مدينة زويل للعلوم والتكنولوجيا
Zewail City of Science and Technology

Software Engineering – Phase 4 Progress Report

Smart Course Registration System (SCRS)

December-2025

Instructor: Mohameds Sami Rakha

Prepared by:

- Mariam Ahmed Maher-202400733(s-mariam.maher@zewailcity.edu.eg)
- Farida Emad-202401764(s-farida.fouad@zewailcity.edu.eg)
- Elaliaa Kotb-202402473 (s-elaliaa.kotb@zewailcity.edu.eg)-team)
- Rahma Ali-202400365 (s-rahma.anwar@zewail.edu.eg)

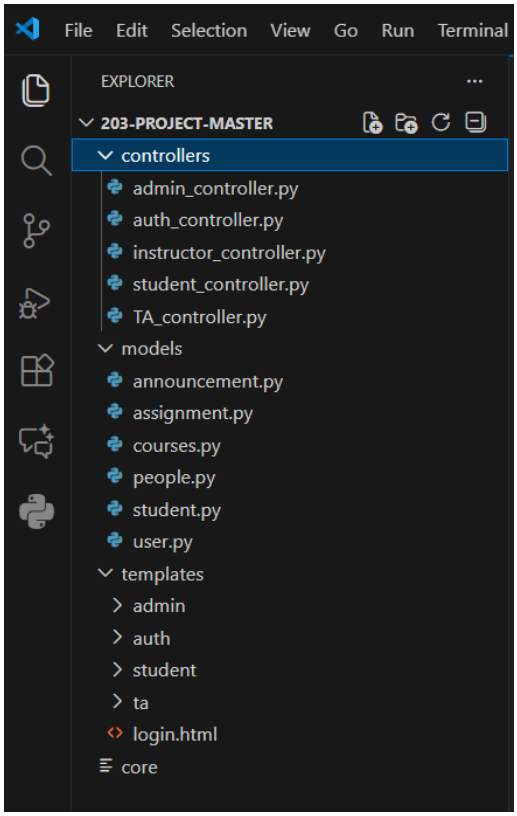
1. Introduction

This document presents the progress made during Phase 4 of the Smart Course Registration System (SCRS) project. Phase 4 focuses on implementing approximately 50% of the system's core functionality, adhering to the requirements outlined in the Software Requirements Specification (SRS). The system is built using Flask and adheres to the Model-View-Controller (MVC) architectural pattern. This report outlines the implemented functional requirements, project structure, team contributions, and collaborative workflow utilizing GitHub.

2. Functional Requirements Status

Function Requirement (FR)	SRS Section	Status	Notes
User Login	3.1	Done	Multi-role login implemented
Role-based Session Handling	3.1 & 2.3	Done	Redirects to the role-specific dashboard
Student Dashboard	3.5.1 & 2.2	Done	Dashboard UI and registered courses
View All Courses	3.1	Done	Shared page for all users
View Registered Courses	2.2 & 3	Done	Student-specific view
Course Registration	3.2	In Progress	Planned for next phase
Grades Management	3.4	In Progress	Planned for next phase
Payment System	3.6	Not Started	Planned for future phases

System Architecture (MVC)



The system is implemented using the Model–View–Controller (MVC) architecture to ensure separation of concerns and maintainability.

- Models define the system's data entities and business logic, such as users and courses.
- Views are implemented using HTML templates and are responsible for presenting data to the users.
- Controllers handle user requests, manage application logic, and coordinate between models and views.

3. GitHub Collaboration:

The team followed a structured GitHub workflow where each member worked on a separate branch. Development progress was tracked through regular commits with descriptive messages. Pull requests were created and reviewed before merging into the main branch, ensuring collaborative development and version control consistency.

4. Pending Features and Future Work:

Several functional requirements were intentionally deferred to future phases, including course registration, grades management, and the payment system. These features will be implemented incrementally based on the foundation developed in Phase 4

5. Conclusion:

Phase 4 successfully delivered the core functionality of the SCRS project, including multi-role authentication, role-based dashboards, and course-viewing features. The MVC architecture was effectively applied, and the team demonstrated strong collaboration through GitHub. The system is now prepared for further expansion in upcoming phases.