Sec 1 Group 1 - Tech Stack/Reasons for Selecting Them

Backend:

- We decided to use the **SpringBoot** framework of Java for backend purposes.
    - It provides built-in tools for creating RESTful APIs quickly, which is ideal for developing the business logic that our application requires, such as managing tours, users, schedules, and feedback.
    - Since Spring Boot provides security with built-in tools such as **Spring Security** it enables us to implement strong authentication, authorization, and other security mechanisms efficiently. We have sensitive data such as High School prioritization, feedback mechanism and payment control. These functionalities should be protected with a strong and secure system which SpringBoot provides.
    - SpringBoot easily integrates with relational databases using Spring Data JPA. This feature allows us to do CRUD (Create, Read, Update, Delete) operations easily.
    - It is a widely used framework with a large community to help us resolve our problems.

Frontend:

- We decided to use **React** which is a library of **Javascript** in addition with **HTML/CSS** for frontend purposes.
    - React is designed to work seamlessly with RESTful APIs, which fits perfectly with the Spring Boot backend.
    - The React ecosystem offers a wide range of libraries and tools, such as **React Router** for handling navigation and **Redux** for state management, which are useful for building complex interfaces like role-based dashboards (e.g., different views for coordinators, counselors, and executives).
    - React uses a virtual DOM, which optimizes rendering and updates, ensuring the frontend is fast and responsive. This is particularly important for applications with real-time updates such as notifications about tour statuses.

Database:

- We decided to use a relational database management system called **MySql** for database operations.
    - MySQL integrates well with Spring Boot using **Spring Data JPA**. This allows us to work with the database using Java entities.
    - It ensures secure data storage.This ensures that sensitive information, such as user credentials and tour data, remains secure.

- MySQL is open-source, making it a cost-effective solution with enterprise-grade features. It has a large community and extensive documentation, providing support and guidance for best practices.