Architectural Decision Record (ADR) for Mobile Food Ordering App Development:

Submitted by: Dave Luna, Jairo De Guzman, Justin James Marquez for CPRG-303-C

Issue:

We are creating a plan for building a mobile app for ordering food while ensuring that everything meets the requirements set by the client. We are paying close attention in ensuring the app is easy to use, can accommodate to multiple users and performs well.

Decision:

In developing this mobile app for food ordering, our group decided to prioritize user experience, security, and efficiency. Ensuring precise delivery estimations and suggestions is ensured by employing GPS capabilities for location tracking. By giving fast information, WebSocket technology's real-time order tracking improves user engagement. Using *Stripe* as the payment gateway integration guarantees safe and easy transactions. Menu information is kept current through API interaction with restaurant inventory management systems. User-generated information is moderated within the app to create a reliable environment for reviews and ratings. Cloud-based storage allows for scalability and easy access to order history data. Effective push notifications are made possible by integrating with Firebase Cloud Messaging, informing users about promotions and order updates. The overall goal of this design is to prioritize security and convenience while providing a dependable and user-friendly meal ordering experience. See *Positions* for more detailed information.

Status: *Pending*

Group:

- Integration: Payment gateways, Restaurant menu synchronization
- Presentation: UI, Real-time order tracking, Review and rating system
- Data: User account and order history management
- Event: Location tracking, Notification System

Assumptions:

The team developing the food ordering app are filled with talented individuals who excel at creating mobile applications. They have a deep understanding of the technologies and platforms selected by the client. The client prioritizes both security and user experience, emphasizing the importance of making the app reliable and easy to use. Given the allocated time and budget, the development team is committed to delivering a top-notch app that meets the client’s expectations.

Constraints:

These are the list of constraints that the mobile applications have:

- **Development** - Building features like real-time tracking and push notifications requires a lot of development efforts to pull off making the application more complex.
- **Compatibility** – One of the constraints that the app will have if the mobile is compatible with the external services being implemented into the app like payment gateways and restaurant inventory management system.
- **Performance** – With the use of real-time updates, restaurant's menu and keeping order history of users can impact the app's performance especially when lots of users are using the app simultaneously.
- **Security**: Implementation of features like real-time-order and payment integrations, the app needs to be locked down tightly so that private user information does not get leaked.
- **Data Privacy** – The app will have constraints on how much data it can collect and store without violating any laws.

Positions:
Upon review, these are the viable options and alternatives that our group considered:

- **Location Tracking Approach** – this involves the access of the user device's GPS and Location Based Services (LBS) to obtain accurate location data. This approach offers high precision but can use up too much RAM and may cause the device to heat up or drain the battery faster.

- **Real-Time Order Tracking Architecture** – this enables real-time updates between the application and the server. With the use of technology such as WebSocket, communication between the two parties can be possible with low latency which is ideal for tracking.

- **Payment Gateway Integration** – the process of connecting the application with a payment gateway service to secure online transactions such as *Stripe, PayPal, Square,* etc. Payment gateways act as a bridge between the application and financial institutions such as banks and credit card companies involved when processing payments.

- **Restaurant Menu Synchronization Method** – with the use of *API Integration* which allows third-party applications to access a specific restaurant's menu, the application can maintain accurate and updated menu. This can ensure that the users can have access to the latest menu offerings, prices, availability and even promos being offered by a restaurant when ordering or browsing food online.

- **User Review and Rating System** – this could enhance user experience and provide transparency for the users. This could enable the users to provide valuable feedback whether is positive or negative for the restaurant's improvement. As this system can be used by anonymous people, in-app moderation should be implemented where the user-generated comments or reviews are moderated within the app or automated filters are programmed before being displayed publicly.

- **Order History Data Storage and Retrieval** – this can be applied through the device's local storage or online through cloud storage to provide convenience when re-ordering their favorite items that they previously enjoyed without having the hassle to search for it again and again. This can help users to save time and effort especially for the users who uses the app frequently.
- **Push Notification Integration** – this could provide an enhanced user experience as the user will be notified automatically when there is an available update for new promotions and products. Users will also be notified real-time about the status of their order without having to open the app. This could also improve communication with the users by sending push notifications as way of communicating. Examples of such are Firebase Cloud Messaging (FCM) and Apple Push Notification Service (APNs).

Argument:

We selected these positions based on their alignment with the client's requirements, industry best practices, and the team's expertise. With each having their own pros and cons, our team weighed each one of them to make sure that the user can have more benefits rather than disadvantages while using the app. For example, despite the potential battery drain and resource usage, utilizing GPS for location tracking ensures accurate recommendations and delivery estimates. While WebSocket may require additional development resources and expertise which may cause higher implementation costs, the benefits are worth it as it can provide technology that enables real-time order tracking, enhancing user experience. Integration with reputable payment gateway such as Stripe can cause high integration for the availability of development, but this ensures secure transactions and lower transaction fees which can provide both security and convenience. Utilizing APIs for data synchronization and a relational database for order history storage ensure efficiency and scalability and implementing a push notification service keeps users informed and engaged at the same time this can act as a "free advertisement" for the restaurants as users will be able to see offers, deals, or promotions with just one swipe of their finger.

Implications:

List of Implications that the application will have:

- **Integration** – App will have implications with the third-party integrations because of the dependencies that the third-party integrations will have.
- **Security** - One of the implications that the app will have with security is with complying with privacy laws.
- **User expectations** – With the number of users that will be using the app the performance of the app will degrade over time. The app must need to be updated and maintain regularly to ensure that the app is meeting user expectations.
- **Environment Constraints –** The implications of using different payment integration system are that the application has rules and limits it must follow. If the dependencies changes, the application will also need to be updated
- **Staff Training –** Bringing in new ways of doing things or new tech mean that we need to give the team extra training so that they are updated on knowledge that they need to know to ensure that

the app runs successfully without problems.

Related Decisions:

- **UI/UX Design for Enhanced User Engagement:** The right user interface and user experience design can make location tracking, ordering updates, and payment transactions easy for the users. More than just visual design and layout, effective UX/UI design ensures that users are engaged and satisfied with your mobile app.
- **Security and Data Privacy Strategy:** User trust and safety are paramount, especially when it comes to their location and payment information. The decision will involve how encryption will be used, how to integrate with APIs securely, and how to be compliant with all applicable data protection regulation.
- **Scalability and Infrastructure:** Decide on the tech stack for the app's infrastructure that will ensure the real-time features, in-app payment processing, and user data remain easy to manage and don't get clogged as the app sees more users and more user data.
- **Content Moderation and Review Management:** We need to make key decisions about how to filter and moderate user-generated content like reviews and ratings to ensure their quality and relevance. This will include automated filters, user reporting mechanisms, and moderation team workflow.
- **Marketing and User Acquisition Strategy:** While not a technical architecture decision, choosing approaches for marketing and user acquisition affects the overall success and adoption of the app. This includes leveraging location-based services for targeted promotions and integrating social sharing features to increase app visibility.

Related Requirements:

- **Accurate Restaurant Recommendations and Delivery Estimates:** Requires integration with GPS and location-based services, as well as real-time data processing for delivery times. The architecture must support efficient data retrieval and processing to meet this requirement.
- **Secure and Seamless Payment Transactions:** Necessitates decisions on integrating multiple payment gateways and implementing robust security measures. The contribution of these decisions to the requirement is evaluated based on security, user experience, and support for multiple payment methods.
- **Real-time Order Tracking:** Involves custom development for real-time updates, requiring scalable real-time messaging. The effectiveness of this decision is assessed by the timeliness and accuracy of order status updates provided to users.
- **Up-to-date Restaurant Menus:** Depends on reliable synchronization with restaurant inventory systems, evaluated by frequency, reliability, and accuracy of menu updates.
- **User Engagement Through Reviews:** Develop a review and rating system for restaurants.
- **Easy Access to Order History:** Involves choosing how best to store and retrieve data, with performance, dependability, and usability of previous order data access serving as important indicators of success.

Related artifacts:

List the related architecture, design, or scope documents that this decision impacts.

- **Security Guidelines** – Responsible for the security section of the application. I.e. Payment transaction, data storage and user authentication.
- **Architecture diagram** – Responsible for the main architectural design of the app.
- **Integration Specification** – Responsible for the integration with external services such as payment and restaurant inventory management system.
- **Performance Requirements** – Responsible for the overall performance of the application
- **Database design** – Responsible for data storage as well as retrieval of data.
- **User Interface Design**: Responsible for the overall design of the application,
- **Deployment Strategy -** Responsible of the deployment of the app as well as any rollback for when the app runs into a problem
- **Testing Plan - Responsible** for testing of different scenarios and different test cases to make sure that the app meets user expectations and client's requirements

Related principles:

Each decision was carefully considered to prioritize efficiency, security, user-centricity, scalability, and innovation - ensuring consistency and alignment with the client's objectives. All these decisions contribute to a strong and consistent approach towards a certain goal which is user satisfaction.

Notes:

Upon discussion, concerns were raised about the scalability of the in-house real-time order tracking system and the potential need for a robust content moderation strategy for user-generated reviews. The team agreed to monitor these areas closely and adjust plans as found necessary.

References

Art Akerman, Jeff Tyree, 2023. *Decision record template by Jeff Tyree and Art Akerman*. https://github.com/joelparkerhenderson/architecture-decision-record/tree/main/locales/en/templates/decision-record-template-by-jeff-tyree-and-art-akerman

Evgeniy Altynpara, Diana Bestaieva, June 2, 2023. *How to Create a Food Delivery App: Features and Development Cost.* https://www.cleveroad.com/blog/create-food-delivery-app/