Zewail City of Science, Technology and Innovation
University of Science and Technology
School of Computational Sciences and Artificial Intelligence

CSAI 203 - Fall 2025

Introduction to Software Engineering

Unify

Phase 4: Core Functionality Prototype

Team Number: #27

Team Members:

Ahmed Moatasem 202300917
Jana Mahmoud 202301597
Mohamed Hatem 202301610
Karim Wael 202202212
Ali Mohab 202300786

Representative Contact info: s-ahmed.momtaz@zewailcity.edu.eg

1. Introduction

This report provides a detailed breakdown of specific functional requirements for the “Unify” software system, as described in its Software Requirements Specification (SRS) document. Each requirement is elaborated upon to provide a clear understanding of its intended func- tionality. The status of these features (completed or pending) is not determined by the SRS itself, which merely defines the requirements.

2. Detailed Functional Requirements (as defined in SRS)

- **AI Optmization Course Scheduling:** The system must incorporate Ma- chine Learning capabilities to intelligently create conflict-free and optimized course sched- ules based on user preferences, such as preferred time slots, days, instructors, or desired workload distribution.
- **NLP-Powered Note Summarization and Dashboard:** The system must leverage Natural Language Processing (NLP) pipelines to process and condense uploaded lecture notes or recordings, generating concise summaries. These summaries will be organized and accessible through a dedicated Notes Dashboard.
- **Google Calendar and LMS Integration:** The system must integrate with Google Calendar and Learning Management Systems (LMS) to achieve seamless synchronization of academic deadlines and personal events, providing a central view for all commitments.
- **Smart Reminders System:** The system must implement an intelligent reminder mech- anism for exams, assignments, and events, including countdown functionality and snooze options for user flexibility.
- **Task Management Functionality:** The system must allow users to add, edit, delete, and organize tasks, providing features for sorting by priority and deadline.
- **Secure Login and Signup:** The system must offer robust and secure login and signup functionalities, incorporating password hashing for security and providing optional two- factor authentication (2FA).
- **Unofficial Academic Transcript Generation:** The system must be capable of gen- erating unofficial academic transcripts, collecting and formatting relevant academic data, and allowing export in PDF format.
- **Secure In-App Messaging:** The system must include a secure in-app messaging feature to facilitate communication among students, Teaching Assistants (TAs), and instructors, supporting both individual and group chats.
- **Accessibility Options:** The system must provide various accessibility options, including dark/light modes, specialized color-blind themes, and dyslexia-friendly fonts, to cater to diverse user needs.
- **Course Catalog Browsing :** A feature that allows users to browse available courses with filtering options such as course title, instructor name, credit hours, and schedule times.

5. Pending Features - Detailed Explanation

- **Natural Language Query Assistant :** This AI-powered feature would allow users to ask questions in plain English (via voice or text) to retrieve their academic data such as schedules, grades, and course information. It requires NLP implementation for query understanding and data retrieval.
- **Pomodoro Focus Timer :** A productivity tool that helps students manage study sessions using the Pomodoro technique (typically 25-minute focused work sessions followed by short breaks). It should also track and record productivity statistics.
- **Role-Based Access Control - RBAC :** This security feature defines different permission levels for different user types (Students, TAs, Instructors, and Admins), ensuring each role can only access appropriate system functions.
- **Simple GPA Calculator:** User enters course name, credit hours, and grade. The system calculates the GPA instantly (front-end only).
- **Course Materials Management:** Instructors can upload course materials such as PDFs, PowerPoints, assignments, videos, and links. Students can view, download, and organize these materials by week or topic. The system should support file previews and categorize content by course.

Team Member	Student ID	Assigned Responsibilities
Ahmed Moatasem	202300917	Database Design and Implementation, Login/Registration Module, Course Registration Module
Jana Mahmoud	202301597	Database Design and Implementation, Task Management Module, Schedule Module
Karim Wael	202202212	Note Summarizer (NLP Integration), In-App Messaging Module, Report
Ali Mohab	202300786	Overview Dashboard, Settings Module, Accessibility Features, Report
Mohamed Hatem	202301610	Transcript Generation Module, Smart Reminders System, Report

Table 1: Team Member Responsibilities and Workload Distribution

The workload was distributed to ensure balanced contributions from all team members while leveraging individual strengths. Collaborative tasks, such as database design, were assigned to multiple members to ensure consistency and integration across modules. Each team mem- ber

was responsible for the complete development cycle of their assigned features, including design, implementation, testing, and documentation.

6. Conclusion

The development team has successfully completed 9 out of 18 functional requirements (64% completion rate). The remaining 5 pending features are critical for the full functionality of the Unify system and should be prioritized in the upcoming development sprints. These pending features include advanced AI capabilities (Natural Language Query Assistant), pro- ductivity tools (Pomodoro Focus Timer), security enhancements (RBAC), and core academic functionalities (Course Catalog Browsing and Automatic Course Enrollment).