Appendix D

Tunku Abdul Rahman University of Management and Technology
BAIT2073 Mobile Application Development

Task Description

Name	:	Ong Yi Xin	ID	:	24WMR09097
Programme	:	RSD	Group	:	G5

Instruction: Answer **ALL** the questions. You may insert extra pages.

1. Please briefly describe the module(s)/function(s) you engaged in the assignment.

   I mainly engage in the appointment module, feedback module and consider a small module which is the customer car management module in the assignment.

   Appointment Module

   I primarily developed the appointment module, creating an appointment list displayed in a weekly calendar view to enable users to filter appointments by date.For each appointment shown on the selected date, customers can click the info button to view details such as the appointment start time, estimated end time, workshop name and included services.Beyond the workshop name, there are two buttons provided to navigate directly to the workshop's details page or open Google Maps for directions. Customers can also cancel appointments that have not yet begun.

   Furthermore, I implemented the make appointment functionality and handled several business rules which are each appointment is tied to a single workshop but may include multiple services from that workshop and each car can only book one appointment per day. Appointment start and end times are calculated based on the workshop's staff availability, maximally allowing three concurrent services for one appointment. The system then assigns services to corresponding time slots of the staff that are available. Customers just need to select the workshop, services, car involved and also the appointment date for booking an appointment.

   Feedback Module

   I also manage the feedback module, which handles customer's pending and completed reviews.customers can only provide feedback after their appointment was completed and payment was paid. Feedback includes a star rating, with the option to add written comments and upload up to three images for further details. Feedback includes a star rating, with the option to add written comments and upload up to three images for further details. I also added filtering and sorting capabilities which enable customers to manage their feedback list more efficiently such as filtering by status (pending or reviewed) and sorting by date in ascending or descending order.

   Service Routine Function

   Furthermore, I developed a routine service schedule to help customers track required services based on their registered vehicle's odometer reading and maintenance intervals. A dedicated page displays both newly added maintenance requests and routine maintenance needs, featuring a tabbed interface for easy switching between categories. This functionality ensures timely reminders for critical maintenance tasks.

   Car Management Module

   Finally, I developed the Car Management module, enabling customers to register, edit and deactivate their cars. When registering a car, customers must provide details such as license plate number, pickup date, color, odometer reading, brand and model. After registration, customers can edit information like license plate number,

odometer reading or color to make corrections or updates. Additionally, customers can choose to deactivate a car when it no longer belongs to them.

2. What are the strengths of the modules/functions created by you?

   The modules I developed offer several key advantages that enhance system reliability and customer experience. First, the appointment module automatically generates booking slots for the next three months (including the current month). This not only reduces database load but also ensures that even when new employees are added mid-month, appointment slots remain accurate and up-to-date. This prevents customers from encountering situations where the workshop has available staff but consistently displays “booking slots are full” or “no slots available.”

   Additionally, the appointment list supports flexible filtering. Customers can optimize results not only by appointment status but also by payment status and car involved, granting customers greater control and visibility. To further improve usability, I have implemented a navigation service that remembers the date selected when customers view the appointment list. This means the list retains the previously chosen date even if users leave and return to the page.

   Another advantage is the automatic status update feature, which will adjust appointment statuses immediately upon reaching start or end times and sends app notifications to customers. This ensures the appointment schedule always reflects reality and can be improved later to incorporate employee-triggered updates.

   Finally, appointment slots are generated based on the workshop's local time zone but stored in Supabase using UTC format. When displayed to users, times are converted to their local time zone, ensuring accuracy and consistency across regions. This design not only prevents scheduling confusion but also enables the app to seamlessly support workshops and customers in different geographic locations, making the system more adaptable for broader market expansion.

3. What are the weaknesses of the modules/functions created by you?

   A major weakness of the module I developed is that customers can only select dates when making appointments, lacking the flexibility to choose specific time slots. This limitation may lead to a lack of user experience as clients must adhere to automatically assigned times by the system. On the other hand, this approach offers advantages for the workshop by helping reduce excessive gaps between appointments thus minimizing resource wastage.

   Another weakness is the application's lack of local caching. Consequently, each time a customer navigates to a new page or makes some filter and sorting action, the system retrieves data directly from the Supabase, resulting in slower loading times and reduced responsiveness.

   Additionally, the appointment process of the appointment module lacks detailed logging, this limiting customers from viewing precise timestamps for each status change such as when an appointment started or was marked as completed. This lack of transparency may undermine user trust and limit their ability to effectively track appointment histories.

4. What have you learned in doing this assignment?

Appendix D

I learned and gained a deeper understanding of the development process and logic behind building mobile applications through this assignment, particularly how to integrate Supabase and FastAPI as the backend to connect with Flutter. I also became more aware of the differences between developing web applications and mobile applications, especially in terms of design considerations and user interactions. While most backend logic remains consistent, frontend approaches have become more diverse and flexible.

Additionally, I also learned the importance of effective state management in Flutter, by using providers like NavProvider to ensure smooth navigation and retain user actions such as remembering the selected appointment date. Finally, I practiced code optimization by reducing redundant queries such as generating appointments times in advance which not only avoid database overload but also improves scalability and overall system efficiency.

5. What are the challenges, if any, faced by you while working on this assignment?

   One of the biggest challenges I faced in this project was developing the appointment scheduling functionality. This proved particularly difficult because the process required integrating data from nearly every module of the application. From workshops and their services, to customers' registered cars, existing appointments, and available time slots at selected workshops, all these elements had to be interconnected to correctly allocate time slots for each service chosen by the customer.

   Beyond this, appointment creation validation added another layer of complexity because there were multiple business rules that had to be enforced. This involved nested checks across different fields, making the logic both intricate and demanding. The real challenge lay not only in ensuring the accuracy of the validations but also in optimizing the code to remain concise, efficient, and maintainable despite the complexity.

   Another major challenge was generating appointment time slots in Supabase while supporting different time zones and ensuring data consistency. It took considerable time researching backend logic workflows before I found a viable solution.

Signature: Date: