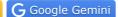
README.md 2025-04-06

# GenAl Chatbot - Chatbot Evolution

A versatile domain-adaptable GenAl chatbot that provides natural conversational interactions across multiple business domains. Built for the Chatbot Evolution Hackathon, this system showcases seamless adaptability from restaurant ordering to clinic appointments and beyond.









## Deployment

Deployed the project on Netlify https://chatbot-evolution.netlify.app/

## Key Features

#### • Multi-Domain Support

- I Restaurant Ordering System
- Clinic Appointment Booking
- mathematical Hotel Reservations

#### **Smart Interactions**

- Natural language processing with Google's Gemini Al
- Context-aware conversations
- Intelligent response enhancement with domain data

#### Advanced UI/UX

- Responsive design for all devices
- Dark/Light theme support
- Real-time typing indicators
- Message markdown support
- Intuitive chat interface

#### **Technical Excellence**

- Domain adapter pattern for easy extensibility
- Smart caching for performance optimization
- Offline support capabilities
- Error boundary protection
- Extensive accessibility features

## Getting Started

### **Prerequisites**

- Node.js (v14 or higher)
- npm/yarn
- Google Cloud Account (for Gemini Al)

README.md 2025-04-06

Firebase Account

#### Setting Up Firebase

- 1. Create a new Firebase project at Firebase Console
- 2. Enable Authentication and Firestore
- 3. Get your Firebase configuration:

```
const firebaseConfig = {
   apiKey: "your-api-key",
   authDomain: "your-auth-domain",
   projectId: "your-project-id",
   storageBucket: "your-storage-bucket",
   messagingSenderId: "your-messaging-sender-id",
   appId: "your-app-id"
};
```

4. Create a .env file in the project root and add:

```
REACT_APP_FIREBASE_CONFIG=your-stringified-config
```

#### Setting Up Google Generative Al

- 1. Visit Google Al Studio
- 2. Create an API key
- 3. Add to your .env:

```
REACT_APP_GEMINI_API_KEY=your-api-key
```

#### Installation

```
# Clone the repository
git clone https://github.com/yashveer132/Chatbot-evolution.git

# Install dependencies
cd chatbot-evolution
npm install

# Start development server
npm start
```

## **Q** Usage Examples

README.md 2025-04-06

```
User: "What vegetarian options do you have?"

AI: "We have several vegetarian dishes:

• Vegetarian Buddha Bowl ($12.99)

• Mushroom Risotto ($13.99)

• Quinoa Power Bowl ($16.99)

All are prepared fresh and can be customized to your preferences."
```

#### Clinic Assistant

#### Hotel Assistant

```
User: "What rooms are available next weekend?"
AI: "For next weekend, we have:
    • Standard King Room ($199/night)
    • Deluxe King Room ($249/night)
Both include free WiFi, daily housekeeping, and city views."
```

## **Architecture**

## **\*** Meeting Hackathon Criteria

### Portability \*

- Domain adapter pattern enables quick integration of new domains
- Consistent interface across different business contexts

README.md 2025-04-06

- Modular architecture for easy adaptation
- Accessible and intelligent assistance

## Code Quality

- Clean code architecture with separation of concerns
- · Comprehensive error handling
- · Consistent coding standards

## Versatility 😉

- Successfully implemented 3 distinct domains. Solves real world problems.
- Extendable prompt enhancement system
- Flexible data structure adaptation
- Integrated APIs, database and multilingual capabilities

#### Usability **£**

- Intuitive chat interface
- Real-time response indicators
- Mobile-responsive design
- · Accessibility features

### Efficiency **∮**

- Smart caching system
- Optimized API calls
- Local data enhancement
- Progressive loading

## Innovation 😯

- Context-aware conversations
- Dynamic response formatting
- Multi-modal input support
- · Seamless domain switching

### **1** Team

Yash Veer Singh (AJAX)

Made with W for Chatbot-Evolution