous jobs and reduce errors; provide innovative services to customers and suppliers; and improve organizational efficiency, speed, flexibility, and morale. In general, the benefits can be viewed as being both tangible and intangible. (F.George J. S., 2021)

1.5.3.1.Benefits

An information system can provide many benefits to an organization. For example, a new or renovated information system can automate monotonous jobs and reduce errors; provide innovative services to customers and suppliers; and improve organizational efficiency, speed, flexibility, and morale. In general, the benefits can be viewed as being both tangible and intangible. (F.George J. S., 2021)

I. **Tangible benefit**

Refers to items that can be measured in dollars and with certainty. Examples of tangible benefits might include reduced personnel expenses, lower transaction costs, or higher profit margins. (F.George J. S., 2021)

Also statistically the estimated tangible benefits that will be resulted from our project succession are stated on the table below.

Benefits	Estimated benefit cost in ETB
Cost of course material	30,000 ETB per student
If students don't have enough memory space in their device they can use online this saves the cost of memory	20,000 ETB
The system save cost of Printing course materials	20,000ETB per student
The system save the cost of transportation	4,000ETB annually
Total	74,000

II. Intangible benefits

Refer to items that cannot be easily measured in dollars or with certainty. Intangible benefits may have direct organizational benefits, such as the improvement of employee morale, or they may have broader societal implications, such as the reduction of waste creation or resource consumption. (F.George J. S., 2021)

One of the most significant benefits of the proposed system is that it gives the convenience oflearning which is independent of time and place constraints. It is hugely time-saving because the website provides all necessary materials in different format put in one place, so the time for searching in different website is saved and the quality of learning is very precise. The proposed system also support both in audio- and video. This mode enables interesting contentto be created in the system. In this way, the intended users can understand the content with clarity, and effectively as well.

1.5.3.2. Costs

Project costs are the funds required to perform a planned business endeavor, and they are a primary subject in project budgeting and cost management. Costs are the entities you

estimatewhen developing a budget. They are the money you actually invest in work and the amounts you track and control until the very end of a project.

In the following table we estimated the cost our system would require on completion. Since most of the software systems we used on the project are open source, we didn't thought it wouldnecessary to include software items cost. The main cost of our project will be hardware devicesrequired to build the system and recurring costs like web hosting. The one time hardware costsare disclosed in the following table.

No	Hardware Item	Unit Cost	Quantity	Estimated cost (ETB)
1	Server	5,000	1	5,000
2	PC	45,000	1	45,000
3	Hard drive	2,000	500GB	2000
4	System maintenance	10,000		10,00
5	System administration	10,000		10,000
Total				72,500

Table 1.2 Estimate cost

Therefore the total estimate cost of the project is 72,500 and as we disclosed the tangiblebenefits on Table 1.1 our system's estimated tangible benefits are equal to 74,000.

Net Cost Benefit = Total Benefit -

Total CostNet= 74,000 – 72,500

Net= 1500 ETB per year

Since we disclosed our net cost benefit from tangible benefits and total estimate costs, and itresult with greater benefit, we can conclude that, the project is economically feasible.

1.4.4. Legal feasibility

legal and contractual feasibility requires that you gain an understanding of any potential legal and contractual ramifications due to the construction of the system. Considerations might include copyright

or non disclosure infringements, labor laws, antitrust legislation (which might limit the creation of systems to share data with other organizations), foreign trade regulations (e.g., some countries limit access to employee data by foreign corporations), and financial reporting standards as well as current or pending contractual obligations. (F.George, 2021)

1.5. Significance of the Project

The main significance of this Project, the teaching learning support system includes:

- Improve education service of the college by provide materials to student.
- The website used to upload a new updated materials to students so, students canimprove their skill in new updated world.
- Student can interact with each other through dialogue platform and improve theirsocial skill in the university
- Students can ask and answer different questions and share knowledge with each other.
- The system addressed the 4 department of College of Natural and Computational Science, so it will be easy to student to find all material in one website than searchingall over the internet.
- The most important aspect of this system is to help make the traditional learningsystem more efficient and effective.

1.6. Beneficiaries of the project

This system benefits College of Natural and Computational Science, student, teachers and alsothe system developers. The College benefit from the system as it increases variety education delivery options. The main beneficiaries of the system are students because they have access to materialrelated to their field at a time without searching all over the internet and students have better opportunities to collaborate with classmates. Teachers can upload course materials to their students at any time and they can also interact with their student through the virtual class they create.

The system developers will have the opportunity to exercise how a real-life problem can be solved. Additionally, the developers shall acquire the skill to work in an industry-paced environment, job satisfaction and experience will be gained.

1.7. Methodology

Methodologies are comprehensive, multiple-step approaches to systems

development that will guide your work and influence the quality of your final product—the information system.

Systems development methodology is a standard process followed in an organization to conduct all the steps necessary to analyze, design, implement, and maintain information systems. (F.George, 2021)

1.7.1. Data collection

Data collection is the process of gathering, measuring, and analyzing accurate data from a variety of relevant sources to find answers to research problems, answer questions, evaluateoutcomes, and forecast trends and probabilities. We collect data from different source, the methods we used:-

- **Observation**

We went to observe how the teaching learning system goes in College of natural and computational science. As students we observe what need to add in class.

- **Interviewing**

We went to CNCS in 4 kilo to meet and interview the dean of College of Natural and Computational Science and ask some questions. We also went to computer science department dean.

1.7.2. System Development Methodology

There are many kinds of process models for meeting different requirements. For our system, we are planning to use system development life cycle (SDLC) models. The systems development life cycle (SDLC) is a common methodology for systems development in many organizations that marks the phases or steps of information systems development. The " iterative Waterfall Model" was chosen by the proposed system to be used in its development. In the Iterative waterfall model, iterative process starts with a simple implementation of a small set of the software requirements and iteratively enhances the evolving versions until the complete system is implemented and ready to be deployed. Development begins by specifying and implementing just part of the software, which is then reviewed to identify further requirements. This process is then repeated, producing a new version of the software at the end of the iteration of the model. (MAL, 2023)

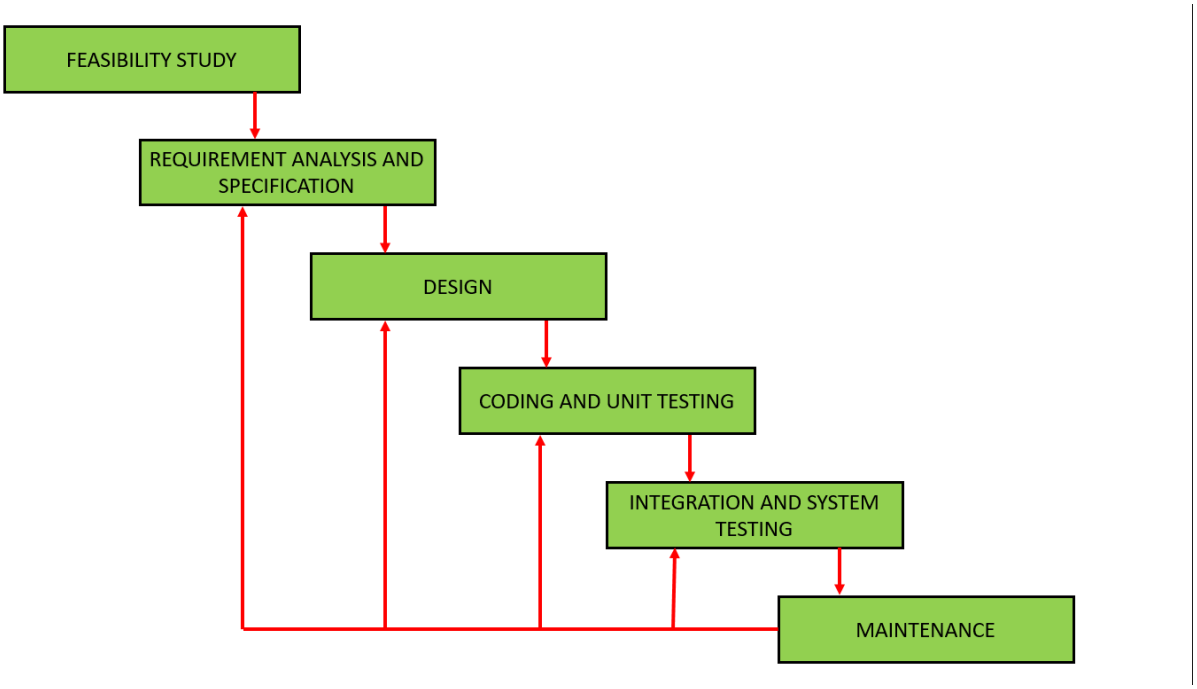


Figure 1.1 iterative waterfall model

1.8.2.1 Process model

A process model is a visual depiction of the flow of work and tasks for specific goals. Often, process models take graphical forms, and they typically depict work flow that companies complete repeatedly. They usually include different activities related to a goal and potential results based on decisions made by the company. Process modeling can make it easier to analyze and understand work flow by dividing the process into smaller and manageable steps.

1.8.2.2 Development approach

System development approach is a methodology for systematically organizing the best ways to develop systems efficiently.

The methodology that we will use all over this project is object oriented approach. It is a new system development approach, encouraging and facilitating re-use of software components. There are three types of Object Oriented Methodologies,

I. Object Modeling Techniques (OMT)

It was one of the first object oriented methodologies and was introduced by Rumbaugh in1991.OMT uses three different models that are combined in a way that is analogous to theolder structured methodologies.

OMT Models can be distinguished as:

a. Object Model : It depicts the object classes and their relationships as a class diagram, which represents thestatic structure of the system. It observes all the objects as static and does not pay any attention to their dynamic nature.

b. Dynamic Model : It captures the behavior of the system over time and the flow control and events in the Event-Trace Diagrams and State Transition Diagrams. It portrays the changes occurring in the states of various objects with the events that might occur in the system.

c. Functional Model :It describes the data transformations of the system. It describes the flow of data and thechanges that occur to the data throughout the system.

II. Object Process Methodology (OPM)

- Analysis in OMT: The main goal of the analysis is to build models of the

world. The requirements of the users, developers and managers provide the information needed todevelop the initial problem statement.

- Design in OMT: It specifies all of the details needed to describe how the system will beimplemented. In this phase, the details of the system analysis and system design are implemented.
- Therefore, object oriented system development approach is suitable for our proposed system because it allows classifying the system in to objects and showing their relations in diagrams such as class diagram,sequence diagram and use cases.

III. Rational Unified Process (RUP)

The rational unified process (RUP) is a software engineering and development process focused on using the unified modeling language (UML) to design and build software. Using the RUP process allows you to operate business analysis, design, testing and implementationthroughout the software development process and its unique stages, helping you create a customized product. You can use beta test models and prototypes of various software components in all phases of RUP to:

- Better achieve milestones
- Calibrate elements of design
- Troubleshoot concerns
- Present the best possible software solutions

From the above methodologies we are going to use object modeling techniques**.**

1.8. Development Tools and Techniques

Software development tool is a program used to create, maintain, test, debug, build and support other applications and software. There are various types of tools like languages, monitoring platforms, databases, frameworks, etc. that programmers use to create software.

1.9.1 Front-end Technologies

Front-end development manages everything that users visually see first in their browser or application. Front-end developers are responsible for the look and feel of the site. (Stewart 2020). The front-end development focuses on the client-side of the development, meaning it focuses on what the users visually see first in their browser or application and interact with them. Front end languages that we use in our project are HTML, CSS, and JavaScript.

- HTML (Hypertext Markup Language)
- CSS (Cascading Style Sheets) and Bootstrap
- JavaScript
- React JS: React is a JavaScript library for building user interfaces. It used to build single-

page applications and it allows us to create reusable UI components. React, sometimes referred to as a frontend JavaScript framework, and is a JavaScript library created by Facebook.

1.9.2 Back-end technologies

Back-end technologies refer to the libraries of server-side languages that are used to create theserver configuration of a website.) Back-end technology that we use during implementation is Node.js for programming and MongoDB for database storage.

- Node JS

  Node.js (Node) is an open source, cross-platform runtime environment for executing JavaScript code. Node is used extensively for server-side programming, making it possiblefor developers to use JavaScript for client-side and server-side code without needing to learn an additional language.

- **MongoDB**

  MongoDB is a document-oriented NoSQL database used for high volume data storage. Instead of using tables and rows as in the traditional relational databases, MongoDB makesuse of collections and documents. Documents consist of key-value pairs which are the basic unit of data in MongoDB. Collections contain sets of documents and function whichis the equivalent of relational database tables.

1.8.1. Documentation and Modeling tools

Documentation tools streamline the process of creating and managing documents by makingwriting or distributing documentation faster and easier.

❖ **Modeling tool**

For modeling tool we use the UML modeling language. UML (universal modeling language) is a standard language for specifying, visualizing, Constructing, and documenting the artifacts of software-intensive systems, as well as forBusiness modeling and other non-software systems" (Booch et al. 1999). (Roger Y. Lee,2019).In this project we use lucid chart that is one of UML

diagramming tool.

There are lists of documentation and modeling tool we used in this paper:

- Microsoft Word
- Microsoft Visio
- Lucid chart for building Project schedule
- Adobe XD for building UI design: is a vector-based UI (User Interface) and UX (UserExperience) design tool and it can be used to design anything from smart watch apps to fully fledged websites. Our project will use this to design the prototype UI for moreupscale human computer interaction

1.8.2. Deployment Environment

The development environment that we use to our project is visual studio for back end and front-end development .As we use JavaScript to both back-end and front-end development. We develop JavaScript on visual studio code. For database development we use Mongo DB CLI. It is easy to install and use it also free available from internet.

1.9. Scope of the project

In this project, we have proposed to develop a system which support the teaching learning process for College Of Natural and Computational science, Addis Ababa University, and enable users to access uploaded course materials, exam questions and reference books.Students can ask question and get answer using this platform. Also instructors can held live virtual classes, give an assignment and publish grades for assignments given.

1.10. Risks, Assumptions and Constraints

Risks, Assumptions and Constraints are elements that must be taken into account in a robust project plan. These factors may or may not happen, but a good project plan will be include accommodations for the most important of these, so that if they do in fact

occur their impact on the projects schedule and budget will as minimal as possible.

1.10.1. Risk

Risk is defined as the potential effect of an event, determined by the likelihood of the event occurring with the effect it should occur. Risks are factors that we are aware of but whose occurrence is uncertain. A project risk is an uncertain event that may or may not occur duringa project. It could have either a negative or a positive effect on progress toward project objectivity.

Risk type	Assessment	Mitigation
User acceptance risk	We might encounter lack of willingness of users to use the system	We can resolve this by advocating the users
Technological risk	Technological risk that we may face is choosing the right technology, as it could make or break our system. using inappropriate software or programming language can increase costs and reduce productivity	By avoiding using outdated technologies, by identifying our primary needs before considering available options for developing the system.
Performance risk	These are risks when the project does not perform as initially expected	By anticipating potential performance risk early on in the planning process
Lack of clarity risk	These types of risks may come in the form of miss communication from stakeholder, vague project scope or unclear deadline.	We can resolve this by check and recheck our requirements to ensure everything is in place during the planning phase.

Cost risk	We might encounter this type of risk when our project goes over the budget we initially set.	By estimating each element of our project accurately and stick closely to our project.

Table 1.3 Project Risk

1.10.2. Assumption

Assumption is any factor relating to a project that is considered to be true, real or certain- without any empirical proof or demonstration. (Project-assumptions, 2022)

Assumptions can be thought as things that we believe to be true and which we therefore build into the project plan.

The assumptions in this project are:-

- The users have access to a smart phone or a computer with a functioning camera.
- The user knows how to operate a smart phone or computer.
- The user has access to the internet.

1.10.3. Constraints

Constraints are things that we know to be true and which must be accounted for in the plan sothat we can work around them. Constraints are limiting factors of our project that can impact quality delivery and overall project success (project-constraints, 2022)

- Lack of a Virtual Learning Culture.
- Digital Technology Troubles.
- Maintaining a Budget.
- The application needs internet to work.

1.11. Phases and deliverable of the project

For different projects people might use different phases to follow system development, we usedthe following 6 phases for building effective, indispensable teaching learning support system.

The phases that we will undergo through developing this project are:

- Planning: proposal
- Requirement gathering: requirements specification list
- System analysis: system analysis model of the system
- System Design: system design model specification
- Coding and Implementation: web based local freelancer system
- Testing: interactive web-based user service
- Deployment

1.11.1. Planning and selection phase

Problem identification is the first step in system planning phase, a need for the “Teaching learning support system", and also plan how to develop the functional requirements of a system.

This phase can be thought to include another sub-phase called specification phase. The specification phase is an important step that helps us focus on creating learning experiences that are tailored to our specific learner. The specification phase defines what the solution will look like and lists the quality assurance acceptance criteria against which the teaching learning support system will later be tested.

- **Consistency** starts with creating a set of principles by which our team is going to abide. We don’t want to lock it down so much that we eliminate creativity, but we do need to provide support structures that enable team members to understand what it is we’re tryingto achieve. Most learning professionals will create a guide as to what tools, style, and branding to use.

- **Functionality** helps us lock down exactly how our course will function. There are sevenkey areas to consider: platform and browser, reporting, media, navigation, accessibility, user interface/creative direction, and acceptance

criteria.

1.11.2. Analysis phase

The analysis phase is all about setting up the project for success. Next we define a scope, understand the design challenge that we are about to address. Several factors can affect scope like: Budget, time, resources, and requirements. We distinguish objectives, experiences,outcomes, impacts, goals, measures and needs analysis.

In this step, we will assess the state of the campus's technical support infrastructure and existing office property inquiries. At this point, additional sources of information regarding the new requirements and the current working habits would also be looked into.

Deliverables of analysis phase are Specification document, which consist of

- System use case modeling
- Class modeling
- Sequence diagram
- Activity diagram
- User interface prototyping

1.11.3. Design phase

The design phase involves bench marking, prototyping, and testing. During this phase, we testyour ideas before we build them. In design phase the look and feel of the project is implemented which includes themes and contents. Also, the proofs of concepts are implemented here, whichare;

- **Bench marking** defines where the project is headed and it is a key to make sure the projecttargets the learner and is in line with our business values. It can also provide with guidelines to determine how we deliver future E-learning courses.
- **Prototyping** involves testing concepts quickly so we can discard what is not viable in yourcontext. The point is to get a visualization of potential solutions without actually making something that we'll find difficult to throw out later.
- **User testing** during the design phase is specifically about testing our design against the end user. Think who our course is aimed at and how they will use it to

solve a problem orfill a gap in their skills or knowledge.

1.11.4. Production phase

The production phase is the point at which our planning and design come together. In thisphase, we map out content, create screens and templates, and involve graphic designers.

- Map out the overall experience using flowchart.
- Story boarding is the method of orchestrating all the elements that will make up the E-learning to create a score. A storyboard explains how all the elements fit together. We included 10 storyboard elements which are: text, graphics, animation, video, audio, resources, links, references, interactions, and activities.
- Create screens and templates from the list of the interaction.

1.11.5. Implementation phase

The actions that start as soon as the system design is finished are referred to as the implementation phase. These stages produce software code in accordance with the system design, analysis, and planning stages. The process of developing the finished system involvescoding and debugging. Deliverables of the implementation phase is making it a reality in the real world includes installation, testing, training and deployment.

The entire software system deliverable will be; User Documentation, System Documentation,Generate Report, software package and user manual.

1.11.6. Testing and deployment

As we mention on the design phase 1.12.4, that user testing is testing our design against theend user, and here system testing is about testing the completely implemented system in terms of functionality, performance, security and usability of the system to the end users.

System testing determines whether the soon to be deployed system is ready to be deployed.

Web app tests fall into five main categories:

- Functional tests: does the app work?
- Compatibility tests: does the app work consistently for everyone?
- Performance tests: does the app respond quickly and how does traffic affectperformance?
- Security tests: is the app secure against attacks?
- Usability tests: is the app easy to use and does it responds to interaction asexpected?

When it comes to deployment, deployment will script the replication of changed files from one's local environment to the live server. It can be done through version control software or resync, resync is a utility for efficiently transferring and synchronizing files between a computer and a storage drive and across networked computers by comparing the modificationtimes and sizes of files.

Web app deployment has three parts: the local preparation (or build), the transfer of files, andthe post-upload remote configuration.

A build is normally associated with the compilation of code into executable files, but theterm can also apply to popular interpreted web languages that don't require compilation, suchas PHP or Ruby. For the purposes of deployment, the release build process normally includes:

- A fresh checkout of the code to a test environment.
- Preparation of files for the live environment, which may include minimization ofJavaScript and CSS files, and bundling individual images into single sprite files.
- A run through all automated tests where the deployment will halt on a failed test.
- Configuration of files for the live environment (database settings, for instance).
- Automatic creation of release notes or updated documentation.

Our main deliverables of testing are full access in real time for user and Web Application.And deliverables for deployment will be the installed and configured system.

1.12. Work-break down Structure

A work breakdown structure (WBS) is a project management tool that takes a step-by-

step approach to complete large projects with several moving pieces. By breaking down the projectinto smaller components, a WBS can integrate scope, cost and deliverables into a single tool. While most WBSs are deliverable-based, they can also be phase-based. This project has a totalof 6members, 4 females and 2 male students, and each member has taken a roll in the project as distinguished in the table below.

No	WORK BREAK DOWN	RESPONSIBLE MEMBER
1.	Chapter 1 – Introduction	
1.1.	Overview	Soliyana Yalewdeg
1.2.	Background of the organization	Soliyana Yalewdeg
1.3.	Statement of the Problem	Soliyana Yalewdeg
1.4.	Objectives of the project 1.3.1. General objective 1.3.2. Specific objectives	Tsion Asdegdg
1.5.	Feasibility Analysis 1.4.1. Economic Feasibility 1.4.2. Operational Feasibility 1.4.3. Schedule Feasibility 1.4.4. Legal and Contractual Feasibility 1.4.5. Political Feasibility 1.4.6. Social Feasibility 1.4.7. Technical Feasibility	Mintamr Agegnehu
1.6.	Significance of the Project	Tsion Asdegdg
1.7.	Beneficiary of the project	Sofonias Mifta
1.8.	Methodology 1.7.1. Data collection 1.7.2. System Development Methodology	Mintamr Agegnehu

1.9.	Development Tools and Technologies 1.8.1. Front-end Technologies 1.8.2. Back-end Technologies 1.8.3. Documentation and Modeling Tools 1.8.4. Deployment Environment	Selamawit Lemma
1.10.	Scope	Selamawit Lemma
1.11.	Risks, assumptions and constraints	Selamawit Lemma
1.12	Phases and deliverable of the project	Sofonias Mifta

1.13	Work-breakdown structure	Temesgen G/abzgi
1.14	Project schedule	Temesgen G/abzgi
2.	Chapter 2- Business Area Analysis and Requirement Definition	
2.1	Introduction	Tsion Asdegdg
2.2	Business area analysis 2.2.1. Activities/functions of the organization 2.2.2. Problems of the current system 2.2.3. Forms and Reports of the current system 2.2.4. Players of the existing system	Tsion Asdegdg ,Selamawit Lemma,Sofonias Mifta
2.3.	Requirements Definition 2.3.1. Functional Requirements 2.3.2. Non-functional requirement	Mintamr Agegnehu, TemesgenG/abzgi, Soliyana Yalewdeg
3.	Chapter Three-Object Oriented Analysis	
3.1.	Overview	All Group members participate here
3.2.	3.2. Use case Modeling 3.2.1. UI identification 3.2.2. Business rules identification 3.2.3. Actor identification 3.2.4. Designing the use case diagram 3.2.5. Use case description	
3.3.	3.3. Conceptual Modeling 3.3.1. Class diagram 3.3.2. Class description	
3.4.	Sequence Diagramming	
3.5.	User interface prototyping	
4.	Conclusion of the first phase	

Table 1.4 Work breakdown structure

1.13. Project schedule

Involves checking, if the project team can develop the proposed system with the time allocatedor not. Meeting project deadline may depend on the size the project team and the availability of the key members of the user group. The table below describes schedule of our project.

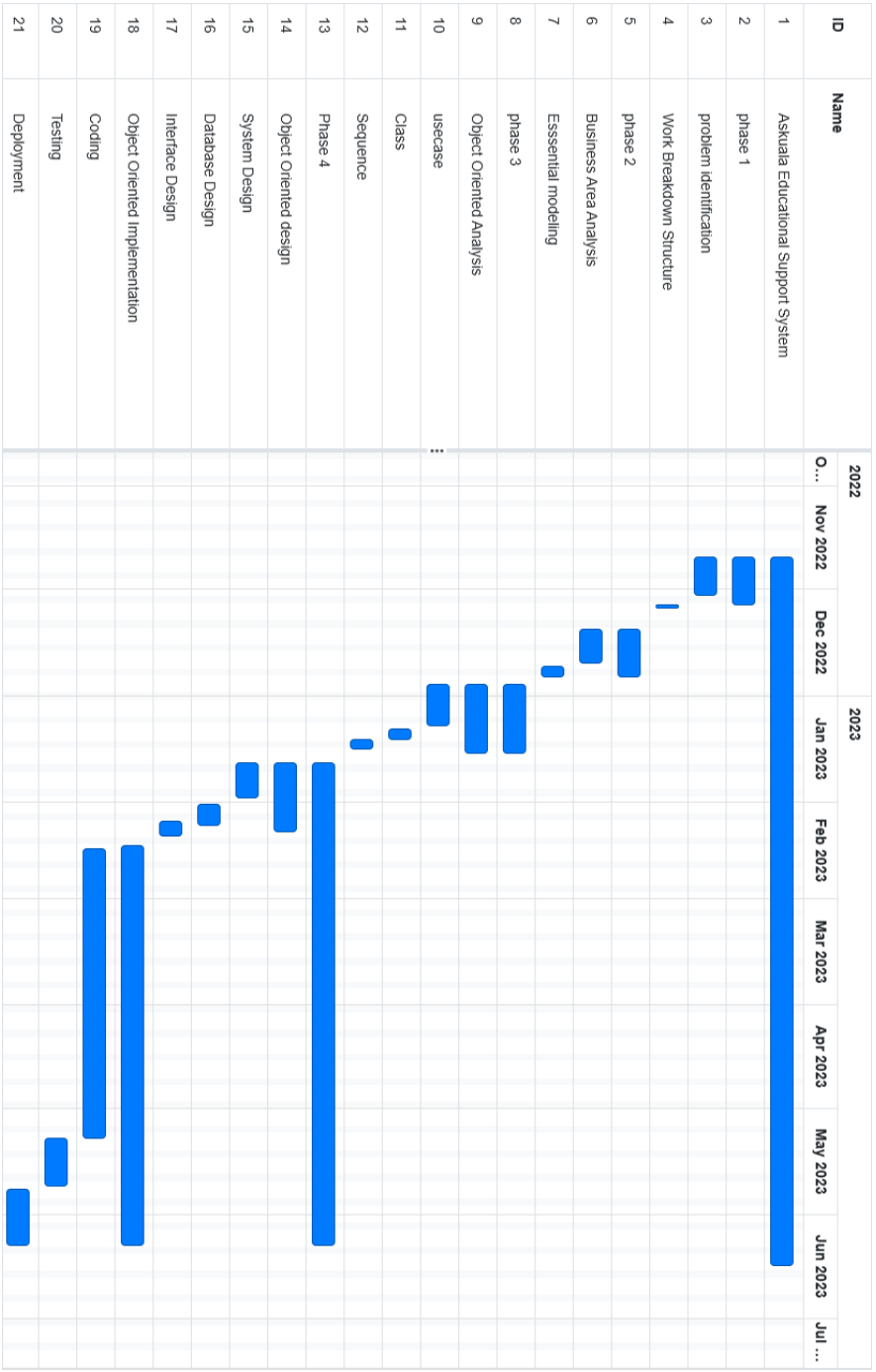


Figure 1.2. project schedule

CHAPTER TWO

2. Business area analysis and requirement definition

Business area analysis and requirement definition is the process of discovering, analyzing, defining, and documenting the requirements that are related to a specific business objective. And it's the process by which you clearly and precisely define the scope of the project, so thatyou can assess the timescales and resources needed to complete it.

2.1. Overview

The contents of this chapter include a listing of the activities of the organization and an analysisof the major functions of the existing teaching learning support system in order to identify the problems that occur. Furthermore, Students, Teacher and Addis Ababa university specially College of Natural and Computational Science, which we develop a system for, play a major role on the overall process of the education system that have been identified followed by an elicitation of the functional and non-functional requirements of the proposed system formed from the analysis conducted.

2.2. Business area analysis

Business analysis is the process of examining and evaluating business demands and identifyingsolutions to potential challenges.

Understanding how the business works to achieve its goals is the goal of business analysis. It comprises identifying the skills required for the company to supply products to external stakeholders. In this section of the project, we will have to figure out how the organization's goals relate to specific goals. We also need to come up with a clear plan to help you meet yourgoals and objectives. We'll define how stakeholders and different organizational units interactin your business study.

2.2.1. Activities/Functions of the organization

Activities or functions of the organization are the descriptions of the primary function of an Organization are to achieve specific goals and objectives. This may involve providing products or services to customers, generating revenue and profits, advancing a social cause or mission, or fulfilling the needs of its members or stakeholders.

The major activities of College of Natural and Computational Sciences are to give course which sectioned with two semester for undergraduate and graduated class. The student evaluate by exams, assessment and also project.

The student first register online through portal of the university and fill the course they want to take and start class.

2.2.2. Problems of the current system

In this section, we will try to identify the problems within in the existing or current information system. Problem of the current system is the gap between the existing system and the desired goal. We used PIECES framework will for this study.
PIECES framework method is a framework used to classify a problem, opportunities, and directives contained in the scope definition of analysis and system design, so that it can generate new things that canbe considered in developing the system this frame work is a problem-solving framework, which Uses to frame your investigation of the problems, opportunities, and requirements. (Fatoni,2020).
We select the PIECES framework to point out or describe the problems of the existing systemin order to develop the basic understanding of the problems that the existing system has in addition to that it also aids in building the new system based on solutions of the problems thathave been identified. There are six fundamental variables to analyze the current system, namely Performance, Information and Data, Economics, Control and Security, Efficiency, and Service.

- **Performance**

The current teaching learning system may not effectively engage all students, leading to a lack of understanding and poor performance. This can result in students not reaching their full potential and a lack of achievement in academic goals.

- **Information**

The existing system often relies heavily on lectures and memorization, rather than hands-onlearning and critical thinking skills. This can result in student's not retaining information as well as they could with more interactive and engaging methods.

- **Economics**

The current teaching learning support methods can be expensive, especially in terms of materials and resources. This can limit access to education for those with limited financial resources.

- **Control and Safety**

There are several control and safety issues in existing process, these issues include lack of accountability and lack of tracking student progress, leading to difficulty in evaluating the effectiveness of the learning process; lack of access to educational resources for all students; and lack of technological literacy, leading to students not understanding how technology can be used to improve learning. Additionally, traditional teaching and learning systems often rely on a one-size-fits-all strategy, leading to students being left behind or not receiving the best education possible.

- **Efficiency**

Traditional teaching methods may not be as efficient as more modern and innovative methods. For example, lectures can be slow-paced and students may not retain information as well as they would with more interactive methods. This can result in a lack of efficiency in the learningprocess and a waste of resources.

- **Services**

The current system doesn’t give learning services like question answering, discussion and adequate education materials.

2.2.3. Forms and Reports of the Current System

College of Natural and Computational Science do not use teaching learning support platform currently, although, the college uses the general AAU portal website as does every other college under AAU, for student registration, courses registration, cost sharing information and other genericinformation that are necessary for a student of AAU to complete before they get enrolled to the learning system.

2.2.4. Players of the existing system

The players of the current system in teaching and learning process are:

- -Instructor: instructors assigned to a specific course by the department and teach students basedon a schedule made by the department.
- Students: the students under a department, who takes course classes held by teachers of thecourse.
- department staff: in charge of assigning teachers to classes and scheduling

2.3. Requirement definition

A requirement is a statement that identifies a product or process operational, functional, ordesign characteristic or constraint, which is unambiguous, testable or measurable, and necessary for product or process acceptability.

Defining the requirements of a project focuses on what needs to be done and the techniques and methods used to solve the problem proposed during the initial discussion. Having a clearunderstanding of the requirements is necessary to produce a successful software project. (Roger Y. Lee, 2019) .In this part we discuss functional requirement and nonfunctional requirement as follows.

2.3.1 Functional Requirement

Functional requirements describe the behavior that the system must exhibit. These requirements define how the functions and underlying structure of the applications are comprised to meet the desired behavior outlined by the customer. Functional requirementsare focused on addressing the quality characteristic of functionality. (Roger Y. Lee, 2019)

Name	Description
Registration	The system allows student to register to the System
Log in to account	The system allows users to login to their account provided that the user filled the right combination of username and password.
Search	The student can search for department for his/her choosing and a student to search for the courses video, audio and text books etc...
Question and answering	The students can communicate via the system by question and answering.
Upload course materials	The teacher can upload course materials that are helpful to the student.

Download course materials	The student can download the course material that is shared for him/her by the teacher.
Create Virtual Class	Instructor can create a class in which virtual discussion can be held, assignments can be given and results can be published for assignments.
Access exam bank	Students can access previous exam collections.
Activate/ deactivate account	The administrator activates and deactivates the accounts of student or tutor.

Table 2.1Functional Requirements

2.3.1. Nonfunctional Requirement

Nonfunctional requirements are the requirements of a system when it is deployed (that is, when the system is in operation). Therefore, they are also called operational requirements

NFRs address issues such as the performance of an entire system under normal business transactions, scalability of a system for varying customer counts, and security of a system deployed over a web-based architecture. Additionally, security, volume, quality of service (QoS), and maintainability are also part of NFRs. (Unhelkar, 2018)

- **Performance**

Usually specified in terms of the speed of response expected from a system. The performance requirement for an Internet-based deployment depends on the available bandwidth. Factors such as available processing power and amount of data to be processed, for example, also impact performance.

- **Availability**

Examples of these requirements include permissible downtime for maintenance, number

of times system is allowed to be offline, and expected Quality of service for different types of system failures. Our system is 24/7 available.

- **Accessibility**

is a NFR that enhances user experience. Designing easy-to-access software solutions requires an understanding of the physical characteristics (and limitations) of users and their usability needs. Since we develop our website using bootstrap framework, users can access the website by different devices. This increases user experience.

- **Usability and user experience**

while usability itself deals with the ease of use of a typical user interface of the system, user experience deals with the overall "take away" of the user when interacting with the organization through the system. When we come to usability of our system the user interface of the website are attractive, easy to interact and understandable. Also user experience of the our website has many things for users like students can get course materials, different years exam questions and also they participate in discussion forum.

CHAPTER THREE

3. Object oriented analysis

Object-oriented analysis and design (OOAD) is a technical approach for analyzing and designing an application, system, or business by applying object-oriented programming, as well as using visual modeling throughout the software development process to guide stakeholder communication and product quality. Object oriented analysis and design (OOAD) views a system as a group of interacting objects. Object-oriented programming (OOP) allows implement of an object-oriented design as a working system. The object-oriented paradigm emphasizes modularity and re-

usability. The goal of an object-oriented approach is to satisfy the "open–closed principle. Requirements are organized as objects in object-oriented analysis in object-oriented. Several models used in OOA, mainly system use case, conceptual modeling, sequence diagramming and user prototypes of user interfaces.

3.1. Overview

These chapters mainly deal with object oriented analysis of the business area in terms of the new system to be developed by the team analysis. Actors who interact with the system are identified in the business rules applied to the business process. A user interface prototype, usecase diagrams, and sequence diagrams, in that order, have all captured these aspects.

3.2. Use case modeling

A use-case model is a model of how different types of users interact with the system to solvea problem. As such, it describes the goals of the users, the interactions between the users andthe system, and the required behavior of the system in satisfying these goals. (use_case_model_CD178AF9.htm, 2023)

We use a use-case diagram to graphically portray a subset of the model in order to make the communication simpler. There will regularly be a numerous use-case diagram which is related to the given model, each demonstrating a subset of the model components related to aspecific purpose. The use case model we used consists of: a use case diagram, a set of use case descriptions, a set of actor descriptions, user interface identification and a set of scenarios (described by flow of events).
Our use case diagram uses four concepts to graphically model the problem domain: use case,actor, relationship link and boundary.

- **Use case**: an ellipse marked with the name of the use case. By convention, we start eachuse case name with a verb to indicate that the use case represents a process. Therefore, we use “maintain customer list” instead of “customer list”, and “process query” instead of “query”.
- **Actor**: a simple stick-figure with the actor’s name.

- **Relationship**: a line connecting actors to use cases.
- **Boundaries**: A rectangle drawn around the use cases that separates them from the actorsfor describing the scope of the system.

3.2.1. UI identification

The user interface is the point at which human users interact with a computer, website or application. The goal of effective UI is to make the user's experience easy and intuitive, requiring minimum effort on the user's part to receive the maximum desired outcome.

UI is created in layers of interaction that appeal to the human senses (sight, touch, auditory andmore). They include both input devices like a keyboard, mouse, track-pad, microphone, touchscreen, fingerprint scanner, e-pen and camera, and output devices like monitors, speakers and printers. Devices that interact with multiple senses are called "multimedia user interfaces." Forexample, everyday UI uses a combination of tactile input (keyboard and mouse) and a visual and auditory output (monitor and speakers).
The following table below identify s the user interface including UI ID, UI name, the users with the privilege to access the respective page and .

	User Interface	**Users**	**Description**
UI_01	Landing page	Instructor, Student, coordinator, Sysadmin	A page that contains basic description about theservice and a gateway for Users, Administrator and Coordinator
UI_02	Home Page	Instructor, Student, coordinator, Sysadmin	A page that contains the links to the services and notification tab. the first page
UI_03	Registration page	Student, Instructor, coordinator, Sysadmin	A page that prompts the User to enter basic information to create an account
UI_04	Login page	Student, Instructor, Sysadmin	A page where User can fill their credentials to enter thesystem

UI_13	Contact us	Students, Instructors	A page that contains address links and phone numbers to contact the system administrators and where users can send feedback to system administrator
UI_14	Request to join class page	Students	A page where a student requests to join a class

Table 3.1 user interface identification

3.1.1. Business rules identification

A business rule is a rule that defines a specific constraints or instructions on how certain day- to-day actions should be performed within the context of a business. In computer software development, the business rules approach is a development methodology where rules are in a form that is used by, but does not have to be embedded in, business process management systems. (O'Berry, 2015)

	Business rule	**Description**	**Constraints**
BR_01	Student registration rule	Students must register in order to get course materials and to being member of a class created by teachers.	Must be a higher education student
BR_02	Material upload rule	Only teachers can upload course materials.	Must register to the system. user type must be instructor and/or data encoder

BR_03	Login authentication	system authenticates filled information	User id must match from the database. Password must match the database.
BR_04	Material access rule	Student gets the course materials and references after registration.	A student must be a member of a virtual class which the teacher created.
BR_05	Assessment rule	After being part of class student uploads an assignment given by the teacher.	An assessment must be uploaded by the course instructor.
BR_06	Teacher's Registration rule	Admin will assign teachers' for courses in charged for.	When a user registers as an Instructor, they must be authenticated by the admin.
BR_07	Evaluation rule	Students give feedback for course materials and assignments provided by instructors	Assessments must be uploaded by the instructor

Table 3.2 Business rule identification

3.1.2. Actor identification

An actor defines a person or entity outside of the system that interacts with it. (Roger Y. Lee, 2019). For this teaching learning support system we identified and described 4

actors in the following table below:

No	Actor	Description
1.	Student	The actor that can access uploaded materials. The actor which can participate in discussion session.
2.	Teacher /Instructors	Teacher or instructor is one who getsassigned by the admin, and has the privilegeto held a virtual class session actor who can upload materials.
3.	System administrator(sysadmin)	An Administrator control user accounts and can activate or deactivate accounts. Add teachers account. Also the admin control the system.
4	Coordinator	Coordinator is the actor who acts like a super admin and has the privilege to access the whole system and monitor main action in the system, generate reports of school data like; how many Users registered in a semester, how many classes created..

Table 3.3 Actor identification

3.1.3. Designing the use case diagram

The following diagram shows the use case diagram of Askuala teaching learning support system. It contains five actors with their distinguished use cases and connection with other actors.

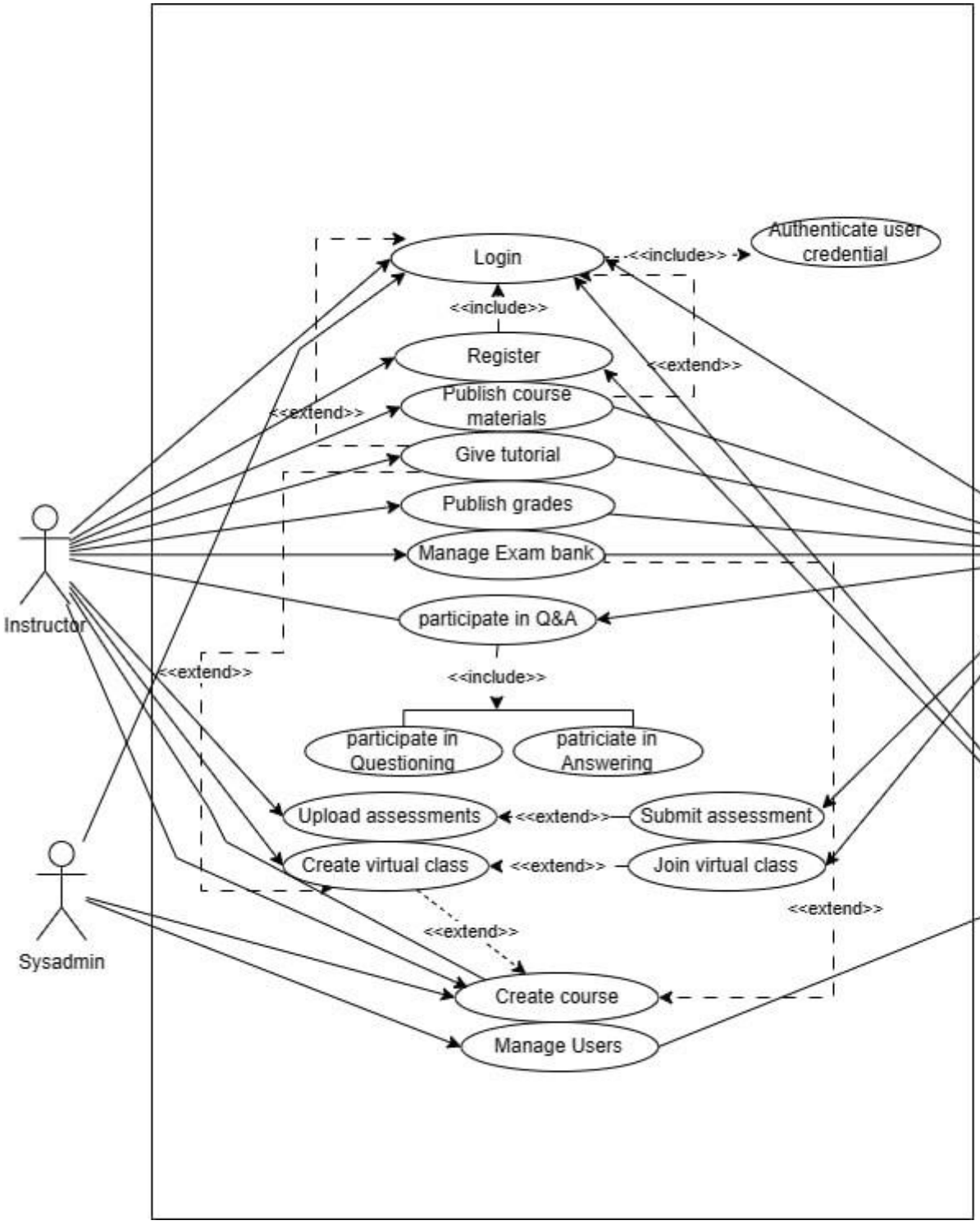


Figure 3.1 Use case diagram

3.2.5 Use case description

We designed eighteen use cases based on the five actors participating in the project;

which areStudent, Teacher, System administrator, Coordinator and Data encoder. The following tables describe all the use cases shown in the use case diagram.

Use case title	Register
Use case ID	UC 01
Description	Users create account to access the system
Actors	Instructor, Student, Coordinator
Precondition	Access to the system platform
Basic course of action: 1. The user wants to register to the system **2.** The user opens the landing page **"UI_01: Landing page".** **3.** The user selects the "**Sign up"** area **"UI_01: Landing page".** **4.** The user fills required information "**UI_03: Registration page".** **5.** The user clicks the "**Register"** button **"UI_03: Registration page".** 6. The system authenticates filled information 7. The system registers a new user into the database 8. The use case ends	
Post condition	The user registered to the system
Alternative course of action A: "the user leave a required information unfilled" **A6.** The system notifies the user there is unfilled required text box **A7**. the use case resumes from step 4	
Include	Landing page
Extend	Login

Table 3.4.Use case description of Register use case

Use case title	Login
Use case ID	UC 02
Description	Accepts user id and password to authorize users
Actors	Student, Instructor, Coordinator, system admin
Precondition	Registration

Basic course of action:

1. The user wants to login to the system
2. The users open the landing page **"UI_01: Landing page".**
3. The user selects the "**Sign in**" area **"UI_01: Landing page".**
4. The user enters user id and password **"UI_04: Login page".**
5. The user clicks the "**Login"** button **"UI_04: Login page".**
6. The system validates user id and password "**BR_08:Login authentication"**
7. User enters the system **"UI_02:Home page"**
8. The use case ends

Post condition	User login

Alternative course of action A: "user information doesn't match with the database"

A6. The system notifies the user the user id or password filled is wrong **"BR_08: login authentication"**

A7. The use case resumes at step 4

Alternative course of action B: "login validation fails three times"

B6. The system restricts user account for a day **"BR_09:login validation fails three times"**

B7. The system directs user to "forget password page"

Include	Register

Table 3.5 Use case description of Login use case

Use case title	Create virtual class
Use case ID	UC 03
Description	Instructors create a virtual class to share course material, assessments and grade results for students
Actors	Instructors
Precondition	Instructor register
Basic course of action: 1. User wants to create a class **2.** User selects the classes page **"UI_02:Home page"** **3.** User clicks "**create virtual class"** button**. "UI_06: Classes page".** **4.** User chooses a course and fills in virtual class description **"UI_06: Classes page".** **5.** User clicks create class button **"UI_06: Classes page".** **6.** System generates a new class **"UI_06: Classes page".** 7. Use case ends	
Post condition	A virtual class created
Include	Login
Extend	Join class

Table 3.6 Use case description of Create virtual class use case

Use case title	Join class
Use case ID	UC 04
Description	Students select their choosing virtual class and request to join the virtual class
Actors	Students
Precondition	Virtual class created
Basic course of action: 1. User wants to join a class **2.** User selects the classes page **"UI_02:Home page"** **3.** User chooses a class to join, from currently available classes **"UI_06: Classes page".** 4. The system displays a new page with the chosen class description and "Request to join class" button **"UI_14: Request to join class page"**. 5. User click the "**Request to join**" button **"UI_14: Request to join class page"**. 6. The class instructor checks the request legitimacy 7. The user joins the class 8. Use case ends	
Post condition	Student joined class
Alternative course of action A: "the requester user is not allowed to join the virtual class **"** **A7.** Instructor rejects the users request **A8.** use case ends	
Include	Create class
Extend	Submit assessment, comment for instructor

Table 3.7 Use case description of join class use case

Use case title	Publish course materials
Use case ID	UC 05
Description	Instructor uploads course materials for the specific class
Actors	Instructor
Precondition	Create class
Basic course of action: 1. User wants to share course materials with class members(students) **2.** User selects the "classes" page **"UI_02:Home page"** **3.** User chooses the specific class they want to upload course material to **"UI_06: Classpage".** **4.** User clicks the attachment button "**UI_06: Class page".** **5.** User chooses material from the file **"UI_06: Class page".** **6.** User shares materials with class members **"UI_06: Class page".** 7. Use case ends	
Post condition	Instructor shares materials with class members
Alternative course of action: none	
Include	Class

Table 3.8 Use case description of publish course materials use case

Use case title	Give tutorial class
Use case ID	UC 06
Description	Instructor conducts a live tutorial class in the virtual class
Actors	Instructor
Precondition	Create virtual class
Basic course of action: 1. User wants to held tutorial class 2. User selects the classes' page **"UI_02: Home page".**	

3. System displays the class page with a “create class” and “go live” button privileges **"UI_06: Class page".** 4. Instructor post an announcement for the tutorial class schedule 5. Based on the schedule, instructor clicks go live button 6. System generates a virtual meeting with the available number of users 7. Class students attend the tutor class via virtual meeting 8. Use case ends	
Post condition	Instructor teaches a tutor class in virtual class
Extend	Attend tutor

Table 3.9 Use case description of Give tutorial use case

Use case title	Attend tutorial class
Use case ID	UC 07
Description	Students member of a specific class attend a tutorial class given by the class instructor
Actors	Students
Precondition	Give tutorial class
Basic course of action: 1. User want to attend a tutorial class **1.** User selects classes page**"UI_02:Home page"** **2.** User Select the specific virtual class “**UI_06: Classes page".** **3.** User clicks “Join live session “button **"UI_06: Classes page".** **4.** User joined virtual class **"UI_06: Classes page".** 5. Use case ends	
Post condition	Students attend virtual tutor class
Alternative course of action: none	
Include	Tutorial class

Table 3.10 Use case description of Attend tutorial use case

Use case title	Publish assessment
Use case ID	UC 08
Description	Upload assessments to students in the class
Actors	Instructors
Precondition	Create virtual class
Basic course of action: 1. User wants to share assessments with class members(students) **2.** User selects the “classes” page **"UI_02:Home page"** **3.** User chooses the specific virtual class they want to upload assessments to **"UI_06: Classes page".** **4.** User clicks the attachment button "**UI_06: Classes page".** **5.** User chooses assessment from the file **"UI_06: Classes page".** **6.** User shares assessments with class members **"UI_06: Class page".** 7. Use case ends	

Post condition	Submit assessment

Table 3.11 Use case description of Publish assessments use case

Use case title	Submit assessment
Use case ID	UC 9
Description	Students submit assessments given by instructors of the class
Actors	Students
Precondition	Upload assessment question

Basic course of action: **1.** User wants to submit assessments **2.** User selects lasses page **"UI_02:Home page"** **3.** User selects a specific virtual class**"UI_06: Classes page".** **4.** User selects an assessment uploaded by the class instructor **"UI_06: Classes page".** **5.** User clicks the Submit button on the bottom of the assessment message **"UI_06: Classes page".** **6.** User clicks the attachment button **"UI_06: Classes page".** **7.** User selects file to upload **8.** User clicks submit button **"UI_06: Classes page".** **9.** Use case ends	
Post condition	Assessment submitted

Table 3.12 Use case description of Submit assessments use case

Use case title	Upload grades
Use case ID	UC 10
Description	Instructors upload students grade from the assessments

Actors	Instructors
Precondition	Assessment submission
Basic course of action: 1. User wants to upload student grades 2. User selects classes page **"UI_02: Home page".** 3. User selects specific class to upload grades to "**UI_06: Classes page".** 4. User selects members of student to send grades to **"UI_06: Classes page".** 5. User enters assessment values and final grade to a student **"UI_06: Classes page".** 6. Use case ends	
Post condition	Grade uploaded to students
Alternative course of action: none	

Table 3.13 Use case description of upload grades use case Table

Use case title	Participate in Question
Use case ID	UC 11
Description	Students Ask troubling question and participate in another students question

Actors	Students, instructors
Precondition	login
Basic course of action: **1.** User wants to ask their troubling question **2.** User goes to home page **“UI_02: Home page".** **3.** User clicks the big question mark button on the top of the page **“UI_02: Home page".** **4.** User directs the user to the Q&A page **“UI_07:Student forum page”** **5.** User types a problem to be solved on the text box provided **“UI_07:Student forum page”** **6.** User clicks “send” **“UI_07:Student forum page”** **7.** The system uploads the question on the list of questions **8.** Use case ends	

Table 3.14 Use case description of Participate in questioning use case

Use case title	Participate in Answer
Use case ID	UC 12
Description	Students Ask troubling question and participate in another students question
Actors	Students, instructors
Precondition	login
Basic course of action: 1. User wants to participate in question listed on the Q&A page 2. User clicks the Student forum page **“UI_02: Home page"** 3. User views questions asked by other students **“UI_07:Student forum page”** 4. User chooses a question to participate **“UI_07:Student forum page”** 5. The system displays the question selected with replay option below **“UI_07:Student forum page”** 6. User types their solution **“UI_07:Student forum page”** 7. Use case ends	

Post condition	participation
Include	Participate in Question

Use case title	Manage Exam Bank
Use case ID	UC 14
Description	Upload exams, upload solutions, delete exam records
Actors	Instructor
Precondition	none
Basic course of action: 1. User select Exam bank page **"UI_02: Home page".** 2. User clicks upload exam button "**UI_12: Exam bank page".** 3. User chooses a course which the uploading exam belongs to "**UI_12: Exam bank page".** 4. User select exam to from file to upload "**UI_12:Exam bank page".** 5. User clicks upload exam button "**UI_12: Exam bank page".** 6. Exam uploaded to the exam bank 7. Use case ends	
Post condition	Exam uploaded to exam bank

Table 3.15 Use case description of Manage exam bank

3.3. Conceptual modeling

Conceptual modeling is a visual representation of conceptual classes or real-world objects ina domain of interest. A conceptual model is illustrated with a set of class diagrams. It may show:

- Domain objects or conceptual classes
- Associations between conceptual classes
- attributes of conceptual classes (Larman, 2004)

3.3.1.Class diagram

A class diagram is defined as “a way of visualizing a software system based on the abstractions,or classes, that make it up, and the relationships between them” (Lee 2013). An object-orientedproject consists of a collection of classes that work together to form the foundation of the system. A class diagram provides a comprehensive view of the system under development by visualizing the classes that comprise it. During analysis, these diagrams are used to “indicate the common roles and responsibilities associated with all of the entities that define the system'sbehavior” **(Lee 2013). (Roger Y. Lee, 2019).** Class diagrams can show business-level classes, as well as technical classes derived from the implementation language (e.g., Java or C++ −). In addition to showing the classes, class diagrams also show the relationships between classes. The entire description of the classes (or entities, as they may be called in the problem space) and the relationships that they will have with each other is static. There is no dependency shown in this diagram and no concept of time. (Sundaramoorthy, 2022)

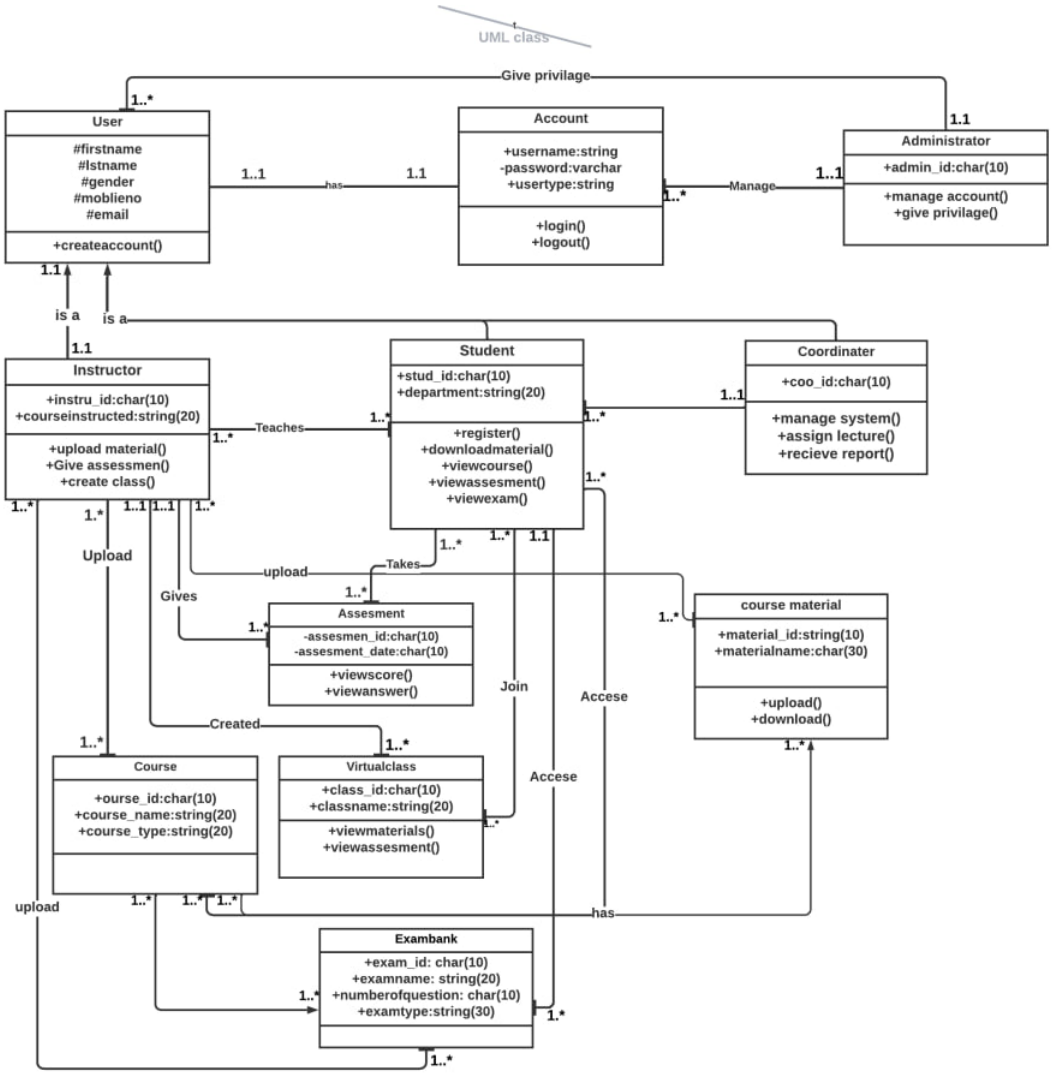

UML class
Give privilage
1..*
1.1
User
#firstname
#lstname
#gender
#moblieno
#email
+createaccount()
1..1
has
1.1
Account
+username:string
-password:varchar
+usertype:string
+login()
+logout()
1..*
Manage
1..1
Administrator
+admin_id:char(10)
+manage account()
+give privilage()
1.1
is a
is a
1.1
Instructor
+instru_id:char(10)
+courseinstructed:string(20)
+upload material()
+Give assessmen()
+create class()
Teaches
1..*
1..*
Student
+stud_id:char(10)
+department:string(20)
+register()
+downloadmaterial()
+viewcourse()
+viewassesment()
+viewexam()
1..*
1..*
Coordinater
+coo_id:char(10)
+manage system()
+assign lecture()
+recieve report()
1..1
1..*
1..*
1..1
1..1
1..*
1.*
Upload
Gives
upload
Takes
1..*
1..*
1..*
1.1
Assesment
-assesmen_id:char(10)
-assesment_date:char(10)
+viewscore()
+viewanswer()
1..*
course material
+material_id:string(10)
+materialname:char(30)
+upload()
+download()
1..*
1..*
Join
Accese
Accese
Created
1..*
1..*
Course
+ourse_id:char(10)
+course_name:string(20)
+course_type:string(20)
Virtualclass
+class_id:char(10)
+classname:string(20)
+viewmaterials()
+viewassesment()
1..*
upload
1..*
1..*
1..*
has
Exambank
+exam_id: char(10)
+examname: string(20)
+numberofquestion: char(10)
+examtype:string(30)
1..*
1..*
1..*

Figure 3.2 Class diagram

3.3.2. Class Description

Class: User			
Elements	type	Visibility	Description
First name	Varchar	protected	It describes the first name of user.
Last_name	Varchar	protected	It describes the last name of user.
Gender	Varchar	protected	It describes the gender of user.
Mobile no	Integer	protected	It describes the phone number of user.
Email	Varchar	protected	It describes the email address of a user
Method	**Visibility**	**Description**	
Create account()	public	It used to create account by the user.	

Class: Student				
Attribute	type	Visibility	Size	Description
Stud_id	char	public	10	It uniquely identifies the student
Stud_department	String	public	20	It identifies the student department
Method	visibility	Description		
Register()	Public	Used to register student		
Download material()	Public	Used to download material		
View course list()	Public	Used to view course available		
View Assessment()	Public	Used to view Assessment give by the Instructor		
View exam()	Public	Used to view exam that is available in the exam bank		

Class: **Instructor**				
Attribute	type	Visibility	Size	Description

Inst_Id	String	public	10	It uniquely identifies the instructor
Course instructed	String	public	30	Identify course that is instructed by the instructor
Method	visibility	Description		
Upload material()	Public	It used for instructor to upload course materials on class		
Give assessment()	Public	It used to give assessment to students		
Create Virtual class()	Public	It used to create class		

4.

Class: **Administrator**				
Attribute	type	Visibility	Size	Description
Admin_id()	char	public	10	It uniquely identifies the Administrator
Method	visibility	Description		
Manage account()	Public	It used to manage accounts		
Give privilege()	Public	It used to give privilege to accounts		

Class: **Course**				
Attribute	type	Visibility	Size	Description
course_id()	char	public	10	It uniquely identifies the Courses
Course name	String	public	30	It identifies the Courses name
Course type	String	public	30	It identifies the type of the material whether it is audio, video or softcopy

Class: **Exam bank**				
Attribute	type	Visibility	Size	Description

exam_id()	char	public	10	It uniquely identifies the exams in the exam bank
Exam name	String	public	30	It identifies the exam name
number of questions	Integer	Public	10	It identify number of questions in each exam
exam type	string	public	30	It identify type of exam

Class: **Coordinator**				
Attribute	type	Visibility	Size	Description
coo_id	char	public	10	It uniquely identifies the Coordinator
Method	visibility	Description		
Manage system()	Public	It used to manage system		
Assign instructors()	Public	It used to assign instructors		
Receive report()	Public	It used to receive report from the system		

Class: **Account**				
Attribute	type	Visibility	Size	Description
Username	char	public	20	It uniquely identifies the username of users.
Password	Varchar	Private	20	Authentication of a user to access a given page.
user type	string	public	20	Identify user type
Method	visibility	Description		
login()	Public	It is used to log in to the system.		
logout()	Public	It is used log out from the system.		

Class: **Assessment**

Attribute	type	Visibility	Size	Description
Assessment_id	char	public	20	It uniquely identifies the assessment for users.
Assessment_name	String	public	20	It identify assessment name
Assessment_date	date	public	10	It identify assessment date
Method	visibility	Description		
View score ()	Public	It is used to view assessment score.		
View answer()	Public	It is used view assessment answer.		

Class: **Virtual class**				
Attribute	type	Visibility	Size	Description
class_id	char	public	20	It uniquely identifies the class.
Class name	String	public	20	It used to name class
Method	visibility	Description		
View material ()	Public	It is used to view material which uploads by Instructor.		
View assessment()	Public	It is used to view assessment which upload by Instructor.		

Class: Course material				
Attribute	type	Visibility	Size	Description
Material_id	char	public	10	It uniquely Material
Material_name	String	public	30	It identifies material name
Method	visibility	Description		
Upload	Public	Used to upload material from the lecture		
Download	Public	Used to download material by the student and teacher		

3.2. Sequence diagramming

Sequence diagrams represent the detailed interaction between actors and a system or between collaborating objects within a given time block. However, information as to what happened before the interaction started and what happens after the time block stops is not shown in the sequence diagram. While messages shown in the sequence diagram can have preconditions andpost conditions, these conditions are not directly visible in the diagram (Unhelkar, 2018)

I. Student Registration sequence diagram

The sequence diagram for student registration is provided in the figure below.

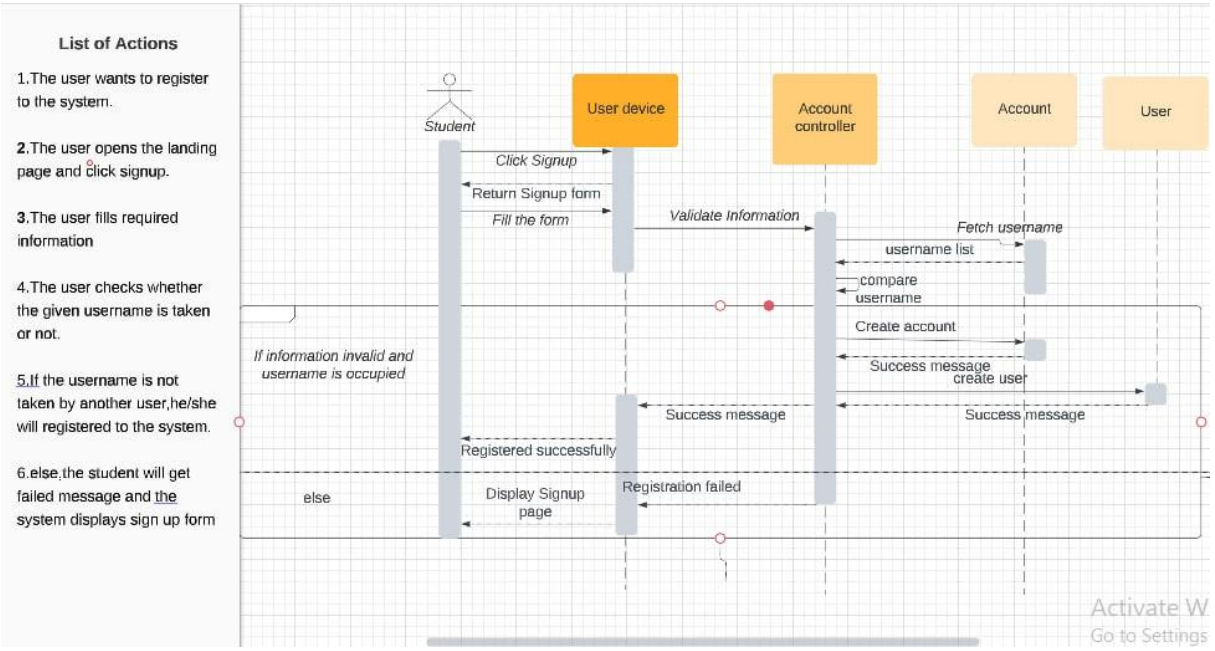


Figure 3.3 Student Registration sequence diagram

II. User login sequence diagram

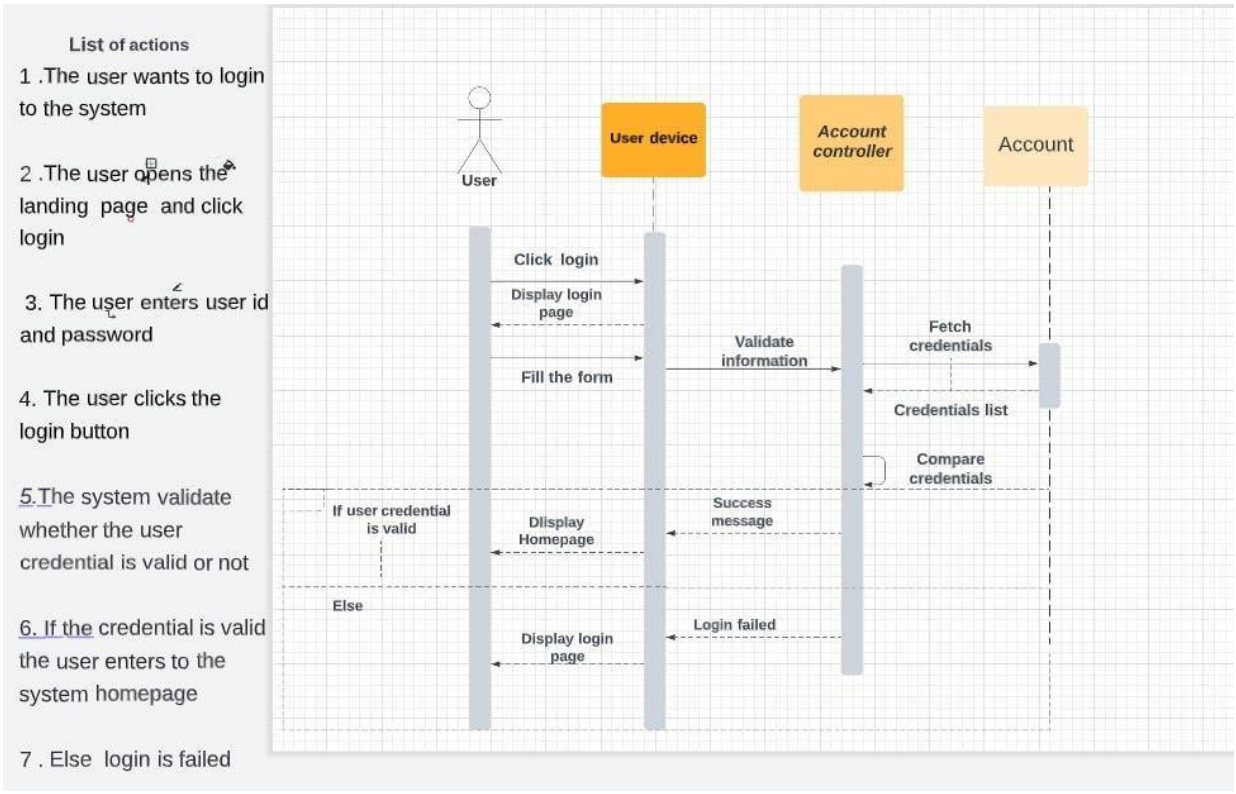


Figure 3.4 Student Registration sequence diagram

III. Upload materials Sequence Diagram

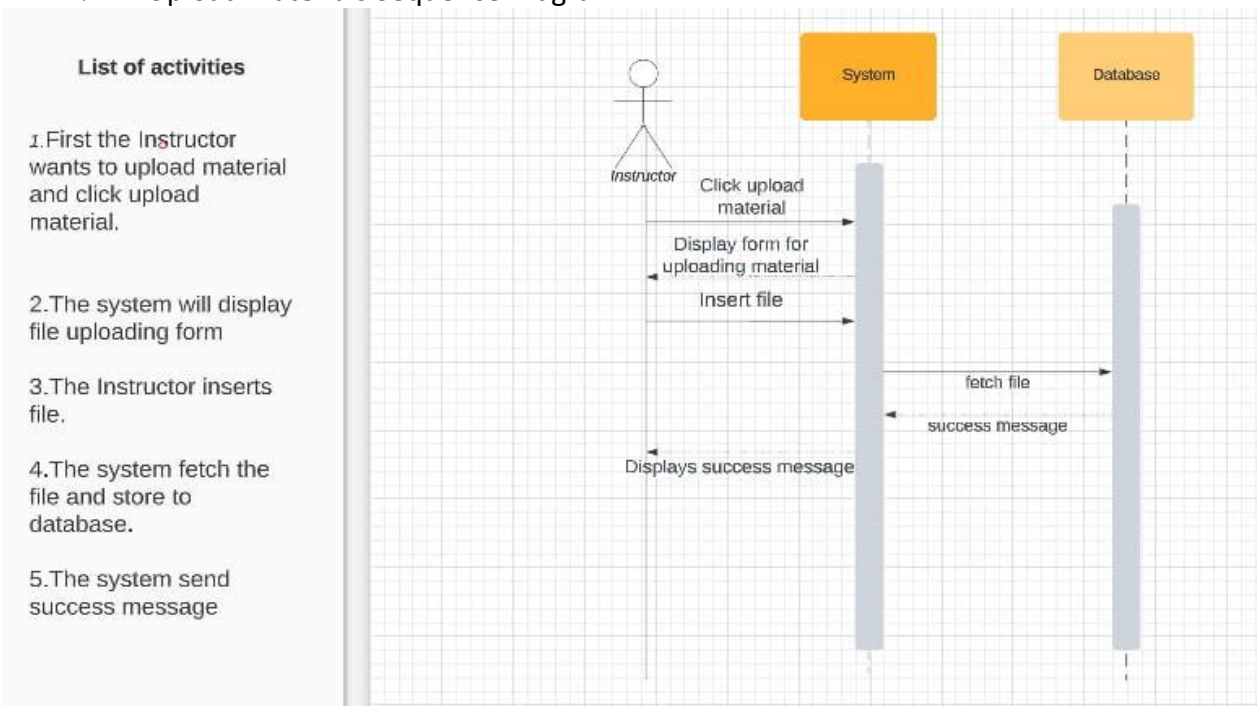


Figure 3.5 Upload materials Sequence Diagram

IV. Create Virtual Class Sequence Diagram

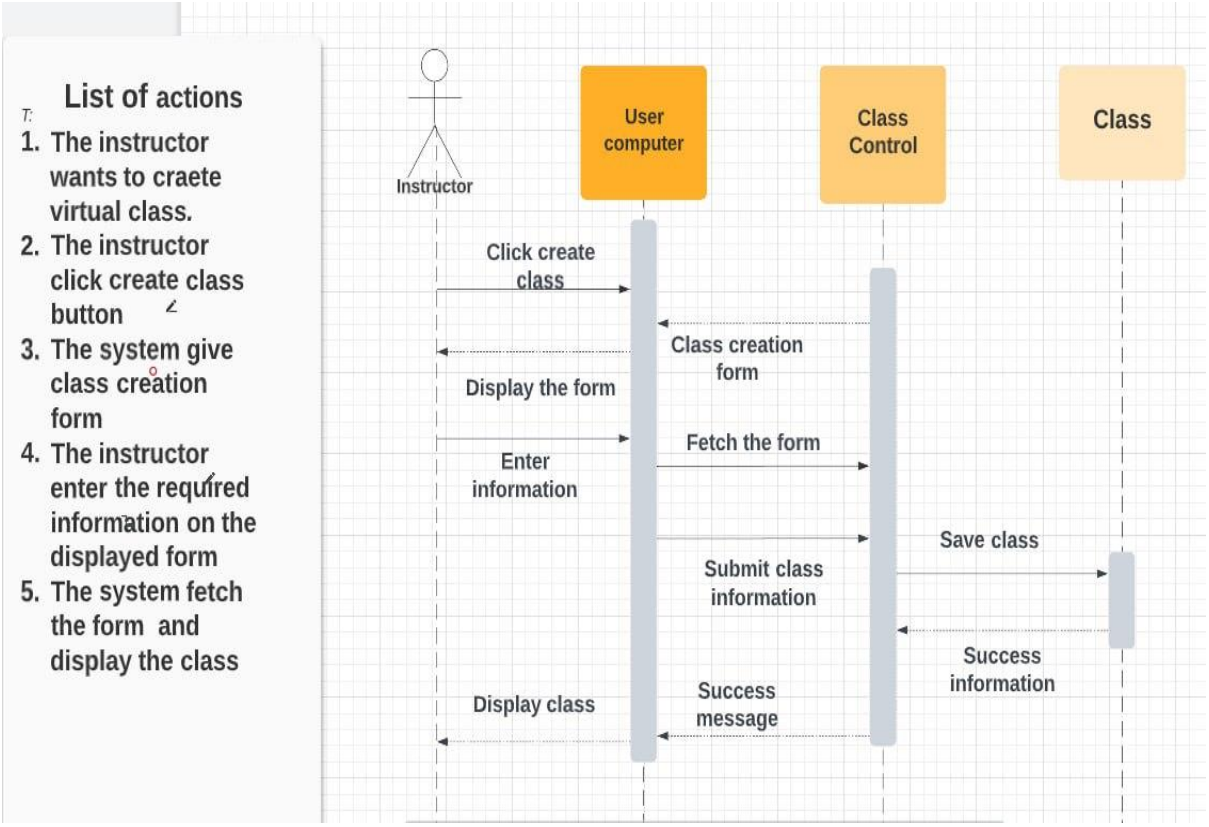


Figure 3.6 Create Virtual Class Sequence diagram

4.1. User interface Prototyping

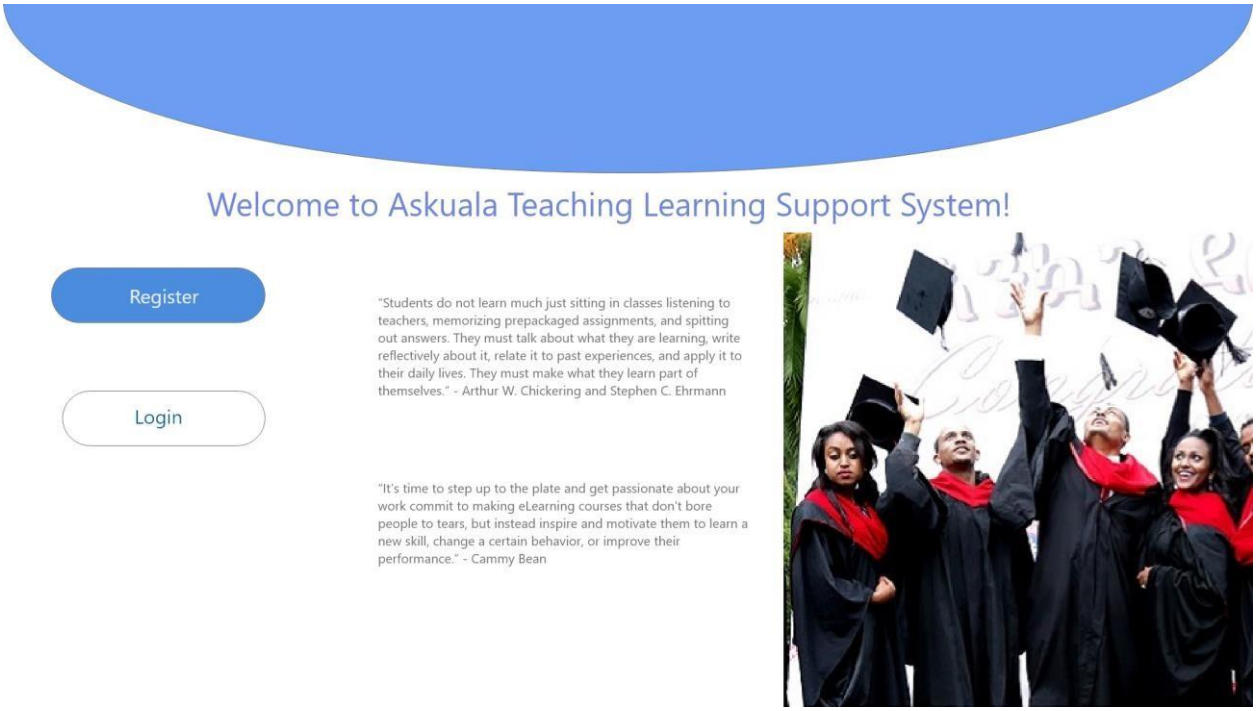


Figure 3.7 Landing page prototype

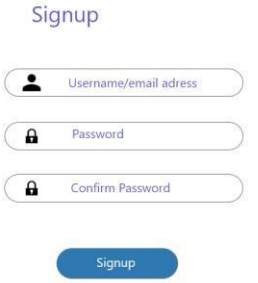


Figure 3.8 Registration page prototype

Figure 3.9 Login page

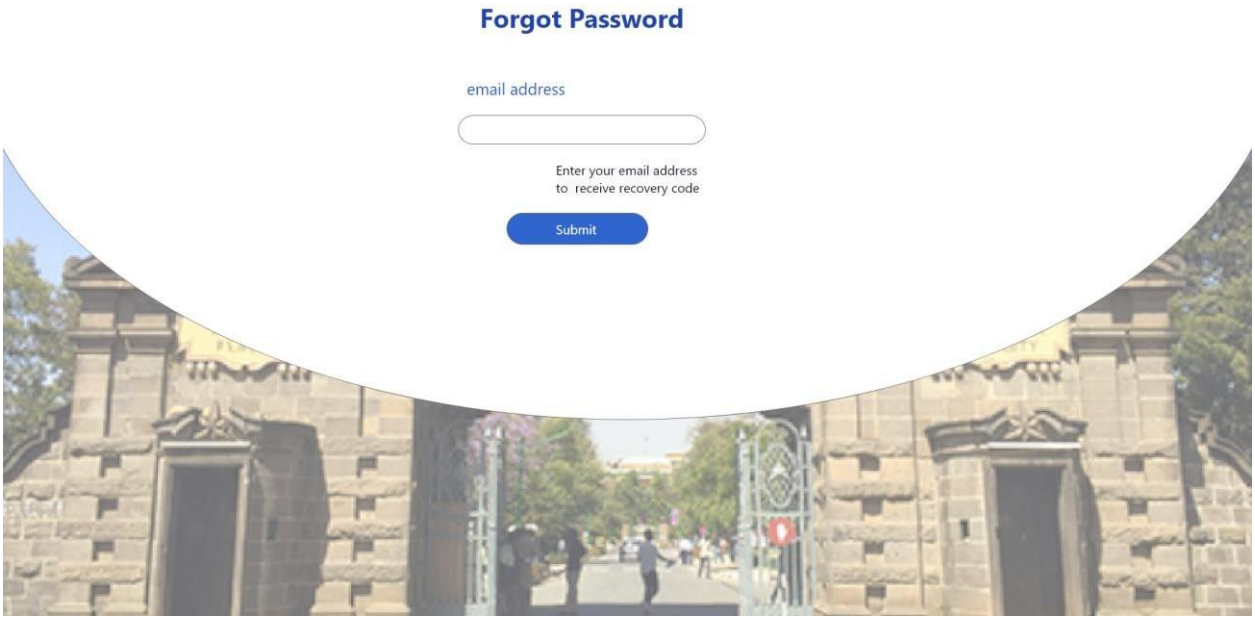


Figure 3.10 Forget password page

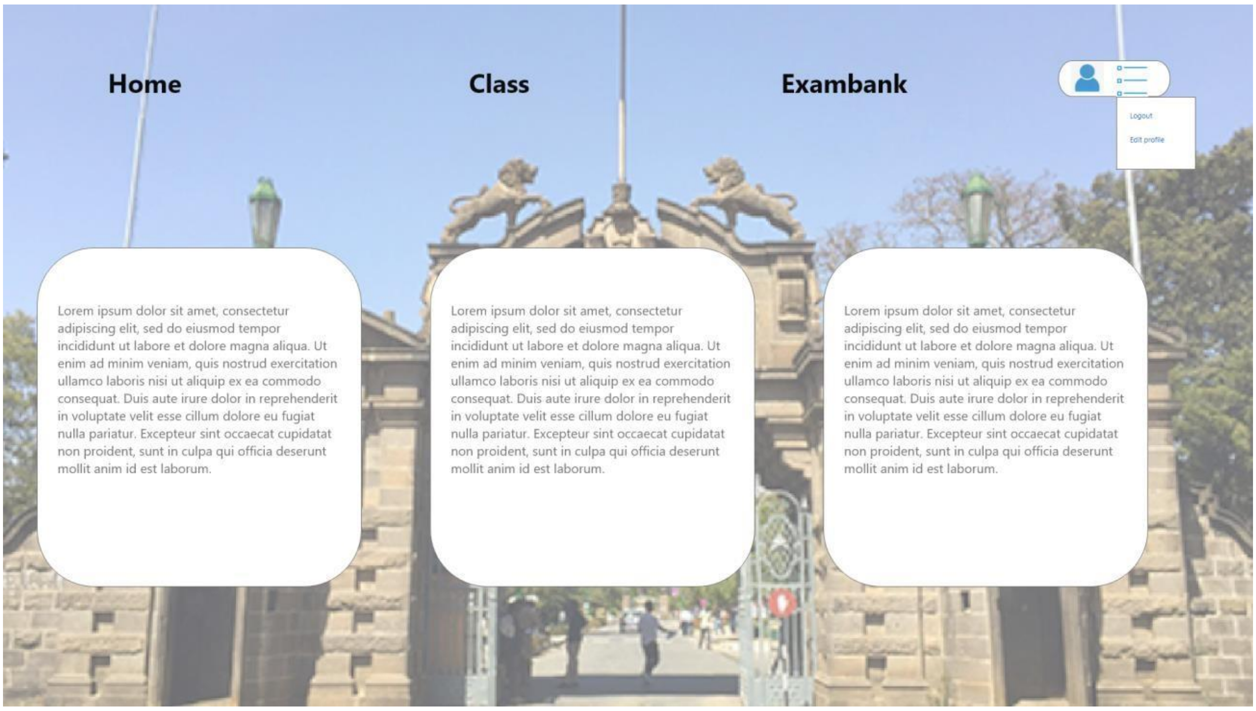


Figure 3.11 Homepage

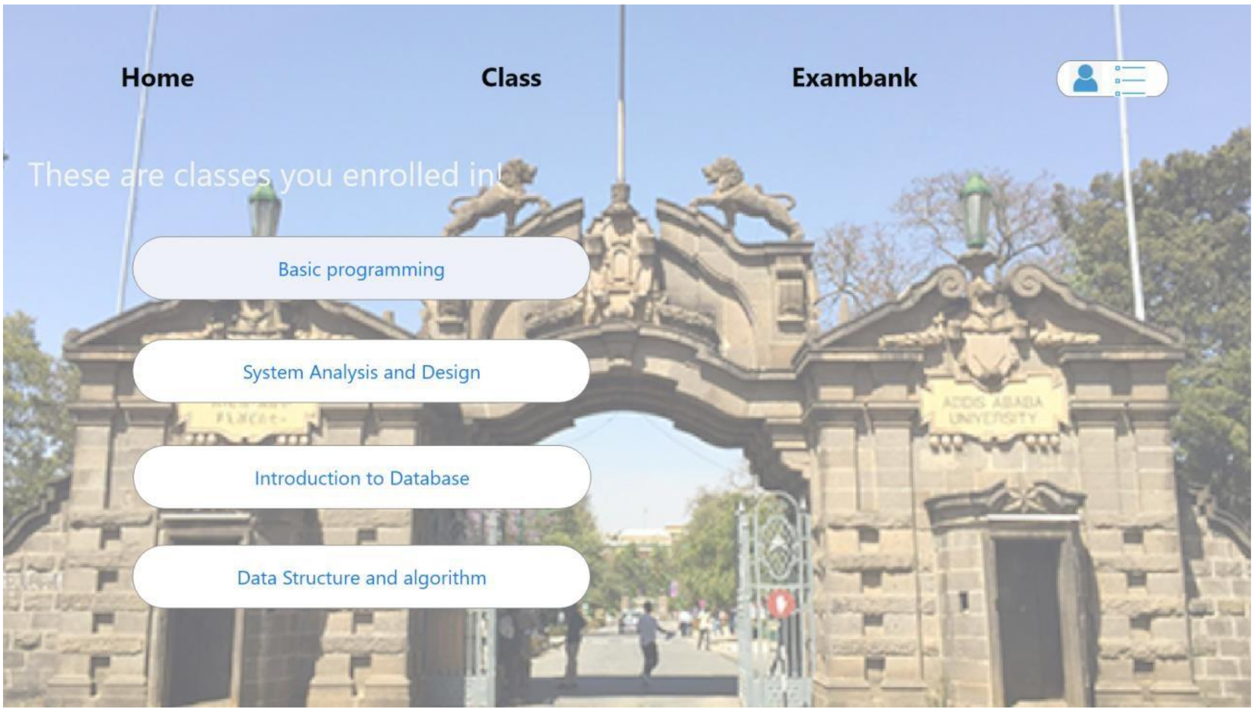


Figure 3.12 Class page

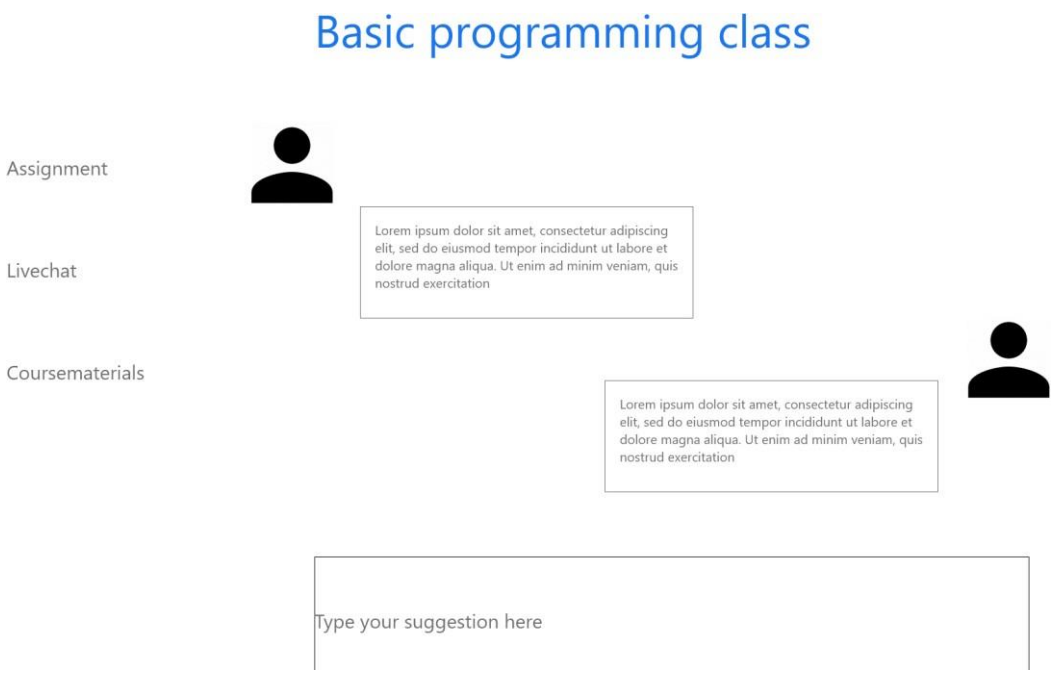


Figure 3.13 Virtual class page

CHAPTER FOUR

5. Conclusion

In chapter one, we determined the title of our project Askuala Teaching Learning Support System. Then the team has identified the problem statement, the objective of the project, the scope and limitation of the project. We have described the parties that benefit from our project;some feasibility studies and work breakdown structure have been discussed including the system methodology of the project which is suitable for conducting this project.

In Chapter 2 the team performed a detailed business area analysis that describes what the current system looks. In business area analysis the team identified the problem of the current system, the form and report of the existing system. After business area analysis we determinedthe requirements of the proposed system in terms of functional and non-functional requirements.

The third chapter of the project discussed about the object oriented analysis which tries to produce the conceptual model of information for the problem domain that raised on the chapterone of the existing system and solve that problem to accomplish this, the team used different types of object oriented analysis tools like system use case because our system is web based system, different diagrams such as sequence diagram and class diagram including user interface prototyping that is an extension of essential user interface. Here the actual proposed system ofthe team will be creates in theoretically with every steps of the system in the manner of the other people can understand our project when they try to maintain our system or modified partially in case of problem occurred or they want to add additional functionality.

References

F.George, J. S. (2021). *Modern System Analysis and Design.* Pearson Education.

Sundaramoorthy, D. S. (2022). UML Diagramming . In D. S. Sundaramoorthy, *UML Diagramming :a Case Study Approach* (p. 416). CRC Press.

MAL, S. K. (2023, February 8). *software-engineering-iterative-waterfall-model*. Retrieved from geeksforgeeks.org: www.geeksforgeeks.org/software-engineering-iterative-waterfall-model/

O'Berry, D. (2015, september 2). *blog/define-business*. Retrieved from quickbase.com: www.quickbase.com/blog/define-business

Roger Y. Lee, P. (2019). Object-Oriented software Engineering with UML. In P. Roger Y. Lee, *Object-Oriented software Engineering with UML* (p. 122). Nova Science Publishers, Inc. .

Sundaramoorthy, D. S. (2022). UML Diagramming . In D. S. Sundaramoorthy, *UML Diagramming :a Case Study Approach* (p. 416). CRC Press.

INDUSTRIAL PROJECT -2

Table of Contents

List of Tables

List of Figures

CHAPTER 5

OBJECT ORIENTED DESIGN

5.1 Introduction

A software design process known as object-oriented design (OOD) concentrates on modeling a system as a group of objects that interact with one another to carry out tasks. The basic building blocks of OOD are called objects, and each object includes properties and methods that describe its state and its actions, respectively.Making software that is simple to comprehend, maintain, and extend is OOD's major objective. OOD helps developers to write modular, reusable code that is simple to modify and adapt to changing requirements by breaking down a system into smaller, more manageable objects.

The advent of the object-oriented paradigm brought data and methods together into a reusable blueprint. Object is a class that defines the attributes (variables) and behaviors (methods) of that object. Using object-oriented design leads to code that is more maintainable, reusable, and adaptable, and is therefore a popular approach in modern software development. In this chapter, we addressed the major Object Oriented Design artifacts, such as class type architecture, class modeling, collaboration modeling, component modeling, deployment modeling, user interface design, and other design artifacts, in order to bridge the gap between analysis and implementation.

5.2 System Architecture

We used an architecture called MVC (Model View Controller).Model-View-Controller (MVC) is a software architecture pattern commonly used for creating user interfaces that divides an application into three interconnected components the model (data), the view (user interface), and the controller (processes that handle input). The main benefit of using MVC is that it helps to create loosely coupled and modular code, which is easier to maintain and update. MVC separates the business logic and presentation layer from each other.

The model

The Model represents the business logic and data of an application, such as the data used by the application or the rules and processes that operate on that data. It receives user input from the controller; the view renders the model's presentation in a specific format; and model code usually represents real-world objects. However, it should be noted that the model is unaware of the source of the data or the method by which it is acquired.

View

The View represents the presentation layer of the application, such as the graphical user interface that users interact with to view and modify data. The View describes in detail what is shown to the user. Most of the time, controllers pass data to every view for rendering in some manner. Views are another method of gathering user data.

Controller

The Controller acts as an intermediary between the Model and the View and is responsible for controlling the flow of data between them. The Controller may specify the procedures for adding tasks and designating others as complete. The controller is responsible for responding to the user input and performs interactions on the data model objects. The controller receives the input; it validates the input and then performs the business operation that modifies the state of the data model.

List of models

- Student
- Instructor
- Coordinator
- System Administrator
- User Account
- course
- Class
- Grade

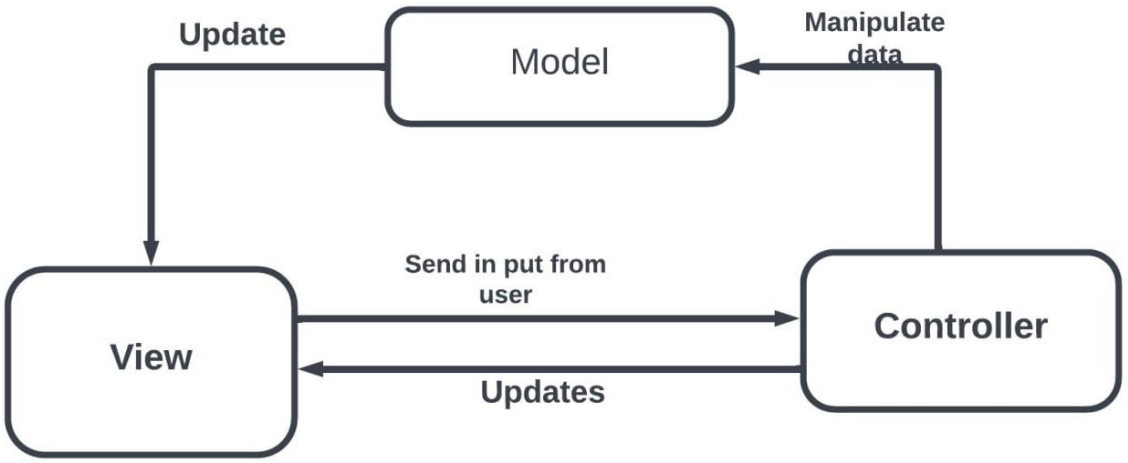
List of View

- Login Page
- Home Page
- Feedback page
- Grade Report
- Assessment
- Profile
- Material list

- Virtual class
- Exam bank

List of controller

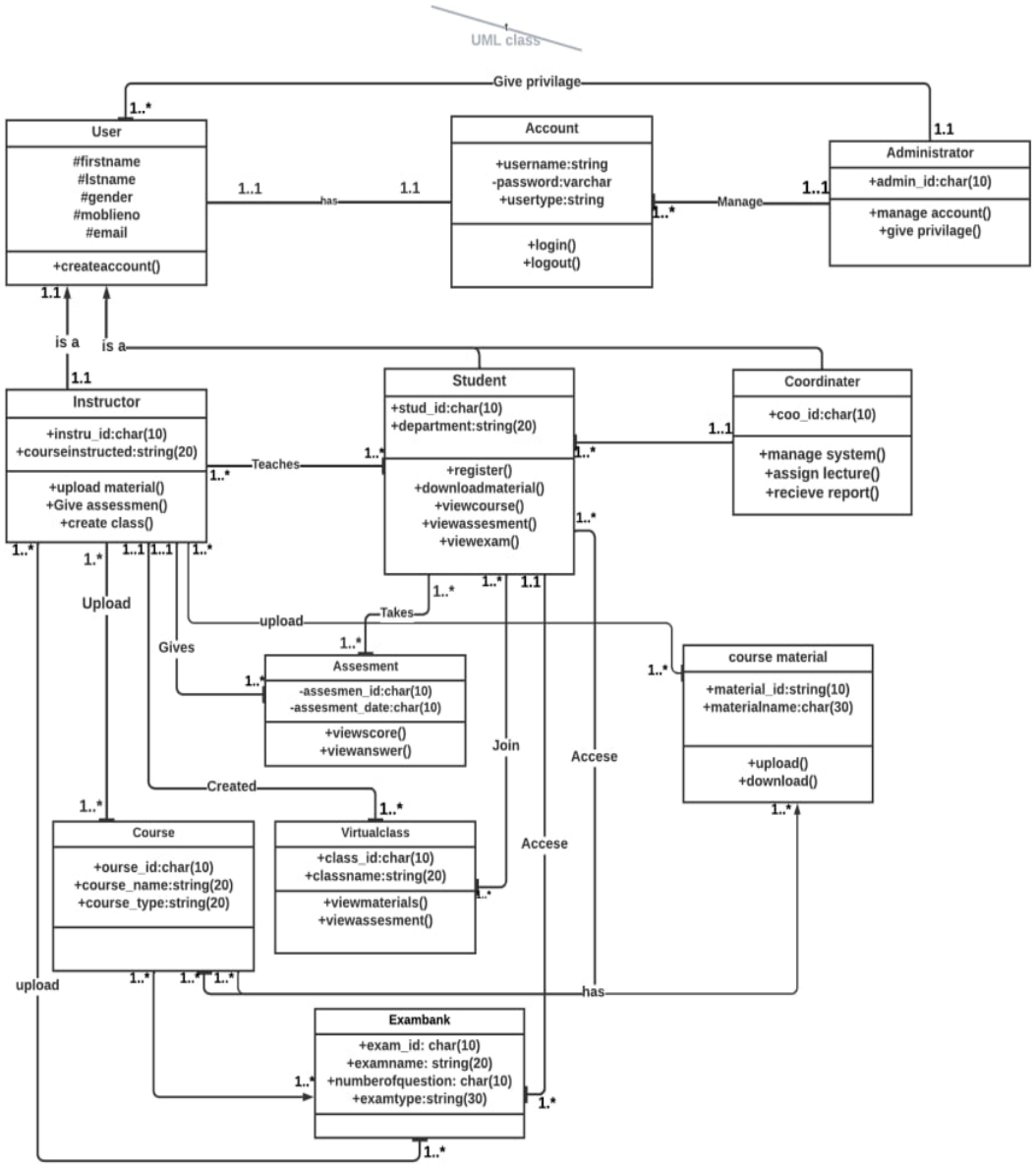
- User Account Controller
- Virtual class Controller
- Student Controller
- Grade Report Controller
- Registrar Controller



5.3 Design Class Modeling

5.3.1Class Diagram

Class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. It is representation of an object. It is a visual representation of the classes and their relationships in a system, which is used in software engineering to design and document object-oriented programming (OOP) systems.


UML class
Give privilage
User
#firstname
#lstname
#gender
#moblieno
#email
+createaccount()
Account
+username:string
-password:varchar
+usertype:string
+login()
+logout()
Administrator
+admin_id:char(10)
+manage account()
+give privilage()
has
Manage
is a
is a
Instructor
+instru_id:char(10)
+courseinstructed:string(20)
+upload material()
+Give assessmen()
+create class()
Student
+stud_id:char(10)
+department:string(20)
+register()
+downloadmaterial()
+viewcourse()
+viewassesment()
+viewexam()
Coordinater
+coo_id:char(10)
+manage system()
+assign lecture()
+recieve report()
Teaches
Upload
upload
Takes
Gives
Assesment
-assesmen_id:char(10)
-assesment_date:char(10)
+viewscore()
+viewanswer()
course material
+material_id:string(10)
+materialname:char(30)
+upload()
+download()
Join
Accese
Accese
Created
Course
+ourse_id:char(10)
+course_name:string(20)
+course_type:string(20)
Virtualclass
+class_id:char(10)
+classname:string(20)
+viewmaterials()
+viewassesment()
upload
has
Exambank
+exam_id: char(10)
+examname: string(20)
+numberofquestion: char(10)
+examtype:string(30)

5.3.2Description of classes

Class: User			
Elements	type	Visibility	Description
First name	Varchar	protected	It describes the first name of user.
Last_name	Varchar	protected	It describes the last name of user.
Gender	Varchar	protected	It describes the gender of user.
Mobile no	Integer	protected	It describes the phone number of user.
Email	Varchar	protected	It describes the email address of a user
Method	**Visibility**	**Description**	
Create account()	public	It used to create account by the user.	

Class: Student				
Attribute	type	Visibility	Size	Description
Stud_id	char	public	10	It uniquely identifies the student
Stud_department	String	public	20	It identifies the student department
Method	visibility	Description		
Register()	Public	Used to register student		
Download material()	Public	Used to download material		

View course list()	Public	Used to view course available
View Assessment()	Public	Used to view Assessment give by the Instructor
View exam()	Public	Used to view exam that is available in the exam bank

Class: **Instructor**				
Attribute	type	Visibility	Size	Description
Inst_Id	String	public	10	It uniquely identifies the instructor
Course instructed	String	public	30	Identify course that is instructed by the instructor
Method	visibility	Description		
Upload material()	Public	It used for instructor to upload course materials on class		
Give assessment()	Public	It used to give assessment to students		
Create Virtual class()	Public	It used to create class		

Class: **Administrator**				
Attribute	type	Visibility	Size	Description
Admin_id()	char	public	10	It uniquely identifies the Administrator
Method	visibility	Description		
Manage account()	Public	It used to manage accounts		
Give privilege()	Public	It used to give privilege to accounts		

Class: **Course**				
Attribute	type	Visibility	Size	Description
course_id()	char	public	10	It uniquely identifies the Courses
Course name	String	public	30	It identifies the Courses name
Course type	String	public	30	It identifies the type of the material whether it is audio, video or softcopy

Class: **Exam bank**				
Attribute	type	Visibility	Size	Description
exam_id()	char	public	10	It uniquely identifies the exams in the exam bank
Exam name	String	public	30	It identifies the exam name
number of questions	Integer	Public	10	It identify number of questions in each exam
exam type	string	public	30	It identify type of exam

Class: **Coordinator**				
Attribute	type	Visibility	Size	Description
coo_id	char	public	10	It uniquely identifies the Coordinator
Method	visibility	Description		
Manage system()	Public	It used to manage system		

Assign instructors()	Public	It used to assign instructors
Receive report()	Public	It used to receive report from the system

Class: **Account**				
Attribute	type	Visibility	Size	Description
Username	char	public	20	It uniquely identifies the username of users.
Password	Varchar	Private	20	Authentication of a user to access a given page.
user type	string	public	20	Identify user type
Method	visibility	Description		
login()	Public	It is used to log in to the system.		
logout()	Public	It is used log out from the system.		

Class: **Assessment**				
Attribute	type	Visibility	Size	Description
Assessment_id	char	public	20	It uniquely identifies the assessment for users.
Assessment_name	String	public	20	It identify assessment name
Assessment_date	date	public	10	It identify assessment date
Method	visibility	Description		
View score ()	Public	It is used to view assessment score.		

View answer()	Public	It is used view assessment answer.

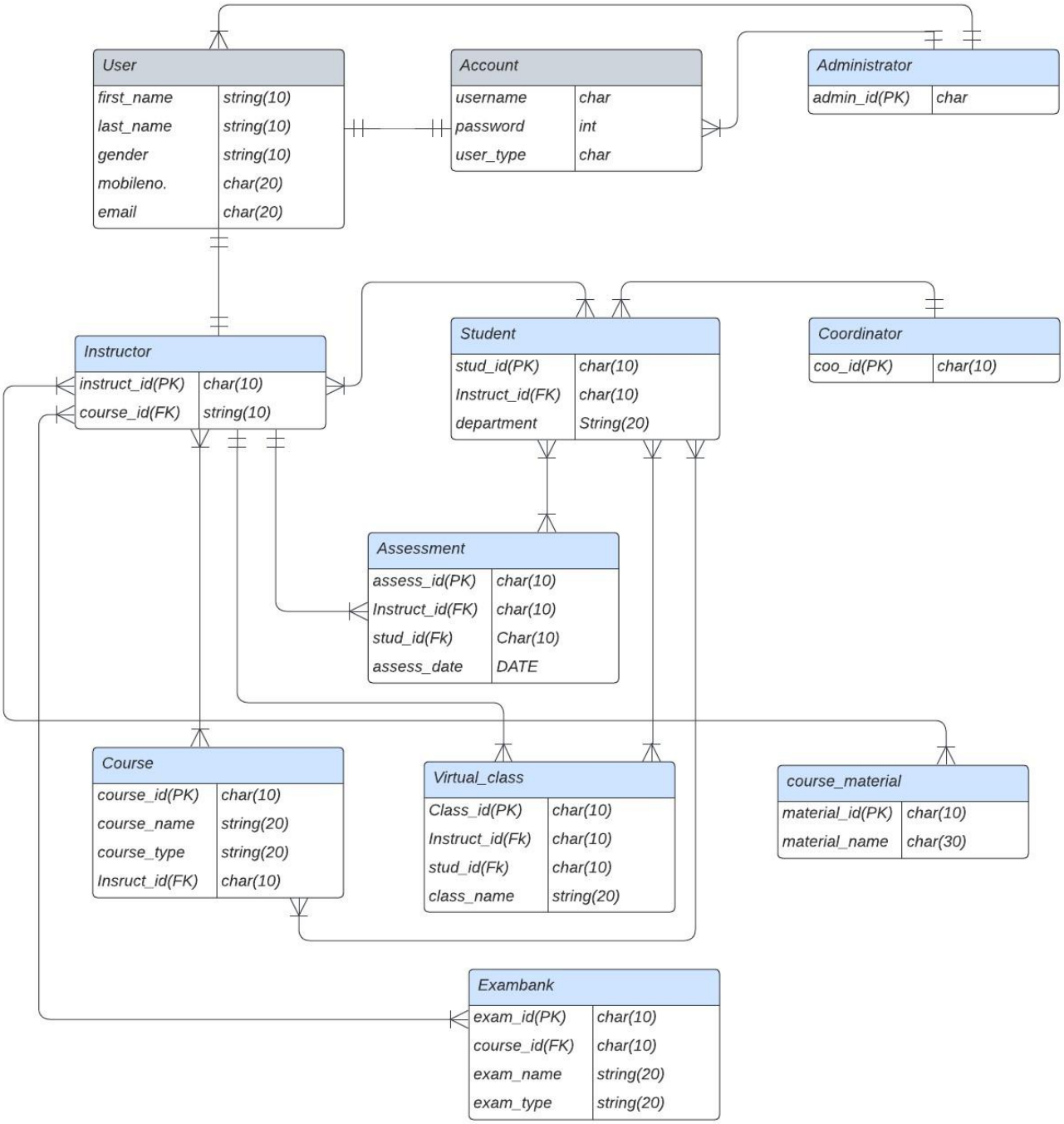
Class: **Virtual class**				
Attribute	type	Visibility	Size	Description
class_id	char	public	20	It uniquely identifies the class.
Class name	String	public	20	It used to name class
Method	visibility	Description		
View material ()	Public	It is used to view material which uploads by Instructor.		
View assessment()	Public	It is used to view assessment which upload by Instructor.		

Class: Course material				
Attribute	type	Visibility	Size	Description
Material_id	char	public	10	It uniquely Material
Material_name	String	public	30	It identifies material name
Method	visibility	Description		
Upload	Public	Used to upload material from the lecture		
Download	Public	Used to download material by the student and teacher		

5.4 Relational Persistent Model

Persistence is "the continuance of an effect after its cause is removed". In the context of storing data in a computer system, this means that the data survives after the process with which it was created has ended. In other words, for

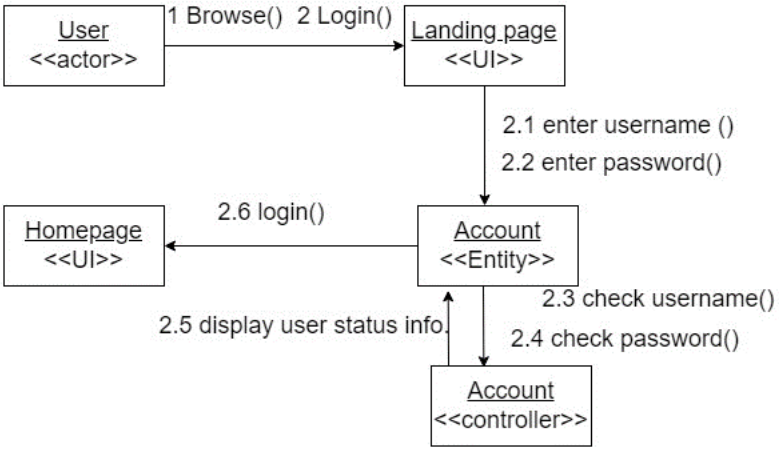
a data store to be considered persistent, it must write to non-volatile storage. We designed the Relational persistent model for Askuala Teaching learning Support system as follows:



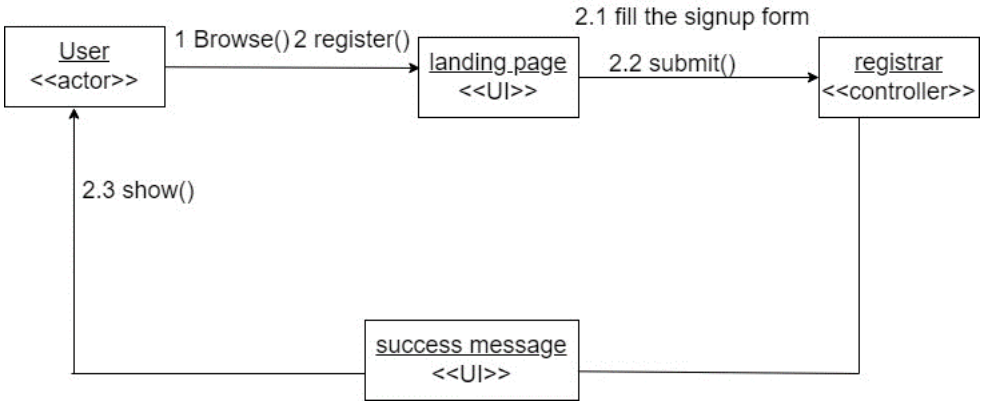
5.6 Collaboration diagram

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML).

1 Collaboration diagram for login

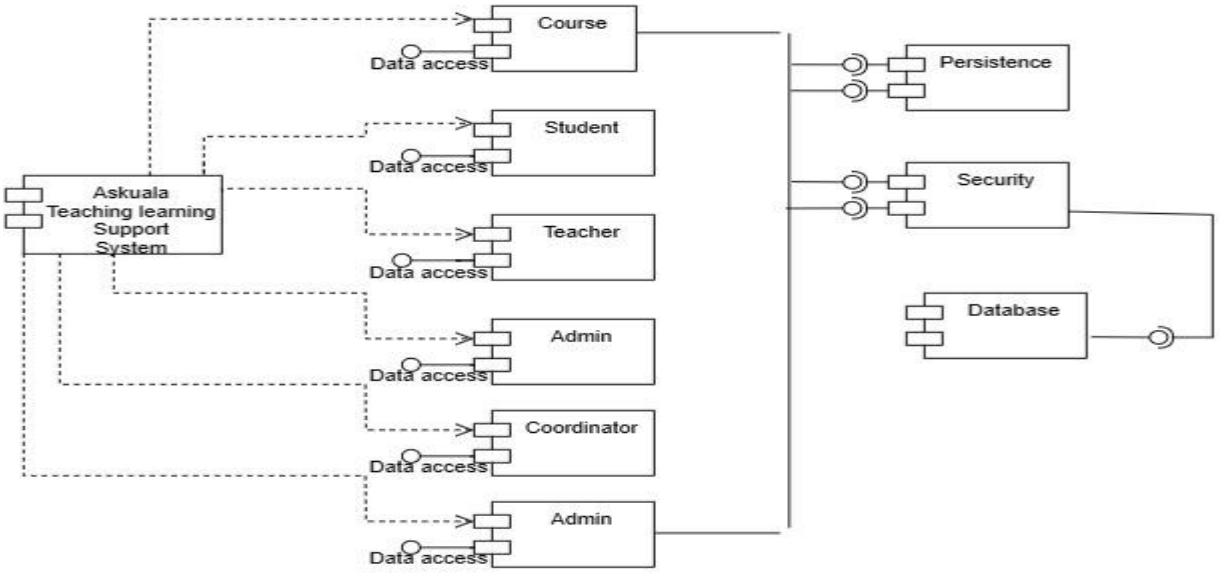


2 .Collaboration diagram for registration



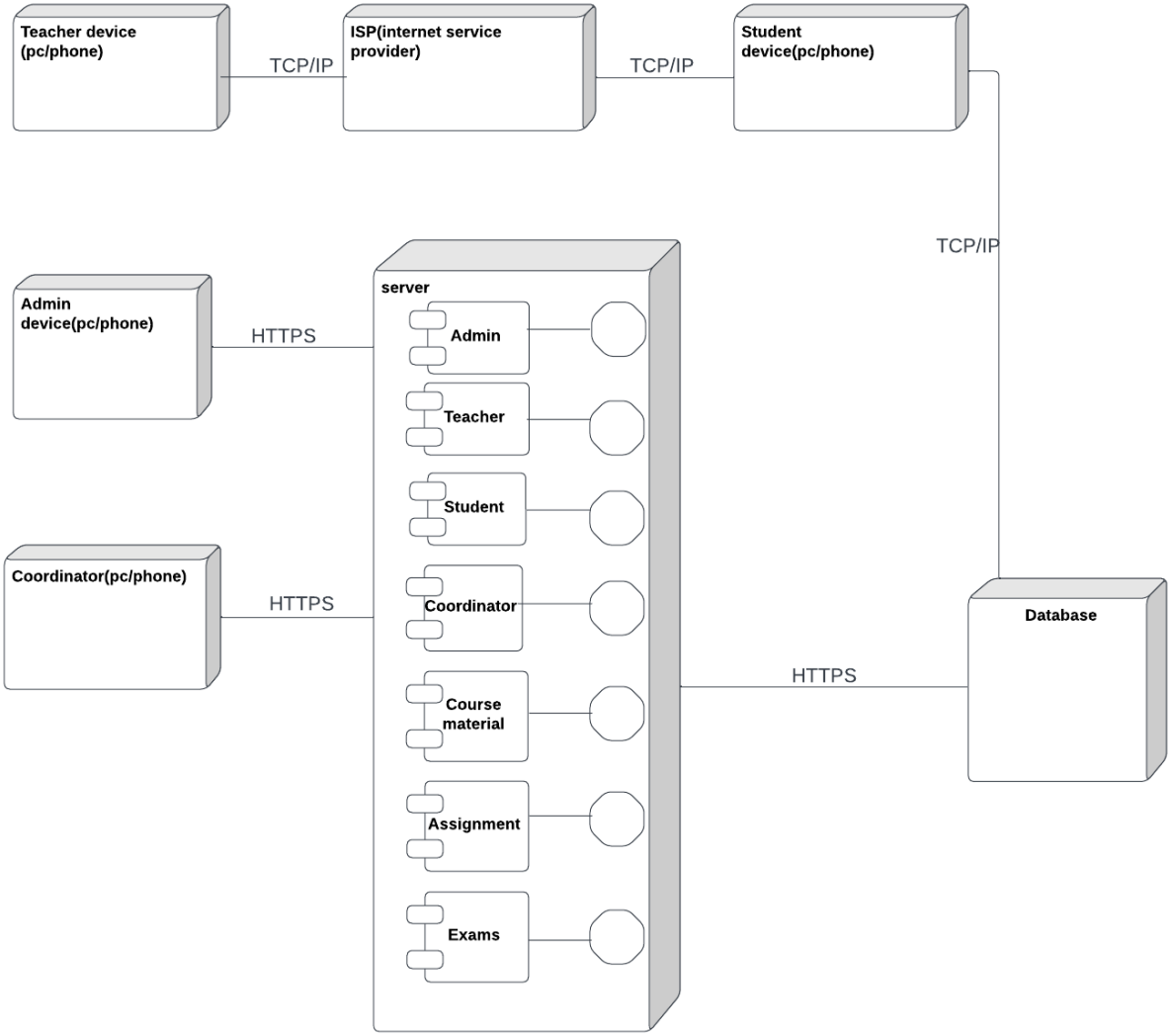
5.7 Component Diagram

Component diagrams are used to visualize the organization of system components and the dependency relationships between them. They provide a high-level view of the components within a system. It is used in Component-Based-Development to describe systems with Service-Oriented-Architecture, Show the structure of the code itself, it can be used to focus on the relationship between components while hiding specification detail help communicate and explain the functions of the system being built to stakeholders.



5.8 Deployment Diagram

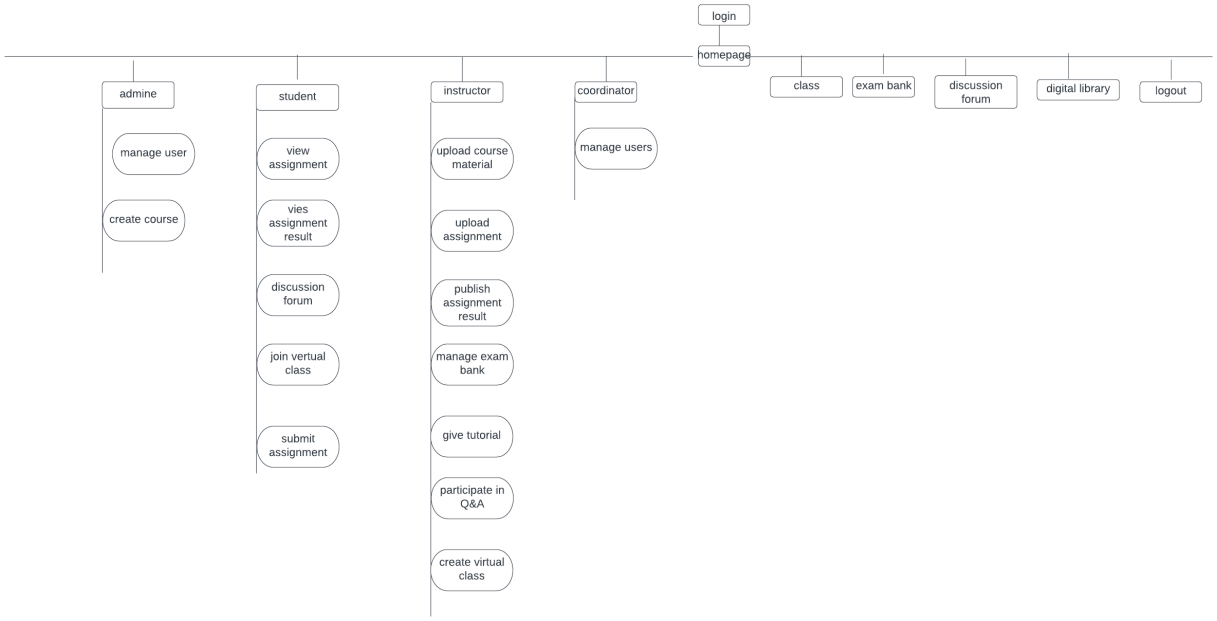
A deployment diagram "depicts the physical configuration of a software system upon hardware" (Lee 2013). It is a useful way of presenting a software project because it allows for both the client and developers to "gain an understanding of the system's distribution across physical resources from a unique overview". The Deployment diagram for Askuala Teaching and learning support system is:



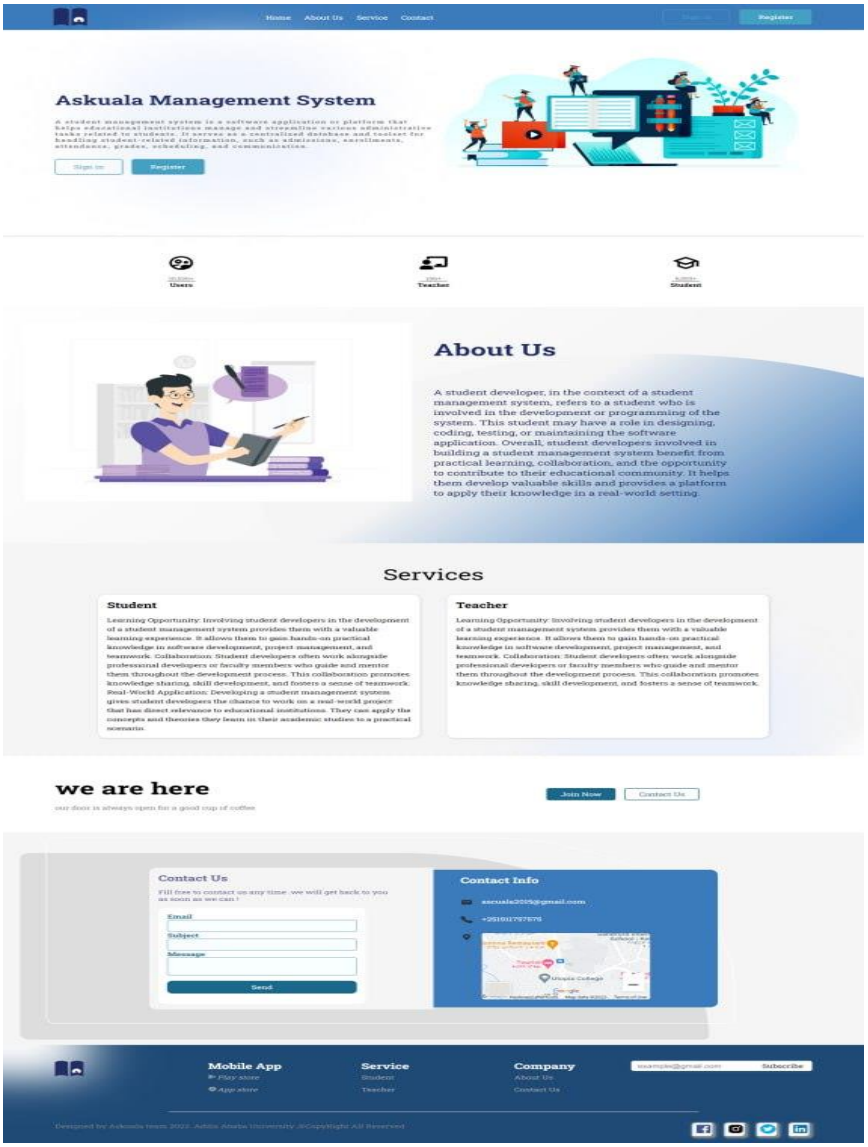
5.9User Interface

5.9.1User Interface Flow Diagram

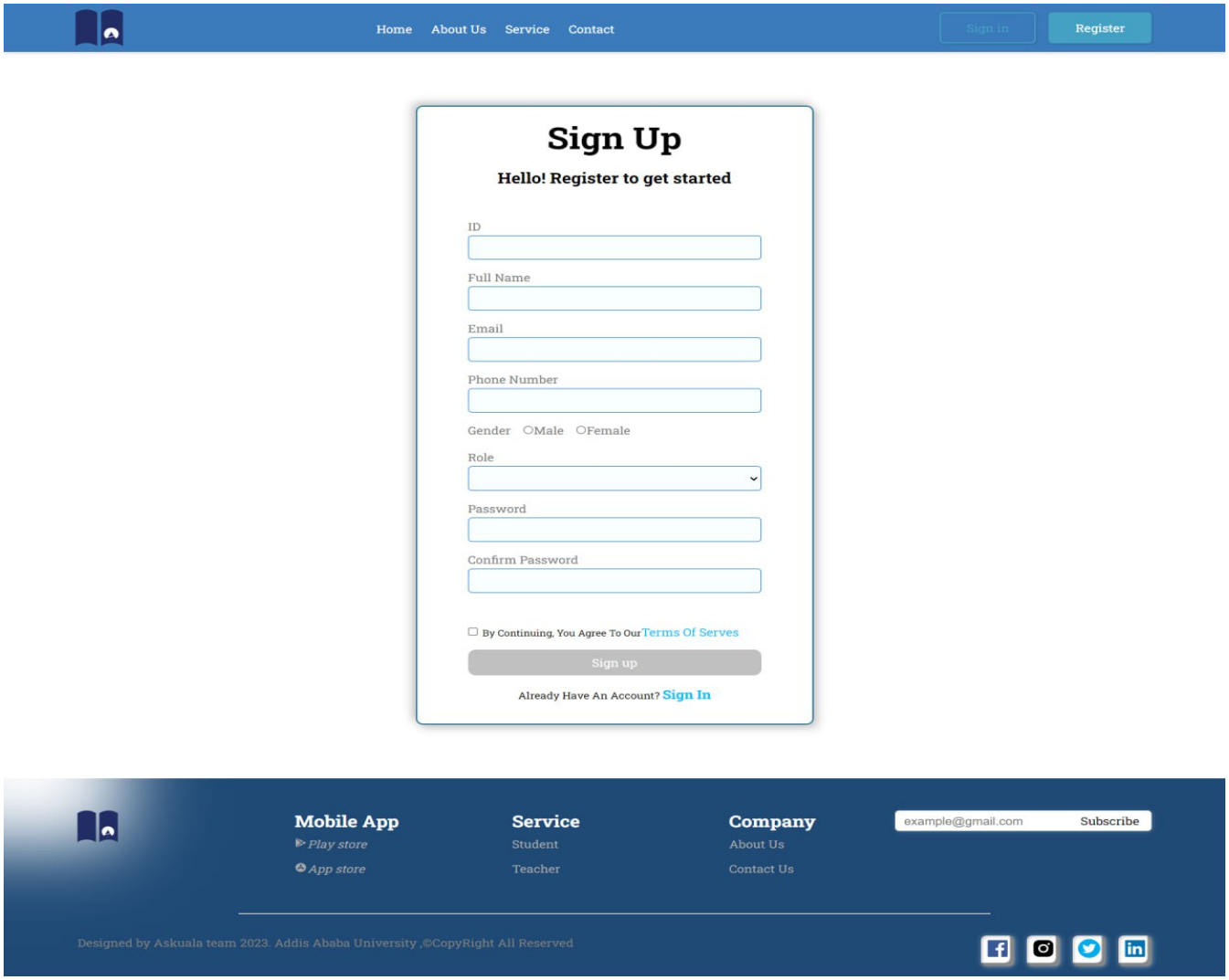
User interface flow diagram enable us to model the high level relationship between major user interface elements



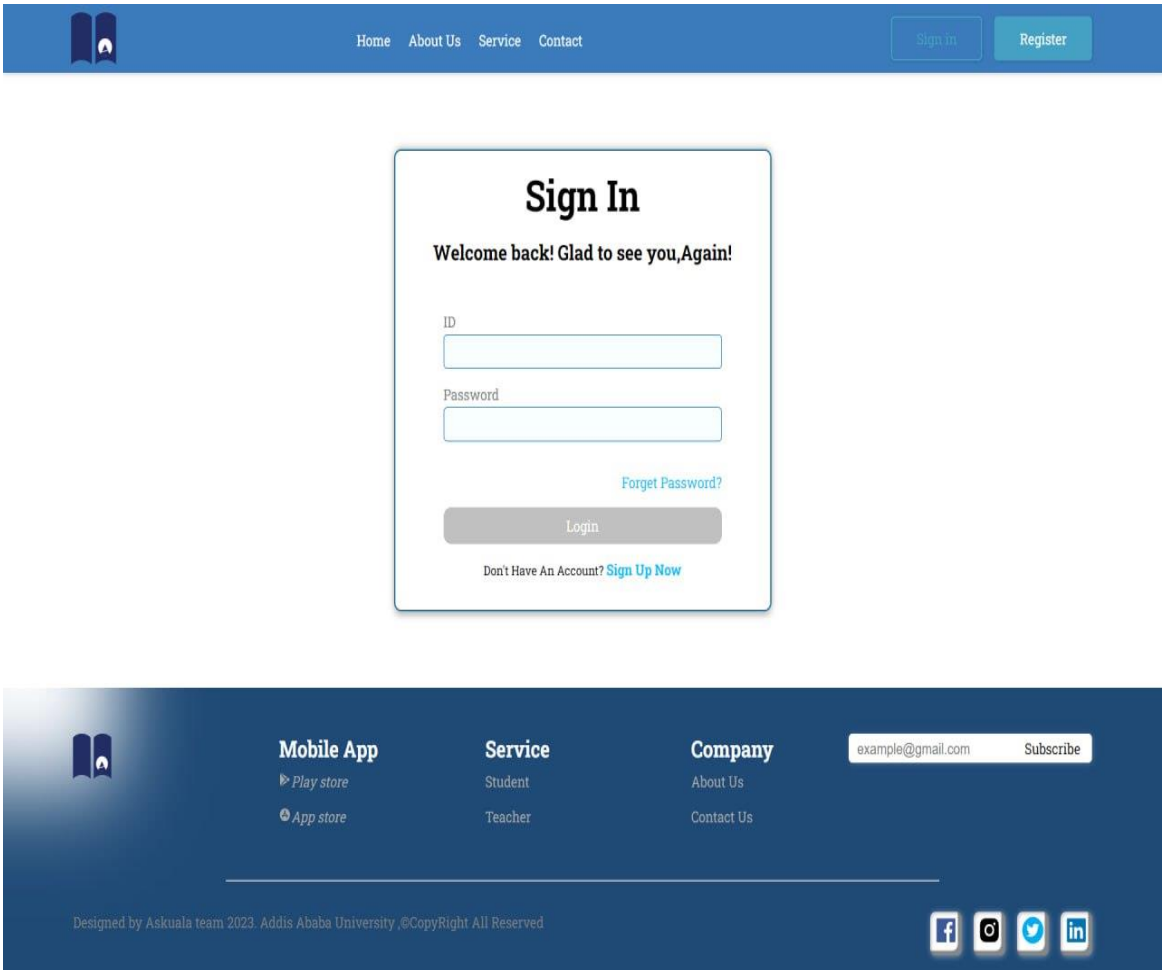
5.9.2 User Interface Design



Landing page prototype

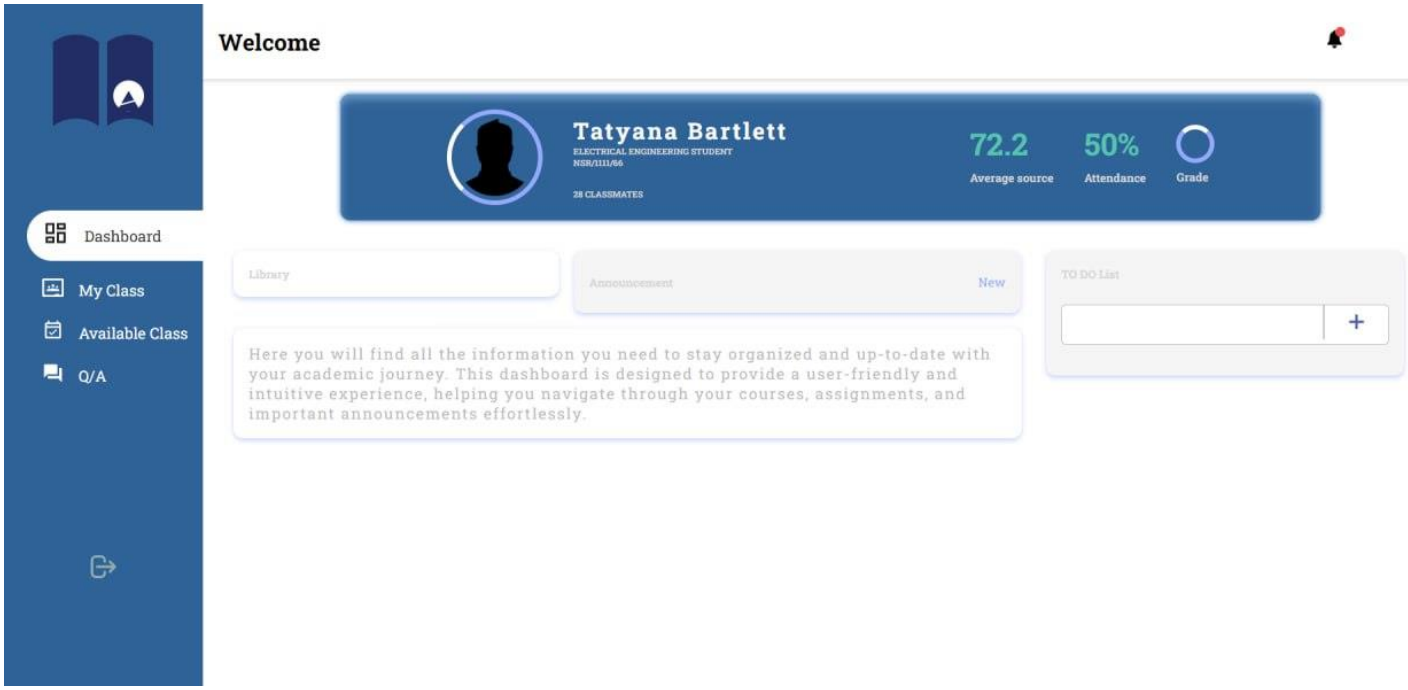


Registration page



Login page

Homepage



Landing page

My class page

Available class page

CHAPTER SIX

OBJECT ORIENTED IMPLEMENTATION

6.1 Introduction

Before a system can be used by the end user, it must be implemented and tested. Implementation is the process of translating a design or specification into a working software system and critical phase in the software development life cycle. Implementation in the system includes implementing the attributes and methods of each object and integrating all the objects in the system, to function as a single system the implementation activity spans the gap between the detailed objects designed model and a complete of source code file that can be compiled together(Ambler, 2004). Implementation in software development involves coding, testing, debugging, integration, installation, and maintenance.

In this chapter the project's implementation will be covered in detail which is implementation technology, that explains which software and hardware systems were used in the development, as well as the testing procedures, which explains how various types of testing techniques were used to ensure that the system was fault tolerant, reliable, and met all other, keeping with the object-oriented design that was shown in the section before. This rigorous approach helps the team complete the project in a timely, effective, and well-coordinated manner, in accordance with the design strategy. It also includes the deployment method that users will need in order to operate the system effectively.

The Implementation Technology

Before a system can be used by the end user, it must be implemented and tested. Implementation is the process of translating a design or specification into a working software system and critical phase in the software development life cycle. Implementation in the system includes implementing the attributes and methods of each object and integrating all the objects in the system, to function as a single system the implementation activity spans the gap between the

detailed objects designed model and a complete of source code file that can be compiled together(Ambler, 2004). Implementation in software development involves coding, testing, debugging, integration, installation, and maintenance.

In this chapter the project's implementation will be covered in detail which is implementation technology, that explains which software and hardware systems were used in the development, as well as the testing procedures, which explains how various types of testing techniques were used to ensure that the system was fault tolerant, reliable, and met all other, keeping with the object-oriented design that was shown in the section before. This rigorous approach helps the team complete the project in a timely, effective, and well-coordinated manner, in accordance with the design strategy. It also includes the deployment method that users will need in order to operate the system effectively.

The Implementation Technology

The implementation of our system is carried out using the technologies stated by the project team including the front end and back end technologies.

Front end languages that we use in our project are HTML, CSS, and JavaScript.

HTML (Hypertext Markup Language)

CSS (Cascading Style Sheets) and Bootstrap

JavaScript

React JS: React is a JavaScript library for building user interfaces. It used to build single-page applications and it allows us to create reusable UI components. React, sometimes referred to as a frontend JavaScript framework, and is a JavaScript library createdby Facebook.

Back-end technologies refer to the libraries of server-side languages that are used to create theserver configuration of a website.) Back-end technology that we use during implementation is Node.js for programming and MongoDB for database storage.

Node JS

Node.js (Node) is an open source, cross-platform runtime environment for executing JavaScript code. Node is used extensively for server-side programming, making it possiblefor developers to use JavaScript for client-side and server-side code without

needing to learn an additional language.

MongoDB

MongoDB is a document-oriented NoSQL database used for high volume data storage. Instead of using tables and rows as in the traditional relational databases, MongoDB makesuse of collections and documents. Documents consist of key-value pairs which are the basic unit of data in MongoDB. Collections contain sets of documents and function whichis the equivalent of relational database tables.

6.2 Testing and testing procedures

This chapter deals with testing and testing procedures used in this project. Testing in software development is the process of evaluating a software application or system to identify defects, errors, or bugs that may affect its functionality, performance, or usability. Testing is an essential part of the software development lifecycle, and it helps to ensure that the final product meets the requirements and expectations of the users. (Pressman, 2014) There are many different types of testing that can be used in software development, including unit testing, integration testing, system testing, acceptance testing, and regression testing. Each type of testing focuses on a specific aspect of the software application or system and uses different techniques and tools to evaluate its performance and functionality. In this section we will Unit testing, Integration testing and System testing

6.2.1 Unit Testing

Unit testing is a type of testing procedure which tries to find faults in participating objects and subsystems with respect to the use case from the use case model.(Hoffer,1999).It is a type of testing that focuses on individual components or modules of the software application.

In order to confirm and validate that each module of our system is ready for usage, we conducted this testing prior to using any other testing methodologies. This testing's goal is to ensure that each component is operating as it should. The project team modules the system since unit testing necessitates breaking the entire system down into individual modules. Each unit tested and founded to be working as intended in implemented system.

6.2.2 Integration Testing

Integration Testing is a type of software testing in which the different units, modules or components of a software application are tested as a combined entity.

Even when each module of the application is unit-tested, some errors may still exist. To identify these errors and ensure that the modules work well together after integration, integration testing is crucial.

In this phase software modules are combined and tested as a group. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements. It occurs after unit testing and before system testing. Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates, applies tests defined in an integration test plan to those aggregates, and delivers as its output the integrated system ready for system testing.

In this project Integration testing is gradual. First we test the coordinating module and only one of its subordinate modules. After the first test, we add one or two other subordinate modules from the same level. Once the program has been tested with the coordinating module and all of its immediately subordinate modules, we add modules from the next level and then test the program. We continue this procedure until the entire program has been tested as a unit.

6.2.3 System Testing

System Testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interfaced with other software/hardware systems. System Testing is defined as a series of different tests whose sole purpose is to exercise the full computer-based system.

System Testing is Blackbox

Two Category of Software Testing

- Black Box Testing
- White Box Testing

System test falls under the **black box testing** category of software testing.

- **White box testing** is the testing of the internal workings or code of a software application. In contrast, black box or System Testing is the opposite. System test involves the external workings of the software from the user's perspective.

The following testing procedure ensure the complete implementation of our system without unexpected and unnecessary errors that may occur and affect the performance of the system .So, to apply it we run the whole system and tested for errors like syntax and logical.

System Testing Sample

1. **Login Test Case**

Test case identifier	Login
Test Location	Landing page
Features to be used	Authentication, Authorization and completeness of the form by the user.
Input data	Username and Password
Test Cases	▪ Verify if a user will be able to login with valid username and valid password. ▪ Verify if a user cannot login with a valid username and invalid password. ▪ Verify the login page for both, when the field is blank and submit button is clicked.
Expectation Output	Pass
Actual Output	Pass

2. **Registration Test Case**

Test Case Identifier	Register
Test Location	Registration page
Features to be Selected	Validity and Completeness of the user's input
Input data	User account registration information.
Test Cases	▪ Verify that all required fields that are present on the registration page ▪ Verify that if a user tries to register an existing username then an error message should get displayed. ▪ Verify that if no value is passed to the fields and submit button is clicked then it leads to a validation error.
Expectation Output	User will register to the system
Actual Output	Register

3. Create Virtual Class Test Case

Test case identifier	Virtual Class
Test Location	Class page
Features to be used	Completeness of the form by Instructor
Input data	Class Creation Information
Test Cases	▪ If the system checks that the instructor information is completely entered.
Expectation Output	Virtual class will be created
Actual Output	Class create

6.3. Deployment and Installation process

Software deployment is the process of making software available for use on a system by users and other programs. Software deployment involves all the activities required to get a software system or application ready for use on a device or a server.

Software deployment process typically includes activities such as provisioning environments, installing and testing new applications, and deploying updates or patches to add new functionality or address bugs, vulnerabilities, or performance issues.

The software deployment process consists of several interrelated activities. These activities generally include provisioning environments (preparing the necessary resources for software to run, such as servers, databases, and network configurations), installing software, comprehensive testing, monitoring system health and performance, and potentially rolling back deployments.

We used 5 stages of software deployment which includes planning, design, test, schedule and deploy.

i.Planning

The first stage in a software deployment process is to make a plan. The kind of software we are trying to deploy, the number of end-users, the risks involved and other considerations are addressed here while creating plan for software deployment.

ii. Design

After making a plan for how to best approach the software deployment, design how our plan will be carried out effectively. Here we consider aspects to select a deployment type that suited our software.

iii. Test

One of the efficient ways to ensure that things don't go wrong on the deployment process is to create a test environment. Use simulations that imitate or are identical to your business's actual. This testing allows us to detect any previously hidden issues before and ensure that the software is completely functional.

iv. Schedule

Breaking the plan for software deployment into manageable-sized tasks. Then, using team members or automated software, create a schedule for when each of these tasks should be completed.

v. Deploy

The final stage is to deploy the software to your endpoints finally. Once an update has been fully tested, it can be deployed to the live environment. Developers may run a set of scripts to update relevant databases before changes can go live.

Finally, another additional installation process is providing documentation. Documentation is created to provide instructions for future installations and troubleshooting guides for any issues that may arise during deployment or use of the software.

CHAPTER SEVEN

7.1 Conclusion

As part of the project, we have gathered and analyzed different information in order to understand the inside and outside of the system clearly. In the course of gathering and analyzing requirements; there were different ups and downs.

To fully understand the challenges with the current manner of operation, we employed a range of methods and sources during requirement collecting, including documents, records, interviews, and several business analytic tools. During requirement gathering and problem identification stuff member weren't fully volunteer to give us the right and timely information about the current system and how the university conduct online learning. Arranging favorable meeting time with different IT staffs. Find the right information from the right person was one of the challenge we faced. Accessing development tools to develop the project like PC.

In conclusion, despite the difficulties we encountered and the limitations of the project, we made an effort to develop a simple and practical web application which can satisfy the need of the student, Instructor and the college and useful web application for our target purposes

7.2 Recommendation

The goal of this project was to create a web based teaching learning support system that would allow users to actively participate in the learning and teaching process. During the design and development of this system, we discovered two big opportunities that have the potential to significantly improve the system. A one-to-one chat, wherein the system user can talk with a specific person, is recommended. An examination system that enables teachers upload exams and students can answer on it and see their result, is the second modification that can be made to the system. The two recommendations above are recommended and encouraged for individuals who desire to improve the system further.

References

what-persistence-and-why-does-it-matter. (2010, October 22). Retrieved from www.datastax.com: https://www.datastax.com/blog/what-persistence-and-why-does-it-matter

searchsoftwarequality/definition/collaboration-diagram. (2023). Retrieved from www.techtarget.com/: https://www.techtarget.com/searchsoftwarequality/definition/collaboration-diagram

Awati, R. (n.d.). *integration-testing*. Retrieved from www.techtarget.com: https://www.techtarget.com/searchsoftwarequality/definition/integration-testing

component-diagram-tutorial. (n.d.). Retrieved from creately.com: https://creately.com/blog/software-teams/component-diagram-tutorial/

Integration_testing. (n.d.). Retrieved from en.wikipedia.org: https://en.wikipedia.org/wiki/Integration_testing

system-testing.html. (n.d.). Retrieved 2023, from www.guru99.com: https://www.guru99.com/system-testing.html

Pressman, R. S. (2014). Software engineering: a practitioner's approach. McGraw-Hill Education.

Scott W.Ambler(2004) The Object Primer Agile Model Driven Development with UML2 (3rd Edition). Cambrige University Press

Hoffer, M. (1999). Object-oriented program design and software engineering. Prentice Hall.

Visual Paradigm. (n.d.). What is Class Diagram? Retrieved June 12, 2023, from https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/

Appendix

Some Sample codes

//authentication

```
import {
createSlice } from
"@reduxjs/toolkit";


const initialState = {
InputLogIn: [],
InputRegister: [],
};


const auth = createSlice({
name: "auth",
initialState:initialState,
reducers: {
login(state, action) {
const newData = action.payload.data;
state.InputLogIn.push({
email: newData.email,
password: newData.password,
});
},
signup(state , action) {
const newData = action.payload.data;
state.InputRegister.push({
id: newData.id,
FullName: newData.FullName,
email: newData.email,
phoneNumber: newData.phoneNumber,
gender: newData.gender,
role: newData.role,
```

```
department: newData.department,
password: newData.password,
cPassword: newData.cPassword,
});
},
registerGet(state){}
},
});


export const { login, signup, registerGet } = auth.actions;


export default auth.reducer;
```