

**ANKARA UNIVERSITY  
FACULTY OF ENGINEERING  
DEPARTMENT OF COMPUTER ENGINEERING**



**COM4061 Research Techniques**

**Smart-Shopping**

**Ibraheem R N Hamdan**

**20290947**

**Ribhi Bishtawi**

**20290078**

**Prof. ENVER BAĞCI**

**January 2024**

## ABSTRACT

This project is a cross-platform application built with Flutter, seamlessly integrates with Firebase and NodeJS as its backend powerhouse. Designed as a solution for store owners, this app redefines the retail industry by simplifying store management, creating engaging offers, and ensuring secure authentication. Users, on the other hand, benefit from an opportunity of smart shopping and real-time tourism experience, discovering nearby stores, exclusive deals, and local attractions. Our project doesn't just streamline store management; it creates an immersive experience where users explore, shop, and engage.

## GITHUB REPOSITORY

<https://github.com/ribhy-bishtawi/Graduation-Project>

# Contents

<b>1. INTRODUCTION .....</b>	<b>1</b>
<b>2. PROJECT DESCRIPTION AND SETUP .....</b>	<b>2</b>
2.1 <i>Description .....</i>	2
2.2 <i>Setup .....</i>	2
<b>3. AUTHENTICATION .....</b>	<b>3</b>
<b>4. TECHNOLOGY .....</b>	<b>5</b>
4.1 <i>Backend .....</i>	5
4.1.1     Node.js .....	5
4.1.2     Prisma for Database Migration .....	5
4.1.3     API Integration .....	6
4.2 <i>Frontend .....</i>	7
4.2.1     Flutter.....	7
<b>5. APP PAGES AND MAIN FUNCTIONALITIES .....</b>	<b>8</b>
5.1 <i>Shops .....</i>	8
5.2 <i>Offers .....</i>	9
5.3 <i>Notifications .....</i>	9
5.4 <i>Account.....</i>	10
<b>6. APP WORKFLOWS OVERVIEW .....</b>	<b>11</b>
6.1 <i>Shop Creation Process .....</i>	11
6.2 <i>Notification System Overview.....</i>	14
6.3 <i>Offers and Offer Creation .....</i>	17
<b>7. CONCLUSION .....</b>	<b>19</b>
<b>8. REFERENCES .....</b>	<b>20</b>

## **1. INTRODUCTION**

In the changing world of shopping and tourism, businesses face challenges in keeping up with consumers and effectively managing the operations of their establishments. Smart-Shopping, a user-friendly mobile application available on Android and iOS, was created to tackle this challenge and help business owners increase their presence by using technology. It offers a simpler way of operating, flawlessly integrating store management, engaging users, and providing real-time exploration of potential nearby attractions.

For businesses, Smart-Shopping serves as a tool for enhancing customer engagement and boosting online visibility. Its features allow store owners to efficiently manage their businesses and build a closer connection with their clients. For users the app enhances the overall shopping experience and provides users with exclusive deals and offers from different stores. The app uses multiple technologies to achieve this goal, including Flutter for ensuring a smooth and responsive user interface, Firebase for OTP, and an efficient backend powered by Node.js and MySQL.

## **2. PROJECT DESCRIPTION AND SETUP**

### **2.1 Description**

Smart-Shopping, a pioneering mobile application, redefines the paradigms of retail and tourism. Downloadable on Google Play and the App Store, the app seamlessly merges store management and user engagement, providing a curated blend of smart shopping and real-time tourism exploration. This transformative project leverages technologies such as Flutter and Node.js, prioritizing a multilingual interface for inclusivity. The objective is clear: simplify store management, redefine the shopping experience, and facilitate dynamic tourism exploration.

### **2.2 Setup**

The technological foundation of Smart-Shopping is a meticulously crafted ensemble of cutting-edge tools, ensuring efficiency, responsiveness, and a seamless user experience. Flutter, a versatile UI toolkit developed by Google, forms the mobile app's interface, offering a unified codebase compatible with both iOS and Android. This is complemented by the robust MySQL database management system, serving as the backbone for data storage with its reliability and scalability features.

For backend development, Node.js takes center stage, renowned for its efficiency in building scalable and real-time applications. Firebase Authentication plays a pivotal role in securing the application through One-Time Password (OTP) authentication, aligning with Smart-Shopping's commitment to adopting state-of-the-art technologies for user authentication. This comprehensive technological setup underscores our commitment to delivering a sophisticated, user-friendly, and secure Smart-Shopping experience to our users.

### **3. AUTHENTICATION**

The authentication process in the Smart-Shopping mobile app was constructed by Firebase, providing a strong and secure framework for user verification. The procedure starts on the app's first page, serving as the key to personalized experiences. Users encounter a login interface featuring a text-field for their phone number, a button to initiate login, and an option to sign up.

After entering a phone number and initiating login, Firebase Authentication sends a One-Time Password (OTP) via SMS, enhancing the login process with an additional layer of security. The user is then redirected to an OTP entry page, where they input the received code. Upon pressing login, Firebase Authentication validates the OTP, granting access if the verification was successful.

Choosing to sign up redirects customers to a thorough registration page with text fields for username and phone number, guaranteeing a smooth onboarding process. . A select field offers the options of Male or Female for gender selection. There are two alternatives available to the user: sign up and login. In order to guarantee a smooth transition between authentication phases, a login button reroutes to the original login page.

Firebase Authentication's integration highlights the commitment to data security and user verification within Smart-Shopping. Using Firebase not only streamlines the authentication process but also ensures that user data is handled with the extreme care and security. This robust authentication mechanism contributes to a trustworthy, secure, and seamless Smart-Shopping experience.

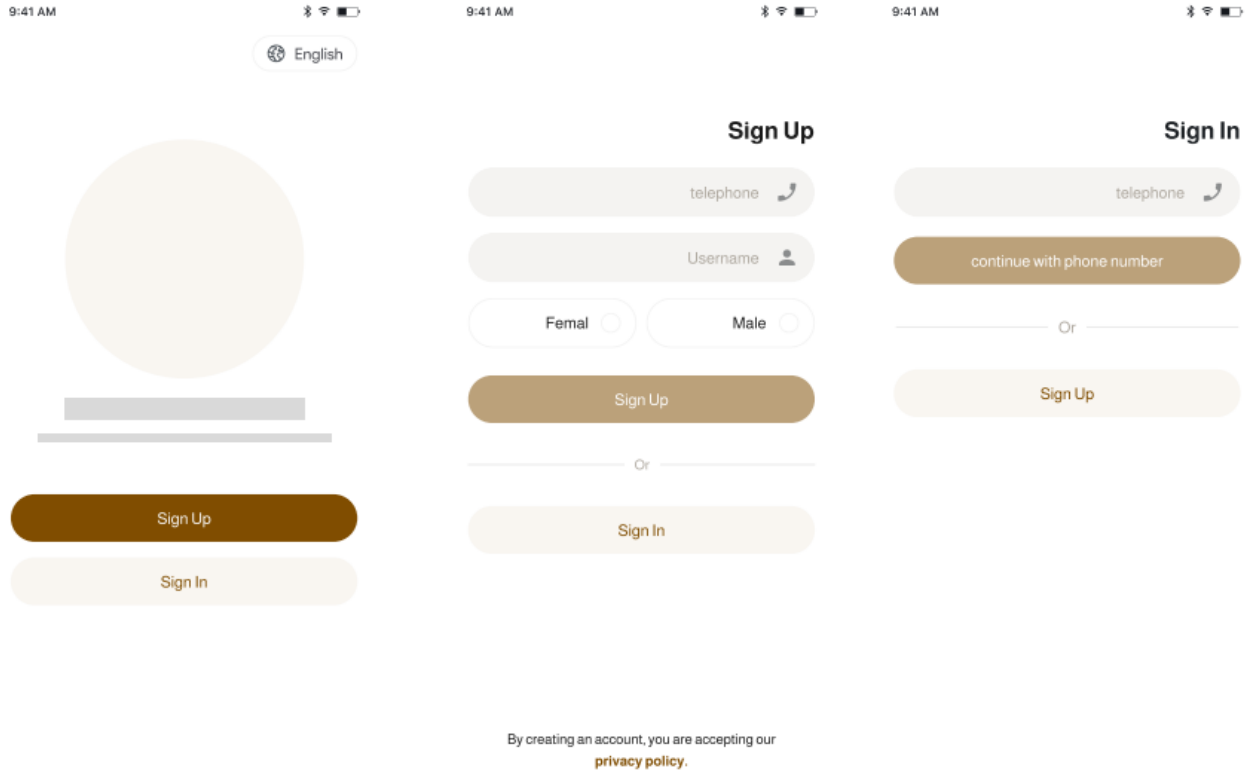


Figure 4.1 Authentication Process

## 4. TECHNOLOGY

### 4.1 Backend

The dynamic backend of Smart-Shopping's mobile application is a result of thorough development using Node.js, supported by Prisma for database migration, and enhanced with API integration.

#### 4.1.1 **Node.js**

Smart-Shopping's backend is powered by Node.js, a versatile and efficient runtime built on the V8 JavaScript engine. Node.js excels in building scalable and real-time applications. In the context of Smart-Shopping, Node.js takes center stage as the server-side technology, managing critical functionalities such as API endpoints, user authentication, and seamless integration with the MySQL database.

Node.js is an excellent option for applications that require responsiveness and real-time interactions because of its ability to manage connections. Smart-Shopping's backend achieves optimal performance by using Node.js's event-driven architecture, guaranteeing an effortless and effective user experience.

#### 4.1.2 **Prisma for Database Migration**

Prisma plays an essential role in Smart-Shopping's backend architecture, specifically focusing on the crucial process of database migration. Database migration ensures the continuous evolution of the database schema, allowing for structural changes without losing existing data integrity.

Prisma is an advanced database toolkit that makes database migrations and interactions easier. By introducing a type-safe database access layer, it reduces the possibility of runtime errors and increases developer productivity. Prisma's capabilities are employed by Smart-Shopping to enable database schema changes that are both efficient and version-controlled.

The data modeling syntax of Prisma easily aligns with the data requirements of Smart-Shopping, allowing for predictable and controllable database migration procedures. Prisma's integration shows Smart-Shopping's ability to maintain a scalable and organized database, which is necessary for the long-term sustainability of the application.



### 4.1.3 API Integration

To improve the functionality of the application, Smart-Shopping carefully integrates an API to complement the backend architecture. By acting as a pathway for smooth communication between multiple components, the API enables Smart-Shopping to dynamically access external services or data sources.

API integration expands the capabilities of Smart-Shopping, providing access to external data or services that enrich the user experience. Whether retrieving real-time information about store reviews, accessing the latest tourism data, or incorporating dynamic content, API integration amplifies the application's versatility.

Because of the effective implementation of APIs, Smart-Shopping is able to stay flexible and responsive to the changing retail and tourism industries. The effort of Smart-Shopping to providing a dynamic and feature-rich user experience is demonstrated by this strategic integration.

In addition to providing a solid basis for current features, Smart-Shopping's backend's efficient integration of Node.js, Prisma, and API integration sets the application up for future growth and improvement. This selection of backend technologies, which adds to the overall efficiency, and adaptability of Smart-Shopping's backend infrastructure, demonstrates the commitment to the best development.

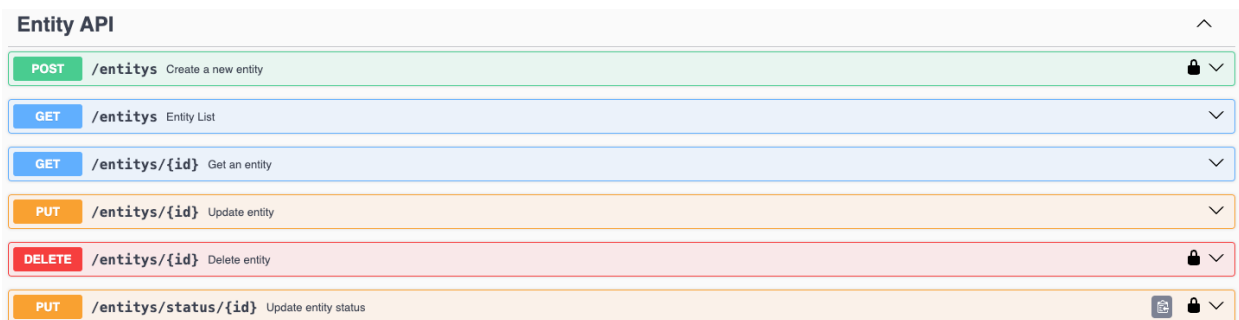


Figure 4.1.3.1 Documentation for the API using Swagger

## **4.2      Frontend**

The frontend of Smart-Shopping's mobile application is created with Flutter, a versatile and open-source UI software development toolkit.

### **4.2.1      Flutter**

Flutter excels in creating visually appealing and responsive user interfaces across multiple platforms, delivering a seamless and consistent experience to Smart-Shopping's users. With its widget-based architecture, Flutter simplifies the development process, allowing for the creation of a unified and engaging frontend.

Smart-Shopping's choice of Flutter for the frontend is driven by its ability to streamline the development of visually rich interfaces and its support for hot reload, enabling rapid iteration and efficient debugging. Flutter's cross-platform nature ensures that Smart-Shopping's mobile application maintains a consistent look and feel across various devices, including both Android and iOS.

The adoption of Flutter in the frontend aligns with Smart-Shopping's commitment to providing a modern and user-friendly interface for its customers. The combination of Flutter on the frontend and the robust backend technologies, including Node.js, Prisma, and API integration, underscores Smart-Shopping's dedication to delivering a dynamic, feature-rich, and scalable mobile application that anticipates and adapts to the evolving needs of the retail and tourism industries.

## 5. APP PAGES AND MAIN FUNCTIONALITIES

After successful authentication, the business owner is directed to the main page, which serves as the hub for managing shops, offers, notifications, and account settings. The user interface is designed with a bottom navigation bar providing access to key functionalities.

### 5.1 Shops

- Shop Management Overview:
  - The "Shops" page provides an overview of all the shops managed from the business owner's account.
  - Each shop entry displays basic information, such as the shop name and location.
  - The list is dynamically updated as the business owner adds or removes shops.
- Add New Shop
  - The business owner can easily add a new shop by pressing the "Add New Shop" button.
  - A form appears for the user to input details such as the shop name, category, location, and contact information.
  - Upon submission, the new shop is integrated into the list of shops on the user's account.
- Edit and Delete Shops
  - For existing shops, the business owner can initiate edits to update information or make corrections.
  - A straightforward editing interface allows modifications to multiple fields.
  - Deletion of a shop is facilitated through a secure confirmation process, preventing accidental removal.

## 5.2 Offers

- Create and Manage Offers:
  - The "Offers" page is the control center for creating and managing promotions and offers.
  - The “Add Offer” button, used for the addition of new offers, generates an offer creation form.
  - Each existing offer is showcased with details such as the offer name, duration, and current status (Running or Expired).
- Offer Details
  - Tapping on a specific offer provides details, including the offer description and relevant dates.
  - The interface allows quick modifications for ongoing offers, while expired offers are removed.

## 5.3 Notifications

- Notification Overview:
  - The "Notifications" page keeps a history of notifications sent by the business owner to customers.
  - Each entry displays the sender's name and message details.
- Sending Notifications:
  - The business owner can compose and send notifications directly from the interface using the “Add Notification” button.
  - Notifications undergo an approval process by administrators before being sent to the application users.

## 5.4 Account

- Account Management:
  - The "Account" page enables the business owner to manage many settings including personal details and language settings.
  - A user-friendly interface allows for a simple editing process to the user's profile information.
- Application Settings:
  - Business owners can customize application settings, including language preferences for a tailored experience.
  - Options for signing out or deleting the account are also included.

This design of the application flow prioritizes simplicity, providing business owners with an efficient platform to manage their shops, create offers, send notifications, and personalize account settings. The bottom navigation bar ensures seamless navigation between these core functionalities, ensuring a simple, user-friendly experience.

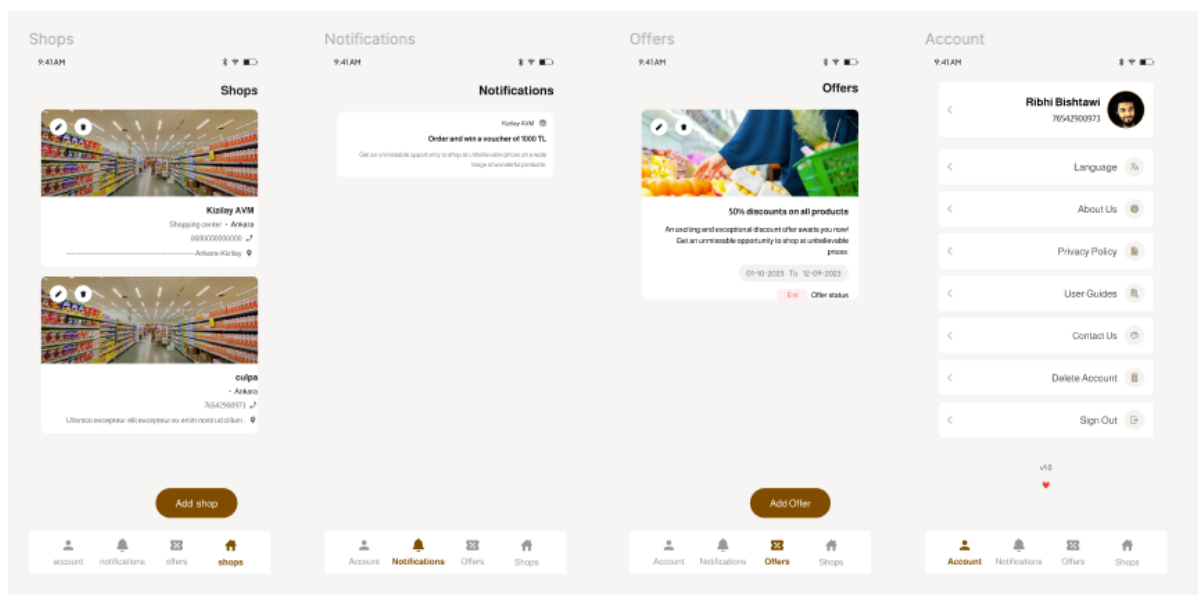


Figure 5.1 Main pages of the application

## 6. APP WORKFLOWS OVERVIEW

### 6.1 Shop Creation Process

The Smart-Shopping application offers a straightforward and thorough shop creation process that enables business owners to quickly enter essential details about their establishments. Here is a detailed explanation of the fields and steps involved:

#### I. Initiating Shop Creation:

- After navigating to the "Shops" screen, the business owner taps the "Add New Shop" button, triggering the shop creation process.
- This action redirects the user to a dedicated screen designed for entering the information of the new shop.

#### II. Data Entry Fields:

The shop creation form includes the following fields, for basic information about the establishment:

- Store Name (in Turkish and English):  
The business owner inputs the name of the establishment in both Turkish and English.
- Categories:  
Users can select one or more categories that best represent the type of services offered by the establishment.
- Keywords (Tags):  
Limited selection of keywords or tags allows the business owner to associate specific attributes with the shop.
- Commercial Registration Number:  
The commercial registration number is a unique identifier that helps establish the legal identity of the shop.
- City:  
The city field allows the business owner to specify which city the shop is located in.

- **Working Days/Hours:**  
Input fields for working days and hours provide information on the schedule of the shop.
- **Location:**  
The exact location of the shop is specified, to provide accurate and easy navigation.
- **Addresses for Multiple Branches:**  
If there are multiple branches, the business owner can provide addresses for each branch, enabling a complete overview.
- **Social Media Links:**  
Facebook, Instagram, and TikTok links offer an opportunity for enhanced online visibility and engagement.
- **Contact Information:**  
The business owner includes the contact number and the name of the person to contact for inquiries.
- **Store Image:**  
An image of the store is uploaded to create a visual appeal, contributing to an attractive and informative shop profile.

### III. Submission and Confirmation:

After completing the form, the business owner submits the information.

### IV. Validation and Approval:

The entered information undergoes validation to ensure accuracy and compliance with any specified guidelines.

#### V. Shop Integration:

- Once approved, the new shop integrates into the list of managed shops on the "Shops" screen.
- Users can immediately access and manage the newly added shop, enhancing the set of features offered by the Smart-Shopping application.

By structuring the shop creation process with a user-friendly form and wide-ranging fields, Smart-Shopping enables business owners to efficiently showcase their establishments within the application.

The figure displays five mobile application screens for shop management:

- Add - shop:** A form for creating a new shop. It includes sections for Shop information (Shop name in Turkish, Shop name in English, Commercial number, Branch name, Categories, City, Time), Contact information (Telephone number, Contact name), Social Media (Facebook page, Instagram page, TikTok page), Tag words, and Shop Image. A "Save" button is at the bottom.
- Cities:** A screen for selecting a city. It features a search bar and a list of cities: Ankara, Istanbul, Mersin, Adana, Izmir, Antalya, Kayseri, and Hatay.
- Categories:** A screen for selecting a category. It features a search bar and a list of categories: Restaurant, Clothes Shop, Mail, and Fast Food.
- Edit- shop:** A form for editing an existing shop. It includes sections for Shop information, Contact information, Social Media, Tag words, and Shop Image. A "Save" button is at the bottom.
- Time:** A screen for setting working hours. It includes a "Working Time" section with a table for days of the week and time slots. A "Save" button is at the bottom.
- Maps:** A screen for selecting a location. It features a map of Florence with a location pin and a list of nearby landmarks: San Marco, Galleria dell'Accademia, Palazzo Medici Riccardi, Santa Maria del Fiore, Palazzo Vecchio, Palazzo Davanzati, Ponte Vecchio, and Palazzo Vecchio.

Figure 6.1.1 Add new shop interface.



## **6.2      Notification System Overview**

Through the addition of a strong notification system, the Smart-Shopping application enables business owners to interact with customers in an efficient and effective way. This system makes sure that crucial announcements, promotions, and updates are delivered within a regulated and efficient process. Here is a quick overview of the notification process:

- I.    Initiation:  
Business owners create messages within a specific character limit on the "Notifications" page, to announce promotions, special deals, or relevant information.
- II.   Sender's Information:  
The sender's name is included, providing users with clear information about the notification.
- III.   Submission to Administrator:  
Messages are submitted for administrator review after pressing the "Send Notification" button, ensuring content aligns with the specified guidelines of the application.
- IV.   Administrator Approval:  
The administrator reviews and approves messages, acting as a quality control measure to prevent spam or inappropriate content from being sent to the customers.
- V.    Delivery to Users:  
Approved notifications are delivered to application users, ensuring controlled communication.

## Key Features:

### I. Efficiency and Transparency:

This process, combined with transparent sender information, guarantees efficient communication.

### II. Character Limit:

The character limit encourages clear and precise communication with the customer, optimizing the user's experience.

### III. Quality Control:

The administrator's approval step adds a layer of security, ensuring that notifications follow the application's standards.

In summary, the notification system in Smart-Shopping enhances user engagement, helps efficient communication, and maintains a standard of quality through the approval process. This feature contributes to a positive user experience within the Smart-Shopping community.

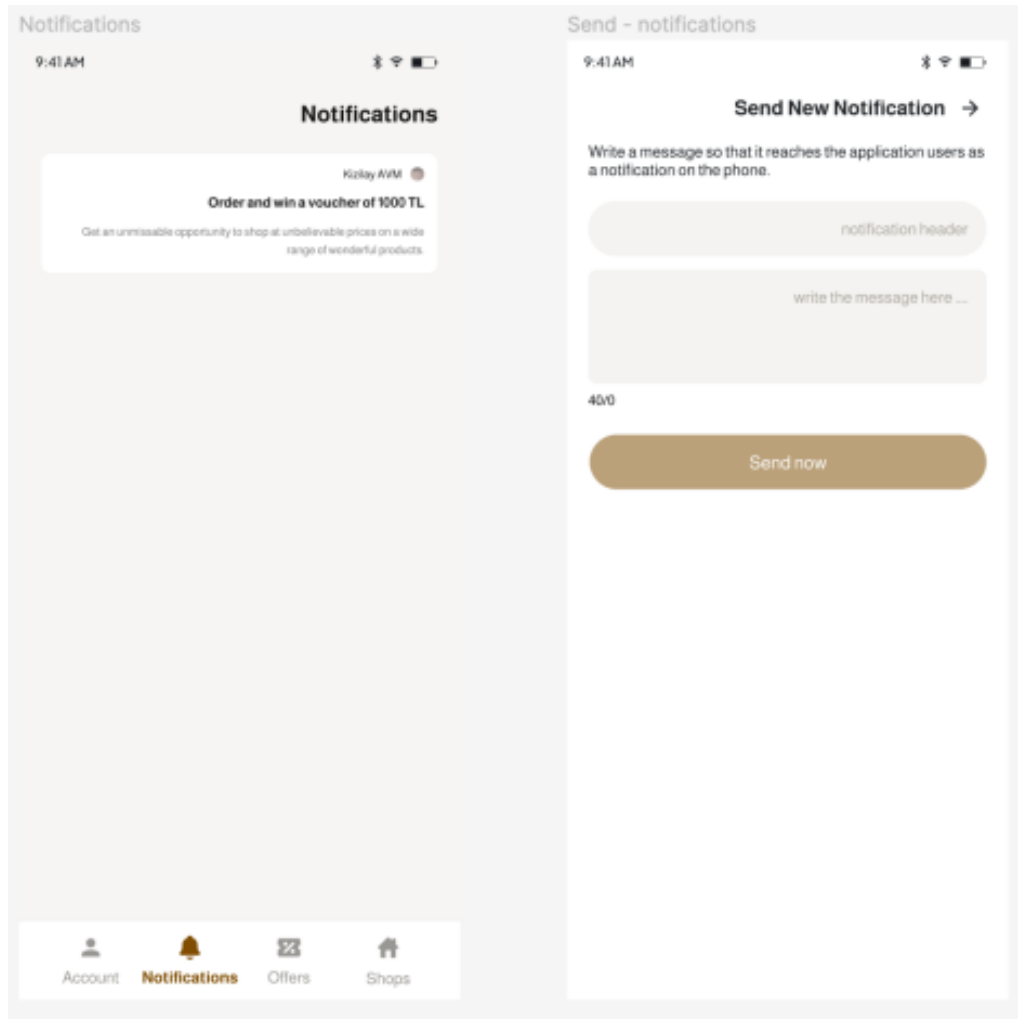


Figure 6.2.1 Notification and Send New Notification interface

### **6.3      Offers and Offer Creation**

Creating a new offer on the Smart-Shopping application is an easy and straightforward process designed for maximum simplicity. Here's a step-by-step explanation on how offer is created:

- I.    Initiation:  
Tap "Add Offer" on the Offers Page to start crafting a new promotion.
  
- II.   Offer Input Page:  
User is redirected to a dedicated page for entering offer details.
  
- III.   Essential Details:  
Input offer name, upload an image, and provide a description with the details of the offer.
  
- IV.   Time Frame:  
Set the offer start and end dates of the offer.
  
- V.    Confirmation:  
Confirm details and submit the offer for it to be displayed on the application.

## Key Advantages:

- I. User-Friendly:  
Simplified and easy steps.
- II. Visual Appeal:  
Boost attractiveness with an image for the offer.
- III. Time-Sensitivity:  
Announce start and end dates of the offer.

This process ensures that businesses can efficiently create and showcase captivating offers, promoting engagement and boosting their presence on the application.

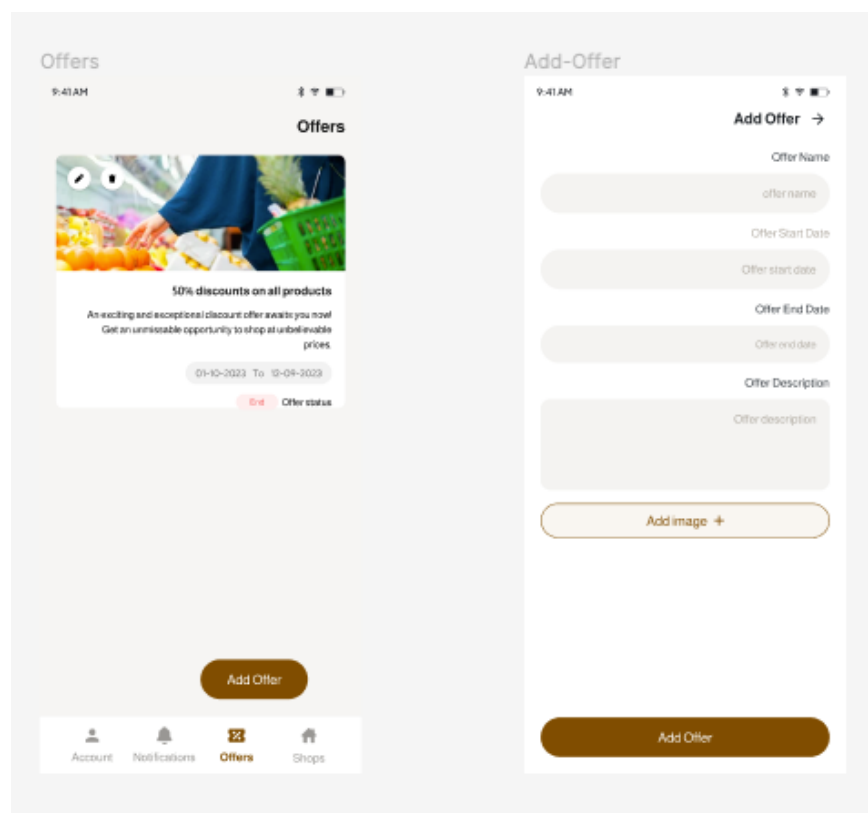


Figure 6.3.1 Offers and Add New Offer interface

## 7. CONCLUSION

In conclusion, Smart-Shopping is a mobile app that helps both shop owners and consumers in an efficient way. The seamless integration of Flutter on the frontend ensures the creation of a visually appealing and responsive interface. The backend, powered by Node.js, Prisma, and API integration is working together to make sure everything runs smoothly.

The authentication process, powered by Firebase, enhances user verification and data security. The app's functionalities, carefully outlined in the report, provide business owners with efficient tools for shop management, offer creation, and communication through a notification system. The Smart-Shopping application, with its user-friendly interface and features, represents a simpler approach to retail and tourism. The careful selection and integration of technologies demonstrate a commitment to delivering a sophisticated, secure, and seamless experience.

The app not only makes shopping easier now but also prepares for more potential developments in the future, making Smart-Shopping an app for modern shopping. In our future plans, we are considering enhancing the app for an even better user experience. We aim to add additional features such as ATMs and museums to provide users with a comprehensive guide when visiting new locations. With everything conveniently accessible in one app, Smart-Shopping will become an incredibly useful tool for everyone.

## 8. REFERENCES

- [Coding With T]. (2022, February 11). Flutter Firebase Phone Number OTP Authentication - Flutter Firebase 2023 [Video]. YouTube.  
[https://www.youtube.com/watch?v=dGwr66EkJKk&ab\\_channel=CodingWithT](https://www.youtube.com/watch?v=dGwr66EkJKk&ab_channel=CodingWithT)
- Firebase. (n.d.). Flutter. <https://docs.flutter.dev/development/data-and-backend/firebase>
- Flutter - Build apps for any screen. (n.d.). <https://docs.flutter.dev/docs>
- Node.js. (n.d.). Node.js Documentation. <https://nodejs.org/en/docs/>
- Prisma. (n.d.). Prisma Documentation. <https://www.prisma.io/docs/>
- [Rivaan Ranawat]. (2022, October 19). Flutter, Firebase & Riverpod Master Class - Build a COMPLETE Reddit Clone App [Video]. YouTube.  
[https://www.youtube.com/watch?v=B8Sx7wGiY-s&ab\\_channel=RivaanRanawat](https://www.youtube.com/watch?v=B8Sx7wGiY-s&ab_channel=RivaanRanawat)
- Swagger. (n.d.). Swagger Documentation.  
<https://swagger.io/docs/specification/about/>
- TypeScript. (n.d.). TypeScript Documentation.  
<https://www.typescriptlang.org/docs/>