**EASTERN INTERNATIONAL UNIVERSITY**

**SCHOOL OF COMPUTING AND INFORMATION TECHNOLOGY**

**DEPARTMENT OF SOFTWARE ENGINEERING**



**MOBILE APP REPORT**

GREEN DRINK MOBILE APP

**Student(s)**

Bui Nguyen Hai Ngan – 2231200071

Nguyen Quang Minh – 2231200136

**Supervisor(s)**

Tran Van Tai

Nguyen Xuan Cuong

**Binh Duong, 12, 2025**

ABSTRACT

By catering to the demands of both consumers and shop managers, the Green Drink Mobile App is a ground-breaking digital solution designed to revolutionize the beverage sector. This mobile application makes use of cutting-edge technologies, such as Firestore for safe, scalable, and real-time data management and React Native for smooth cross-platform performance. The app's dual-purpose design gives company owners the ability to efficiently optimize their operations while also offering a complete platform to improve customer comfort. Reducing the time it takes for consumers to purchase and get their beverages is one of its main goals in order to provide a quicker and more effective service experience.

The Green Drink Mobile App offers consumers a feature-rich, user-focused experience designed to make buying beverages and using services easier. A user-friendly interface allows customers to quickly and easily browse and select their preferred beverages, making the ordering procedure pleasurable. Through the integration of a membership program, the app gives users access to special advantages including discounts and incentives, as well as coupons to improve their shopping experience. Customers who are constantly on the go may find local stores with the aid of a shop finder function that uses geolocation technology. Users may also access a comprehensive transaction history for improved financial management, modify their profiles to customize the experience, and remain up to speed on the newest announcements, promotions, and news. These features collectively aim to improve customer engagement, build loyalty, and create a highly personalized service ecosystem, while also significantly reducing the time needed to place and fulfill orders.

From a commercial standpoint, the software gives store managers strong tools to effectively oversee their operations. Drink menu management is a crucial administrative function that allows retailers to dynamically change their selections and guarantee uniformity across all locations. In order to give consumers correct information about addresses and operation hours, the app also offers capabilities for managing shop locations. Order management features improve operational transparency and customer happiness by enabling administrators to monitor and update the progress of client orders in real-time. Customers are guaranteed to receive their beverages on time thanks to the expedited order procedure, which also reduces waits. Additionally, the app has a news management system that lets retailers share events, promotions, and updates with their consumers directly. These features empower businesses to maintain a high level of organization, responsiveness, and customer engagement, thereby driving growth and operational excellence.

The creation of the Green Drink mobile app is an example of how cutting-edge technology and clever commercial ideas can coexist. The software fills the gap between conventional beverage service methods and digital convenience by incorporating necessary features for both customer interaction and administrative efficiency. While its administrative tools tackle the operational challenges encountered by beverage retailers in a competitive market, its customer-facing features are meant to satisfy the increasing need for individualized, on-demand services.

This research emphasizes how mobile applications have the ability to revolutionize established sectors by utilizing technology to provide more value. One example of how digital platforms may promote a win-win connection between companies and consumers, guaranteeing long-term growth and competitiveness, is the Green Drink Mobile App. The app is a crucial tool for companies looking to adjust to changing customer preferences and attain operational excellence in the digital age because of its emphasis on cutting down on order and service times while still offering extensive functionality.

ACKNOWLEDGEMENT

With great appreciation, we would like to thank all the people and organizations that helped us develop and finish the Green Drink Mobile App project.

Above all, we would want to express our sincere gratitude to our academic advisers and mentors for their immense support, knowledge, and encouragement along our trip. Their helpful criticism and suggestions have been crucial in transforming our project into a thorough and workable solution for the beverage sector.

Additionally, we would like to sincerely thank the users and stakeholders who took part in testing sessions, interviews, and surveys. Their suggestions and practical viewpoints were invaluable in helping to improve the app's functionality and design to better serve the needs of store managers and customers.

We also thank our development team members for their cooperation and assistance, as their commitment and technical know-how made it possible to successfully deploy cutting-edge technologies like React Native and Firestore. Overcoming obstacles during the development process was made possible in large part by their dedication to creativity and problem-solving.

We also want to express our gratitude to the companies and beverage merchants that shared information about their operations, which helped us customize the app's features to solve practical problems. Their readiness to share their operational challenges and experiences significantly increased the project's applicability and relevance.

Finally, we would want to express our gratitude to our friends and family for their constant encouragement and support along this journey. Their confidence in our skills inspired us to keep going and accomplish our objectives.

This initiative serves as evidence of the teamwork and common goal of all those engaged. We sincerely thank everyone who has contributed, no matter how tiny, to the development of the Green Drink mobile app. Many thanks to all of you.

**TABLE OF CONTENTS**

[ABSTRACT i](#_Toc184990574)

[ACKNOWLEDGEMENT iii](#_Toc184990575)

[LIST OF FIGURES vi](#_Toc184990576)

[LIST OF TABLES vii](#_Toc184990577)

[LIST OF ABBREVIATIONS viii](#_Toc184990578)

[CHAPTER 1. OVERVIEW 1](#_Toc184990579)

[1.1. Introduction 1](#_Toc184990580)

[1.2. Project objectives 1](#_Toc184990581)

[1.3. Challenges 3](#_Toc184990582)

[1.4. Report structure 4](#_Toc184990583)

[CHAPTER 2. INTRODUCTION TO TECHNOLOGIES 6](#_Toc184990584)

[2.1. Introduction to TypeScript, React Native and NoSQL (Firebase) 6](#_Toc184990585)

[2.1.1. TypeScript 6](#_Toc184990586)

[2.1.2. React Native 7](#_Toc184990587)

[2.2. Introduction to NoSQL – Firestore 7](#_Toc184990588)

[CHAPTER 3. APPLICATION ANALYSIS, DESIGN AND IMPLEMENTATION 9](#_Toc184990589)

[3.1. Analysis 9](#_Toc184990590)

[3.1.1. Understanding of the Core Challenges 9](#_Toc184990591)

[3.1.2. Target Users 9](#_Toc184990592)

[3.1.3. Objectives and goals 9](#_Toc184990593)

[3.1.4. User stories 10](#_Toc184990594)

[3.1.5. Functionalities 11](#_Toc184990595)

[3.1.6. Non-functionalities 11](#_Toc184990596)

[3.2. Database 13](#_Toc184990597)

[3.3. Design 16](#_Toc184990598)

[3.3.1. UI Principles 16](#_Toc184990599)

[3.3.2. Navigation Flow 17](#_Toc184990600)

[3.3.3. System Architecture 17](#_Toc184990601)

[CHAPTER 4. EXPERIMENT AND DISCUSSION 18](#_Toc184990602)

[4.1. Installation Environment 18](#_Toc184990603)

[4.1.1. Development Environment and Dependencies 18](#_Toc184990604)

[4.1.2. Development Tools and Dependencies 18](#_Toc184990605)

[4.1.3. Installation Commands 19](#_Toc184990606)

[4.2. Results and Discussion 19](#_Toc184990607)

[4.2.1. Results: 19](#_Toc184990608)

[4.2.2. Discussion: 19](#_Toc184990609)

[CHAPTER 5. CONCLUSION AND FUTURE WORKS 21](#_Toc184990610)

[5.1. Conclusion 21](#_Toc184990611)

[5.2. Future works 22](#_Toc184990612)

[5.2.1. Branch-Specific Functionality 22](#_Toc184990613)

[5.2.2. Branch-Specific Functionality 22](#_Toc184990614)

[5.2.3. Branch-Specific Functionality 22](#_Toc184990615)

[REFERENCES 24](#_Toc184990616)

LIST OF FIGURES

[Figure 1. Use case diagram of Green Drink App (by authors) 10](#_Toc184990524)

[Figure 2. Navigation flow of Green Drink App (by authors) 17](#_Toc184990525)

LIST OF TABLES

[Table 1. Users 13](#_Toc184990783)

[Table 2. Categories 13](#_Toc184990784)

[Table 3. Drinks 13](#_Toc184990785)

[Table 4. Transaction 14](#_Toc184990786)

[Table 5. Stores 14](#_Toc184990787)

[Table 6. News 14](#_Toc184990788)

LIST OF ABBREVIATIONS

|  |  |  |
| --- | --- | --- |
| **No.** | **Term** | **Meaning** |
| 1 |  |  |
| 2 |  |  |

# OVERVIEW

## Introduction

The Green Drink Mobile App is specifically designed to meet the changing needs of consumers and businesses in the beverage sector. With the increasing reliance on mobile technology in daily service, this app aims to improve the beverage ordering process by providing customers with seamless, efficient and personalized experiences while equipping store managers with powerful management tools to streamline operations. As consumers increasingly value convenience and speed, especially in the food and beverage (F&B) industry, mobile apps such as Green Drink play a critical role in providing real-time updates, adjustments, and faster ordering (Chen & Lu, 2020; Kromeon, 2023 ).

The app provides a user-friendly interface that allows customers to browse and select drinks, track orders, and take advantage of features such as coupons, membership rewards, and transaction history management. Store managers can use the app to process orders in real time, update menus, manage store locations and notify customers of news or promotions. These features aim to improve customer satisfaction, operational efficiency and engagement, which is crucial in the competitive beverage market (W2S Solutions, 2020).

To ensure smooth and scalable operations, the app uses React Native for cross-platform compatibility and Firestore for real-time data synchronization. These technological decisions ensure the app delivers fast and reliable performance even during high demand, which is critical for beverage companies that often experience peak order periods. Through its innovative combination of customer-centric features and robust management tools, the Green Drink Mobile App sets a new standard for the beverage service industry by improving the efficiency of beverage ordering and beverage service (Statista, 2022; W2S Solutions, 2020).

## Project objectives

The Green Drink Mobile App project's main goal is to create a customized mobile application that streamlines and improves consumers' drink ordering experiences while giving shop managers effective management capabilities. With features particularly crafted to enhance customer happiness and operational efficiency, this app is made to cater to the demands of beverage businesses. The following are the project's particular goals:

* Simplify the Procedure for Ordering Drinks

Give clients a smooth ordering process so they can peruse the menu, place orders for pickup or delivery, and monitor the progress of their orders.

* Reduce Time to Order and Receive Drinks

In order to guarantee that orders are processed and delivered effectively, reduce wait times for both pick-up and delivery services, and optimize order processes by integrating real-time data synchronization.

* Enhance Customer Engagement and Retention

To increase satisfaction and foster loyalty, incorporate customer-focused features like transaction history monitoring, unique coupons, membership rewards programs, and personalized profiles.

* Assist Customers in Easily Finding Stores

Provide a store finder feature so that customers can easily locate and navigate to their chosen shop locations and adjacent beverage outlets.

* Provide Store Administrators with Robust Management Tools

Give store managers the ability to control drink menus, update location information, track past transactions, and post news or promotional materials in order to successfully interact with consumers.

* Improve Information Access and Updates

To keep consumers informed and involved with the company, provide them with the most recent news, special offers, and updates straight from the app.

* Ensure Scalability and Reliability

To ensure that the application runs smoothly under a range of loads, leverage scalable cloud-based technologies, like Firestore, to enable heavy user traffic and real-time data management.

* Utilize Cross-Platform Development for Broad Accessibility

Leverage React Native to build a single codebase that delivers a consistent and responsive user experience on both Android and iOS devices, maximizing the app’s reach.

* Enhance Operational Efficiency for Beverage Businesses

Simplify store operations by providing tools for managing orders, updating menus, and tracking transactions, enabling stores to focus on delivering high-quality customer service.

* Support the Digital Transformation of Beverage Services

Demonstrate how digital solutions can modernize traditional drink-ordering processes, fostering innovation and improving customer experiences in the beverage industry.

## Challenges

To guarantee the application's success and usability, a number of obstacles must be overcome throughout the development of the Green Drink mobile app. These difficulties fall into three categories: technological, operational, and user-experience.

* Integrating Multiple Functionalities

Balancing the implementation of diverse features such as browsing menus, ordering (pick-up and delivery), membership programs, vouchers, store locators, and news updates while maintaining a seamless user experience requires meticulous planning and execution.

* Real-Time Data Synchronization

Ensuring that data such as order statuses, store information, and transaction histories are updated in real-time across multiple devices and user roles (customers and store administrators) demands a reliable and scalable backend solution.

* Scalability and Performance

A major technological difficulty is creating an application architecture that can manage heavy traffic and high transaction volumes, particularly during peak hours. The application must continue to function consistently under different loads.

* Cross-Platform Compatibility

Using React Native to create a uniform user experience across the iOS and Android platforms necessitates paying close attention to platform-specific testing, debugging, and optimizations.

* User Data Security and Privacy

Managing sensitive customer information, such as profile details, transaction histories, and payment information, demands robust security measures and compliance with data protection regulations.

* User Adoption and Retention

Encouraging users to adopt the app and engage with features such as membership programs, vouchers, and news updates requires intuitive design and effective promotional strategies. Retaining users in a competitive market also involves continuous improvements and feature updates.

* Efficient Delivery and Pick-Up Coordination

Integration with dependable delivery partners or systems is necessary for the order delivery capability to ensure accurate and timely delivery. The app must offer precise time estimates and real-time updates for orders that are picked up.

* Store Management Complexity

It might be difficult to give shop managers the tools they need to effectively and intuitively handle menus, locations, orders, and promotional content, especially when expanding to many sites with different setups.

* Localization and Accessibility

Ensuring the app accommodates diverse user preferences, languages, and accessibility needs can increase its adoption across different demographics and regions.

* Testing and Quality Assurance

To guarantee a flawless and error-free user experience, the app must be thoroughly tested on a variety of devices, networks, and user scenarios. Ensuring dependability and covering all edge circumstances provide a difficulty.

## Report structure

The report is organized into the following chapters to provide a comprehensive overview of the Green Drink Mobile App project:

* Chapter 1: Overview

This chapter provides an overview of the project, including its goals, parameters, and general format. It explains the primary objectives of the Green Drink Mobile App and gives background information on the project's purpose.

* Chapter 2: Introduction to Technologies

The main technologies utilized in the app's development, such TypeScript, React Native, and Firestore, are examined in this chapter. It talks about how each technology enhances the application's performance and usefulness.

* Chapter 3: Application Analysis, Design, and Implementation

This chapter examines the app's needs, outlines the design process, and goes into depth about the steps of execution. The database design, the app's architecture, and how the features are combined to satisfy user and business goals are all covered in this chapter.

* Chapter 4: Experiment and Discussion

This chapter describes the app's testing and assessment process, including the installation environment, the outcomes, and any difficulties encountered. The performance and usability of the app are also discussed in this chapter.

* Chapter 5: Conclusion and Future Works

The project's findings are compiled in the final chapter, which also makes inferences from the app's testing and deployment outcomes and recommends possible directions for future development.

* References

This section lists all the academic papers, articles, and sources referenced throughout the report, following APA citation standards.

* Appendix

The appendix contains supplementary materials, such as diagrams, code snippets, and additional data relevant to the project.

# INTRODUCTION TO TECHNOLOGIES

## Introduction to TypeScript, React Native and NoSQL (Firebase)

### TypeScript

TypeScript is an advanced, statically typed programming language designed as a superset of JavaScript. Created and maintained by Microsoft, it combines static typing and other modern programming paradigms with the dynamic nature of JavaScript. TypeScript compiles to standard JavaScript, ensuring compatibility with all JavaScript engines, including web browsers and Node.js environments (Microsoft, 2023). The integration of modern JavaScript features coupled with robust static type checking has made it a popular choice for developers building large-scale, maintainable applications.

When developing the Green Drink app, TypeScript plays a crucial role by improving code quality and reducing runtime errors. Its static typing system ensures that variable types, function signatures and interfaces are explicitly defined. This results in code that is more predictable and easier to debug than JavaScript, which is dynamically typed (Johnson & Wilson, 2022). These features are particularly valuable in large, collaborative projects where multiple developers contribute and ensure consistency and clarity across the codebase.

TypeScript's robust tool ecosystem also contributes to its popularity. Features like IntelliSense, which provides code completion, real-time type checking, and error detection, improve developer productivity. The TypeScript language server integrates seamlessly with modern integrated development environments (IDEs) such as Visual Studio Code, providing a comprehensive development experience. Additionally, TypeScript supports advanced features such as generics, decorators, and modules, allowing developers to create scalable and modular code bases (Chowdhury & Kazi, 2022).

The compatibility of TypeScript with React Native is particularly important for the Green Drink app. It provides advanced type definitions for React components, props, and states, reducing the risk of common runtime errors and simplifying the refactoring process as the application evolves. This ensures that features like drink customization, order tracking, and user rewards are implemented consistently and efficiently.

TypeScript also makes it easier to adopt modern JavaScript features, including ES modules, async/await syntax, and destructuring, while maintaining backwards compatibility. This allows the Green Drink app to use state-of-the-art technology without affecting functionality on older platforms. Additionally, TypeScript's interface and class structures help enforce a robust data schema for the app, ensuring seamless interaction with backend services such as Firebase and Firestore.

Choosing TypeScript for the Green Drink app shows their commitment to developing a robust, scalable and maintainable solution. By addressing common challenges in large software projects, TypeScript helps deliver a great user experience while streamlining development workflows.

### React Native

React Native is a powerful framework for building cross-platform mobile applications using JavaScript and React. It enables developers to build powerful apps that run on both iOS and Android using a single codebase. React Native leverages native components and combines them with JavaScript to ensure apps provide a user experience close to that of native apps while sharing most of the application logic (Facebook, 2020). This is particularly beneficial in app development for services like Green Drink, which require a fast, responsive and efficient interface to handle real-time drink orders, locations and personalized user experiences.

In Green Drink App, React Native plays a crucial role in providing a consistent experience across both platforms while allowing developers to focus on building features instead of maintaining two separate codebases. The app makes extensive use of libraries such as react-native-maps for location-based services, react-native-picker for customizable drink selection, and react-native-paper for creating consistent and attractive UI components. React Native's “learn once, write anywhere” approach helps streamline development, reduce time to market, and reduce costs by allowing the app to be built and maintained with fewer resources (Ramos, 2021).

Green Drink App benefits from React Native integration with backend services like Firestore for real-time data storage and management. Using React-Native-Firebase, the app ensures that user data, transactions and orders are efficiently synced across devices, providing an optimal experience for customers and store managers alike.

## Introduction to NoSQL – Firestore

NoSQL databases represent a modern approach to data management and are particularly suitable for applications that require flexibility, scalability and real-time performance. Unlike traditional relational databases that rely on structured schemas and tables, NoSQL databases store data in a variety of formats, such as key-value pairs, document structures, charts, or wide column stores. This schemaless nature allows developers to dynamically adapt data models as application requirements evolve (Couchbase, 2021).

Firebase, a backend-as-a-service (BaaS) platform developed by Google, integrates a powerful NoSQL database solution in the form of Firestore and the Firebase Realtime Database. These databases are intended to provide cloud-hosted, scalable, and synchronized data management tailored to modern mobile and web applications (Google, 2023). Firebase NoSQL databases simplify development workflows by handling complex tasks such as real-time synchronization, offline support, and security rule enforcement.

Firestore, one of Firebase's flagship NoSQL databases, organizes data into collections and documents, providing a flexible and intuitive structure. Each document can store data in key-value pairs or arrays, making it highly adaptable to applications with different data requirements. This flexibility is critical for apps like the Green Drink Mobile App, where features like customizable drink orders, user preferences, and transaction histories require dynamic data structures.

One of the main advantages of Firebase's NoSQL databases is their real-time synchronization capability. Changes made to the database are immediately reflected across all connected devices, giving users live updates. This is especially important in multi-user environments where data consistency is critical, such as when updating order status in the Green Drink app. Additionally, Firebase offers offline persistence so the app works seamlessly without an active internet connection and syncs data once connectivity is restored (Ramos, 2022).

Firebase also includes built-in security and access management through customizable rules. These rules allow developers to define granular access permissions to ensure sensitive user data is protected and enable seamless data sharing when needed. For example, the Green Drink app can restrict access to administrative functions such as order management while allowing customers to securely view their personalized data.

In addition to its technical strengths, Firebase integrates closely with the broader Google Cloud ecosystem and offers features such as analytics, machine learning, and hosting. This holistic approach allows developers to build and scale applications efficiently, making Firebase the ideal choice for the Green Drink app.

# APPLICATION ANALYSIS, DESIGN AND IMPLEMENTATION

## Analysis

### Understanding of the Core Challenges

The beverage sector encounters a myriad of challenges pertaining to operational efficiency, consumer satisfaction, and digital transformation. Conventional processes for ordering beverages frequently result in delays, miscommunication, and restricted customization alternatives for consumers. Enterprises encounter difficulties in effectively managing real-time orders, promotional initiatives, and customer engagement. The fundamental challenges encompass:

* Order Management: Ensuring efficient handling of real-time orders from a specific location or franchise network.
* Customer Retention: Building loyalty through vouchers, rewards, and personalized engagement.
* Operational Efficiency: Reducing manual processes in managing orders, inventory, and customer preferences.
* Brand Identity: Maintaining a unique identity in the digital space with a custom-branded app experience.

The Green Drink Mobile Application aspires to mitigate these challenges by delivering an integrated ordering experience for consumers and comprehensive management tools for enterprises.

### Target Users

The primary target users of the Green Drink Mobile App are:

* End Customers: Regular patrons and new visitors seeking convenient, fast, and customizable drink-ordering options.
* Store Managers/Employees: Staff responsible for handling orders, managing menus, and updating promotions in real-time.

### Objectives and goals

The aims of the Green Drink Mobile Application comprise:

* Enhanced Customer Engagement: Delivering a proprietary application experience that cultivates brand loyalty.
* Operational Optimization: Streamlining internal procedures such as order monitoring, menu modifications, and promotional oversight.
* Revenue Enhancement: Promoting repeat consumer transactions through incentives such as vouchers and loyalty rewards.
* Data-Informed Decision Making: Furnishing insights into consumer preferences and operational efficacy to facilitate strategic planning.

### User stories

User narratives facilitate a comprehensive comprehension of the application’s functionalities from the viewpoint of the user. By employing Unified Modeling Language (UML) diagrams, one should methodically traverse the following stages: evaluate the actors involved, determine the requirements of the actors for the system, establish associations, model the relationships, and refine the diagram (Ambler, n.d.). The culmination of these processes results in the development of a use case diagram, which serves to delineate in a lucid, accessible, and exhaustive manner the features that the application must fulfill.

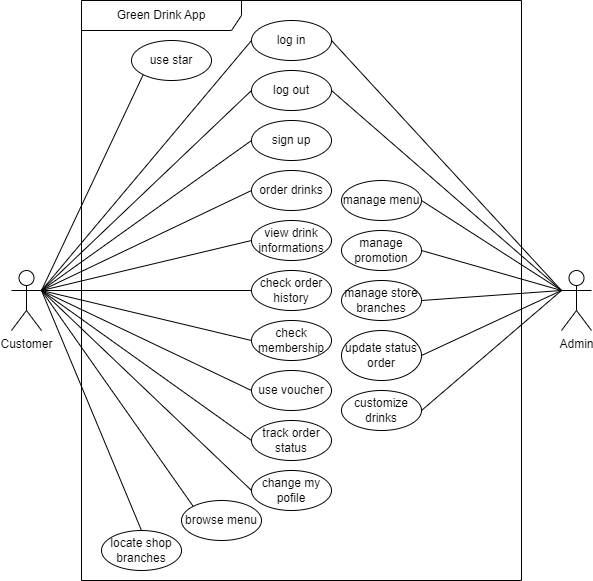


Figure . Use case diagram of Green Drink App (by authors)

### Functionalities

* Customer functionalities
* Order Management: Users are afforded the opportunity to peruse, personalize, and execute orders for beverages directly via the application.
* Loyalty and Rewards: Membership programs that reward customers with points and discounts for their purchases.
* Promotions and Discounts: Access to exclusive offers and promotional vouchers through the app.
* Store Locator: A map-based feature to find nearby beverage outlets or delivery zones.
* User History: The application maintains a comprehensive record of prior orders and transactions to facilitate expedited reordering.
* Business functionalities
* Menu Management: Enables businesses to make real-time updates to beverage menus, introduce new offerings, and adjust pricing as needed.
* Order Processing: Facilitates the effective management of orders, providing options to prioritize tasks or handle busy periods.
* Promotion Announcements: Offers tools for disseminating information about special promotions or the launch of new products.
* Customer Insights: Provides analytical data on customer preferences and buying patterns to enhance marketing strategies.

### Non-functionalities

The key non-functional requirements for the Green Drink Mobile App include:

* Performance

For users to experience seamless navigation and few loading pauses, the application must have quick reaction times and effectively manage several concurrent transactions. While React Native guarantees effective rendering for a flawless experience on both iOS and Android devices, Firebase's real-time database capabilities aid in optimizing order processing performance.

* Scalability

The application needs to grow with the amount of users and data quantities. The app can accommodate increasing traffic and expanding drink menus without experiencing performance deterioration because to Firebase's cloud infrastructure's horizontal scalability.

* Availability

For the app to efficiently support consumers and beverage businesses, high availability is essential. Because of its distributed architecture, Firebase provides strong uptime guarantees, guaranteeing that the application will always be available, even during periods of high traffic.

* Security

It is crucial to protect user data, including payment details and login credentials. The application uses Firestore's role-based security rules to manage data access and Firebase Authentication to provide safe login procedures. Furthermore, TypeScript's robust typing helps avoid frequent programming mistakes that could result in security flaws.

* Usability

The app's user-friendly interface makes it easy to buy drinks and handle delivery, thanks to its user-centric design. The component-based architecture of React Native makes it possible to create reusable and easily accessible user interface elements that improve usability. The user experience is further simplified by features like geolocation and in-app navigation, which are made possible by libraries like react-native-maps.

* Cross-Platform Compatibility

React Native minimizes development and maintenance efforts while guaranteeing consistent functionality across devices by allowing the program to run smoothly on both the iOS and Android platforms with a single codebase.

* Maintainability

Modern, modular technologies like TypeScript and React Native, which enable clean and scalable codebases, are used in the app's development. Long-term maintenance and debugging are made easier by TypeScript's error-catching features and Firebase's automated updates.

## Database

Table 1. Users

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Description** |
| Key | String (Auto Generate) |  |
| Username\* | String |  |
| Password\* | String | Hash |
| FormOfAddress | String | Mr, Ms |
| PhoneNumber | String |  |
| Address | String |  |
| Email\* | String |  |
| CreatedAt | TimeStamp |  |
| Status | String | Active, Inactive, Banned |
| Role | String | Customer, Admin |

Table 2. Categories

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Description** |
| Key | String (Auto Generate) |  |
| Name | String |  |
| Image | String |  |

Table 3. Drinks

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Description** |
| Key | String (Auto Generate) |  |
| Name | String |  |
| Description | String | Describe detail about a product |
| CategoryName | String |  |
| Customization | Size | String: S, M, L |
| Sweetness | String: Regular, Less sweet |
| Price | Number |  |

Table 4. Transaction

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Description** |
| Key | String (Auto Generate) |  |
| Customer Key | String |  |
| Drinks | Array[Drink] |  |
| Type | String | Order&Pickup, Delivery |
| Total | Number |  |
| Store Name | String |  |
| CreatedAt | TimeStamp |  |
| Status | String | In process, Successful, Cancelled |

Table 5. Stores

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Description** |
| Store | String (Auto Generate) |  |
| Name | String |  |
| Address | String |  |
| Contact | String | PhoneNumber |
| Location | Latitude: Number |  |
| Longitude: Number |  |

Table 6. News

|  |  |  |
| --- | --- | --- |
| **Attribute** | **Type** | **Description** |
| Key | String (Auto Generate) |  |
| Title | String |  |
| Image | String |  |
| Content | String |  |

## Design

### UI Principles

* Consistency: By sticking to a unified color palette of green, orange and white, the app guarantees design consistency, improving the user experience. Users can easily navigate and interact with the app when its many sections have consistent user interface elements and behaviors (Nielsen, 1994)
* Simplicity: To cut down on clutter and let customers concentrate on important tasks like placing drink orders, the design takes a basic approach. Effective navigation and interaction are made possible by the interface's simplicity, cleanliness, and ease of understanding (Garrett, 2010).
* Feedback: Giving consumers instant feedback on their actions—for example, by changing the color of a button when they press it or by showing loading indicators—reassures them that their inputs are being processed, improving the user experience as a whole (Norman, 2013).
* Usability: To guarantee usability, the app makes use of big, readable fonts, lots of space, and buttons that are the right size. By using flexbox, the layout is made to adapt to the size of the screen, guaranteeing a responsive experience on a range of devices (Gerrard, 2020).
* Accessibility: The app uses bottom navigation to ensure users can easily control the app with one hand. This design principle helps to make the app more user-friendly and convenient, especially for users on the move (Nielsen, 1994).

### Navigation Flow

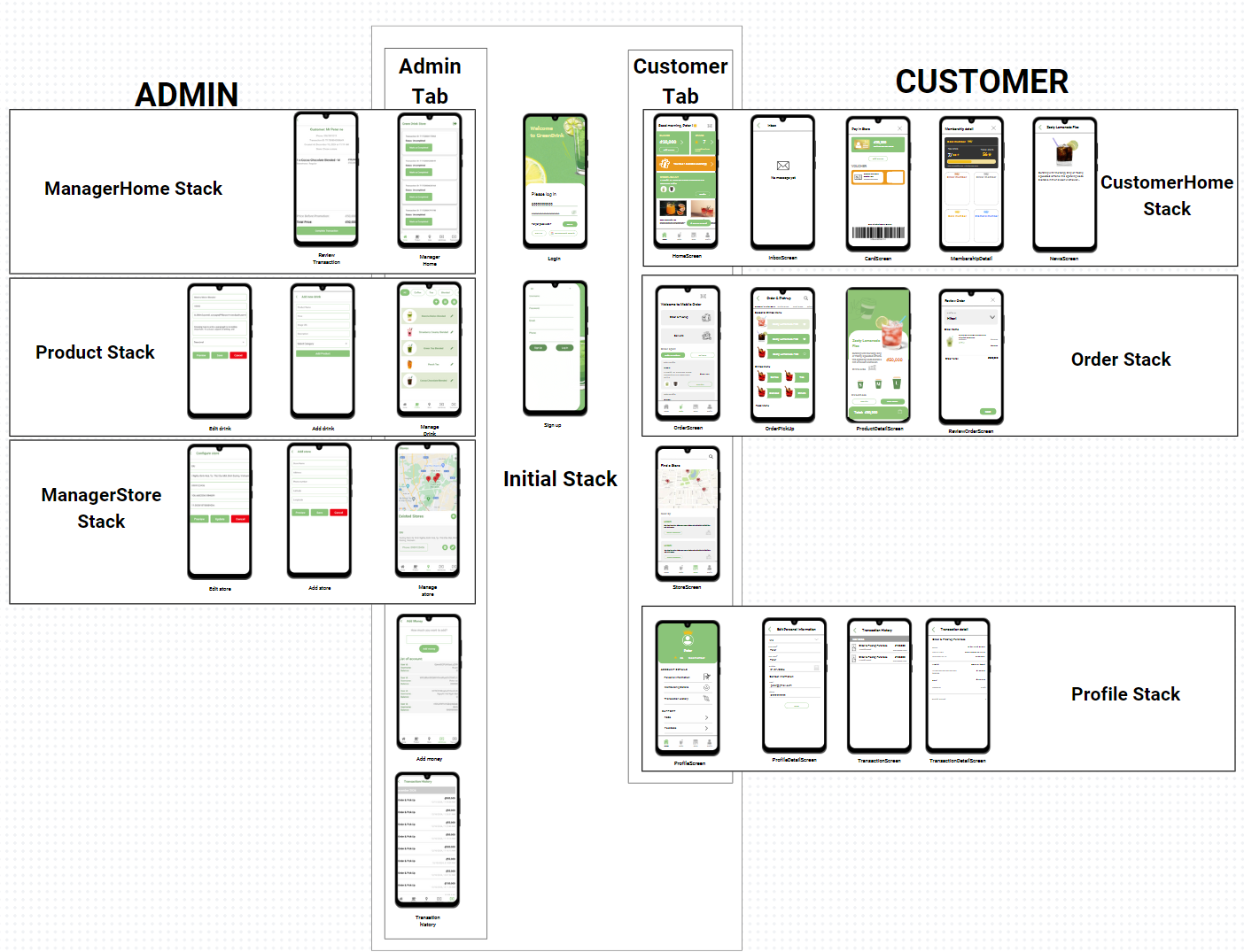


Figure 2. Navigation flow of Green Drink App (by authors)

### System Architecture

* Front-end:
* The component folder includes reusable UI components such as buttons, headers, and themed elements.
* The screen folder include all of the app view (e.g., LoginScreen, MenuScreen) that combines functions or logic and UI into a cohesive unit.
* Back-end:

Firestore is used by the app's backend to store data and manage user profiles, product information, and orders, among other crucial functions. User authentication is supported via custom logic, including Google Sign-In integration and email-password login. To guarantee maintainability and reusability, store.js modularizes shared utilities like database queries and local storage management. This design complies with the beverage business needs of the app and facilitates scalability.

# EXPERIMENT AND DISCUSSION

## Installation Environment

A range of technologies were used in the development of the Green Drink mobile app, all of which cooperate to guarantee responsiveness and seamless operation. An outline of the development environment, including the required tools and dependencies, is provided below.

### Development Environment and Dependencies

The app is developed with React Native, using TypeScript for type safety. It relies on several important libraries and dependencies to provide core functionality, navigation, authentication, and more.

Key dependencies:

* React Native: This framework is used to build cross-platform mobile applications. The version used is 0.75.4, which allows for efficient development on both Android and iOS platforms.
* React Navigation: Used for handling the navigation flow within the app, with libraries such as @react-navigation/native and @react-navigation/bottom-tabs installed.
* Firebase: Firebase services, such as authentication (@react-native-firebase/auth) and Firestore (@react-native-firebase/firestore), are utilized for backend services and real-time data management.
* Async Storage: For local storage management, @react-native-async-storage/async-storage is used.
* Other Libraries: The app also integrates with other components such as react-native-maps for map features, react-native-paper for UI elements, and react-native-gesture-handler for gesture handling.

### Development Tools and Dependencies

* TypeScript: Version 5.0.4 of TypeScript is used to ensure strong typing in the codebase.
* Babel and Metro Bundler: These are configured to support the latest JavaScript and React Native features.
* Java Version: For building and running the app, a Java Development Kit (JDK) version JDK 11 or higher is recommended for Android development. The Java version should be configured correctly to avoid issues during compilation and ensure compatibility with the Android build tools.

### Installation Commands

* Install Dependencies: “npm install …”

Ex: npm install @react-navigation/native

* Run on Android: react-native run-android
* Run on iOS: react-native run-ios
* Start the Metro Bundler: npm start

## Results and Discussion

The Green Drink Mobile App offers a smooth beverage ordering experience, thereby achieving its stated objectives. A cross-platform application with effective functionality and an intuitive user experience was made possible by the development process, which made use of React Native, TypeScript, and Firebase. An overview of the findings and important conversations is provided below:

### Results:

* Core Features:
* User Authentication: Both a manual login option and secure Google Sign-In login are supported by the app. For client profiles, Firebase Firestore guarantees dependable data management and storage.
* Order Management: Customers may explore and choose drinks with ease. Orders are handled and stored effectively, and their status is updated in real time.
* Rewards System: Customer engagement is improved via a reward system that monitors user behavior and offers rewards.
* UI Design:

The color scheme’s mix of orange, white, and green creates a lively and energizing look. A bottom navigation bar was created for one-handed use, and Flexbox was used to guarantee responsive layouts.

* Performance and Reliability:

Because of Firebase’s real-time database capabilities and React Native’s lightweight architecture, the app performs very well on both iOS and Android.

### Discussion:

* Technology Choices:
* React Native: was chosen for its ability to build cross-platform applications with a single codebase, significantly reducing development time. The framework offers near-native performance and a large ecosystem of pre-built components and libraries, such as React Navigation for seamless navigation and React Native Gesture Handler for enhanced user interaction. Its Flexbox-based layout system simplified UI design, ensuring the app is visually consistent across devices.
* TypeScript: added a robust layer of type safety, improving code reliability and maintainability. By catching errors during development, it reduced runtime bugs and facilitated better collaboration through clear definitions of data structures. This ensured that core functionalities, such as order management and user authentication, were implemented with precision.
* Firebase: Firebase Firestore provided a flexible, scalable NoSQL database solution that integrated well with React Native. Its real-time capabilities allowed for instantaneous updates to user order status and rewards. Firebase also simplified authentication through Google Sign-In and manual login, though manual adaptations were made for custom authentication flows.
* Challenges and Solutions:
* Authentication Flow: Integrating bespoke authentication with Google Sign-In while preserving Firestore data integrity was a major difficulty. Verifying credentials and conditionally directing users (e.g., admin vs. customer views) were part of the user authentication process. Robust state management and thorough error handling were necessary for this. By breaking down login functionality into reusable routines and carefully testing edge situations, the problem was solved.
* Navigation: There were difficulties in creating a navigation flow that could be used with one hand, particularly when it came to making sure it worked on different devices. The stack and tab navigators in React Navigation provided the freedom to efficiently organize the application. During authentication, dynamic routing and role-based logic were used to overcome issues with deep linking and conditional navigation depending on user roles (admin/customer).
* Performance Optimization: Even though Firebase manages real-time updates effectively, customers with slower networks had speed problems. Significantly lowering load times and increasing responsiveness were achieved by optimizing Firestore queries by indexing frequently used data and by lazy loading components.
* Scalability: Careful preparation was necessary to manage an expanding dataset of clients, orders, and awards. This was made easier by Firebase's scalability, but duplicate reads and writes were avoided by implementing bespoke logic. For instance, utilizing libraries like as AsyncStorage to cache user data locally decreased the frequency of Firestore requests.

# CONCLUSION AND FUTURE WORKS

## Conclusion

An important advancement in using technology to improve the beverage company's consumer experience is the Green Drink Mobile App. The application, which was created using React Native, TypeScript, and Firebase, has shown the value of cross-platform solutions that integrate scalability, performance, and user-centric design. The software effectively achieves its goals and establishes a framework for future development by attending to fundamental needs including order administration, client interaction, and efficient operations.

The project's successful integration of real-time data management via Firebase was one of its major accomplishments. Customers were given real-time information regarding their orders and loyalty points, and this allowed for responsive user experience in addition to fast order management. The adaptability of React Native reduced development costs and improved accessibility by enabling consistent performance on both the iOS and Android platforms. By lowering the possibility of mistakes and enhancing maintainability, TypeScript's static typing improved the codebase's dependability.

The significance of a modular and reusable architecture has also been emphasized by this project. The architecture of the application facilitates simple scalability and flexibility to future needs by employing common components and state management approaches. The design principles - focusing on simplicity, clarity, and accessibility - ensured that the user interface catered to a wide audience, offering a seamless navigation flow and intuitive interactions.

The difficulties faced along the growth process provided priceless teaching moments. To guarantee data security and a seamless user experience, for example, integrating third-party APIs for Google Sign-In necessitated extensive testing and debugging. Similarly, incremental changes and improvements were required to achieve cross - platform uniformity in UI design and functionality. Careful study, group problem-solving, and the use of contemporary development technologies were used to overcome these obstacles.

The Green Drink Mobile App's success demonstrates how technology can revolutionize small companies. The project has opened the door for innovation in the beverage sector by implementing scalable, dependable, and user-focused solutions. It also illustrates how digital transformation may improve customer happiness and provide competitive benefits.

In conclusion, the Green Drink Mobile App is a platform that can adapt to new demands in addition to providing a solution for present business needs. Future projects will benefit greatly from the expertise and information gathered throughout its development, which will promote an innovative culture and ongoing advancements in technology-driven business solutions.

## Future works

The ordering process for beverages has been effectively simplified by the Green Drink Mobile App. To improve both the user experience and the usefulness for businesses, the app may be improved in a number of ways. The three characteristics listed below are considered to be crucial for future advancements:

### Branch-Specific Functionality

Users may now place purchases and have them delivered or picked up in-store, but they are unable to track their items in real time. Each shop would be able to handle orders separately with the introduction of branch-specific features, offering a more customized experience. Customers would be able to monitor the progress of their orders, including whether they are being prepared, available for pickup, or on their way to be delivered, by designating specific panels for each branch inside the app. In addition to increasing user happiness, this functionality would bring the app into line with contemporary industry standards, where real-time order tracking is now expected.

### Branch-Specific Functionality

Stores may design, distribute, and oversee promotional vouchers or discount coupons right within the app with the help of a voucher management function. Such a system may provide store-specific discounts, loyalty benefits, or limited-time specials. This functionality would provide shop managers the freedom to execute marketing campaigns straight through the app, while also encouraging client retention and attracting new users. Given that digital voucher systems have been shown to boost engagement and sales, it may be a major boost for company expansion.

### Branch-Specific Functionality

To add money to their accounts, clients currently need to physically visit the business. This procedure would be made simpler in the future by integrating direct bank payment alternatives, which would enable consumers to add money straight from their bank accounts or through digital wallets. This modification will significantly improve the app's usefulness by allowing users to do transactions fully online without having to go to a real store. Since smooth digital payments are becoming more and more essential in contemporary mobile apps, such an integration would increase the app's convenience and competitiveness.

REFERENCES

1. Chen, S., & Lu, L. (2020). The impact of mobile app technology on customer satisfaction in the food and beverage industry. Journal of Hospitality and Tourism Technology, 11(2), 205-222
2. Statista. (2022). Online food delivery market size worldwide 2018-2027. Retrieved from<https://www.statista.com>
3. Kromeon. (2023). The impact of mobile apps on the food & beverage industry. Retrieved from<https://www.kromeon.com>
4. W2S Solutions. (2020). Why are food and beverage mobile apps getting a fabulous response?. Retrieved from<https://www.w2ssolutions.com>
5. Chowdhury, S., & Kazi, S. (2022). React Native framework for mobile application development: A comprehensive study. International Journal of Computer Science and Engineering, 10(3), 152-160.
6. Facebook. (2020). React Native: Learn once, write anywhere. Retrieved from<https://reactnative.dev>
7. Ramos, A. (2021). Cross-platform mobile development with React Native. O'Reilly Media.
8. React Native. (2020). Getting started with React Native. Retrieved from <https://reactnative.dev/docs/getting-started>
9. Chowdhury, S., & Kazi, S. (2022). Advanced TypeScript for Modern App Development. International Journal of Computer Science and Engineering, 10(3), 112-120.
10. Johnson, D., & Wilson, A. (2022). TypeScript in Practice: A Guide to Scalable Web Applications. Apress.
11. Microsoft. (2023). TypeScript Documentation. Retrieved from<https://www.typescriptlang.org/>
12. Couchbase. (2021). NoSQL vs. SQL: Understanding database evolution. Retrieved from <https://www.couchbase.com/resources>
13. Google. (2023). Firebase Documentation. Retrieved from https://firebase.google.com/docs
14. Ramos, A. (2022). NoSQL databases for modern app development. O'Reilly Media.
15. Ambler, S. W. (n.d.). *Use case diagrams*. Agile Modeling. Retrieved December 11, 2024, from <https://agilemodeling.com/artifacts/usecasediagram.htm>
16. Garrett, J. J. (2010). The Elements of User Experience: User-Centered Design for the Web and Beyond. Pearson Education.
17. Gerrard, S. (2020). Practical Flexbox: Building Responsive User Interfaces in React Native. Packt Publishing.
18. Nielsen, J. (1994). Usability Engineering. Academic Press.
19. Norman, D. A. (2013). The Design of Everyday Things: Revised and Expanded Edition. Basic Books.