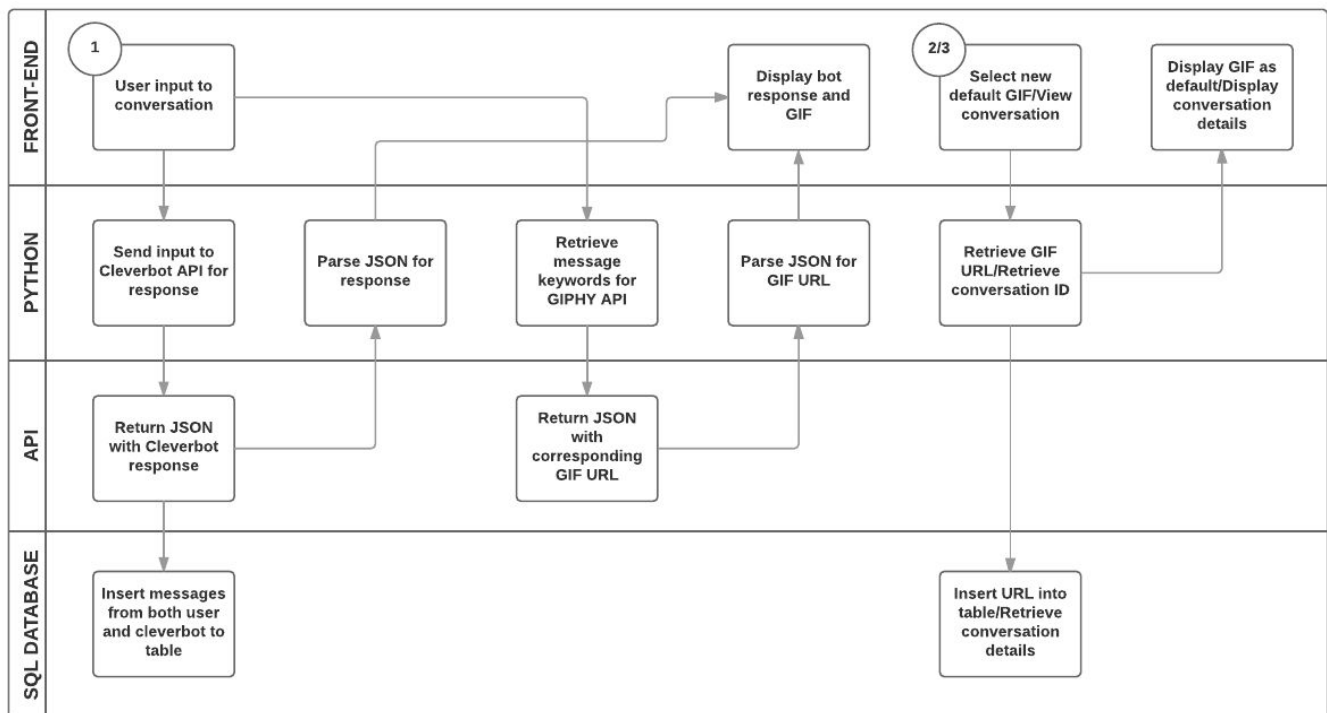# CLEVERGIPH: Bot Communication with Pop-up GIFs

By PancakeSafari - Adam Abbas, Terry Guan, Irene Lam, Shannon Lau (PM)
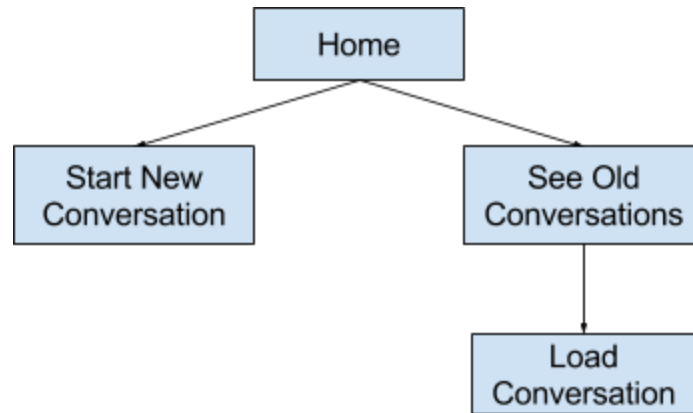
I. Summary

CleverGiph is a melding of the Cleverbot and the Giphy APIs. The user of our website will be talking to a Cleverbot (via API calls) while pop-up gifs will be generated based on the context of the conversation. This will be done by parsing through the conversation and finding meaningful words and then passing them to the Giphy API. The user can keep his favorite gif generated. The favorite gif will always be displayed within the conversation. Once a user quits the conversation, the conversation history will be stored but the user may no longer be able to respond to the conversation.

II. Component Map

1. User input is sent to Cleverbot API which returns a response in a JSON representation. Both the user and Cleverbot messages are stored in a SQL table. They are also displayed to the viewer. Also, for each message sent, message keywords are sent to the GIPHY API to retrieve a URL that displays the GIF.
2. User selects default GIF. Its URL is stored in the table and passed to the front-end display.
3. User chooses to view a past conversation. With the conversation ID, the conversation detailed are retrieved from the SQL table and passed to the front-end display.

III. Site Map

Page Visions:

Home

<table>
<tr><td>

Clever Gif

</td></tr>
<tr><td>

Start New Conversation

See Old Conversations

</td></tr>
</table>

Start New Conversation

Title

Hi. **Hello.**	[Gif representing most recent line]	[Previously selected GIF to represent this conversation]

Start typing here...

Save conversation	Save GIF

See Old Conversations

Conversation 1 (Timestamp)	Conversation 2 (Timestamp)	Conversation 3 (Timestamp)
Conversation 4 (Timestamp)	Conversation 5 (Timestamp)	Conversation 6 (Timestamp)
Conversation 7 (Timestamp)	Conversation 8 (Timestamp)	Conversation 9 (Timestamp)

Load Conversations

"Title"
How do you like the name "Clever Giph"? **It's ok.. Why?** Only ok? **Well I don't know what that is.** It'll be you plus gifs. **No.**   [Previously saved Gif]
[Return to Conversations](#)

IV. Database Schema

TABLE NAME: **threads**

Primary Key id	time_stamp	message_log	gif_url

0	2014-10-18
	21:31:12
		{
		  "cs":"112nxdylJO2...AAA",
		  "interaction_count":"2",
		  "input":"Hello.",
		  "output":"How are you?",
		  "conversation_id":"AYAR1E3ITW",
		  "interaction_1": "Hello.",
		  "interaction_1_other": "How are you?",
		  "interaction_2": "",
		  "interaction_2_other": "Good afternoon.",
		  ...
		}
			https://media.giphy.com/media/l0Hluv3yV9QRirYc0/giphy.gif

1			
2			
3			

V. Team Roles

Adam -- Frontend
- Create Bootstrap templates
- Work with Jinja
- Make sure things look pretty
- Load the conversation in a way that works

Terry
- Work with Cleverbot API
- Coordinate input and outputs between user messages and Cleverbot
  - Input user message and retrieve Cleverbot reply, using Cleverbot's id parameter
  - Keeping track of number of messages sent by user (Cleverbot parameter)

Irene
- More CSS, connecting webpages via app.py
- Python function that will retrieve message log and display appropriately for a specific ID
- Creating database Python file with the following functionalities/methods:
  - Creating the *threads* table
  - Adding rows to the table when the first message of a new chat is sent
  - Appending to the "message_log" string for each message from both user and Cleverbot
  - Accessing "message_log" for a given ID

Shannon (Project Manager)
- Working with Giphy API
- Creating the blacklist of prepositions (like "of", "and", "with", etc) to successfully choose a keyword
- For every message, choose a random word *not* in the blacklist and send to Giphy API to retrieve random corresponding gif for the user to select