

21Summer

Jamii Appp

Functional Requirements

User Management

1. **Sign In**
 - Users can sign in using their credentials or social media accounts via OAuth.
2. **Sign Up**
 - Users can register by providing necessary details and verifying their email.
3. **Log Out**
 - Users can log out securely from their accounts.
4. **Delete Account**
 - Users can delete their accounts, with all associated data being removed from the system.
5. **Followers and Following**
 - Users can follow and be followed by other users, receiving updates and posts from those they follow.
6. **Chats**
 - Users can engage in direct messaging with other users, with real-time chat capabilities.
7. **Profile Update**
 - Users can update their profile information, including bio, profile picture, and other personal details.

Post Management

CRUD Operations: Users can create, read, update, and delete posts.

Comments Management: Users can add, edit, and delete comments on posts.

Likes Management: Users can like and unlike posts and comments.

External Sharing: Users can share posts to external platforms like social media and messaging apps.

Store Management

CRUD Operations: Businesses can create, read, update, and delete their stores.

Reviews and Rating Management: Customers can leave reviews and ratings for stores, which businesses can respond to.

Product Management

1. **CRUD Operations:** Businesses can create, read, update, and delete products.
2. **Reviews and Rating Management:** Customers can leave reviews and ratings for products, which businesses can respond to.
3. **Direct Messaging Management:** Businesses can send and receive direct messages from customers regarding products.

AI Search and Chat Management

1. **Retrieval-Augmented Generation (RAG):** AI assists in generating context-aware responses based on retrieved documents.
2. **Completions:** AI provides text completion for user queries.
3. **Computer Vision:** AI analyzes images for various purposes, such as auto-filling product details.
4. **Text to Speech and Vice Versa:** AI converts text to speech and speech to text.
5. **Audio Input:** Users can interact with the AI using voice commands.
6. **Sentiment Analysis:** AI analyzes user sentiment in messages and feedback.
7. **Voice Customization:** Users can customize the AI's voice for a more personalized experience.
8. **Human-like Customization:** AI responses can be tailored to appear more human-like in interactions.

Recommender System

1. **User Preference-Based:** Recommendations based on individual user preferences.
2. **Location-Based:** Recommendations based on the user's geographic location.
3. **Behavioral-Based:** Recommendations based on user behavior and activity on the platform.

Community Management

CRUD Operations: Users can create, read, update, and delete community groups.

Feed Management: Users can manage the feed content within their community groups.

File Uploading

Multimedia Input

Users can upload images, videos, and other multimedia files.

Data Analytics and Reporting

Businesses and administrators can access analytics and reports on user behavior, sales, and other key metrics.

Non-Functional Requirements

Availability

1. **Introducing Redundancy in Python Cloud Functions:** Implement redundancy to ensure continuous availability of cloud functions.
2. **Failover Mechanisms:** Establish failover protocols to handle server failures without service interruption.
3. **Utilize Firebase Multi-Region Support:** Enable data replication across multiple regions for high availability.
4. **Implement Caching Strategies:** Reduce server load by caching frequently accessed data.
5. **Use a CDN:** Employ a Content Delivery Network for fast access to static content and frequently accessed data.
6. **Utilize Cloud Horizontal Scaling:** Automatically scale resources horizontally to handle increased load.
7. **Utilize Firebase Monitoring and Alert Systems:** Monitor system performance and set up alerts for any issues.

Testability

Write and Execute Tests

1. **Unit Tests:** Test individual components for correctness.
2. **Integration Tests:** Test combined parts of the application for correct interaction.
3. **System Tests:** Test the entire system to ensure it meets requirements.

4. **Regression Tests:** Ensure new changes do not affect existing functionality.
5. **User Tests:** Test the system from an end-user perspective for usability.

Choice of Technology

1. **Flutter**
 - Cross-platform development framework for building the client-side application.
2. **Firebase**
 - Backend-as-a-Service (BaaS) platform for authentication, database, analytics, and more.
3. **Python Cloud Functions**
 - Serverless functions for handling backend logic and integrations.