Beverages Application with Personalize Health Management

By

KONG PHUI YIN

A PROPOSAL

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR INFORMATION SYSTEMS (HONOURS)
BUSINESS INFORMATION SYSTEMS
Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2024

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Beverages Application with Personalize Health Management

Academic Session: __JAN 2024___

I _____KONG PHUI YIN_____

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

_____PhuiYin_____ _____wong_____

(Author's signature) (Supervisor's signature)

Address:
__4 Lintang Merak_____
__Desa Permai Indah Timur Laut____
__11700, Sungai Dua, Pemang_____

Ts Dr Wong Pei Voon
Supervisor's name

Date: ___22 April 2024_____ **Date**: 24/4/2024

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

UNIVERSITI TUNKU ABDUL RAHMAN

Date: __22 April 2024____

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that **_____**KONG PHUI YIN**_____**(ID No: **_20ACB02499_**) has completed this final year project/ dissertation/ thesis* entitled "**Beverages Application with Personalize Health Management**" under the supervision of _____Ts. Dr Wong Pei Voon____ (Supervisor) from the Department of ___Digital Economy Technology_____, Faculty of __Information and Communication Technology_.

I understand that University will upload softcopy of my final year project / dissertation/ thesis* in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

PhuiYin

_____

(*Student Name*)

DECLARATION OF ORIGINALITY

I declare that this report entitled "**Beverages Application with Personalize Health Management**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : _____PhuiYin_____

Name : _KONG PHUI YIN_____

Date : _22 April 2024_____

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisor, Ts. Dr. Wong Pei Voon, and my moderator, Mr. Tey Chee Chiah who have given me this golden opportunity to engage in a mobile application development project. It is my first step to establishing a career in the application development field. A million thanks to my supervisor and moderator.

To a very special person in my life, my supervisor, Dr. Wong, for her patience, unconditional support, encouragement, and for standing by my side during hard times. Finally, I must say thanks to my parents and my family for their love, support, and continuous encouragement throughout the course.

ABSTRACT

This project endeavors to develop a beverage application integrated with a personalized health management system to address deficiencies in existing beverage applications. The objectives focused on implementing drink recommendations, enhancing health management features, and improving categorization. Leveraging industry trends and successful models like Zus Coffee, the project utilized React Native for cross-platform compatibility, APIs for weather-based recommendations, and Firebase for data management.

The review of existing applications highlighted gaps in recommendation systems, health management tools, and categorization features, underscoring the need for innovation. The implementation phase prioritized user-centric design, incorporating personalized drink recommendations, health alerts, and efficient categorization and search functionalities. Black box testing evaluated user and admin functionalities, validating successful implementation of objectives.

The evaluation of the project's success is based on several key criteria. The implementation of drink recommendation features was validated through user testing, assessing the effectiveness of weather-based recommendations and user preferences. Health management tools were evaluated based on user engagement with health alerts and guidance. The categorization and search functionalities were tested for usability and efficiency, ensuring ease of navigation and access to beverage options.

Black box testing provided insights into the application's performance across user and admin interfaces, highlighting strengths and areas for improvement. User feedback and satisfaction surveys were instrumental in assessing the application's impact on user experience and wellness promotion. Overall, the project's success is measured by its ability to address user needs, enhance engagement, and promote healthier beverage choices through innovative digital solutions.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

API	Application Programming Interface
UI	User Interface
UX	User Experience
QR Code	Quick Response Code
RAM	Random Access Memory

Chapter 1 Introduction

In today's era of rapid technological advancement and growing awareness of personal health, the integration of digital tools into our daily routines has transformed how we approach wellness. One such innovation, the Beverages Application, exemplifies the intersection of technology and health management by offering a personalized platform to track, analyze, and optimize beverage consumption based on individual health profiles and goals. Order ahead applications have changed the customer dining experience and provide the ultimate convenience. People can order the food and beverage they wish anytime and anywhere through the applications of food and beverages.

Toast's 2017 Restaurant Technology Industry Report showed that 95% of restaurant owners concur that technology boosts their company's productivity and improves their business efficiency [1]. Food and beverages applications are now getting popular over the years. There are many types of food and beverage applications offered to users. However, there's a unique combination of a food and beverage application integrated with personalized health management systems, which addresses the concerns most prevalent among people today.

Furthermore, there are many famous beverage brands that offer official mobile applications for their own brand, such as Starbucks, Zus Coffee, Tealive, and more. One of the popular latest beverage applications in Malaysia is the Zus coffee application that we can study for this proposed project. Zus emphasized to the public that Zus Coffee is Malaysia’s first tech-driven coffee chain [2]. Innovative coffee in an innovative era is the slogan of Zus where their retail model is built upon their mobile app and store network. However, the Zus Coffee application received positive feedback from the public about its beverage application and the application’s ranking has gone to the top 1 in the Appstore (Figure 1.0). Zus received over 4.9/5 positive reviews on their application and there are 116,000 users who participated in the review section (Figure 1.1). Hence, the Zus coffee application plays an important role in this chapter for us to study their technology, problem, and motivation for this project.

In this chapter, we focus on the project background, problem statements, motivation, objectives, project scope and direction, and contribution of the proposed beverage application.

1.1 Problem Statement and Motivation

1. No feature recommending drinks to users.

Starbucks and Zus Coffee are two well-known beverage applications that allow users to browse menus, locate nearby stores, and in some cases, offer delivery services based on the user's geographical location. However, research from Fyp1 revealed that neither Starbucks nor Zus Coffee initially included features that recommend their best-selling products or suggest drinks to users. Fortunately, Zus Coffee has since updated their application to include recommended drinks, whereas Starbucks has not made this enhancement yet. For first-time users ordering beverages through these apps, the absence of recommendations can make decision-making challenging. Therefore, it is crucial for us to incorporate recommendation features that aid users in making choices and ultimately boost sales.

Additionally, the current objective of the Starbucks application primarily focuses on allowing users to register as members and accumulate points, with other features still deemed immature based on App Store reviews (Figure 1.2).

2. Lack of health management system to guide users select the products.

Health concerns products getting popular nowadays. Research showed that the current trend of users who are focusing on the product based on their health status has reached major consumers worldwide. Research showed that the relationship between health concern of developing diseases and attitudes toward healthy eating was fully mediated by food choice motives [3]. Furthermore, according to the F&B Formulation trend 2 which is reducing sugar said that people focus of health concerns products especially in the food and beverages industry includes sugar level, fats, and nutrition [4]. Therefore, it is important to develop a food and beverages application integrated with personalized health management to maximize customer satisfaction and increase user engagement by catering to individual preferences and needs such as health concerns.

3. Lack of categorization and search field function of products.

In the current version of the beverage's application, there is a notable issue regarding the lack of product categorization. This absence of categorization presents a significant challenge for users seeking to efficiently navigate and explore beverage options. The absence of clear categorization impedes user experience and may result in frustration and reduced engagement

with the application [5]. Therefore, addressing this issue is crucial to enhance usability and overall user satisfaction with the beverage's application.

However, search functionality plays an important role in improving the usability and general efficacy of a beverage application. By enabling users to quickly locate specific beverages or categories of interest, a well-designed search feature streamlines the browsing experience and empowers users to navigate the application with ease [6]. This functionality not only enhances user satisfaction but also promotes increased engagement and adoption of the application, ultimately contributing to a positive user.

1.2 Objectives

- **Include the feature of recommending the best-selling product or recommended drinks for the user.**

There are 2 ways to develop the feature of recommendation for the best-selling product or recommended drinks to the users. The first way is to access the Firebase user's database and get the related data and analyze it to convey meaningful information. For example, we can access the user's purchasing record in the database in Firebase and find the highest frequency drinks that have been bought by all the customers. Hence, recommending the highest frequency drinks that have been bought to the other user.

The second way is to recommend drinks to users according to the weather. APIs are used to gain access to get the weather temperature and the location of the users. Thus, recommend drinks to users according to the weather. OpenWeather provides free API services for developers to use in developing related features. In this case, we use the free APIs keys generated from the OpenWeather website access to the weather and location information and implement the recommendation features.

- **Implement a personalized health management system for user guidance.**

Feature a personalized health management system in the proposed beverage application to guide users in selecting the beverages products based on their health condition. During the account registration process, users will be prompted to input their diabetes and cholesterol levels, which will be stored in Firebase. Users with diabetes or high cholesterol will receive alert messages when selecting drinks, advising them on healthier options. However, the system will not block users from choosing drinks with normal sugar levels but will provide informative alerts. For instance, diabetic users will be recommended drinks with lower sugar content.

Furthermore, the personalized health management system will alert users when selecting larger cups of beverages containing caffeine. While coffee does not contain cholesterol, research suggests that it can negatively affect cholesterol levels [7]. Therefore, the system will educate users about potential health impacts associated with caffeine consumption.

- **Include the proper categorization and search field features.**

Design an informative categorization section that organizes drinks into distinct categories such as Hot Series of Coffee, Ice Series of Coffee, Kids Friendly, and Non-Coffee options. This categorization aims to provide users with informative information, aiding them in selecting drinks based on their specific needs and preferences.

Furthermore, incorporating a search field feature within the proposed beverages application aligns with research indicating its positive impact on User Experience (UX) design, as highlighted in the problem statement.

Implementing proper categorization and a search field will significantly enhance user experience, encouraging increased usage and engagement with the application. Moreover, the well-designed categorization feature will support the achievement of the project's first objective, which involves recommending drinks to users based on respective categories corresponding to weather conditions.

1.3 Project Scope and Direction

The goal of this project is to develop a comprehensive beverage mobile application with a personalized health management system for Android and iOS users. Key features include recommending best-selling products or drinks, implementing a personalized health management system for user guidance, and design proper categorization and search functionalities.

The development of the beverage application will leverage APIs, React Native development tools, Visual Studio Code, and Firebase technologies. The target audience comprises Millennials (aged 18-34) and Generation X (aged 35-50), aligning with research indicating high usage of food and beverage delivery applications among these demographics [8].

Project Scopes

- Feature recommending the best-selling product or recommended drinks using APIs.
- Develop a personalized health management system to guide users based on their health conditions.
- Include the proper categorization and search field features to enhance user experience and drink selection.

1.4 Contributions

One key contribution of this project is the **implementation of a personalized health management system within the beverage application.** By incorporating this feature, users can receive tailored guidance based on their specific health conditions. For instance, individuals with diabetes or cholesterol concerns will be alerted to suitable beverage choices, promoting healthier consumption habits and overall well-being. This personalized approach not only enhances user satisfaction but also demonstrates a commitment to user-centric design and health-conscious solutions in mobile applications.

Furthermore, the inclusion of **recommended drinks using APIs** enhances the user's experience by providing valuable insights into top-rated beverage options according to the weather. Leveraging external data through **APIs allows the application to provide informative information such as real time weather and location**, ensuring users have access to the recommended beverages. This feature contributes to increased user engagement and satisfaction, fostering a positive relationship between consumers and the application.

Another significant contribution lies in the design and **implementation of proper categorization and search field functionalities**. By categorizing beverages into meaningful groups such as Hot Series of Coffee, Ice Series of Coffee, Kids Friendly, and Non-Coffee categories, users can easily navigate and explore beverage options based on their preferences and needs. The addition of a robust search functionality further enhances usability, enabling users to quickly find specific beverages within the application. These features collectively contribute to an intuitive and user-friendly interface, improving overall user experience and encouraging continued usage of the beverage application.

In terms of technological contribution, the utilization of modern development tools like **React Native**, **Visual Studio Code**, and **Firebase** signifies a commitment to efficiency and scalability in mobile app development. React Native enables cross-platform compatibility, allowing the beverage application to reach a broader audience across Android and iOS devices. Visual Studio Code enhances code editing and development workflow, facilitating the creation of robust and reliable features. Firebase, on the other hand, provides essential backend services such as data storage, authentication, and real-time updates, ensuring the application operates seamlessly and securely.

1.5 Report Organization

Chapter 1: Introduction

The introduction section of this report provides an overview of the project's objectives, motivations, and the problem statement. It highlights the significance of integrating digital tools into health management within beverage applications.

Chapter 2: Literature Review

The literature review examines industry trends and successful models like Zus Coffee, emphasizing the importance of personalized recommendations and health management systems in enhancing user engagement and satisfaction.

Chapter 3: System Design

The system design phase focuses on a user-centric approach to application design, ensuring intuitive navigation and seamless interactions. It details the implementation of features like personalized drink recommendations and health alerts.

Chapter 4: Implementation

The implementation section discusses the technologies used, such as React Native, APIs for weather-based recommendations, and Firebase for data management. It also covers the prioritized scalability and cross-platform compatibility of the application.

Chapter 5: System Evaluation and Discussion

System evaluation involves rigorous black box testing across user and admin interfaces. It evaluates functionalities related to account management, categorization, search, checkout, and health management, demonstrating successful implementation of defined objectives.

Chapter 6: Project Challenges

This section outlines challenges faced during development, such as managing incompatible React Native libraries and configuring the development environment for optimal performance.

Chapter 7: Conclusion and Recommendations

The conclusion summarizes project outcomes, highlighting the successful implementation of personalized recommendation features, health management systems, and efficient categorization and search functionalities. It also provides recommendations for future enhancements, including machine learning-based recommendations, integration with wearable technology, analytics dashboard development, and continuous user feedback loops.

Chapter 2 Literature Review

2.1 Introduction

In recent years, many food and beverage apps have emerged due to advances in technology and a growing focus on personal health. However, research shows that very few of these apps fully incorporate personalized health management features. The proposed beverage app with personalized health management is an innovative blend of technology and health, offering a customized platform to monitor, analyze, and improve beverage choices based on individual health needs and goals. In this chapter, we will discuss what has been done by other researchers or developers to resolve the problem stated in Chapter 1, the strengths of their solutions, the weaknesses, and limitations of the solutions, and how they can be resolved. The reviewed topic includes the feature of the related application and the technology used in this application which is APIs. In addition, similar beverage applications including Starbucks, Zus Coffee, and Tealive of the related features are being reviewed.

2.2 Review on types of Technology use for proposed beverage application with personalized health management.

2.2.1 Application Programming Interface, API

An Application Programming Interface (API) acts as a crucial software intermediary that enables communication and interaction between different applications [9]. In this chapter, we explore the utilization of APIs to address specific functionalities within our proposed beverage application, particularly focusing on the implementation of features related to recommending drinks based on weather conditions and track location.

Both the Android and iOS platforms offer robust APIs that empower developers to access user location data and develop geolocation-based mobile applications [10]. This technology allows our beverage app to provide personalized recommendations and services tailored to the user's current location and weather conditions.

Furthermore, API keys from OpenWeather are instrumental in the development of our proposed beverage application, providing free access to their weather APIs (Figure 2.2) for retrieving current location and weather data from users. However, we also reviewed the differences between the APIs and APIs keys. APIs keys provide project authentication and project identification and make the call to an API. APIs keys are used to Identify the application

that calls to this API Project authorization and to check if the calling application has been granted access to call the API and has enabled the API in their project [11].

However, despite the utility of APIs, there are certain limitations where there are limits and quotas on the API request. In the API Console, there is a limitation of the quota to as Requests per 100 seconds per user and it is set to 100 requests per 100 seconds per user. The number of requests to the API is restricted to a maximum of 10 requests per second per user although the limit can be adjusted to the maximum value of 1000 requests [12].

2.2.2 React Native libraries.

React Native is a popular framework for building mobile applications using JavaScript and React. It allows developers to create cross-platform applications that can run on both iOS and Android devices with a single codebase.

In the context of React Native, a "library" refers to a collection of pre-built components, modules, and utilities that developers can use to build mobile user interfaces and add functionality to their applications [13]. These libraries are typically created and maintained by the React Native community or third-party developers.

To develop the proposed beverages application with personalized health management, React Native libraries play the crucial roles for us to integrate our project with the built-in function and the most common one such as React Native Navigation, Axious, Firebase and Ionicons of library. Additionally, React Native libraries are commonly obtained from the npm registry through a Node.js package manager like npm CLI or Yarn Classic for installation.

2.3 Review the features on similar beverages applications.

2.3.1 Feature recommending the best-selling product or recommended drinks for the user.

Products recommendation play an important role in online applications store strategy. If the products were present and suggested to the user at the right time, this may improve the average order value of the product. However, there are limited beverages application align the recommendations features with their beverage’s application. From the research showed that most of the applications such as Starbucks, and Tealive does not included the recommended drinks feature, However, there are many ways to recommend the product to the user such as

recommending the product based on the reviews, using dynamic product recommendations, promoting what others or major customers have bought, or more ways to recommend the drinks or products to the user [14]. For example, recommending a product based on reviews can be done by showing users the “Best Reviewed Products” or displaying how many stars each item has received from reviewers.

The limitation of this reviewed feature is that the recommended drinks or the best-selling products may not always reach the target market or individual users' preferences. For example, relying solely on previous purchase frequency for recommendations may not be reliable if some users consistently buy the same product.

A more effective approach is to leverage APIs for recommending drinks. By using weather APIs to guide recommendations, we can enhance customer satisfaction and cater to a broader range of user preferences. This method involves tracking the user's current location and weather conditions. For instance, recommending beverages from the "Icy Series" category when the temperature exceeds 30 degrees Celsius, or suggesting hot beverages otherwise, can provide more relevant and personalized recommendations based on real-time factors.

2.3.2 Personalized health management systems guide users in selecting the beverages products based on their health condition.

As health concerns become increasingly prevalent in the market, existing beverage applications like ZUS Coffee, Starbucks, and Tealive have not yet addressed this growing trend. The proposed beverages application is suggested to include the personalized health management in its application to gain competitive advantage and apply the innovative and creativity to the application where the user is alert to suitable beverage choices, promoting healthier consumption habits and overall well-being. This helps to improve the overall performance of the application and gain the competitive advantage in the market.

However, one limitation of personalized health management is related to privacy concerns. Not all users are comfortable sharing personal health information, such as their health details, especially on social platforms. Without complete user information, it's challenging to effectively implement and track personalized health management features.

2.3.3 Categorization and search field features.

According to research, existing beverage applications like Zus and Starbucks do provide categorization fields for items. However, as shown in Figure 2.3 and Figure 2.3.1, the categories in both applications are complex and numerous, which can make it challenging for users to focus on finding their desired products. This complexity may lead to decision paralysis and decreased user satisfaction. An article has discussed the concept of choice overload, suggesting that an abundance of options can lead to decision paralysis and decreased satisfaction. This principle can be applied to category design, where a smaller set of well-organized categories can make it easier for users to navigate and make choices without feeling overwhelmed [15].

- Search field features.

Zus and Starbucks currently lack a search function within their beverage applications, preventing users from efficiently searching for specific products. The absence of this feature can hinder user experience by making it difficult for users to quickly locate and access desired items. Implementing a search function would enhance usability and convenience, allowing users to easily find and select their preferred beverages within the application.

Summary:

Technology and Feature	**Advantages**	**Limitations**
APIs	Gain access to use the services.	There are limits and quotas on the API request.
React Native libraries	Provided numbers of the built-in components.	Incompatible format.
Drinks recommendation	Improve the average order value of the product.	May not reach the target user or market / can't meet all the user preferences
Personalized health management	Gain competitive advantage.	Privacy Issues.
Categorization	Better user experience.	Not meaningful categories can be made.
Search Field	Better user experience.	Search accuracy.

Table 2.3.3 Advantages and Limitations of existing technology and features reviewed.

2.4 Comparison between proposed application with others existing reviewed application.

Application / Features	Starbucks	Zus Coffee	Tealive	Proposed Beverages Application
Platform	IOS/Android	IOS/Android	IOS/Android	IOS/Android
Categorization	✓	✓	✓	✓
Recommending Drinks	×	✓	×	✓
Personalized Health Management	×	×	×	✓
Search Field	✓	×	×	✓
Add to cart (Edit, Delete)	×	✓	✓	✓

Table 2.3.3.2 Comparison between proposed beverages application with existing application.

Chapter 3 System Methodology

3.1 Methodology

A methodology refers to a structured approach for applying the System Development Life Cycle (SDLC). The SDLC methodology encompasses the planning, analysis, design, implementation, and maintenance of software applications [16]. It provides a set of guidelines to ensure that software development projects are completed within specified timelines, budgets, and quality standards. The methodology comprises common phases such as planning, analysis, design, and implementation, each tailored to achieve specific objectives and deliverables. The diagram below illustrates the typical stages involved in the SDLC method. This systematic framework ensures efficient and effective software development processes, leading to successful project outcomes aligned with business objectives and user requirements.

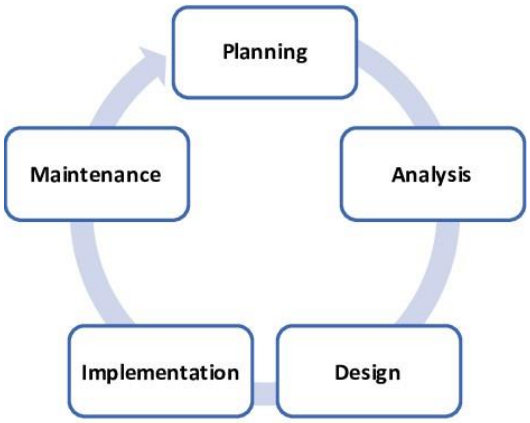

Figure 3.1 System Development Life Cycle (SDLC)

The five key phases of a simple SDLC methodology includes Planning, Analysis, Design, Implementation, and Maintenance. During the Planning phase, project goals, scope, and resource allocations are defined. The Analysis phase involves detailed requirements gathering and stakeholder consultations to understand specific needs. In the Design phase, system architecture and specifications are created based on gathered requirements. Implementation is where the actual coding and development of the software occur, translating designs into functional software. Finally, the Maintenance phase involves ongoing support, bug fixes, and updates to ensure the software remains effective and aligned with evolving user needs. These

phases ensure a systematic approach to software development, leading to the creation of high-quality software that meets user expectations and business objectives.

3.1.1 Prototyping Methodology

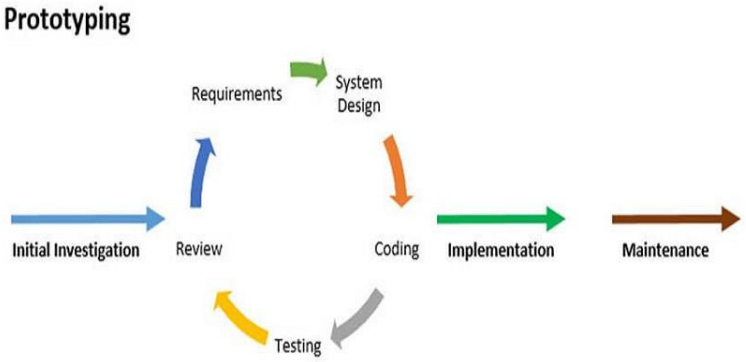

Figure 3.2.7 Prototyping Methodology

The System Development Life Cycle Methodology to be used in this proposed beverage application is the Prototyping Methodology. The purpose of using this methodology is to create an initial version of a software product or system that can be tested and evaluated before the final version is built. Furthermore, there are 5 stages involved in the prototyping methodology used in this proposed application. The first stage is **requirements gathering**. The requirement and resources of the proposed beverage application will be collected as much as possible to determine the possible problems and user expectations. Second, is the **design stage**. Based on the information that has been collected design for the proposed application's prototype. The design is based on the user's needs and requirements and should be as detailed as possible to meet the user's expectations. The third stage is **to build**. In this stage, the designed prototype is built, and this can be done using a variety of software development tools and platforms. In this case, we are using React Native, Expo Go, Firebase, and Visual Studio 2022 to complete the SDLC life cycle. Next, is **the most important process, the testing stage**. The prototype is being tested in this stage and evaluates the outcomes where to ensure the prototype meets the user's requirements. The last step in this prototyping methodology is the **refining stage**, based on the feedback gathered during the testing phase, the prototype is refined and improved. This may involve making changes to the design, functionality, or user interface.

In addition, the prototyping methodology allows the developers to repeat the process several times until meets the user's requirements, unlike the waterfall methodology. Hence, prototyping methodology is a useful approach to software development because it allows developers to identify and address design flaws and functional issues before development

begins, ultimately resulting in a better system. Below is the prototyping methodology that is divided into three phases of the proposed beverage application development.

First phase – Requirements Gathering

Key tasks in this phase:

- Business Understanding and Market Research
- Identify problem statement.
- Establish the scope of the project.
- Outline the goals of the project.

During the requirements gathering stage, several well-known beverage applications currently available in the market are studied and reviewed for insights into the proposed beverage application. This includes studying existing applications like Zus Coffee, Starbucks, and Tealive to gain insights into their functionality and features. Next, identify problem statement, project scope, and project goals. At this phase, a weekly meeting with the supervisor is required. This phase results in the creation of a preliminary report that includes the backdrop, problem description, project scope, and project goals of the application.

Second phase – System Analysis, Design, and Implementation

Key tasks in this phase:

Analysis:

- Benchmark the project.
- Identify project innovation and contribution.
- Identify the technologies and software used.
- Compare the existing beverages application.

Design:

- Describe the methodology used.
- Create Gantt chart.
- Design diagrams (Block, UML, Class diagram)
- Create prototype.

Implementation:

- Code the system to translate the design into functional software.
- Develop a comprehensive test plan to identify and resolve issues or bugs.
- Demonstrate the system's functionality.
- Present the final project.

In the second stage, the fundamental design will be evaluated based on the review in stage 1. After the fundamental design is designed without any problems, the first, prototype will be built to check whether the function is suitable and buildable. After the prototype is completed, it will be evaluated. In this case, the design of the proposed beverage application was done by using the research method to study other similar applications and guide the overall idea of the design. For example, almost every beverage application shares the same design layout and elements to improve user experience.

Third phase – Iterative Development/Design, Coding, Testing, Maintenance

- Refine and enhance the design prototype based on iterative feedback.
- Implement coding changes to improve software functionality and address feedback.
- Conduct rigorous testing to ensure system stability and identify and resolve issues.
- Maintain and optimize system performance through ongoing updates and enhancements.
- Ensure the software meets specified requirements and user expectations.
- Iterate on the development process to continuously improve the application based on feedback and testing results.

In the final stage, an appropriate design prototype will be coded, tested, and maintained. The proposed project will be implemented through coding. In this proposed project, Visual Studio 2022 is used for the functionality and features of application coding and front-end design code. At the same time, the data should store in the real-time database in which Firebase is to be used in this proposed project. In addition, after implementing the system, a testing phase will be performed to find out whether the coding is correct. Expo Go and React Native play the role in the testing of the proposed beverage application. Lastly, the developers need to perform maintenance work to ensure the performance of the system.

3.1.2 Hardware

A computer and IOS devices are used as a hardware in this project. By using computer can quickly read through lengthy, coded manuals to create and maintain a successful beverages application. IOS device is used for testing and deploying the performance of the proposed beverages application.

Description	Specifications
Model	Lenovo T460s
Processor	Intel(R) Core (TM) i7-6600U
Operating System	Windows 10
Graphic	Intel (R) HD Graphics 520
Memory	8GB RAM
Storage	512 GB

Table 3.2.1 Specifications of laptop

3.1.3 Tools and Technology Involved

Tools and Technologies	Description
React Native	- An open-source framework supporting app development for Android, iOS, and Windows platforms. Used by popular applications like Instagram, Netflix, and WhatsApp. - Easy to update by using Code Push techniques [14]
Node.js	- Event-driven, non-blocking I/O environment suitable for real-time web applications with push functionality [17]. - Simplifies client-server communication [18].
Expo	- Provides UI components and tools for developing and testing React Native applications without native code. - Eliminate the process of creating native iOS or Android code [19].

Visual Studio Code	- Lightweight source-code editor with robust features like IntelliSense, parameter hints, and graphical debugging [20]. - Allow user to debug their code and run the react-native commands from the command palette [21]. - Fantastic tool for web technologies and comes with enhanced built-in support for Node.js development [22].
Firebase	- Enable secure direct client-side code access to the database and can help create sophisticated, collaborative apps. - Allows control access to data in an application. (Log In and Log Out) - Data is locally stored, and real-time events continue to happen even when the user is offline, providing a responsive experience [16].
OpenWeather	- Provides weather data and weather-related information through APIs (Application Programming Interfaces). - Offers real-time weather data, forecasts, historical weather data, and weather maps. - Integrate OpenWeather APIs into their applications to access weather-related information and can be fetched based on geographic coordinates [10 new].

Table 3.2.2 Tools and Technologies Involved

3.2 Requirements and Specifications

The requirements form the proposed system was categorized into functional and non-functional requirements.

3.2.1 Functional Requirements

These are the compulsory functional requirements in the proposed beverages application with personalized health management that describe that what system can do and the features it should have to meet the needs of the user.

Users:

1. Allow existing users to log in and new users to register account with unique email and password.
2. Allow users to fix their password with forget password features.
3. Allow user to select and view various categories of beverages.
4. Allow users to search for the desired beverages.
5. Allow users add to cart to make payment.
6. Allow users view, edit, and delete the products added in cart in the checkout screen showed with the accumulated price.
7. Diabetes and Cholesterol users will be alerted to suitable beverages choice.
8. Allow user to make payment.
9. Allow user to update profile.
10. Allow user to submit feedback.

Proposed Beverages Application Admin

1. Allow admin to register new account and log in to existing account.
2. Allow admin to upload the new beverages drinks.
3. Allow admin to edit the beverages drinks from the application.
4. Allow admin to delete the beverages drinks from the application.
5.

3.2.2 Non-Functional Requirements

These are requirements that specify how the system should perform or operate, rather than its functionalities.

Usability:

1. The proposed beverages application must be user-friendly, with a simple and consistent user interface that allows users to have a better experience and a sense of 'ease of use,' particularly for new users.
2. When the input data is irrelevant or incorrect, the proposed beverages application must be able to show an error message.
3. Meaningful Categorization.

Availability:

1. The proposed beverages application must be accessible whenever and whenever the user has an internet connection (view menu etc.).

Security:

1. The website must prevent unauthorized access to the system.

2. The website should be able to periodically back up the data to safeguard against data loss.

Performance:

1. There must be no lag time between the user clicking a link and seeing the result on the website.

3.3 Timeline

3.3.1 Timeline (Gantt Chart) – FYP1

Activity	Period														
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14	
1. Introduction															
Background Introduction															
Problem Statements															
Project Scope															
Project Objetives															
Contributions															
2. Literature Reviews															
Introduction															
Review of Types of Technology															
Review similar system's features and functionalities															
3. Design and Methodology															
Block Diagram															
Use Case Diagram															
Activity Diagram															
Class Diagram															
Methodology															
Timeline															
4. Preliminary Work															
Test Prototype															
5. Conclusion															
Report Submission															
Presentation															

Figure 3.4.1 Timeline – FYP1

3.3.2 Timeline (Gantt Chart) – FYP2

Activity	Period													
	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
1. Development Phase														
Develop each module														
Fix Bugs														
2. Implementation Phase														
Code the system														
Test the system														
Identify and resolve issues or bugs														
Demonstrate the system's functionality														
3. Evaluation Phase														
Evaluate System														
Refine and enhance the design prototype														
Submission of the project														

Figure 3.4.1.2 Timeline – FYP2

Chapter 4 System Design

4.1 Block Diagram

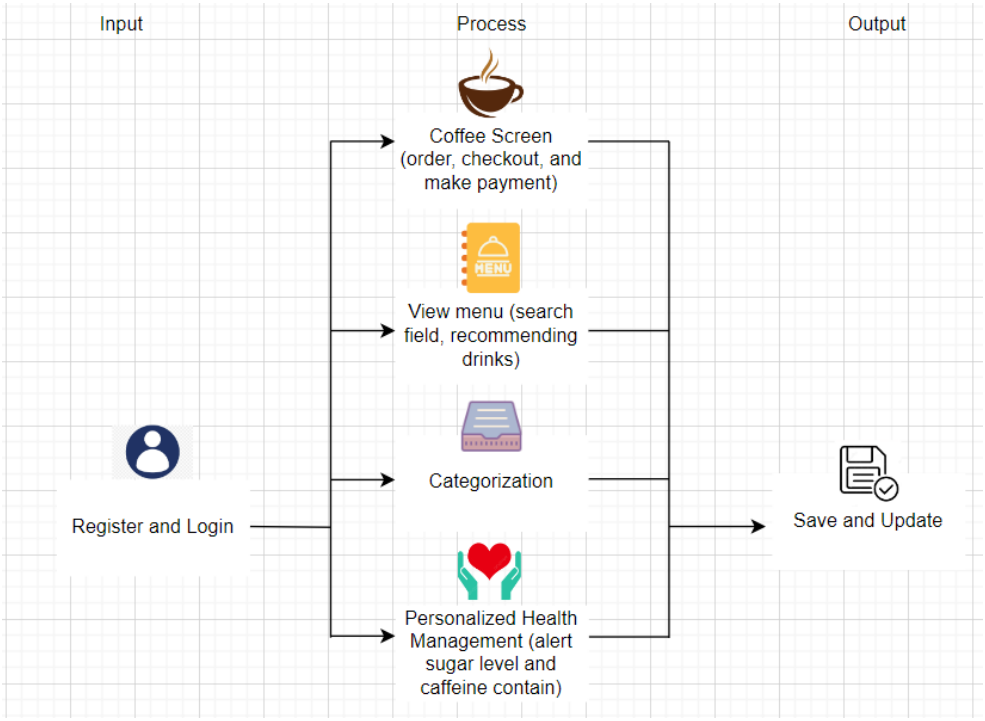

Figure 4.1 Block Diagram for Proposed Beverages Application

Input (Register and Login):

Users interact with the proposed beverages application by registering or logging in to their accounts.

Process:

- **Coffee Screen:**

  This component manages the coffee ordering process, including selecting items, checking out, and making payments.

- **View Menu:**

  Users can view the menu, which includes features such as a search field for finding specific drinks and recommendations for drinks based on preferences.

- **Categorization:**

  The application categorizes drinks into different groups such as hot, cold, kids-friendly, non-coffee for easier navigation and selection.

- **Personalized Health Management:**

This feature provides health-related alerts, such as notifications about sugar levels and caffeine content in drinks based on user preferences and health conditions.

Output (Saved and Update):

The application saves and update user information, preferences, and health-related information.

4.2 UML Diagrams

4.2.1 Use Case Diagram

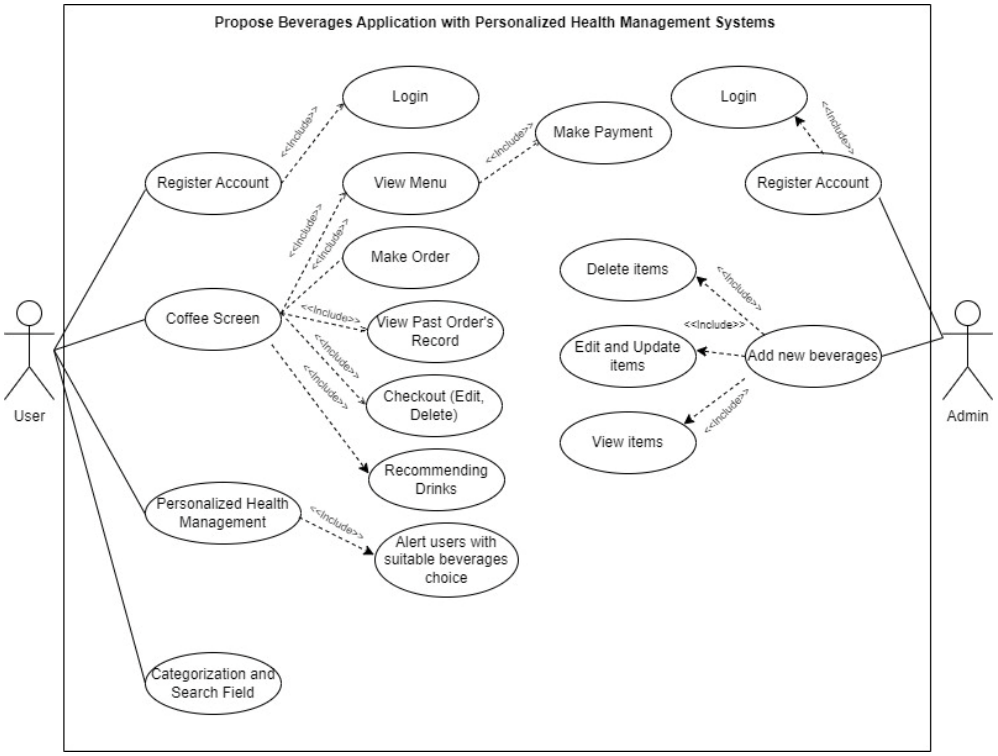

Figure 4.2 Use Case Diagram for Proposed Beverages Application

Figure 4.2 describe the main characters in the proposed beverages application and its functionalities within the proposed beverages application for both user and administrator. Below is the description for the 2 main characters of the proposed beverages application.

User:

- **Register Account includes Login:**

  Users can create a new account by registering with their details. They can then use these credentials to log in to the application.

- **Coffee Screen includes View Menu, Make Order, View Past Order's Record, Checkout, Recommending Drinks:**

  Users can view the menu of available beverages, make orders, and view their past order records. They can proceed to checkout to complete their purchases. The application also recommends drinks based on the weather and temperature condition.

- **Personalized Health Management includes Alert users with suitable beverage choices:**

This feature alerts users about suitable beverage choices based on their health preferences or restrictions.

- **Categorization and Search Field:**

  Users can browse beverages through categorized lists and use a search field to find specific items.

Admin:

- **Register Account, includes Login:**

  Admins have the ability to register an account and log in to the administrative interface of the application.

- **Add new Beverages, includes Delete Items, Edit and Update Items, View Items:**

  Admins can manage the beverage inventory by adding new items, deleting existing ones, editing item details, and updating the inventory. They can also view a list of all items available in the application.

4.2.2 Activity Diagram

i. Register and Login Account Use Case Description and Activity Diagram

Use Case Name	Register and Login Account
Purpose	Allow user to register new account and login to existing account.
Actor	User, Admin
Normal Flows of Event	- User login by entering correct email and password. - System checks for the authentication. - Redirect user to home screen/ Coffee Screen.
Alternate Flows	- If user haven't registered an account can be made through account registration page. - Registration info will be saved on Firebase.

Table 4.2.2 Use Case Description for Register and Login Account

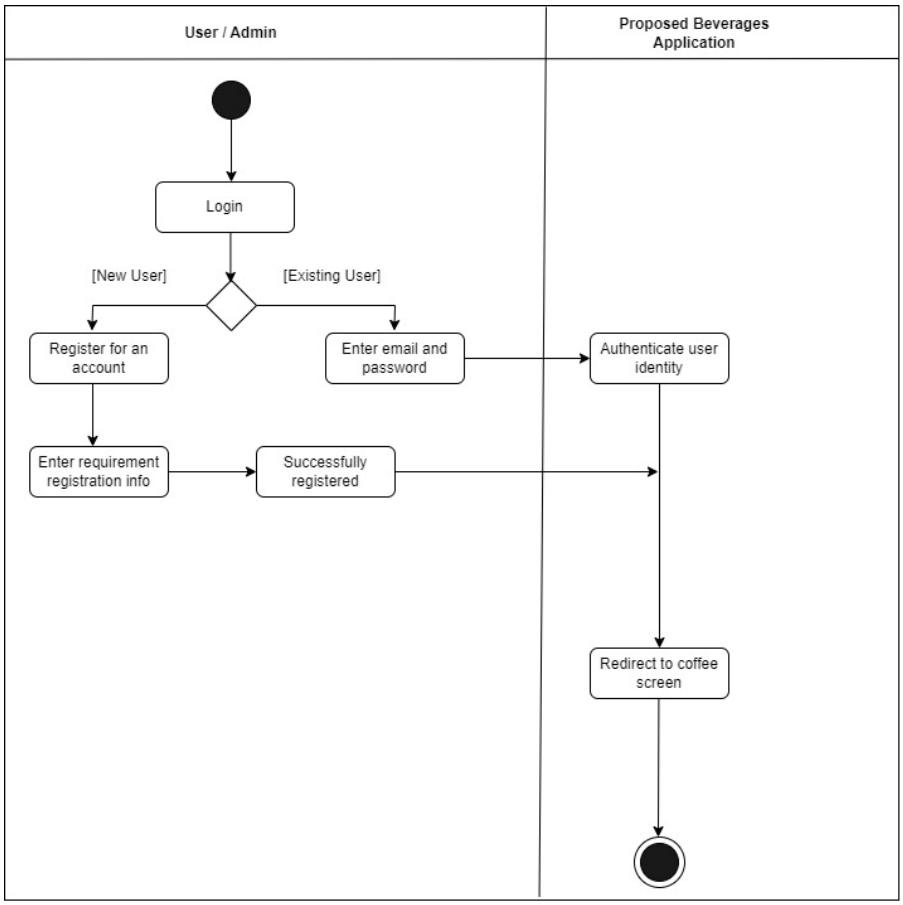

Figure 4.2.2 Activity Diagram for Login and Register Account

This use case outlines the process of user authentication within the beverage application. Users can log in by entering their email and password, and upon successful authentication, they are redirected to the main screen of the application. If a user does not have an existing account, they can proceed to create one through the registration page, where their registration details are securely stored for future login and account management. This process ensures secure access to the application's features and functionalities for both users and admins.

ii. Coffee Screen/ Home Screen Use Case Description and Activity Diagram

Use Case Name	Home Screen Functionalities
Purpose	Allow user to make order.
Actor	User
Normal Flows of Event	- User will be recommended drinks corresponding to the weather condition. - User able to view menu. - User able to select products and add to cart. - User can checkout the cart and make payment. - User able to view the order history.
Alternate Flows	- If the user is the diabetes or cholesterol patient, the system will alert user with suitable beverages drinks option.

Table 4.2.2.1 Use Case Description for Coffee Screen

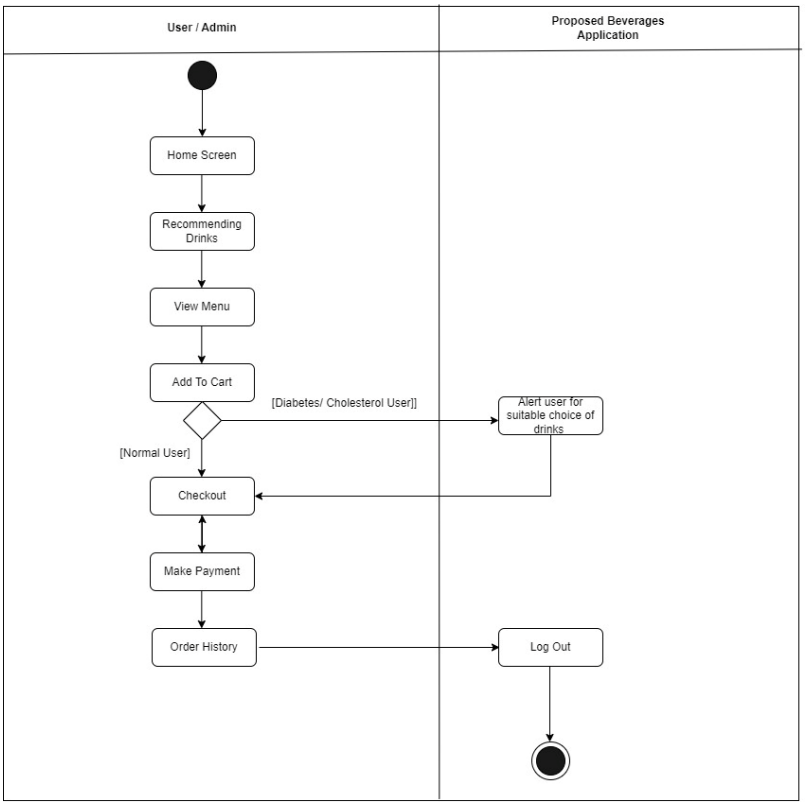

Figure 4.2.2.1 Activity Diagram for Coffee Screen

This use case describes the key functionalities available on the home screen of the beverage application. Users can receive beverage recommendations based on weather conditions, browse the menu, add items to their cart, checkout, make payments, and view their order history. Additionally, users with specific health conditions will receive personalized alerts and recommendations for suitable beverage options to meet their health requirements. This enhances the user experience by providing tailored recommendations and convenient ordering features.

iii. Categorization and Search Field Use Case Description and Activity Diagram

Use Case Name	Categorization and Search Field
Purpose	Allow user to view various categories of drinks. Allow user to search for the items.
Actor	User
Normal Flows of Event	- User able to view the drinks by categories. - User able search for the desired items.

Alternate Flows	- None

Table 4.2.2.2 Use Case Description of Categorization and Search Field

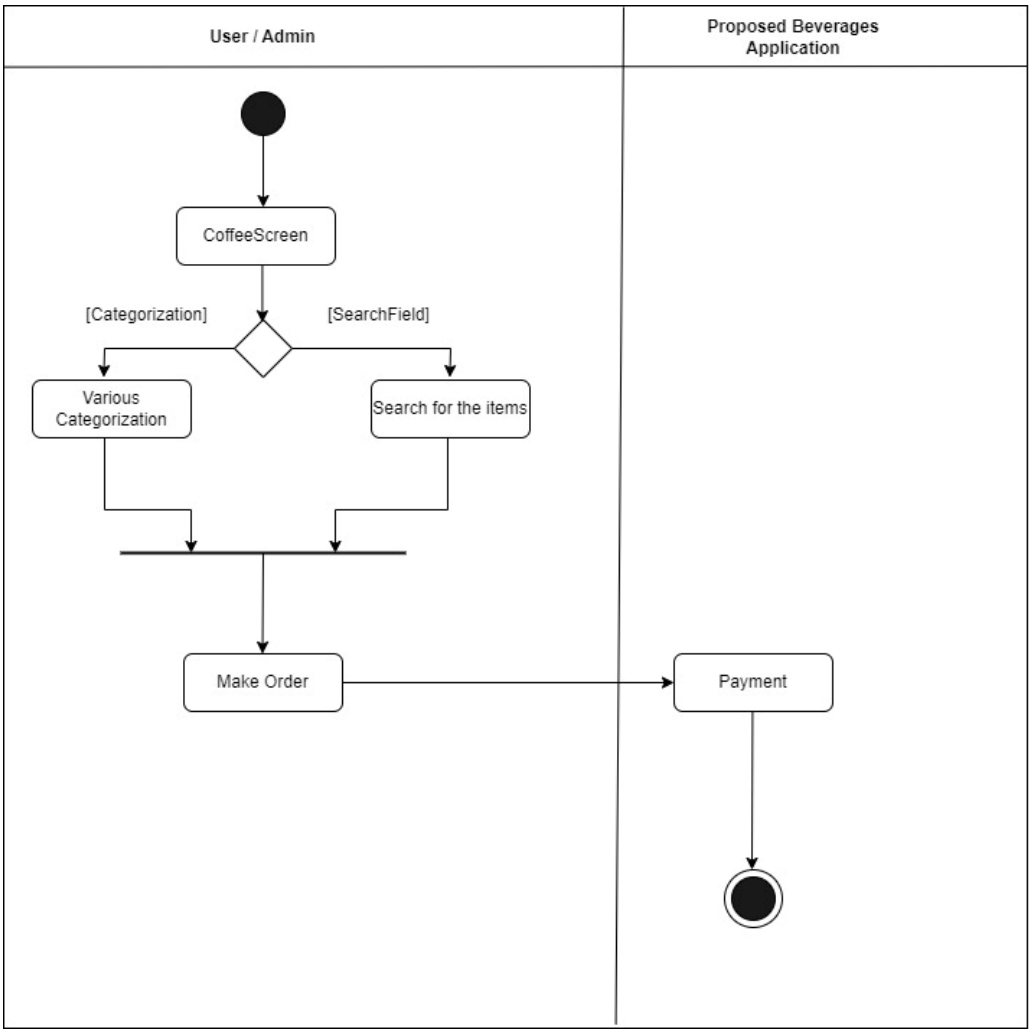

Figure 4.2.2.2 Activity Diagram for Categorization and Search Field

iv. Add to Favorite and View Use Case Description and Activity Diagram

Use Case Name	Add to favorite and view
Purpose	Allow user to add their favorite drinks and view favorite.
Actor	User
Normal Flows of Event	- User able add the drinks to favorite. - User able to view their favorite drinks.
Alternate Flows	- None

Table 4.2.2.3 Use Case Description of Add to Favorite and View

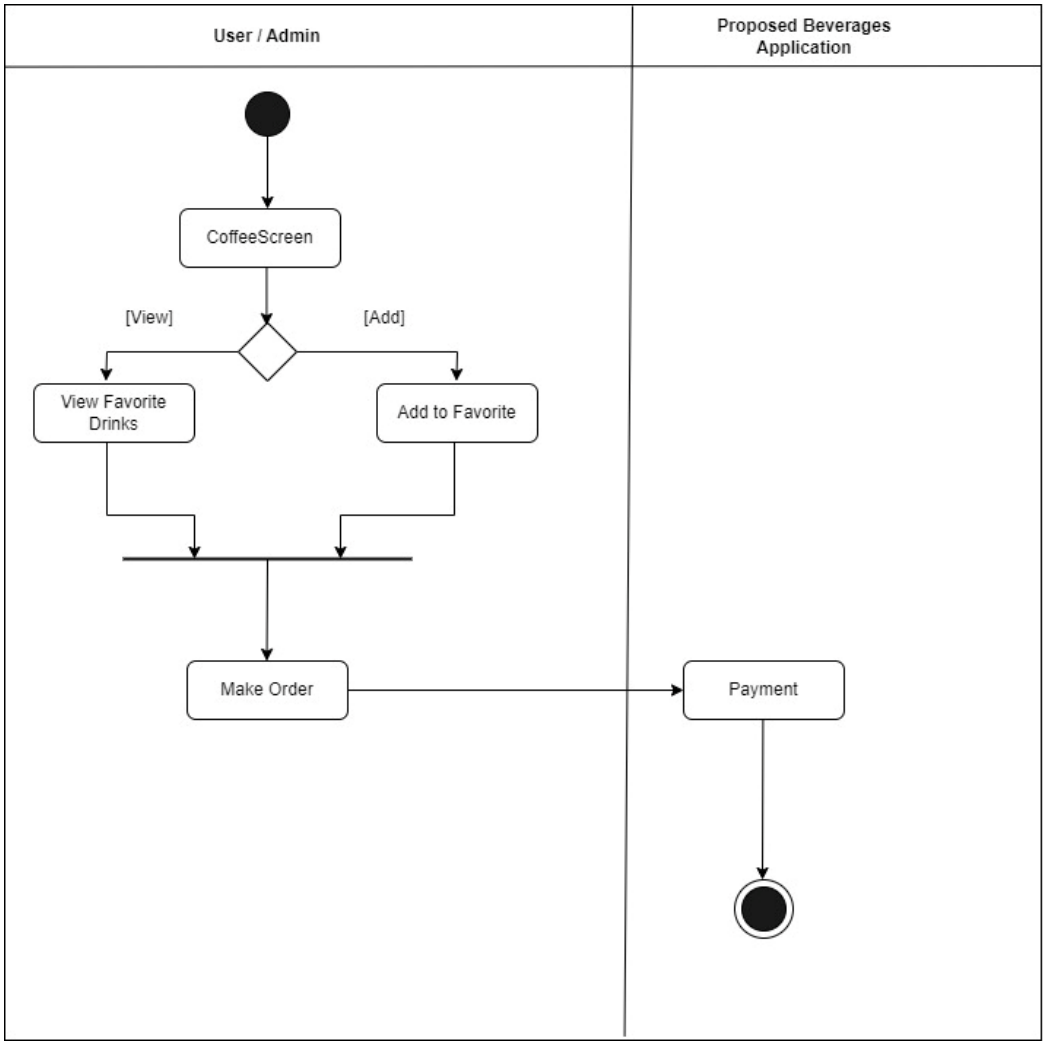

Figure 4.2.2.3 Activity Diagram of Add to Favorite and View

v. Order History Use Case Description and Activity Diagram

Use Case Name	Order History
Purpose	Allow user to view order history.
Actor	User
Normal Flows of Event	- User able to view order history.
Alternate Flows	- None

Table 4.2.2.4 Use Case Description of Order History

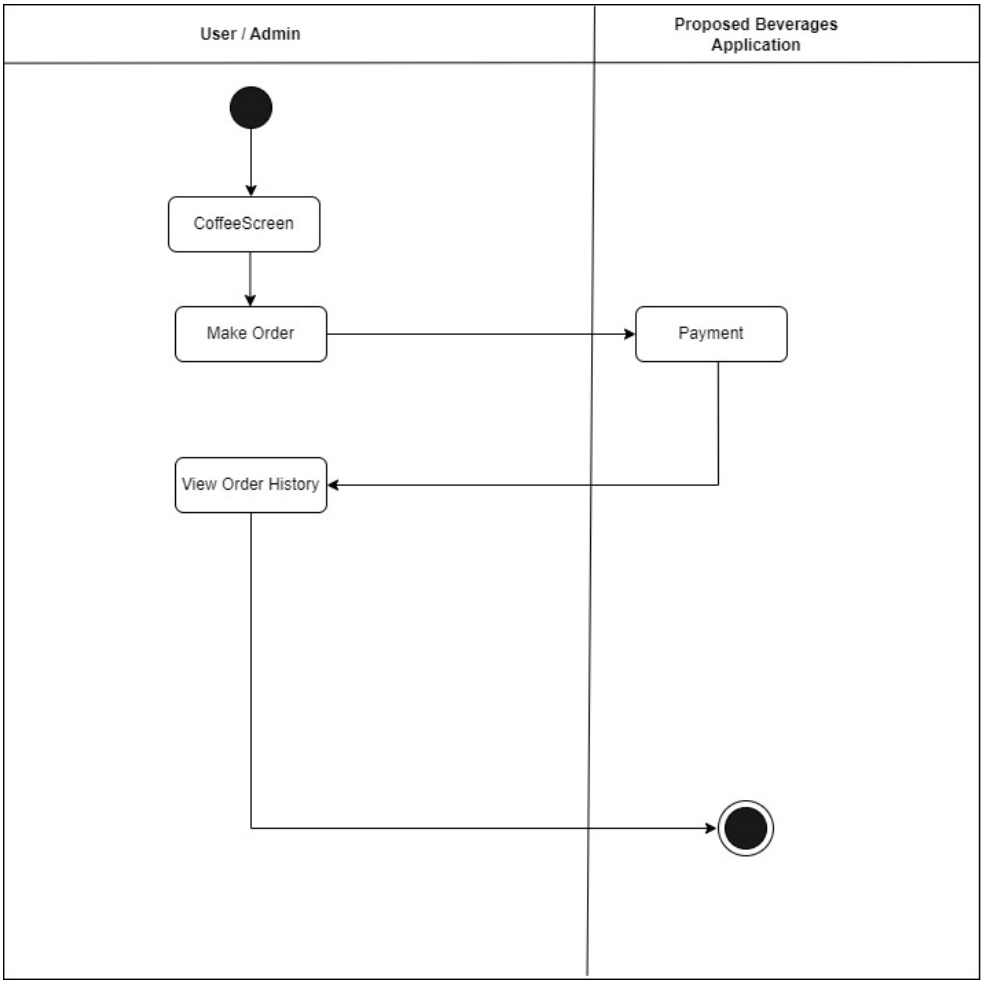

Figure 4.2.2.4 Activity Diagram of Order History

4.3 ERD Diagram

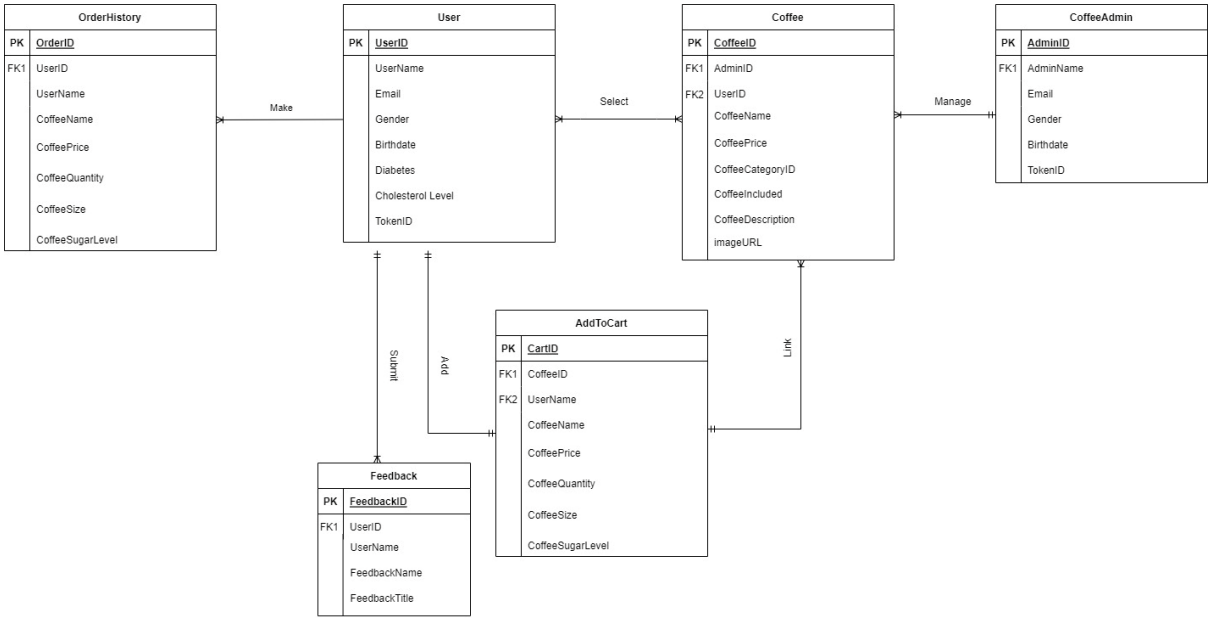

Figure 4.3 ERD Diagram of Proposed Beverages Application

Figure 4.3 shows the Proposed Beverages Application's ERD Diagram. ERD is a visual representation used to design databases and describe the relationships between entities within a system. It has six class in the diagram above such as OrderHistory, User, Coffee, CoffeeAdmin, AddToCart, Feedback. These classes data are stored in the Firebase retrieving in the proposed beverages application. First, User class is to store the user information such as UserID, UserName, Gender, BirthDate, Email, Cholesterol, Diabetes Level and Token ID identify user's authentication. Meanwhile CoffeeAdmin is used to store the admin of the beverage's application for user's authentication. Furthermore, the Coffee class is to act as an intermediately between admin and user where Coffee class stored required coffee information that insert by the admin and display in the proposed beverages application for users. The relationship between User, Coffee, CoffeeAdmin where one CoffeeAdmin can manage one or more coffee items and many Users can order one or more Coffee. Furthermore, AddToCart class stored the required information of the order user made and user can make payment from the information store in AddToCart. Moreover, User are allowed to view the OrderHistory that store in the OrderrHistory class with the required information. Feedback class is to store the Feedback submit from user for future improvements and applications updates.

Chapter 5 System Implementation

5.1 Hardware Setup

To ensure optimal performance and functionality of the proposed beverages application, certain hardware specifications are recommended for users' devices. The following are the minimum hardware requirements:

Operating System:

- For iOS development: macOS is required.
- For Android development: macOS, Linux, or Windows.

Computer Specifications:

- Mac: MacBook or iMac with at least 8GB of RAM.
- Windows: A PC with at least 8GB of RAM, ideally running Windows 10.
- Linux: Similar specifications as Windows machines.

Storage:

- Ensure you have sufficient free space for development tools, IDEs, and virtual devices/emulators.

5.2 Software Setup

1. React Native

React Native is a robust open-source framework utilized for developing cross-platform mobile applications for Android, iOS, and Windows platforms [17]. Popular applications such as Instagram, Netflix, and WhatsApp leverage React Native due to its extensive multi-platform support and efficient update capabilities using Code Push techniques.

2. Firebase

Firebase, developed by Google, provides a real-time database solution with secure client-side access for storing critical application data [18]. It enables the creation of collaborative apps by controlling data access based on user authentication. Firebase ensures data persistence locally and supports real-time events even when the user is offline, delivering a responsive user experience [19].

3. Node.js

Node.js is essential for implementing the event-driven, non-blocking I/O model necessary for this project [20]. It empowers the development of real-time websites with push functionality, enabling efficient communication initiation between clients and servers. Node.js simplifies

development and maintenance by allowing entire websites to operate within a single "stack," keeping the focus on project objectives [21].

4. Expo

Expo simplifies React Native development by providing pre-built UI and service components along with tools for creating and testing native React applications [22]. It eliminates the need to write native iOS or Android code, significantly reducing development time and effort.

5. Visual Studio Code

Visual Studio Code is chosen as the primary source-code editor for its lightweight, yet robust features tailored for developers [23]. It includes powerful tools like IntelliSense code completion, parameter hints, and graphical debugging, enhancing the development experience [24]. Visual Studio Code supports direct debugging and execution of React Native commands from the command palette, making it ideal for mobile app development with enhanced Node.js support [25].

5.3 Settings and Configuration

1. Node.js and npm:

Installation: Ensure Node.js and npm are installed on your machine.

2. React Native CLI:

Installation: Install the React Native command-line interface (CLI) globally using npm. Example, npm install -g react-native-cli.

3. Development IDE:

Visual Studio Code: Install Visual Studio Code or another preferred code editor. Visual Studio Code is recommended for its lightweight nature and powerful features for JavaScript development.

4. Create a New React Native Project:

Command: Use the React Native CLI to create a new project: Example, react-native init MyProject.

5. Start the React Native Metro Bundler:

Command: Navigate into your project directory and start the Metro Bundler (JavaScript bundler for React Native). Example, cd MyProject, npm start.

6. Run the App on Android or iOS:

Android: Open an Android emulator from Android Studio or connect a physical Android device, then run react-native run-android

iOS: Open the generated. xcworkspace file in the ios directory using Xcode, then run the app on an iOS simulator or a connected iOS device, react-native run-ios.

7. Additional Configuration:

Configure Libraries: If you're using additional libraries such as navigation, state management, follow the installation and setup instructions provided by each library.

Troubleshooting: Check the official React Native documentation and community forums for troubleshooting common issues and configuring advanced features.

Summary:

Setting up and configuring a React Native project involves installing necessary dependencies, setting up development tools, configuring Android and iOS environments, and creating a new project. Once configured, you can start developing mobile applications using React Native efficiently.

5.3.1 Database Configuration

1. Create Firebase Project:

Go to the Firebase Console and create a new project.

2. Set Up Firebase Database:

In the Firebase Console, navigate to "Firestore Database" or "Realtime Database" based on your database preference.

3. Install Firebase SDK:

In your React Native project directory, install the Firebase JavaScript SDK using npm or yarn, npm install firebase.

4. Initialize Firebase in Your Project:

Create a Firebase configuration file (e.g., firebase.js) in your project's directory, import firebase from 'firebase/app';

import 'firebase/database'; // Import only the Firebase services you need.

```
const firebaseConfig = {
  apiKey: 'YOUR_API_KEY',
  authDomain: 'YOUR_AUTH_DOMAIN',
  databaseURL: 'YOUR_DATABASE_URL',
  projectId: 'YOUR_PROJECT_ID',
  storageBucket: 'YOUR_STORAGE_BUCKET',
  messagingSenderId: 'YOUR_MESSAGING_SENDER_ID',
```

```
  appId: 'YOUR_APP_ID',
};
// Initialize Firebase
if (!firebase.apps.length) {
  firebase.initializeApp(firebaseConfig);
}
export default firebase;
```

5. Retrieve Firebase Configurations:

Get Firebase project configurations (apiKey, authDomain, databaseURL) from the Firebase Console under project settings.

6. Integrate Firebase Database:

Use the initialized Firebase instance (firebase) to interact with the Firebase database in your React Native components.

7. Database Operations:

Perform database operations (read, write, update, delete) using Firebase Realtime Database or Firestore as needed.

Summary:

By following these steps, you can configure Firebase for database usage in your React Native project. Initialize Firebase with your project configurations, import the necessary Firebase services, and integrate Firebase database operations into your components to interact with the database in real-time. Customize database paths, security rules, and data retrieval methods based on your project requirements using Firebase's comprehensive SDK and features.

5.4 System Operation

5.4.1 Authentication

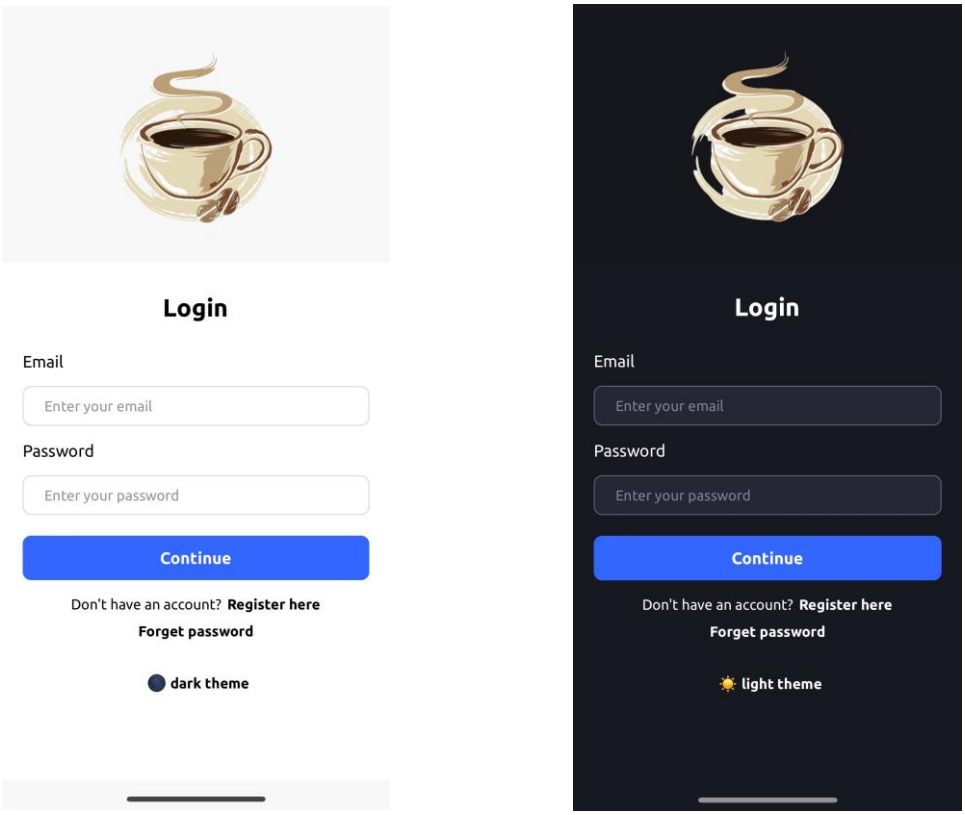

Figure 5.5.1 Screenshot of Sign In Page

In a React Native app with Firebase authentication, users can sign in their account by entering the registered email and password. Users can register by tapping "Register Here" to access the sign-up page. They provide required details like email and password, securely stored in Firebase for authentication. For password recovery, users click "Forgot Password" to access a reset screen, triggering a password reset email from Firebase to their registered address. Additionally, users can personalize their experience by toggling between light and dark themes, enhancing usability and visual comfort. These integrated features optimize user interaction, ensuring seamless registration, password management, and customizable themes within a streamlined mobile application experience.

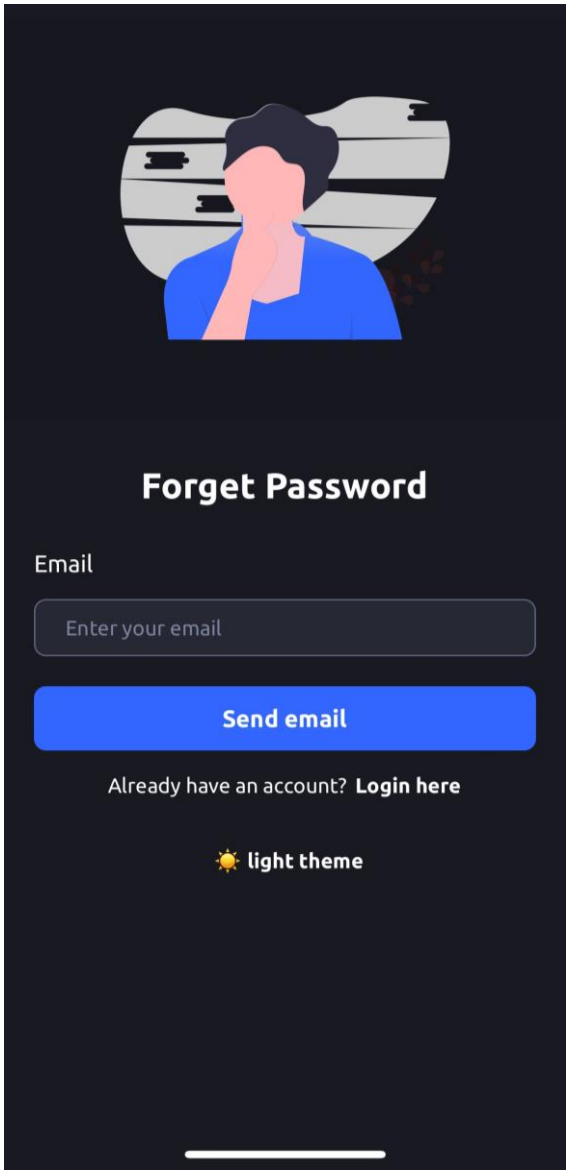

Figure 5.5.1.1 Screenshot of Forget Password Page

When a user redirects to the "Forgot Password" page, they are prompted to enter their email address. Upon submission, a password reset link is sent to the provided email address. This link allows the user to initiate the password reset process securely. The implementation leverages Firebase Authentication, which handles the password reset functionality by sending the reset link to the specified email. This user-friendly approach simplifies the password recovery process, ensuring that users can regain access to their accounts conveniently and securely.

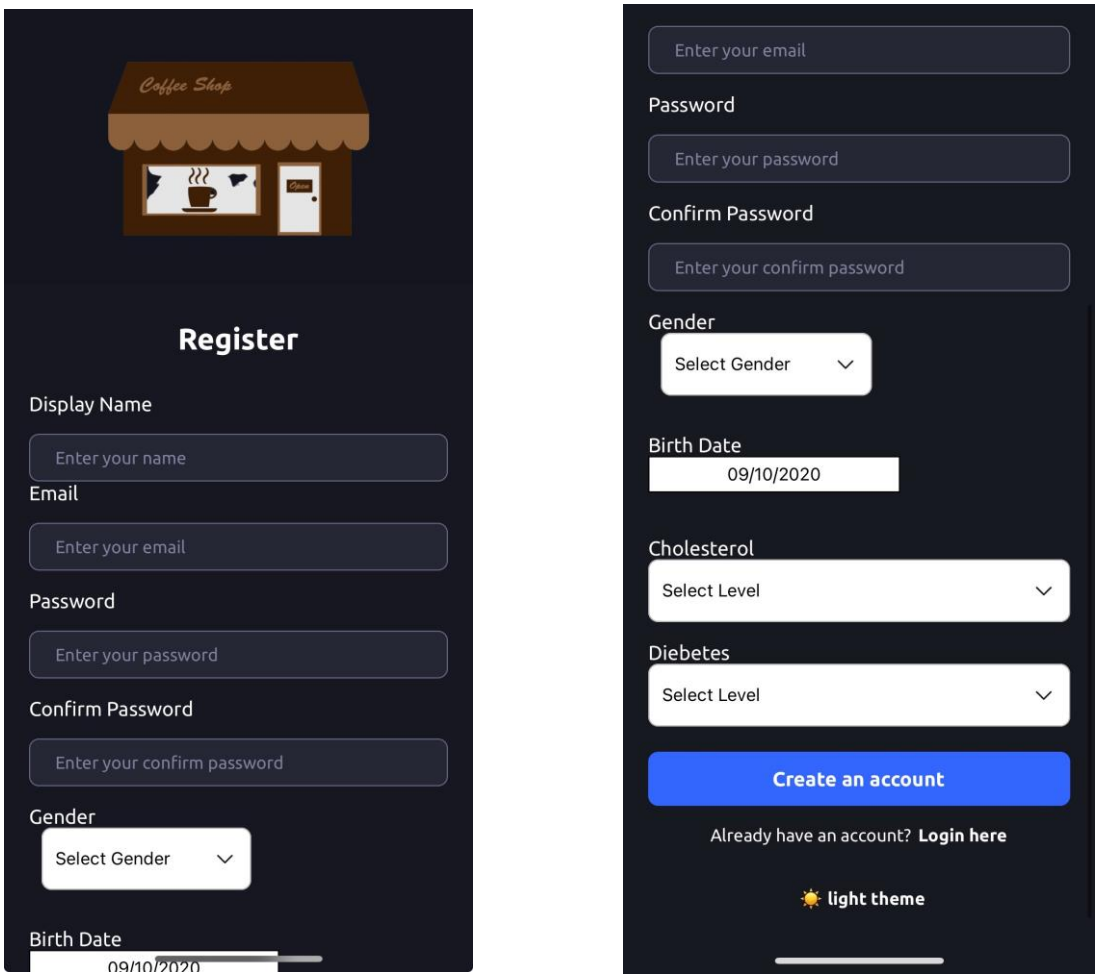

Figure 5.5.1.2 Screenshot of Register Page

Register page required user to fill up their Display Name, Email, Password, and Password reconfirmation, Gender, Birth Date and two of the most important questions that allow the personalized health management work which are Cholesterol and Diabetes level. User will be able to get the alert message if they are Cholesterol or Diabetes patient. Upon completion, if a user indicates Cholesterol or Diabetes conditions, the app will provide alerts accordingly.

After entering all required details, users can click "Create an Account" to securely save their information in Firebase for authentication. This process ensures that user data is stored and managed securely, facilitating personalized health management, and enabling seamless access to the app's features. The integration of Firebase Authentication ensures a smooth registration process while prioritizing user data privacy and security.

5.4.2 Pop-Up Modal

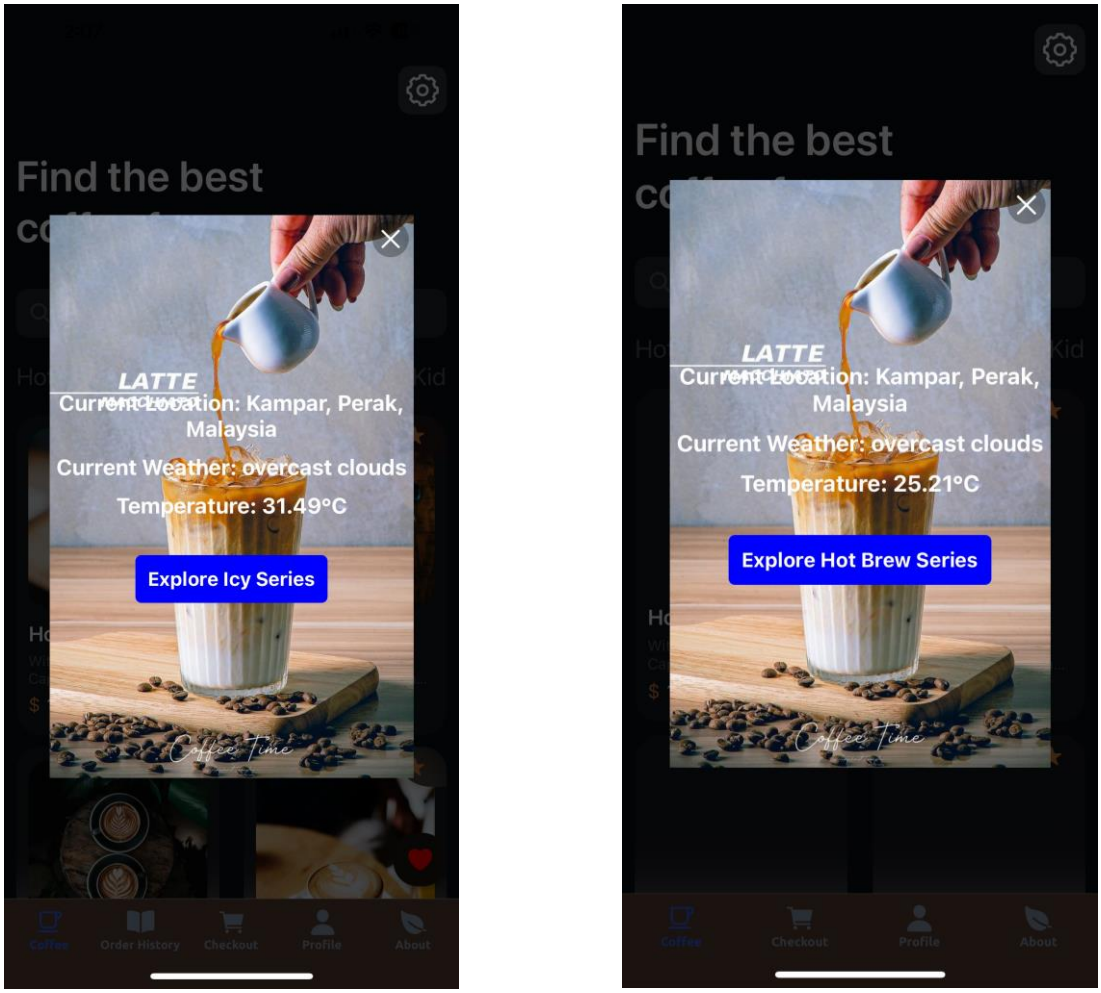

Figure 5.5.2 Screenshot of Pop-Up modal for Drinks Recommendation

Figure 5.5.2 shows a screenshot of a pop-up modal for drinks recommendation. The modal includes the user's current location, weather, and temperature, along with suggested drinks corresponding to the weather condition. If the temperature exceeds 30 degrees Celsius, the system recommends icy series drinks; otherwise, it suggests hot series drinks. The modal provides buttons for users redirect to the categories accordingly.

This feature aligns with the project objective of suggesting drinks based on weather conditions. Additionally, the pop-up modal is configured to display for only 5 seconds if the user does not make a selection, ensuring a concise and user-friendly experience. The interactive nature of the modal enhances user engagement and promotes personalized drink recommendations based on real-time weather data.

5.4.3 Coffee Screen/ Home Screen

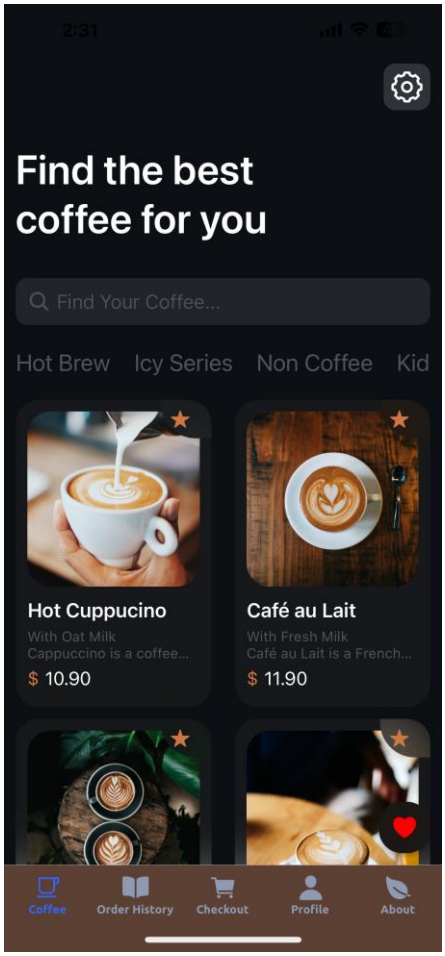

Figure 5.5.3 Screenshot of Home Screen

Figure 5.5.3 shows Screenshot of Home Screen of the proposed beverages application. At the top right corner, there is a user settings icon that redirects to the Settings Page. Below the settings icon, the screen displays the title of the application, followed by a search field and categories for easy navigation. Last is the display of the coffee items. Down right is the favourite icon where user can see the drinks, they add to favourite by clicking the heart shape icon at the down right. Moreover, the tab bar below of the application consists of the Coffee Page (Home Page), view Order History, Checkout page, Profile, and About Page.

5.4.4 Search Field

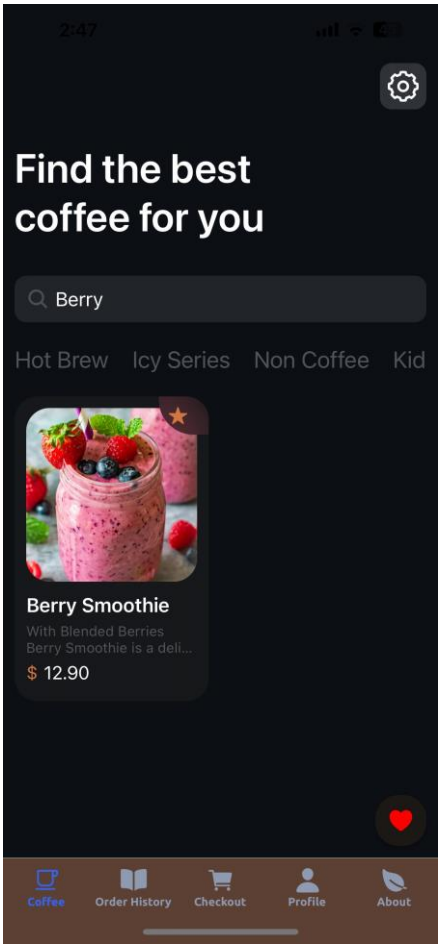

Figure 5.5.4 Screenshot of the Search Field

Figure 5.5.4 illustrates the search functionality of the proposed beverages application. Users can input desired products into the search field to find specific items within the application's Firebase. As the user types, the search results dynamically appear below the search field, displaying relevant products matching the entered keywords.

5.4.5 Categorization

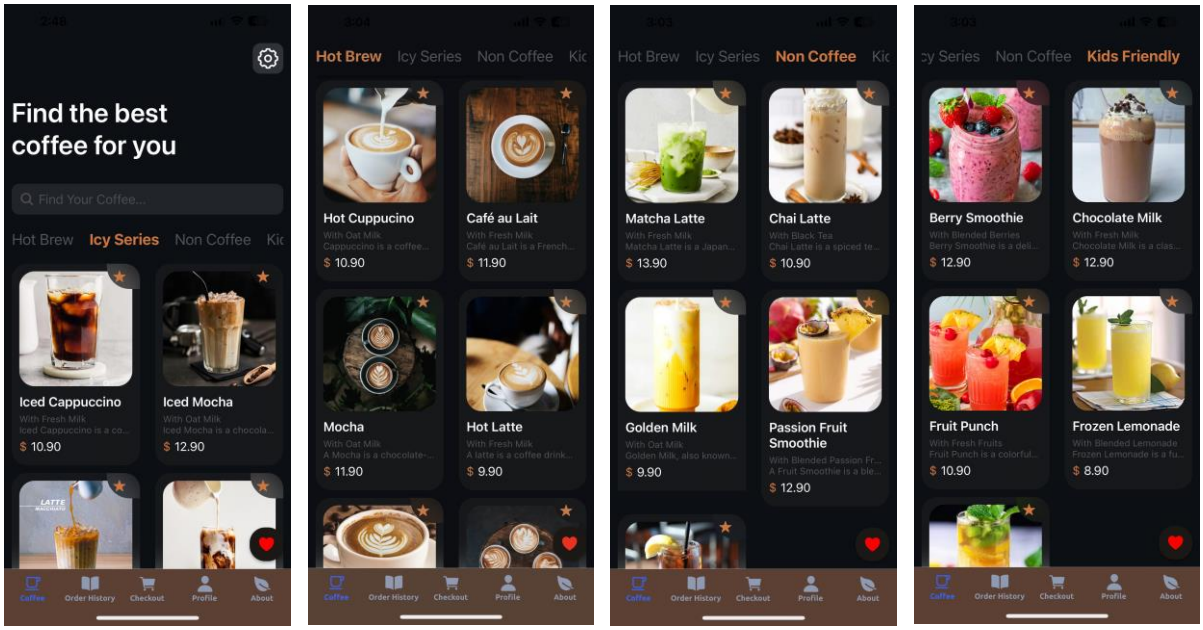

Figure 5.5.5 Screenshot of the Categorization

Figure 5.5.5 presents the categorization of the proposed beverages application, featuring four distinct categories:

Hot Brew: This category encompasses all varieties of hot brewed coffees, catering to enthusiasts of traditional coffee preparations.

Icy Series: Within this category, users can explore a selection of iced coffees, ideal for those seeking refreshing and chilled coffee options.

Non-Coffee: This category offers non-coffee choices tailored for adult consumers who prefer alternatives to coffee beverages.

Kids Friendly: Designed specifically for children, the Kids Friendly category features beverages suitable and appealing to younger audiences.

Each category drinks sharing common characteristics, ensuring users can easily navigate and discover beverages aligned with their preferences. This structured categorization enhances user experience by simplifying product exploration and facilitating informed choices based on desired beverage types.

5.4.6 Coffee Details Page

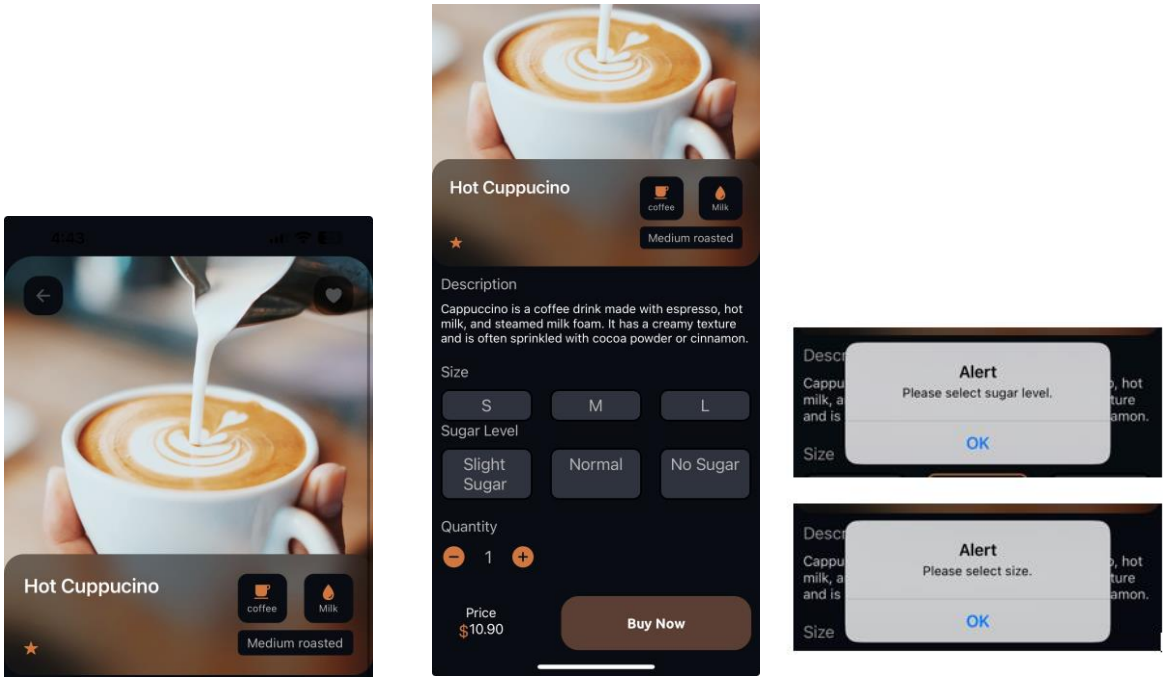

Figure 5.5.6 Screenshot of Coffee Details Page

Figure 5.5.6 showcases the Coffee Details Page within the proposed beverages application. Upon selecting a coffee item from the Home Screen, users are redirected to this page.

At the top right corner, a **Favourite** icon enables users to mark specific coffees as favourites by clicking the heart-shaped icon, which turns red upon selection.

Below, users view detailed coffee information and can proceed to purchase by clicking "Buy Now." Before adding to the cart, users must select size and sugar level; otherwise, a prompt reminds them to make selections first.

Additionally, users with diabetes or high cholesterol receive alerts suggesting suitable drink choices based on their health conditions. This comprehensive feature set enhances user engagement and ensures personalized drink selections aligned with individual preferences and health considerations.

5.4.7 Favorite Page

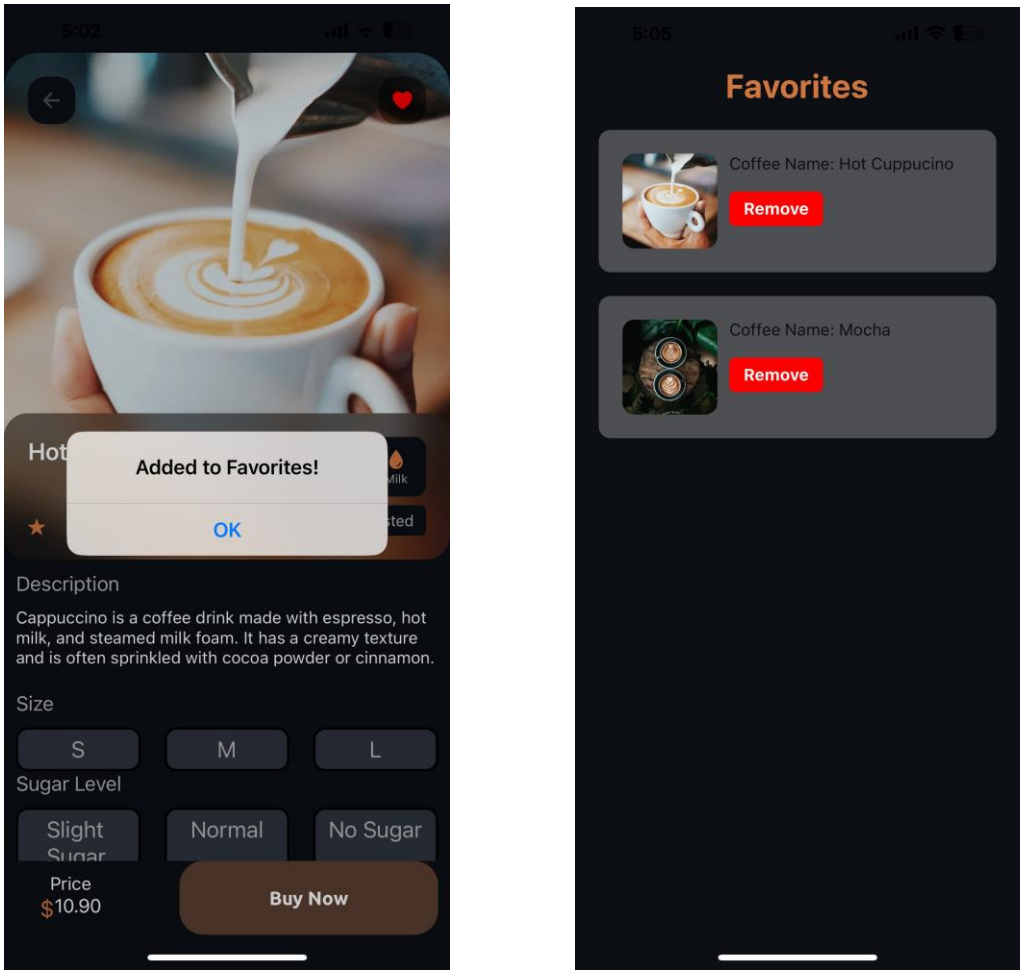

Figure 5.5.7 Screenshot of Favourite Page

Figure 5.5.7 shows the screenshot of Favourite Page within the proposed beverages application. The first screenshot shows when user click the favourite icon within the Coffee Details Page and the system will prompt users added to Favourites if the drinks is successfully saved in the Firebase of Favourite collection. The second screenshot shows the Favourite screen of the favourite drinks that user has added from the details page. User can easily remove the favourite drinks undo the click on the Favourite icon in Coffee Details Page or clicking the delete button to remove the selected favourite drinks.

5.4.8 Personalized Health Management

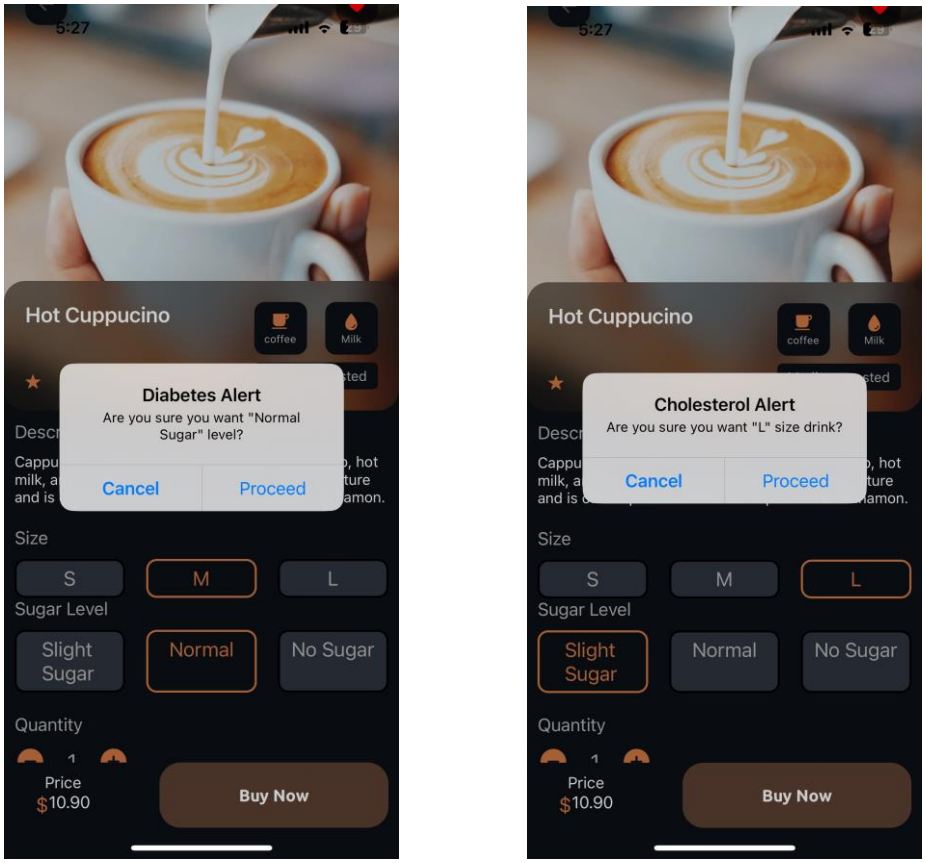

Figure 5.5.8 Screenshot of Personalized Health Management

Figure 5.5.8 illustrates the flow of Personalized Health Management within the proposed beverages application. After users provide personal information during registration, the system saves this data and utilizes the two health-related questions regarding Cholesterol Level and Diabetes Level.

For Diabetes users, the system alerts them when selecting Normal Sugar level drinks before adding to cart, suggesting alternative options better suited to their health. Users can choose to proceed with their selection or switch to the recommended option.

Similarly, Cholesterol patients receive alerts when selecting large cups of caffeinated drinks, as caffeine may impact cholesterol levels according to research highlighted earlier. These alerts prioritize user health by providing tailored recommendations based on individual health concerns, enhancing user safety and wellness considerations during beverage selection.

5.4.9 Order History

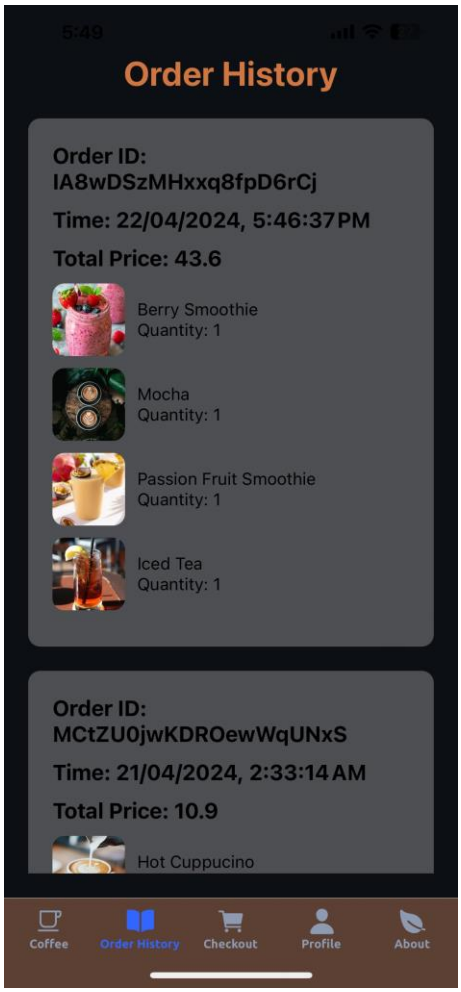

Figure 5.5.9 Screenshot of Order History

Figure 5.5.9 shows the screenshot of the order history page. In this page, user able to view their order history per each transaction. In each transaction, user will be able to view the order ID, order time, and the total price of each of the transaction. Furthermore, user will be able to view the coffee items, including the image url of what user ordered. Each of the coffee items include coffee title, and coffee quantity.

5.4.10 Checkout Screen

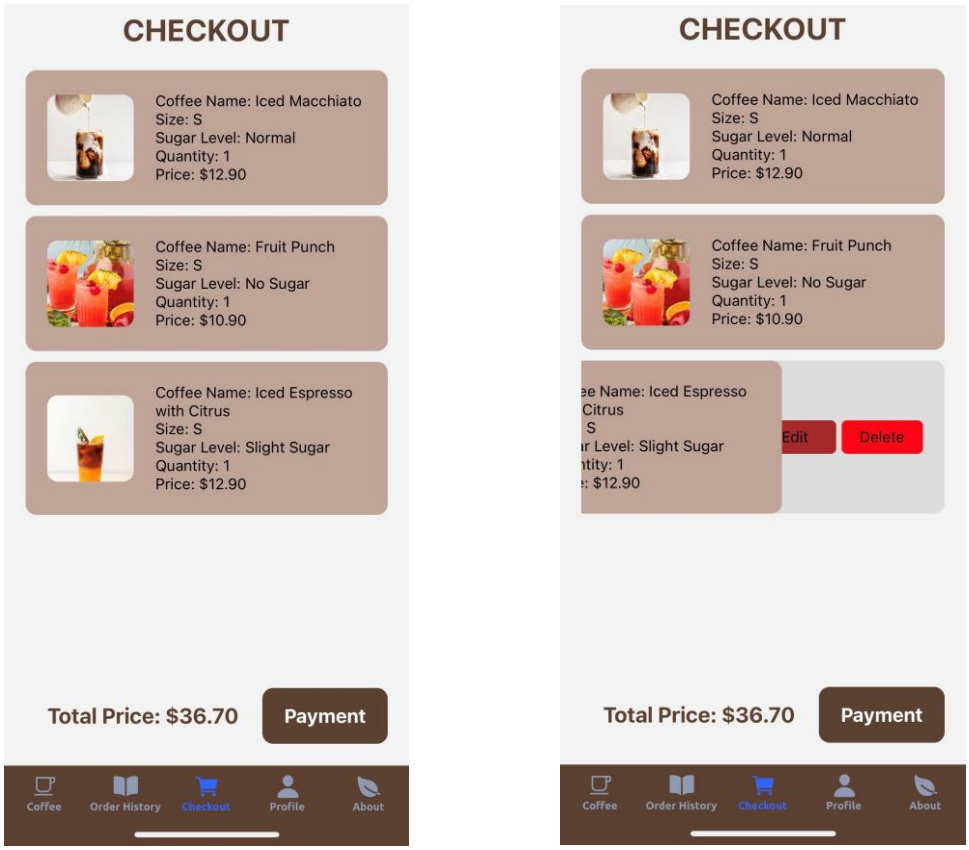

Figure 5.5.10 Screenshot of Checkout Screen

Figure 5.5.10 shows the screenshot of the checkout screen within the proposed beverages application. When user click the "Buy Now" button within the Coffee Details Screen, the coffee will save in firebase and display in the Checkout screen, user can view their added to cart button for the Checkout at the tab bar of the proposed application. In the checkout screen, the page come with the coffee items and the total accumulated price of the coffee items listed in the checkout screen. However, if the user adds to cart the same drinks and same size and sugar level separately, the system will help user sums up the quantity of that drinks with the variety in the checkout screen. Furthermore, the checkout screen using the swipe list view for user to swipe to left and edit or delete of the coffee items. After user has make changes of the product or coffee items in the checkout screen, the system will update in the Firebase accordingly. In addition, the checkout screen will be empty after user has successfully make the payment.

5.4.11 Payment

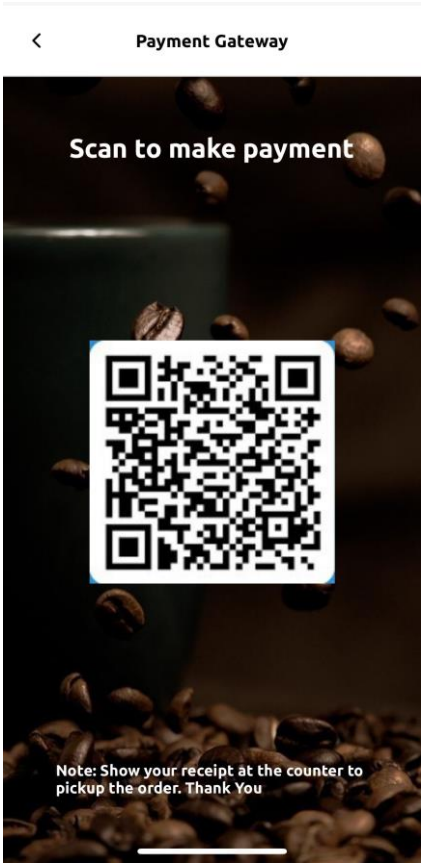

Figure 5.5.11 Screenshot of Payment Gateway

Figure 5.5.11 shows the screenshot of the payment gateway of the proposed beverages application. User is required to scan the provided QR code in the Touch n Go application to make payment. Furthermore, after user has successfully make payment, user need to show the payment receipt at the counter to pick up order according to the order id showed in the Order History page. However, due to the restrictions and the confidentiality of the data, the proposed beverages application decided to use the show QR payment method in the payment gateway to keep user data secret and confidential.

5.4.12 Profile

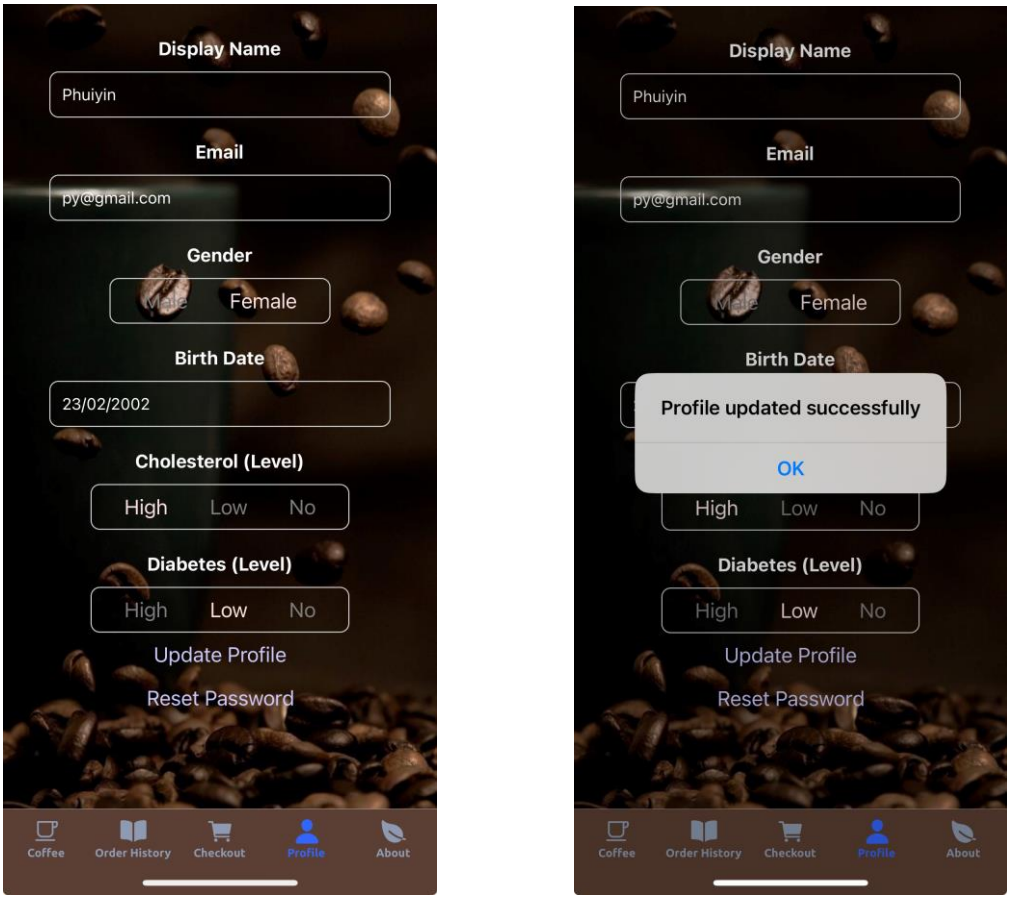

Figure 5.5.12 Screenshot of Profile Page

Figure 5.5.12 shows the screenshot of profile page within the tab bar of the proposed beverages application. User can update their personal information time to time based on their preference. The profile page allows users to update their display name, gender, birthdate, cholesterol, and diabetes level. Users are not allowed to edit their email address where the email address is the authentication of one and cannot be simply change. After user has fill up the information, user can click on the update profile and the system will prompt users that the profile updated successfully. Furthermore, user can reset their password by clicking on the reset password and system will redirect users to the Reset Password page.

5.4.13 Feedback Page

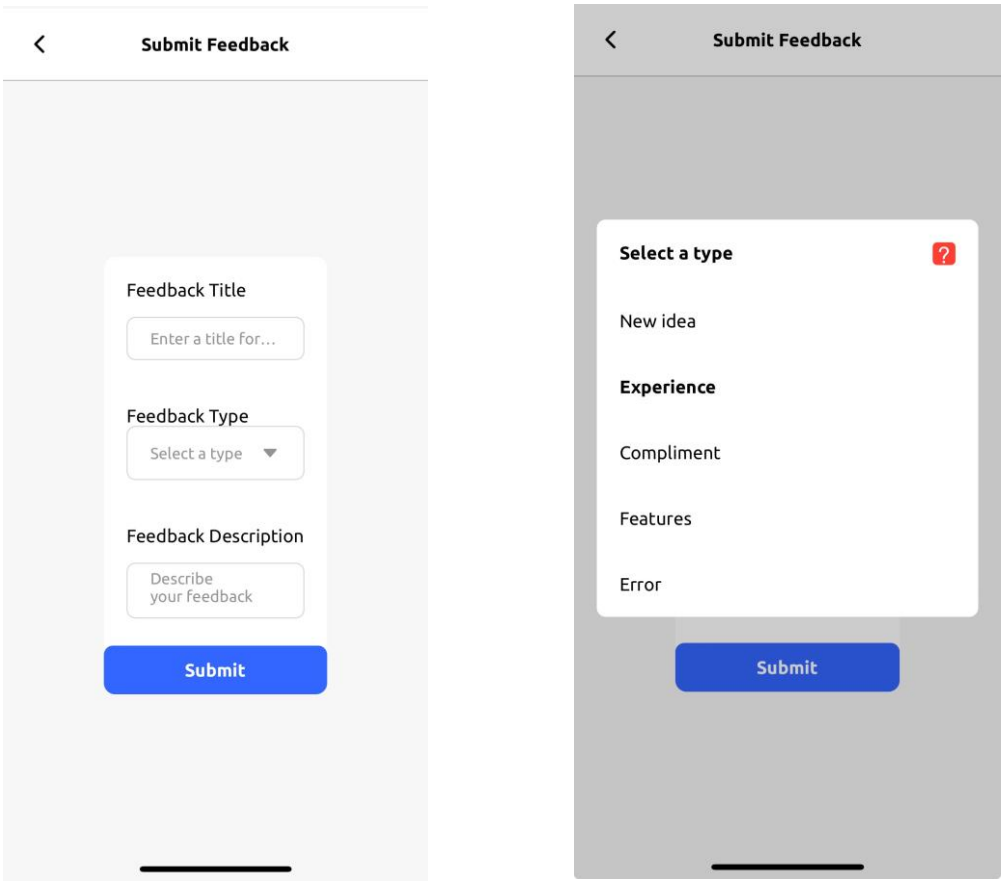

Figure 5.5.13 Screenshot of Feedback Page

Figure 5.5.13 shows the screenshot of Feedback Page within the proposed beverages application. When the user clicks on the Feedback button in Settings Page, the system will redirect users to the Feedback Page. The first screenshot shows the Feedback screen where it allows users to submit the feedback according to the Feedback title, Feedback Type, and Feedback Description. The second screenshot shows the Feedback Type provided for user to select the feedback falls under which categories. Additionally, after user click on the submit button, the system will prompt users that feedback successfully submits.

5.4.14 Settings

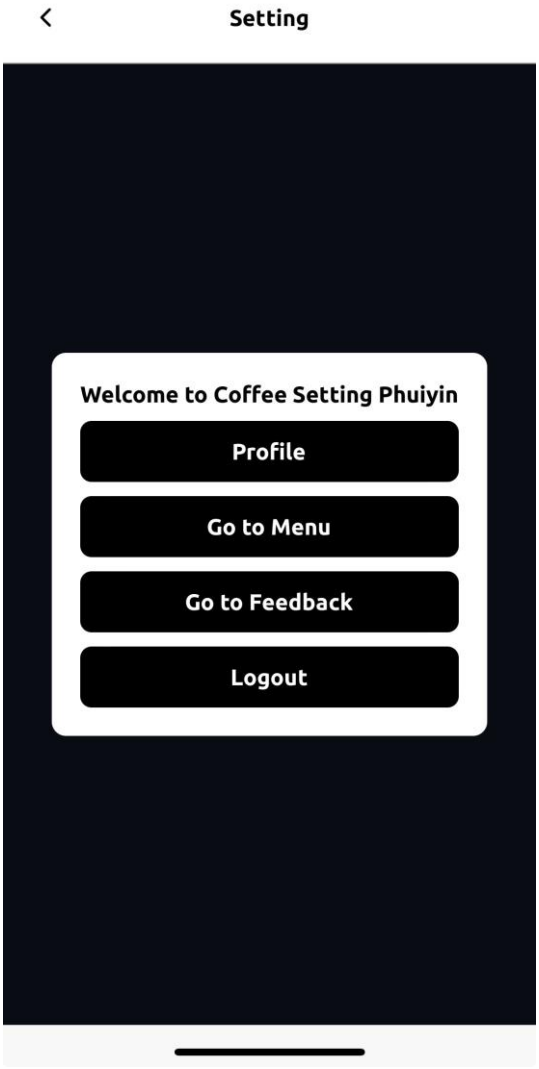

Figure 5.5.14 Screenshot of Settings Page

Figure 5.5.14 presents the Settings Page within the proposed application. The page dynamically retrieves the user's display name from Firebase, welcoming them with a personalized message like "Welcome to Coffee Settings, [User's Display Name]." This feature enhances user experience and adds a professional touch to the interface. Additionally, users can navigate to different sections by clicking buttons on the Settings Page:

Redirect to Profile: Allows users to view and edit their profile information.

Back to Menu: Returns users to the main menu or previous screen.

Go to Feedback: Directs users to the Feedback Page to submit suggestions or comments.

Logout: Enables users to log out of their account securely.

5.4.15 About Page

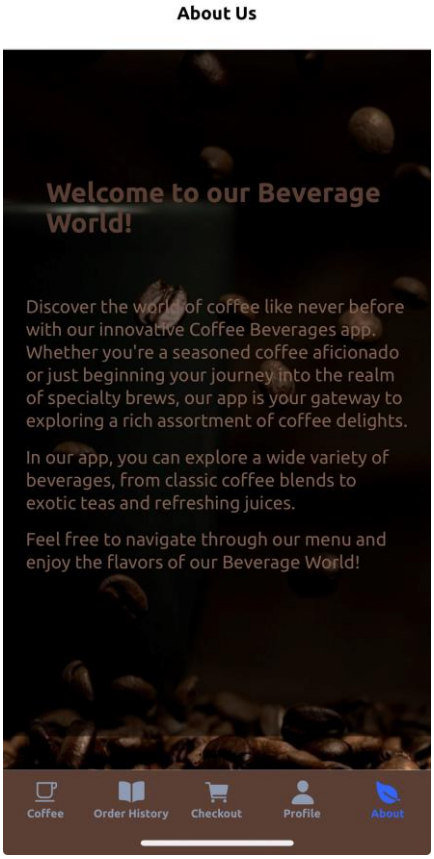

Figure 5.5.15 Screenshot of About Page

Figure 5.5.15 shows the last element in the tab bar of the proposed beverages application such as About Page. User able to view the information of the proposed beverages application in the About page.

5.4.16 Confidentiality

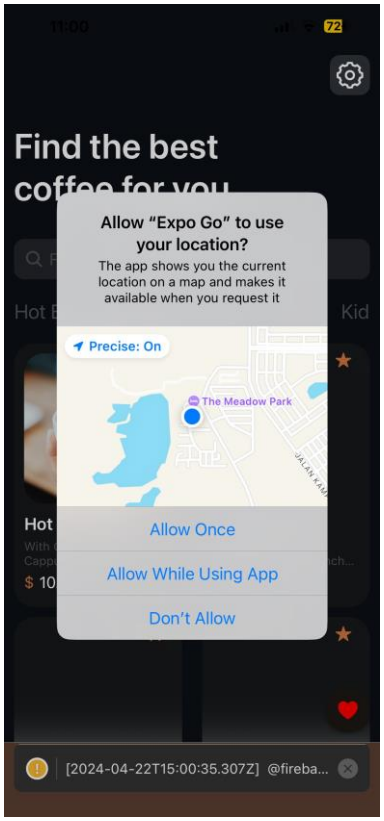

Figure 5.5.16 Screenshot of Prompt Message to Grant User Access

Figure 5.5.16 depicts a screenshot of a prompt message requesting user permission to grant access within the application. Following user login, the system requests access to location and weather services. Once granted by the user, they can then view the pop-up modal, likely displaying relevant information based on their location and current weather conditions and recommending drinks to the user.

5.4.17 Admin Site

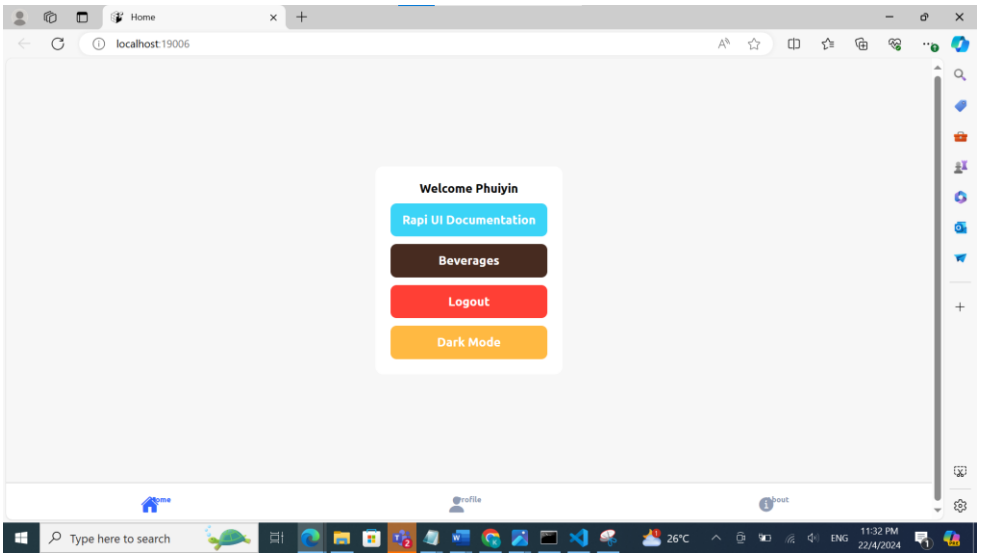

Figure 5.5.17 Screenshot of Admin Home Page

Figure 5.5.17 shows the screenshot of Admin Home Page. This page is the home screen page after has log in to the project. The login, register, and forget password page shared the same design with the proposed beverages screen. After admin user log in to the Home Page, and user can click on the Beverages button redirect them to the Beverages functionality page.

5.4.18 Beverages Screen

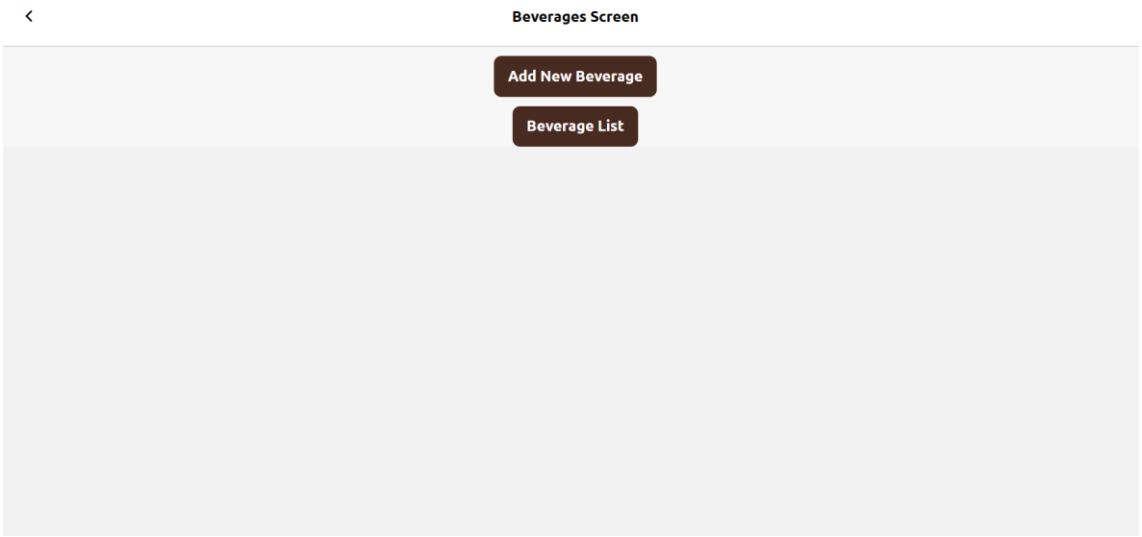

Figure 5.5.18 Screenshot of Beverages Screen

In the admin site of the Beverages Screen, users are allowed to add new beverages and view the beverages list by clicking the provided button.

5.4.19 Add New Beverages

Figure 5.5.19 Screenshot of Add New Beverages Page

Figure 5.5.19 shows the screenshot of Add New Beverages page. Admin required to update the Coffee Items image, Coffee ID, Coffee Price, Description, Coffee Category and Coffee Included. The user required to input the Coffee Category by numeric number of 1,2,3, or 4. Each of the numeric number represent different category in the user screen. After user click the Add button, the system will prompt user that Beverages Added Successfully.

5.4.20 Beverages List

Figure 5.5.20 Screenshot of Beverages List

Figure 5.5.20 shows the screenshot of beverages list that has entered by the admin. The admin user able to view the added beverages list including the Coffee ID, Admin (added by who), Date and Time added, and the Coffee Name and Image. Furthermore, admin user able to Edit and Delete the added items by swiping the list to left.

5.4.21 Beverages Detail

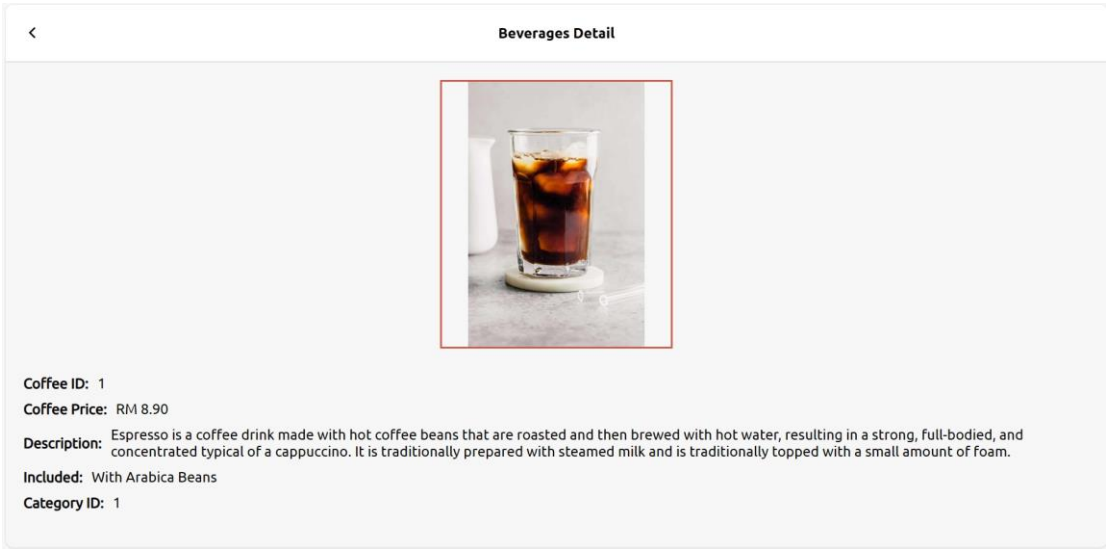

Figure 5.5.21 Screenshot of Beverages Details

Figure 5.5.21 shows the screenshot of beverages details that allow admin user to view the details of the added beverages. In the Beverages Details screen, users are allowed to view all the details that has entered during the add beverages list.

5.4.22 Edit Added Beverages

Figure 5.5.22 Screenshot of the Edit Added Beverages Page

Figure 5.5.22 shows the screenshot of the edit added beverages page. Admin user can edit the added beverages items by clicking the edit from the list. User able to update Coffee details such as Coffee ID, Coffee Name, Coffee Price, Description, Coffee Category, Coffee Included.

5.4.23 Update Profile

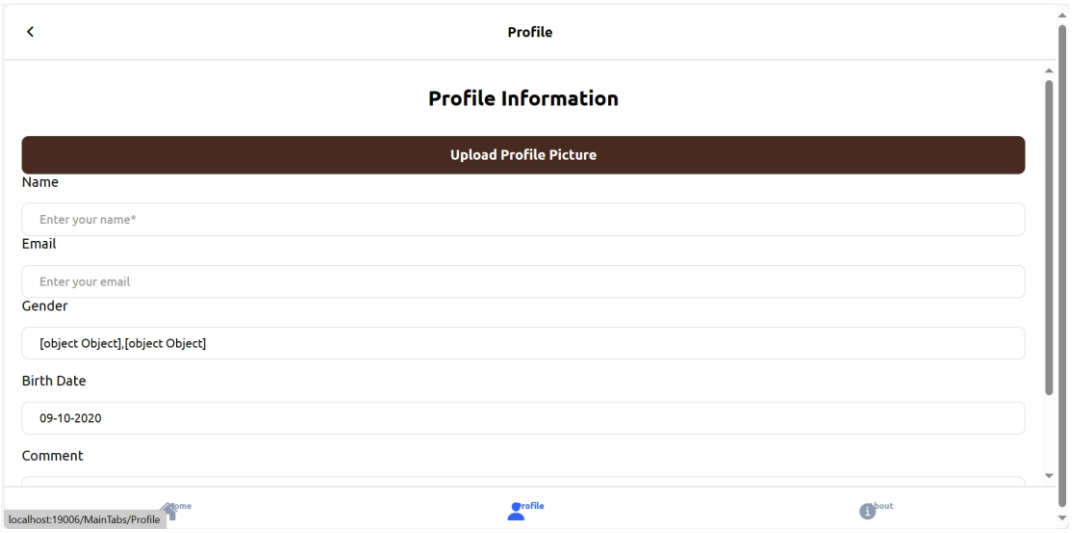

Figure 5.5.23 Screenshot of Update Profile

Figure 5.5.23 shows the admin user update profile page. The profile page allows admin users to update their name, gender, birthdate, and comment. After user has fill up the information, user can click on the update profile and the system will prompt users that the profile updated successfully.

5.4.24 About Screen

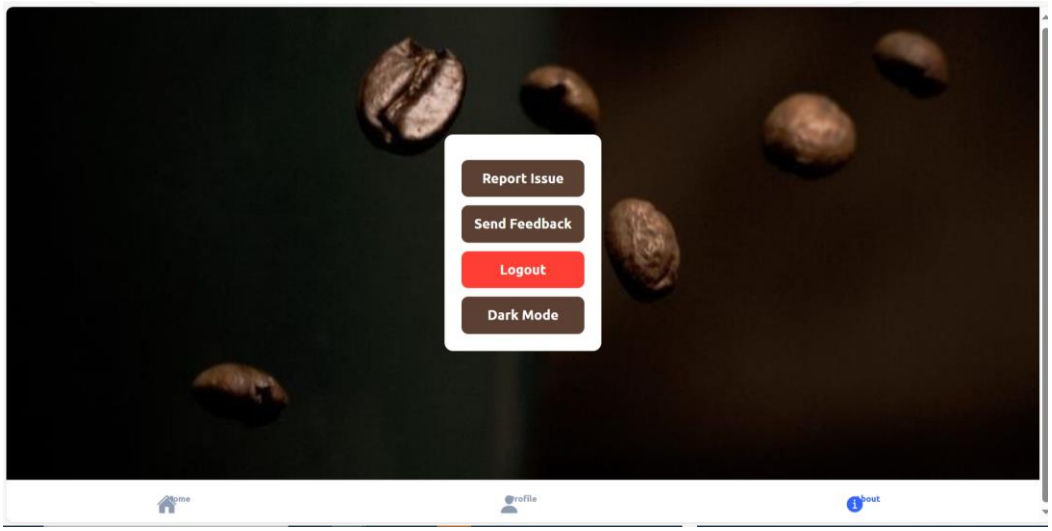

Figure 5.5.24 Screenshot of About Screen

Figure 5.5.24 shows the about screen that allows the admin user to report issues, send feedback, and log out. The dark mode button allow user to switch in light or dark mode based on their user preference.

5.4.25 Report Issues

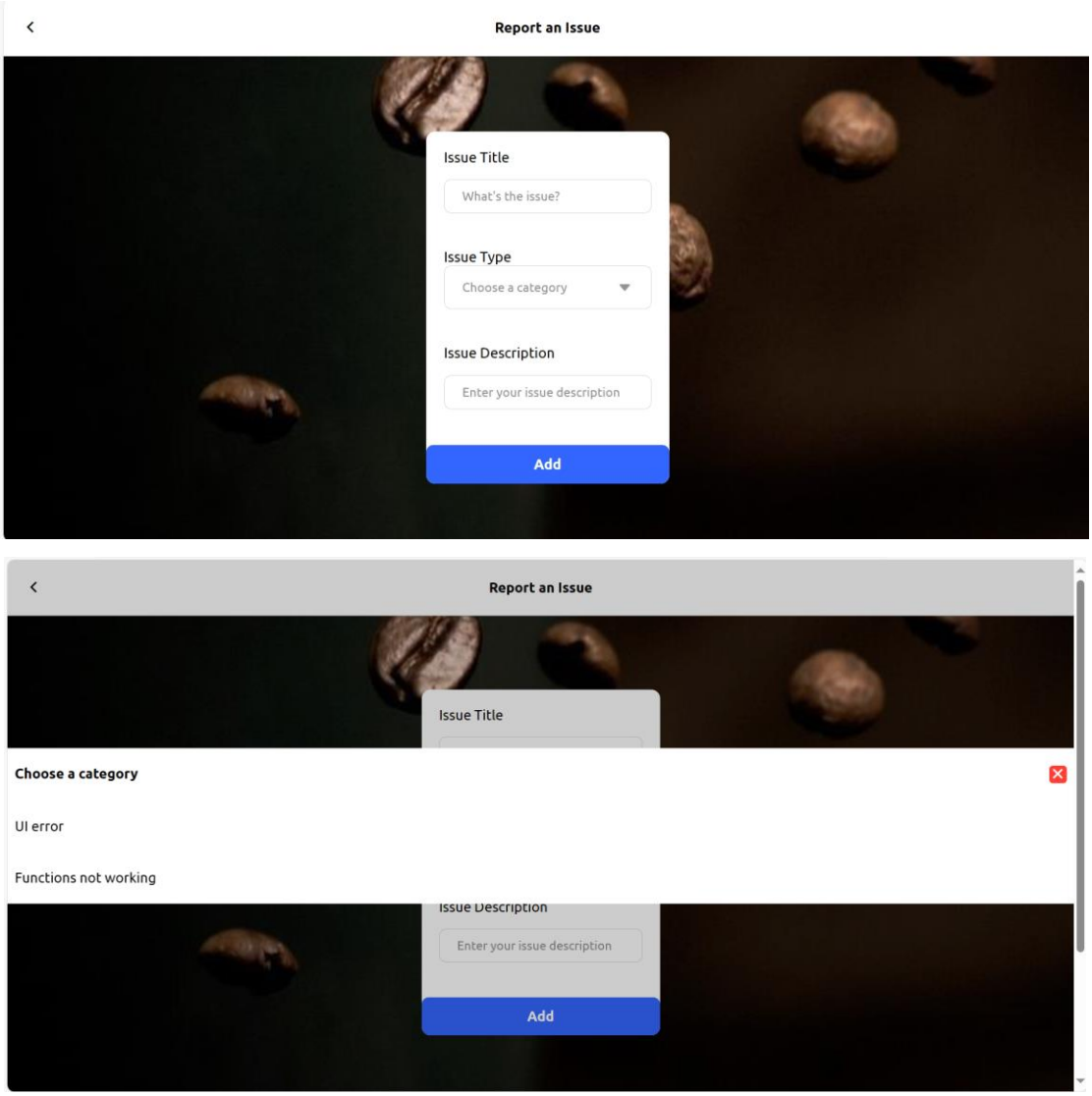

Figure 5.5.25 Screenshot of Report Screen

Figure 5.5.25 displays a screenshot of the report screen in the web system, allowing administrators to report issues categorized appropriately. This feature enables admins to identify and document various types of issues within the system, facilitating efficient management and resolution.

5.4.26 Feedback Page

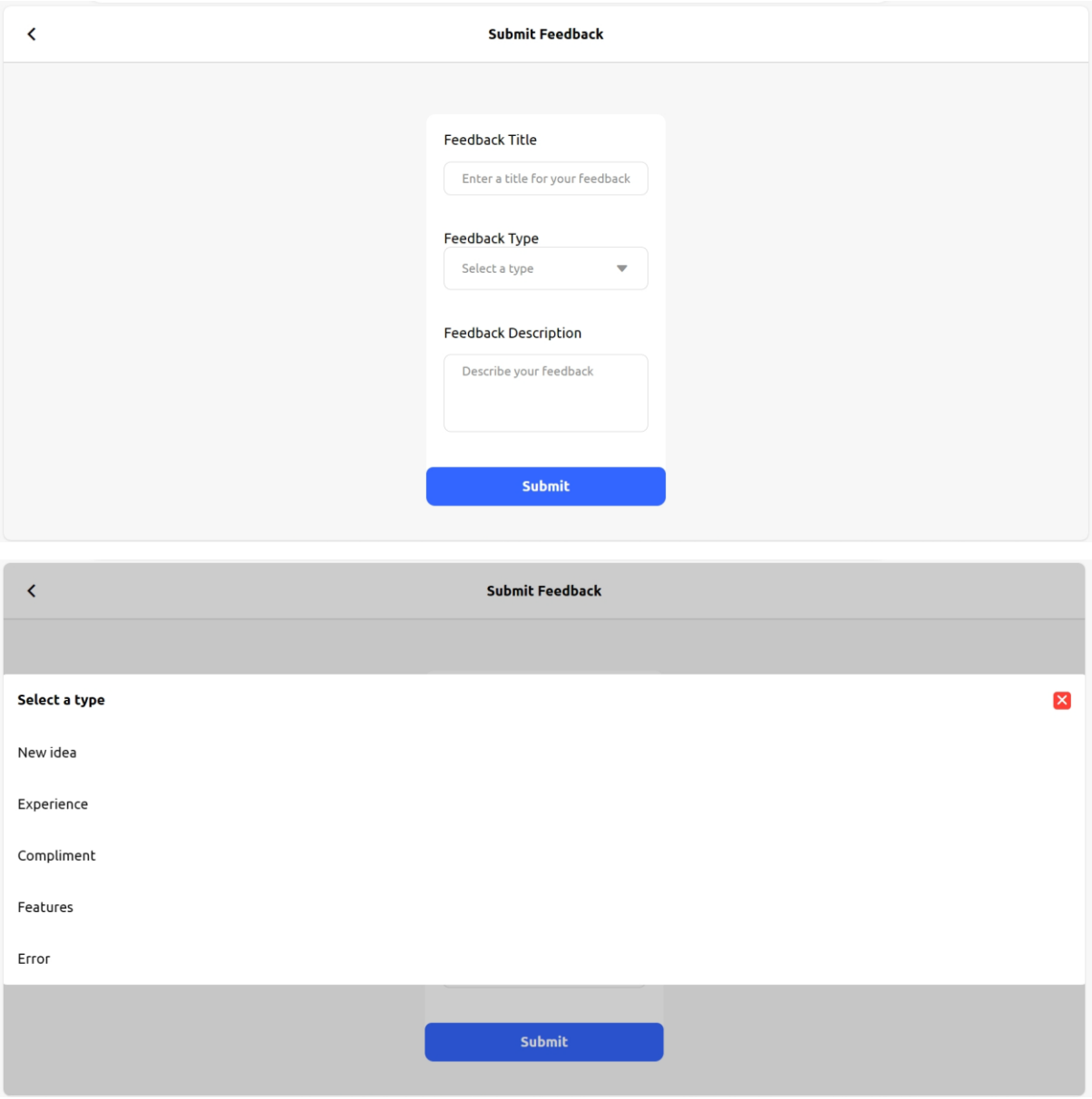

Figure 5.5.26 Screenshot of Submit Feedback

Figure 5.5.26 shows screenshot of submit feedback page. The first screenshot shows the Feedback screen where it allows admin users to submit the feedback according to the Feedback title, Feedback Type, and Feedback Description. The second screenshot shows the Feedback Type provided for user to select the feedback falls under which categories. Additionally, after user click on the submit button, the system will prompt users that feedback successfully submits.

5.5 Implementation Issue and Challenges

1. Incompatible React Native Libraries:

One of the primary challenges has been dealing with incompatible React Native libraries. This issue has required us to carefully select and integrate libraries that align with our project requirements to ensure stability and functionality.

2. Incompatible Version Library:

Similar to the above, managing incompatible library versions has been a recurring challenge. We've had to meticulously review and update library versions to maintain compatibility and mitigate potential conflicts.

3. React Native Set Up:

Setting up React Native environments can be intricate, and we've faced specific challenges in configuring our development environment to ensure seamless integration and performance.

4. Fast Updates and Compatibility:

React Native's rapid development pace has occasionally led to compatibility issues. We're proactive in monitoring updates and adjusting our project's React Native version to align with the latest stable releases, ensuring ongoing compatibility and support.

Chapter 6 System Evaluation and Discussion

6.1 Block Box Testing

In this section, we will evaluate the system's functionalities through block box testing, which includes a series of test scenarios that focus on the system's output based on different inputs. The block box testing is divided into two main categories: testing the user site functionalities and the admin site functionalities.

6.1.1 User Site Functionality Testing

This subsection focuses on the testing scenarios pertaining to the user site functionalities.

6.1.1.1 Test Scenario: Check Account Login Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Verify email and password after click on the login button in login page with valid email and password.	Email: pyinkong@gmail.com Password: 00000000	User is successfully logged in and directed to the Home Screen page.	Same as expected result
2	Attempt to login with invalid email or password.	Email: pyinkong@gmail.com Password: 00001111	Display an error message: “Invalid Email or Password.”	Same as expected result
3	Attempt to login without entering a username and password.	Email: NULL Password: NULL	Display an error message: “User Not Found.”	Same as expected result

Table 6.1.1.1 Check Account Login Functionality on User Site

6.1.1.2 Test Scenario: Check Account Registration Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Verify all the information enters by the users after click on the register button with all valid input.	Enter Valid User Details: Display Name, Email, Password, Confirm Password, Gender,	User account is Successfully Registered, and the user is redirected to the intended page.	Same as expected result

		Birthdate, Cholesterol and Diabetes Level.		
2	Attempt to register with an email that is already registered.	Enter an email that is already associated with an existing account.	Display an error message: "Email Already in Use."	Same as expected result
3	Attempt to register with an invalid email format.	Enter an email with an incorrect format (e.g., missing '@','.com' symbol).	Display an error message: "Invalid Email."	Same as expected result
4	Attempt to register with missing fields.	Leave one or more required fields (e.g., Display Name, Email, Gender, Birthdate, and Password) blank	Display an error message: "Email(missing) is Required."	Same as expected result
5	Attempt to register with valid information and navigate away from the registration page before submitting.	Enter valid user details but leave the registration page without submitting.	Registration failed, and no account is registered.	Same as expected result

Table 6.1.1.2 Check Account Registration Functionality on User Site

6.1.1.3 Test Scenario: Check Forgot Password Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to enter the registered email to send the reset password link to the authentic email.	Send by registered email.	Users receive the reset password link in the registered email.	Same as expected result
2	Attempt to reset password using the reset password link.	Reset Password Email: pyinkong@gmail.com	Received the reset password link and	Same as expected result

			redirect to the reset process.	
3	Attempt to enter invalid email	Email: NULL	Do not received the reset password link.	Same as expected result

Table 6.1.1.3 Check Forgot Password Functionality on User Site

6.1.1.4 Test Scenario: Check Categorization and Search Field in Home Screen Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to select the coffee categories.	4 Categories: Hot Brew, Icy Series, Non- Coffee, Kids Friendly	Users can click on the categories name to explore the same coffee categories.	Same as expected result
2	Attempt to search for the desired coffee or drinks.	Enter valid coffee items name.	Shows user the coffee items by their input data.	Same as expected result

Table 6.1.1.4 Check Categorization and Search Field in Home Screen Functionality on User Site

6.1.1.5 Test Scenario: Coffee Details Screen Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to select drinks from Home Screen and redirect to the coffee details screen.	Click on the interested drinks.	Navigate user to the details screen of selected drinks.	Same as expected result
2	Attempt to make order by selecting the sugar level and size.	Sugar Level: Normal Size: L	Display an message: “Added to Cart Successfully.”	Same as expected result

3	Attempt to make order without selecting the sugar level.	Sugare Level: Null	Display an alert message: "Please select sugar level."	Same as expected result
4	Attempt to make order without selecting the size.	Size: Null	Display an alert message: "Please select size."	Same as expected result
5	Attempt to add to cart of the drinks.	Click the Buy Now button in the Coffee Details Screen.	Prompt a pop-up modal by listing down the selected beverage's details for confirmation.	Same as expected result

Table 6.1.1.5 Coffee Details Screen Functionality on User Site

6.1.1.6 Test Scenario: Checkout Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to Add to Cart.	Users click checkout confirmation of pop-up modal in the Coffee Detail Page.	Add the drinks into the Checkout page.	Same as expected result
2	Attempt to edit the checkout screen item.	Scroll to left and click Edit.	Redirect users to the Edit Screen and update the changes.	Same as expected result
3	Attempt to Delete the added item in Checkout screen.	Scroll to left and click Delete.	Prompt Users: "Deleted Successfully"	Same as expected result
4	Attempt to make payment.	Click "Payment" button in checkout screen.	Redirect user to payment screen. Once successful, move the paid items to Order History and empty the checkout screen.	Same as expected result

Table 6.1.1.6 Checkout Functionality on User Site

6.1.1.7 Test Scenario: Personalized Health Management on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Normal User attempt to Add to Cart.	Select "Normal" Sugar Level Select "L" Size	Prompt Users: "Added Successfully"	Same as expected result
2	Diabetes patient with low- and high-level attempt to Add to Cart.	Select "Normal" Sugar Level	Alert Message: "Are you sure want to select Normal Sugar Level", User can proceed or cancel the selection of normal sugar level.	Same as expected result
3	Diabetes patient with low- and high-level attempt to Add to Cart.	Select "Slight" or "No Sugar" Sugar Level	Prompt Users: "Added Successfully"	Same as expected result
4	Cholesterol patient with low- and high-level attempt to Add to Cart.	Select "L" Size	Alert Message: "Are you sure want to select L Size", User can proceed or cancel the selection of large cup of option.	Same as expected result
5	Cholesterol patient with low- and high-level attempt to Add to Cart.	Select "S" or "M" Size	Prompt Users: "Added Successfully"	Same as expected result

Table 6.1.1.7 Personalized Health Management Functionality on User Site

6.1.1.8 Test Scenario: Check Favorite Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result

1	Attempt to add items to Favorite.	Click on the heart icon in Coffee Details Screen.	Prompt Users: "Favorite Added Successfully"	Same as expected result
2	Attempt to view Favorite.	Click on the heart icon in downright of Home Screen.	View Favorite Items.	Same as expected result
3	Attempt to delete Favorite item.	Click "Delete" to the selected favorite item.	Prompt Users: "Favorite Deleted Successfully"	Same as expected result

Table 6.1.1.8 Check Favorite Functionality on User Site

6.1.1.9 Test Scenario: Check Payment Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to check out and make payment.	Click on the payment button in Checkout Screen.	Redirect to the payment screen.	Same as expected result
2	Attempt to make payment.	Click on the QR code.	Redirect users to the TnG Log In screen scan QR.	Same as expected result
3	Attempt to pay the items.	Click the QR and made payment.	Prompt Users: "Paid Successfully"	Same as expected result

Table 6.1.1.9 Check Payment Functionality on User Site

6.1.1.10 Test Scenario: Check Settings Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to go to Settings Page.	Click on the Setting icon in Home Screen.	Redirect to the Setting screen.	Same as expected result
2	Attempt to select in Setting Page.	Setting Page Button: Profile, Go to Menu, Feedback, Log Out	Redirect users to the selected page accordingly.	Same as expected result

3	Attempt to select Profile in Setting Page.	Update Profile and fill in the information.	Prompt Users: "Profile Update Successfully"	Same as expected result
4	Attempt to select Go to Menu in Setting Page.	Click on the "Go to Menu" button.	Redirect user to Home Screen.	Same as expected result
5	Attempt to select Feedback in Setting Page.	Send Feedback by categories.	Prompt Users: "Feedback Summited"	Same as expected result

Table 6.1.1.10 Check Settings Functionality on User Site

6.1.1.11 Test Scenario: Check Order History Functionality on User Site

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to check Order History.	Click on the Order History icon in Tab Bar of Home Screen.	Redirect to the Order History screen.	Same as expected result
2	Attempt to view Order History.	View on the Order History.	Redirect to the Order History screen without editable.	Same as expected result

Table 6.1.1.11 Check Order History Functionality on User Site

6.1.2 Admin Site Functionality Testing

Testing the admin site functionality is essential to ensure the reliability and security of the system.

6.1.2.1 Test Scenario: Check Admin Registration Account Functionality

No	Test Case	Value	Expected Result	Actual Result
1	Verify all the information enters by the admin after click on the register button with all valid input.	Enter Valid User Details: Display Name, Email, Password, Confirm Password, Gender, Birthdate.	User account is Successfully Registered, and the user is redirected to the intended page.	Same as expected result

2	Attempt to register with an email that is already registered.	Enter an email that is already associated with an existing account.	Display an error message: "Email Already in Use."	Same as expected result
3	Attempt to register with an invalid email format.	Enter an email with an incorrect format (e.g., missing '@','.com' symbol).	Display an error message: "Invalid Email."	Same as expected result
4	Attempt to register with missing fields.	Leave one or more required fields (e.g., Display Name, Email, Gender, Birthdate, and Password) blank	Display an error message: "Email(missing) is Required."	Same as expected result
5	Attempt to register with valid information and navigate away from the registration page before submitting.	Enter valid user details but leave the registration page without submitting.	Registration failed, and no account is registered.	Same as expected result

Table 6.1.2.1 Test Scenario: Check Admin and Staff Login Functionality

6.1.2.2 Test Scenario: Check Add New Beverages Functionality

No	Test Case	Value	Expected Result	Actual Result
1	Attempt to add new beverages product.	Click the Add New Beverages Button. Fill up required information: Upload Image, Coffee ID, Coffee Name, Coffee Price, Description, Coffee Category, Coffee Included.	Prompt Admin: "Beverages Added Successfully"	Same as expected result

2	Attempt to View added beverages list.	Click on the view beverages list button.	Display list of added beverages items with image, name, created by who (which admin), date and time, Coffee ID.	Same as expected result
3	Attempt to View details of added beverages product.	Click on the selected beverages list.	Redirect admin to beverages details.	Same as expected result
4	Attempt to Edit details of added beverages product.	Swipe left of the selected beverages drink and click Edit.	Redirect admin to beverages edit screen.	Same as expected result
5	Attempt to Delete details of added beverages product.	Swipe left of the selected beverages drink and click Delete.	Prompt Admin: "Deleted Successfully"	Same as expected result

Table 6.1.2.2 Test Scenario: Check Add New Beverages Functionality

6.2 Project Challenges

As we progress through the development of our proposed beverages application with a personalized health management system, we have encountered several challenges specific to our use of React Native. These hurdles have required innovative solutions and careful navigation to ensure the success and stability of our project.

One of the primary challenges we've faced is managing incompatible React Native libraries. Integrating third-party libraries is essential for expanding functionality, but compatibility issues can arise, requiring us to select and integrate libraries that align seamlessly with our project goals. Similarly, navigating incompatible library versions has been a recurring challenge. Developers have to plan a systematic approach to reviewing and updating library versions, ensuring that project's dependencies remain compatible and reliable throughout the development process.

Furthermore, configuring React Native environments presents unique challenges, particularly for companies unfamiliar with this technology. The React Native market in Malaysia is still developing, with relatively few companies utilizing this environment. As a result, setting up the proposed development environment for optimal performance and seamless

integration has required meticulous attention to detail and a proactive troubleshooting approach.

Moreover, the rapid pace of updates within the React Native ecosystem presents ongoing compatibility challenges. With each new update, there is a risk that existing libraries or projects may become incompatible, requiring continuous monitoring and adaptation to maintain project stability. Regular updates are essential to ensure compatibility and troubleshoot any issues that arise with React Native libraries. This proactive approach allows us to address potential compatibility issues promptly and keep the project on track towards successful completion.

6.3 Objective Evaluation

Objective 1: Recommend Best-Selling Products

Evaluation Criteria:

Feature Implementation: Assess the integration of recommendation features based on user preferences and purchasing patterns.

API Utilization: Evaluate the use of APIs access to the Weather and Location service and provide meaningful recommendations.

Black Box Testing Relevance:

User Site Functionality Testing: Test scenarios related to account login, home screen categorization, and Pop-Up modal functionalities used to support recommendation features.

Conclusion: The recommendation feature's success will be determined by its ability to enhance user decision-making and overall satisfaction.

Objective 2: Implement a Personalized Health Management System

Evaluation Criteria:

Incorporation of Health Parameters: Assess how diabetes and cholesterol levels are integrated into the user profile.

Alert System Effectiveness: Evaluate the system's ability to provide informative alerts based on health conditions during beverage selection.

User Engagement: Measure user engagement with health management features and feedback received.

Black Box Testing Relevance:

Personalized Health Management Testing: Test scenarios focused on personalized health management will reveal the system's performance in guiding users based on health conditions.

Conclusion: The success of this objective will be determined by the application's impact on promoting healthier beverage choices and improving user well-being.

Objective 3: Include Proper Categorization and Search Field Features

Evaluation Criteria:

Categorization Design: Assess the effectiveness of categorization in aiding user navigation and exploration.

Search Functionality: Evaluate the search feature's efficiency in helping users locate specific beverages.

Usability Enhancement: Measure improvements in user experience attributed to proper categorization and search functionalities.

Black Box Testing Relevance:

User Site Functionality Testing: Test scenarios related to home screen categorization, search field functionality, and coffee details screen provide insights into the usability and effectiveness of these features.

In conclusion, the application's success in this area will be determined by improved usability and user engagement facilitated by effective categorization and search capabilities. The evaluation will check how well the app meets its goals and how users interact with it.

Chapter 7 Conclusion and Recommendations

7.1 Conclusion

Throughout the journey of this project, we embarked on the development of a **beverage application with a personalized health management system**, aimed at addressing critical challenges faced by existing beverage apps. The motivation stemmed from the rapid integration of digital tools in health management and the desire to enhance user experience in the beverage consumption realm. The objectives of this project focused on filling gaps such as the absence of drink recommendations, limited health management features, and inefficient categorization within current beverage applications.

Drawing from industry trends and successful models like Zus Coffee, this project set out to implement innovative features that resonate with health-conscious consumers. The literature review underscored the significance of personalized recommendations and health management systems in driving user engagement and satisfaction. This informed our approach to leverage technologies like React Native, APIs for weather-based recommendations, and Firebase for robust data management.

The system design phase emphasized a user-centric approach, ensuring intuitive navigation and seamless interactions. Implementation encompassed features like personalized drink recommendations based on weather and location and health alerts to guide users toward healthier choices and meet user preferences. The application's prioritized scalability and cross-platform compatibility to reach a broader audience.

Through systematic black box testing in Chapter 5, we rigorously evaluated the application's functionalities across user and admin interfaces. Test scenarios spanned account management, categorization, search, checkout, and health management, providing valuable insights into usability and performance. The outcomes demonstrated successful implementation of our defined objectives:

Firstly, the recommendation feature effectively suggested drinks based on user preferences and real-time weather conditions using APIs, enhancing user decision-making and satisfaction. Secondly, the health management system delivered personalized alerts and guidance, empowering users to make informed, health-conscious choices when selecting beverages. Lastly, the proper categorization and search functionalities improved user experience by facilitating efficient navigation and enabling quick access to desired beverage options.

In conclusion, the proposed beverage application with a personalized health management system represents a significant advancement in the digital beverage app landscape. By integrating innovative features and addressing user needs, the application promotes wellness through beverage consumption, enhances user engagement, and offers a seamless digital experience. Moving forward, user feedback will guide iterative improvements to ensure the app remains impactful and user-centric in promoting healthier lifestyles through beverages.

7.2 Future Enhancement and Recommendations

For future enhancements and recommendations of the proposed beverages application, several strategies can further elevate the beverage application with a personalized health management system. Firstly, expanding personalized recommendations through machine learning algorithms could enhance user satisfaction by analyzing evolving preferences and behaviors to deliver more precise drink suggestions. This approach would leverage user data over time to tailor recommendations based on individual tastes and health goals, offering a more refined and personalized experience.

The second recommendation is to integrate wearable technology like fitness trackers or smartwatches would enable real-time health metric monitoring, facilitating immediate personalized recommendations based on the user's current health status. By syncing with these devices, the app could provide contextual insights and suggest beverages that align with users' health metrics, such as heart rate, activity levels, or sleep patterns.

Furthermore, developing an extensive analytics dashboard for administrators to track user behaviors and health trends would provide valuable insights for strategic app improvements. Understanding user preferences and usage patterns can inform future updates and features, ensuring the app remains relevant and effective.

In addition, establishing a continuous feedback loop with users for iterative updates to enhance usability, functionality, and overall experience is essential. Regularly gathering user feedback and incorporating suggestions into app updates demonstrates responsiveness to user needs and ensures ongoing improvement.

REFERENCES

[1] A. Jacob, “How technology is elevating the food and beverage industry,” How Technology is Elevating the Food and Beverage Industry. [Online]. Available: https://www.ordermentum.com/blog/how-technology-is-elevating-the-food-and-beverage-industry. [Accessed: 01-Dec-2022].

[2] “About Us - Zus Coffee: Malaysia #1 tech-driven coffee chain,” *ZUS Coffee | Malaysia #1 Tech-Driven Coffee Chain*, 17-May-2022. [Online]. Available: https://zuscoffee.com/about-us/. [Accessed: 27-Mar-2023].

[3] Author links open overlay panelYu-Hua Christine Sun et al., “Health Concern, food choice motives, and attitudes toward healthy eating: The mediating role of Food Choice Motives,” Appetite, https://www.sciencedirect.com/science/article/abs/pii/S0195666307004254 [Accessed: 27-Mar-2024].

[4] “Beverage formulation and food formulation: 4 key trends,” *CPG, F&B and Cosmetic: a blog on NPD issues and software*, 19-Dec-2018. [Online]. Available: https://www.inno-foodproducts-brainbox.com/2018/04/25/4-trends-food-and-beverage-formulations/. [Accessed: 01-Dec-2022].

[5] M. Sandesara et al., “Design and experience of mobile applications: A pilot survey,” MDPI, https://www.mdpi.com/2227-7390/10/14/2380 [accessed Mar. 20, 2024].

[6] L. Harrison, “The Effects of Adding Search Functionality to Interactive Visualizations on the Web,” codementum.org, http://codementum.org/ [Accessed: 27-Mar-2024].

[7] “Coffee and cholesterol: Health risks, benefits, and more,” Medical News Today, https://www.medicalnewstoday.com/articles/coffee-and-cholesterol#:~:text=While%20coffee%20does%20not%20contain,total%20cholesterol%20and%20LDL%20levels. [Accessed: 27-Mar-2023].

[8] “Food delivery app: User statistics 2022,” *Food Delivery App | User Statistics 2022*. [Online]. Available: https://www.appsrhino.com/blogs/food-delivery-app-user-statistics-2022. [Accessed: 27-Mar-2023].

[9] “What is an API? (application programming interface),” *MuleSoft*. [Online]. Available: https://www.mulesoft.com/resources/api/what-is-an-api#:~:text=API%20is%20the%20acronym%20for,you're%20using%20an%20API. [Accessed: 01-Dec-2022].

[10] G. Singh, “How to add geolocation feature in your application,” *Appinventiv*, 29-Aug-2022. [Online]. Available: https://appinventiv.com/blog/guide-on-adding-geolocation-in-mobile-app/. [Accessed: 01-Dec-2022].

[11] “Why and when to use API keys | cloud endpoints with openapi | google cloud,” *Google*. [Online]. Available: https://cloud.google.com/endpoints/docs/openapi/when-why-api-key#:~:text=API%20keys%20provide%20project%20authorization,-To%20decide%20which&text=By%20identifying%20the%20calling%20project,or%20enabled%20in%20the%20API. [Accessed: 01-Dec-2022].

[12] “Limits and quotas on API requests | analytics core reporting API | google developers,” *Google*. [Online]. Available: https://developers.google.com/analytics/devguides/reporting/core/v3/limits-quotas#:~:text=General%20quota%20limits,-The%20following%20quotas&text=10%20queries%20per%20second%20(QPS,a%20maximum%20value%20of%201%2C000. [Accessed: 01-Dec-2022].

[13] “Using libraries · REACT NATIVE,” React Native RSS, https://reactnative.dev/docs/libraries [accessed Mar. 20, 2024].

[14] N. Lorincz, “18 product recommendation examples to increase sales,” *OptiMonk*, 18-Nov-2022. [Online]. Available: https://www.optimonk.com/product-recommendation/. [Accessed: 01-Dec-2022].

[15] The paradox of choice - positive psychology in practice, https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118996874.ch8 [Accessed: 27-Mar-2023].

[16] Charter Global, "What is SDLC? what are the 5 phases in the Software Development Life Cycle (SDLC)?," Charter Global, https://www.charterglobal.com/what-are-the-5-phases-in-the-software-development-life-cycle-sdlc/ [Accessed: 27-Mar-2023].

[17] "Why use react native for mobile app development?," *TheCodeWork*, 15-Jun-2022. [Online]. Available: https://thecodework.com/blog/why-use-react-native-for-mobile-app-development/. [Accessed: 05-Nov-2022].

[18] "Firebase vs react native firebase: What are the differences?," *StackShare*. [Online]. Available: https://stackshare.io/stackups/firebase-vs-react-native-firebase. [Accessed: 05-Nov-2022].

[19] "Firebase realtime database," Firebase. [Online]. Available: https://firebase.google.com/docs/database. [Accessed: 02-Nov-2022].

[20] T. Capan, "Why the hell would I use Node.Js? A case-by-case tutorial," Toptal Engineering Blog, 13-Aug-2013. [Online]. Available: https://www.toptal.com/javascript/why-the-hell-would-i-use-node-js. [Accessed: 02-Nov-2022].

[21] "What is node.Js? Complex guide for 2022," Netguru.com. [Online]. Available: https://www.netguru.com/glossary/node-js. [Accessed: 02-Nov-2022].

[22]E.Ighosewe,"ShouldIuseexpoforReactNative,"Upstackhq.com.[Online].Available: https://upstackhq.com/blog/should-i-use-expo-for-react-native. [Accessed: 02-Nov-2022].

[23]M.Heller,"WhatisVisualStudioCode?Microsoft'sextensiblecodeeditor,"InfoWorld, 08-Jul-2022. [Online]. Available: https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html. [Accessed: 02-Nov-2022].

[24] Microsoft, “Microsoft/vscode-react-native: VSCode extension for react native - supports debugging and editor integration,” *GitHub*. [Online]. Available: https://github.com/microsoft/vscode-react-native. [Accessed: 06-Nov-2022].

[25] “Why visual Studio Code?,” Visualstudio.com. [Online]. Available: https://code.visualstudio.com/docs/editor/whyvscode. [Accessed: 02-Nov-2022].

Appendices

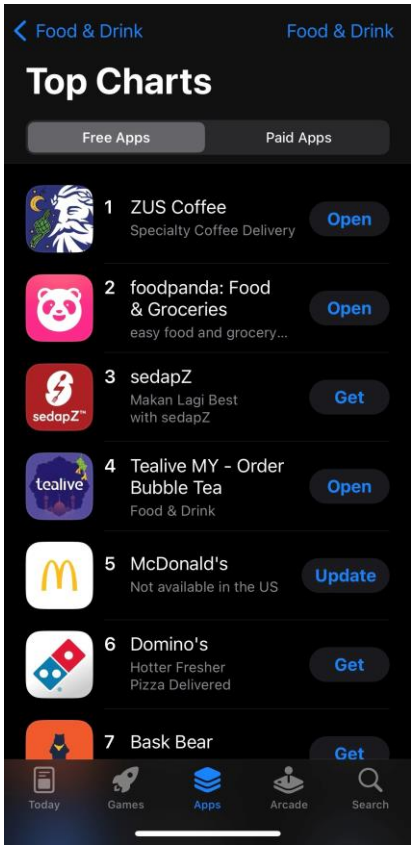

Figure 1.0 Zus Application Ranking and Reviews

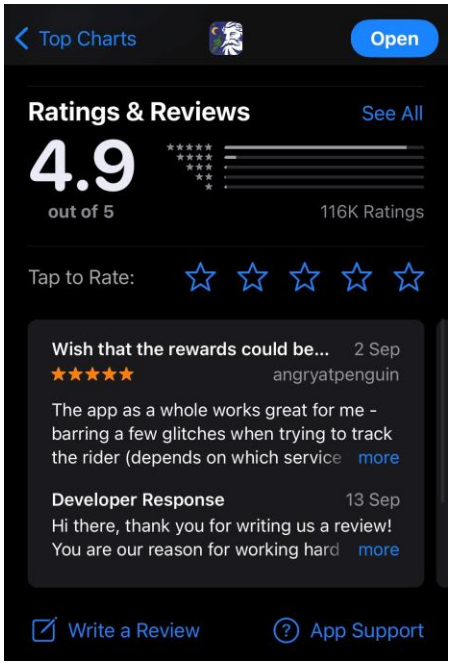

Figure 1.1 Zus Application Reviews Section

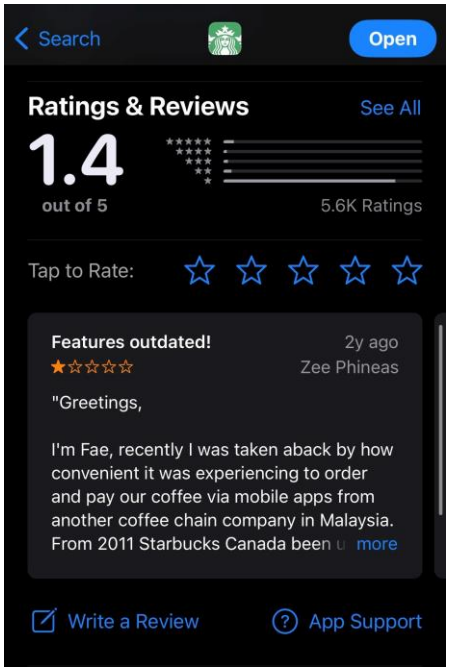

Figure 1.2 Zus Application Reviews Section

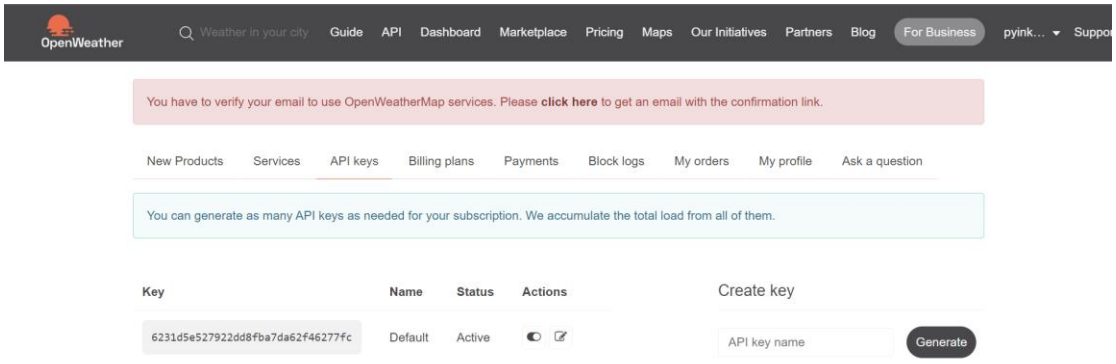

Figure 2.2 Open Weather Service Provider

Figure 2.3 Categorization of Zus Application

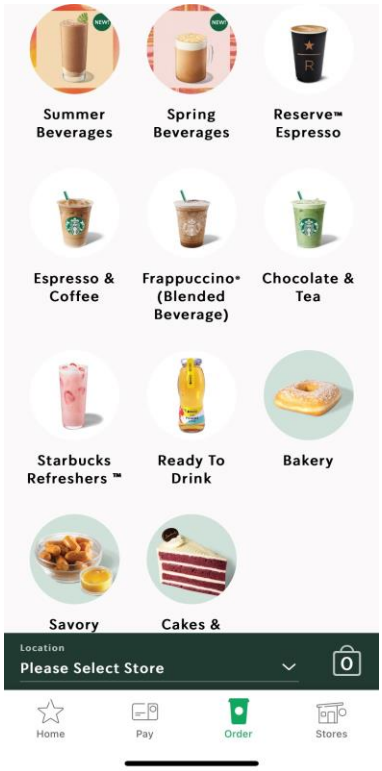

Figure 2.3.1 Categorization of Starbucks Application

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 2**
Student Name & ID: Kong Phui Yin	
Supervisor: Dr Wong Pei Voon	
Project Title: Beverages Application with Personalized Health Management	

1. WORK DONE

Discuss with supervisor for Title Refinement.

2. WORK TO BE DONE

Review of Chapter 1,2 3.

3. PROBLEMS ENCOUNTERED

No problem encounter in week 2.

4. SELF EVALUATION OF THE PROGRESS

The progress is smooth where Dr Wong has guide me on my project's title refinement and the overall project flow.

wong

Supervisor's signature

PhuiYin

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 4**
Student Name & ID: Kong Phui Yin	
Supervisor: Dr Wong Pei Voon	
Project Title: Beverages Application with Personalized Health Management	

1. WORK DONE

Reviewed Chapter 1, 2,3

2. WORK TO BE DONE

Chapter 4: System Design
Chaoter 5: System Implementation
Visual Studio Code 2022.

3. PROBLEMS ENCOUNTERED

No problem encounter in week 4.

4. SELF EVALUATION OF THE PROGRESS

The progress is smooth and keep updating with Dr Wong.

wong
_____
Supervisor's signature

PhuiYin
_____
Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 6**
Student Name & ID: Kong Phui Yin	
Supervisor: Dr Wong Pei Voon	
Project Title: Beverages Application with Personalized Health Management	

1. WORK DONE

Reviewed Chapter 1, 2,3
Chapter 4: System Design

2. WORK TO BE DONE

Chaoter 5: System Implementation
Visual Studio Code 2022.

3. PROBLEMS ENCOUNTERED

No problem encounter in week 6.

4. SELF EVALUATION OF THE PROGRESS

The progress is smooth and starting work on the development of the proposed application.

wong
_____
Supervisor's signature

PhuiYin
_____
Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 8**
Student Name & ID: Kong Phui Yin	
Supervisor: Dr Wong Pei Voon	
Project Title: Beverages Application with Personalized Health Management	

1. WORK DONE

Reviewed Chapter 1, 2,3
Chapter 4: System Design

2. WORK TO BE DONE

Chaoter 5: System Implementation
Visual Studio Code 2022.

3. PROBLEMS ENCOUNTERED

Debug the error on project development process.

4. SELF EVALUATION OF THE PROGRESS

The progress is smooth and starting work on the development of the proposed application.

wong
Supervisor's signature

PhuiYin
Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 10**
Student Name & ID: Kong Phui Yin	
Supervisor: Dr Wong Pei Voon	
Project Title: Beverages Application with Personalized Health Management	

1. WORK DONE

Reviewed Chapter 1, 2,3
Chapter 4: System Design
Chapter 5: System Implementation

2. WORK TO BE DONE

Chapter 6: System Evaluation and Discussion

3. PROBLEMS ENCOUNTERED

Align developed proposed application with project objectives.

4. SELF EVALUATION OF THE PROGRESS

The progress is smooth and keep going work on the system evaluation.

wong
_____
Supervisor's signature

PhuiYin
_____
Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 12**
Student Name & ID: Kong Phui Yin	
Supervisor: Dr Wong Pei Voon	
Project Title: Beverages Application with Personalized Health Management	

1. WORK DONE

Reviewed Chapter 1, 2,3
Chapter 4: System Design
Chapter 5: System Implementation
Project Development.

2. WORK TO BE DONE

Chapter 6: System Evaluation and Discussion
- Black Box Testing

Chapter 7: Conclusion and Recommendation

3. PROBLEMS ENCOUNTERED

No problem encounter in week 12.

4. SELF EVALUATION OF THE PROGRESS

The progress is smooth, and the development of proposed application has been done.

wong
Supervisor's signature

PhuiYin
Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 14**
Student Name & ID: Kong Phui Yin	
Supervisor: Dr Wong Pei Voon	
Project Title: Beverages Application with Personalized Health Management	

1. WORK DONE

Reviewed Chapter 1, 2,3
Chapter 4: System Design
Chapter 5: System Implementation
Chapter 6: System Evaluation and Discussion
Chapter 7: Conclusion and Evaluation
Development of proposed application.

2. WORK TO BE DONE

Compiling proposed beverages application and report.
Submission of the project report.

3. PROBLEMS ENCOUNTERED

No problem encounter in week 14.

4. SELF EVALUATION OF THE PROGRESS

Good.

wong
_____
Supervisor's signature

PhuiYin
_____
Student's signature

POSTER


PROPOSED BEVERAGES APP
Introduction
A new mobile beverages application integrated with personalized health management in Malaysia that supports Android and IOS users which involves certain enhancements of the features compared to the existing beverages application.
Problem Statements
• No feature recommending drinks to users.
• Lack of health management system to guide users select the products.
• Lack of categorization and search field function of products.
Objectives
• Feature recommending drinks based on weather and location.
• Personalized Health Management System
• Meaningful Categorization and Search Field
Deliverable
A beverages application that comes with creativity, innovation and meet the health concern issues of the user in the market.
Organize Coffee Screen
Find the best coffee for you
Cappuccino
Espresso
Latte
Flat White
Personalized Health Management
Hot Cuppucino
Diabetes Alert
Are you sure you want "Normal Sugar" level?
Cancel
Proceed
Size

PLAGIARISM CHECK RESULT

Beverages Application with Personalize Health Management

ORIGINALITY REPORT

6%	4%	1%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dev.to Internet Source	<1%
2	Submitted to University of Central England in Birmingham Student Paper	<1%
3	digitalcollection.utem.edu.my Internet Source	<1%
4	developers.google.com Internet Source	<1%
5	harshp.com Internet Source	<1%
6	Submitted to University of Wales Institute, Cardiff Student Paper	<1%
7	open-innovation-projects.org Internet Source	<1%
8	www.coursehero.com Internet Source	<1%
9	web.era-edta.org	

	Internet Source	<1%
10	Submitted to National College of Ireland Student Paper	<1%
11	Shanshan Tu, Muhammad Ayaz, Abdullah Arshad, Usman Iftikhar, Youseuf Harrath, Muhammad Waqas. "Cloud-based Smart Parking System using Internet of Things", 2023 International Wireless Communications and Mobile Computing (IWCMC), 2023 Publication	<1%
12	ijsred.com Internet Source	<1%
13	Submitted to University of Technology, Sydney Student Paper	<1%
14	core.ac.uk Internet Source	<1%
15	Submitted to United International Business Schools Student Paper	<1%
16	umpir.ump.edu.my Internet Source	<1%
17	www.lib.ncsu.edu Internet Source	<1%

Universiti Tunku Abdul Rahman			
Form Title : Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	Kong Phui Yin
ID Number(s)	20ACB02499
Programme / Course	FICT - IB
Title of Final Year Project	Beverages Application with Personalized Health Management

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceeds the limits approved by UTAR)**
Overall similarity index: 6 % **Similarity by source** Internet Sources: 4 % Publications: 1 % Student Papers: 3 %	
Number of individual sources listed of more than 3% similarity: 0	
Parameters of originality required and limits approved by UTAR are as Follows: **(i) Overall similarity index is 20% and below, and** **(ii) Matching of individual sources listed must be less than 3% each, and** **(iii) Matching texts in continuous block must not exceed 8 words** ***Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.***	

Note Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

wong

Signature of Supervisor

Name: Ts Dr Wong Pei Voon

Date: 24/4/2024

Signature of Co-Supervisor

Name: _____

Date: _____

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY
(KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	20ACB02499
Student Name	Kong Phui Yin
Supervisor Name	Ts. Dr Wong Pei Voon

TICK (√)	DOCUMENT ITEMS Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

PhuiYin_____

(Signature of Student)
Date: 22 April 2024