**Twitch Channel Guide**

***Explanation*:**

This web application would allow users to view a guide to streamers’ schedules that they have followed in an hourly view. To use this guide, twitch users would log in with their twitch account and a guide would appear with their followed streamers time slots. Each user’s guide could be interactive, use filters, and have customizations. Twitch users would also be able to navigate to their “channel settings” were they could set their schedule, future streaming titles, and customize their appearance on the schedule.

***What is new/original about this idea? What are related websites/apps? (Be able to answer the question: isn’t somebody already doing this?)***

On twitch you have to navigate through a few pages just to see who you follow that is currently streaming. You have to keep track of when your favorite streamers are currently streaming.

***Why is this idea worth doing? Why is it useful and not boring?***

This idea would increase viewership for channels because more users will know when a streamer is streaming and lower the hassle for twitch users planning what they are going to watch/what they can watch.

***What resources will be required for you to complete this project that are not already included in the class. i.e. you already have the Microsoft stack, server, database so what else would you need? Additional API’s, frameworks or platforms you’ll need to use.***

The Twitch API would be the main outside resource we would be using. Other than the API would depend on how we would implement the guide.

***What algorithmic content is there in this project? i.e. what algorithm(s) will you have to develop or implement in order to do something central to your project idea? (Remember, this isn’t just a software engineering course, it is your CS degree capstone course!)***

The main algorithm would be one that builds the schedule for the user. Additional ones would be added features (filters, etc).

***Rate the topic with a difficulty rating of 1-10. One being supremely easy to implement (not necessarily short though). Ten would require the best CS students using lots of what they learned in their CS degree, plus additional independent learning, to complete successfully.***

8