Fuel Delivery Mobile App

Project Plan

Fahad Riaz (101027495)

October 11, 2024

Supervised By: Vasudha Darbari

Department of Computer Science,

Royal Holloway, University of London

B.Sc. CS Final Year Project

Abstract

There is an increasing demand for fuel because of many cars being introduced in the auto industry and this has led to a need for building fuel delivery applications that require the users to place an order for their vehicle when their fuel runs out or when it is low without the hassle of going to the petrol stations [1]. The app aims to provide the users with a convenient and efficient platform for ordering fuel to their vehicles at any given location, this can be done by using real-time GPS tracking of the delivery vehicle which will allow the users to monitor the status. Also, the app will implement secure payment integration for ordering fuel. Users will be presented with an intuitive platform that will ease the whole process of ordering and receiving fuel at their locations [2]. This will change how people approach refuelling in their day to day lives because it will be a time saving and seamless experience [2]. By using Flutter, the app will have support for both Android and iOS mobile phones making it cross platform with an attractive user interface because of Flutter. The app will also integrate with Google Maps because of its powerful navigation capabilities for real time GPS tracking. There are many benefits for this app, users will be able to order fuel from anywhere which will also provide accessibility to remote areas which will make sure fuel is available for everyone [2]. The process for the app will be simple: Users register on the app and login, share their current location, vehicle details and select the fuel type they want, order fuel, then, the fuel truck will reach the user's location at the given time. There was research conducted on this matter in which users were interviewed to find out if such an app would be successful or not, the response for this research showed that majority (68.8 percent) of the users were interested in having fuel delivered to them and they wanted such an app to be developed [3]. An in-built GPS is found in almost all mobile devices nowadays which enables such devices to accurately determine their location anywhere on Earth, this is done by receiving signals from GPS satellites [4]. Here is a simple image below to understand how the app will work [5].

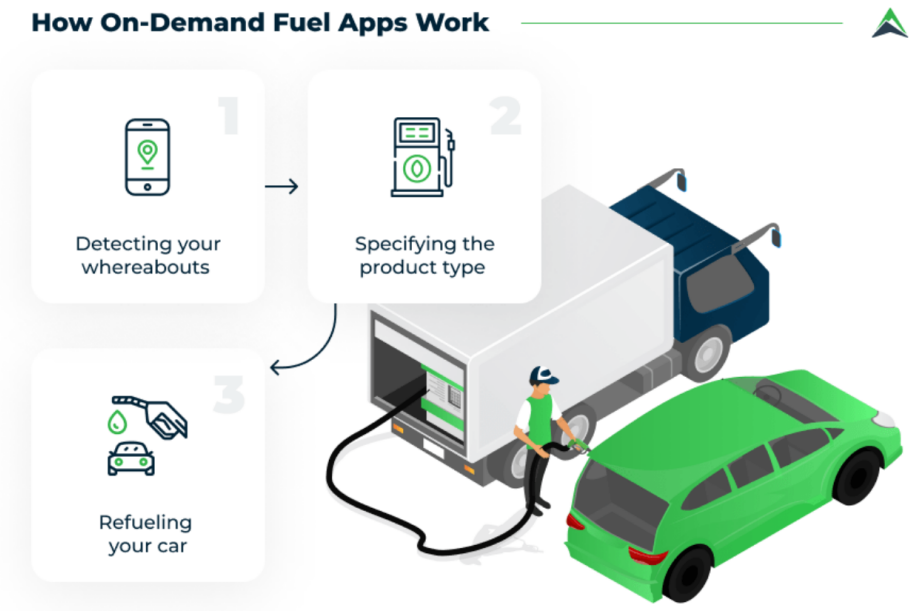

Figure 1: Workflow of the fuel delivery app.

The project's main objectives are:

1. Develop an easy to use and attractive interface/application where users can place fuel orders by selecting fuel type, and delivery location.
2. Implement real time GPS tracking to provide users with live updates on the fuel delivery status.
3. Ensure secure payment processing to protect user data.
4. Design and build a scalable system capable of handling multiple user orders and managing delivery status updates.

Timeline

Term 1

- **Week 1:** Setup the Flutter development environment for iOS and Android and research similar apps.
- **Weeks 2–3:** Working on the app design, finalizing the main features and interactions of the app, implementing firebase authentication, and building the UI for placing fuel orders.
- **Week 4:** Design database schema, some backend planning like using APIs etc.
- **Weeks 5–6:** Integrating Google maps for real-time GPS tracking, Integrating secure payment system.
- **Weeks 7–8:** Testing for core features and remaining development from previous weeks.
- **Weeks 9–10-11:** Preparing and working on the interim report and presentation, any bug fixes or quality of life improvements.

Term 2

- **Weeks 1–2:** Implementing push notifications and real time updates.
- **Weeks 3–4:** Developing interface for drivers and developing user profiles
- **Weeks 5–6:** Any remaining development and testing the performance of the app.
- **Weeks 7–8:** User Acceptance Testing for feedback.
- **Week 9:** Deployment of the app if required.
- **Weeks 10–11:** Final testing, bug fixes, preparing and working on the final report and final submission.

Planning

I will be using the MoSCoW method to prioritise the features of the app into Must Have and Should Have accordingly. My timeline and milestones will follow the prioritisation laid out in the plan below.

Must Have Features

These are the core features that are required for the app/project to function properly as intended:

- **User Registration and Login:** Users must be able to create accounts and log in to place fuel orders.
- **Fuel Ordering System:** The core feature where users select the fuel type, add all their details, select their delivery location, and place the order
- **Real Time GPS Tracking:** Users should be able to track the fuel delivery vehicle/driver in real time.
- **Secure Payment Integration:** A payment gateway must be implemented to deal with user transactions for placing orders.
- **Google Maps Integration:** This should be implemented to allow location selection and navigation for delivery.
- **Notifications:** Users should receive real time notifications about their order status and ETA.

Should Have Features

These are also important but not critical features, these will enhance the user experience or add value to the project:

- **Push Notifications:** Implementing push notifications to notify users about their order updates.
- **Driver Interface:** A seperate interface for the fuel provider drivers to update the status of the fuel deliveries.
- **Order History:** Users should be able to view their past orders and details related to these orders.
- **User Profile Management:** Allow users to manage and update their personal details, payment methods and other details.

Risks and Mitigations

- **Hardware Failure:** While working on my mobile app, there is a possibility that the hardware on which I am doing the project might fail for some reasons leading to data loss and the progress made being wasted. To counter this, I will make sure that I am working on the project and keeping everything related to it under version control (Gitlab) and I will make sure that I am committing and pushing regularly to the repository.

- **Uneven Balance Between Report and Code:** Focusing too much on either the technical part (coding) or the written reports could lead in one of them being incomplete in time or insufficient in detail. To counter this, I will regularly review progress for both to make sure that both are receiving balanced attention, and I will make sure to follow my timeline and milestones to prevent this from happening.

- **Technical Issues with GPS Tracking:** Real time GPS Tracking may be inaccurate or may fail sometimes, to mitigate this I will properly integrate Google Maps with error handling.

- **Cross-Platform Compatibility Issues:** There are differences between apps for iOS and Android which could lead to inconsistent behaviour of the app or performance issues. For this, I will test frequently on both platforms and use Flutter as best as I can.

- **Scope Creep:** A project's goal or requirements can change beyond what was originally planned upon, this can happen by adding too many features which could lead to missing deadlines. For this, I will stick to the original plan and use agile principles to manage priorities.

- **Insufficient Testing:** Poor testing during or after production might result in unfound bugs or poor user interface. For this, I will regularly test my product during development and afterwards as well.

- **Time Management Issues:** Delays in successfully achieving the milestones mentioned above in the timeline section could push the project beyond the given deadlines, so, to reduce this risk I will strictly follow the timeline, regularly review the progress, and break tasks into smaller sub-tasks which are manageable and easier to implement.

- **Lack of Required Skills:** It is possible that while working on this project, I might need to deal with new programming languages or technologies which I have not used before. For this, I will make sure that I also schedule time for learning these new skills by researching and doing.

References

[1] Mishra, N., Raghuwanshi, R., Maurya, N.K. and Kumar, I., 2023, August. Efficient Fuel Delivery at Your Fingertips: Developing a Seamless On-Demand Fuel Delivery App with Flutter. In International Conference on Cognitive Computing and Cyber Physical Systems (pp. 134-147). Cham: Springer Nature Switzerland.

[2] Kumar, A., Surya, S., Keerthiga, A. and Shalini, S., 2024. FUEL PULSE: Digital Fuel Logistics Revolutionizing Delivery Management. Available at SSRN 4776337.

[3] Prabhu, M. and Sarker, Y. . (2023) "On-Demand Fuel Delivery Mobile App for OmanOil", Journal of Student Research. Houston, U.S. Available at: https://www.jsr.org/index.php/path/article/view/2317 (Accessed: 9October2024).

[4] Gunthe, S., Sangale, A., Brahmankar, Y., Kulkarni, P. and Baddi, P., 2023. Fuel Delivery Application. International Journal for Research in Applied Science and Engineering Technology (IJRASET), 11(5), pp.1-7. Available at: `https://doi.org/10.22214/ijraset.2023.52655`.

[5] On-Demand Fuel Delivery Apps Development Explained Blog, Available at: `https://easternpeak.com/blog/on-demand-fuel-delivery-apps-development/`.