**COMSATS University Islamabad**

A blue and white logo

Description automatically generated**Abbottabad Campus**

Assignment NO 1:

Student Name**: Fizza Razzaq**

Registration Number: **FA22-BCE-100**

Class: **6A**

Course: CSC291 Software Engineering Concepts

Instructor: Dr. Ghulam Mujtaba

Date: **29-March -2025**

**Department of Computer Engineering**

**COMSATS University Islamabad, Abbottabad**

**Campus**

**1. Introduction**

**1.1 Purpose**

The purpose of this document is to present a detailed Software Requirements Specification (SRS) for the development of an Online Course Enrollment Platform. This platform is designed to streamline the process of course selection, registration, and payment for students through a user-friendly web interface.

**1.2 Scope**

The Online Course Enrollment Platform will allow students to register an account, browse available courses, enroll in selected courses, and make simple online payments using basic payment methods like mobile banking or uploading proof of bank transfer. It will also provide administrative capabilities for instructors and administrators to manage course offerings and enrollment records. The system will be developed as a basic web application using HTML, CSS, JavaScript, and a simple backend database like MySQL with PHP or a Google Form alternative.

**1.3 Definitions, Acronyms, and Abbreviations**

* **SRS**: Software Requirements Specification
* **UI**: User Interface
* **UX**: User Experience
* **HTTP**: Hypertext Transfer Protocol
* **DBMS**: Database Management System
* **OTP**: One-Time Password

**1.4 References**

* IEEE SRS Standard (IEEE 830-1998)
* HTML, CSS, JavaScript documentation
* MySQL and PHP tutorials
* Google Forms and Google Sheets integration documentation (optional)

**2. Stakeholder Analysis**

**2.1 Key Stakeholders**

* **Students:** Primary users who will browse, enroll in, and pay for courses.
* **Instructors:** Manage course content and track student enrollment.
* **Administrators:** Manage users, monitor enrollments, and oversee course offerings.
* **Technical Team:** Developers and testers involved in building and maintaining the platform.

**3. Requirements Gathering**

**3.1 Techniques Used**

1. **Interviews:** Conducted with students and instructors to understand enrollment pain points.
2. **Questionnaires:** Distributed to academic staff and students to identify desired features.
3. **Brainstorming:** Sessions held with the development team to finalize core modules.
4. **Observation:** Reviewed existing manual registration processes in educational institutes.

**4. Overall Description**

**4.1 Product Perspective**

The Online Course Enrollment Platform will be a basic standalone web-based system that interacts with a simple backend database. It is designed to replace manual and spreadsheet-based registration systems in educational institutions. The application will include modules for student interaction, instructor management, and administrative oversight.

**4.2 Product Functions**

* Student registration and login with basic email verification
* Course search and filtering
* Course enrollment (single or multiple courses)
* Payment via mobile banking upload or confirmation code entry
* Profile and enrollment history management
* Admin dashboard for managing courses, enrollments, and users

**4.3 Assumptions and Dependencies**

* Users must have internet access and a modern web browser
* Payment will be done using simple methods (no complex APIs)
* System will be hosted on a free or low-cost hosting service
* Students must have a valid email address for registration

**5. Specific Requirements**

**5.1 Functional Requirements**

1. **User Registration and Authentication:**
   * Students can sign up using their email and password
   * Login/logout functionality
2. **Course Browsing and Filtering:**
   * Users can search for courses using keywords, categories, or instructors
   * Filter options include course duration and difficulty level
3. **Enrollment Module:**
   * Users can enroll in one or more courses
   * Enrolled courses are stored in the user's profile
   * Confirmation emails are sent upon enrollment
4. **Payment Processing:**
   * Simple manual confirmation: user uploads payment proof or enters reference number
   * Admin verifies and updates enrollment status
5. **Admin Interface:**
   * Add/update/delete courses
   * View all users and their enrollments
   * Manage course categories and enrollment status
6. **User Profile Management:**
   * View/update personal details
   * Track current and completed courses

**5.2 Non-functional Requirements**

* **Usability:** Intuitive UI with minimal learning curve
* **Performance:** Should handle up to 200 concurrent users
* **Reliability:** Basic error handling and data validation
* **Scalability:** Able to add new courses without affecting performance
* **Security:** Password-protected login, input sanitization

**5.3 External Interface Requirements**

* **User Interfaces:**
  + Web-based interface for students and administrators
  + Mobile-friendly design
* **Hardware Interfaces:**
  + No specialized hardware required
  + Standard PC or smartphone with internet access
* **Software Interfaces:**
  + Web browser (Chrome, Firefox, Safari)
  + Backend using PHP and MySQL or Google Sheets/Form

**5.4 Constraints**

* The platform must be deployed within 2 months
* The system must store data securely but using basic tools
* Budget limits the use of advanced paid services

**6. Appendices**

**6.1 Glossary**

* **Enrollment:** The act of registering for a course.
* **Dashboard:** A visual interface showing summarized user or admin data.
* **Reference Number:** A code entered by the user as proof of payment.

**6.2 Supporting Documents**

* User journey diagrams (to be attached separately)
* Database schema draft (to be designed in development phase)