University of Science and Technology of Ha Noi

Department of Information and Communication Technology

Group Project Report

USTH Connect

Integrated app for university life assistant and student networking

by

Nguyen Thi Van	22BI13459
Chu Hoang Viet	22BI13462
Nguyen Hoai Anh	22BI13021
Nguyen Dang Nguyen	22BI13340
Do Minh Quang	22BI13379

Submission Date:	December 31, 2024
Supervisors:	Dr. Tran Giang Son

Contents

1 Introduction **6**

1.1 Context and Motivation 6

1.2 Process Flow 6

1.2.1 System Construction 6

1.2.1.1 Hardware Setup 6

1.2.1.2 Software Environment Configuration 7

1.2.1.3 Application Software 7

1.2.1.4 Initial Testing 8

1.2.2 Machine Learning Model Integration and Model Training 8

1.3 Project Objectives 8

1.4 Desired Outcomes 8

1.5 Structure of Thesis 8

1.6 Related works 9

2 Requirement Analysis **10**

2.1 System requirements 10

2.1.1 Functional Requirements 10

2.1.2 Non-functional Requirements 10

2.1.3 Desired Functionalities 10

2.2 Use Case 10

2.2.1 Use Cases Diagram 10

2.2.2 Use Case Characteristics 10

2.3 Use Case and Scenario Description 10

3 Methodologies **10**

3.1 System Architecture 10

3.2 Database Design 10

3.3 Use Case Implementation 10

4 AI Model Analysis and Training **10**

5 Results and Discussion **10**

5.1 Results 10

5.1.1 Mobile App Results 10

5.1.2 Machine Learning Results 10

5.2 Discussion 11

6 Conclusion & Future Work **11**

6.1 Conclusion 11

6.2 Future Work 11

List of Figures

List of Tables

1 Introduction

1.1 Context and Motivation

College life throws a lot of curveballs at students and those who run the show. At USTH, for example, juggling classes, making friends, and finding your way around campus can get pretty overwhelming. Traditional systems often feel clunky, with a lot of manual work, things scattered everywhere, and outdated information. This makes it tough for students to stay on top of things, find the resources they need, and connect with other students.

To overcome these challenges, we developed ULASNA – the University Life Assistant and Student Networking Application. ULASNA brings together some powerful tools like Spring Boot, Google Calendar, MapBox, and Moodle, along with a bit of machine learning magic. This platform makes academic life smoother by giving students easy access to their schedules, course materials, and all the important campus info. But ULASNA isn't just about academics – it helps students connect with each other too. The "StudyBuddy" feature, powered by machine learning, matches students with similar interests and goals, making it easier to find study partners and build friendships. We designed ULASNA with the user in mind, creating a user-friendly mobile app for Android devices. And we take data security seriously, using a robust PostgreSQL database and a secure VPN (Tailscale) to keep everything safe.

This report dives deep into the design and development of ULASNA, highlighting the challenges we faced along the way and how we overcame them. Ultimately, we believe ULASNA has the potential to significantly improve the college experience for both students and faculty

1.2 Process Flow

This section outlines the comprehensive process flow of University Life Assistant and Student Networking Application, detailing the progress from system construction to feature integration, including authentication with Spring Boot, Google Calendar integration, MapBox integration, Moodle resource fetching, real-time notifications, and the StudyBuddy matching system with machine learning algorithms.

1.2.1 System Construction

This section details the process of building the system. It covers the setup of hardware components, the configuration of software environment, the creation of the application's software components, and initial testing to make sure everything works smoothly.

1.2.1.1 Hardware Setup

The hardware components required for the system include:

- **Development Machines:** Computers used for developing and hosting the backend services and databases (Spring Boot and PostgreSQL).
- **User Devices:** Smartphones or Virtual Devices running the Android app to access features such as Google Calendar, MapBox maps, and Moodle resources.
- **Networking Equipment:** Tailscale VPN for secure and reliable communication between user devices and the backend server.

1.2.1.2 Software Environment Configuration

The hardware components required for the system include:

- **Operating System:** Window was used for hosting the backend services and database, while Android was the main platform for the app.
- **Database:** PostgreSQL was installed and configured as the relational database management system to store user information, calendar events, location data, and StudyBuddy profiles.
- **Backend Framework:** Spring Boot was deployed to manage REST API endpoints and handle authentication, authorization, and data synchronization.
- **Mobile Development Tools:** Android Studio served as the primary IDE for developing the Android application, integrating Java and libraries such as Retrofit and MapBox SDK.

1.2.1.3 Application Software

The application software consists of several key modules:

- **Authentication and Authorization Module:** Implements JWT-based authentication and Role-Based Access Control (RBAC) to manage access permissions for ADMIN and USER roles.
- **Google Calendar Integration Module:** Fetches event data from Google Calendar, detects changes, and delivers notifications to the user through the mobile application.
- **MapBox Integration Module:** Stores and serves latitude and longitude coordinates of campus locations to dynamically render maps within the application.
- **Moodle Resource Module:** Interacts with Moodle APIs to retrieve course-related resources such as slides, source code, and PDFs.
- **StudyBuddy Matching Module:**
  - Collects user profile data (e.g., interests, personality).
  - Utilizes a machine learning recommendation system to suggest suitable matches based on shared interests and compatibility.

  - Incorporates a chat feature for text-based communication between matched users.
  - Enables audio calls between users through the Linphone library, which provides SIP-based VoIP functionality for real-time communication.

- **Notification System:** Facilitates real-time notifications for calendar event updates, and received call.

1.2.1.4 Initial Testing

Initial testing was performed to verify the functionality and integration of all components:

- **Hardware Testing:** Verified the correct installation and operation of servers, development machines, and networking equipment, utilizing a Tailscale VPN for secure connections.
- **Software Testing:** Ensured proper configuration and performance of the operating system, PostgreSQL database, and Spring Boot services.
- **Integration Testing:** Validated the seamless interaction between backend APIs and mobile app features, including:
  - Google Calendar synchronization.
  - MapBox map rendering.
  - Moodle resource retrieval.
  - StudyBuddy matching functionality.
  - Linphone-based audio calling capabilities.

1.2.2 Machine Learning Model Integration and Model Training

1.3 Project Objectives

The primary objective of this project is to design, develop and implement an integrated system that enhances university life and fosters student networking through advanced technological solutions. The goal is to create a platform that seamlessly intergrates personal academic data, event annoucements, and campus resources. By using advanced frameworks and APIs, together with recommendation system, we aim to simplify the administrative tasks, improve campus accessibility, and encourage more interaction among students. This platform should be user-friendly and accessible through all devices, making it easier for students to experience campus life, stay informed, and connect with their peers.

1.4 Desired Outcomes

1.5 Structure of Thesis

The thesis will be structured as follows:

- **Part I: Introduction**

  Provide a general introduction to the thesis, including an overview of the project, its objectives, and the scope of the work.

- **Part II: Requirement Analysis**

  Lists all the tools, techniques, and system requirements used in the project. It includes both functional and non-functional requirements, as well as desired functionalities.

- **Part III: Methodologies**

  System architecture, database design, and implementation details of various features, illustrated with sequence diagrams.

- **Part IV: AI Model Analysis and Training**

  Analysis and training of AI models for recommend system for study buddy matchmaking, including datasets and model development, with (Model Name) integration.

- **Part V: Results and Discussions**

  Summarizes the implementations and achievements of the system. It reflects on how the objectives were met and provides a summary of the project's outcomes.

- **Part VI: Conclusion and Future Work**

1.6 Related works

In this part we will cite some related works/papers that we used mainly for this project. We also summarize the content of these resources.

2 Requirement Analysis

2.1 System requirements

2.1.1 Functional Requirements

2.1.2 Non-functional Requirements

2.1.3 Desired Functionalities

2.2 Use Case

2.2.1 Use Cases Diagram

2.2.2 Use Case Characteristics

2.3 Use Case and Scenario Description

3 Methodologies

3.1 System Architecture

3.2 Database Design

3.3 Use Case Implementation

4 AI Model Analysis and Training

5 Results and Discussion

5.1 Results

5.1.1 Mobile App Results

In this part we can have the demo for each feature of the app.

5.1.2 Machine Learning Results

In this part we will show the result of the clustering algorithm, using the evaluation metrics that we mentioned in the previous section.

5.2 Discussion

6 Conclusion & Future Work

6.1 Conclusion

6.2 Future Work