**Student Name: James Bareng | Cheng Zhang | Ivan Templora | Tien La Weight: 15%**

**Student ID: 499449 | 897055 | 899145 | 890883** **Marks:** **/25**

# Assignment: Architectural Decisions

## Introduction

It’s important that appropriate architecture is decided before the mobile app is built. Ignoring architecture in the first phase results in downstream issues with maintainability, scalability, and costs. You will need to consider things like the target audience, devices, and problem before building an app.

Plagiarism detection software will be used on all student submissions. Please refer to SAIT’s policies and procedures on Student Academic Conduct ([A.C.3.4.3](https://www.sait.ca/assets/documents/about-sait/policies-and-procedures/academic-student/ac-3-4-3-student-academic-conduct.pdf)) for more information.

## Instructions

1. This is a group assessment. Follow your instructor’s guidelines for forming a group. Groups may be assigned by the instructor as needed.
2. In this assignment, you will practice making necessary architectural decisions, through an Architectural Decision Record (ADR), and explaining why you made those decisions.
3. While some class time will be provided, you should also plan to invest time outside of class.
4. Review all instructions and the Marking Criteria section before beginning your work.
5. Your group will be assigned one of the scenarios outlined below.
6. Before you begin, review *Activity – Determining Architecture & Requirements*. Found in Brightspace under the Learning Activities in the Mobile Architecture Unit.
7. Create an Architectural Decision Record (ADR) to record your decisions. Be sure to include the rationale behind the decisions in your ADR.
   * Review [Architecture decision record (ADR)](https://github.com/joelparkerhenderson/architecture-decision-record) (https://github.com/joelparkerhenderson/architecture-decision-record)
8. Within your group, discuss the following architectural decisions (AD):
   * Native, web, or hybrid app **James Bareng**
   * UI Framework **Cheng Zhang**
   * Backend language **Tien La**
   * Permissions **Tien La**
   * Data storage **Ivan Templora**
   * Any additional frameworks or technology stacks **James Bareng**
9. Please see the course schedule and Brightspace for exact due dates.
10. Only one Brightspace submission per group is required. Your submission should include:
    * An Architectural Decision Record (.docx, .pdf, or .md)
    * A link to your group’s ADR in GitHub

**Scenario 1 – Group4’s selection scenario**

You are a team responsible for developing a new mobile app for a retail company. The app will allow customers to browse and purchase products, view their order history, and track the status of their deliveries. Additionally, the app will have a loyalty program feature, where customers can earn and redeem points for discounts on future purchases. The following requirements must be considered:

1. The retail company wants the app to support offline mode, allowing customers to browse products and view their order history even when they are not connected to the internet. The app should sync data with the server once an internet connection is available.
2. The retail company wants to send push notifications to customers to notify them about order updates, new product arrivals, and exclusive offers. The app should integrate with a push notification service to handle the delivery of notifications.
3. The app needs to integrate with various payment gateways to facilitate secure and convenient transactions for customers. The team should select and integrate a suitable payment gateway or a combination of gateways based on security, ease of use, and compatibility with the app's target platforms.
4. The retail company wants to track user behavior, such as product views, purchases, and loyalty program interactions. The team should select an analytics tool or service that provides detailed insights and metrics to help improve the app's performance and user experience.
5. The app will display product images, which may vary in size and resolution. To ensure optimal performance, the team needs to decide on an image storage solution and implement image optimization techniques such as caching, lazy loading, and compression.
6. The retail company plans to expand its customer base internationally. The app should support multiple languages and adapt to various cultural preferences. The team needs to implement localization techniques and decide on the appropriate localization framework or libraries.

**Scenario 2**

You are a team responsible for developing a social networking mobile app for a university. The app will allow students to connect with each other, share information about classes, view events and clubs, and manage their schedules. The app will also have a feature for professors to post announcements, assignments, and grades. The following requirements must be considered:

1. To ensure widespread adoption and accessibility, the app should support both iOS and Android platforms. The team needs to decide on a framework or technology stack that allows for efficient development and maintenance across multiple platforms.
2. As students and professors may not always have access to a stable internet connection, the app should support offline mode to enable users to access certain features and data even when offline. The team should implement offline capabilities and data synchronization mechanisms to ensure a seamless user experience.
3. Considering that students and professors may or may not have a top of the line smartphone, it is important to optimize the app's performance. The team should employ techniques to enhance app compatibility and minimize data usage.
4. To secure user data and ensure appropriate access control, the team needs to integrate the app with an existing Active Directory system. Students and professors should be able to securely log in, and their roles and permissions should be properly enforced.
5. To keep users informed about new announcements, assignment deadlines, and other relevant updates, the app should integrate with a push notification service. The team needs to select a suitable push notification provider that supports both iOS and Android platforms.
6. As the app will handle sensitive information, such as student grades and personal profiles, the team needs to prioritize data privacy and security. The team should implement encryption, secure data transmission protocols, and adhere to relevant data protection regulations.
7. To ensure inclusivity, the app should be designed and developed with accessibility features in mind. The team needs to consider accessibility guidelines and implement features such as text-to-speech, high contrast modes, and support for assistive technologies.

**Scenario 3**

You are a mobile app development team tasked with creating a new app for a client. The app will allow users to order food from local restaurants for delivery or pickup. Users must be able to create an account and save their payment information for future orders. The app must have a user-friendly interface that is easy to navigate and allows for seamless ordering. The following requirements must be considered:

1. The app needs to track the user's location to provide accurate restaurant recommendations, estimate delivery times, and display nearby options. The team should decide on an approach for location tracking, such as utilizing GPS capabilities or integrating with location-based services.
2. To enhance the user experience, the app should provide real-time order tracking, allowing users to monitor the status and progress of their orders. The team needs to determine the architecture and technology stack for implementing real-time updates.
3. The app requires integration with various payment gateways to facilitate secure and seamless transactions. The team should evaluate different payment gateway options, considering factors such as security, user experience, supported regions, and transaction fees.
4. To ensure accurate and up-to-date restaurant menus, the app should integrate with the restaurants' inventory management systems. The team needs to determine the most efficient and reliable method of synchronizing menu data, such as through APIs or web scraping techniques.
5. To help users make informed decisions, the app should include a feature for users to leave reviews and ratings for restaurants and food items. The team needs to design and implement a system for collecting, displaying, and moderating user-generated content.
6. The app should provide users with access to their order history, allowing them to easily reorder their favorite items. The team needs to decide on the data storage and retrieval mechanisms to efficiently manage and display order history.
7. To keep users informed about order confirmations, delivery updates, and promotional offers, the app should include a notification system. The team needs to determine the architecture and integration approach for sending and managing push notifications.

**Scenario 4**

You are part of a team that has been tasked with creating a mobile app for a modern transportation company. The app will allow users to book and track rides, view driver details and ride history, and make payments. The app must provide authentication and keep track of both the driver and passenger's location. The following requirements must be considered:

1. To enable ride booking and tracking, the app needs to incorporate geolocation services. The team should decide on the most suitable geolocation service provider, such as Google Maps or Mapbox, based on factors like accuracy, coverage, and cost.
2. To track the driver and passenger locations in real-time, the app requires continuous updates. The team should consider using technologies like WebSocket or server-sent events to establish a real-time connection between the app and the server for seamless location updates.
3. The app should provide a map interface to display ride routes, estimated arrival times, and driver locations. The team should choose a mapping service and routing algorithm that meets the transportation company's requirements, considering factors such as accuracy, traffic data, and customization options.
4. To facilitate secure and convenient payment transactions, the app needs to integrate with one or more payment gateways. The team should evaluate different payment gateway options based on security, ease of use, supported regions, and transaction fees.
5. The app should implement a secure authentication and authorization system for both drivers and passengers. The team needs to decide on the authentication mechanisms, such as email/password, social login, or two-factor authentication, and ensure proper authorization rules are enforced.
6. To store user profiles, ride history, and payment information, the team needs to design an appropriate database schema. Considerations should include data integrity, scalability, and efficient querying to provide a seamless user experience.
7. To notify users about ride confirmations, driver updates, and important announcements, the app should incorporate a push notification system. The team should choose a push notification service provider that supports both iOS and Android platforms and offers features like personalization and message targeting.
8. As the app will handle sensitive user information and transactions, implementing robust security measures is crucial. The team needs to ensure secure communication over networks, implement encryption techniques, and adhere to industry standards and best practices.

## Marking Criteria

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Criteria** | **Needs Improvement** | **Good** | **Excellent** | **Marks** |
| **Architectural Decisions Record** | ADR is incomplete or missing elements. Business requirements and user needs are not considered.  (2 marks) | ADR includes all required elements and most details. Some business requirements and user needs are considered.  (6 marks) | ADR includes all required elements and additional details. Business requirements and user needs are considered and met.  (10 marks) | **/10** |
| **Rationale** | Rationale for design choices is unreasonable, not explained, or missing.  (2 marks) | Rationale for design choices is provided, but unclear or incomplete.  (6 marks) | Rationale foe design choices is logical, fully explained and supported.  (10 marks) | **/10** |
| **Formatting, Spelling, Grammar** | ADR is formatted poorly with multiple spelling and grammar errors.  (1 mark) | ADR is formatted well with some spelling and grammar errors.  (3 marks) | ADR is well-formatted and easy to read. No spelling or grammar errors.  (5 marks) | **/5** |
| Total | | | | **/25** |