

DASTARKHAN

Restaurant Mobile Application

Software Requirements Specification (SRS)

Version 1.0

February 21, 2026

Submitted by

Hamid Saleem

Registration No: 9061

Course: Mobile Application Development (MAD)

Instructor: Sadaf Tanvir

Technology Stack: React Native + Expo (Frontend MVP)

Section 1 – Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document provides a thorough description of the functional and non-functional requirements for the Pakistani Restaurant Mobile Application MVP. It serves as the authoritative reference for designers, developers, testers, and stakeholders throughout the software development lifecycle, defining system behaviors, user interactions, data structures, and quality attributes that the application must satisfy.

The application is a mobile-first restaurant management and ordering platform built with React Native and Expo. It targets two distinct user groups: customers who wish to browse menus, place orders, and make table reservations; and restaurant managers who oversee daily operations including order processing, menu management, and reservation handling.

1.2 Scope

The functionalities listed below fall within scope for the initial MVP release.

Customer-Facing Features

- User registration and authentication with secure login and logout capability
- Home screen presenting featured daily specials and quick-access category navigation
- Full menu browsing with category-based filtering and real-time search
- Detailed item view supporting quantity selection and custom special instructions
- Shopping cart management including adding, removing, and updating item quantities
- Checkout flow with order-type selection (Dine In or Takeaway), table assignment, and order summary with tax calculation
- Order status tracking across the full lifecycle: Placed, Preparing, Ready, and Served
- Order history accessible at any time for review of past transactions
- Table reservation booking by selecting date, time, and number of guests
- Reservation management including viewing and cancelling existing bookings
- User profile management for maintaining personal account information

Manager-Facing Features

- Manager dashboard with live statistics covering active orders, revenue, reservations, and menu item count
- Order management with step-by-step status progression through the order lifecycle
- Detailed order view displaying customer information, itemized breakdown, and financial summary
- Full menu management: adding, editing, deleting items, and toggling availability
- Reservation management enabling confirmation or cancellation of table bookings
- Secure logout from the manager panel

Out of Scope (Planned for Future Releases)

- Online payment gateway integration

- Push notifications and real-time WebSocket updates
- Multi-restaurant support
- GPS-based delivery tracking
- Customer loyalty and rewards program
- Backend API server and cloud database (MVP uses local mock data)
- Admin analytics and reporting dashboards

1.3 Definitions and Acronyms

Term / Acronym	Definition
SRS	Software Requirements Specification A document that describes what a system should do
MVP	Minimum Viable Product the first functional release delivering only core features
React Native	An open-source JavaScript framework for building cross-platform mobile applications from a single codebase
Expo	A platform and toolset built on top of React Native, enabling streamlined development and testing
NoSQL	A non-relational database approach using document-based storage (e.g., MongoDB, Firestore)
CRUD	Create, Read, Update, Delete — the four fundamental data operations
UI	User Interface the visual and interactive layer that a user directly engages with
UX	User Experience the overall quality, ease, and satisfaction of using the application
Dine In	An order type in which the customer consumes food at the restaurant premises
Takeaway	An order type in which the customer collects the food and consumes it off-site
AsyncStorage	A local key-value storage mechanism in React Native used to persist data on the device between sessions
Context API	A React feature for sharing global state across components without requiring prop drilling
Navigation Stack	A navigation pattern where screens are layered on top of one another in a push-and-pop manner
Bottom Tab Navigator	A navigation pattern that renders a persistent tab bar at the screen bottom for primary section switching
PKR / Rs.	Pakistani Rupee the currency used for all monetary values displayed within the application

Section 2 – Overall Description

2.1 User Roles

Role	Key Permissions & Goals
Customer	Browse the restaurant menu by category; search for items; view detailed item information including description, price, and image; add items to the cart with quantity and special instructions; manage cart contents; place orders (Dine In or Takeaway); track order status in real time; view order history; book table reservations by selecting date, time, and number of guests; view and cancel existing reservations; manage personal profile including name, phone, and email; log in and log out securely.
Restaurant Manager	View a live operations dashboard summarizing active orders, today's revenue, upcoming reservations, and total menu items; manage incoming orders by advancing their status through the order lifecycle; view full order details; manage the menu by adding new items, editing existing ones, deleting discontinued dishes, and toggling item availability; manage table reservations by confirming or cancelling bookings; log out of the manager panel securely.

2.2 User Stories

Customer User Stories

1. As a customer, I want to register an account with my name, email, password, and phone number so that I can access the restaurant's services.
2. As a customer, I want to log in with my email and password so that I can access my personalized experience, cart, and order history.
3. As a customer, I want to see a Home screen displaying today's specials and featured items so that I can quickly discover popular dishes without browsing the full menu.
4. As a customer, I want to browse the complete menu organized by categories (Starters, Main Course, Breads, Desserts, Drinks) so that I can easily locate the type of food I want.
5. As a customer, I want to search for menu items by name so that I can quickly find a specific dish.
6. As a customer, I want to tap a menu item to view its full description, price, and image, then select a quantity and add special instructions (e.g., no onions) before adding it to my cart.
7. As a customer, I want to view my cart showing all selected items with their names, quantities, individual prices, and running total so that I can review my order before checkout.
8. As a customer, I want to increase, decrease, or remove items from my cart so that I can adjust my order at any point before placing it.
9. As a customer, I want to proceed to checkout where I can choose between Dine In and Takeaway, enter a table number for Dine In, review the full order summary with subtotal, tax (10%), and total, and then confirm my order.
10. As a customer, I want to track my order status in real time (Placed, Preparing, Ready, Served) after placing it, so that I know exactly when my food will be ready.

11. As a customer, I want to view my past orders in an Order History screen so that I can keep a record of what I have previously ordered.
12. As a customer, I want to book a table reservation by selecting a date, time, and number of guests, and optionally adding special requests (e.g., Birthday celebration), so that I can plan my visit in advance.
13. As a customer, I want to view all my reservations and their statuses (Confirmed, Completed, Cancelled) so that I can keep track of my bookings.
14. As a customer, I want to cancel a reservation I no longer need so that the table can be made available for other guests.
15. As a customer, I want to view and update my profile information (name, phone number, email) so that my account details stay current.
16. As a customer, I want to log out of my account so that my session is securely terminated.

Manager User Stories

17. As a manager, I want to log in with my manager credentials so that I can access the management dashboard and all administrative functions.
18. As a manager, I want to see a dashboard showing key statistics (number of active orders, today's total revenue, today's reservations, total menu items) at a glance, so that I can assess the restaurant's current operational status quickly.
19. As a manager, I want to view a list of all incoming orders and update their status (Placed to Preparing, Preparing to Ready, Ready to Served) so that I can coordinate kitchen operations efficiently.
20. As a manager, I want to tap on an order to view its full details — customer name, items, quantities, prices, subtotal, tax, total, order type, table number, and special requests — so that I can review it thoroughly.
21. As a manager, I want to add a new menu item by filling in its name, category, description, price, and image URL so that I can expand the restaurant's offerings.
22. As a manager, I want to edit an existing menu item's details (name, category, description, price, image) so that I can keep the menu accurate and current.
23. As a manager, I want to delete a menu item so that discontinued dishes are removed from the menu permanently.
24. As a manager, I want to toggle a menu item's availability so that out-of-stock items are hidden from customers without needing to delete them entirely.
25. As a manager, I want to view all table reservations and either confirm or cancel them so that I can manage table allocation effectively.
26. As a manager, I want to log out of the manager panel so that my session is securely ended.

Section 3 – Functional Requirements

FR-01: User Registration

- The system shall provide a registration screen with input fields for name, email, password, and phone number.
- All fields shall be validated as non-empty before submission is permitted.
- The email field shall be validated against a standard email format.
- The password field shall require a minimum length of six characters.
- Upon successful registration, the system shall display a confirmation message and redirect the user to the Login screen.

FR-02: User Authentication (Login / Logout)

- The system shall provide a login screen with fields for email and password.
- The system shall authenticate the user against stored credentials and return a descriptive error message for invalid entries (e.g., Invalid email or password).
- Upon successful login, the system shall persist the user session using AsyncStorage so that the user remains logged in across app restarts.
- The system shall provide a logout function that clears the stored session and returns the user to the Login screen.

FR-03: Role-Based Navigation

- The system shall route authenticated users to different interfaces based on their assigned role:
 - Customer → Customer Bottom Tab Navigator (Home, Menu, Cart, Orders, Profile)
 - Manager → Manager Bottom Tab Navigator (Dashboard, Orders, Menu, Reservations)
- Unauthenticated users shall only have access to the Login and Register screens.

FR-04: Home Screen (Customer)

- The system shall display a personalized greeting incorporating the logged-in user's name.
- The system shall display a horizontal scrollable list of Today's Specials, drawn from items flagged as isSpecial.
- The system shall display quick-access category buttons for all menu categories (Starters, Main Course, Breads, Desserts, Drinks).
- Tapping any special item shall navigate directly to the Item Detail screen for that item.

FR-05: Menu Browsing

- The system shall display all available menu items organized by category.
- The system shall provide a category filter allowing the user to view items from a chosen category (All, Starters, Main Course, Breads, Desserts, Drinks).
- The system shall provide a search bar that filters menu items by name in real time as the user types.

- Each menu item card shall display the item's image, name, price in Rs., and a quick Add to Cart action button.
- Tapping a menu item card shall navigate to the Item Detail screen.

FR-06: Item Detail Screen

- The system shall display the selected item's image, name, description, price, and category.
- The system shall provide a quantity selector with increment and decrement controls, with a minimum value of one.
- The system shall provide a special instructions text input field for customization requests.
- The system shall provide an Add to Cart button that adds the item with the selected quantity and instructions to the cart.
- Upon adding to cart, the system shall display a confirmation message and navigate back.

FR-07: Shopping Cart

- The system shall display all items added to the cart with name, quantity, unit price, and line total (price multiplied by quantity).
- The system shall allow the user to increase or decrease the quantity of any cart item.
- The system shall allow the user to remove an individual item from the cart entirely.
- The system shall display a cart badge on the tab bar icon showing the total number of items currently in the cart.
- The system shall display the subtotal, tax (10%), and grand total at the bottom of the cart screen.
- The system shall provide a Proceed to Checkout button, which shall be disabled if the cart is empty.
- The system shall include an order-level special request text input for general notes.

FR-08: Checkout & Order Placement

- The system shall display a complete order summary showing items, quantities, prices, subtotal, tax, and total.
- The system shall allow the customer to select an order type: Dine In or Takeaway.
- When Dine In is selected, the system shall display a mandatory table number input field.
- The system shall validate that a table number is provided for all Dine In orders.
- The system shall provide a Confirm Order button to finalize the transaction.
- Upon confirmation, the system shall generate a unique order ID (format: ORD-XXXX), clear the cart, and navigate to the Order Tracking screen.

FR-09: Order Status Tracking

- The system shall display the current order's details and a visual progress tracker showing the full order lifecycle: Placed, Preparing, Ready, and Served.
- The currently active stage shall be visually highlighted to differentiate it from completed and upcoming stages.

- The system shall display the order ID, order type, table number (for Dine In orders), itemized list, and total.
- The customer shall be able to cancel the order only when its status is still Placed.

FR-10: Order History

- The system shall display a chronological list of the customer's past orders, sorted by date with the most recent appearing first.
- Each order card shall show the Order ID, date, status, order type, and total amount.
- Tapping an order shall navigate to the Order Tracking screen for that specific order.

FR-11: Table Reservation Booking

- The system shall provide a reservation form containing a date picker (restricted to future dates), a time picker, a number of guests input, and an optional special requests text input.
- The system shall provide a Check Availability button that simulates availability verification.
- Upon availability confirmation, the system shall present a Confirm Reservation button.
- Upon confirmation, the system shall create a reservation with status Confirmed, assign a table number, and display a success message with the reservation details.

FR-12: My Reservations

- The system shall display a list of the customer's reservations including date, time, guests, table number, status, and any special requests.
- Reservation statuses shall include Confirmed, Completed, and Cancelled.
- The system shall allow the customer to cancel an active reservation, with status changing to Cancelled immediately upon confirmation.

FR-13: User Profile Management

- The system shall display the logged-in user's name, email address, and phone number.
- The system shall allow the user to update their name and phone number via an editable form.
- The system shall provide navigation links to the Book a Table and My Reservations screens.
- The system shall provide a clearly labelled Logout button.

FR-14: Manager Dashboard

- The system shall display the following summary statistics as styled stat cards:
 - Active Orders — count of orders currently in Placed, Preparing, or Ready status
 - Today's Revenue — sum of totals for all orders placed on the current date
 - Today's Reservations — count of reservations scheduled for the current date
 - Menu Items — total count of items present in the menu
- Each stat card shall include an icon, a descriptive label, and the numeric value.
- The dashboard shall display a greeting with the manager's name and provide a Logout button.

FR-15: Manage Orders (Manager)

- The system shall display all orders in a filterable list supporting filtering by status.
- Each order card shall display the Order ID, customer name, status, order type, total, and timestamp.
- Tapping an order shall navigate to the Order Detail screen for that order.

FR-16: Order Detail & Status Update (Manager)

- The system shall display full order information: customer name, order ID, itemized list with quantities and prices, subtotal, tax, total, order type, table number, special request, and current status.
- The system shall provide a status update control allowing the manager to advance the order to the next logical status: Placed to Preparing, Preparing to Ready, Ready to Served.
- Status changes shall be reflected immediately on the screen without requiring a page reload.

FR-17: Menu Management (Manager)

- The system shall display all menu items in a list organized by category.
- Each item shall show its name, price, category, and an availability toggle switch.
- The manager shall be able to toggle item availability directly from the list view.
- The system shall provide an Add New Item button navigating to the Add/Edit screen.
- Tapping an existing item shall navigate to the Add/Edit screen pre-filled with that item's current data.
- The system shall allow the manager to delete a menu item with a mandatory confirmation prompt before permanent removal.

FR-18: Add / Edit Menu Item (Manager)

- The system shall provide a form containing fields for Name, Category (picker/dropdown), Description, Price, and Image URL.
- All fields shall be validated as non-empty before the form can be submitted.
- The Price field shall be validated as a positive numeric value.
- In Add mode, submitting the form shall create a new menu item and return to the menu list.
- In Edit mode, the form shall be pre-populated with the existing item's data, and submitting shall update the item in place.

FR-19: Reservation Management (Manager)

- The system shall display all reservations including customer name, date, time, guests, table number, status, and special requests.
- The manager shall be able to confirm a reservation, setting its status to Confirmed.
- The manager shall be able to cancel a reservation, setting its status to Cancelled.
- Status changes shall be reflected in the list immediately upon action.

Section 4 – Non-Functional Requirements

NFR-01: Performance

- The application shall load the Home screen within two seconds on a mid-range mobile device under normal operating conditions.
- Screen transitions and navigation shall complete within 300 milliseconds to maintain a fluid user experience.
- Cart operations such as adding, removing, and updating items shall reflect instantaneously in the interface without perceptible delay.
- All list views (menu items, orders, reservations) shall use optimized rendering components (FlatList) capable of handling 30 or more items at 60 frames per second without degradation.

NFR-02: Usability

- The application shall adhere to Material Design guidelines (Android) and iOS Human Interface Guidelines to provide a familiar, native-feeling experience on each platform.
- All interactive elements such as buttons and links shall have a minimum touch target of 44 by 44 density-independent pixels for accessibility compliance.
- The application shall display clear inline error messages for all form validation failures (e.g., invalid email format, empty required fields).
- Loading states shall always display a visible spinner or activity indicator to inform the user that an operation is in progress.
- A consistent visual language shall be applied throughout the application using the defined color palette: deep maroon (#8B0000) as the primary color, golden yellow (#D4A017) as the accent, and warm off-white (#FFF8F0) as the background.

NFR-03: Reliability & Availability

- The application shall handle loss of internet connectivity gracefully by displaying meaningful error messages rather than crashing or freezing.
- The user session shall persist across app restarts using AsyncStorage so that users are not required to re-authenticate unnecessarily.
- The application shall not crash or enter an unrecoverable state when presented with unexpected input such as empty fields, excessively long text, or special characters.

NFR-04: Security

- User passwords shall not be stored in plain text in any production deployment. The MVP stores passwords in mock data solely for development convenience.
- Session data persisted in AsyncStorage shall be limited to non-sensitive fields: user ID, name, email, and role. Passwords shall never be stored client-side.
- The application shall enforce role-based access control to ensure customers cannot navigate to manager screens and managers cannot enter customer flows.
- Input fields shall be sanitized to prevent injection attacks when the application is connected to a live backend.

NFR-05: Scalability

- The application architecture shall follow a Component, Context, and Data layering pattern that can be swapped with a real backend (REST API or Firebase) without significant refactoring.
- The mock data layer shall be isolated within a dedicated /data directory, making it straightforward to replace individual data files with API calls.
- The Context API abstraction layer shall be replaceable with a more robust state management library such as Redux or Zustand as the application grows in complexity.
- The NoSQL database schema defined in Section 5 shall be designed to support scaling to thousands of concurrent users, orders, and menu items.

NFR-06: Portability & Compatibility

- The application shall run on both Android (version 10 and above) and iOS (version 14 and above) through the React Native cross-platform framework.
- The MVP build shall be fully compatible with Expo Go for rapid development and on-device testing without requiring a native build pipeline.
- The application shall render correctly across screen widths ranging from 320 dp (small phones) to 428 dp (large phones).

NFR-07: Maintainability

- The codebase shall follow a modular directory structure with dedicated folders for screens, components, context, data, navigation, and constants.
- Reusable interface elements such as MenuItemCard, CartItem, OrderCard, StatusBadge, and LoadingSpinner shall each be implemented as independent, self-contained components.
- All color values and visual theme settings shall be centralized in /constants/colors.js and /constants/theme.js to support easy future theming and rebranding.

NFR-08: Localization Readiness

- All monetary values throughout the application shall be displayed in Pakistani Rupees (Rs.).
- The menu shall feature authentic Pakistani cuisine items with culturally appropriate names such as Karahi Chicken, Biryani, Nihari, Haleem, and Gulab Jamun.
- The application architecture shall support future localization by centralizing all user-facing text strings for straightforward translation.

Section 5 – Database Schema (NoSQL)

The following tables define the NoSQL document-based collections that the system uses. Each collection stores documents identified by a unique id field. This schema is designed to support both the MVP mock data layer and a future cloud database (e.g., Firebase Firestore or MongoDB Atlas).

5.1 Users Collection

Collection	Field	Type	Description
users	id	String	Unique user identifier (e.g., u1, u2)
	name	String	Full display name of the user
	email	String	User email address used for login; must be unique across the collection
	password	String	User password; hashed in production, plaintext in MVP mock data only
	phone	String	User phone number (e.g., 03001234567)
	role	String	User role — either customer or manager

5.2 Menu Items Collection

Collection	Field	Type	Description
menuItems	id	String	Unique menu item identifier (e.g., m1, s1, dr3)
	name	String	Name of the dish (e.g., Karahi Chicken)
	category	String	Category — one of: Starters, Main Course, Breads, Desserts, Drinks
	description	String	Short description of the dish and its preparation style
	price	Number	Price in Pakistani Rupees (e.g., 950)
	image	String	URL pointing to the item's display image
	available	Boolean	Whether the item is currently available for ordering (true or false)
	isSpecial	Boolean	Whether the item is featured as a Today's Special on the home screen

5.3 Orders Collection

Collection	Field	Type	Description
------------	-------	------	-------------

orders	id	String	Unique order identifier (format: ORD-XXXX, e.g., ORD-1001)
	customerId	String	Reference to the users.id of the customer who placed the order
	customerName	String	Display name of the customer, denormalized for quick rendering
	items	Array<Object>	Array of line items each containing: itemId (String), name (String), quantity (Number), price (Number)
	subtotal	Number	Sum of (price x quantity) for all items in Rs.
	tax	Number	Tax amount calculated at 10% of subtotal
	total	Number	Grand total (subtotal + tax) in Rs.
	status	String	Current status — one of: Placed, Preparing, Ready, Served
	orderType	String	Type of order — either Dine In or Takeaway
	tableNumber	String Null	Assigned table number for Dine In orders; null for Takeaway
	specialRequest	String	Any special instructions or requests for the overall order
	timestamp	String (ISO 8601)	Date and time the order was placed (e.g., 2026-02-20T12:30:00.000Z)

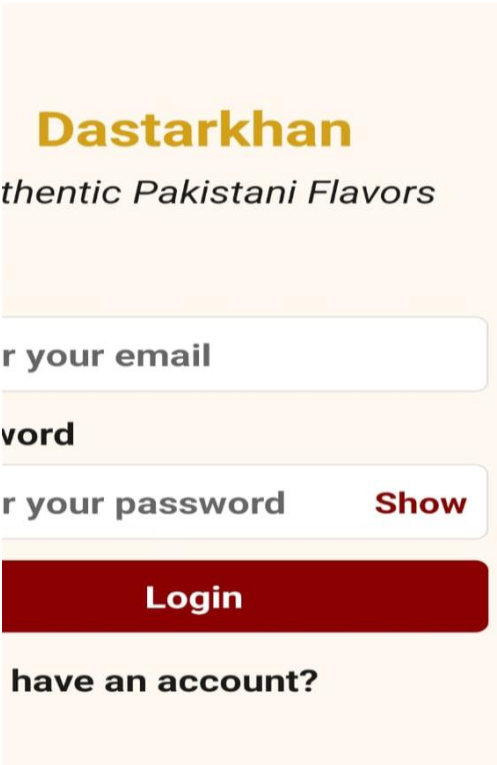
5.4 Reservations Collection

Collection	Field	Type	Description
reservations	id	String	Unique reservation identifier (e.g., res1)
	customerId	String	Reference to the users.id of the customer who made the reservation
	customerName	String	Display name of the customer, denormalized for quick rendering
	date	String	Reservation date in YYYY-MM-DD format (e.g., 2026-02-25)
	time	String	Reservation time as a display string (e.g., 7:00 PM)
	guests	Number	Number of guests for the reservation
	tableNumber	String	Assigned table number (e.g., 5)
	status	String	Reservation status — one of: Confirmed, Completed, Cancelled

	specialReque sts	String	Any special requests from the customer (e.g., Birthday celebration)
--	---------------------	--------	---

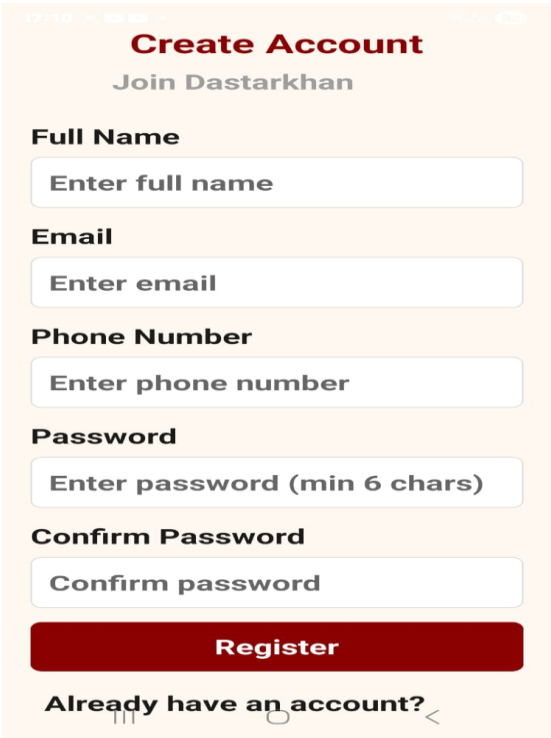
SCREENSHOTS OF APP

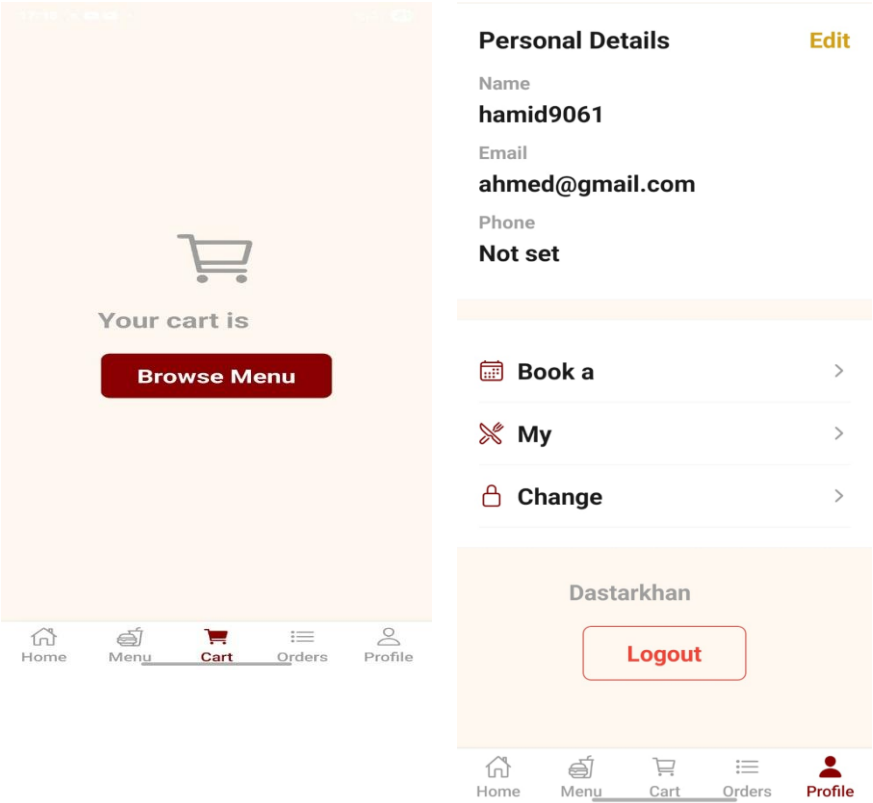
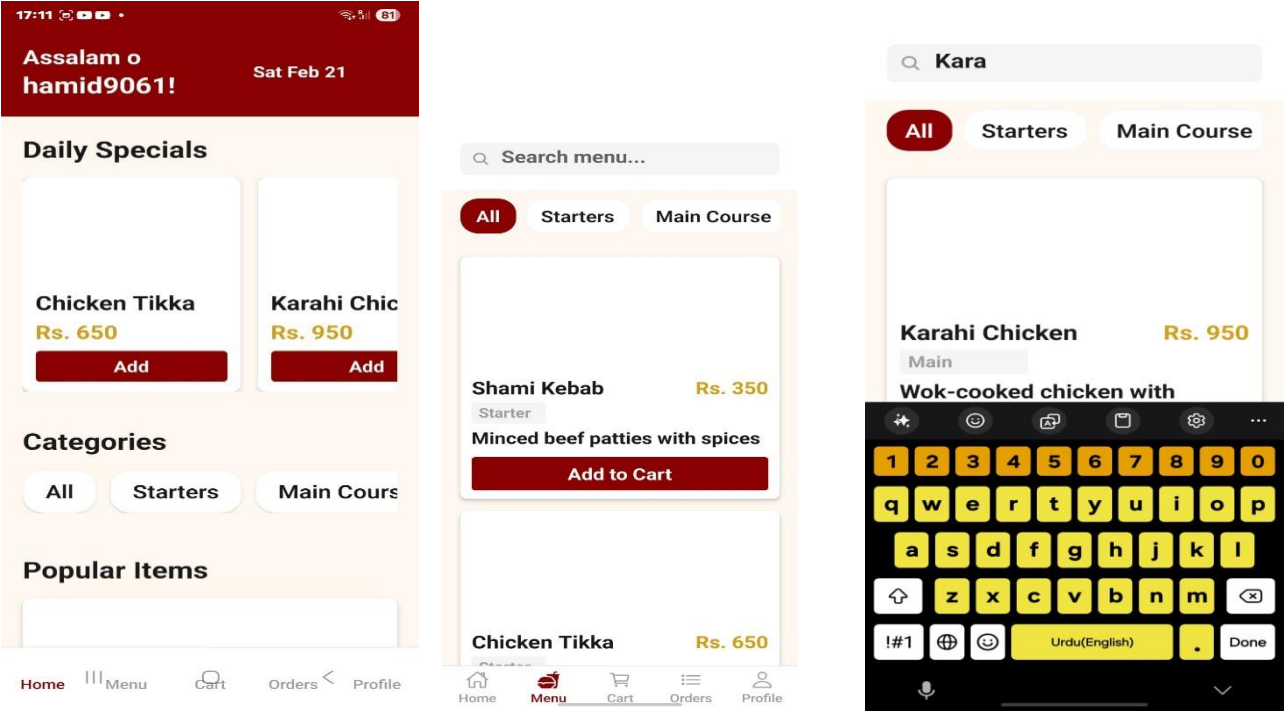
[HOME SCREEN]



FOR CUSTOMERS

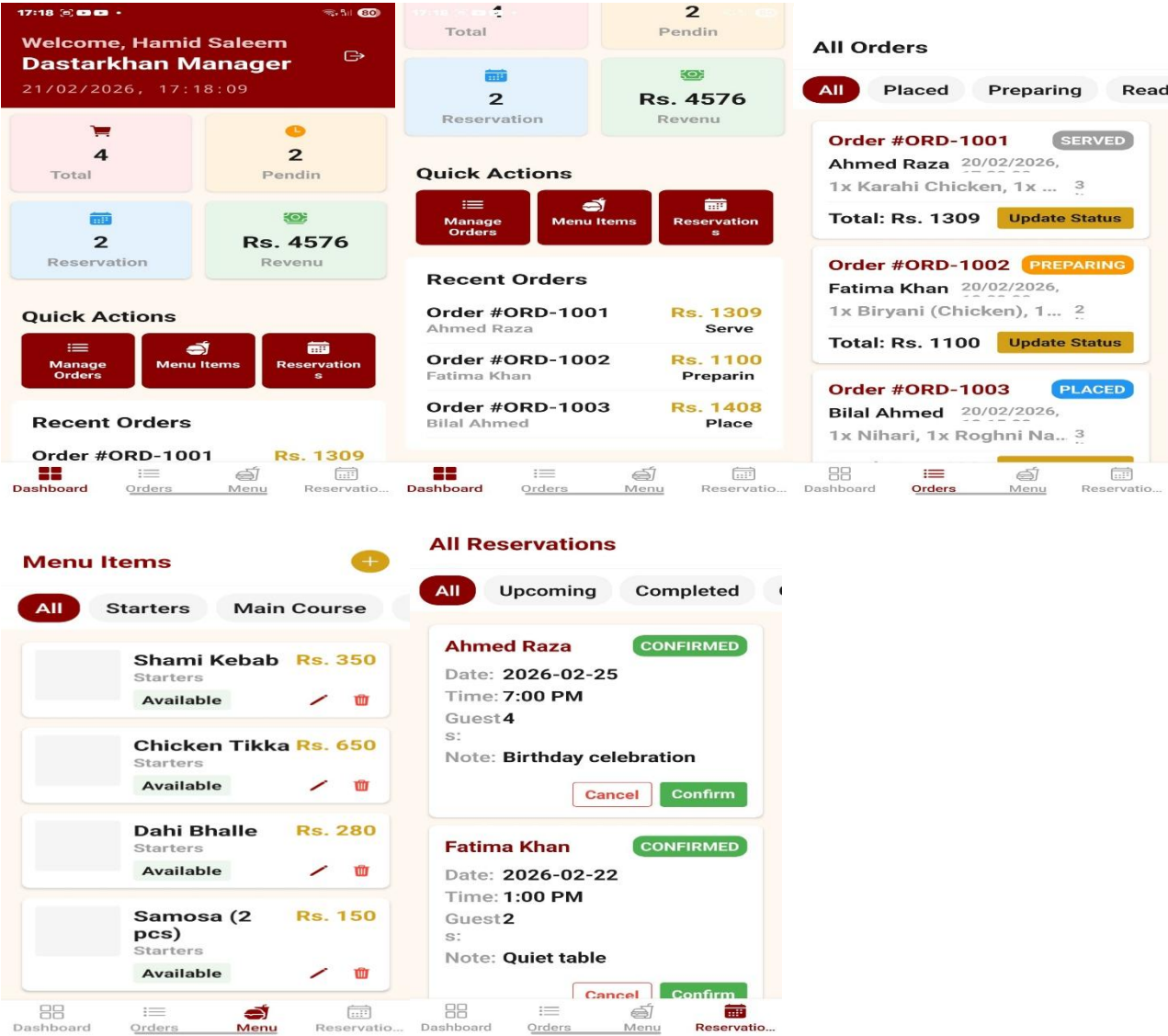
REGISTRATION SCREEN





F

FOR MANAGER



I