Product Requirements Document (PRD) for AskZen

Overview

Project Name: AskZen

Project Type: Conversational AI Platform

Goal: Build an AI-driven chatbot creation and management platform for businesses, similar to GaliChat, allowing automation of customer support and enhanced live chat integrations.

1. Objective

Create a user-friendly platform where businesses can:

- Design and deploy chatbots to automate customer interactions.
- Seamlessly integrate live chat for escalations requiring human intervention.
- Access detailed analytics to optimize chatbot performance and user experience.

2. Target Audience

- **Small and Medium Enterprises (SMEs)** - For managing customer support with limited resources.
- **Enterprises** - To enhance customer experience and scale support operations.
- **Developers** - As a tool to integrate chatbot solutions into custom apps and websites.

3. Features

Core Features

1. **Chatbot Builder**
   - Drag-and-drop interface for building conversation flows.
   - Predefined templates for common use cases (e.g., FAQs, bookings, and customer inquiries).
   - Natural Language Processing (NLP) for understanding user queries.
   - Multi-language support.

2. **Knowledge Base Management**
   - Upload files (PDF, CSV, etc.) or crawl websites for content.
   - Manual addition of FAQs and structured responses.
   - Real-time updates to chatbot knowledge.

3. **Integration Options**
   - Embed chatbots on websites and apps (SDKs and embed scripts).
   - Integrations with third-party platforms (e.g., WhatsApp, Facebook Messenger, Slack).
   - CRM and help desk integrations (e.g., HubSpot, Salesforce, Zendesk).

4. **Live Chat Support**
   - Human escalation workflow: Allow switching from bot to a live agent.

- Unified inbox for live chat and chatbot escalations.
- Notifications for pending human assistance requests.

5. **AI and Machine Learning**

- Contextual NLP for personalized responses.
- Sentiment analysis to identify user emotions.
- Continuous learning from user interactions to improve accuracy.

6. **Analytics Dashboard**

- Chatbot performance metrics (e.g., response time, resolution rate).
- User behavior insights (e.g., most common queries, sentiment trends).
- Feedback loops for improving chatbot responses.

Optional Advanced Features

- **Voice Integration** - Support for voice-based chat interactions.
- **Custom AI Models** - Allow businesses to train models specific to their domain.
- **API Access** - For developers to embed AskZen functionality into their applications.

User Roles

- **Admin** - Manage organization settings, integrations, and billing.
- **Bot Designer** - Build and update chatbots.
- **Agent** - Handle live chat escalations.

5. Platform Architecture

Frontend

- **Built with:** React.js or Next.js
- **Features:**
  - Responsive design.
  - Intuitive drag-and-drop chatbot builder.

Backend

- **Built with:** Node.js + Express
- **Database:** MongoDB for scalability.
- **Real-time communication:** Socket.IO for live chat.

AI and NLP

- **Use:** OpenAI's GPT-4 API, Google Dialogflow, or Hugging Face models for chatbot logic.

Infrastructure

- **Cloud hosting:** AWS, Google Cloud, or Azure.
- **Content delivery:** Cloudflare or similar CDN for fast bot responses.
- **Database management:** Firebase for real-time data, or PostgreSQL for structured storage.

6. Key Milestones

- **Phase 1: Discovery (1-2 Weeks)**
  - Market research and competitive analysis.
  - Define MVP scope.
- **Phase 2: Prototyping (3-4 Weeks)**
  - Create wireframes for chatbot builder and dashboard.
  - Develop frontend prototypes.
- **Phase 3: Core Development (8-10 Weeks)**
  - Implement chatbot building and NLP capabilities.
  - Develop backend for data handling and analytics.
- **Phase 4: Integration and Testing (4 Weeks)**
  - Add third-party integrations (WhatsApp, Facebook Messenger, etc.).
  - Test with a closed group of beta users.
- **Phase 5: Launch and Iteration**
  - Release MVP.
  - Collect feedback and iterate.

7. Success Metrics

- Number of chatbots deployed by users.
- Average response and resolution time improvement.
- User retention rate on the platform.
- Reduction in businesses' support costs.

8. Technology Stack

- **Frontend:** React.js, Tailwind CSS.
- **Backend:** Node.js, Express, MongoDB.
- **AI/NLP:** OpenAI GPT, Google Dialogflow.
- **Real-Time Communication:** Socket.IO.

9. Assumptions and Risks

- **Assumptions:**
  - Users have a basic understanding of chatbot design.
  - Integrations with popular communication platforms will be straightforward.
- **Risks:**
  - Adoption might be slow in the initial stages.
  - Dependence on third-party platforms (e.g., OpenAI, Google) for AI features.