

## 6. PROJECT RESULT:

### 6.1 FOODBOX: APPLICATION SCREENSHOTS

#### 6.1.1 Users App Screenshots

Splash Screen



Login Screen



➤ **Splash Screen:**

This is the first screen when the users open the app. It can provide visual feedback to the user, indicating that the application is loading or starting up.

➤ **Login Screen:**

The primary purpose of a login screen is to authenticate the user's identity before granting access to the application. In this activity user should login to his/her account using registered email and password. This helps ensure that only authorized individuals can access the application and its associated data.

### Forgot Password



### Sign-in Screen



➤ **Forgot Password:**

In this activity, if the user forgot his/her password then the user can reset the password through email. A reset link is sent to user's email.

➤ **Sign-in Screen:**

In this activity, user should register into the application to gain access to personalized features. Registering into the application requires some fields such as name, email, phone no. and password.

**Home Screen**



**Select City**



➤ **Home Screen:**

Home screen is the central interface of the application that users encounter after logging in or launching the application. It provides access to various features and content within the application such as Dhaba & Tiffin options.

➤ **Select City:**

When the users select the Dhaba or Tiffin option they will be directed to the food fragment where the users can select desired city from a provided set of options. The screen displays a list of available cities for selection.

### Select Dhaba



### Select Dhaba Food Items



#### ➤ Select Dhaba:

When the users select the desired location/city then a list of dhaba will be displayed. This screen allows users to browse and select their desired dhaba for further actions, such as placing an order.

#### ➤ Select Dhaba Food Items:

After Selecting the desired dhaba of their choice by the users, a list of food items get displayed provided by that dhaba. This screen allows users to browse and select specific food items from the dhaba's menu for their order.

### Select Quantity



### Dhaba Cart Screen



#### ➤ Select Quantity:

This screen allows users to input or select the desired quantity of the food items before proceeding with an action, such as adding items to a cart, placing an order. Users can click on the plus button to increase the quantity of the food or the minus button to decrease it. After selecting the quantity users can add it to cart and then select the view cart option to view the cart.

#### ➤ Dhaba Cart Screen:

This screen displays a list of the selected food items, including their names, quantities, individual prices before the user proceeds with the order. This cart screen also includes the subtotal, which represents the total cost of the selected food items.

**Select Tiffin Menu Screen**



**Tiffin Cart Screen**



➤ **Select Tiffin Menu Screen:**

Users can also select the menu from the Tiffin Service providers. This screen allows users to choose their desired tiffin menu or meal plan from a list of available options such as breakfast, lunch and dinner. After selecting the menu users can add it to the cart and then select the view cart option to view the cart.

➤ **Tiffin Cart Screen:**

This screen display a list of selected menu or meal plan including their names, quantities, individual prices before the user proceeds with the order. This cart screen also includes the subtotal, which represents the total cost of the selected food items.

**Fill User Details Screen**



**Payment Screen**



➤ **Fill User Details Screen:**

This screen appears when the user select the place order button in the cart section. In this activity user needs to fill details such as Name, Phone no, Pincode, House no, Area/Colony to place the order.

➤ **Payment Screen:**

This screen allows users to select a payment method, and complete the transaction. Users should select the COD payment method to place the order.

### Order Placed Screen



### Order Details Screen



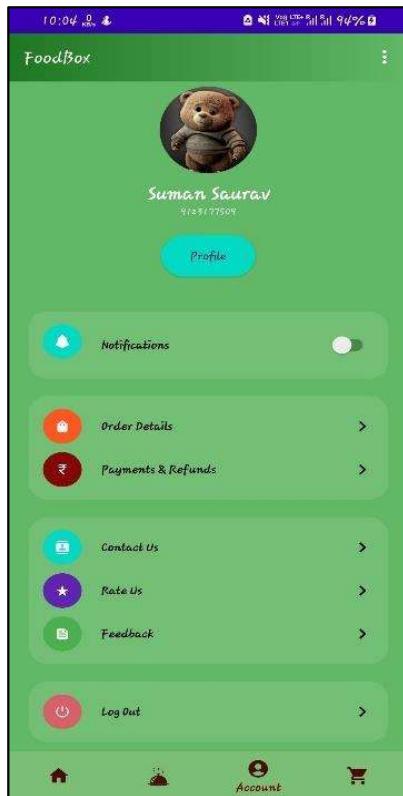
#### ➤ Order Placed Screen:

The screen displays a message that confirms the successful placement of the order. It includes text such as "Order Placed Successfully".

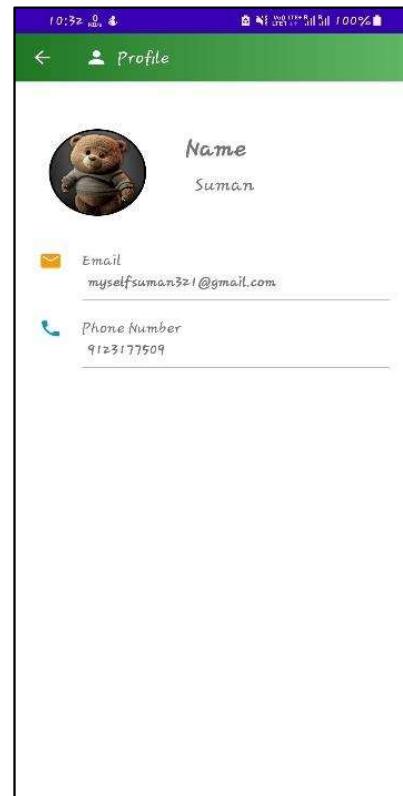
#### ➤ Order Details Screen:

This screen allows users to view and review specific details related to their order. It displays a summary of the order, including the order id, date and time of the order. It displays the shipping address associated with the order and also indicates the current status of the order such as accepted, delivered or canceled.

**Account Screen**



**Profile Screen**



➤ **Account Screen:**

This screen serves as a central hub for users to manage their account details and customize their experience. It displays the various options such as profile, notifications, order details, payment method, contact us, rate us, feedback and logout.

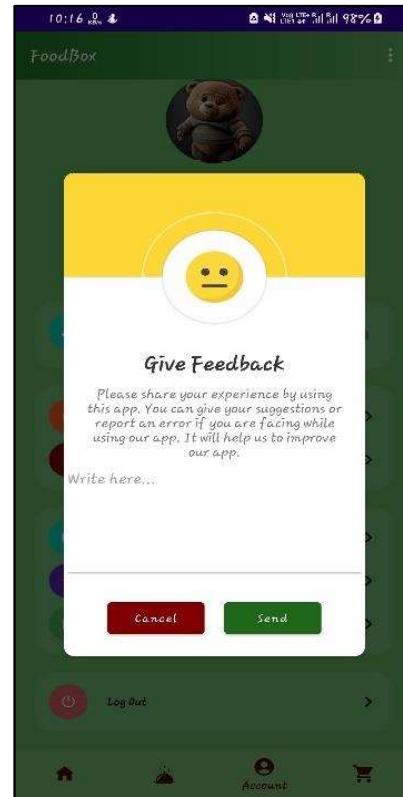
➤ **Profile Screen:**

This screen provides users with an overview of their personal details. This screen displays the user's profile picture and also display essential personal information, such as the user's full name, email, and phone number.

**Contact us Screen**



**Feedback Screen**



➤ **Contact us Screen:**

This screen allows users to reach out for assistance, ask questions, or report issues. It displays the contact information of the support or developer team. This typically includes a phone number and email address.

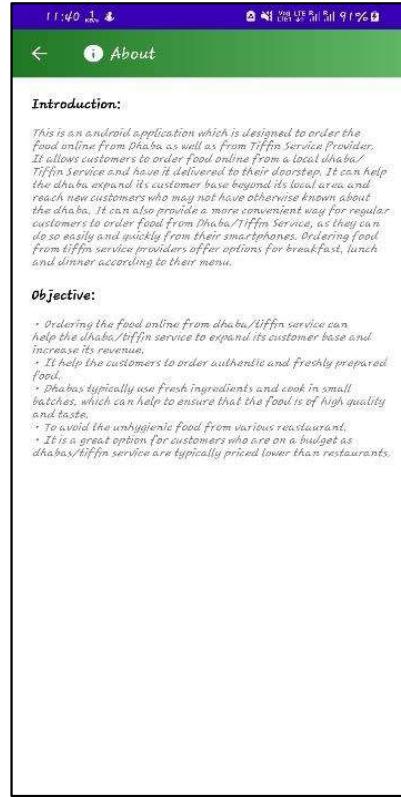
➤ **Feedback Screen:**

This screen provides users with an opportunity to provide feedback, suggestions, or report any issues they have encountered while using the app. It will help us to improve our app.

### Toolbar Menu



### About Screen



#### ➤ Toolbar Menu:

The toolbar menu provides navigation options that allow users to use various options within the application. This can include options such as Share and About.

#### ➤ About Screen:

This screen provides a brief description or overview of the application.

### 6.1.2 Dhaba/Tiffin Service App Screenshots

**Login Screen**



**Forgot Password**



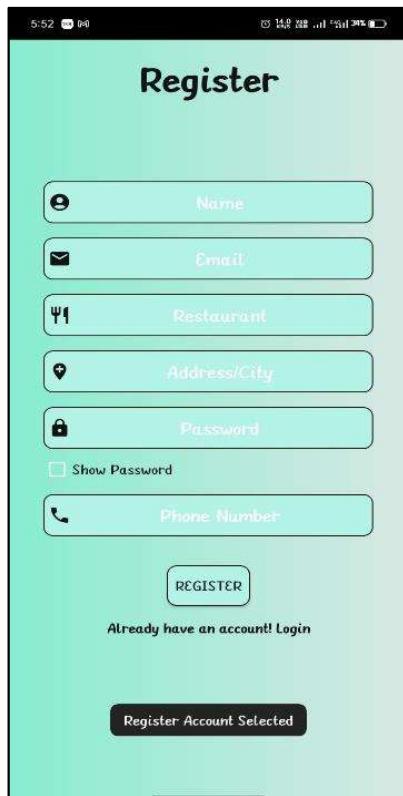
➤ **Login Screen:**

In this activity, dhaba/tiffin service management should login using the registered email and password. This helps ensure that only authorized dhaba/tiffin service management can access the application and its associated data.

➤ **Forgot Password:**

This activity allows dhaba/tiffin service owner to reset the password through email and regain access to their accounts. A reset link sent to the dhaba/tiffin owner.

**Sign-in Screen**



**Home Screen**



➤ **Sign-in Screen:**

In this activity, dhaba/tiffin owner should register into the application to gain access to personalized features. Registering into the application requires some fields such as name, email, dhaba/tiffin service name, address/city, phone no. and password.

➤ **Home Screen:**

Home screen is the central interface of the application that dhaba/tiffin service management encounter after logging in or launching the application. It provides access to various features and content within the application such as Dhaba & Tiffin options to manage their food items/menu.

**Dhaba Menu Screen**



**Add Food Details Screen**



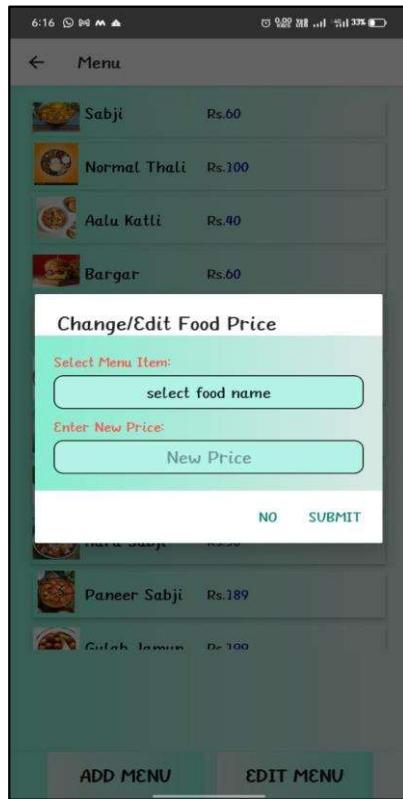
➤ **Dhaba Menu Screen:**

This screen displays a list of food items which is added by the dhaba management to display the available food items to users. Each food item is typically represented with its name, description, price and possibly with an image. This information helps users to plan their orders and make budget-conscious decisions.

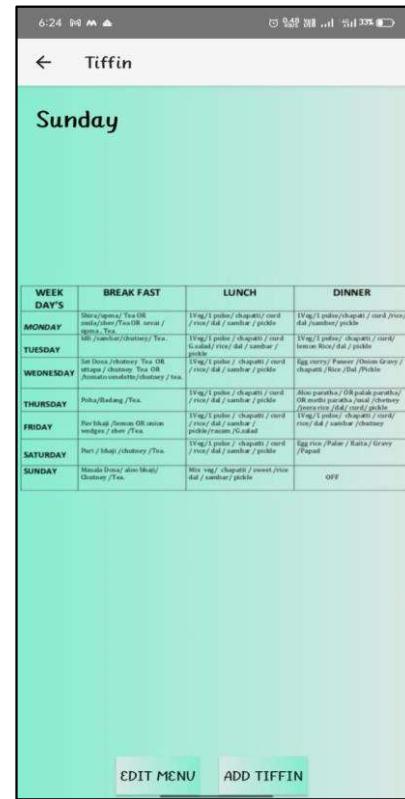
➤ **Add Food Details Screen:**

This screen enables dhaba owners or manager, to input essential information about the food item for proper display and order management. It includes fields where dhaba owner can enter the name and price of the food item and also add image of the food. This name and image will be used to identify the dish in the menu or order list.

## **Change/Edit Price Screen**



## Tiffin Menu Screen



#### ➤ Change/Edit Price Screen:

This screen allows the dhaba owners or manager to change/edit the price of the food items based on their business needs. It provides an input field where dhaba owner can enter the new price for the food item.

#### ➤ Tiffin Menu Screen:

This screen displays a list of tiffin or meal options offered by the tiffin owners or manager such as breakfast, lunch and dinner. Tiffin owner can also add another menu if the previous menu is no longer offered.

### Updatae Tiffin Price Screen



### Order Details Screen



#### ➤ Update Tiffin Price Screen:

This screen allow the tiffin owner or manager to update the price of the meal such as breakfast, lunch and dinner based on their business needs. It provides an input field where tiffin owner can enter the new price for the meal.

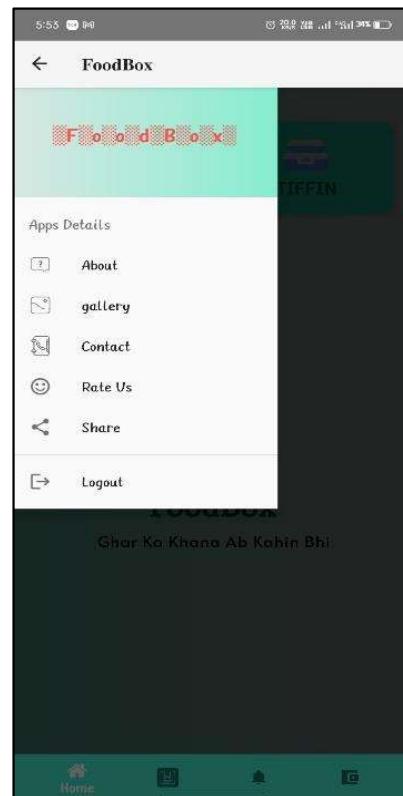
#### ➤ Order Details Screen:

The screen provides dhaba/tiffin service owner with essential information about the order. This information helps dhaba/tiffin service staff to plan and coordinate the order delivery process.. It includes the customer's name, phone number, and list of food items which is ordered by the user.

**Account Profile Screen**



**Navigation Drawer**



➤ **Account Profie Screen:**

This screen provides dhaba/tiffin owner with an overview of their personal details. This screen displays the dhaba's or tiffin's owner profile picture and also display essential personal information, such as the dhaba's or tiffin's owner full name, dhaba/tiffin service name, email, phone number and address/city.

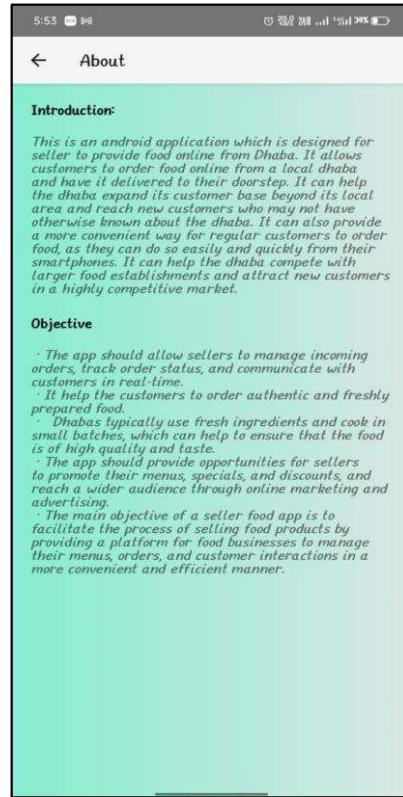
➤ **Navigation Drawer:**

The navigation drawer allows dhaba/tiffin owner to access different sections or features of the application. It displays a list of menu items or navigation options. These options can include sections such as about, gallery, contact, rate us, share and logout.

### Help Contacts Screen



### About Screen



#### ➤ Help Contacts Screen:

This screen allows dhaba/tiffin owner to reach out for assistance, ask questions, or report issues. It displays the contact information of the support or developer team. This typically includes a phone number and email address.

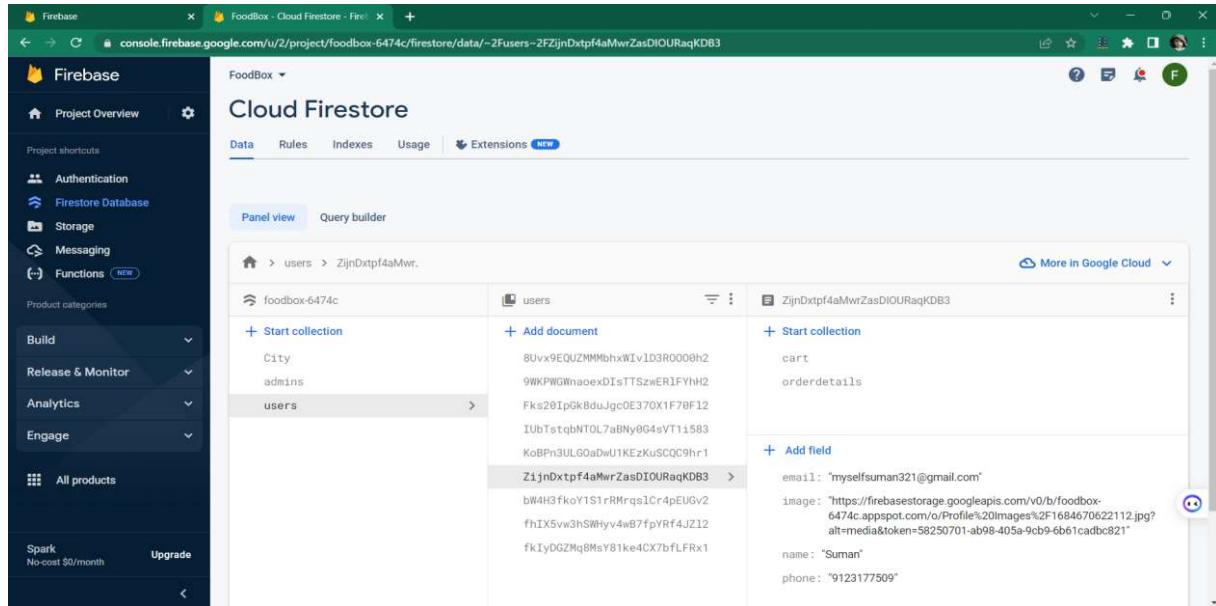
#### ➤ About Screen:

This screen provides a brief description or overview of the dhaba/tiffin service application.

## 6.2 FIREBASE STRUCTURE SCREENSHOTS:

### 6.2.1 Cloud Firestore

#### a) Users

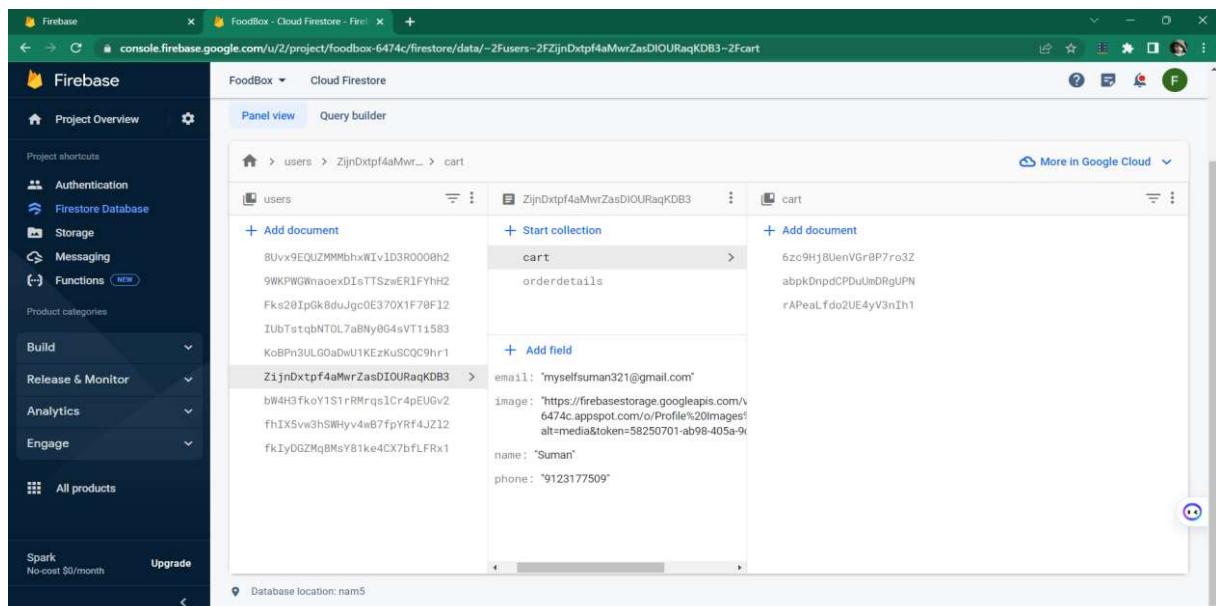


The screenshot shows the Firebase Cloud Firestore interface. On the left, the project navigation bar includes 'Project Overview', 'Authentication', 'Firestore Database', 'Storage', 'Messaging', 'Functions', 'Analytics', 'Engage', and 'All products'. Under 'Build', it shows 'Spark No-cost \$0/month' and 'Upgrade'. The main area is titled 'Cloud Firestore' with tabs for 'Data', 'Rules', 'Indexes', 'Usage', and 'Extensions'. It shows a 'Panel view' of the 'users' collection under the document 'ZijnDxtpf4aMwrZasDIORaqKDB3'. The collection contains documents with fields like 'email', 'image', 'name', and 'phone'.

Document ID	Fields
ZijnDxtpf4aMwrZasDIORaqKDB3	email: "myselfsuman321@gmail.com" image: "https://firebasestorage.googleapis.com/v0/o/Profile%20Images%2F1684670622112.jpg?alt=media&token=58250701-ab98-405a-9cb9-6b61cadbc821" name: "Suman" phone: "9123177509"

Users collection refers to a collection of documents that store information about the users of the application. Each document within the collection represents a specific user and contains fields such as name, email, phone no. and image.

#### b) Cart



The screenshot shows the Firebase Cloud Firestore interface. The left sidebar is identical to the previous screenshot. The main area shows the 'users' collection with the document 'ZijnDxtpf4aMwrZasDIORaqKDB3'. A new 'cart' collection is added under this document. The 'cart' collection contains documents with fields like 'email', 'image', 'name', and 'phone'.

Document ID	Fields
ZijnDxtpf4aMwrZasDIORaqKDB3	email: "myselfsuman321@gmail.com" image: "https://firebasestorage.googleapis.com/v0/o/Profile%20Images%2F1684670622112.jpg?alt=media&token=58250701-ab98-405a-9cb9-6b61cadbc821" name: "Suman" phone: "9123177509"

Cart collection refers to a collection of cart documents that store information about items added to a user's cart. Each cart document represents a specific user's cart and contains details such as the food items, quantities, prices.

### c) Cart(Dhaba View)

The screenshot shows the Firebase Cloud Firestore interface. On the left, the navigation bar includes 'Project Overview', 'Authentication', 'Firestore Database', 'Storage', 'Messaging', 'Functions', 'Analytics', 'Engage', and 'All products'. The main area displays the 'FoodBox - Cloud Firestore' database. A path 'users > ZjnDxtpf4aMwrZasDIOURaqKDB3 > cart > yZH0bJmNuC8kaG6f14M' is shown. This document contains fields: email, image, name, phone, food\_name ('Panner Sabji'), food\_price (189), hotelId ('0Dc1dKFKRlyzqaNlpxmG6bdrC2'), quantity (1), rest\_name ('GoFood'), total\_price (189), and value ('cart').

When the Cart Screen is loaded, the application retrieves the user's cart data from the Firebase Firestore which is added by the users. This data includes information about the items in the cart from dhaba menu such as food items, quantities, prices, and any other relevant details.

### d) Cart(Tiffin View)

The screenshot shows the Firebase Cloud Firestore interface. The navigation bar and database structure are identical to the previous screenshot. The path 'users > ZjnDxtpf4aMwrZasDIOURaqKDB3 > cart > qTG8jOZBuwyH3WlxuSF' is shown. This document contains fields: email, image, name, phone, food\_name ('Lunch'), food\_price (60), hotelId ('0Dc1dKFKRlyzqaNlpxmG6bdrC2'), quantity (1), rest\_name ('GoFood'), total\_price (60), and value ('cart').

When the Cart Screen is loaded, the application retrieves the user's cart data from the Firebase Firestore which is added by the users. This data includes information about the items in the cart from tiffin menu such as breakfast, lunch and dinner, quantities, prices, and any other relevant details.

### e) Order Details

The screenshot shows the Firebase Cloud Firestore interface. On the left sidebar, under the 'Project Overview' section, the 'Cloud Firestore' option is selected. In the main panel, the path 'users > ZijnDxtpf4aMwrZasDIOURaqKDB3 > orderdetails' is shown. A specific document, 'UbrwmTPdS18b9n2uKoDW', is selected and expanded. The document contains the following fields:

- email: "myselfsuman321@gmail.com"
- image: "https://firebasestorage.googleapis.com/6474c.appspot.com/o/Profile%20Images?alt=media&tken=58250701-ab98-405a-9e3f-103a2a233333"
- name: "Suman"
- phone: "9123177509"
- UbrwmTPdS18b9n2uKoDW (document ID)
- area: "mahendergarh"
- cust\_id: "0EBF60DE4F164C0FACFD6BD31C2FAE22"
- cust\_name: "Suman"
- flat\_no: "15"
- hotelId: "0xDc1dKFKYRlyzqaNlpxmG6bdrc2"
- order: "Panner Masala Sabji 1 x 100 = 100 Roti 1 x 5 = 5"
- order\_id: "7D7874A67539469CB1C16FD6980BBA1D"
- pay\_method: "COD(Cash On Delivery)"
- phone\_no: "9876543210"
- pin\_no: "123029"
- rest\_name: "GoFood"
- status: "Delivered"

This screenshot is identical to the one above, showing the same document 'UbrwmTPdS18b9n2uKoDW' from the 'orderdetails' collection. The document fields and their values are the same as described in the first screenshot.

Order details collection refers to a collection of documents that store information about the details of customer orders. Each document within the collection represents a specific order and contains various fields to capture relevant order details. This data includes information such as the order ID, customer ID, customer name, total cost, delivery address, payment method, phone no dhaba name and other relevant details associated with the order. It displays the order details which is ordered from the dhaba menu. It also displays the current status of the order, indicating whether it is pending, accepted, or delivered.

### f) Order Details(Tiffin View)

The screenshot shows the Firebase Cloud Firestore interface. On the left, the navigation bar includes 'Project Overview', 'Authentication', 'Firestore Database', 'Storage', 'Messaging', 'Functions', 'Analytics', 'Engage', and 'All products'. The main area displays the 'orderdetails' collection under the 'users' document for user 'ZijnDxtpf4aMwrZasDIOURaqKDB3'. The document contains fields such as 'email', 'image', 'name', 'phone', and a list of order items. The right panel shows the detailed structure of the 'orderdetails' document.

In order details tiffin view firestore user's application fetches the specific order data from Firestore based on a unique order identifier or reference. This data include various information similar to dhaba order details. It display the order details which is ordered from the tiffin service menu such as breakfast, lunch and dinner.

### g) Admins(Dhaba/Tiffin Service)

The screenshot shows the Firebase Cloud Firestore interface. On the left, the navigation bar includes 'Project Overview', 'Authentication', 'Firestore Database', 'Storage', 'Messaging', 'Functions', 'Analytics', 'Engage', and 'All products'. The main area displays the 'admins' collection under the 'foodbox-6474c' document. The collection contains documents for 'City' and 'users'. One user document is expanded, showing fields like 'Address', 'AdminImage', 'Restaurant', 'admin\_name', 'admin\_phone', and 'email'.

Admins collection refers to a collection of documents that store information about dhaba/tiffin service owner. It displays relevant dhaba/tiffin service owner data stored in Cloud Firestore, such as profile information, order details, dhaba menu, tiffin menu or other specific data.

## h) Menu

The screenshot shows the Firebase Cloud Firestore interface. On the left sidebar, under the 'FoodBox' project, the 'Firestore Database' section is selected. In the main panel, the 'Menu' collection is displayed. A specific document, 'eEDn8fw13Qwhd8pEXB4T', is selected and expanded. This document contains fields for 'image', 'menu\_name' (set to 'Paner Sabji'), and 'price' (set to 189). Other documents in the collection have IDs starting with '01MUWZ2pKQkdE4QaWmZn' and so on.

Menu collection refers to a collection of documents that store information about the menu items offered by a dhaba. This includes information about the available food items, such as their names, prices, images. This allows users to view and interact with the menu of a Dhaba food delivery application using Firestore as the backend database

## i) Tiffin

The screenshot shows the Firebase Cloud Firestore interface. Under the 'FoodBox' project, the 'Firestore Database' section is selected. The 'Tiffin' collection is shown, which contains documents for 'Breakfast', 'Dinner', and 'Lunch'. Each document includes a field 'Tiffin' with its respective description and a field 'Image' with a URL pointing to a Firebase storage image. For example, the 'Breakfast' document has an image URL starting with 'https://firebasestorage.googleapis.com/v0/b/foodbox-6474c.appspot.com/o/Admin%20Tiffin%20Menu%2F1684673639710.jpg'.

Tiffin collection refers to a collection of documents that store information about the menu items offered by the tiffin service. It include information about tiffin meal names such as breakfast, lunch and dinner , price and images. This menu accessed by the users to order food.

### j) Order Details

The screenshot shows the Firebase Cloud Firestore interface. On the left, the navigation bar includes 'Project Overview', 'Authentication', 'Firestore Database', 'Storage', 'Messaging', 'Functions', 'Analytics', 'Engage', 'All products', 'Spark', and 'Upgrade'. The main area displays the 'FoodBox - Cloud Firestore' database. A path 'admins > 0vDc1dKFKYRlyzqaNlpxmG6bdrC2 > orderdetails' is selected. The 'orderdetails' collection contains several documents, each representing an order. One document is expanded to show fields such as 'Address: "Haryana"', 'AdminImage: "https://firebasestorage.googleapis.com/o/Admin..."', 'Restaurant: "GoFood"', 'admin\_name: "Gaurav Sharma"', 'admin\_phone: "8709658958"', and 'email: "gofood@gmail.com"'. Other documents in the list have similar structures.

The order done by the users is stored in order details collection. It display the order details which is ordered from the dhaba/tiffin service menu. In order details collection firestore, dhaba/tiffin service application fetches the specific order data from Firestore to deliver the food items.

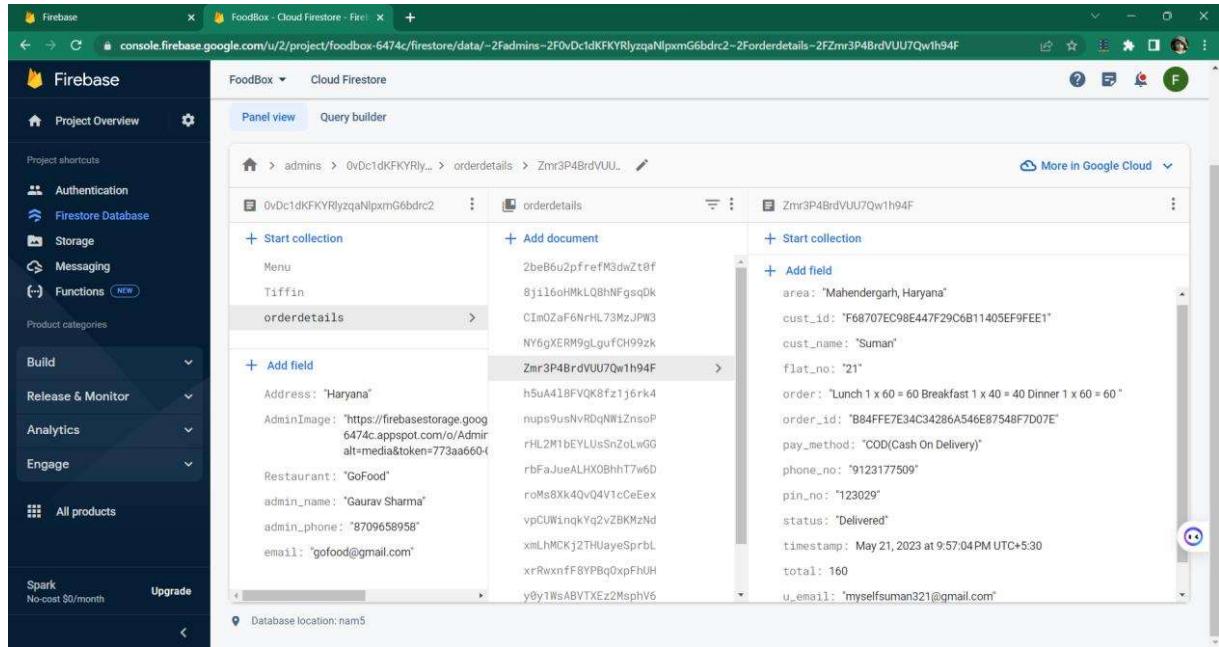
### k) Dhaba View

The screenshot shows the Firebase Cloud Firestore interface, similar to the previous one but focusing on a specific document. The path 'admins > 0vDc1dKFKYRlyzqaNlpxmG6bdrC2 > orderdetails > nups9usNvRDqNWiznsnP' is selected. The document 'nups9usNvRDqNWiznsnP' is expanded to show detailed information. Fields include 'area: "Central University of Haryana, Mahendergarh"', 'cust\_id: "ADBF93C59C8D0495B8D5BE0115A3841D9"', 'cust\_name: "Suman Saurav"', 'flat\_no: "013"', 'order: "Roti 1 x 5 = 5 Normal Thali 1 x 100 = 100 Paneer Sabji 1 x 189 = 189 Sabji 1 x 60 = 60"', 'order\_id: "SECBA402327134E91ACF78C7893F8D277"', 'pay\_method: "COD(Cash On Delivery)"', 'phone\_no: "9123177509"', 'pin\_no: "123029"', 'status: "Cancelled"', 'timestamp: May 21, 2023 at 7:13:17 PM UTC+5:30', and 'total: 354'.

It shows the order details of the food items which is ordered by the users from dhaba menu collection. It display the data with various information regarding the orders.

## FoodBox: Food Ordering From Dhaba & Tiffin Service

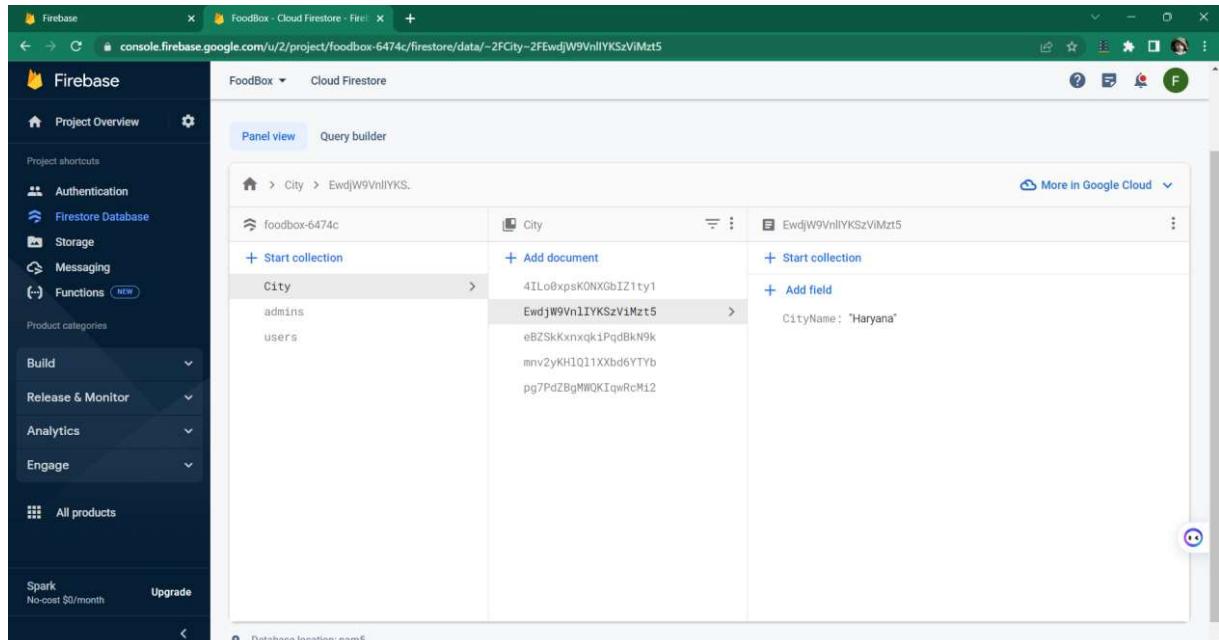
### l) Tiffin View



The screenshot shows the Firebase Cloud Firestore interface. On the left, the navigation bar includes 'Project Overview', 'Authentication', 'Firestore Database', 'Storage', 'Messaging', 'Functions', 'Analytics', 'Engage', and 'All products'. The main area displays the 'orderdetails' collection under the '0vDc1dKFKYRlyzqaNlpxmG6bdr...'. A specific document, 'Zmr3P4BrdVUU7Qw1h94F', is selected, showing its fields: area ('Mahendergarh, Haryana'), cust\_id ('F68707EC98E447F29C6B11A05EF9FEE1'), cust\_name ('Suman'), flat\_no ('21'), order ('Lunch 1 x 60 = 60 Breakfast 1 x 40 = 40 Dinner 1 x 60 = 60'), order\_id ('B84FFE7E34C34286A546E87548F7D07E'), pay\_method ('COD(Cash On Delivery)'), phone\_no ('9123177509'), pin\_no ('123029'), status ('Delivered'), timestamp ('May 21, 2023 at 9:57:04 PM UTC+5:30'), total ('160'), and u\_email ('myselfsuman321@gmail.com').

It shows the order details of the food items which is ordered by the users from tiffin menu collection. It also displays the data with various information regarding the orders.

### m) City



The screenshot shows the Firebase Cloud Firestore interface. The navigation bar is identical to the previous one. The main area displays the 'City' collection under the 'foodbox-6474c' database. A specific document, 'EwdjW9Vn1IYKSzViMzt5', is selected, showing its field: CityName ('Haryana').

In Firestore, here city collection refers to a group of city documents stored within a Firestore database. Each city document represents a specific city. The Firestore city collection can be accessed, queried, and manipulated using Firestore's API or SDKs, allowing applications to retrieve city data.

### 6.2.2 Firebase Storage

The screenshot shows the Firebase Storage interface for the 'FoodBox' project. On the left, there's a sidebar with various project settings like Authentication, Firestore Database, Storage, and Functions. The main area is titled 'Storage' and shows a list of files and folders under the URL `gs://foodbox-6474c.appspot.com`. The list includes:

Name	Type	Last modified
Admin Profile Image/	Folder	—
Admin Tiffin Menu/	Folder	—
FoodBox/	Folder	—
Menu Images/	Folder	—
Profile Images/	Folder	—

At the top right, there are buttons for 'Upload file' and other file management options.

Firebase Storage allows users and dhaba/tiffin service owner to securely store and serve user-generated content, such as images. Firebase Storage provides built-in security rules that allow developers to control access to stored files. Security rules can be defined based on user authentication ensuring that only authorized users can upload, access, or modify files.

### 6.2.3 Firebase Authentication

The screenshot shows the Firebase Authentication interface for the 'FoodBox' project. The left sidebar has the same navigation as the storage screen. The main area is titled 'Authentication' and shows a table of users:

Identifier	Providers	Created	Signed in	User UID
sumansaurav550@gmail.c...	[Email icon]	May 21, 2023	May 21, 2023	Fks20IpGk8duJgcOE37OX1F70Fl2
baitak4u@gmail.com	[Email icon]	May 21, 2023	May 21, 2023	WxAbAQK70pYNvau9TREYKieqBj03
kajukatri@gmail.com	[Email icon]	May 21, 2023	May 21, 2023	5VjmzbUBMMTMGMIhTuNwFeGaT...
ashokahotel@gmail.com	[Email icon]	May 21, 2023	May 21, 2023	03gDeimJZ2PhJhqSMtrQglmxnjg1
nandu4u@hotmail.com	[Email icon]	May 21, 2023	May 21, 2023	o1Mbr932rCZDibmavVqgGmddIRO...
suchismita321@gmail.com	[Email icon]	May 21, 2023	May 21, 2023	gne967JGIEPRWgWWdse5tLjdPk1
raushan321@hotmail.com	[Email icon]	May 21, 2023	May 21, 2023	uDRsjkhXIRPgLq2T8YNQqP7EXor2
sameer123@gmail.com	[Email icon]	May 21, 2023	May 21, 2023	cjcSOl2uLeOd6KG4h46bkIRXjQ82
rahul@gmail.com	[Email icon]	May 21, 2023	May 21, 2023	Vau6zj0KXevGb8bcE2qShFVUjql
gaurav@hotmail.com	[Email icon]	May 21, 2023	May 21, 2023	fklyDGZIMq@MsY81ke4CX7bfLFRx1

Firebase Authentication is a service provided by Firebase that allows developers to easily integrate user authentication and authorization into their applications. It offers various registration methods, including email/password, phone number etc.

## 7. CONCLUSION:

- We studied the efficient use of developing **FoodBox** app an android application and making the great use of it.
- We focused on the problems faced by the users who can't afford ordering food from large restaurants from various food ordering app and developed this android application to overcome the cause affecting people.
- Users can also order the food from tiffin service provider who provide options for breakfast, lunch and dinner according to their menu.
- We use these results to determine the usage of this application is very easy and everyone can make use of it by ordering their desired food at any selected location at any time.
- Finally based on our research, we proposed an approach to find the possibilities of reaching the users with their ordered food, which account for the expected data transfer time over the available duration for the order to be placed.
- Main objective to develop the app was to get user an healthy & fresh food at very reasonable price as compared to restaurants.