**Description**

**“ChatBot which answers Syncfusion component related common queries**”

We have collected the frequently asked queries of customers by analyzing data in incidents and discussion forums along with the answers and stored them in the ChatBot. Customers can use the ChatBot to get instant replies for their common queries which may be related to the company or the products, thus saving them a lot of time.

**Existing problem**

Currently our customers who have queries, create incidents or start a discussion forum. Of these, most of them are frequently asked queries which are either documented in our help site or have already been discussed in a forum. These information’s can be easily accessed by them to solve their problems, but they will not be aware of that.

Even for simple queries, they create incident. So unwantedly it would increase the incidents count as well as consumed the support engineers time.

**Proposed solution**

Answers for these common and frequently asked queries (based on data analyzed from previous incidents and forums and documentation) have been stored in the ChatBot, customers can use this to clear their doubts. This will help customers to get instant answers and reduce them the time delay. If their query is not answered, they will be directed to contact the support team. For employees, work will be reduced based on the incident count.

**Working of ChatBot**

Customers can input in their queries in the ChatBot which is like a normal chatting application and it responds with the matched answer and suggested links (if any) which are related to the query. All these data have been stored in the ChatBot database. If the answer is not related to their query, the ChatBot directs them to contact the support team to help them further.

**Key features**

* Simple chat application which gives ease of access to customers.
* Available 24/7 for customer queries.
* Saves time for customers for common queries.
* Reduces the Incident/Forum count.
* Reduce support engineer works.
* Provides suggested results which might also be related to customer query.

**Architecture**

The following architecture explains the functionality of the ChatBot,

**INPUT**

**OUTPUT**

**DECISION MAKING**

**SEARCH MATRIX**

**Advantage of ChatBot for customers**

* This ChatBot will be very useful for customers as it **solves their simple queries instantly**.
* Customers who have simple or commonly asked queries can get instant responses with this and it **save them a lot of time.**
* They can easily access it like as a simple chat application.
* Customer can get more related results for their queries.

**Advantage of ChatBot for our company**

* This ChatBot can be very helpful in customer support. There are many frequently asked incidents/forums which are based on general queries. Instead of support engineers providing the same solution each time, these can be answered by this ChatBot instantly which in-turn improves the customer support.
* Reduces the Incident/Forum count.
* Reduce support engineer works.
* Increase customer count and revenue to our company by resolving the customer’s query instantly.

**Importance of ChatBot for customer service**

* A ChatBot gives instance response to queries and saves a lot of time.
* Although this ChatBot cannot handle all customer queries, it can handle the repeated queries.
* Customer can get more related results even for their simple query. It will guide them to proceed further in those components.
* Ease access of customers

**Modules in ChatBot**

* Create Simple Chat Application
* Data Analysis
* Mapping the key values based on custom search
* Filtering
* Response based on customer query

**Create Simple Chat Application:**

*Author*–

We developed a simple chat application with the help of botkit(Open source application from github) based on our requirements. It is a Node.js module, and works with Node and npm. For more information about botkit please refer to - <https://botkit.ai/getstarted.html>

**Data Analysis:**

*Author*–

We considered four components(TreeView, Dialog, MultiSelect & DatePicker) and analyzed the documentation and frequent queries asked related to the components. We compiled the data together and stored it in our application.

**Mapping the key values based on custom search:**

*Author*–

Analyzed the frequently searched keywords by users related to these queries in google. With that data, we mapped the related key values with the results for custom search.

**Filtering:**

*Author*–

Wrote feasible algorithm for filtering the results based on user’s input and provided exact or related results.

**Response based on customer query:**

*Author*–

Provided response to the user based on customer query. If user asks common questions like, company or company details we provide them with common responses. When they ask queries related to component or its features, we return them the related links. For queries not documented or not available, we direct the customers to contact Syncfusion support.

**Future Enhancements**

* Update new queries asked by customers to expand search results.
* Get customer suggestion to know whether the given information is helpful to them.
* Store customer queries to analyze their frequent requirements.