Project Title:

Project Summary: It should be a 1-2 paragraph description of what your project is.
**Our project is an educational platform designed to simulate the stock market using live data. Most investment apps are confusing and have far too many features for a novice investor. We intend to make a simple yet easy to use paper trading app that anyone can get on and practice investing in a risk free way.**

**Users are provided with a fixed amount of virtual dollars to invest in stocks, allowing them to buy and sell shares based on actual market conditions. Once users are logged in they will have the ability to create several portfolios to try various investment strategies. Within each portfolio, the stocks they own will be displayed along with percent change and dollar change. There will also be a transaction log where users can see the prices at which they've bought and sold various stocks.**

Description: Description of an application of your choice. State as clearly as possible what you want to do. What problem do you want to solve, etc.?
**The main problem our web application is seeking to solve is to lower the threshold for people looking to get started in investing. Novice users should not be overloaded with complicated details right off the bat. Users will simply be able to create portfolios and buy and sell stocks at market prices within each portfolio.**

Creative Component: What would be a good creative component (technically challenging function) that can improve the functionality of your application? (What is something cool that you want to include? How are you planning to achieve it?)
**The main creative component we are planning to implement in this project is to utilize an API to request real-time data. This will allow for accurate stock information so the user can truly simulate investing in the market. Users can watch their positions' value change in real time. We also intend to implement charting features for users to easily visualize their stocks performance over time.**

Usefulness: Explain as clearly as possible why your chosen application is useful. What are the basic functions of your web application? (What can users of this website do? Which simple and complex features are there?). Make sure to answer the following questions: Are there any similar websites/applications out there? If so, what are they, and how is yours different?

**Similar Apps: While there are other investment apps they are cluttered with highly technical information that is overwhelming to someone just beginning in the stock market.**

**Options, different order types, margin trading, short selling, and other types of securities may be frightening to a novice retail investor. The purpose of this app is to strip out all of the excess features and risk and allow users to simply simulate buying and selling stocks at current market values. Investing is a critical means of wealth creation that should not be limited to advanced Wall Street investors. This is intended to open up the market to the everyday person.**

Realness: We want you to build a real application. So, make sure to locate real datasets. Describe your data sources (Where is the data from? In what format [csv, xls, txt,...], data size [cardinality and degree], what information does the data source capture?). It would be hard to satisfy stage 2 requirements with one dataset. Thus, we strongly recommend identifying at least two different data sources for your project.

**The data we need for our project is user data: Users, Portfolios, Positions, Transactions, and Logins. The user information includes emails, passwords, and addresses. Because this is sensitive data, the only way to proceed is by creating a synthetic data set. The plan is to use the Python library Faker. This can create large amounts of user style data that includes realistic emails and passwords. This is an ethical way of satisfying the project requirements. The other data points such as stocks will be assigned via other Python scripting. These will be used to track users' positions over time. The realness of this app comes to life with the use of the yfinance API to gather real time stock data from Yahoo Finance website. This information will be the company's stock price, the percent change and dollar change in the value of their positions. This data is returned from the API as a JSON file and is parsed by the yfinance library into Python dictionaries and dataframes. Based on the API response and the SQL tables, users will be able to see the value of their stock positions and buy or sell them at the real market value.**

Functionality: A detailed description of the functionality that your website offers. This is where you talk about what the website delivers. Talk about how a user would interact with the application (i.e., things that one could create, delete, update, or search for). Read the requirements for stage 4 to see what other functionalities you want to provide to the users. You should include:

- A low-fidelity UI mockup: What do you imagine your final application's interface might look like?
    - A PowerPoint slide or a pencil sketch on a piece of paper works!
    -

411Project Design

The user interacts with the app by first creating their account. They will need to provide a username, password, address, and phone number. Once their account is created users will return to the logging page to sign in with their new credentials.

Project work distribution: Who will be responsible for each of the tasks or subtasks? Explain how backend systems will be distributed across members. Be as specific as possible as this could be part of the final peer evaluation metrics.

Zack Edds: Backend development and API implementation. Develop back-end logic and API for handling stock information such as company's stock price, the percent change and dollar change in the value of their positions.

Emilia Michalek: Front-end and UI Design Focus on user interface design and implementation, including front-end development of user account management interface and personalized portfolios.

Sulabh Patel: Front-end and Database Management and maintenance. Focus on user experience with stock data and scenario simulation tools.

Nikhil Thomas: Backend and Database Management. Implement user authentication and authorization to ensure the security of data transmission.