  
Illustration 1: Data flow diagram

The above diagram is the dataflow diagram meant for our chatbot.

It’s main components are the following:

1) The user / end customer who is represented by the smiley

2) The FB messenger or other messaging platform where in the user will send queries or text responses . It is the interface through which the user and the system will communicate.

3) The server which performs various logical , analytical and preprocessing tasks.

4) The api.ai layer which decides the intent and performs the NLU task.

The entire dataflow is explained via the example given below.

Smiley is the user

1. **User** messages the Anirban da’s **facebook page** “Hey! Can I have your contact? I want to know about paragliding in Sikkim.”
2. text\_unedited = “Hey! Can I have your contact?I want to know about paragliding in Sikkim”.
3. This is sent unedited via the messenger to our server along with other metadata.
4. **Facebook messenger** then sends it **to our server** in  the same manner as the user sends it, with meta data like the senderID (user psid (page-scoped id)), recipientID (the page), time of message, and other related meta-data.

1. **Our server** then **processes** the text.
   1. sanitizes it - remove useless symbols (preventing MySQL injection)
   2. breaks into sentences - smaller pieces for better NLP.

1. text = [“Hey!”, “Can I have your contact?”,”I want to know about paragliding in Sikkim”];
2. Then each sentence is **sent to API.AI if and only if the query doesn’t satisfy some rule-based query. For example , if the text is “ quiz 3”, the server will first look up to see if a quiz3 module exists or not. If yes, it executes that module else it sends the query to api.ai**
   1. it processes the text, tries to find the intent and entities associated with it.
      1. “Hey!” = { **Intent** : “Greeting”, **entities** : “Hey” }

fetches a response like “Hello/ Greetings”

* + 1. “Can I have your contact?” = { **Intent** : “get\_Contact” , **entities** : “contact” }
       1. **fetches response** - “My contact number is 999199123 and mail ID is [mailme@xyz.com](mailto:mailme@xyz.com).”
    2. “I want to know about paragliding in Sikkim” ={ **Intent:** “ search.blog” , **entities**: “paragliding, Sikkim”}  
       response would be in the form of one or more urls regarding the queries.

1. b. It might not find any intent, and in such case it returns the default fallback intent
   * 1. Response like - “I didn’t get that, could you try again?”
2. The processed data (success or failure) is returned back to server.
   1. if server receives response, it is sent to the corresponding user via the page. Multiple sentences can be clubbed together to form a single paragraph.
   2. if no response from api.ai, then-
      1. either a default fallback intent is fired
      2. or no response is sent.

Now the server has an important job of analyzing the queries sent to it. This is something we will think later after construction of a simple machinery.