# CSCI 370 Project proposal

# Student-Professor Appointment Booking System (SPABS)

# John Lamprecht

# Proposal

## What and Who:

The purpose of this project is to develop a Student-Professor Appointment Booking System (SPABS) designed for a university setting. This system aims to facilitate the scheduling of one-on-one meetings between students and professors, enhancing the academic experience by providing a convenient and efficient method for booking appointments. The target audience includes university students, professors, and academic administrators who are involved in the scheduling of academic advising, tutoring, or research meetings.

The student-Professor Appointment Booking System (SPABS) is designed for the academic domain within a university setting. It addresses the scheduling challenges faced by students, professors, and academic administrators in arranging one-on-one meetings for academic advising, tutoring, or research discussions.

The primary audience for SPABS includes university students seeking academic support, professors offering guidance and mentorship, and academic administrators responsible for facilitating academic interactions.

## The problem:

The current method of scheduling one-on-one meetings between students and professors, primarily through email or on a first-come, first-serve basis during scheduled office hours, often lacks flexibility and efficiency. This approach frequently results in students being unable to meet with professors and administrators due to conflicting classes and other time-oriented factors. Additionally, students may have to book appointments at times that are not convenient for them, leading to missed appointments and reduced access to academic support. Professors, conversely, face challenges in managing their schedules and ensuring they are available for all students who need to book appointments.

This project aims to address these issues by introducing a system that allows students to select multiple preferred times for their appointments, and similarly, allows professors to select times that are convenient for them. The system will then find matches between the proposed times and schedule appointments accordingly. This innovative approach not only improves the scheduling process but also enhances the overall academic experience by ensuring that students and professors can meet at mutually convenient times, thereby facilitating more effective academic advising, tutoring, and research meetings.

## Primary objectives, goals, and methods:

The primary objectives of this project are to:

1. Develop a user-friendly interface for students to book appointments by selecting multiple preferred time slots.
2. Implement a matching algorithm that efficiently matches students' preferred times with professors' available times.
3. Ensure secure and efficient communication between the client and server, with the server acting as a single application that communicates with an Oracle database to manage appointments.

The project will be implemented using C++ for both the client and server components, with the server and client acting as a single application that communicates with an Oracle database. The development process will involve designing the system architecture, implementing the frontend and backend logic, testing the system, and deploying it for use.

## Expected outcomes and implications:

The expected outcomes of this project include:

1. A functional Student-Professor Appointment Booking System that allows students to book one-on-one meetings with professors by selecting multiple preferred time slots.
2. Improved scheduling efficiency and reduced missed appointments due to inconvenient times.
3. Enhanced academic support and advising for students by ensuring they have access to professors' expertise.

The implications of this project are significant for both the university and its students. It will contribute to a more efficient and effective scheduling process, leading to better academic outcomes. Furthermore, it will serve as a practical demonstration of C++ application development and database interaction, providing valuable learning experiences for CSCI 370 students.

## 

# Implementation

## Database schema

### Entities:

* Student
* Professor
* Appointment
* TimeSlot

### Relationships:

* A student can book multiple Appointments.
* A professor can
  + have multiple Appointments.
  + book multiple Appointments (stretch goal)
* An appointment is associated with one TimeSlot.

### Schema:

* Student

StudentID (Primary Key)

FirstName

LastName

Email

Password

* Professor

ProfessorID (Primary Key)

FirstName

LastName

Email

* Appointment

AppointmentID (Primary Key)

StudentID (Foreign Key)

ProfessorID (Foreign Key)

TimeSlotID (Foreign Key)

AppointmentDate

Status (e.g., Pending, Confirmed, Cancelled)

* TimeSlot

TimeSlotID (Primary Key)

StartTime

EndTime

DayOfWeek

## Sample data

* Student

StudentID: 123456789, FirstName: John, LastName: Doe, Email: john.doe@viu.ca

StudentID: 987654321, FirstName: Jane, LastName: Smith, Email: jane.smith@viu.ca

* Professor

ProfessorID: 123789456, FirstName: Huizhu, LastName: Liu, Email: Huizhu.Liu@viu.ca

ProfessorID: 987321654, FirstName: David, LastName: Wessels, Email: davidwessels@viu.ca

* Appointment

AppointmentID: 1, StudentID: 123456789, ProfessorID: 987321654, TimeSlotID: 1, AppointmentDate: 2023-04-01, Status: Pending

AppointmentID: 2, StudentID: 987654321, ProfessorID: 123789456, TimeSlotID: 2, AppointmentDate: 2023-04-02, Status: Confirmed

* TimeSlot

TimeSlotID: 1, StartTime: 09:00, EndTime: 10:00, DayOfWeek: Monday

TimeSlotID: 2, StartTime: 14:00, EndTime: 15:00, DayOfWeek: Tuesday

## Interface design