DEPLOYMENT PLAN

List of Contributors

Beril Aydın 231101002

Nisa Naz Keleşoğlu 231101040

Zeynep Saçaklı 221101051

Task Matrix

Beril Aydın	Nisa Naz Keleşoğlu	Zeynep Saçaklı
Deployment Overview	Deployment Process	Configuration Plan

Table Of Contents

➢ Deployment Overview

Todayz Meal is deployed as a web application using Vercel for frontend hosting and Firebase for authentication. The project fetches recipe data from the Spoonacular API. The deployment includes:

1. **Frontend Hosting:**
    - Hosted on Vercel to ensure easy accessibility and fast content delivery.
    - Continuous updates and improvements managed via GitHub.
2. **Backend Service:**
    - Firebase Authentication for secure user login, registration, and session management.
    - Firestore Database for storing user-related data such as favorite recipes.
    - Spoonacular API integration for retrieving recipe data dynamically.
3. **Code Repository:**
    - API keys and authentication credentials are stored securely using environment variables.
    - Firebase security rules ensure data privacy and integrity.
    - HTTPS enforced for secure communication.
4. **Scalability & Maintenance:**
    - Firebase and Vercel provide a scalable and low-maintenance deployment approach

➢ Deployment Process

1. **Push Latest Code to GitHub Repository**
   - git add .
     git commit -m "Final deployment commit"
     git push origin main

2. **Deploy on Vercel**
   - Log in to Vercel and create a new project.
   - Connect your Github repository and select the relevant repository.
   - Start the deployment process and use the Vercel-provided URL to access the deployed application.

3. **Firebase Authentication Setup**
   - Set up a Firebase project in the Firebase console.
   - Enable Firebase Authentication and allow email/password sign-ups.
   - Configure Firebase SDK in register.js and login.js

4. **Database Configuration**
   - Enable Firestore in Firebase console.
   - Define security rules to restrict unauthorized access.
   - Structure collections properly.

5. **API Key Management & Security**
   - Restrict the Spoonacular API key to the domain of the deployed frontend.
   - Use environment variables or secure storage for API keys.

6. **Monitoring and Maintenance**
   - Monitor Firebase database activity to ensure data integrity.

➢ Configuration Plan

Frontend Deployment	Hosted on Vercel for easy accessibility and fast content delivery. React.js framework used to build a dynamic, responsive user interface.
Authentication	Firebase Authentication is used to handle user sign-ups and logins securely. Users sign up using their email and password, and authentication state is tracked within the app. Firebase rules ensure that unauthorized access is blocked.
Recipe API	Spoonacular API is integrated to fetch recipe data based on user inputs (such as ingredients or dietary preferences). The API key is securely stored in environment variables and restricted for use on the deployed website's domain.
Database	Firestore Database is used to store user-related data, such as saved recipes and preferences. Security rules are implemented to control access to this data, ensuring only authenticated users can access and modify their data. Firestore is structured with collections like users and favorites to organize user data efficiently. User information can be manually viewed via Firebase console.
Monitoring	No active monitoring tools (e.g., Firebase Analytics or Crashlytics) are in use. However, user data and interactions are manually checked via the Firebase console
Security	All sensitive data, including API keys and authentication credentials, is stored securely using environment variables. Firebase security rules ensure that only authorized users can read or write data to the database. API keys for external services like Spoonacular are also securely managed to prevent unauthorized access.