A Minor Project Mid-Term Report on

ELabX

Submitted in Partial Fulfillment of the Requirements for
The Degree of **Bachelor of Engineering in Computer Engineering**
Under Pokhara University

Submitted by:

Abhilekh Gautam, 191304

Aishu Gyawali, 191306

Bishal Poudel, 191318

Riya Pant, 191333

Under the supervision of

Amit Srivastava

Date: 24 Jun 2023

Department of Computer Engineering

NEPAL COLLEGE OF INFORMATION TECHNOLOGY

Balkumari, Lalitpur, Nepal

ABSTRACT

The proposed project aims to address the limitations of traditional lab classes by developing a lab management system that incorporates an online code editor, compiler, and digital course management features. This system will provide students with a sandboxed environment to experiment with code in various programming languages, allowing them to practice and check their results. Additionally, the system will offer a comprehensive digital course platform where teachers can create courses, upload topics, and add practice questions. Students will enroll in courses and work on practice questions in chronological order, with auto-grading and submission for review. The project promotes collaboration and coordination among teachers and students, streamlining the grading process and enabling progress tracking. It also extends accessibility to independent learners who can practice coding at their own pace. The development approach follows an incremental software development model, prioritizing research, design, and implementation using HTML, CSS, ReactJS, Python, REST APIs, JS, MySQL, and Webassembly technology. Supporting tools such as version control systems, Docker, and CI/CD pipeline integration will facilitate effective teamwork and documentation. The project, named ELabX, has the potential to revolutionize programming education and serve as a platform for learning and practicing programming across various languages.

Keywords: Sandbox, Webassembely, Lab, REST API, Autograding, Containerization

Table of contents

ABSTRACT.....I
Table of contents.....II
1. Introduction.....1
2. Problem Statement.....2
3. Project Objectives.....3
4. Scope and Limitations.....4
5. Literature review.....5
    5.1 Sakai LMS.....5
    5.2 Rust Playground.....6
    5.3 W3school.....6
    5.4 Programiz.....7
6. Methodology.....8
    6.2 Increment 2.....9
    6.3 Increment 3.....9
    6.4 Increment 4.....10
    6.5 Increment 5.....10
7. System Design.....11
    7.1 ER Diagram.....11
    7.2 Class Diagram.....12
    7.3 Use case Diagram.....13
    7.4 Data Flow Diagram (DFD).....14
8. Task Done so far.....15
9. Result and Discussion.....16
10. Task Remaining.....17
11. Deliverable.....18
12. Task and Time Schedule.....19
References.....20

1. Introduction

The project, ELabX is a kind of lab management system that aims to improve the quality of lab classes in the context of programming. It will feature an online code editor and a compiler to allow students to experiment with codes in different languages through sandboxing and also check the results. Moreover, it will provide features of a digital course management system with topics and practice questions related to each subject offered by an institute. Students must enroll in a course and work on practice questions in chronological order. When a question is answered correctly, it will be auto-graded as well as submitted to the teacher and other students for review. This way, it becomes easier for teachers to grade and track the progress of a student in a certain course.

This project promotes collaboration and coordination among teachers and students. A digital course platform like this makes traditional hand-written lab reports and assignments obsolete. It also encourages students to work diligently and teachers grade each student sincerely. All offered courses and content are organized neatly and may be updated easily.

When a project like this is developed on a large scale, it works as a platform for learning and practicing programming in various languages. Individuals that are not affiliated with any college or educational institute can practice coding in a systematic manner as well as per their own schedule.

We will develop the ELabX with the software engineering concept at heart, following an incremental software development model. Each increment will add an additional working feature and the complete system is expected to be finished in 5 increments. We will be paying great attention to the initial research and design phases. As for programming languages and tools, we will be using HTML, CSS, ReactJS, and Figma for the front end; Python, Django, and REST API for the back end, MySQL for database management, and Webassembly technology for sandboxing. Additional tools are version control systems(git), Github for effective teamwork and documentation, Docker for containerization, and CI/CD pipeline integration.

2. Problem Statement

The laboratory syllabus remains incomplete, hindering students' progress in their scientific education. The outdated reliance on pen-and-paper laboratory techniques further exacerbates the issue. In this rapidly evolving era of technology, where advancements in scientific research and experimentation are at the forefront, it is crucial that students are equipped with the necessary skills and knowledge to keep up with the demands of the modern world. Unfortunately, the lack of necessary materials and a conducive technical environment in the laboratory further limits students' ability to engage in hands-on practical learning experiences. Additionally, the restricted time teachers have in the laboratory prevents them from dedicating adequate attention to each student and tracking their progress individually. As a result, students may feel less accountable and become more careless, leading to ineffective performance in their laboratory work.

Our product, ELabX provides such an environment so that there is no need to write lab work on paper, and with the growing technology advancement, it provides coupling with tech so that students can complete their laboratory work within a sandboxed environment. Also, there is an interaction between the teacher and student in the form of the teacher's review of the student's work.

3. Project Objectives

- To develop a system that helps to keep track of students' laboratory progress.
- To develop a system that provides users with a sandboxed environment where they can run and compile the code relative to the course they enrolled in.

4. Scope and Limitations

- Students don't need to be physically present in the college for their lab work as they can easily complete their work from home.
- This product can be easily expanded as a college or school course management system.
- It will help in progress tracking and grading for students.

The current limitation of our project Elabx is:

- It lacks compatibility with subjects related to databases and other non-programming faculties such as civil engineering, architecture engineering, etc.

5. Literature review

5.1 Sakai LMS

The Sakai Project began as a $6.8 million community source software development project started as a partnership between four universities: Indiana University, the University of Michigan, Stanford University, and the University of Massachusetts Institute of Technology, and two well-established higher education projects in the uPortal activity and Open Knowledge Initiative (OKI). The purpose of the project was to produce open-source collaboration and Learning Environment (CLE) software by integrating and synchronizing their many educational software into a pre-integrated collection of open-source tools[1,2].

The project was first released in March 2005 using Java programming language. The system of the Sakai project can also be considered a Learning Management System (LMS), Course Management System(CMS), or Virtual Learning Environment(VLE). This open-source education software platform was designed to support Enterprise services-based portals, a complete course management system with sophisticated assessment tools, a research support collaboration system, a workflow engine, and a tool portability profile. The Sakai LMS can be downloaded and used freely from sakailms.org. After registration, it gives a beautiful dashboard where a user can edit/set up his/her profile, read and add the contents, add a group, and create a research project and a course. Hundreds of institutions use Sakai as their primary LMS, mainly in the USA but also in Canada, Europe, Asia, Africa, and Australia. Sakai provides the flexibility of customization, third-party software integration, resource sharing, online discussion, etc., and hence supports improving teaching quality[1,2].

The Sakai LMS can be useful for every subject as a general LMS, but in the case of programming subjects like C++, Data Structure and Algorithm(DSA), it fails to provide such a sandboxing environment where a user can write, compile and submit their code for evaluation.

5.2 Rust Playground

Rust playground is a sandboxed environment that allows users to experiment with Rust code. In the Rust playground a React frontend communicates with an Axum backend and docker containers are used to provide the various compilers and tools to run and execute the program. It also provides modern IDE features like syntax highlighting [4].

When a user tries to run the code, the source code from the client side is taken back to the backend of the program which has a docker container in it consisting of compilers and other tools. The source code is compiled and executed on the container and the output of the compilation is sent back to the client.

Since everything is running inside the container it has some limitations like limited compilation and execution time, and the amount of memory compilers and resulting executables can use is limited as well [4].

However, our project aims to provide a sandboxed environment that compiles and executes the source code entirely on the client side.

5.3 W3school

Similarly, there is a website called W3school that provides tutorials, examples, and references for various technologies with an online editor that enables users to experiment with code samples and view the outcomes in real-time. W3Schools also has a community forum where users may post queries, share information, and get help from other developers. A vast number of web technologies are also covered by W3Schools, including HTML, CSS, JavaScript, jQuery, PHP, SQL, and others.

Although W3Schools is a valuable resource for anyone interested in learning web development, some developers criticize it for being too simplistic and not covering some advanced topics. Additionally, it doesn't effectively monitor student achievement, and neither does it provide support for students or an assessment of their development or any certificates.

Shortcoming these limitations, our product ‘ELabX’ stays up-to-date on the latest trends and best practices by following blogs, Books, and news sources. As we collaborate with teachers and colleges/universities there LMS effectively monitors student achievement, provides support for students, and assists them by reviewing their work up to date.

5.4 Programiz

Programiz is an online platform for learning and practicing programming. It offers various tutorials covering topics such as Python, C, Java, DSA, JavaScript, C#, and mobile and web development. The content management system used by the developers to neatly organize all courses is Drupal.

One of the key features of Programiz is its interactive online code editor, which allows users to write and run code directly in their web browser, which was made possible using Docker and Kubernetes. Their mobile app “Learn Python” was developed using Flutter.

Although no official information is given, it is likely that the team used an iterative and incremental development approach. It involves breaking down a project into smaller, more manageable parts and completing each part through a series of iterations. It also allows for testing and feedback to be incorporated throughout the development process.

6. Methodology

Software Development Life Cycle(SDLC) provides a method for building and delivering software projects. There are various stages of SDLC such as communication, planning, design, implementation, and deployment. There are various software development models like waterfall, incremental, spiral, and prototyping, which are based on the framework provided by SDLC. Each model has their usage and criteria. We found an Incremental model also known as Rapid Application Development(RAD) is best suited for our project. This model combines elements of the waterfall model applied in an iterative fashion [3].

In our project there will be five increments, each increment will traverse through the stages of SDLC. Our end product is somehow clear but there are many requirements/features that we have to add or modify over time. In each increment, we will develop a working product and add new features to it. The incremental model provides such flexibility for software development, and the functionality in increments of our product will be something like this:

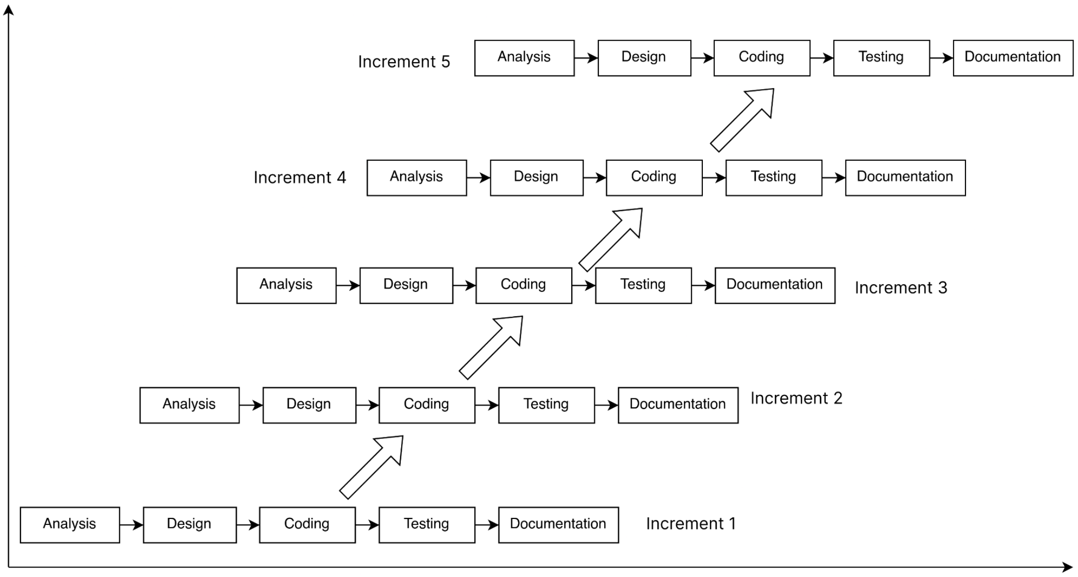


Figure 1. Incremental software development model

6.1 Increment 1

- Gather requirements of the system and review existing system
- Design high-level system flowchart, use-cases, and UI for code editor as well as other pages
- Design architecture for sandboxing, and overall flow of code in wasm module
- Create a wasm module for compiling and running the code in the sandbox, with proper error handling
- Develop client side for above UI and integrate sandbox
- Test the environment
- Writing proposal

6.2 Increment 2

- Analysis DBMS for courses and contents for C++
- Design schema diagram for courses and UI for a single course
- Develop database integration for the course.
- Develop APIs for CRUD course contents
- Containerize the project
- Write test cases for each and every question
- Searching for the course (*searching algorithm)
- Develop front-end interface for the course and integrate with code editor
- Testing the API endpoints

6.3 Increment 3

- Analysis requirement of user and types
- Design database for users and relationship with a course, UI for user views
- APIs for different user authentication like a student, teacher
- Frontend for login and registration
- Validation and verification (*algorithm)
- Testing APIs and integration with client-side

- Documenting the system

6.4 Increment 4

- Integration of auto-grading
- Tracking the progress of the student (*algorithm)
- Self-enrollment
- Review the submitted solution by the teacher, and student
- Designing and implementing above task in the client and server side

6.5 Increment 5

- Analysis about the new courses and appropriate CI/CD tools
- Design whole system workflow
- Extend for more courses
- Implement CI/CD along with containerization
- Testing for final phase
- Document and prepare final report
- Improving the product by adding more increments, if needed

7. System Design

7.1 ER Diagram

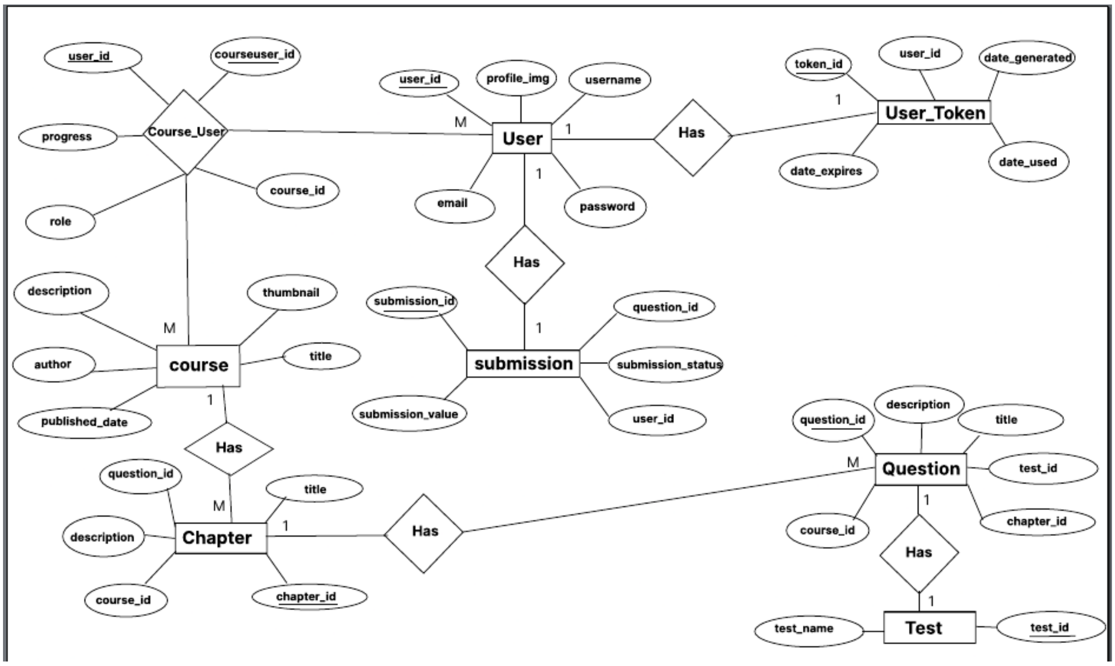


Figure 2. ER diagram for ELabX

7.2 Class Diagram

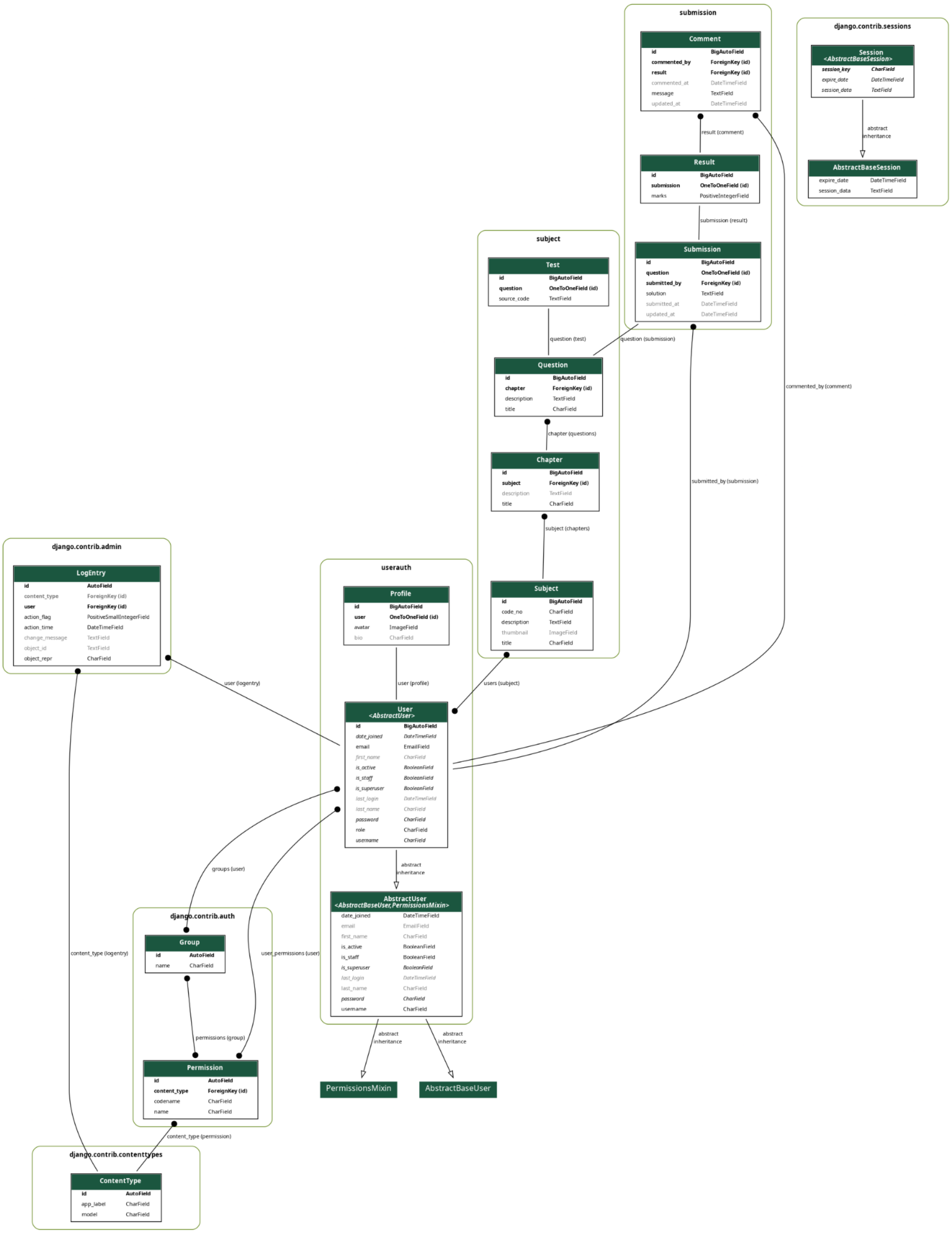


Figure 3. ELabX class diagram

7.3 Use case Diagram

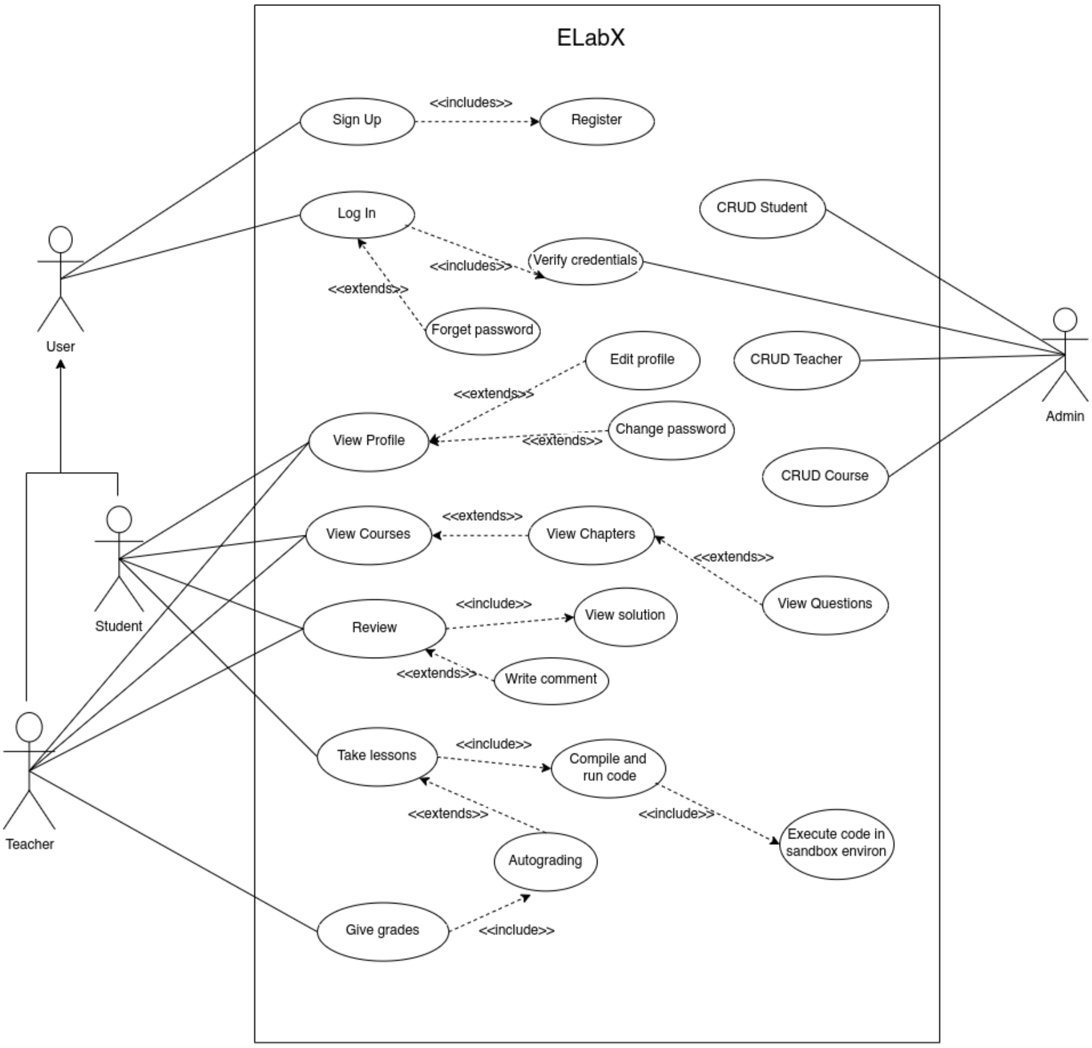


Figure 4. Use case diagram for ELabX

7.4 Data Flow Diagram (DFD)

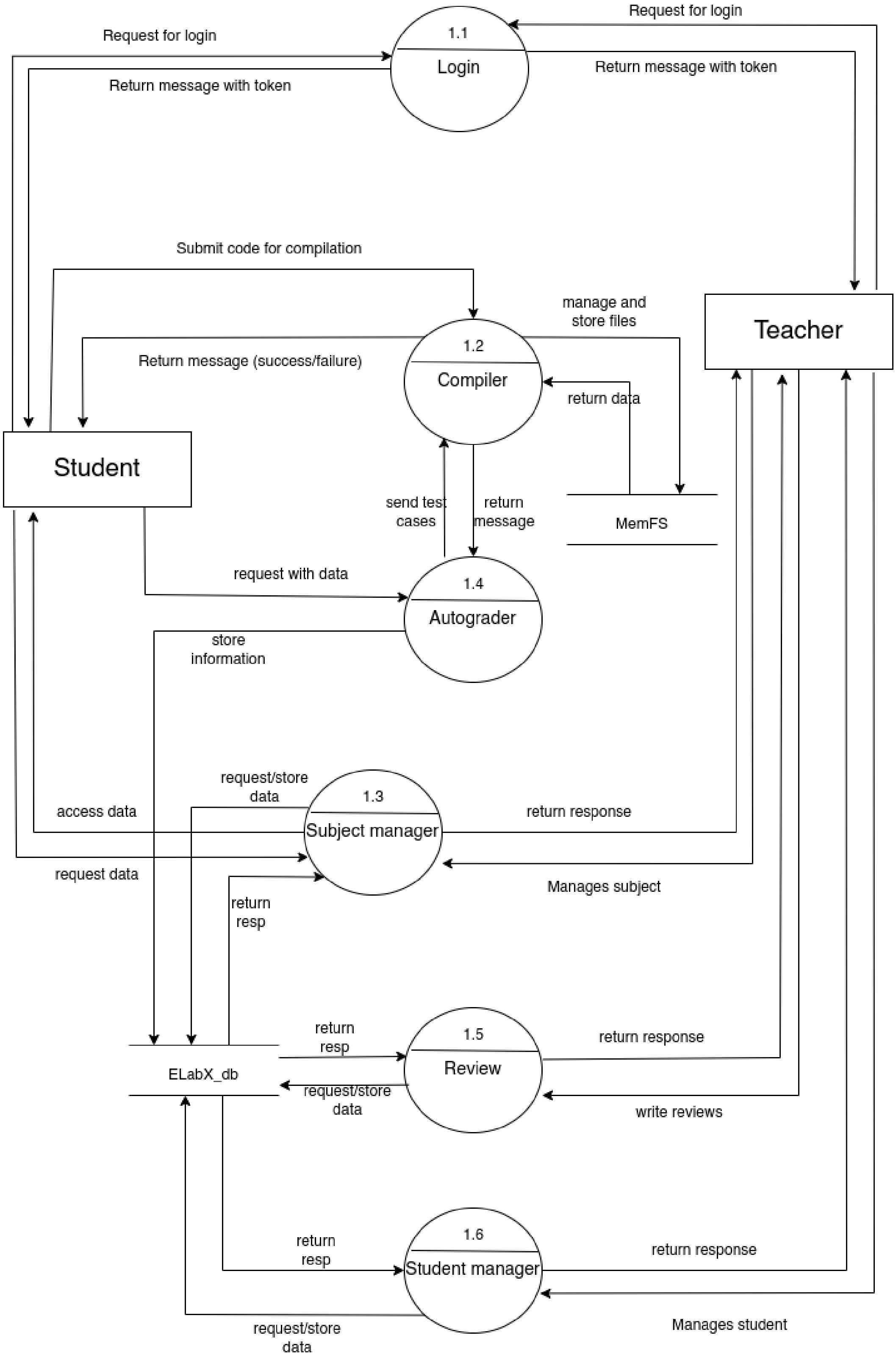


Figure 5. DFD for ELabX

8. Task Done so far

- System design and architecture
- Sandbox environment for compiling C++ code
- User interface for course management and code editor
- Database design and implementation
- CRUD APIs for course contents and user authentication
- test cases for C++ questions
- API testing for course management
- client and server integration
- Containerization

9. Result and Discussion

The result of the proposed project, ELabX, is a comprehensive lab management system that successfully integrates an online code editor, compiler, and digital course management features. We are implementing the project using HTML, CSS, ReactJS, Python, Django, REST API, and Webassembly, ensuring a robust and efficient system. The project is being developed following an incremental software development model, with each increment adding additional working features to the system.

The development of ELabX course management features brings several benefits to the field of programming education. By providing an online environment for students to experiment with code and check results, the project enhances the quality of lab classes and fosters practical programming skills. Students can actively engage in coding exercises and gain a better understanding of programming concepts.

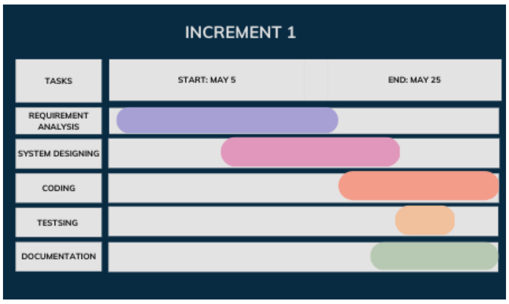
10. Task Remaining

- Autograding and Progress Tracking
- User interface for login, registration, user profile, and product landing page
- Implementation of searching algorithm
- Self-enrollment and Review section
- Integration testing and deployment using CI/CD pipelining

11. Deliverable

ELabX will be an online platform (ie. web application) that will digitize the lab system by providing a centralized platform for managing and delivering digital learning resources and tools. It provides a sandbox environment for compiling and running students’ code within the browser.

12. Task and Time Schedule



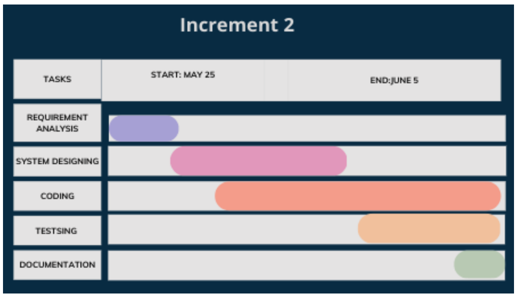
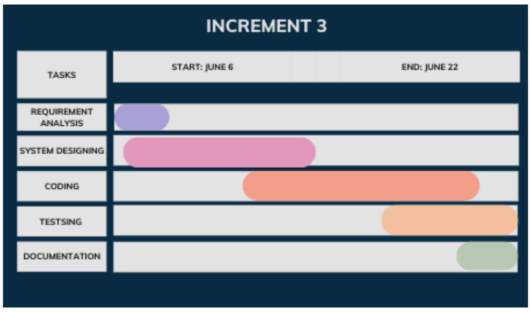


Fig 3.1 Gantt chart for increment 1

Fig 3.2 Gantt chart for increment 2



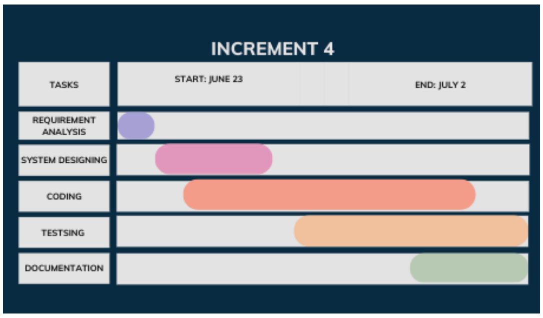


Fig 3.3 Gantt chart for increment 3

Fig 3.4 Gantt chart for increment 4

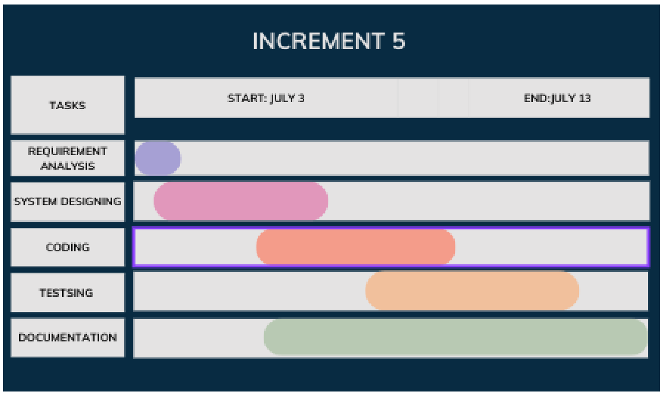


Fig 3.5 Gantt chart for increment 5

References

[1] Jovanovic S, Ignjatovic M. Implementing Sakai OAE. International Journal of Emerging Technologies in Learning (Online). 2013;8(1):64.

[2] Farmer J, Dolphin I. Sakai: eLearning and more. EUNIS 2005-Leadership and Strategy in a Cyber-Infrastructure World. 2005 Jun.

[3] Salve SM, Samreen SN, Khatri-Valmik N. A Comparative Study on Software Development Life Cycle Models. International Research Journal of Engineering and Technology (IRJET). 2018 Feb;5(2):696-700.

[4] Rust-Lang, "Rust-Lang/rust-playground: The rust playground," GitHub, https://github.com/rust-lang/rust-playground (accessed Jun. 5, 2023).