

Table of Contents

[**Introduction 1**](#_gjdgxs)

[**Description Model 1**](#_30j0zll)

[**Class Diagram 1**](#_1fob9te)

[**Use Case Diagram 1**](#_3znysh7)

[**Use Case Scenarios 1**](#_2et92p0)

[**System Sequence Charts 1**](#_tyjcwt)

# **Introduction DG**

The purpose of this document is to outline the requirements for using Campus Connect and outline the interactions the system has with the different types of users within the application. This document contains a text-based description model, class diagram, use case diagram and detailed descriptions, and a system sequence diagram.

# **Description Model: JR**

Campus Connect system is a mobile application designed to enhance the student experience by offering an all-in-one platform for managing academic schedules, extracurricular activities, study groups, textbook trading, and tracking graduation. Campus Connect will cater to college students, providing tools for better organization, time management, social interaction, and academic success.

Output Requirements:

**Student Dashboard**: A personalized interface displaying academic schedules, extracurricular events, and study group activities.

**Schedules**: A calendar feature showing class schedules, assignment deadlines, and event reminders.

**Study Group Chats**: Real-time messaging platforms where students can communicate about coursework and projects.

**Textbook Listings**: A marketplace for students to buy, sell, or trade textbooks.

**Graduation Tracker**: A tool that tracks academic progress towards completing your degree, including credits earned and upcoming required courses.

Input Requirements:

**Student Information**:Data profile including name, contact information, enrolled courses, and academic history.

**Course Information**: Course schedules, instructors, and course-specific details.

**Textbook Listings**: Information on textbooks for sale or trade, including condition, price, and availability.

**Study Groups Data**: Information related to study groups, including group members, meeting schedules, and topics.

Process Requirements:

**Schedule Management**: The ability to add, edit, and remove events from the student's calendar, as well as academic schedules.

**Study Group Formations**: Mechanisms for students to create, join, and manage study groups, including communication tools and group event coordination.

**Textbook Trading**: A feature where you can search for, and purchase textbooks from other students within the platform.

**Graduation Tracking**: An automated system to track the student’s academic progress, including completed courses, upcoming requirements, and an estimated graduation date.

Performance Requirements:

**Real-time Updates**: Any changes made within the system and must be reflected in real time across all devices.

**Responsiveness**: The application must have a response time of less than 3 seconds for any user interaction or page load as well as swift updates to data such as schedules and chat messages.

**Data Synchronization**: The system must ensure data synchronization between multiple devices and cloud storage. This includes real-time syncing of calendar events, study group data, and textbook listings.

**Error Handling**: The system must detect and display user-friendly error messages for common issues such as network problems, invalid inputs, or failed transactions.

Security Requirements:

**User Authentication**: All users must authenticate through secure login mechanisms, such as email/password or 2FA.

**Data Encryption**: All sensitive user data, including personal details, course information, and financial transactions, must be encrypted both in transit and at rest.

**Privacy and Data Protection**: The system must adhere to privacy regulations to protect user information. Users should have control over what data they share and with whom, and the platform should offer features to delete accounts and data upon request.

# **Class Diagram JR**

On Github

# **Use Case Diagram DG**

On Github

# **Use Case Scenarios HM**

On Github

# **System Sequence Charts HM**

On Github