

Middle East Technical University

Department of Statistics

STAT 295

Object Oriented Programming

Student Time Management Assistant

May 2025

Lecturer: Pelin Erkaya

Aykut Ünyazıcı

Furkan Koç

Oğuz Sezgin

Rabia Demircan

Timuçin Eke

Table of Contents

1. INTRODUCTION	3
2. Motivation.....	3
3. UML DIAGRAM.....	4
4. METHODOLOGY.....	4
4.1 User Class	4
4.2 Session Manager	4
4.3 Application Panels Architecture.....	4
4.3.1 Goal Panel.....	5
4.3.2 StudyToolPanel.....	5
4.3.3 GPAPanel.....	5
4.3.4 WeeklyOverviewPanel.....	5
4.3.5 AnalyticsPanel.....	5
5. Data Management	5
6. Development Tools & Libraries.....	6
7. FEATURES.....	6
7.1 Login Panel.....	6
7.2 Goal Panel.....	6
7.3 GPA Panel	7
7.4 Study Tool	8
7.5 Group project panel	9
7.6 Notification Panel	9
7.7 Analytics Panel.....	10
7.8 Weekly overview	11
8. CONCLUSION.....	11

1. INTRODUCTION

In today's academic environment, students often struggle to manage their time efficiently. From juggling multiple deadlines to keeping track of exams, homework, and study sessions, the need for an integrated time management solution has become increasingly essential. SmartTime is a Java-based desktop application designed specifically to address this need by offering a complete academic productivity toolkit.

The purpose of SmartTime is to provide students with a structured and visual approach to managing their study habits, academic goals, and performance metrics. The application enables users to log study sessions using multiple focus models such as Pomodoro, Deep Work, or Custom duration. In addition, it offers GPA and CGPA tracking tools, a goal management panel for projects and deadlines, and insightful weekly analytics based on real-time data logging.

This report shows the design, implementation, and evaluation of the SmartTime application. Through this project, we aim to demonstrate how an integrated software solution can significantly improve a student's academic time awareness and planning skills.

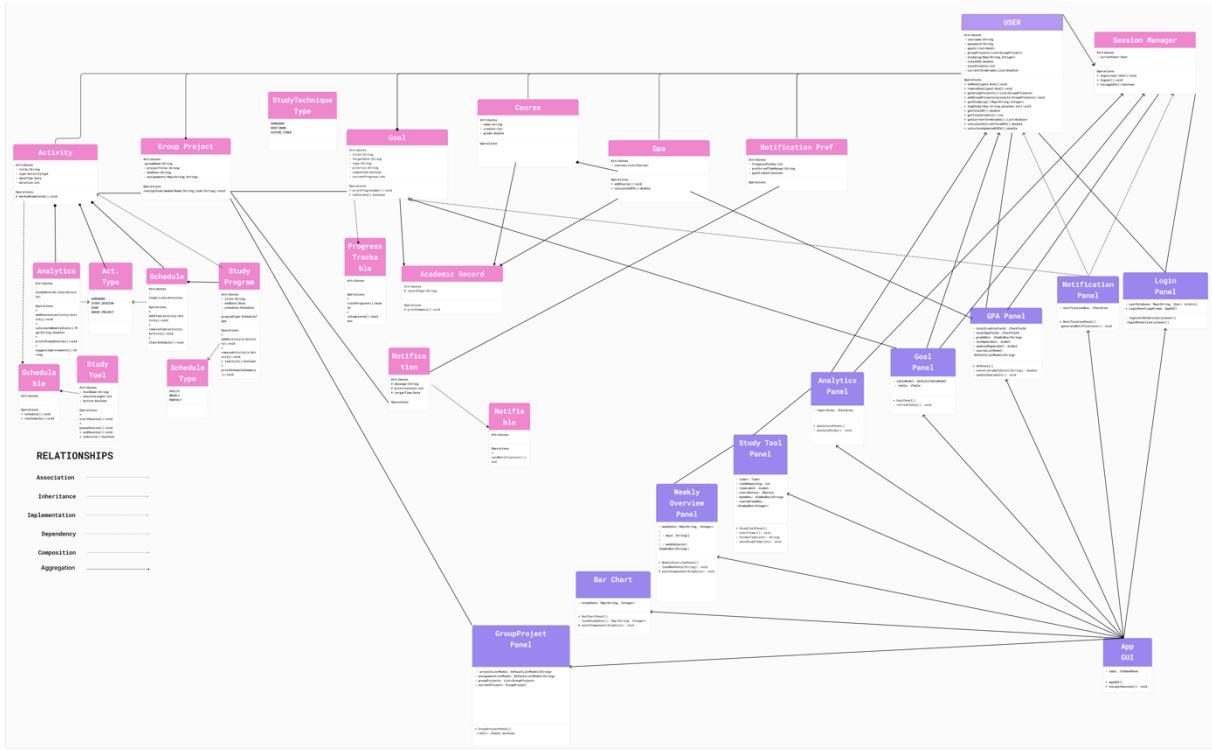
2. Motivation

The idea for SmartTime originated from a personal academic need: the challenge of managing multiple study goals, assignments, and exams in a structured and consistent manner. Like many students, I frequently found myself overwhelmed by approaching deadlines, unclear study patterns, and a lack of real-time feedback on how effectively I was using my time. While various time management applications exist, most are either too general-purpose for the academic workflow of a university student.

Traditional tools like calendars, simple timers, or mobile apps often fail to integrate GPA tracking, study session logging, and personalized performance analysis in a cohesive environment. This gap led to the conception of SmartTime—a desktop application that could unify these critical functions in a single, intuitive interface.

SmartTime reflects a synergy between academic productivity and technical creativity. It is a tool built by a student, for students—designed to make planning smarter, studying more intentional, and self-assessment more meaningful.

3. UML DIAGRAM



4. METHODOLOGY

The development of the SmartTime application was guided by object-oriented principles and modular software design. Each component of the application was implemented as a separate Java Swing panel, allowing for focused responsibilities, reusability, and maintainability. The methodology followed an iterative and feature-first development cycle, where the application was gradually expanded by adding core panels and functionality.

4.1 User Class

Acts as the central data holder for goals, GPA values, and study logs. Provides methods for logging study time, calculating term GPA and CGPA, and managing academic goals.

4.2 Session Manager

A singleton-like class that provides global access to the current user object. Ensures data consistency across multiple panels.

4.3 Application Panels Architecture

Each functionality of the application is encapsulated in a separate Swing panel:

4.3.1 Goal Panel

- Allows the user to create and manage academic goals (exams, homework, projects).
- Uses JTable and DefaultTableModel to display and update goal status.
- Tracks completion and deadline attributes.

4.3.2 StudyToolPanel

- Provides three session modes: Pomodoro (25 mins), Deep Work (60 mins), and Custom.
- Built with Timer objects to count down study time.
- Also writes to a local study_log.json file for persistence.

4.3.3 GPAPanel

- Allows students to input their previous CGPA and credits.
- Lets them enter new term grades (credit + letter grade), computes updated GPA.
- Utilizes grade-to-point conversion and a weighted average formula.
- Displays updated GPA and CGPA values live as data is entered.

4.3.4 WeeklyOverviewPanel

- Reads from local JSON files (study_log.json and study_log_week_last.json).
- Visualizes the number of study minutes per day using custom-drawn bar charts.
- Automatically highlights the current day using SessionManager.getToday().

4.3.5 AnalyticsPanel

- Analyzes the study duration of the current day.
- Gives motivational feedback such as:
 - Over 180 mins: "□ Excellent work!"
 - 90–180 mins: "□ Keep it up!"
 - 1–89 mins: "□ Try to stay focused longer!"
 - 0 mins: "□ Time to get to work!"

5. Data Management

- All study session data is saved in study_log.json using key-value pairs of Day: Minutes.
- Weekly summaries are stored in separate files for “This Week” and “Last Week” tracking.
- Data is read using Java’s java.nio.file.Files and parsed line-by-line for reliability.

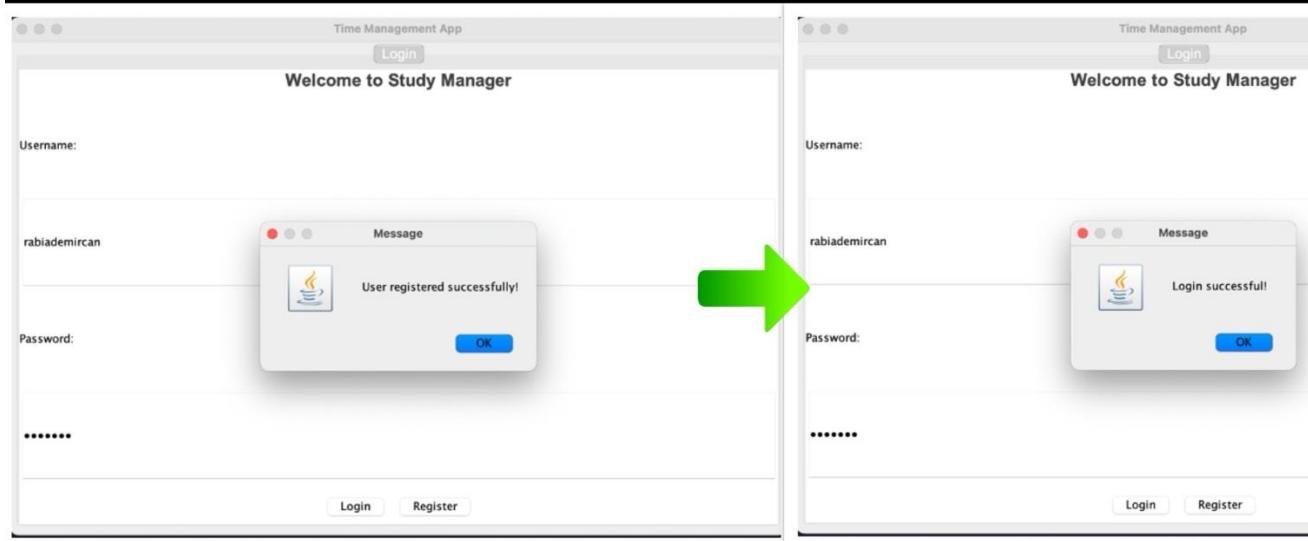
6. Development Tools & Libraries

- **Language:** Java (JDK 17+)
- **Framework:** Java Swing (no external UI framework used)
- **IDE:** IntelliJ IDEA CE
- **Libraries:** Only built-in Java APIs and optional org.jfree.chart for pie charts in Analytics

7. FEATURES

7.1 Login Panel

- Allows users to securely register with a username and password.
- Credentials are stored locally (for offline access).
- On successful login, users are directed to the main dashboard.



7.2 Goal Panel

- Users can add, update, or delete study goals such as:
 - Exam
 - Homework
 - Project
- Each goal has a title, due date, type, and priority (High/Medium/Low).
- Goals can be marked as completed via checkbox.

Time Management App

Goals GPA Study Tool Group Projects Notifications Analytics ▾

My Study Goals

Done	Title	Type	Due Date	Priority
<input checked="" type="checkbox"/>	STAT 295 Final...	Project	2025-05-22	High
<input type="checkbox"/>	STAT 295 Assi...	Homework	2025-05-27	Medium
<input type="checkbox"/>	STAT 295	Exam	2025-05-21	Low

Title:

Due Date (yyyy-MM-dd):

Type: Exam

Priority: Low

Add Goal **Delete Selected**

7.3 GPA Panel

- Students input:
 - Previous CGPA and total earned credits
 - Current term course grades (credit + letter grade)
- Calculates:
 - **Current Term GPA**
 - **Updated Cumulative GPA (CGPA)**

Time Management App

Goals GPA Study Tool Group Projects Notifications Analytics ▾

Previous CGPA	Current Term
Total Credits: 51	Credit: 1
CGPA: 3.11	Grade: AA

Save CGPA Info **Add Course**

Added Courses

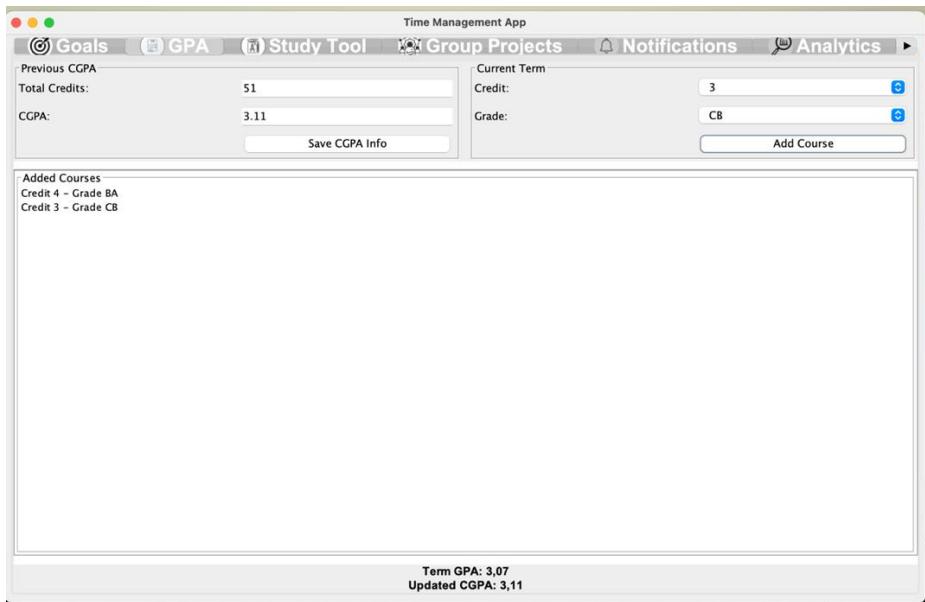
Message



Previous CGPA saved.

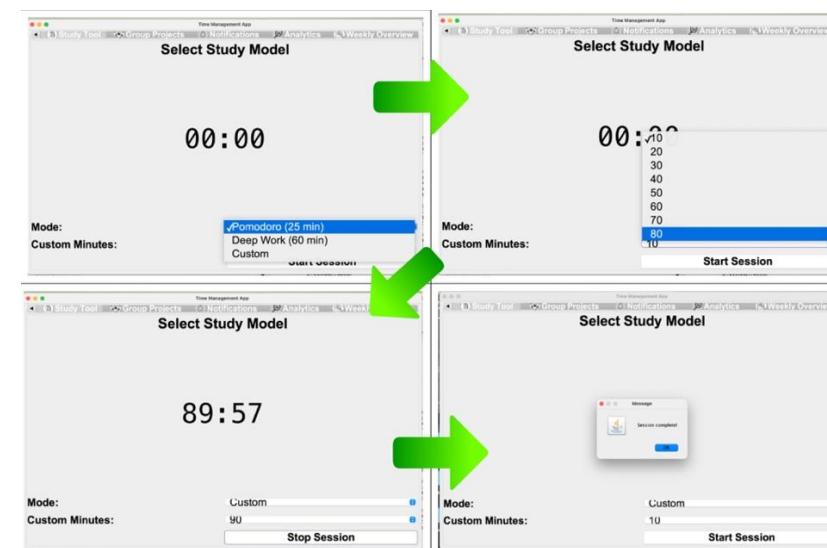
OK

Term GPA: 0.00
Updated CGPA: 0.00



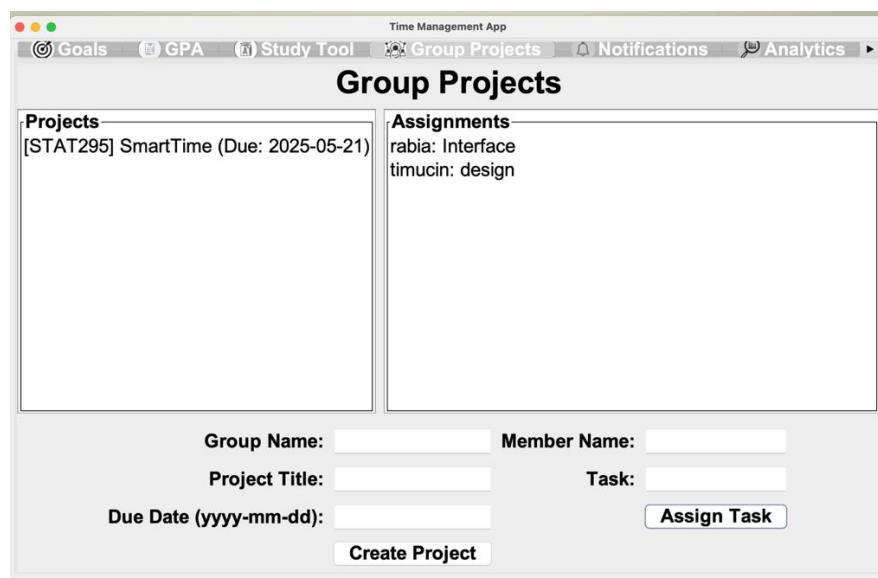
7.4 Study Tool

- Three study session modes options:
 - Pomodoro (25 mins)
 - Deep Work (60 mins)
 - Custom (User-selected: 10–90 mins)
- Countdown timer tracks each session
- When the session ends, the duration is:
 - Added to User.studyLog
 - Stored in study_log.json



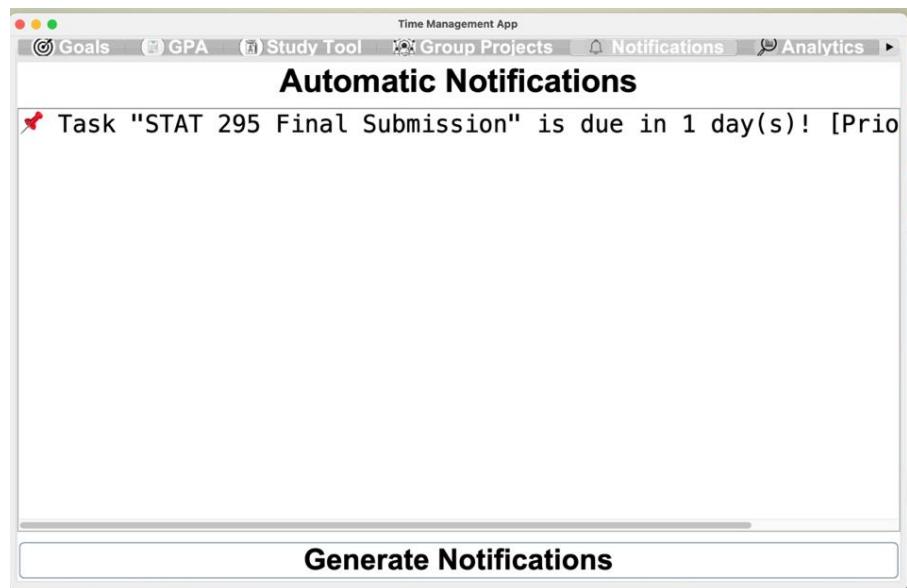
7.5 Group project panel

- Allows users to create and manage collaborative study projects.
- Each group project includes:
 - Project Title
 - Group Members (manually entered)
 - Deadline
- Users can track multiple projects in a table.



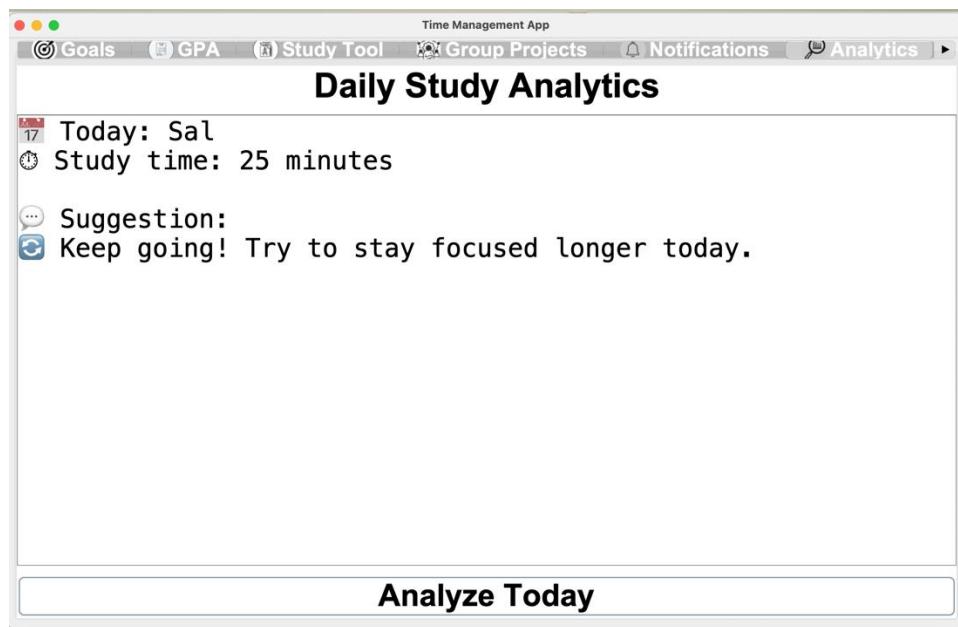
7.6 Notification Panel

- According to input which added in Goal Panel:
 - For high priority: Receive notification every day.
 - For medium priority: Receive notification 7, 3 and 1 day ago before the due date.
 - For low priority: Receive notification only 1 day ago before due date.



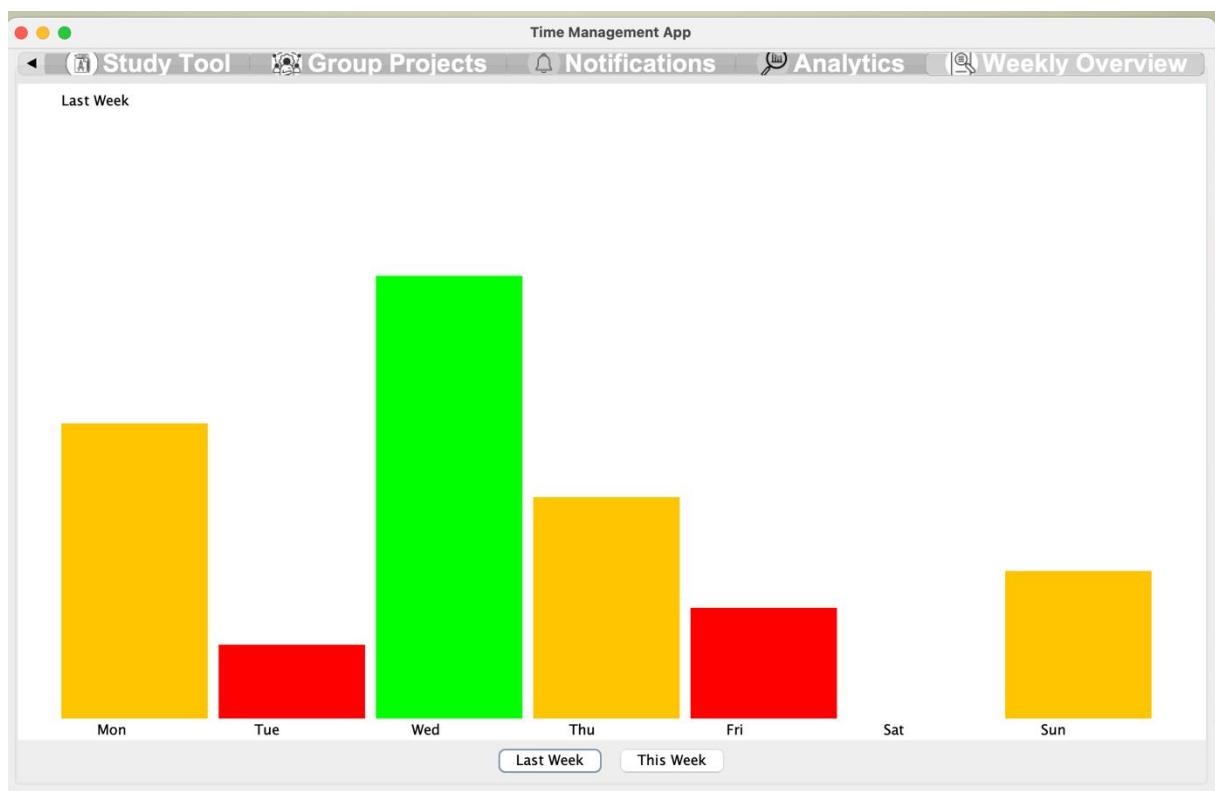
7.7 Analytics Panel

- Analyzes today's study duration from study log.
- Provides motivational suggestions:
 - 180+ min → “Excellent work today!”
 - 90–179 min → “Keep going!”
 - 1–89 min → “Try to stay focused longer!”
 - 0 min → “Time to get to work!”



7.8 Weekly overview

- Displays total study minutes for each weekday.
- Visualized as a bar graph using custom paintComponent.
- Shows data from:
 - study_log.json (current week)
 - study_log_week_last.json (past week)
- Users can switch between "This Week" and "Last Week" using dropdown.
- Automatically includes today's study if session was logged.



8. CONCLUSION

SmartTime was developed as a response to the growing need for structured and personalized academic time management among university students. Throughout the project, the focus remained on designing

a system that is both technically sound and practically useful for day-to-day student life. By integrating study session tracking, GPA analysis, weekly visualizations, and goal management into a single desktop application, SmartTime offers a holistic solution to boost productivity and academic awareness. The development process of SmartTime also provided valuable hands-on experience in modular Java Swing programming, event-driven user interface design, and file-based data persistence. The use of object-oriented design principles ensured that the application remained scalable, testable, and easy to maintain. Each component was built with clarity of purpose, allowing users to interact intuitively with the system. As a student-built tool for students, SmartTime achieves its goal: to help learners take control of their time, track their efforts, and understand their academic patterns more clearly. It demonstrates how thoughtful software design can directly support better habits and outcomes in real life.