Green University of Bangladesh

Department of Computer Science and Engineering (CSE)
Semester: (Fall, Year: 2024), B.Sc. in CSE (Day)

Green Virtual Classroom

Course Title: Integrated Design Project - II
Course Code: CSE 406
Section: 212 D3

Students Details

Name	ID
Waliyel Hasnat Zaman	212002022
Bibi Fatema Priya	212002089
Abdur Rahman	212002002

Submission Date: 20/12/2024
Course Teacher's Name: Mr. Palash Roy

[For teachers use only: Don't write anything inside this box]

Lab Project Status	
Marks:	**Signature:**
Comments:	**Date:**

Contents

Chapter 1

Introduction

1.1 Introduction

The "Green Virtual Classroom" project aims to modernize higher education by creating an interactive online platform for students, teachers, and administrators. It focuses on effective course delivery, robust user management, and smooth interaction to meet the increasing need for adaptable and accessible learning solutions. Drawing inspiration from platforms such as Google Classroom [1], Microsoft Teams [2], Moodle [3], and Canvas [4], key features include communication tools, assessment systems, and content management, with a strong emphasis on security and compliance.

By meeting user requirements for registration, enrollment, and course management, the project ensures a user-friendly experience for all stakeholders. Non-functional requirements prioritize performance, scalability, and usability, aiming to deliver a high-quality learning environment. Through innovation and collaboration, the Green Virtual Classroom project seeks to revolutionize education, making learning more engaging, inclusive, and effective.

1.2 Problem Description

1.2.1 Literature Review

The shift to online learning, especially in higher education, has highlighted significant challenges with existing virtual classroom platforms:

- Many platforms lack flexibility, leading to mismatches with institutional needs [3, 4].
- Platforms often struggle with performance issues under high user loads [1, 2].
- Robust features for managing diverse user roles and permissions are often missing [3].
- Tools for comprehensive student evaluations and feedback are limited [1].
- Concerns about data protection and compliance with regulations are prevalent [2].
- Many platforms lack interactive features to promote active learning [4].

- Integration with existing Learning Management Systems (LMS) and other educational tools can be challenging [3, 4].
- Tools for detecting and preventing academic dishonesty, such as plagiarism detection, are inadequate [1].

1.2.2 Addressing Challenges

The "Green Virtual Classroom" project tackles these challenges by focusing on the following key questions:

1. **How can we tailor the platform to meet diverse educational needs?**
   Solution: Implement customizable templates and features inspired by platforms such as Moodle [3].
2. **How can we ensure the platform supports a growing number of users?**
   Solution: Use cloud infrastructure and load balancing, as adopted by Microsoft Teams [2].
3. **How can we design effective role-based access control?**
   Solution: Develop a robust user management module similar to features in Canvas [4].
4. **How can we enhance assessment and feedback mechanisms?**
   Solution: Introduce advanced grading and feedback tools inspired by Google Classroom [1].
5. **How can we ensure data security and regulatory compliance?**
   Solution: Implement strong encryption and secure authentication measures, leveraging best practices from Microsoft Teams [2].
6. **How can we increase student interaction and engagement?**
   Solution: Add real-time chat, forums, and collaborative tools like those found in Canvas [4].
7. **How can we seamlessly integrate with existing LMS and tools?**
   Solution: Develop APIs for smooth data exchange and interoperability, drawing inspiration from Moodle's integration capabilities [3].

By addressing these challenges, the Green Virtual Classroom project aims to create a versatile and effective virtual learning environment for higher education.

1.3 Aims and Objectives

1.3.1 Aims

The aim of the Green Virtual Classroom project is to develop an innovative and adaptable online learning platform tailored for higher education institutions, ensuring effective course delivery, seamless user management, and enhanced student engagement. By leveraging modern technologies, the platform seeks to address limitations in existing solutions while prioritizing flexibility, performance, security, and user experience.

1.3.2 Objectives

1. Develop an "University Virtual Classroom" for seamless interaction among students, teachers, and admins.
2. Prioritize robust user management and role-based access control.
3. Implement efficient course management and access to materials.
4. Provide communication tools like forums and optionally real-time chat.
5. Ensure secure content management, compliance with accessibility standards and optimal platform performance.

1.4 Motivation

- Enhance learning accessibility for all students.
- Improve educational resource management.
- Foster interactive and engaging learning environments.
- Ensure secure and compliant online education.
- Streamline administrative and academic processes.

1.5 Project Description

The Green Virtual Classroom is an interactive and scalable online platform designed to meet the specific needs of higher education institutions. This project aims to modernize virtual learning environments by providing a user-friendly interface and robust tools for students, teachers, and administrators. The system focuses on seamless course management, effective communication, secure assessments, and academic integrity checks to enhance the quality of online education. The key components of the project include:

1. User management system for login, profile management, and role-based access control.
2. A central hub where users can access all functionalities, such as courses, materials, communication tools, assessments and notices
3. Teachers can create and manage courses, while students can enroll and access course content.
4. Teachers can upload study materials for students to access.
5. Real-time chat, discussion forums, and messaging enable collaboration between students and teachers.
6. Teachers can assign assignments, and students can submit their work for grading and feedback.

7. Both students and the teachers can seek help by submitting a help request and also seamlessly reset their password if they forget it.

8. Users can securely exit the system after completing their tasks.

1.5.1 Block Diagram of the features

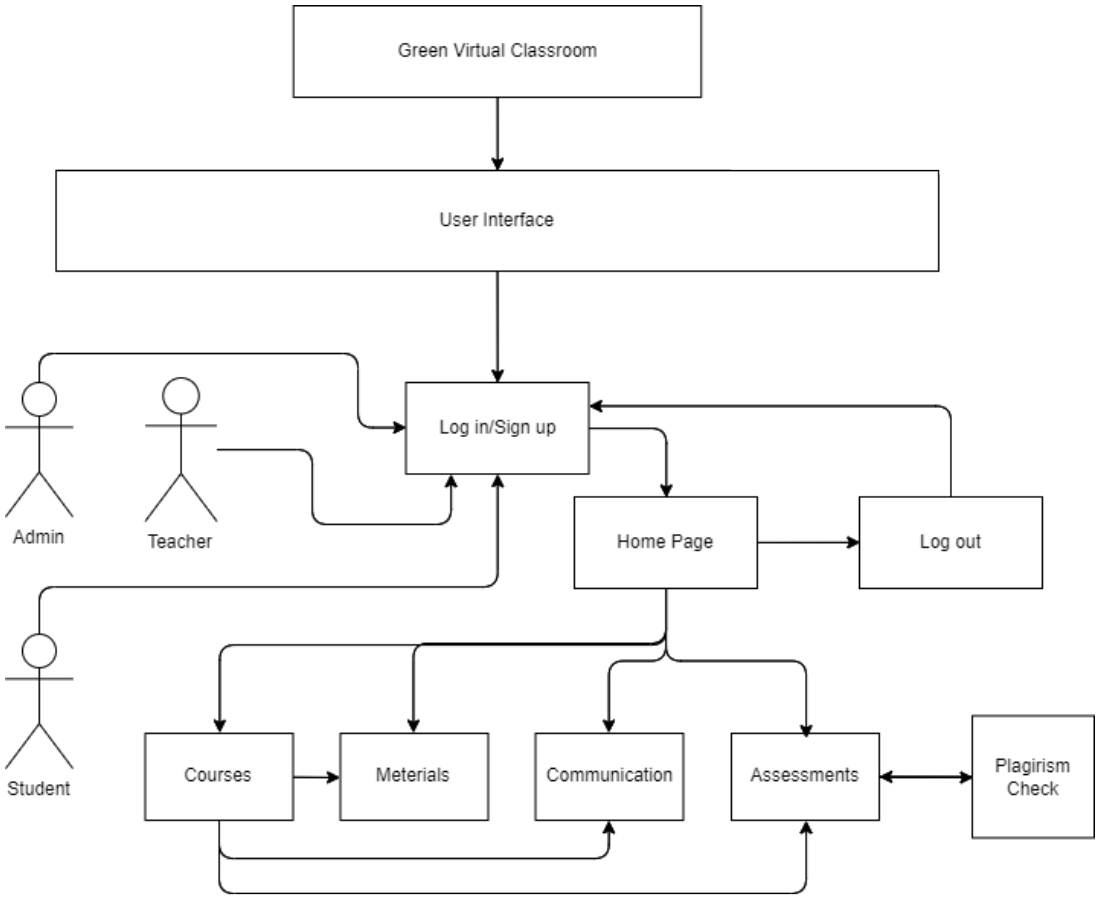


Figure 1.1: Block Diagram of GVC

Chapter 2

Requirement Analysis and Design

2.1 Requirement Analysis

1. User management system for registration, profile management, and role-based access control.
2. Course management system for creating, editing, and deleting courses, along with materials upload functionality.
3. Communication system featuring discussion module ensuring real-time chat.
4. Assessment system for online assignments.
5. Content management system for secure storage and easy access to course materials.
6. Integration capability with existing learning management systems if applicable.

2.1.1 Functional Requirement

1. **Students**:
    - Ability to login
    - Ability to visit enrolled courses
    - Ability to make and edit posts in stream
    - Ability to access materials
    - Ability to participate in discussions
    - Ability to submit assignments
    - Ability to view notices
    - Ability to submit help request
2. **Teachers**:
    - Ability to visit assigned courses courses
    - Ability to upload materials and create assignments

- Ability to make and edit posts in stream
- Ability to check submissions of the assignments
- Ability to view notices
- Ability to submit help request
- Ability to participate in discussions

3. **Administrators**:
    - Ability to manage enrollments
    - Ability to manage user accounts
    - Ability to manage roles and permissions
    - Ability to monitor system performance
    - Ability to ensure data security

2.1.2 Non-functional Requirement

1. **Performance**: Responsive platform handling multiple users concurrently with fast loading times for content and communication tools.
2. **Scalability**: Ability to accommodate a growing number of users and courses.
3. **Availability**: High availability with minimal downtime and backup/disaster recovery mechanisms in place.
4. **Security**: Secure authentication, authorization, and encryption of data transmission and storage to protect sensitive information.
5. **Usability**: User-friendly interface, intuitive navigation, clear instructions, and support for accessibility standards.
6. **Maintainability**: Easy maintenance and updates with a well-documented codebase for developers.

2.2 Tools and Techniques

The development of the Green Virtual Classroom project involves various tools, technologies, and frameworks to ensure a robust, scalable, and user-friendly platform. Below is a detailed overview of the tools and technologies used:

- **Hardware Requirements:**
    - **RAM:** Minimum 8GB for optimal performance.
    - **Processor:** Intel i5/i7 or equivalent.
- **Software Tools:**
    - **VSCode:** Code editor for writing and debugging code.

  - **XAMPP:** Local server environment (Apache, MySQL, PHP).

- **Programming and Web Technologies:**
  - **HTML:** Structures the web pages.
  - **CSS:** Styles and enhances page visuals.
  - **Bootstrap:** Provides responsive design components.
  - **PHP:** Handles backend operations and server-side logic.
  - **Java:** Manages advanced logic and APIs.
- **Database:**
  - **MySQL:** Stores and manages relational data.
- **Additional Tools:**
  - **Web Browsers:** Testing platform compatibility (Chrome, Opera GX).

2.3 Gantt Chart

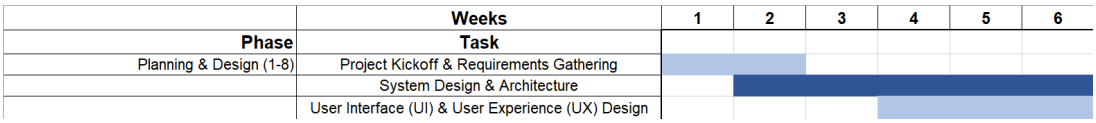


Figure 2.1: Gantt Chart of Phase-1

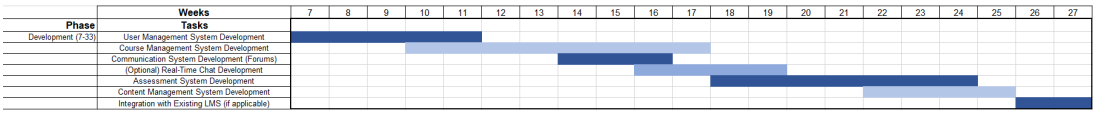


Figure 2.2: Gantt Chart of Phase-2

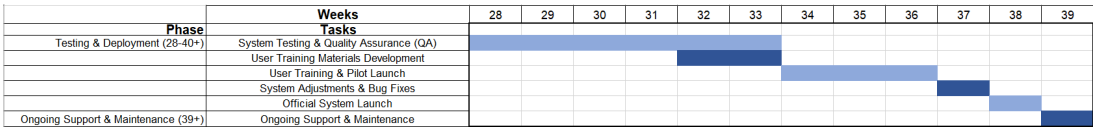


Figure 2.3: Gantt Chart of Phase-3

2.4 Use Case Diagram

A use case model is a visual representation of the interactions between users and a system to achieve specific goals.

- Understanding Requirements: Use cases help capture functional requirements by illustrating how the system should behave in various scenarios.

- Communication: They provide a clear and concise way to communicate with admins, including clients, users, and developers, ensuring that everyone has a common understanding of the system's functionality.
- Scope Definition: Use cases define the scope of the system by specifying what will and will not be included, which helps in managing project scope and preventing scope creep.
- Guidance: They guide the system design and development process by detailing the interactions between the system and its users, which helps developers understand what needs to be implemented.
- Test Case Development: Use cases serve as a basis for developing test cases, ensuring that all functional requirements are tested.

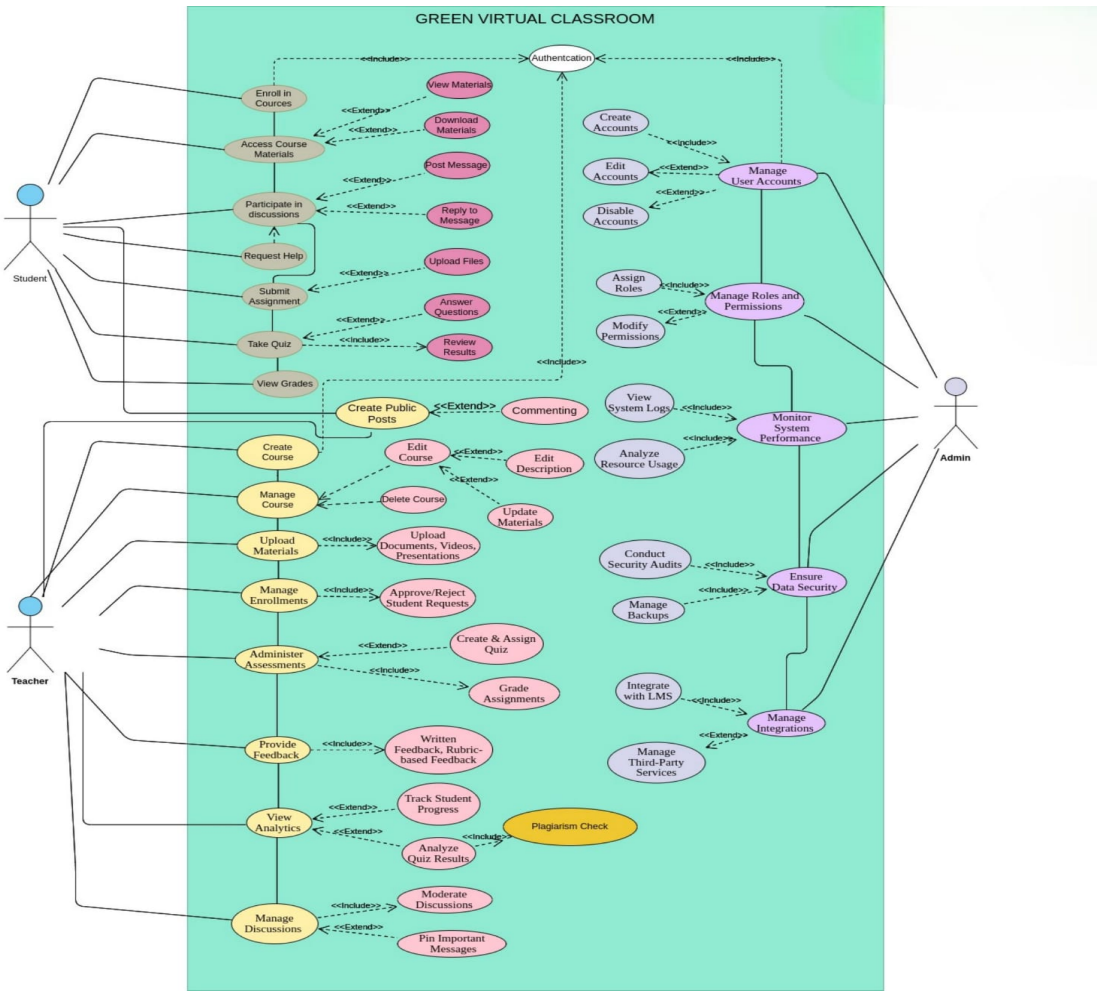


Figure 2.4: Use Case Diagram

- Student: This section shows the features available to a student in a virtual classroom. These include enrolling in courses, accessing course materials, participating in discussions, submitting assignments, taking quizzes, viewing grades and tracking their progress.
- Teacher: This section shows the features available to a teacher in a virtual classroom. These include creating and editing courses, uploading materials, including documents,

videos and presentations, conducting security audits, managing enrollments, creating and assigning quizzes, grading assignments, integrating with other learning management systems, providing feedback to students, tracking student progress, moderating discussions and managing third-party services.

2.5 Data Flow Diagram

The main objective of using Data Flow Diagram models is to visualize and analyze the flow of data within the system, ensuring efficient data processing and management.

- Visualize Data Movement: Provide a clear graphical representation of how data flows through the system, including inputs, processes, and outputs.
- Identify System Components: Break down the system into smaller, manageable components to understand their interactions and data dependencies.
- Simplify Complexity: Simplify the understanding of complex data processes and interactions within the system.
- Document System Requirements: Serve as part of the system documentation, aiding in future maintenance, enhancements, and ensuring that all data processing requirements are captured.

2.5.1 DFD Level - 0

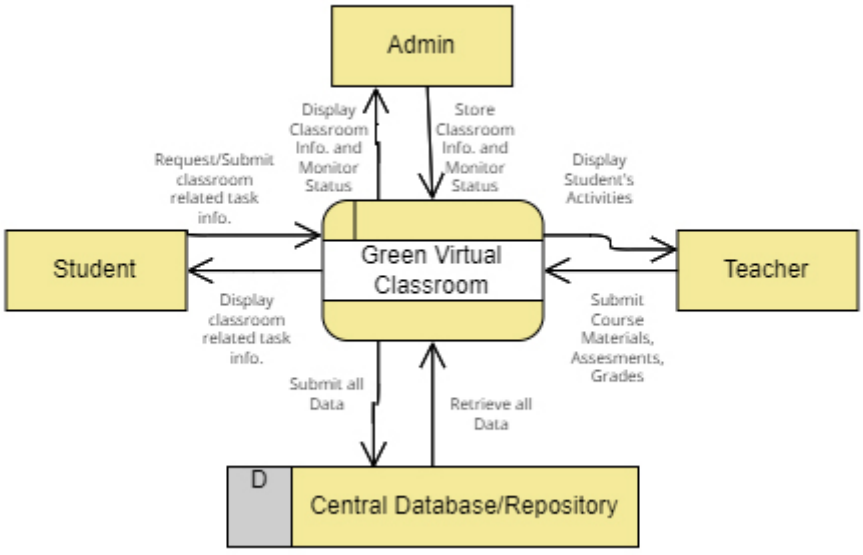


Figure 2.5: DFD Level 0

- Admin can manage classroom information and monitor statuses.

- Students can interact with the system to get and submit task information.
- Teachers can upload materials and monitor student activities.
- Central Database serves as the main storage for all the data, which the system accesses for both storing new information and retrieving existing information.

2.5.2 DFD Level - 1

- Entities
  - Student: Central user who interacts with the system to access materials, participate in discussions, submit assignments, view grades, and request help.
  - Teacher: Manages course materials, takes quizzes,enrollments, and assessments.
- Processes
  - Course Management: Handles course content and materials.
  - Enrollment Management: Manages student enrollment status.
  - Assessment Management: Oversees the creation and grading of assessments.
  - Reports Storage: Stores various reports, including assignments and quiz answers.
- Data Stores
  - Course Directory: Stores information about available courses.
  - Materials Directory: Stores course materials.
  - Discussion Directory: Contains discussion records.
  - Grades Info: Maintains students' grade information.

Data Flows

- For Students
  - Enroll in courses: Students select courses from the Course Directory.
  - Access Course Materials: Students retrieve materials from the Materials Directory.
  - Participate in Discussion: Interaction within the Discussion Directory.
  - Submit Assignment: Assignments are stored in Reports Storage.
  - View Grade: Grades are displayed from Grades Info.
  - Request Help: Help requests are processed and solutions provided.
- For Teachers
  - Upload Course Materials: Materials are added to the Materials Directory via Course Management.
  - Manage Enrollments: Enrollments are updated in Enrollment Management.
  - Submit Grades: Assessment results are sent to Grades Info.
  - Plagiarism Check: Checking plagiarism through Assessment.

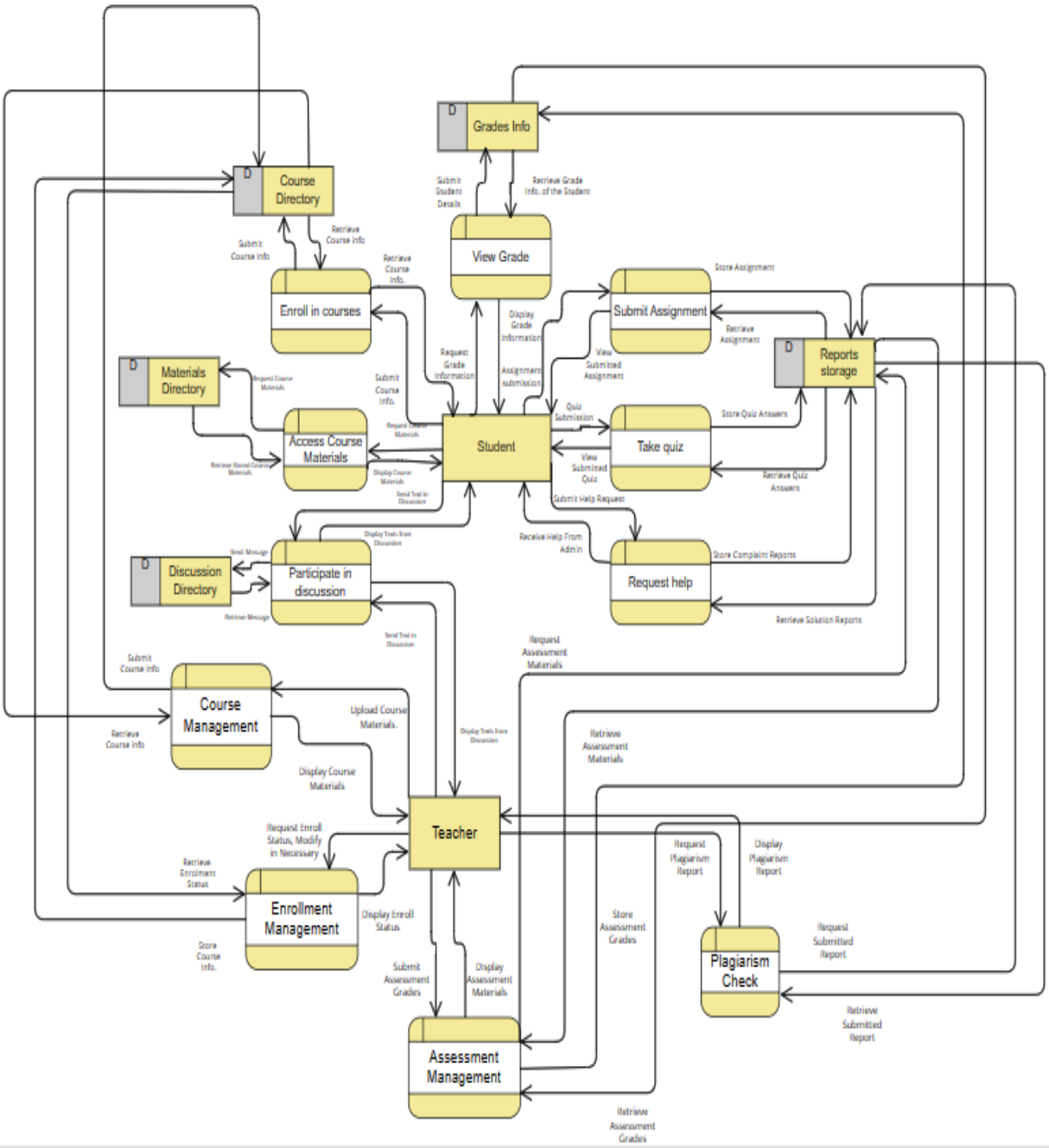


Figure 2.6: DFD Level 1

2.6 E-R Diagram and Database Schema

The E-R diagram for Green Virtual Classroom shows how data is organized and connected in an online learning system. This system allows students, teachers, and administrators to interact with courses, assignments, discussions, and other classroom activities. Here's a simple breakdown of the key parts of the diagram:

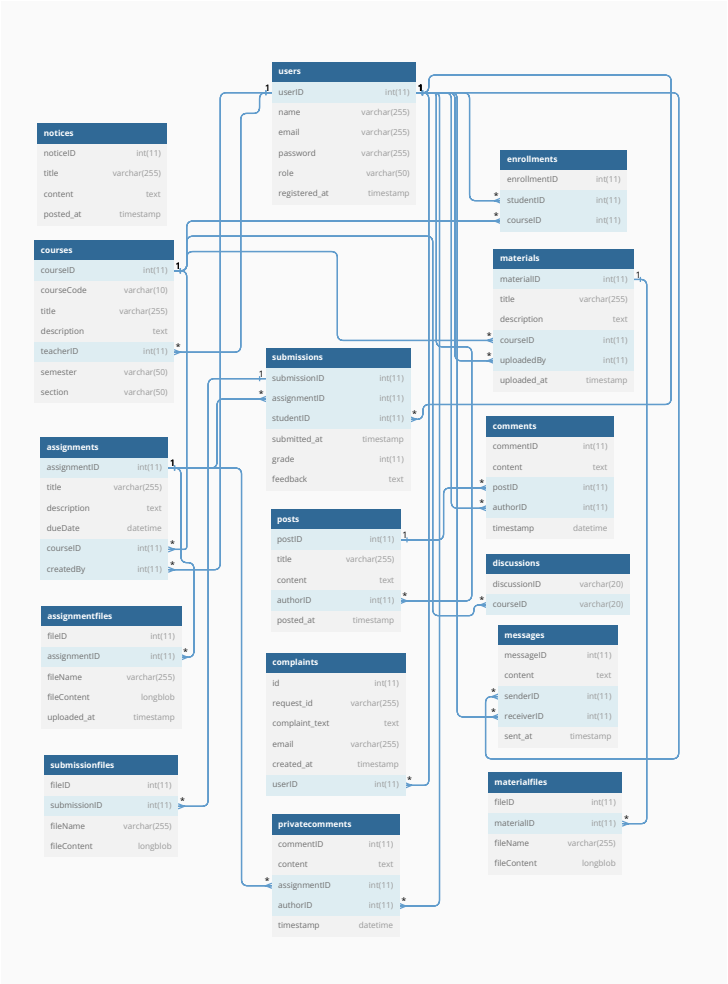


Figure 2.7: ER Diagram

Entities, Attributes and Their Roles

- **Users**
  - **Attributes:**
    - `userID`: Unique identifier for a user.
    - `name`: Full name of the user.
    - `email`: Email address of the user.
    - `password`: Encrypted password for authentication.
    - `role`: Role of the user (e.g., admin, teacher, student).
    - `registered_at`: Timestamp of when the user registered.
  - **Roles:** Represents all users of the system with different roles.
- **AssignmentFiles**
  - **Attributes:**
    - `fileID`: Unique identifier for the file.
    - `assignmentID`: Links to the assignment to which the file belongs.
    - `fileName`: Name of the file.

    - `fileContent`: Binary content of the file.
    - `uploaded_at`: Timestamp of when the file was uploaded.
  - **Roles:** Stores files associated with assignments.

- **Assignments**
  - **Attributes:**
    - `assignmentID`: Unique identifier for an assignment.
    - `title`: Title of the assignment.
    - `description`: Detailed description of the assignment.
    - `dueDate`: Deadline for submission.
    - `courseID`: Links to the course the assignment belongs to.
    - `createdBy`: Links to the user who created the assignment.
  - **Roles:** Manages assignments for courses.

- **Comments**
  - **Attributes:**
    - `commentID`: Unique identifier for a comment.
    - `content`: Text content of the comment.
    - `postID`: Links to the post the comment belongs to.
    - `authorID`: Links to the user who authored the comment.
    - `timestamp`: Time when the comment was made.
  - **Roles:** Stores comments on posts.

- **Complaints**
  - **Attributes:**
    - `id`: Unique identifier for a complaint.
    - `request_id`: Reference ID for the complaint request.
    - `complaint_text`: Description of the complaint.
    - `email`: Email address of the complainant.
    - `created_at`: Time when the complaint was created.
    - `userID`: Links to the user who submitted the complaint.
  - **Roles:** Manages user complaints and their details.

- **Courses**
  - **Attributes:**
    - `courseID`: Unique identifier for a course.
    - `courseCode`: Short code for the course (e.g., CSE324).
    - `title`: Title of the course.
    - `description`: Detailed description of the course.
    - `teacherID`: Links to the teacher of the course.
    - `semester`: Semester when the course is offered.

        * `section`: Section of the course.
    - **Roles:** Represents the courses offered in the system.

- **Discussions**
    - **Attributes:**
        * `discussionID`: Unique identifier for a discussion.
        * `courseID`: Links to the course the discussion belongs to.
    - **Roles:** Represents group discussions for each course.

- **Enrollments**
    - **Attributes:**
        * `enrollmentID`: Unique identifier for an enrollment.
        * `studentID`: Links to the student enrolled in the course.
        * `courseID`: Links to the course in which the student is enrolled.
    - **Roles:** Tracks course enrollments of students.

- **MaterialFiles**
    - **Attributes:**
        * `fileID`: Unique identifier for the file.
        * `materialID`: Links to the material to which the file belongs.
        * `fileName`: Name of the file.
        * `fileContent`: Binary content of the file.
    - **Roles:** Stores files associated with course materials.

- **Materials**
    - **Attributes:**
        * `materialID`: Unique identifier for a material.
        * `title`: Title of the material.
        * `description`: Detailed description of the material.
        * `courseID`: Links to the course the material belongs to.
        * `uploadedBy`: Links to the user who uploaded the material.
        * `uploaded_at`: Time when the material was uploaded.
    - **Roles:** Stores course materials.

- **Messages**
    - **Attributes:**
        * `messageID`: Unique identifier for a message.
        * `content`: Text content of the message.
        * `senderID`: Links to the user who sent the message.
        * `receiverID`: Links to the user who received the message.
        * `sent_at`: Time when the message was sent.

  - **Roles:** Represents private messages between users.

- **Notices**
  - **Attributes:**
    * `noticeID`: Unique identifier for a notice.
    * `title`: Title of the notice.
    * `content`: Text content of the notice.
    * `posted_at`: Time when the notice was posted.
  - **Roles:** Stores notices posted in the system.

- **Posts**
  - **Attributes:**
    * `postID`: Unique identifier for a post.
    * `title`: Title of the post.
    * `content`: Text content of the post.
    * `authorID`: Links to the user who authored the post.
    * `posted_at`: Time when the post was created.
  - **Roles:** Represents posts created by users.

- **PrivateComments**
  - **Attributes:**
    * `commentID`: Unique identifier for a private comment.
    * `content`: Text content of the comment.
    * `assignmentID`: Links to the assignment the comment is about.
    * `authorID`: Links to the user who authored the comment.
  - **Roles:** Tracks private comments on assignments.

- **SubmissionFiles**
  - **Attributes:**
    * `fileID`: Unique identifier for the file.
    * `submissionID`: Links to the submission the file belongs to.
    * `fileName`: Name of the file.
    * `fileContent`: Binary content of the file.
  - **Roles:** Stores files associated with submissions.

- **Submissions**
  - **Attributes:**
    * `submissionID`: Unique identifier for a submission.
    * `assignmentID`: Links to the assignment the submission is for.
    * `studentID`: Links to the student who made the submission.
    * `submitted_at`: Time when the submission was made.
  - **Roles:** Represents submissions for assignments.

2.7 System Architecture

The system architecture of the Green Virtual Classroom (GVC) project is presented at both high and low levels, offering a comprehensive view of how the system's components interact.

2.7.1 High Level Architecture:

The high level system architecture of the Green Virtual Classroom (GVC) shows how various systems like course management, authentication, and assessment connect to the central GVC system. Each module communicates through commands and data exchange, with the GVC database acting as the primary data repository.

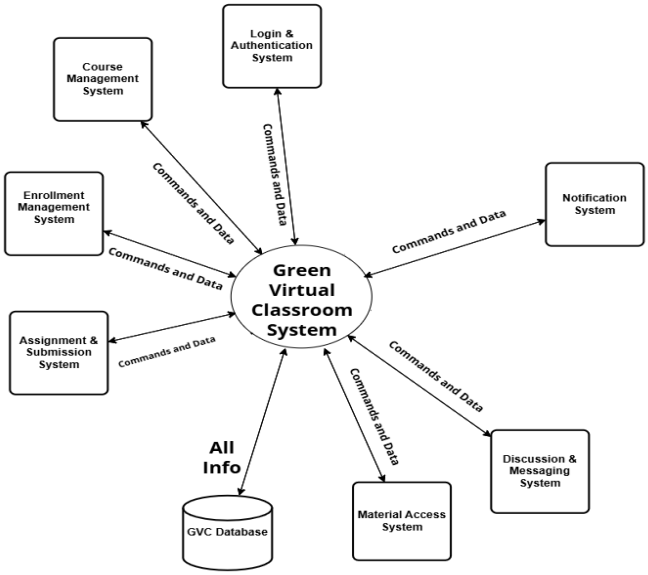


Figure 2.8: High Level System Architecture of GVC

2.7.2 High to Low Level System Architecture:

The high to low level system architecture of the Green Virtual Classroom (GVC) breaks down each module into specific subsystems, illustrating how they interconnect and function. It details individual components such as course management, enrollment, quiz assessment, plagiarism checking, and notifications, highlighting the data flow and dependencies between these parts. This layered approach provides a detailed view of the operational processes within GVC, supporting efficient development, integration, and maintenance.

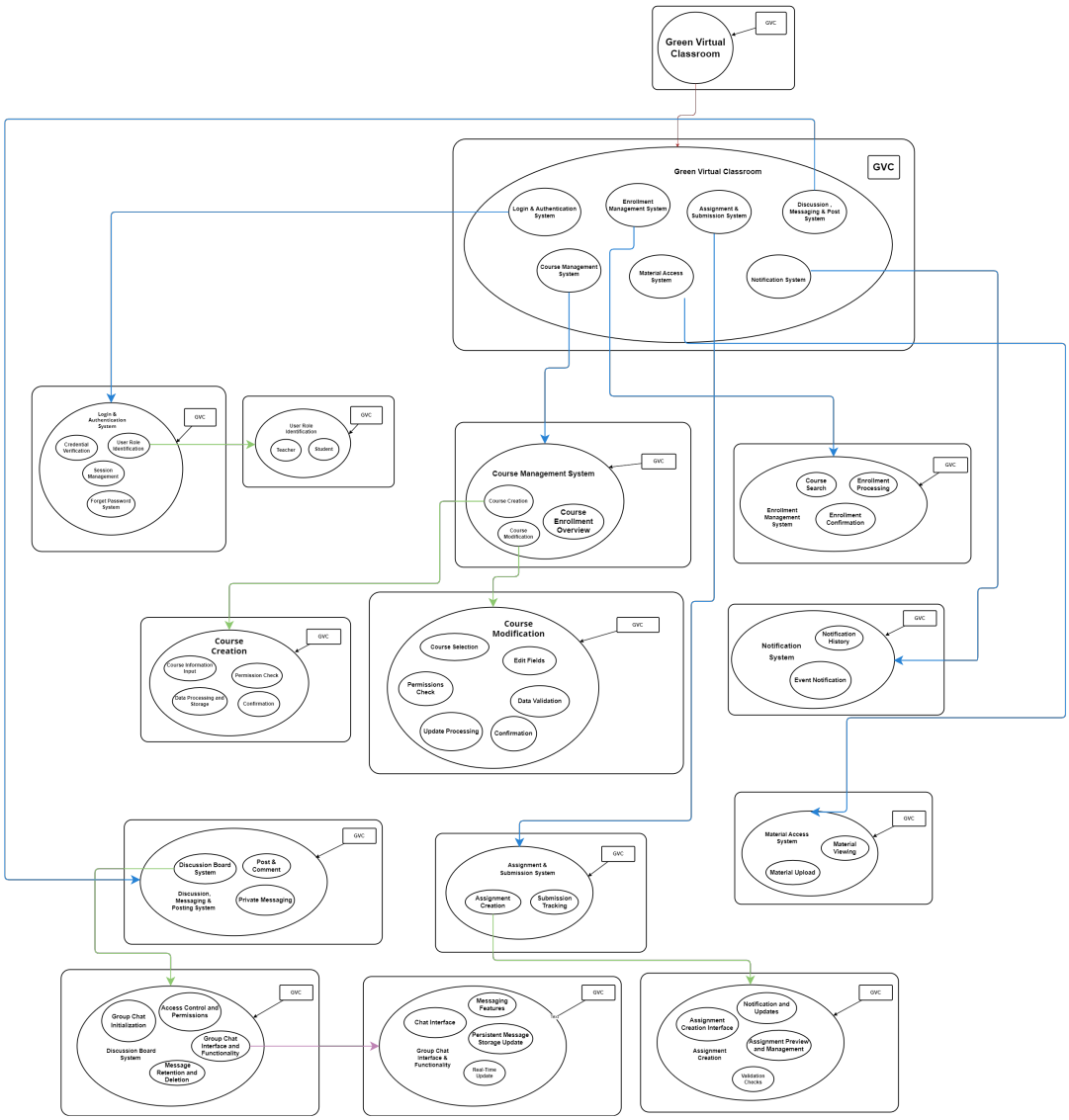


Figure 2.9: High to Low Level System Architecture of GVC

2.7.3 Login & Authentication System:

The Login & Authentication System in the GVC architecture handles user verification, session management, and password recovery. It identifies user roles as either "Teacher" or "Student" to determine access and permissions within the system.

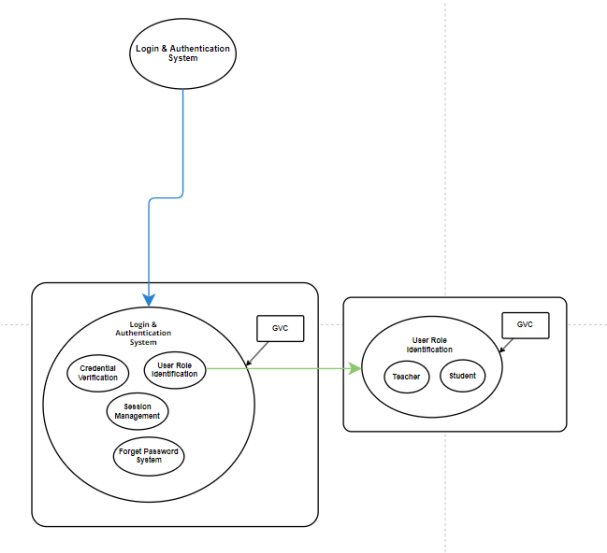


Figure 2.10: Login & Authentication System

2.7.4 Enrollment Management System:

The Enrollment Management System in GVC allows users to search for courses, process enrollment requests, and confirm enrollments. It facilitates streamlined course access and management for users within the platform.

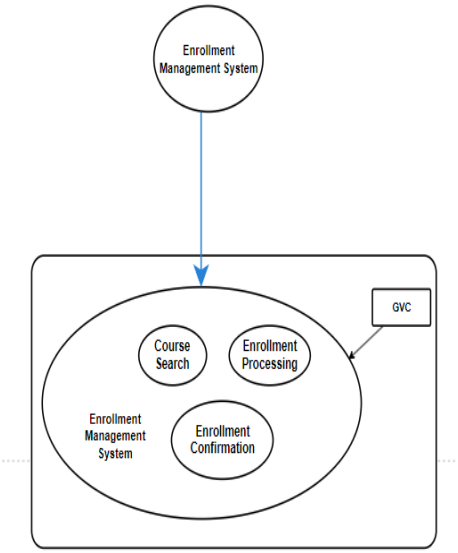


Figure 2.11: Enrollment Management System

2.7.5 Course Management System:

The Course Management System manages Course Creation, Course Modification, and Enrollment, with steps like permissions, data processing, and confirmation, connected through an

external "GVC" entity.

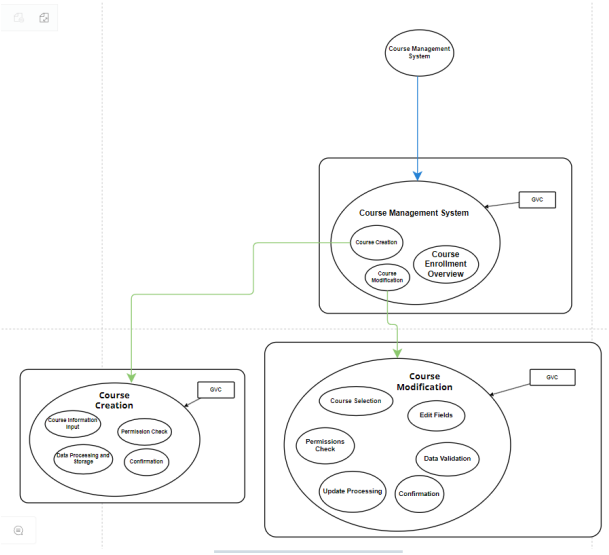


Figure 2.12: Course Management System

2.7.6 Material Access System:

The Material Access System manages the uploading and viewing of materials, with an interaction point for external access through "GVC." It facilitates efficient access and organization of learning resources.

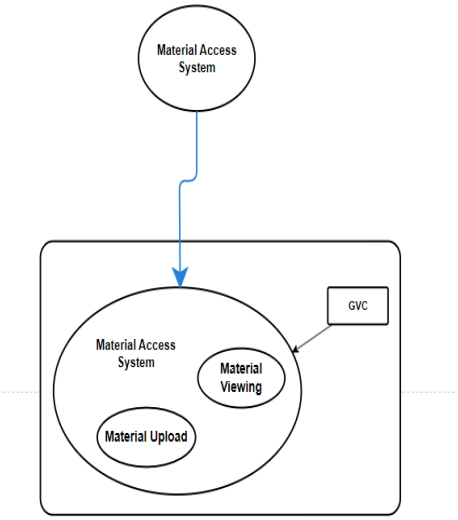


Figure 2.13: Material Access System

2.7.7 Assignment & Submission System:

The Assignment & Submission System in GVC enables students to submit assignments and manage submissions, while allowing teachers to review and evaluate them. This system supports seamless tracking and feedback for academic assignments within the virtual classroom.

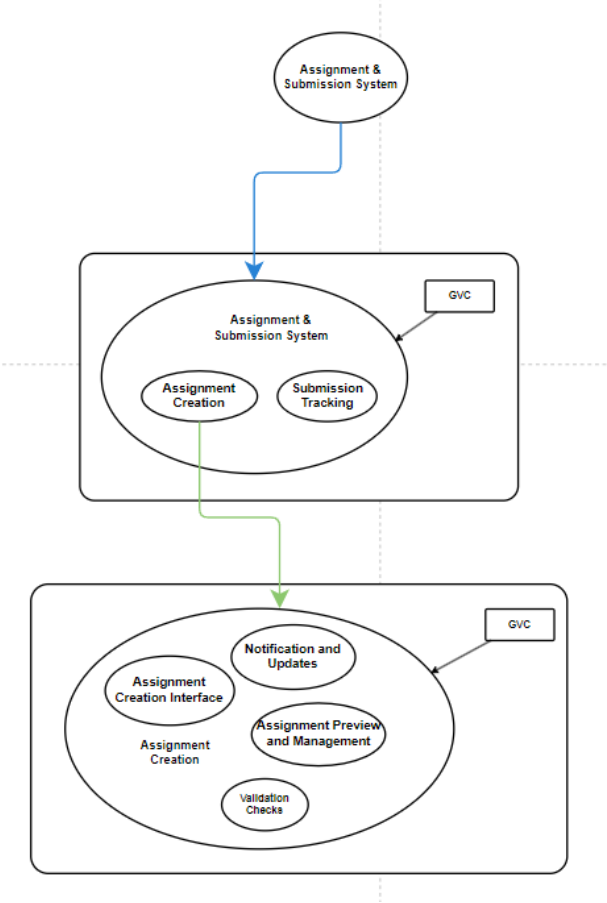


Figure 2.14: Assignment & Submission System

2.7.8 Discussion , Messaging & Post System:

The Discussion, Messaging & Post System enables group chat, group discussions, and dynamic content posting, allowing users to communicate and collaborate in real-time. It supports efficient information exchange and interaction within the Green Virtual Classroom environment.

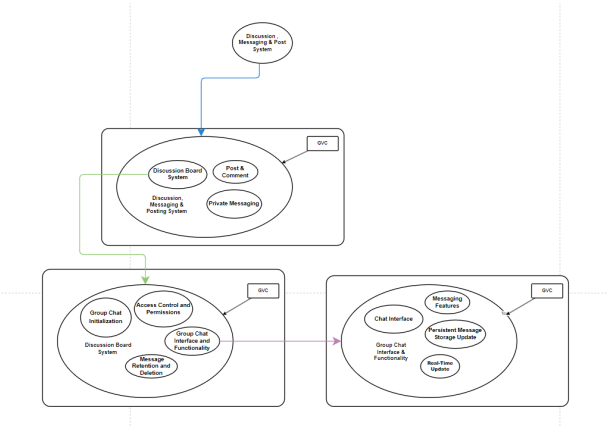


Figure 2.15: Discussion , Messaging & Post System

2.7.9 Notification System:

The Notification System manages data notifications and user alerts, keeping users informed of important updates and actions required. It helps ensure timely awareness of system activities within the Green Virtual Classroom.

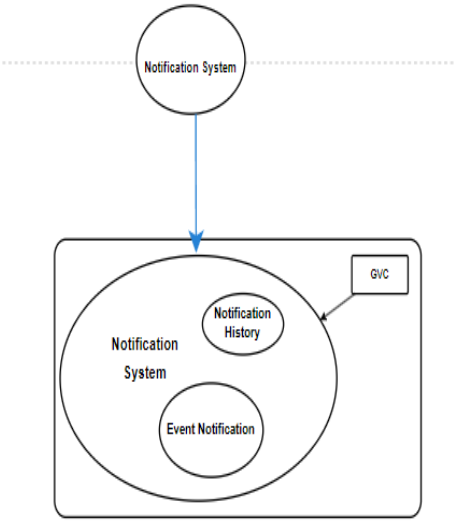


Figure 2.16: Notification System

2.8 Conclusion

The Green Virtual Classroom (GVC) project offers a user-friendly and efficient platform for online education, connecting students, teachers, and administrators. With features like course management, assessments, material sharing, and discussions, it ensures an engaging learning experience.

Built using tools like HTML, CSS, PHP, and MySQL, GVC meets performance, scalability,

and security requirements. Its modular design allows for easy maintenance and future enhancements, making it a reliable and adaptable solution for modern education.

Chapter 3

Interface Design and Implementation

This chapter provides a comprehensive overview of the interface design and implementation of the Green Virtual Classroom project. Each module is detailed with screenshots and step-by-step descriptions to illustrate design choices and functionalities.

3.1 User Authentication Module

3.1.1 Login Interface

Figure 3.1: Login Interface of the Green Virtual Classroom

The login interface allows users to access the platform by entering their credentials. The system verifies the inputs against stored data in the database. Key features include:

- Username and password fields with validation.
- `Forgot Password` link directing to the password recovery module.
- Error messages for invalid credentials.

The implementation utilizes PHP for backend processing and Bootstrap for styling. The backend ensures security through hashed password storage and validation.

3.1.2 Password Recovery Interface

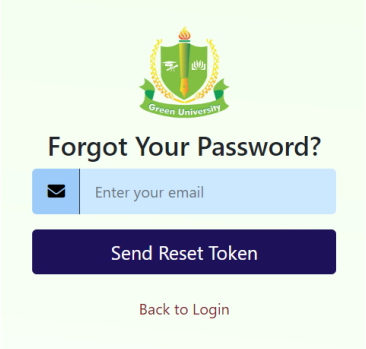


Figure 3.2: Password Recovery Interface 1

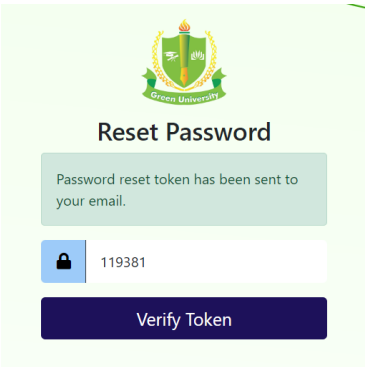


Figure 3.3: Password Recovery Interface 2

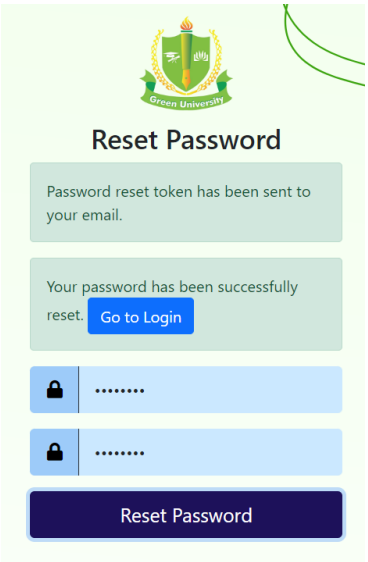


Figure 3.4: Password Recovery Interface 2

The password recovery interfaces provides users with an option to reset their password. Features include:

- Input field for the registered email address.
- Email verification and token generation for secure resets.
- Redirects to a secure password reset form.

3.2 Course Management Module

3.2.1 Course Dashboard

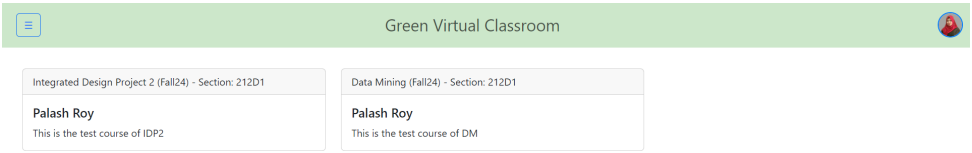


Figure 3.5: Course Dashboard

The course dashboard displays a list of courses available to the user. Functionalities include:

- Displaying course details such as name of instructor, section, course code, course title and description in a course card.
- Navigation to course stream, materials, people, notices, and discussion interfaces.
- Role-based access (student or teacher).

The implementation involves dynamic rendering using PHP and database queries to fetch user-specific data.

3.2.2 Stream Interface

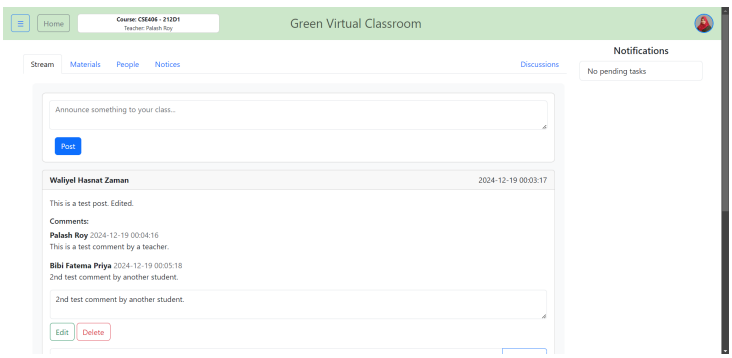


Figure 3.6: Stream Interface

The stream interface facilitates course-related announcements and interactions. Key features include:

- Creation of announcements and posts by instructors or students.
- Commenting, editing, or deleting posts based on user roles.
- Real-time updates to foster collaboration.

This interface is powered by PHP for dynamic updates and a responsive user experience.

3.2.3 Course Materials Interface

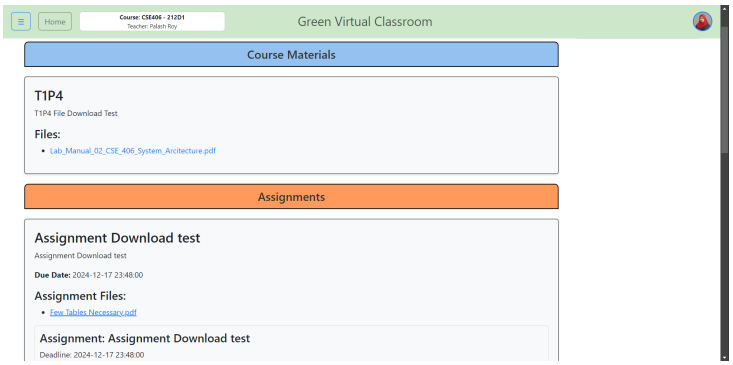


Figure 3.7: Course Materials Interface (Student)

This interface enables students to view and download course materials and submit assignments uploaded or created by teachers (shown in fig: 3.12 and fig: 3.13). It includes:

- Materials with downloadable File list.
- Assignments with upload attachments functionality and due date checking.
- Upload functionality for instructors (Materials and Assignments).

3.3 People Module

3.3.1 People Interface

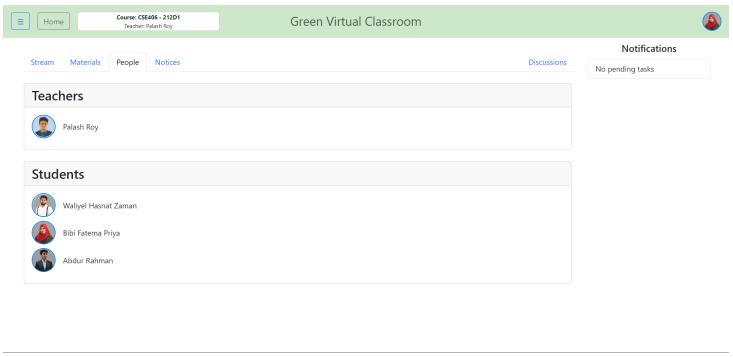


Figure 3.8: People Interface

This module allows users to view participants in a course. Features include:

- List of students and instructors with profile links.
- Role-based visibility of user details.

3.4 Notice Board Module

3.4.1 Notice Board Interface

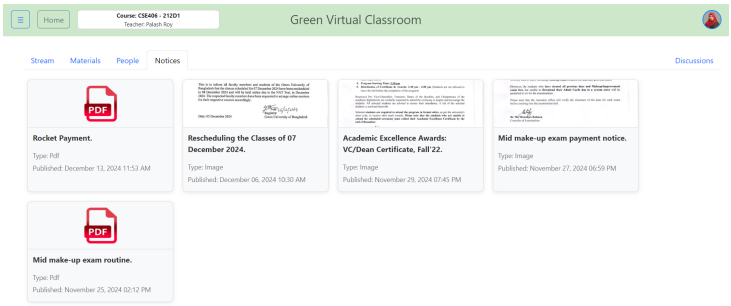


Figure 3.9: Notice Board Interface

The notice board provides important updates to users. Key functionalities include:

- Listing notices with timestamps.
- Filtering by category (e.g., image or pdf).
- Admin functionality to post notices.

3.5 Communication Module

3.5.1 Discussion Module

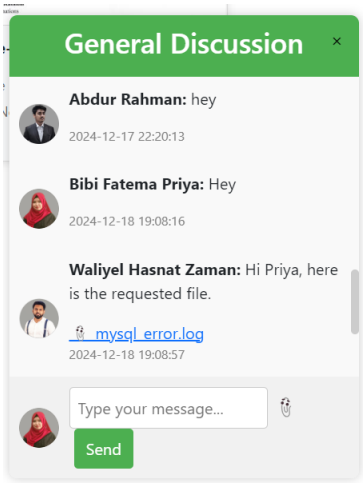


Figure 3.10: Discussion Module Interface

The discussion module fosters communication among users. Features include:

- Threaded discussions with reply functionality.
- Role-based moderation.
- Text and file input for messages and replies.

3.6 Help and Support Module

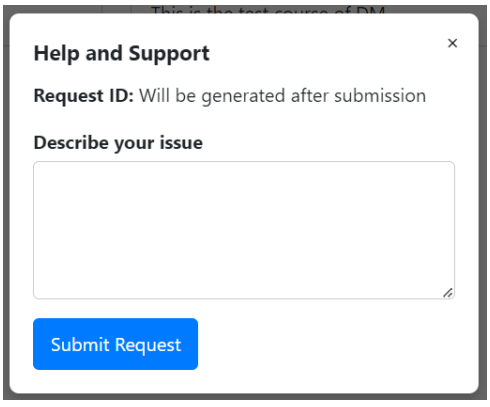


Figure 3.11: Help and Support Interface

The Help and Support module provides users with assistance and issue resolution. Features include:

- Submission of support tickets with auto-generated IDs.
- Email notifications using PHPMailer for updates.
- Status tracking for submitted tickets.

This module leverages PHP for backend operations and integrates email functionality to enhance communication.

3.7 Teacher Materials Dashboard

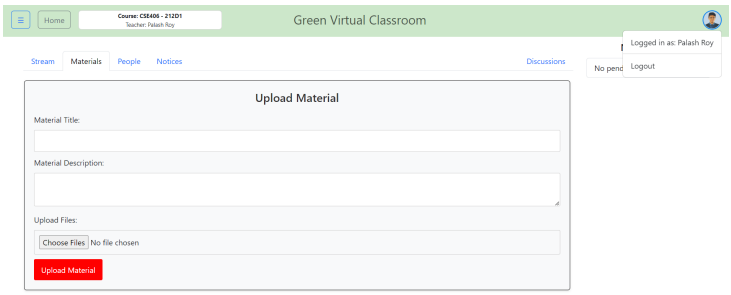


Figure 3.12: Teacher Materials Dashboard (Materials Upload)

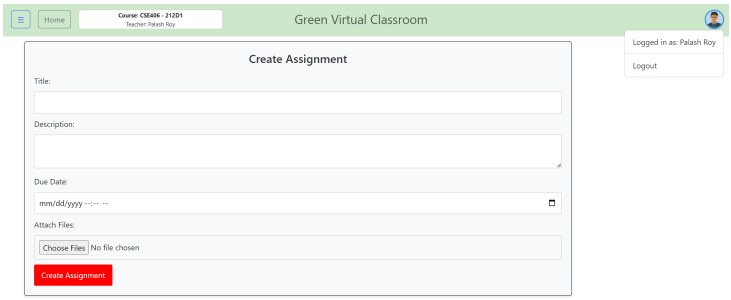


Figure 3.13: Teacher Materials Dashboard (Assignment Creation)

The teacher material dashboard provides tools for managing courses. Functionalities include:

- Upload of course materials.
- Post assignment and view submissions.
- Access other functionalities such as the stream page, discussions, help or support module, posted materials, and notices.

3.8 Conclusion

This chapter elaborated on the interface design and implementation of the Green Virtual Classroom, showcasing user-friendly designs and robust functionalities to enhance the virtual learning experience.

Chapter 4

Test Cases

4.1 Test Cases

4.1.1 Test Case 1(User)

For student accepts a valid numeric ID; errors for non-numeric input, errors for symbols, errors for spaces, and errors for special characters.

The teacher accepts only alphabetic characters with a specific special character '@' for domains; Error for usernames with numbers; errors if unsupported special characters are used, error for usernames containing spaces; error for usernames with unsupported symbols.

Accepts complex passwords with at least 8 characters that must contain a mix of letters, characters, and numbers.

ID: 212002022, 212002002 & 212002089					
User Table					
Test Case ID	Test_001	**Test Case Description**	Test the visitor table data and check validation		
Created By	Waliyel Hasnat Zaman	**Reviewed By**	Bibi Fatema Priya	**Version**	1
Tester's Name	Abdur Rahman	**Date Tested**		**Test Case (Pass/Fail/Not Executed)**	

Test Scenario 1	Validate that the Student ID field
S #	**Prerequisites:**
1	Access to GVC Website

S #	**Test Data**
1	Student ID = 212A02022
2	Student ID = @21200202
3	Student ID = 212002002
4	Student ID = 212 002 022
5	Student ID = 21200-2022

Step #	**Step Details**	**Expected Results**	**Actual Results**	**Pass / Fail / Not executed / Suspended**
1	Given, Student ID = 212A02022	Accepts valid numeric ID; Error for non-numeric input, error for symbols, error for spaces, error for special characters.		
2	Given, Student ID = @21200202			
3	Given, Student ID = 212002002			
4	Given,Student ID = 212 002 022			
5	Given, Student ID = 21200-2022			

Figure 4.1: Test case 1

ID: 212002022, 212002002 & 212002089					
User Table					
Test Case ID	Test_001	Test Case Description	Test the visitor table data and check validation		
Created By	Waliyel Hasnat Zaman	Reviewed By	Bibi Fatema Priya	Version	1
Tester's Name	Abdur Rahman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that the Student ID field
S #	Prerequisites:
1	Access to GVC Website

S #	Test Data
1	Student ID = 212A02022
2	Student ID = @21200202
3	Student ID = 212002002
4	Student ID = 212 002 022
5	Student ID = 21200-2022

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Given, Student ID = 212A02022	Accepts valid numeric ID; Error for non-numeric input, error for symbols, error for spaces, error for special characters.		
2	Given, Student ID = @21200202			
3	Given, Student ID = 212002002			
4	Given,Student ID = 212 002 022			
5	Given, Student ID = 21200-2022			

Figure 4.2: Test case 1

4.1.2 Test Case 2(Assignments)

Accepts a mix of letters an numbers; Error for empty title, only numbers, special characters, too long title greater than 30 characters.

Acceps any general texts, including empty string, letters, numbers and specialcharacters; Error for texts length greater than 150 characters.

Assignments Table					
Test Case ID	Test_001	Test Case Description	Test the visitor table data and check validation		
Created By	Waliyel Hasnat Zaman	Reviewed By	Bibi Fatema Priya	Version	1
Tester's Name	Abdur Rahman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that the Title field
S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	Integrated Design Project 2
2	""
3	jkhdjfdsjflasajfasdkljakdjsakjakhsjakhsajkljkhsajklsajklsklkas;lkokm;kjksm;kmxs
4	213213213213213213213
5	234-@*& *hkdhjk

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Integrated Design Project 2	Accepts a mix of letters an numbers; Error for empty title, only numbers, special characters, too long title greater than 30 characters.		
2	""			
3	jkhdjfdsjflasajfasdkljakdjsakjakhsjakhsajkljkhsajklsajk			
4	213213213213213213213			
5	234-@*& *hkdhjk			

Test Scenario 2	Validate that the Description field
S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	General text
2	""
3	kjfkjdhjopppjklfkldfhggiofjdjjdhdhfjf
4	23423jhjkhjkhjl@##4#
5	Too long text greater than 150 characters
6	

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Given, General text			

Figure 4.3: Test case 2

Test Scenario 3	Validate that the Due Date field

S #	Prerequisites:
1	Access to Course Page

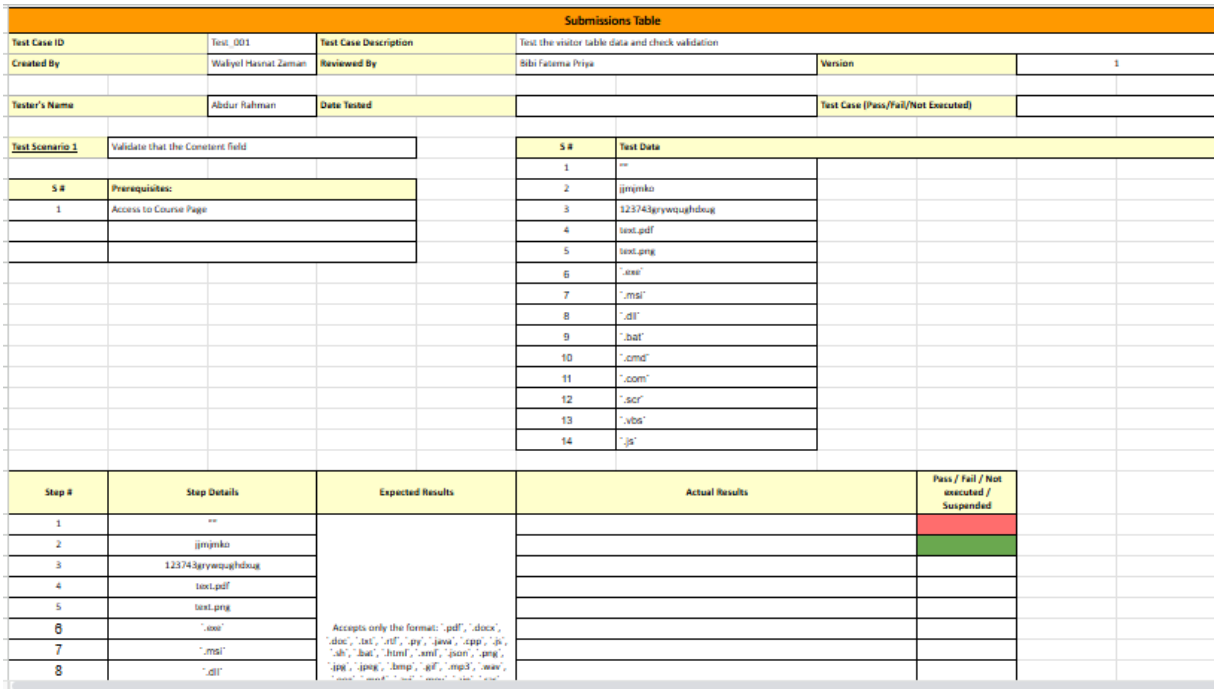
S #	Test Data
1	2024-11-15
2	15-11-2024
3	11-15-2024
4	15.12.1998
5	12/45/2004
6	2024-11-15
7	11/15/2024
8	2024-02-29
9	15/11/2024
10	2024-12-31
11	2024.11.15
12	2023-02-29
13	2024-11-32
14	15-11-2024
15	2024-13-01
16	Today
17	2024/11-15
18	2024-November-15
19	Yesterday
20	Next Monday

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	2024-11-15			
2	15-11-2024			
3	11-15-2024			
4	15.12.1998			
5	12/45/2004			
6	2024-11-15			
7	11/15/2024			

Figure 4.4: Test case 2

4.1.3 Test Case 3(Submissions)

Accepts only the format: ‘.pdf‘, ‘.docx‘, ‘.doc‘, ‘.txt‘, ‘.rtf‘, ‘.py‘, ‘.java‘, ‘.cpp‘, ‘.js‘, ‘.sh‘, ‘.bat‘, ‘.html‘, ‘.xml‘, ‘.json‘, ‘.png‘, ‘.jpg‘, ‘.jpeg‘, ‘.bmp‘, ‘.gif‘, ‘.mp3‘, ‘.wav‘, ‘.ogg‘, ‘.mp4‘, ‘.avi‘, ‘.mov‘, ‘.zip‘, ‘.rar‘, ‘.tar.gz‘; Any other format will result in error.



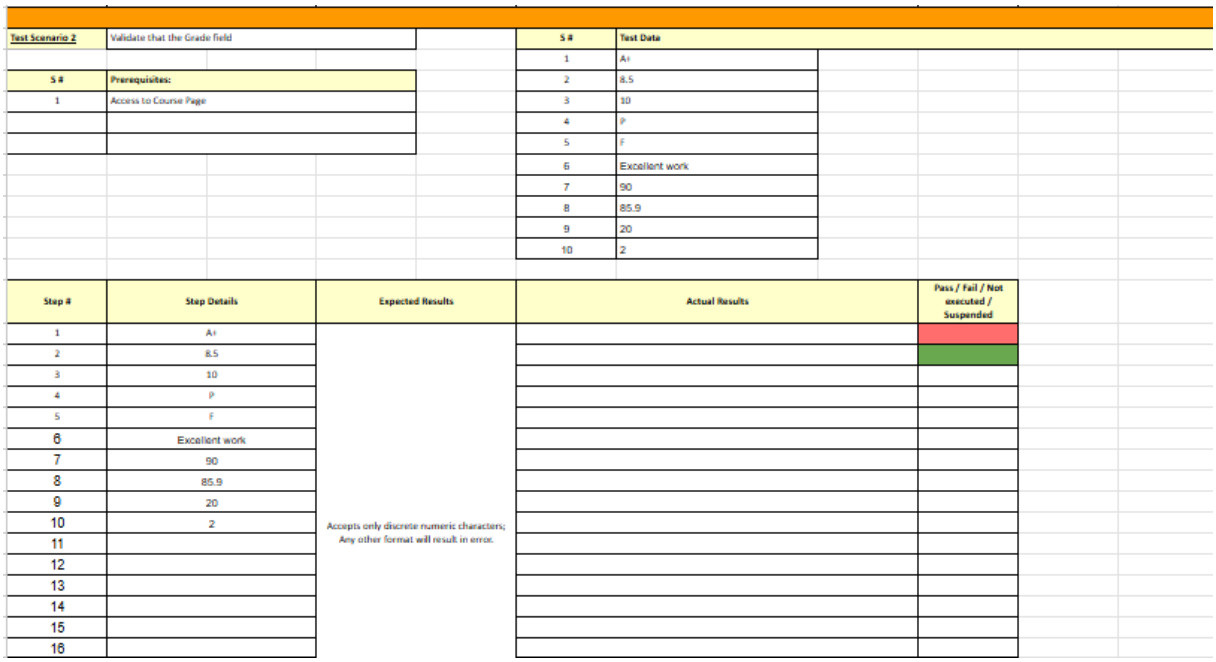
Submissions Table					
Test Case ID	Test_001	Test Case Description	Test the visitor table data and check validation		
Created By	Wahyel Hasnat Zaman	Reviewed By	Bibi Fatema Priya	Version	1
Tester's Name	Abdur Rahman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that the Conetent field
S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	""
2	jjmjmko
3	123743jryrwquqhdxug
4	text.pdf
5	text.png
6	".exe"
7	".msi"
8	".dll"
9	".bat"
10	".cmd"
11	".com"
12	".scr"
13	".vbs"
14	".js"

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	""	Accepts only the format: ".pdf", ".docx", ".doc", ".txt", ".rtf", ".py", ".java", ".cpp", ".js", ".sh", ".bat", ".html", ".xml", ".json", ".png", ".jpg", ".jpeg", ".bmp", ".gif", ".mp3", ".wav",		
2	jjmjmko			
3	123743jryrwquqhdxug			
4	text.pdf			
5	text.png			
6	".exe"			
7	".msi"			
8	".dll"			

Figure 4.5: Test case 3



Test Scenario 2	Validate that the Grade field
S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	A+
2	8.5
3	10
4	P
5	F
6	Excellent work
7	90
8	85.9
9	20
10	2

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	A+	Accepts only discrete numeric characters; Any other format will result in error.		
2	8.5			
3	10			
4	P			
5	F			
6	Excellent work			
7	90			
8	85.9			
9	20			
10	2			
11				
12				
13				
14				
15				
16				

Figure 4.6: Test case 3

Test Scenario 3	Validate that the feedback field

S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	Excellent work! Your analysis is thorough and well-structured
2	Check this out: www.example.com
3	.pdf, .docx, .jpg, .mp4
4	😊👍🔥
5	@#$%^&
6	Buy my product at www.example.com!
7	Good attempt, but you need to focus more on optimizing your solution. Consider using a more efficient algorithm for this problem
8	<script>alert('Hacked!');</script>
9	Your submission is incomplete. Please ensure all required sections are included.
10	Focus on improving your report formatting. Use bullet points or headings to make it easier to read

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	2024-11-15	Accepts only textual feedback; Any other format will result in error.		
2	15-11-2024			
3	11-15-2024			
4	15.12.1998			
5	12/45/2004			
6	2024-11-15			
7	11/15/2024			
8	2024-02-29			
9	15/11/2024			
10	2024-12-31			
11	2024.11.15			

Figure 4.7: Test case 3

Test Scenario 3	Verify successful file submission by a student

S #	Prerequisites:
1	The student is logged in.
2	An assignment has been created and is visible in the course portal.

S #	Test Data
1	Click 'Turn in'
2	
3	
4	
5	
6	
7	
8	
9	
10	

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Click 'Turn in'	Submitted file stred in the database, within the time given.		
2				
3				
4				
5				
6				
7				
8				
9				
10				
11				
12				
13				
14				
15				
16				

Test 1 | Test 2 | Test 3 | Test 4 | Test 5 | Test 6 | Test 7 | Test 8 | Test 9 | Test 10 | Test 11 | Test 12 | Test 13

Figure 4.8: Test case 3

4.1.4 Test Case 6(Courses)

Accepts only the format ABC123 or ABCD123; otherwise, give error.

ACCEPTS ANY TYPE OF TEXT AND NUMBER COMBINATION

Courses Table					
Test Case ID	Test_001	**Test Case Description**	Test the Courses table data and check validation		
Created By	Wahyel Hasnat Zaman	**Reviewed By**	Bibi Fatema Priya	**Version**	1
Tester's Name	Abdur Rahman	**Date Tested**		**Test Case (Pass/Fail/Not Executed)**	

Test Scenario 1	Validate that the Course Code field
S #	**Prerequisites:**
1	Access to admin Panel

S #	**Test Data**
1	CSE324
2	GED 405
3	678678678
4	BBA 103
5	TEX
6	AIDS101

Step #	**Step Details**	**Expected Results**	**Actual Results**	**Pass / Fail / Not executed / Suspended**
1	CSE324	Accepts only the format ABC123 or ABCD123; Otherwise, Give error.		
2	GED 405			
3	678678678			
4	BBA 103			
5	TEX			
6	AIDS101			

Test Scenario 2	Validate that Title field
S #	**Prerequisites:**
1	Access to Admin Panel

S #	**Test Data**
1	Data Mining
2	Structured Programming
3	abcdefg@E43r56ty7
4	265732284502259 36
5	@819731083019
6	-wyougyrAW26187

Figure 4.9: Test case 6

Test Scenario 2	Validate that description field
S #	**Prerequisites:**
1	Access to Admin Panel

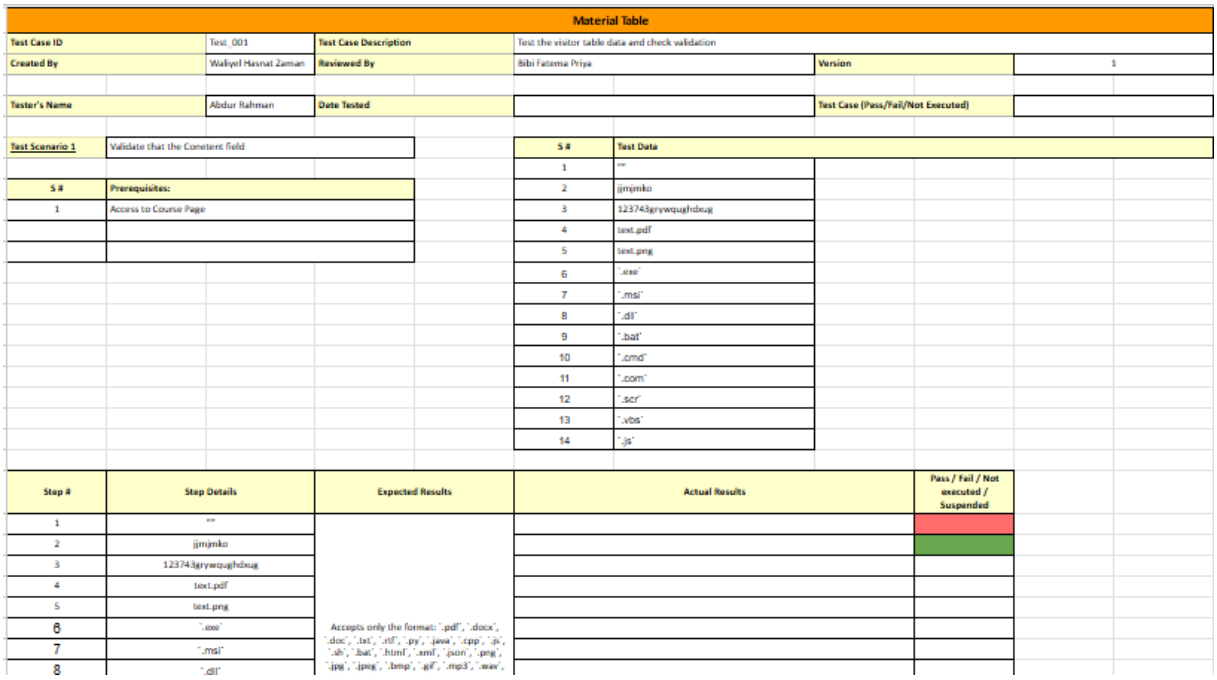
S #	**Test Data**
1	Data Mining
2	Structured Programming
3	abcdefg@E43r56ty7
4	265732284502259 36
5	@819731083019
6	-wyougyrAW26187
7	11/15/2024
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	

Step #	**Step Details**	**Expected Results**	**Actual Results**	**Pass / Fail / Not executed / Suspended**
1	Data Mining			
2	abcdefg@E43r56ty7			
3	265732284502259 36			
4	@819731083019			
5	-wyougyrAW26187			
6	11/15/2024			

Figure 4.10: Test case 6

4.1.5 Test Case 7(Material)

Accepts only the format: ‘.pdf‘, ‘.docx‘, ‘.doc‘, ‘.txt‘, ‘.rtf‘, ‘.py‘, ‘.java‘, ‘.cpp‘, ‘.js‘, ‘.sh‘, ‘.bat‘, ‘.html‘, ‘.xml‘, ‘.json‘, ‘.png‘, ‘.jpg‘, ‘.jpeg‘, ‘.bmp‘, ‘.gif‘, ‘.mp3‘, ‘.wav‘, ‘.ogg‘, ‘.mp4‘, ‘.avi‘, ‘.mov‘, ‘.zip‘, ‘.rar‘, ‘.tar.gz‘; Any other format will result in error.



Material Table

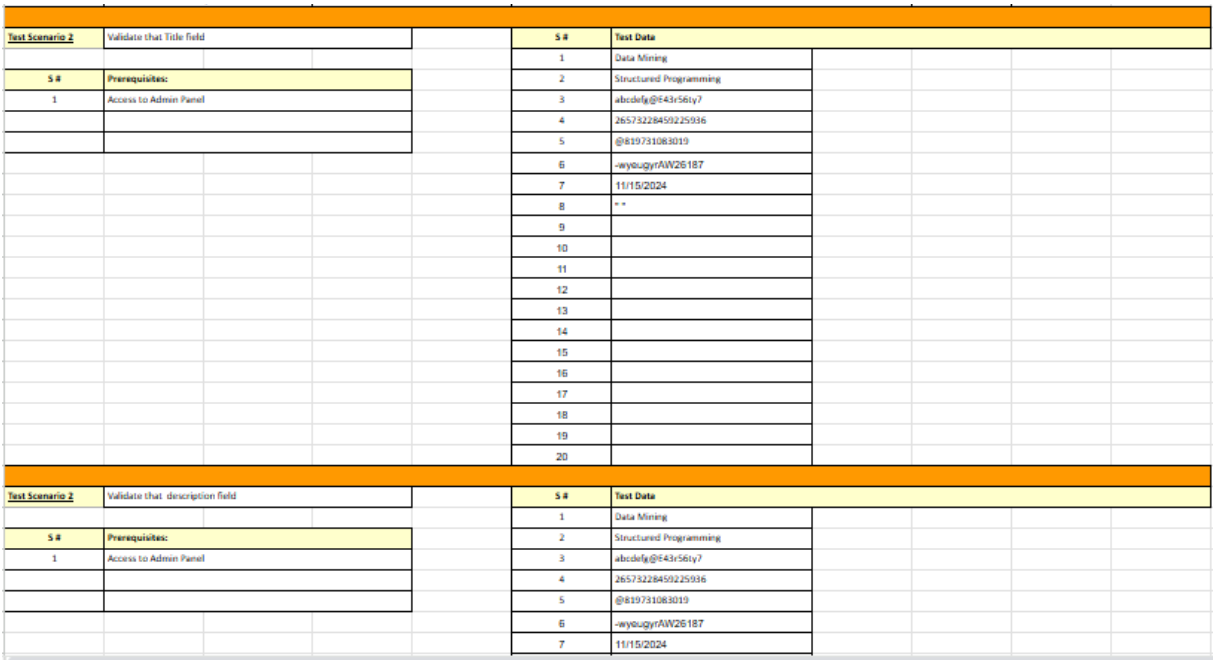
Test Case ID	Test_001	Test Case Description	Test the visitor table data and check validation		
Created By	Wahyel Hasnat Zaman	Reviewed By	Bibi Fatema Priya	Version	1
Tester's Name	Abdur Rahman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that the Conetent field
S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	""
2	jjmjmko
3	123743gryvwquqjhdxug
4	text.pdf
5	text.png
6	`.exe`
7	`.msi`
8	`.dll`
9	`.bat`
10	`.cmd`
11	`.com`
12	`.scr`
13	`.vbs`
14	`.js`

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	""	Accepts only the format: '.pdf', '.docx', '.doc', '.txt', '.rtf', '.py', '.java', '.cpp', '.js', '.sh', '.bat', '.html', '.xml', '.json', '.png', '.jpg', '.jpeg', '.bmp', '.gif', '.mp3', '.wav',		
2	jjmjmko			
3	123743gryvwquqjhdxug			
4	text.pdf			
5	text.png			
6	`.exe`			
7	`.msi`			
8	`.dll`			

Figure 4.11: Test case 7



Test Scenario 2	Validate that Title field
S #	Prerequisites:
1	Access to Admin Panel

S #	Test Data
1	Data Mining
2	Structured Programming
3	abcdefg@643r56ty7
4	26573228459225936
5	@819731083019
6	-wyougyrAW26187
7	11/15/2024
8	" "
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	

Test Scenario 2	Validate that description field
S #	Prerequisites:
1	Access to Admin Panel

S #	Test Data
1	Data Mining
2	Structured Programming
3	abcdefg@643r56ty7
4	26573228459225936
5	@819731083019
6	-wyougyrAW26187
7	11/15/2024

Figure 4.12: Test case 7

4.1.6 Test Case 8(Posts)

Acceps any general texts, letters, numbers, specialcharacters & BLOB contents; an error for empty string, text length greater than 150 characters, invalid font types not supported in BLOB.

If posts done by other's are visible, posts successfully posted.

Posts Table					
Test Case ID	Test_001	Test Case Description	Test the Courses table data and check validation		
Created By	Abdur Rahman	Reviewed By	Bibi Fatema Priya	Version	1
Tester's Name	Wahyel Hasnat Zaman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that the Post content field
S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	General text
2	""
3	kjfkjdhjoppjklfkldfhggofjdjjdhdhfjf
4	23423jhjkhjkhjl@##4#
5	Too long text greater than 150 characters
6	BLOB Contents

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Given, General text	Acceps any general texts, letters, numbers, specialcharacters & BLOB Contents; Error for empty string, texts length greater than 150 characters, invalid file types not supported in BLOB.		
2	Given, ""			
3	Given, kjfkjdhjoppjklfkldfhggofjdjjdhdhfjf			
4	Given, 23423jhjkhjkhjl@##4#			
5	Given, Too long text greater than 150character			
6	Given, BLOB Contents			

Test Scenario 2	Validate that if a post is successfully posted.
S #	Prerequisites:
1	Access to Course Portal
2	Access to Stream Page

S #	Visuals
1	Posts by others
2	
3	
4	
5	
6	

Figure 4.13: Test case 8

4.1.7 Test Case 9(Comments)

Acceps any general texts, letters, numbers, specialcharacters & BLOB contents; error for empty string, text length greater than 150 characters, invalid font types not supported in BLOB.

If comments done by me & other's are visible, comments are successfully posted.

Comments Table					
Test Case ID	Test_001	Test Case Description	Test the Courses table data and check validation		
Created By	Bibi Fatema Priya	Reviewed By	Wahyel Hasnat Zaman	Version	1
Tester's Name	Abdur Rahman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 2	Validate that the Post content field
S #	Prerequisites:
1	Access to Course Page

S #	Test Data
1	General text
2	""
3	kjfkjdhsjoppjklfkldfhggofjdjjdhdhfjf
4	23423jhjkhjkhjl@#4$4$#
5	Too long text greater than 150 characters
6	BLOB Contents

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Given, General text	Acceps any general texts, letters, numbers, specialcharacters & BLOB Contents; Error for empty string, texts length greater than 150 characters, invalid fle types not supported in BLOB.		
2	Given, ""			
3	Given, kjfkjdhsjoppjklfkldfhggofjdjjdhdhfjf			
4	Given, 23423jhjkhjkhjl@#4$4$#			
5	Given, Too long text greater than 150character			
6	Given, BLOB Contents			

Test Scenario 2	Validate that if a coments is successfully posted.
S #	Prerequisites:
1	Read the cmments
2	Read the cmments from outhers account

S #	Visuals
1	Comments by me & others
2	
3	
4	
5	
6	

Figure 4.14: Test case 9

4.1.8 Test Case 10(Discussions)

Acceps any general texts, letters, numbers, specialcharacters & BLOB Contents, successful if the discussion messages can be seen from every box of all the participants in a course; Error for empty string, text length greater than 150 characters, invalid fle types not supported in BLOB.

Discussions Table					
Test Case ID	Test_001	Test Case Description	Test the Courses table data and check validation		
Created By	Abdur Rahman	Reviewed By	Bibi Fatema Priya	Version	1
Tester's Name	Wahyel Hasnat Zaman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that message is being passed in the discussion box
S #	Prerequisites:
1	Access to Course Page
2	Access to discussion box

S #	Test Data
1	General text
2	""
3	kjfkjdhsjoppjklfkldfhggofjdjjdhdhfjf
4	23423jhjkhjkhjl@#4$4$#
5	Too long text greater than 150 characters
6	BLOB Contents

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Given, General text	Acceps any general texts, letters, numbers, specialcharacters & BLOB Contents, successfull if the discussion messages can be seen from every box of all the participants in a course; Error for empty string, texts length greater than 150 characters, invalid fle types not supported in BLOB.		
2	Given, ""			
3	Given, kjfkjdhsjoppjklfkldfhggofjdjjdhdhfjf			
4	Given, 23423jhjkhjkhjl@#4$4$#			
5	Given, Too long text greater than 150character			
6	Given, BLOB Contents			

Figure 4.15: Test case 10

4.1.9 Test Case 11(Messages)

Acceps any general texts, letters, numbers, specialcharacters & BLOB Contents, successfully if the discussion messages can be seen from every box of all the participants in a course; Error for empty string, text length greater than 150 characters, invalid fle types not supported in BLOB.

Messages Table					
Test Case ID	Test_001	Test Case Description	Test the Courses table data and check validation		
Created By	Bibi Fatema Priya	Reviewed By	Wahyel Hasnat Zaman	Version	1
Tester's Name	Abdur Rahman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that message is being passed in the message box
S #	Prerequisites:
1	Access to Course Page
2	Access to message box

S #	Test Data
1	General text
2	""
3	kjfkjdlsjopppklflkldfhggjofjdjjdhdhfjf
4	23423jhjkhjkhjl@#4$4$#
5	Too long text greater than 150 characters
6	BLOB Contents

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Given, General text	Acceps any general texts, letters, numbers, specialcharacters & BLOB Contents, successfull if the discussion messages can be seen from every box of all the participants in a course; Error for empty string, texts length greater than 150 characters, invalid fle types not supported in BLOB.		
2	Given, ""			
3	Given, kjfkjdlsjopppklflkldfhggjofjdjjdhdhfjf			
4	Given, 23423jhjkhjkhjl@#4$4$#			
5	Given, Too long text greater than 150character			
6	Given, BLOB Contents			

Figure 4.16: Test case 11

4.1.10 Test Case 12(Notifications)

Check if notifications are visible or not; error for non-visible notifications.

Notifications Table					
Test Case ID	Test_001	Test Case Description	Test the Courses table data and check validation		
Created By	Abdur Rahman	Reviewed By	Bibi Fatema Priya	Version	1
Tester's Name	Wahyel Hasnat Zaman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Validate that notifications can be seen in the notification box.
S #	Prerequisites:
1	Access to Course Page
2	

S #	Visuals
1	Pending course work notifications.
2	
3	
4	
5	
6	

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Pending course work notifications.	Check if the notifications are visible or not; error for not visible notifctions.		
2				
3				
4				
5				
6				

Figure 4.17: Test case 12

4.1.11 Test Case 13(Notices)

Check if the notices are visible or not; error for not visible notice.

Notices Table					
Test Case ID	Test_001	Test Case Description	Test the Courses table data and check validation		
Created By	Bibi Fatema Priya	Reviewed By	Wahyel Hasnat Zaman	Version	1
Tester's Name	Abdur Rahman	Date Tested		Test Case (Pass/Fail/Not Executed)	

Test Scenario 1	Notices can be seen in the notic page.
S #	Prerequisites:
1	Viesulas the notice
2	

S #	Test Data
1	
2	
3	
4	
5	
6	

Step #	Step Details	Expected Results	Actual Results	Pass / Fail / Not executed / Suspended
1	Notice viesulas in notice page	Check if the notice are visible or not; error for not visible notice.		
2				
3				
4				
5				
6				

Figure 4.18: Test case 13

Chapter 5

Conclusion and Future Work

5.1 Summary of the Project

The Green Virtual Classroom (GVC) is an interactive online learning platform designed to enhance higher education. It connects students, teachers, and administrators through features like course management, material sharing, assessments, discussions, and feedback systems. Built using tools such as HTML, CSS, PHP, MySQL, and Bootstrap, the system ensures scalability, security, and usability. GVC aims to provide a user-friendly, efficient, and adaptable solution to meet the evolving needs of virtual education.

5.2 Future Works

The Green Virtual Classroom (GVC) project has significant potential for further enhancement. Future work includes:

- Integrate advanced plagiarism-checking tools to ensure academic integrity in assignments and quizzes.
- Strengthen system security with advanced measures like multi-factor authentication, real-time threat monitoring, and secure data encryption.
- Implement a private messaging system to facilitate one-on-one communication between students, teachers, and administrators.
- Seamlessly integrate GVC with existing university student portals for centralized access to grades, enrollment, and course updates.
- Add AI tools for analyzing student performance and predicting learning outcomes.
- Develop a mobile app version to increase accessibility and usability for users on the go.

References

[1] Google LLC. Google classroom, 2014.

[2] Microsoft Corporation. Microsoft teams, 2017.

[3] Moodle Pty Ltd. Moodle, 2002.

[4] Instructure, Inc. Canvas learning management system, 2011.