**PROJECT PHASE I**

|  |  |
| --- | --- |
| DATE | 26-09-2023 |
| NAAN MUTHALVAN TEAM MEMBERS ID | au513121104005  au513121104023  au513121104024  au5131211040708 |
| PROJECT NAME | CREATE A CHATBOT USING PYTHON |

**TABLE** **OF CONTENTS:**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **CONTENT** | **PAGE NO** |
| 1. | PROPOSED SOLUTION | 2 |
| 2. | BRAINSTORMING AND IDEA PRIORITIZATION | 4 |
| 3. | EMPATHY MAP | 6 |
| 4. | SOLUTION ARCHITECTURE | 7 |

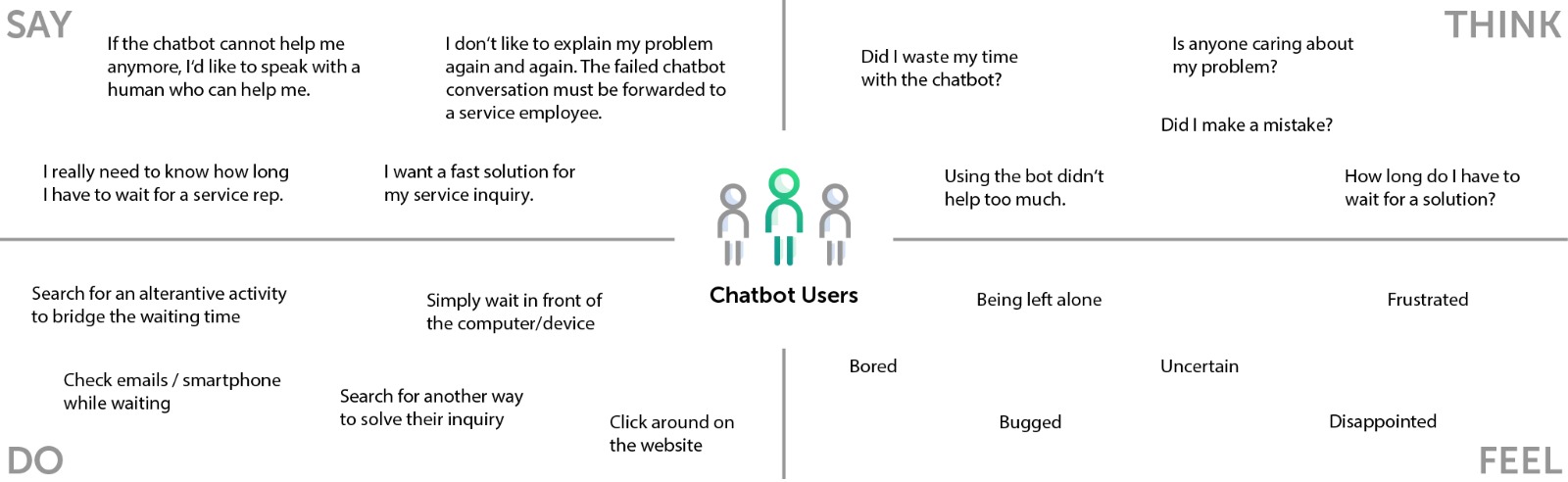
**PROPOSED SOLUTION:**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
|  |  |  |
| 1. | Problem statement | * Users often have questions or require assistance while navigating a website, and addressing their inquiries promptly can be challenging. * There is a need for a chatbot solution that can enhance user experience, provide real-time assistance, and support customers' needs effectively. |
|  |  |  |
|  |  |  |
|  |  |  |
| 2. | Novelty / Uniqueness | * Advanced NLP techniques for understanding user intent and context. * Continuous learning and improvement based on user feedback. |
|  |  |  |
|  |  |  |
|  |  |  |
| 3. | Social Impact / Customer Satisfaction | **Improved User Experience:** The chatbot enhances website navigation, providing quick answers and assistance.    **Accessibility:** It ensures that users can access support or information 24/7, improving accessibility for a broader audience. |
|  |  |  |
|  |  |  |
| 4. | Business Model (Revenue Model) |  |
|  |  | Offer a basic version of the chatbot for free and charge for premium features, such as advanced personalization or priority support. |
|  |  |  |
|  |  |  |
| 5. | Scalability of the Solution | Ensure that the chatbot architecture is scalable to accommodate an increasing number of users and interactions.  Use cloud-based infrastructure and containerization to easily scale resources as needed. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**BRAINSTORMING AND IDEA PRIORITIZATION:**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Parameter** | **Description** |
| 1. | Objectives | * **Appointment Management:** Allow patients to schedule, reschedule, or cancel appointments. * **Patient Records:** Provide access to medical records, test results, and treatment history. * **Information Hub:** Answer FAQs about hospital services, hours, parking, and billing. * **Doctor Details:** Assist in finding doctors, specialties, availability, and contact information. * **Medication Support:** Handle prescription refills and set medication reminders. * **Emergency Assistance:** Offer guidance during medical emergencies. * **Insurance Help:** Assist with insurance inquiries and billing questions. |
| 2. | Frameworks | * **ChatterBot**: A Python library for creating chatbots that can be trained to understand and respond to user queries. * **Dialogflow**: A Google Cloud platform for building natural language understanding applications. * **Rasa**: An open-source platform for building AI-driven chatbots and assistants. |
| 3. | Dataset | [**https://www.kaggle.com/datasets/grafstor/simple-dialogs-for-chatbot**](https://www.kaggle.com/datasets/grafstor/simple-dialogs-for-chatbot) |
| 4. | Model training and integration | * Train the chatbot model using the collected data. * Use the selected framework's training tools or APIs to build your chatbot's natural language understanding capabilities. |
| 5. | Natural Language Processing (NLP) | * Implement NLP techniques to understand and interpret user queries accurately. * Handle user input, extract intent, entities, and context to generate meaningful responses. |

**EMPATHY MAP:**



**SOLUTION ARCHITECTURE:**

* User Interface: Determine how users will interact with your chatbot. This can be through a web interface, a messaging app, or any other platform. You might need to use frameworks like Flask or Django for web-based interfaces.
* Input Processing: When a user sends a message, the chatbot needs to process it. This involves text preprocessing to clean and normalize the input.
* Natural Language Understanding (NLU): Use NLU techniques to extract meaning from user messages. This often includes tasks like intent recognition (what the user wants) and entity recognition (identifying specific details like dates, names, etc.). Libraries like spaCy or NLTK can be helpful here

