**SUBJECT:**

**WEB DEVELOPMENT**



**SUBMITTED BY:**

**MUHAMMAD BILAL SALEEM**

**JAWAD AHMED**

**ROLL NO:**

**23-SET-016**

**23-SET-007**

**SUBMITTED TO:**

**SIR WAHAB AHMAD**

**DEPARTMENT OF SOFTWARE ENGINEERING TECHNOLOGY**



# PROJECT: CodeHub Documentation.

## 1. Introduction

### 1.1 Purpose

This document specifies the requirements for **CodeHub**, a course selling platform designed for users to access online courses. The purpose of CodeHub is to offer a centralized website where users can register, log in, and directly access courses that include video lectures and downloadable materials. The document defines the system's functionality, features, and constraints for development and future enhancements.

### 1.2 Scope

CodeHub is a web-based application built using HTML, CSS, JavaScript, and Bootstrap. Its primary functions include:

* User registration and login
* Course listing with detailed information (duration, instructor details, prerequisites, ratings/reviews)
* Course search and filter functionality
* Direct course access (streaming video lectures and downloading supporting materials)

Future enhancements include an administrative interface for course management (adding, editing, deleting course details).

### 1.3 Overview

This document details the system’s functional and non-functional requirements, overall system description, and design constraints. It is intended for stakeholders, developers, and testers involved in the project.

## 2. Overall Description

### 2.1 Product Perspective

CodeHub is a stand-alone web application that is designed to be simple and user-friendly. Although initially built as a local project, it mimics functionalities of commercial course platforms. The system is modular, allowing for future integration of additional features (e.g., an administrative interface for course management).

### 2.2 Product Functions

* **User Authentication:**

Users can register and log in using a unified login module.

* **Course Management and Display:**
  + Courses are displayed on a dedicated “Courses” page.
  + Each course includes details such as course duration, instructor information, prerequisites, and user ratings/reviews.
  + A search and filter functionality allows users to quickly find courses by topic (e.g., cybersecurity, graphic designing, web development, software engineering).
* **Content Delivery:**  
  Courses are delivered through:
  + Video lectures (streaming directly on the website)
  + Downloadable materials (supporting documents, slides, etc.)
* **Navigation:**  
  The website consists of four basic pages:
  + Home
  + About
  + Courses
  + Contact Us

## 3. Specific Requirements

### 3.1 Functional Requirements

**3.1.1 User Authentication and Registration**

* **FR1:** The system shall allow new users to register by providing basic information (username, email, password).
* **FR2:** The system shall allow registered users to log in using their credentials.
* **FR3:** The system shall provide a logout option for authenticated users.

**3.1.2 Course Listing and Details**

* **FR4:** The system shall display a list of available courses on the “Courses” page.
* **FR5:** Each course entry shall display:
  + Course Title
  + Course Duration
  + Instructor Details
  + Prerequisites
  + Ratings and Reviews
* **FR6:** The system shall allow users to click on a course to view detailed information.

**3.1.3 Search and Filter Functionality**

* **FR7:** The system shall provide a search functionality that allows users to search courses by title or keywords.
* **FR8:** The system shall provide filters to sort courses based on categories such as cybersecurity, graphic designing, web development, and software engineering.

**3.1.4 Content Delivery**

* **FR9:** The system shall allow users to access video lectures directly on the website.
* **FR10:** The system shall allow users to download supplementary materials related to the courses.

### 3.2 Non-Functional Requirements

**3.2.1 Usability**

* **NFR1:** The user interface shall be simple and intuitive, utilizing Bootstrap for responsive design.
* **NFR2:** Navigation across the four basic pages (Home, About, Courses, Contact Us) shall be straightforward.

**3.2.2 Performance**

* **NFR3:** As a test website, performance requirements are minimal; however, the system should load pages within a reasonable time on modern browsers.

**3.2.3 Security**

* **NFR4:** The system shall implement basic security measures for user authentication.
* **NFR5:** Sensitive data (such as passwords) shall be handled with basic encryption methods (not intended for production-level security).

**3.2.4 Compatibility**

* **NFR6:** The system shall be compatible with modern web browsers.
* **NFR7:** The system shall support various screen sizes through responsive design techniques.

**3.2.5 Scalability and Extensibility**

* **NFR8:** The codebase should be modular to allow easy addition of future features (e.g., the admin interface, potential payment gateway integration).

## 4. Use cases

### 4.1 Use Case 1: User Registration and Login

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Use Case Title** | User Registration and Login |
| **Primary Actor** | User |
| **Actions** | * User enters registration details (username, email, password). * System validates details and creates a new account. * User logs in using credentials. |
| **Description** | A new user creates an account on CodeHub and logs in to access course content. |
| **Alternative Paths** | * User may choose to reset a forgotten password if login fails. * User may opt to login directly if already registered. |
| **Pre-conditions** | * User is not already registered (for registration). * User is on the login/registration page. |
| **Post-conditions** | A new account is created, or an existing user session is started. |
| **Exceptions** | * Required fields are missing or invalid. * Incorrect login credentials provided. |

### 4.2 Use Case 2: Course Browsing and Searching

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Use Case Title** | Course Browsing and Searching |
| **Primary Actor** | Registered User |
| **Actions** | * User navigates to the "Courses" page. * User uses the search bar and filter options to locate courses by title, category (cybersecurity, graphic designing, etc.). |
| **Description** | A registered user browses through the list of available courses and filters them based on interests or keywords. |
| **Alternative Paths** | User may browse courses by navigating through predefined categories instead of using the search functionality. |
| **Pre-conditions** | User is logged in and has accessed the Courses page. |
| **Post-conditions** | Relevant courses matching the search/filter criteria are displayed. |
| **Exceptions** | * No courses match the search criteria. * Search functionality fails to load. |

### 4.3 Use Case 3: Course Access and Content Delivery

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Use Case Title** | Course Access and Content Delivery |
| **Primary Actor** | Registered User |
| **Actions** | * User selects a course from the Courses page to view its detailed information. * User accesses video lectures through embedded streaming and downloads supplementary materials. |
| **Description** | A user accesses a selected course, views its content (video lectures), and downloads available course materials for offline study. |
| **Alternative Paths** | If a course fails to load, the user may refresh the page or select another course. |
| **Pre-conditions** | * User is authenticated. * The selected course exists in the system. |
| **Post-conditions** | The course content (video lectures and downloadable materials) is successfully delivered to the user. |
| **Exceptions** | * Streaming or download errors occur. * Course content is temporarily unavailable. |

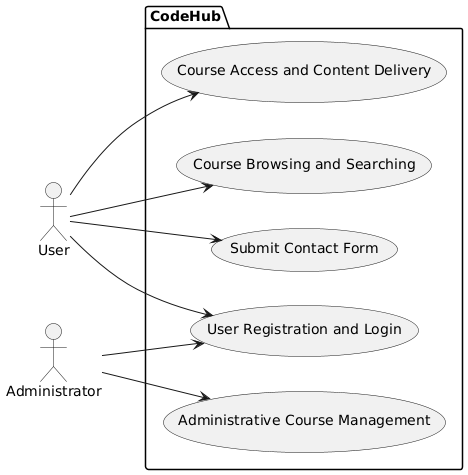
### 4.4 Use Case 4: Administrative Course Management (Planned Enhancement)

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Use Case Title** | Administrative Course Management |
| **Primary Actor** | Administrator (Planned) |
| **Actions** | * Admin logs in and accesses the administrative dashboard. * Admin adds, edits, or deletes course details (including duration, instructor info, prerequisites, and ratings/reviews). |
| **Description** | An administrator manages the course listings by updating course information and ensuring that the course repository is current and accurate. |
| **Alternative Paths** | * Admin may choose to revert changes if needed. * Alternatively, the admin can preview changes before finalizing them. |
| **Pre-conditions** | Admin is authenticated and has necessary permissions. |
| **Post-conditions** | The course repository is updated with new or modified course details. |
| **Exceptions** | * Insufficient permissions to perform the action. * Duplicate or invalid course details provided. |

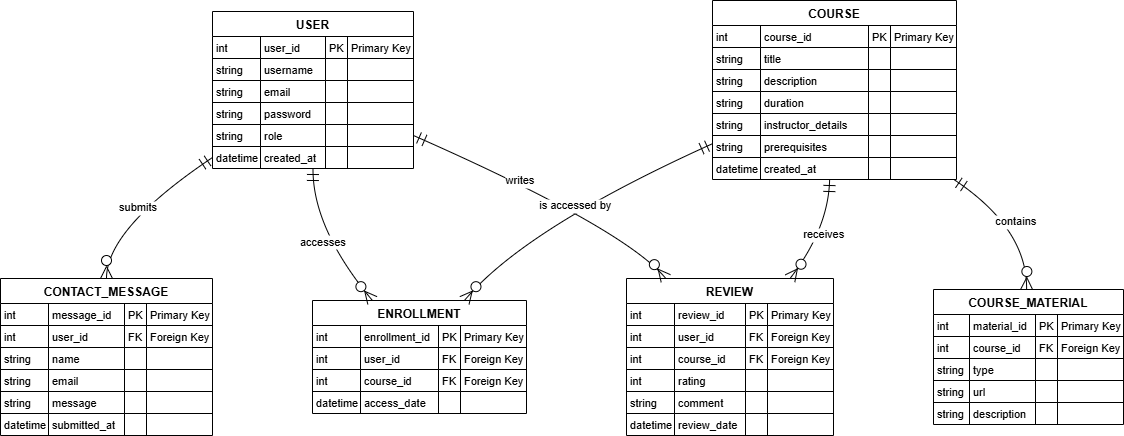
### 4.5 Use Case 5: Submit Contact Form

|  |  |
| --- | --- |
| **Field** | **Details** |
| **Use Case Title** | Submit Contact Form |
| **Primary Actor** | Visitor/User |
| **Actions** | * User navigates to the "Contact Us" page. * User fills in the contact form with required information (name, email, message). * User submits the form. |
| **Description** | A visitor or user submits a query, feedback, or support request via the Contact Us page. |
| **Alternative Paths** | User may also choose to contact via direct email if provided. |
| **Pre-conditions** | User is on the Contact Us page. |
| **Post-conditions** | The contact form data is successfully submitted and queued for response by the system administrator. |
| **Exceptions** | * Form submission fails due to missing required fields or technical errors. * Invalid data entered in the form. |

## 5. Use case diagram



## 6. ERD diagram



## 7. sequence diagram