CSCI-4250-002 Software Engineering Project Design Document
Project: Student Location Tracker
Repository: https://github.com/Jakehebert201/CSCI4250-ScrumProject
Version: 1.0
Date: Nov 24, 2025
Authors: Group 7 (Shane Austin, Ishwari Patel, Jacob Hebert, Deep Desai, Eloghosa Erhabor)

Introduction

This Detailed Design Document describes the internal structure, components, data flow, and functional behavior of the Student Location Tracker system. The goal is to allow developers, maintainers, and instructors to clearly understand how the application works at a technical level and how its subsystems interact.

The Student Location Tracker is a Flask-based web application that enables campus staff to:

- View real-time student locations
- Track attendance through clock-in/clock-out
- Send notifications (push, email, in-app)

Students can:

- Share location
- Enroll in classes
- Track attendance time
- View location history

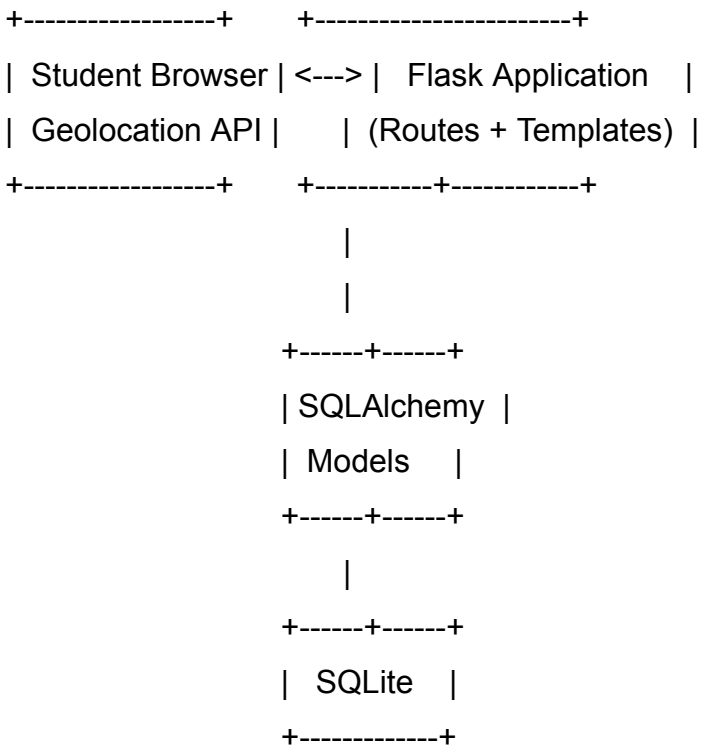
Architectural Style

The system follows a modular MVC-inspired Flask architecture.
Flask + Blueprints for modular routing

- **SQLAlchemy** for database models

- **Jinja2 templates** for HTML pages
- **Leaflet.js** for mapping and real-time location display
- **JavaScript fetch/AJAX** for sending location updates
- **SQLite** as the main development database

High-Level Architecture Diagram



System Components

Backend Components

- **Auth**: Login/registration for students and professors; optional Google OAuth
- **Student Dashboard**: Location sharing, clock in/out, history
- **Professor Dashboard**: Live map, view all latest student positions
- **Class Management**: Create classes, enroll students, view rosters
- **Notification System**: In-app messages, email/push options
- **Utility Functions**: Reverse geocoding, time tracking, template helpers

Frontend Components

- Map Interface (Leaflet)
- Clock in/out buttons
- Class enrollment UI
- Notifications inbox

Table	Purpose
User	Students & professors
Class	Course information

Enrollment	Links students ↔ classes
Location	Latitude, longitude, city, timestamp
ClockEvent	Stores clock in/out actions
DailyTotal	Calculated total time per student per day
Notification	Messages sent to users
NotificationPreferences	User settings

Data Flow Summary

Student Location Flow

1. Browser captures GPS.
2. Sends to /app/update_location.
3. Backend saves location + city name.
4. Professor dashboard fetches updated locations.

Attendance Flow

1. The student clicks the clock in/out.
2. Event stored in ClockEvent.
3. System updates DailyTotal.

Class Flow

1. Students enrol.
2. Checks capacity.
3. Enrollment stored.

6. Security

- Password hashing
- Sessions secured with FLASK_SECRET
- Role-based access control
- Optional OAuth login

Conclusion

This design allows a scalable, easy-to-use tracking and attendance system. The architecture is modular, the database is cleanly structured, and the app supports core features needed by both students and professors.