

The Scenario:

What is it that makes a topic so popular? Who is the most popular celebrity? What is the most searched animal on Google? How has the popularity of the top 30 searches changed in the past 30 days? The application I am proposing would not answer these questions. The users of the application would answer them. The application I am creating will have a user interface similar to that of a stock simulator, with graphs of past performance, a portfolio, and a watchlist. This application would allow people to "bet" on the popularity rankings or buy "shares" of different popular Google search results.

This application could help incentivise people to stay up-to-date on current events. This application could also help recovering gambling addicts. Another demographic that this application would target is those people that have and enjoy a stock simulator app. The desired interface of my application is very similar to the layout and graphs of those apps. This application would be more of a game, but it would be based on real trading searches on Google, just like how the stock simulators are based on real market trading activity.

My client would actually be my business partner, and would help me develop, release, and market the application. My client would also have a hand in developing the application, although I will be doing most if not all of the programming. When I talked to my client during the "initial interaction," my client liked the concept, and was talking over the details with me.

The Proposition:

I decided to create an application that allows users to "bet" on the popularity of future Google search terms. The users will have an amount of "money" to "invest" in different search topics. The users will have a "portfolio" of the topics they are currently "invested" in, and a watchlist for the search queries they want to keep tabs on. There will be graphs of past performance for the different search trends. There will not be any "spread," as there will only be one "price." There also will not be a "broker fee" or anything like that. There will be a percentages for gains and losses for individual searches and users as well. Users will also be able to see their gains or losses on a graph just showing the amount of "money" in their account. I think that the "money" will be the number of people who searched

that topic in millions. I will most likely only develop a website for this project, due to the time commitment.

I will be working in HTML5 and CSS3 for the frontend, perhaps on the Skel framework. I chose the Skel framework because it is light, open-source, and used by the project that I will be using to build the website. I chose to work in Python for the backend, because of my personal familiarity with it, because the APIs I wish to work with are written in Python, and for the Kivy resource to help develop the applications for multiple platforms.

Success Criteria:

- ❖ It must be intuitive (test on someone unfamiliar with the application/project)
- ❖ It must run off of real-time/real data: based on Google Trends trending searches
- ❖ It must allow users to “buy” and “sell” “shares” of search terms
- ❖ It must show the user the performance of their portfolio over time
- ❖ It must allow users to see “more details” on specific search terms such as:
 - Graphs of past performance
 - Percentage gained/lost
 - Related news articles
- ❖ It must have “Explore,” “Watchlist,” and “Portfolio” tabs.
 - With the ability to search for specific terms in the “Explore” tab
 - With the biggest gains and losses of the day in the “Explore” tab