Objective:

The goal of this final project is to demonstrate the skills and knowledge learned in this course by developing a mobile application using React Native that integrates with native APIs and potentially external APIs. You should aim to use at least three of the covered topics, such as API, device sensors, external sensors, BLE, Geolocation, and Camera.

Instructions:

1. Choose a topic for your final project from the list of suggested topics or come up with your own idea with the instructor's approval. The project must involve at least 3 of the covered topics (API, device sensors, external sensors, BLE, Geolocation, and Camera).
2. Develop a mobile application using React Native that integrates with native APIs and potentially external APIs.
3. The final project should demonstrate the ability to access native APIs and to communicate with external APIs if necessary.
4. The final project should include a final report that documents the development process, including the problem being solved, the design and implementation of the solution, and any challenges encountered and how they were overcome. You also need to include your Figma link. A sample doc template is provided.
5. The final project should also include a demo and presentation of the application as a video submission.

Minimum UI Requirements:

The mobile application must have a user-friendly interface that includes:

1. A clear and intuitive layout, making it easy for users to navigate through the app.
2. Consistent and visually appealing design elements, such as colors, fonts, and icons.
3. Meaningful and informative feedback for user actions, such as error messages, success messages, or progress indicators.
4. Responsive design to adapt to different screen sizes and orientations.

Requirements:

The mobile application must have the following basic features:

1. A user-friendly interface that is easy to navigate (as outlined in the Minimum UI Requirements).
2. The ability to access native features (e.g., camera, accelerometer, etc.).
3. The ability to communicate with external APIs (e.g., OpenWeatherMap, etc.).
4. Proper error handling and user feedback.
5. Proper documentation and comments in the code.

Bonus (10%):
In addition to the basic requirements, the following features will earn bonus points:
1. Integration with multiple external APIs.
2. Use of animations and other visual effects to enhance the user experience.
3. Innovative features that go above and beyond the basic requirements. (you can brainstorm with the teaching team.)

Suggested Topics (these topics have an open-ended scope, you need to define your scope based on the requirement and submit it for the teaching team approval as a separate pre-final project assignment:
1. A fitness app that tracks the user's exercise and physical activity using device sensors (e.g., accelerometer, pedometer) and BLE for connecting to external sensors (e.g., heart rate monitor).
2. A personal finance app that allows users to track their expenses and income using device sensors (e.g., camera for scanning receipts) and external APIs (e.g., Plaid for financial data).
3. A transportation app that displays the current traffic conditions and provides real-time updates using GeoLocation and external APIs (e.g., Google Maps API, public transportation APIs).
4. A virtual treasure hunt app that uses GeoLocation and device sensors (e.g., accelerometer, compass) to guide users to hidden treasures in their vicinity while incorporating external APIs (e.g., Google Maps API) for mapping and location data.
5. A mood tracker app that allows users to log their emotions using device sensors (e.g., camera for facial recognition, microphone for voice analysis) and external APIs (e.g., emotion recognition, sentiment analysis) to provide insights and suggestions for improving their mood.
6. A smart home control app that uses BLE to communicate with and control various smart home devices (e.g., lights, thermostat, security system) and external APIs (e.g., weather data, energy usage) to provide a personalized, energy-efficient living experience.
7. A language learning app that incorporates device sensors (e.g., microphone for speech recognition, camera for object recognition) and external APIs (e.g., language translation, text-to-speech) to help users practice speaking and understanding foreign languages in real-world scenarios.
8. A multiplayer quiz app that uses device sensors (e.g., accelerometer for user input) and external APIs (e.g., trivia databases, real-time data feeds) to provide a fun and engaging quiz experience for users to test their knowledge on various topics.

9. A virtual pet care app that utilizes device sensors (e.g., accelerometer, camera) and external APIs (e.g., weather data, location-based services) to create an immersive and interactive experience where users can take care of a virtual pet by engaging in various activities and tasks.
10. A music discovery app that uses device sensors (e.g., microphone for audio recognition) and external APIs (e.g., Spotify API, Last.fm API) to identify and recommend new songs and artists based on the user's listening habits and preferences.
11. A gamified fitness app that incorporates device sensors (e.g., accelerometer, pedometer) and external APIs (e.g., fitness data, social media) to create a fun and interactive experience where users can compete with friends, complete challenges, and unlock achievements while improving their physical health.
12. A travel planning app that uses GeoLocation and external APIs (e.g., flight data, accommodation booking, local event data) to help users plan their perfect trip by offering personalized recommendations, itinerary planning, and real-time updates on travel conditions.
13. A stargazing app that utilizes device sensors (e.g., accelerometer, compass) and external APIs (e.g., astronomy data, weather data) to help users explore and learn about the night sky by identifying stars, constellations, and other celestial objects in real-time.

Submission:
Submit your final project report, demo video, and APP.tsx through Canvas by the deadline:
- Due Date: June 2nd at midnight for a 10% bonus
- 1st extension: June 3rd at midnight for a 5% bonus
- 2nd extension: June 4th at midnight for no bonus

Your submission should include the following:
1. A final report documenting the development process. Use the example final report template provided to you. 📄 Final Project Report Template
2. A demo of the mobile application. (a YouTube link) – be sure the quality is good

Grading:
The final project will be graded based on the following criteria:
- Functionality and usability of the app (60%)
- Quality and clarity of the code (10%)
- Final report (30%)