Mobile Application Development

Project Report

Project Title Maximum two lines



**B.S. (CS/SE) 2021**

**Submitted by:**

**Name : Enrolment**

**Name: Enrolment**

**Name: Enrolment**

**Name: Enrolment**

**13th November 2024**

**Department of Computer Science and Information Technology**

**Jinnah University for Women**

5-C Nazimabad, Karachi 74600

Contents

[Project Overview 3](#_Toc182308530)

[Features 3](#_Toc182308531)

[Tools and Technologies 3](#_Toc182308532)

[Front-End Tools 3](#_Toc182308533)

[Back-End Tools 3](#_Toc182308534)

[Database 3](#_Toc182308535)

[Third-party APIs and Services 3](#_Toc182308536)

[Algorithm(s) 3](#_Toc182308537)

[Code Snippets 3](#_Toc182308538)

[Screenshots 3](#_Toc182308539)

[Appendix 3](#_Toc182308540)

[Heading Level 1: Sample Heading 1 4](#_Toc182308541)

[Heading Level 2: Sample Heading 2 4](#_Toc182308542)

[Heading Level 3: Sample Heading 3 4](#_Toc182308543)

## ****Recipe App Project Report****

**Repository Link:** [GitHub - Recipe App](https://github.com/instamobile/recipes-app-react-native)

### ****1. Project Overview****

The Recipe App is a full-featured mobile application developed using **React Native** that enables users to explore, create, and save recipes. The app integrates user authentication, recipe posting, and a chatbot powered by the **Hugging Face API** to assist users with cooking-related queries.

The goal of the application is to provide an all-in-one platform for users to discover recipes, contribute their own, and get real-time help while cooking.

### ****2. Features****

The Recipe App includes the following features:

1. **User Authentication:**
   * Secure login and registration functionality using Firebase Authentication.
2. **Recipe Catalog:**
   * A collection of diverse recipes categorized for easy browsing.
3. **Post Recipe:**
   * Users can post their own recipes by providing a title, ingredients, steps, and an image.
4. **View Recipe:**
   * Users can view detailed recipes including:
     + Ingredients
     + Step-by-step instructions
     + Cooking time
     + Calorie information
5. **Favorites Management:**
   * Save and manage favorite recipes for quick access.
6. **Search Functionality:**
   * Search recipes based on keywords or ingredients.
7. **Chatbot (Hugging Face API):**
   * An integrated AI chatbot that answers cooking-related questions and provides suggestions.
8. **Detailed Recipe View with Images:**
   * High-quality recipe images to enhance user experience.
9. **User-Friendly Navigation:**
   * Smooth navigation between screens and categories.
10. **Offline Support:**
    * Allows saving data locally for offline access.

### ****3. Tools and Technologies****

#### ****Front-End Tools****

* **React Native:** Cross-platform mobile app development framework.
* **JavaScript (ES6+):** For building interactive components.
* **Redux:** State management for handling global state.
* **React Navigation:** Navigation library for handling in-app navigation.
* **Expo:** Toolchain for simplifying React Native development.

#### ****Back-End Tools****

* **Node.js** (Optional for custom APIs).
* **Firebase Firestore:** Backend database for storing recipes and user data.

#### ****Database****

* **Firebase Firestore**: Real-time NoSQL database for user authentication, recipes, and favorites.

#### ****Third-Party APIs and Services****

* **Hugging Face API:** Provides AI chatbot functionality for real-time cooking assistance.
* **Firebase Authentication:** User login and registration management.
* **Spoonacular API:** Recipe and nutrition data.

### ****4. Algorithm(s)****

1. **Search Algorithm:**
   * Filters recipes based on search queries (title or ingredients).
2. const searchRecipes = (query, recipes) => {
3. return recipes.filter((recipe) =>
4. recipe.title.toLowerCase().includes(query.toLowerCase()) ||
5. recipe.ingredients.some((ingredient) =>
6. ingredient.toLowerCase().includes(query.toLowerCase())
7. )
8. );
9. };
10. **Favorites Management:**
    * Adds or removes recipes from the user's favorites.
11. const toggleFavorite = (userId, recipeId) => {
12. firebase.firestore().collection('favorites').doc(userId).set({
13. recipeIds: firebase.firestore.FieldValue.arrayUnion(recipeId),
14. });
15. };
16. **Chatbot Integration (Hugging Face API):**
    * Sends a user query to Hugging Face API and retrieves a response.
17. const fetchChatbotResponse = async (userQuery) => {
18. const response = await fetch('https://api-inference.huggingface.co/models/chatbot-model', {
19. method: 'POST',
20. headers: { Authorization: 'Bearer YOUR\_API\_KEY' },
21. body: JSON.stringify({ inputs: userQuery }),
22. });
23. const data = await response.json();
24. return data.generated\_text;
25. };

### ****5. Code Snippets****

1. **Login/Register Component**
2. import firebase from 'firebase/app';
3. import 'firebase/auth';
4. const loginUser = (email, password) => {
5. firebase.auth().signInWithEmailAndPassword(email, password)
6. .then(() => console.log('User logged in'))
7. .catch((error) => console.error(error.message));
8. };
9. const registerUser = (email, password) => {
10. firebase.auth().createUserWithEmailAndPassword(email, password)
11. .then(() => console.log('User registered'))
12. .catch((error) => console.error(error.message));
13. };
14. **Post Recipe Form**
15. const postRecipe = (recipe) => {
16. firebase.firestore().collection('recipes').add(recipe)
17. .then(() => console.log('Recipe posted'))
18. .catch((error) => console.error(error.message));
19. };
20. **Chatbot Integration**
21. const handleChat = async (query) => {
22. const response = await fetchChatbotResponse(query);
23. console.log('Chatbot Reply:', response);
24. };

### ****6. Screenshots****

| **Login Page** | **Register Page** |
| --- | --- |
|  |  |

| **Recipe List** | **Post Recipe** |
| --- | --- |
|  |  |

| **Recipe Details** | **Chatbot Interface** |
| --- | --- |
|  |  |

### ****7. Appendix****

* **Project Dependencies:**
  + React Native
  + Firebase Authentication
  + Firebase Firestore
  + Hugging Face API
  + Redux
  + React Navigation
  + Expo
* **Future Enhancements:**
  + Voice-enabled recipe instructions
  + Recipe rating and reviews
  + Push notifications for recipe suggestions

**Formatting Guidelines:**

Font-Family: Times New Roman

Font-size for normal text: 12

Line Spacing: 1

Justified, Headings left Aligned

# Heading Level 1: Sample Heading 1

## Heading Level 2: Sample Heading 2

### Heading Level 3: Sample Heading 3

#### Heading Level 4: Sample Heading 4