Hugo Agnola
CART 451
ID: 40116274

Chatter Showdown

Chatter Showdown is an interactive streaming experience not between the streamer and the chat, but rather between chatters themselves. Twitch chatter's will "level themselves up" by simply chatting in a streamer's chat. I intend on implementing a simple turn based battle system in order for chatters to be able to fight amongst each other.

The theme I am tackling with this project is twitch chat interactions. Most chat interactions happen between the streamer themself, and the odd sentient entity referred to as chat. But rarely does one see chatters interacting with each other. Although this isn't necessarily the case among smaller streamers with view counts ranging from twenty to one hundred. In these streams, one may notice chat interacting amongst each other as their messages don't get completely flooded in the mass. This is a problem with big streamers, chat messages mold together as one into an entity that streamers jokingly denominate as the *chat hivemind*. When one enters the chat of a streamer with upwards of a thousand viewers, one loses individuality. They become simply a user chat among many others. The only way to stand out among this crowd would be to pay money in order to subscribe to the channel or donate and attach a message that will then appear on the stream itself. This presents an issue as not everybody has money to just throw away in order to have themselves noticed. There isn't much inclusivity with this business model. Users shouldn't feel forced to spend money in order to make themselves distinguished, they already spend their time watching the stream itself. With this project, I hope to turn time and investment into a form of currency. After all, there is a saying that "time is money".

My project will take the form of a node/web application. For my first prototype, I will focus on functionality and proof of concept. I will first and foremost, establish how I quantify viewer time spent/stream investment. Since I will be using tmi.js (more on this in later responses), there is no way to check the total amount of time spent by a user but you can quantify the total amount of chats sent over time and store that digit into a database. The target audiance in this case would be the streaming community and in particular, the streamers themselves. Relatively larger streamers do not know 99% of the viewers in their chat, in fact they only remember a select few that have either been subscribed for a while, or donated large sums of money.

I intend on using a culmination of tmi.js, the twitch related javascript library mentioned before, Node.js, to host and run the server, express.js, for the sockets and app functionalities, P5.js, for the visual side of things (also its quite easy to use), and potentially Ngrok, for hosting the server itself on the world wide web (make accessible not only by LAN, but from anywhere). To start off, let's elaborate on tmi.js. Twitch messaging interface or tmi.js is a javascript library that pulls twitch chat data from specified channels and allows those utilizing it to develop bots and automations. @Shmoopiie on GitHub is the lead developper, based in Quebec, they are constantly updating the library itself to make utilization seemless and functional with each new

Hugo Agnola
CART 451
ID: 40116274

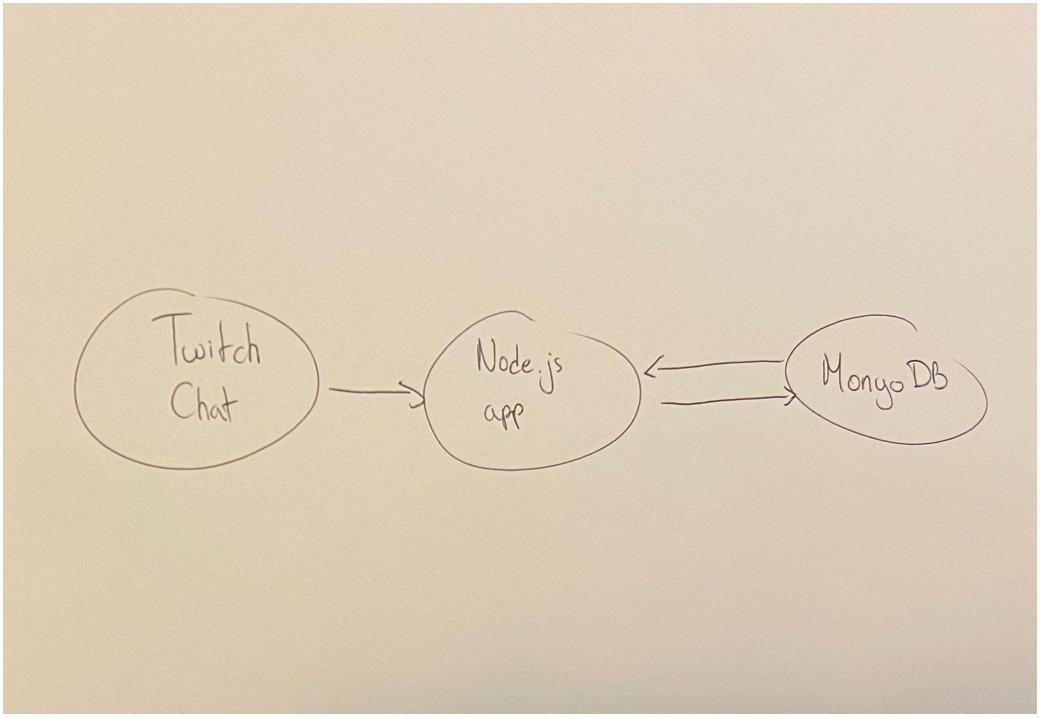
update from twitch. Node.js and express will serve their purposes by managing the application website's potential websocket usages, general requests (post, pull, push) and communication with the MongoDB database. An aspect I want to implement is cross-streamer profiles. Since a user's profile (chat count etc…) will be stored and updated in a general Mongo database, their total chat count will be counted across all streams where the streamer is implementing this web-app. P5.js is what I plan on using for the visual side of things, the representation of chaters as little characters, the fighting animations/sequance etc… And Ngrok in order to have this application available to all.

tmi.js will collect the user's message count data from a stream chat. In short, the more a user sends message in streamers' chat the more their "chatter power" will increase. Said chatter power will be quantified with a decimal number that is correlated to the total amount of chats sent by said user. One can think of this as leveling up a video game character, however instead of earning exp points to reach levels, the user's "level" will be represented by a decimal number.

To list some of the works that have inspired me, firstly I will denominate the SNES game Final Fantasy V. FFV is a turn based rpg which originated in 1987 Japan. The aspect of this game I found inspiration in is the battle system. Although dated and slow, it has this retro feeling and still retains a certain level of difficulty. The next project which inspired me is actually one of my own called "Twitch Pets". In my second year I discovered the tmi.js library and decided to play around with it for a final CART project. Here is the link for my demo video:

https://www.youtube.com/watch?v=Ssx3wuinaFI&t=71s

In this project, streamers battled each other using their "twitch pets" which their own chat helped level up. This project was quite gimicky however as it implemented a lot of random libraries that served no real purpose other then just "wow factor". So for this project I wish to focus on less libraries and prioritize seemless usability and functionality.



Hugo Agnola
CART 451
ID: 40116274