****

A Project Report On

**Tech School – An E-learning Platform**

Course Code: SWE 431(Thesis/Project)

Submission Date: 13-12-2023

**Submitted By**

Prosenjit Kumar Pal

Id: 172-35-2201

Department of Software Engineering, FSIT

Daffodil International University

**Supervised By**

Mr. A.H.M Shahariar Parvez

Associate Professor

Department of Software Engineering, FSIT

Daffodil International University

This Project report has been submitted in fulfillment of the requirements for the degree, Bachelor of Science in Software Engineering

**© All right Reserved by Daffodil International University**

**Acknowledgement**

First and foremost, I extend my deepest gratitude to The Almighty God for granting me the strength and guidance to successfully complete this project. Next, I would like to express my sincere appreciation to my supervisor, Mr. A.H.M Shahariar Parvez, Associate Professor in the Department of Software Engineering at Daffodil International University. His expertise, unwavering support, and invaluable guidance played a pivotal role in shaping this project. I am also profoundly thankful to Dr. Imran Mahmud, Associate Professor and Head of the Software Engineering Department, for his continuous encouragement and support throughout this journey. My heartfelt thanks extend to all the esteemed faculty members of the Software Engineering Department for their assistance and encouragement. Lastly, but certainly not least, I want to convey my deepest appreciation to my parents for their unwavering love, support, and care, which have been the pillars of my strength.

Chapter – 1

Introduction

* 1. **Project Overview**

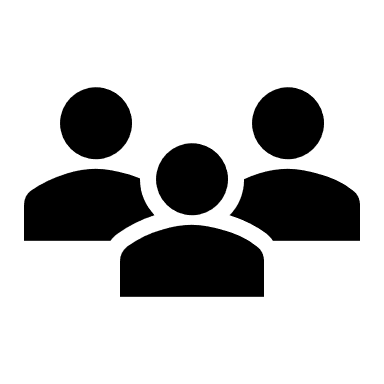
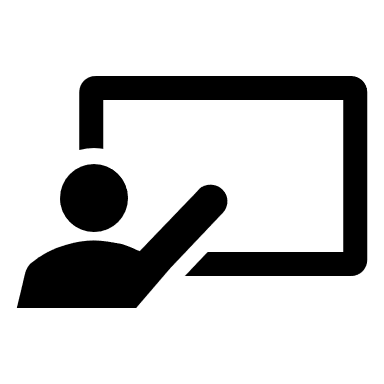
In the rapidly evolving landscape of technology and education, "Tech School" emerges as a visionary project poised to redefine the learning experience. Tech School is an innovative platform designed to provide a comprehensive and interactive educational environment, fostering the growth and development of individuals in the realm of technology and beyond. This initiative is driven by a commitment to bridging the gap between theoretical knowledge and practical application, preparing learners to thrive in a dynamic and tech-centric world.

* 1. Project Purpose

The mission of Tech School is to empower individuals with the knowledge and skills needed to excel in the fast-paced world of technology. By combining educational excellence with a practical, hands-on approach, Tech School aspires to be a catalyst for personal and professional growth, creating a community of tech-savvy individuals poised to make a significant impact in their respective fields.

1.2.1 Proposed System Model

This model will help to simulate the system architecture. It is tried to design it as user friendly as possible.



**Instructors**

* **Manage Course Material**
* **Write Blog**

**Learners**

* **Enroll Course**
* **Give Feedback**
* **Manage Cart**
* **Read Blog**



**Tech School**



**Admin**

* **Mange Users**
* **Manage Courses**
* **Manage Blog**

1.3 Benefits and Beneficiaries

Tech School emerges as a transformative force in education, delivering a myriad of benefits to a diverse range of beneficiaries. For learners, the platform serves as an empowering gateway to acquire relevant and practical tech skills, fostering not only professional growth but also a commitment to lifelong learning. Instructors find a platform to contribute their expertise, shaping the next generation of tech professionals while enhancing their own professional development. The project's commitment to accessible tech education extends its benefits globally, reaching learners irrespective of geographical constraints. The collaborative community cultivated by Tech School enriches the learning experience for individuals seeking interaction, shared insights, and project collaboration. As learners worldwide access quality education, the platform becomes a catalyst for innovation, contributing to the ongoing advancement of the tech industry and benefiting businesses and society at large. Tech School, with its inclusive and innovative approach, stands as a beacon of opportunity for tech enthusiasts, professionals, and the broader global community.

**1.4 Problem Statement**

In today's digital age, learning is taking place online more than ever before. But there's a challenge: existing online education systems are often complex and not user-friendly. Our website aims to solve this problem. We want to create an easy-to-use online learning platform for Learners, Instructors, and administrators. Our goal is to make learning online simple, engaging, and accessible to everyone. We believe that education should be interactive and enjoyable. With our project, we hope to provide a better way for people to access knowledge and connect with others in a global community of learners. Our mission is to make online learning a fun and enriching experience for all, bringing education into the digital age.

1.5 Project Schedule

A project schedule indicates what feature needs to be done, which resources must be utilized, and when the project is not completed. Assuming short time to develop the system, it is required to make a good scheduling to finish the project on time. This schedule also refers to communicate with what tasks is needed to be done within short time period.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Activities | W1 | W2 | W3 | W4 | W5 | W6 | W7 | W8 | W9 | W10 | W11 | W12 | W13 |
| Research Req. |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Analysis |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Planning |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Designing |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Development |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Assessment |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |

Table 1.1 – Project Gantt Chart

Chapter – 2

System Design

**2.1** Feasibility Analysis

1. Technical Feasibility:

Existing Technology: The project leverages common web development technologies, ensuring compatibility with standard browsers and devices.

Development Expertise: As the sole developer, the necessary technical skills are available to design, develop, and maintain the E-Learning Management System.

Scalability: The chosen technologies allow for scalable development and deployment, ensuring the system's capability to handle future growth.

2. Operational Feasibility:

User Acceptance: Regular feedback sessions and demos with potential users have been conducted, ensuring that the system aligns with their needs and expectations.

Training Requirements: Minimal training is anticipated for users due to the intuitive design and user-friendly interface.

3 Legal and Ethical Feasibility:

Compliance: The system is designed to comply with data protection regulations, ensuring user privacy and adherence to legal standards.

Ethical Considerations: The project aligns with ethical standards in education, promoting inclusivity, accessibility, and fair treatment of users.

4. Social Feasibility:

User Engagement: Stakeholder involvement and regular communication with potential users ensure that the system meets their social and educational needs.

Adaptation to User Feedback: The Agile approach allows for the incorporation of user feedback, fostering a sense of user involvement and ownership.

**2.2 Requirement specification**

**2.2.1 Functional Requirement**

Functional requirements (FRs) provide an explanation of the tasks or functions that a system or software application should perform. In essence they outline the characteristics, behaviors and connections that the system needs to demonstrate in order to fulfill the requirements and meet the desires of its users or stakeholders.

Functional requirements are mandatory for the any system. For this system functional requirements are mentioned below.

|  |  |
| --- | --- |
| FR-01 | Registration |
| Description | Users must have the ability to register for the system by providing their name, email, and password. |
| Stakeholders | All users (Learners, Instructors, admins) |
| Priority | High |

|  |  |
| --- | --- |
| FR-02 | Login |
| Description | Registered users should be able to log in securely using their username and password. |
| Stakeholders | All users (Learners, Instructors, admins) |
| Priority | High |

|  |  |
| --- | --- |
| FR-03 | Course Management |
| Description | Only an admin able to create new courses by providing a course title, description, and other relevant information. And also, can edit existing courses, and delete courses from the system. |
| Stakeholders | Admin |
| Priority | High |

|  |  |
| --- | --- |
| FR-04 | Course Enrollment |
| Description | Learners are able to browse available courses and enroll in them. |
| Stakeholders | Learners |
| Priority | High |

|  |  |
| --- | --- |
| FR-05 | Payment Processing |
| Description | The system supports payment processing for course fees. |
| Stakeholders | Learners |
| Priority | High |

|  |  |
| --- | --- |
| FR-06 | Blog Post Creation and Mange |
| Description | Instructors should have full control over their blog posts, including the ability to create new blogs, edit the content and title of their existing blogs, and delete any of their posted blogs. |
| Stakeholders | Instructors |
| Priority | High |

|  |  |
| --- | --- |
| FR-07 | Blog Post Reading |
| Description | All users should be able to read blog posts created by Instructors. |
| Stakeholders | All users (Learners, Instructors, admins) |
| Priority | High |

|  |  |
| --- | --- |
| FR-08 | Provide Course Feedback |
| Description | Learners should have the capability to provide feedback on courses in which they are enrolled. |
| Stakeholders | Learners |
| Priority | Medium |

|  |  |
| --- | --- |
| FR-09 | Course Feedback Viewing |
| Description | all users should be able to view and access feedback provided by learners for a particular course. |
| Stakeholders | All users (Learners, Instructors, admins) |
| Priority | Medium |

|  |  |
| --- | --- |
| FR-10 | Course Material Management |
| Description | Instructor should have the capability to add course materials to their courses. Additionally, Instructors should be able to edit and delete the course materials they have created. |
| Stakeholders | Instructors |
| Priority | High |

|  |  |
| --- | --- |
| FR-11 | Course Material Viewing |
| Description | Learners should be able to view the course materials for courses in which they are enrolled. |
| Stakeholders | Learners |
| Priority | High |

|  |  |
| --- | --- |
| FR-12 | User Role Management |
| Description | When a new user completes registration for the system, their default role should be set as "Learner." Admins should have the authority to update user roles, changing them from "Learner" to "Instructor" as needed. |
| Stakeholders | Admin |
| Priority | Medium |

|  |  |
| --- | --- |
| FR-13 | User Removal |
| Description | Admins should have the authority to remove any user from the system as needed. This action permanently deletes the user's account and associated data from the system. |
| Stakeholders | Admin |
| Priority | High |

|  |  |
| --- | --- |
| FR-14 | User Profile Editing |
| Description | Users should be able to edit their profiles and update personal information, including email and password. |
| Stakeholders | All users (Learners, Instructors, admins) |
| Priority | Medium |

|  |  |
| --- | --- |
| FR-15 | User Logout |
| Description | Users, including Learners, Instructors, and admins, should have the ability to log out securely from the system. |
| Stakeholders | All users (Learners, Instructors, admins) |
| Priority | Medium |

**2.2.2 Non-functional Requirement**

|  |  |
| --- | --- |
| NFR-01 | Performance |
| Description | The system should provide a responsive and fast user experience, with page load times not exceeding [specified time limit] even with a large number of concurrent users. |
| Stakeholders | All Users (Learners, Instructors, admins) |
| Priority | High |

|  |  |
| --- | --- |
| NFR-02 | Security |
| Description | User data, including personal information and payment details, should be securely stored and transmitted using encryption and other appropriate security measures. Authentication and authorization mechanisms should prevent unauthorized access. |
| Stakeholders | All Users |
| Priority | High |

|  |  |
| --- | --- |
| NFR-03 | Usability |
| Description | The user interface should be intuitive and user-friendly, providing a positive user experience. The system should be accessible to users with disabilities, adhering to relevant accessibility standards. |
| Stakeholders | All Users |
| Priority | Medium |

|  |  |
| --- | --- |
| NFR-04 | Reliability |
| Description | The system should be available 24/7, with minimal downtime for maintenance. Backup and recovery mechanisms should be in place to prevent data loss and ensure system reliability. |
| Stakeholders | All Users |
| Priority | High |

|  |  |
| --- | --- |
| NFR-05 | Documentation |
| Description | Comprehensive documentation should be available for users, administrators, and developers, providing clear instructions on system usage, configuration, and maintenance. |
| Stakeholders | All Users |
| Priority | Medium |

**Chapter – 3**

**System Design**

3.1 Development Model

The development of the E-Learning Management System was executed following the Agile Software Development Life Cycle (SDLC) model. As the sole developer of this project, the Agile approach offered a flexible and iterative framework that perfectly suited the dynamic nature of the web development landscape.

The Agile Software Development Life Cycle (SDLC) model offers several advantages, particularly for individual developers. Its iterative and incremental approach allows for flexibility and adaptability, enabling the incorporation of changes even late in the development process. Regular feedback loops and continuous collaboration with stakeholders ensure that the product aligns closely with user expectations. Agile promotes transparency through frequent demonstrations, enhancing communication and trust. Additionally, the emphasis on delivering working software in short iterations allows for early and incremental value delivery, reducing the time to market and enhancing overall project visibility and control.

3.2 Use Case Diagram

Use Case Diagram for “Tech School”

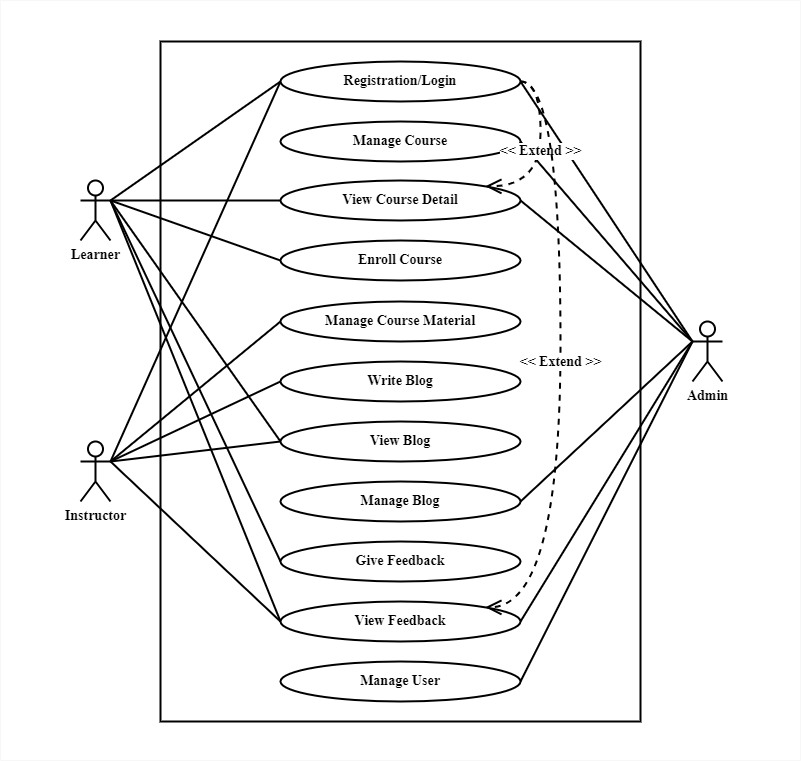


Figure 3.1 – Use Case Diagram for “Tech School”

3.2.1 **Use Case Description**

**Use Case Description for Registration/Login**

|  |  |
| --- | --- |
| Use Case ID | UC-01 |
| Use Case Name | Registration/Login |
| Actor | All users (Learners, Instructors, admins) |
| Description | Users (learners, instructors, admins) can register for the system and log in to access their accounts. |
| Precondition | User is not logged in. |
| Trigger | User accesses the system. |
| Flow of Events | * User selects the "Register" option. * User provides registration details (name, email, username, password). * User submits the registration form. * User receives a confirmation email. * User selects the "Login" option. * User enters their username and password. * User clicks "Login." |
| Postcondition | User is logged in and can access their account. |

**Use Case Description for Manage Course**

|  |  |
| --- | --- |
| Use Case ID | UC-02 |
| Use Case Name | Manage Course |
| Actor | Admin |
| Description | Admins can create, edit, and delete courses in the system. |
| Precondition | Admin is logged in. |
| Trigger | Admin selects the course management option. |
| Flow of Events | * Admin selects "Create Course" to add a new course. * Admin enters course details (title, description, instructor). * Admin clicks "Create." * Admin selects an existing course to edit. * Admin modifies course details. * Admin clicks "Save Changes." * Admin selects a course to delete. * Admin confirms deletion. |
| Postcondition | Course management is updated according to admin actions. |

**Use Case Description for View Course Detail**

|  |  |
| --- | --- |
| Use Case ID | UC-03 |
| Use Case Name | View Course Detail |
| Actor | All Users |
| Description | All users can view details of a course, including its title, description, and instructor. |
| Precondition | User is logged in. |
| Trigger | User selects a course to view. |
| Flow of Events | * User navigates to the course details page. * User selects a course to view. |
| Postcondition | User can see the details of the selected course. |

**Use Case Description for Manage Course Material**

|  |  |
| --- | --- |
| Use Case ID | UC-04 |
| Use Case Name | Manage Course Material |
| Actor | Instructor |
| Description | Instructors can add, edit, and delete course materials for the courses they teach. |
| Precondition | Instructor is logged in. |
| Trigger | Instructor selects the course material management option. |
| Flow of Events | * Instructor selects "Add Material" for a course. * Instructor uploads course material (documents, videos, etc.). * Instructor clicks "Add." * Instructor selects an existing material to edit. * Instructor modifies material details. * Instructor clicks "Save Changes." * Instructor selects a material to delete. * Instructor confirms deletion. |
| Postcondition | Course materials are updated according to instructor actions. |

**Use Case Description for Enroll Course with Payment**

|  |  |
| --- | --- |
| Use Case ID | UC-05 |
| Use Case Name | Enroll Course with Payment |
| Actor | Learner |
| Description | Learners can enroll in courses by making payments for the chosen courses. |
| Precondition | Learner is logged in. |
| Trigger | Learner selects a course to enroll in. |
| Flow of Events | * Learner selects a course to enroll in. * Learner proceeds to payment. * Learner provides payment details and confirms payment. |
| Postcondition | Learner is enrolled in the selected course upon successful payment. |

**Use Case Description for Write Blog**

|  |  |
| --- | --- |
| Use Case ID | UC-06 |
| Use Case Name | Write Blog |
| Actor | Instructor |
| Description | Instructors can create and publish blog posts. |
| Precondition | Instructor is logged in. |
| Trigger | Instructor selects the "Write Blog" option. |
| Flow of Events | * Instructor enters blog post content (title, content). * Instructor clicks "Publish." |
| Postcondition | The blog post is published and visible to all users. |

**Use Case Description for View Blog**

|  |  |
| --- | --- |
| Use Case ID | UC-07 |
| Use Case Name | View Blog |
| Actor | All Users |
| Description | All users can view blog posts created by instructors. |
| Precondition | User is logged in. |
| Trigger | User selects a blog post to view. |
| Flow of Events | * User navigates to the blog post. * User selects a blog post to view. |
| Postcondition | User can see the content of the selected blog post. |

**Use Case Description for Manage Blog**

|  |  |
| --- | --- |
| Use Case ID | UC-08 |
| Use Case Name | Manage Blog |
| Actor | Admin |
| Description | Admins can edit and delete blog posts created by instructors. |
| Precondition | Admin is logged in. |
| Trigger | Admin selects the blog management option. |
| Flow of Events | * Admin selects a blog post to edit. * Admin modifies the blog post content. * Admin clicks "Save Changes." * Admin selects a blog post to delete. * Admin confirms deletion. |
| Postcondition | Blog posts are updated or removed according to admin actions. |

**Use Case Description for Give Feedback**

|  |  |
| --- | --- |
| Use Case ID | UC-09 |
| Use Case Name | Give Feedback |
| Actor | Learner |
| Description | Learners can provide feedback on courses in which they are enrolled. |
| Precondition | Learner is logged in and enrolled in at least one course. |
| Trigger | Learner selects a course to provide feedback. |
| Flow of Events | * Learner selects a course to provide feedback. * Learner enters feedback comments and ratings (if applicable). * Learner submits the feedback. |
| Postcondition | Learner's feedback is recorded for the selected course. |

**Use Case Description for View Feedback**

|  |  |
| --- | --- |
| Use Case ID | UC-10 |
| Use Case Name | View Feedback |
| Actor | All Users |
| Description | All users can view feedback provided by learners for specific courses. |
| Precondition | User is logged in. |
| Trigger | User selects a course to view feedback. |
| Flow of Events | * User navigates to the course feedback section. * User selects a course to view feedback. |
| Postcondition | User can see feedback comments and ratings for the selected course. |

**Use Case Description for Manage User**

|  |  |
| --- | --- |
| Use Case ID | UC-11 |
| Use Case Name | Manage User |
| Actor | Admin |
| Description | Admins can manage user accounts, including creating, editing, and deleting user profiles. |
| Precondition | Admin is logged in. |
| Trigger | Admin selects the user management option. |
| Flow of Events | * Admin selects "Create User" to add a new user. * Admin enters user details (name, email, role). * Admin clicks "Create." * Admin selects an existing user to edit. * Admin modifies user details. * Admin clicks "Save Changes." * Admin selects a user to delete. * Admin confirms deletion. |
| Postcondition | User management is updated according to admin actions. |

3.2 Activity Diagram

Activity Diagram for Registration/Login

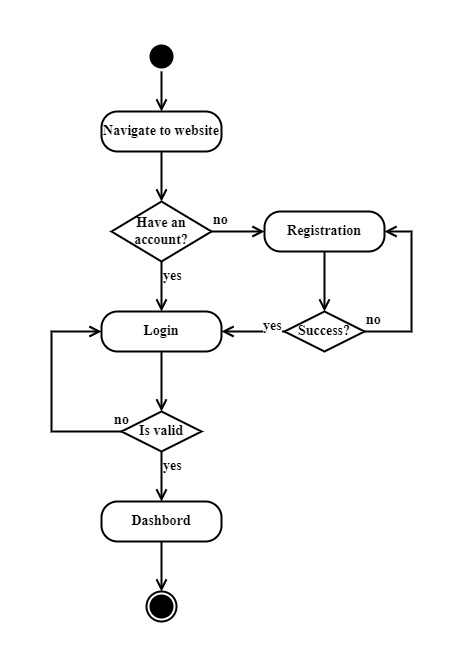


Figure 3.2 - Activity Diagram for Registration/Login

Activity Diagram for Manage Course

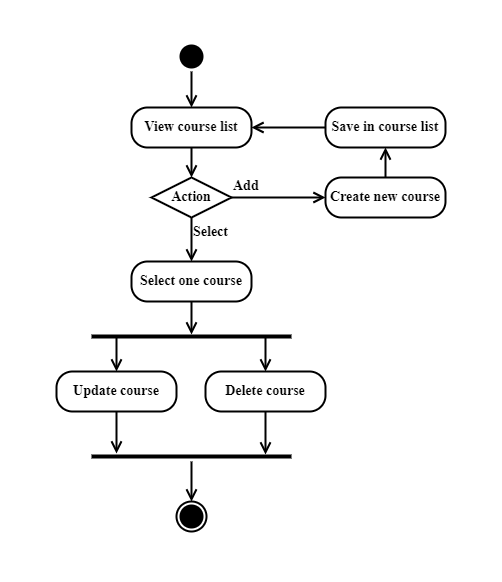


Figure 3.3 - Activity Diagram for Manage Course

Activity Diagram for View Course Detail

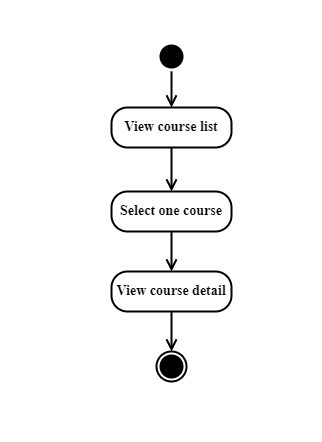


Figure 3.4 – Activity Diagram for View Course Detail

Activity Diagram for Enroll Course

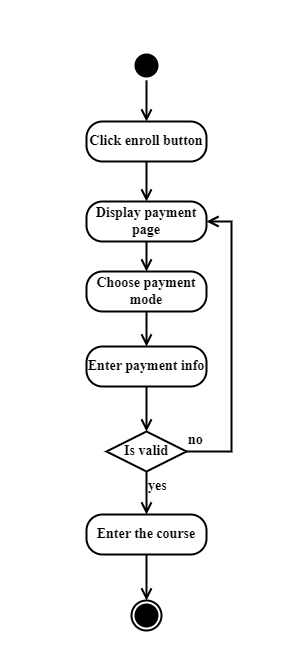


Figure 3.5 - Activity Diagram for Enroll Course

Activity Diagram for Manage Course Material

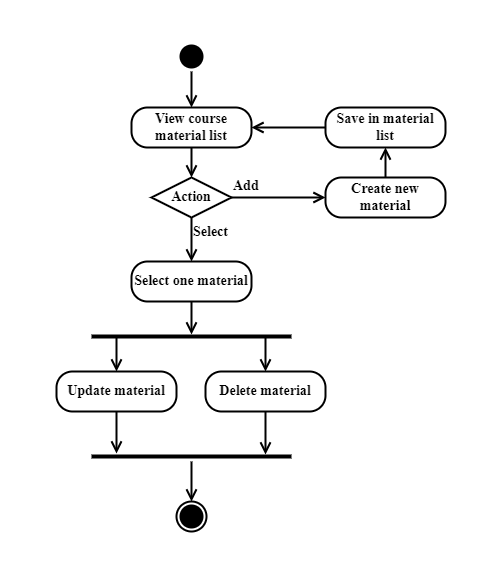


Figure 3.6 - Activity Diagram for Manage Course Material

Activity Diagram for Write Blog

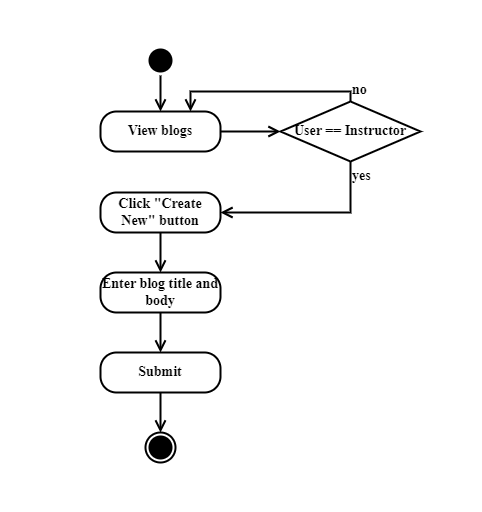


Figure 3.7 - Activity Diagram for Write Blog

Activity Diagram for View Blog

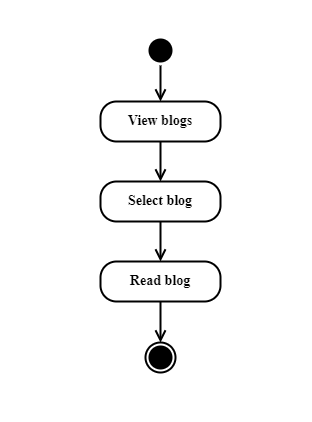


Figure 3.8 - Activity Diagram for View Blog

Activity Diagram for Manage Blog

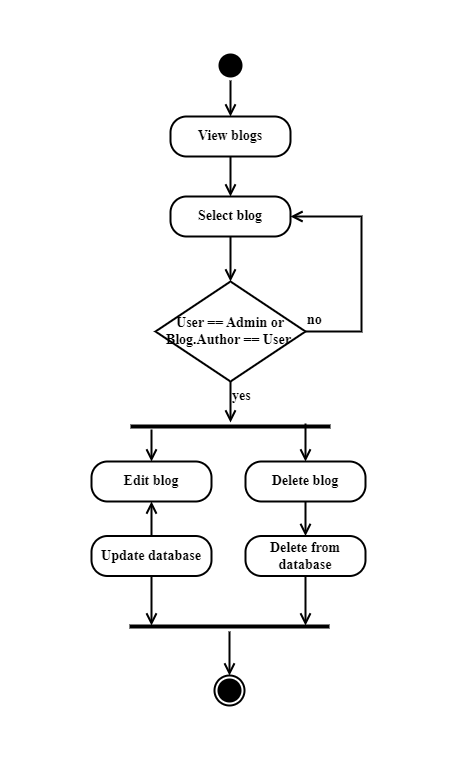


Figure 3.9 - Activity Diagram for Manage Blog

Activity Diagram for Give Feedback

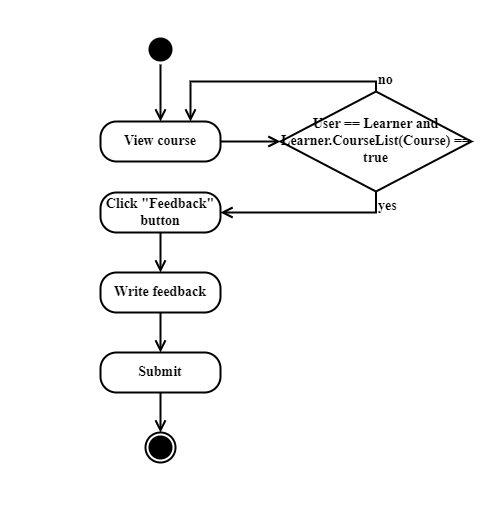


Figure 3.10 - Activity Diagram for Give Feedback

Activity Diagram for View Feedback

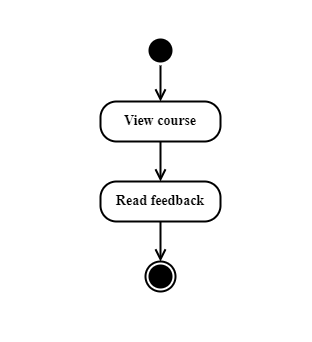


Figure 3.11 - Activity Diagram for View Feedback

Activity Diagram for Manage User

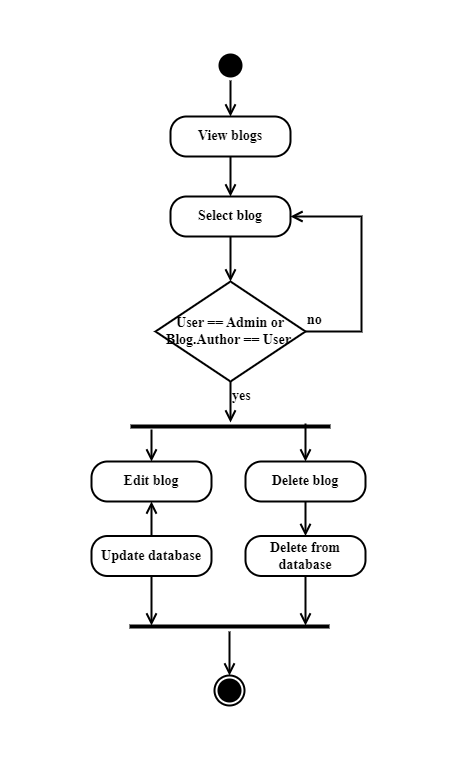


Figure 3.12 - Activity Diagram for Manage User

3.3 Sequence Diagram

Sequence Diagram for Registration

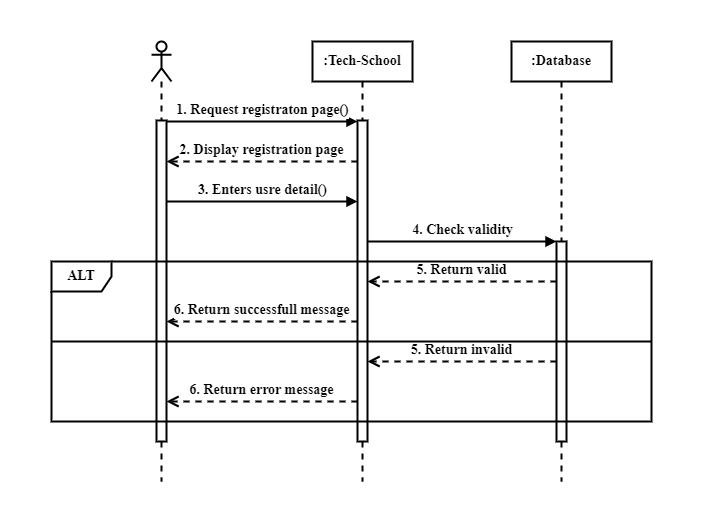


Figure 3.13 - Sequence Diagram for Registration

Sequence Diagram for Login

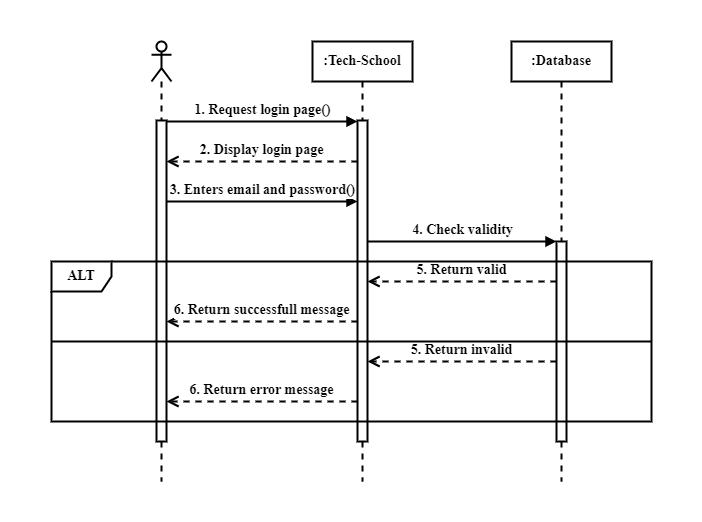


Figure 3.14 - Sequence Diagram for Login

Sequence Diagram for Add Course

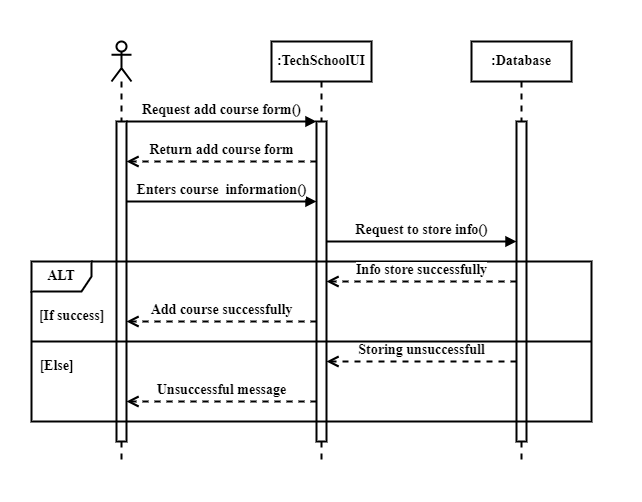


Figure 3.15 - Sequence Diagram for Add Course

Sequence Diagram for Edit Course

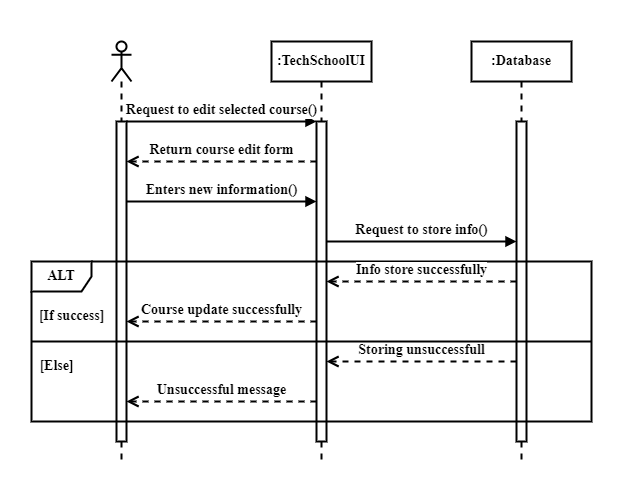


Figure 3.16 - Sequence Diagram for Edit Course

Sequence Diagram for Delete Course

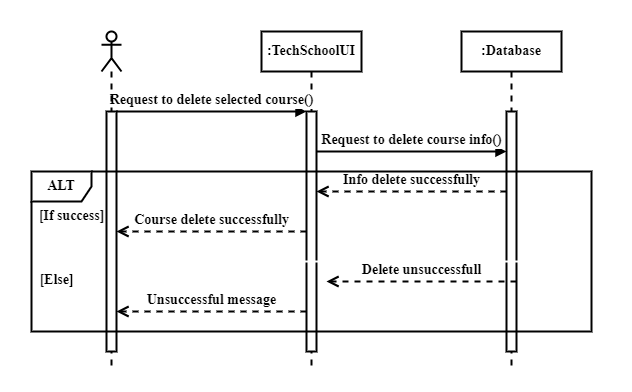


Figure 3.17 - Sequence Diagram for Delete Course

Sequence Diagram for Add Course Material

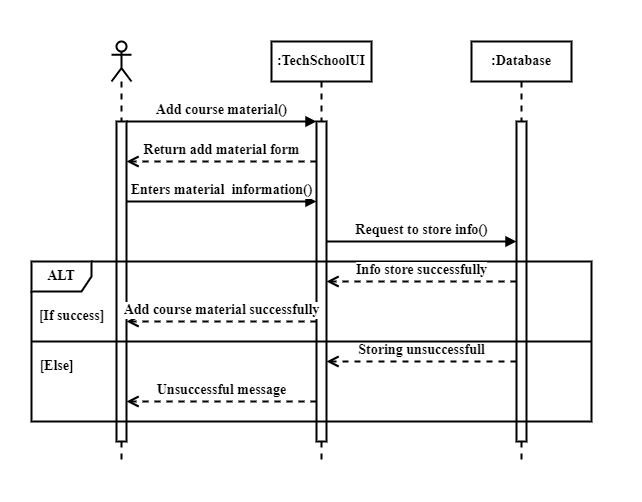


Figure 3.18 - Sequence Diagram for Add Course Material

Sequence Diagram for Edit Course Material

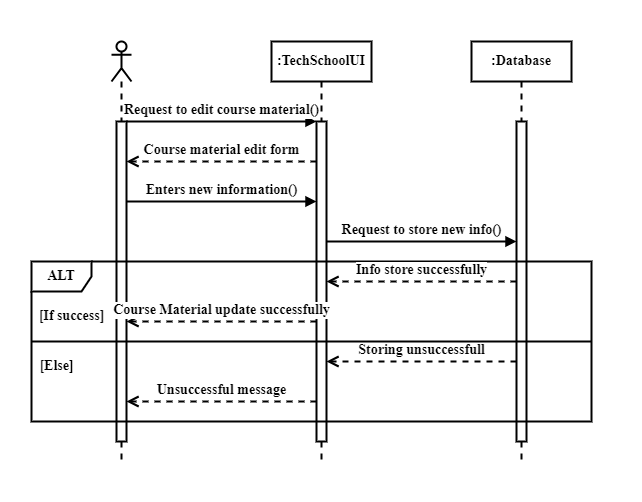


Figure 3.19 - Sequence Diagram for Edit Course Material

Sequence Diagram for Delete Course Material

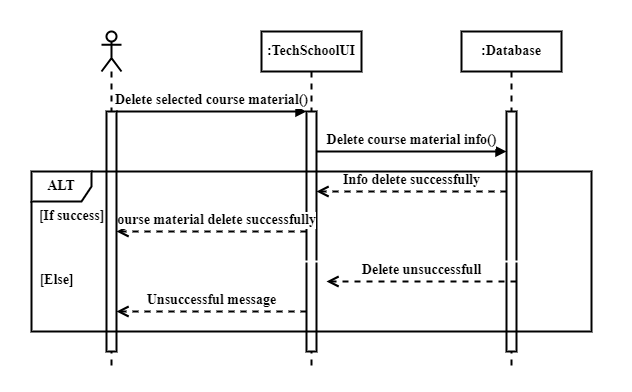


Figure 3.20 - Sequence Diagram for Delete Course Material

Sequence Diagram for Add Blog

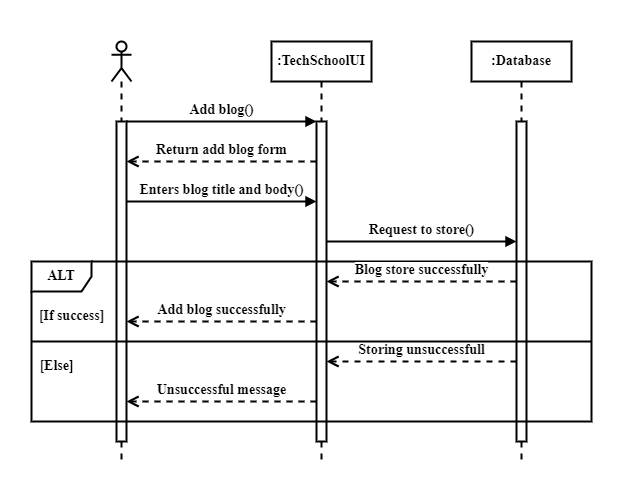


Figure 3.21 - Sequence Diagram for Add Blog

Sequence Diagram for Edit Blog

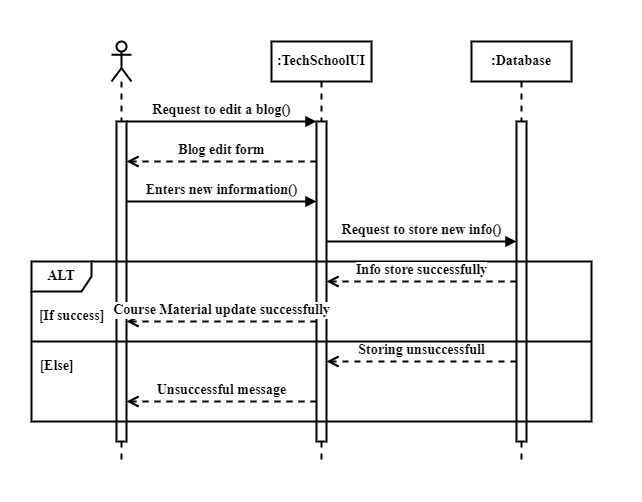


Figure 3.22 Sequence Diagram for Edit Blog

Sequence Diagram for Delete Blog

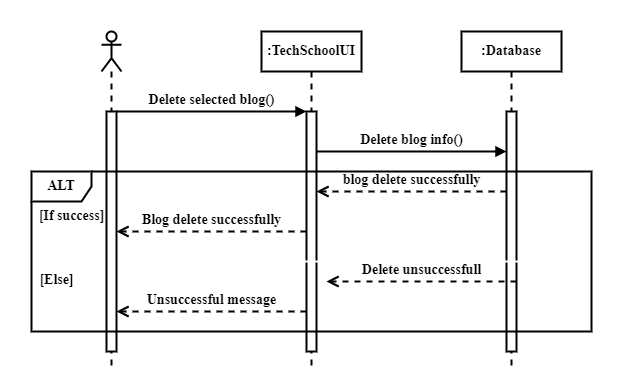


Figure 3.23 - Sequence Diagram for Delete Blog

Sequence Diagram for Add Feedback

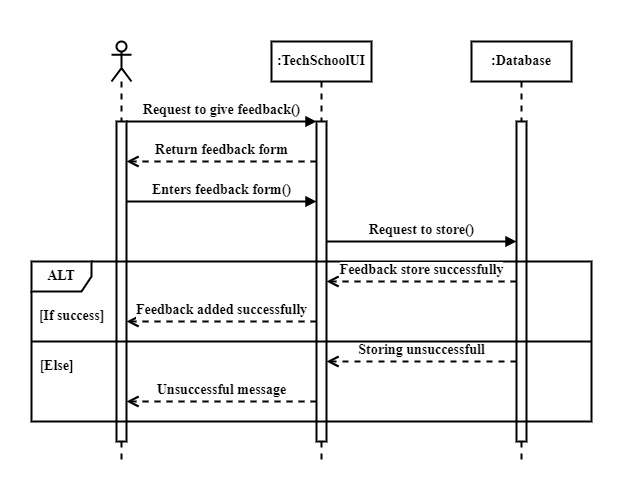


Figure 3.24 - Sequence Diagram for Add Feedback

Sequence Diagram for Enroll Course

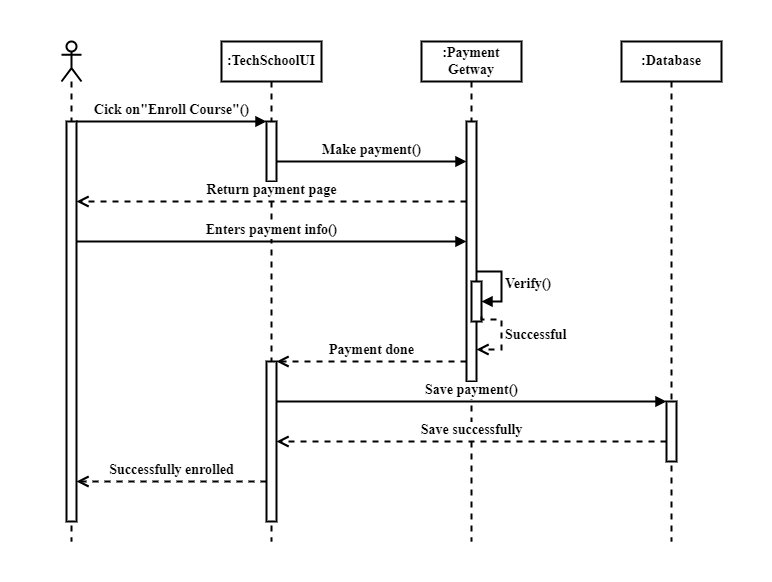


Figure 3.25 - Sequence Diagram for Enroll Course

Sequence Diagram for Delete User

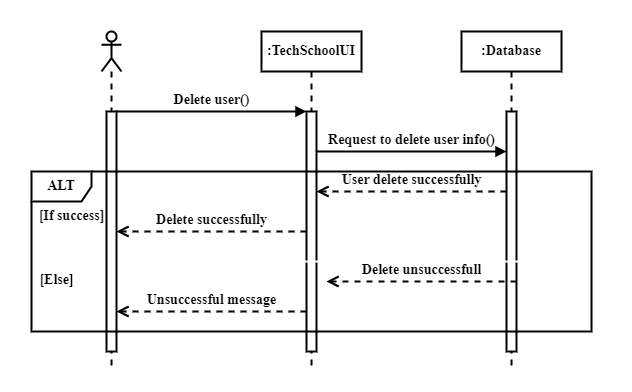


Figure 3.26 - Sequence Diagram for Delete User

3.4 ER Diagram

ER Diagram for Tech School

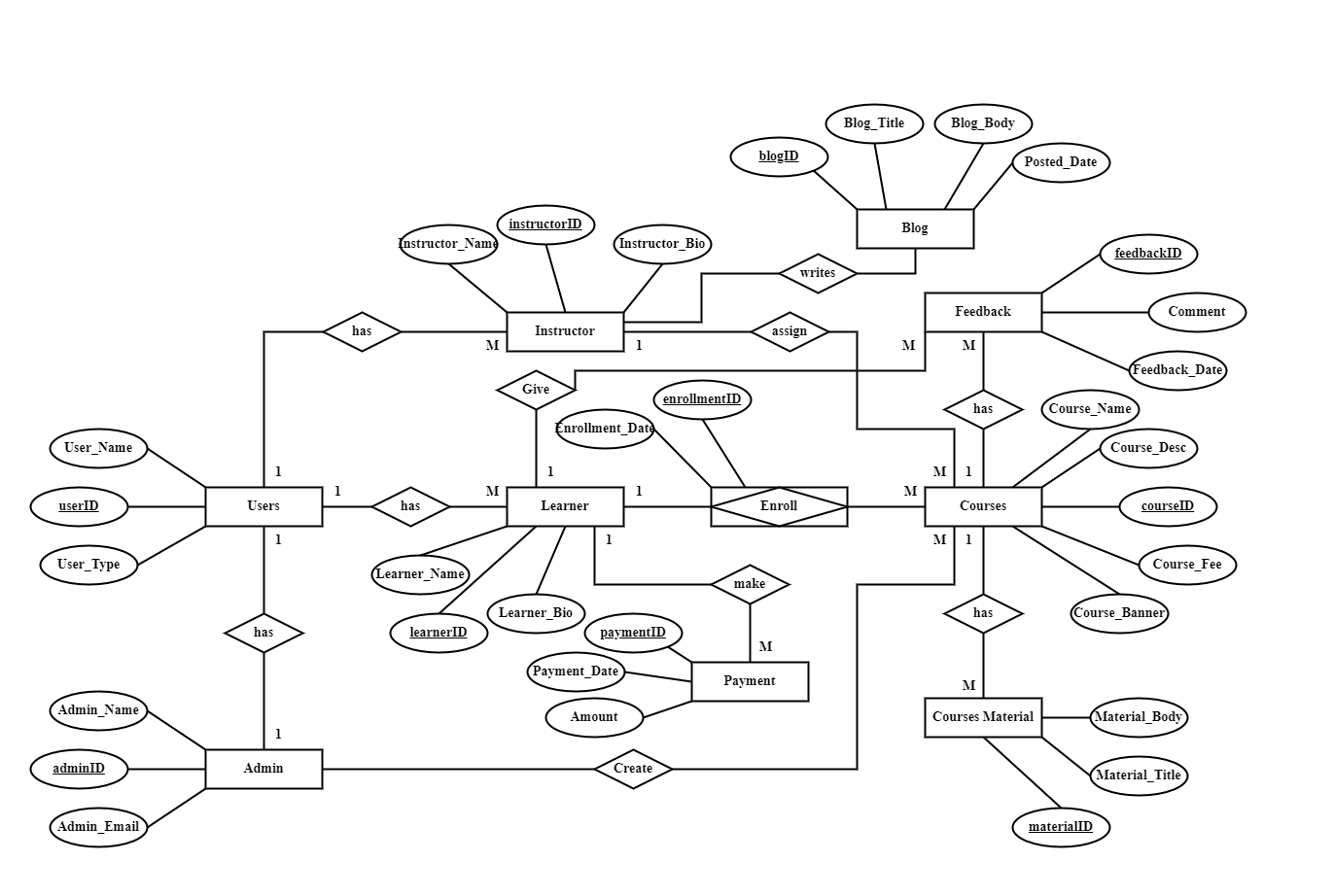


Figure 3.27 – ER Diagram for Tech School

Chapter – 4

Development tool & Technology

4.1 Integrated Development Environment (IDE)

* Visual Studio

4.2 Programming Language

* JavaScript

4.3 User interface Design

* Tailwind CSS (A utility-first CSS framework)
* Daisy UI
* React JS

4.4 Database

* MongoDB

4.5 Deploy and hosting

* Firebase (For frontend)
* Vercel (For backend)

Chapter – 5

System Testing

5.1 Testing Features

Feature testing is the software development process where multiple variations of a feature is tested to find out the best user experience of that feature. Using feature testing we can validate if a new feature for web page or app is a good fit and at the same time it is possible to determine which variation of the feature will be more successful. Feature Testing helps us to develop an efficient, reliable, useful, effective and secure web application.

4.1.1 Features to be tested

|  |  |  |
| --- | --- | --- |
| **Feature** | **Priority** | **Description** |
| Registration | High | The registration feature allows new users to create accounts on the platform. |
| Login | High | This critical feature enables users to securely log into the Tech School platform, ensuring authentication and access to personalized content, courses, and user-specific functionalities. |
| Update Profile | Medium | Users can update their profiles, including personal details, preferences, and additional information |
| Add Course to Course Cart | Medium | The course cart functionality allows users to add courses of interest to a virtual cart for later consideration. |
| Course Cart Management | Medium | This feature enables users to manage the contents of their course cart, providing options to review, remove, or proceed to payment for the selected courses. |
| Payment | High | The payment feature facilitates secure and seamless transactions for course enrollment. |
| Give Feedback | Medium | Users can provide feedback on completed courses, sharing their thoughts on the content, instructor, and overall learning experience |
| Create Course | High | Admin can utilize this feature to create new courses, providing details such as course content, objectives, and prerequisites. |
| Course Management | High | The course management feature empowers admins to oversee and administer their courses. |
| Add Course Material | Medium | Instructors can add supplementary course materials, such as documents, presentations, and videos link, enriching the learning experience for participants. |
| Write Blog | Medium | Users with the role of an instructor can write and publish blogs related to their expertise. |
| Blog Management | High | This feature enables the management of blogs, allowing instructors to edit or remove their published blogs. |
| Log Out | Low | The session must me killed after logging out |

Table 5.1 - Features priority table with description

4.2 Test Strategies

4.2.1 Test approach

To ensure the quality of the system, two different sort of testing has been used. Mainly, it is focused on Black Box Testing and White Box Testing.

Black Box Testing: Black Box Testing is also called functional testing. It is a testing strategy where internal mechanism is ignored and only focus on the output. For some specific input some outputs are generated. Then these outputs compare with the expected output. If it matches, the function is accepted.

White Box Testing: White Box Testing are called structural testing. In this testing strategy, internal mechanism of the system is considered.

4.2.2 Testing Schedule

|  |  |
| --- | --- |
| Test Phase | Time |
| Testing plan creating | 1 Week |
| Unit test | During Development |
| Component testing | During Development |
| Testing user interfaces | 1 Week |
| Performance test | 2 Week |
| Accessibility test | 2 Week |

Chapter - 6

User Manual

Chapter - 7

Conclusion

**Mobile App:** Develop a mobile app for Android and iOS devices to make learning accessible on smartphones and tablets.

7.1 **Project Summary**

The E-Learning Management System project signifies a significant step forward in reshaping the landscape of online education. This innovative platform is designed to tackle challenges inherent in digital learning, providing a user-friendly interface for learners, instructors, and administrators. The project's focus on creating a seamless and efficient system is visually depicted through detailed use case and activity diagrams, offering a clear understanding of its functionalities. The successful completion of this project is attributed to collaboration, innovation, and the collective effort of those involved. It stands as a testament to the commitment to advancing online education through technology.

7.2 Project Link

[https://tech-school.app/](https://tech-school-iprosenjitp.web.app/)

7.3 Limitations

There are certain features that have not been developed yet. I am working on expanding the functionality of the platform, and these features are planned for future development.

* Feedback Feature
* Course Material Feature

7.4 Future Scope

**Discussion Forums:** Implement discussion forums where learners can engage in discussions, ask questions, and collaborate with peers and instructors. Forums can be organized by course or topic.

**Interactive Assessments:** Incorporate interactive quizzes, assignments, and simulations to assess learners' understanding and skills.

Course Request System: A course request system will be implemented to allow users to suggest topics or subjects they would like to see on the platform.

In-Platform Notifications: In -platform notifications will keep users informed about new courses relevant announcements. This feature enhances user engagement and ensures that learners and instructors stay updated on platform activities.