Project Proposal: Bulls and Bears

Team 1:

David Patrzeba

Eric Jacob

Evan Arbeitman

Christopher Mancuso

David Karivalis

Jesse Ziegler

January 26, 2014

Hyperlinks:

Webapp Link
Project Repository
Reports Repository

Revision History:

Version No.	Date of Revision
v.1	1/26/2014

Contents

1 Team Profile

Team 1 will be working on Project 5: Stock Market Investment Fantasy League and have named their project "Bulls and Bears" for the tentative future. This project is intended to serve as an exercise in software engineering focusing on building experience in the design, architecture, construction, test, and maintenance of a small-to-mid sized complex software application.

At this time a project lead has not been elected, but David Patrzeba will be acting as technical lead. All members will have input on the decisions of the team and communication is being facilitated by a project mailing list, git repositories, a wiki, and google+ hangouts.

1.1 David Patrzeba

David is proficient with the Java, C, and C++ languages, RESTful APIs, SQL, and is highly familiar with iterative software design and object oriented design patterns. David also has experience with Android development, relational database schema, and user experience. David will be acting as a technical lead on the project.

1.2 Eric Jacob

Eric is proficient with the Java, C, and C++ languages, and SQL. Eric also has experience with Bash scripting and Python.

1.3 Evan Arbeitman

Nick is familiar with the C++ and PHP languages, Bash scripting and Python.

1.4 Jesse Ziegler

Jesse has experience with the C and C++ languages.

1.5 David Karivalis

David is proficient with the PHP, Java language. David is familiar with C, JavaScript, and HTML/CSS. David has experience in iOS and Android development, user experience, and photoshop. David will be the UI lead.

1.6 Christopher Mancuso

Chris is familar with the C++ language.

2   Project Proposal

Team 1 has elected to work on Project 5: Stock Market Investment Fantasy League with the goal of implementing a web application. We intend to use a RESTful API for executing all requests by the end user allowing easy expansion to the desktop and mobile application domains. We are deploying to a DigitalOcean Virtual Private Server (droplet) which will allow us to scale as necessary, both vertically and horizontally.

The goal of our web application is to act as the initial and primary interface to our backend services. These will include the ability to conduct buy, sell, short, stop, and limit orders on at least the NYSE and NASDAQ stock exchanges. We also plan to support global leader boards, individual leagues with goals (eg. first to double their money, first to gain 8% in a day, etc...), and global achievements (similar to xBox achievments).

Because our project will be built using RESTful principles, it will be able to act as a platform that can be extended by third parties to implement new and innovative features. Some scenarios include trading tutorials, and stock prediction integration. These are not of our primary concern.

2.1   Log In, Authentication, and Validation

The project plans to use OpenID in order to perform authorization and validation of our users. It also allows the project to easily scrape data from our users and prefill their profile for them. This should also allow the project to integrate easy with social media.

2.2   Social Media Integration

The project will initially integrate with social media to push out messages about trades and achievments. This feature will be able to be turned on and off by the end user, and will support multiple social applications.

2.3   Transactions Ticker

The project plans to implement a static transactions ticker which will scoll across the bottom of the users screen similar to what you would see on CNBC. It will be visible to all users of the website, whehter logged in or not. We will also include the indices of the Dow Jones Industral Average (DJIA), the Standards & Poor 500 (S&P500), and the NASDAQ in a static box also much like you would see on CNBC.

2.4 Data Source

The project will use a combination of HTTP Yahoo Finance API, and their Yahoo Query Language (YQL). This should provide some redundancy since YQL has been known to fail service some times. Since this information is provided free of charge, there are limits on the amount of queries we can run against these APIs, this limits the size of our initial userbase, but as the application grows, additional resources can be purchased. Another limitation of the Yahoo! data is it is out of date. This is fine for a beta run, but is easily "gamed" once the application goes live. This is a limitation that the project is willing to work with for now.

2.5 Mobile and Tablet Interfaces

The project plans to use Bootstrap to provide mobile first front-end for the website. By doing this, it allows the project to provide a consistant user experience across all platforms. This also serves as a first iteration into the mobile/tablet market and gives us a good blue print for a second iteration focused on native mobile/tablet applications.

2.6 Portfolio Graphs and News

The project plans to implement interactive charting for your portfolio using HighStock. This will allow end users to conduct analysis on their portfolio using visual aids. The project will also implement a news feed, which will update with the latest finacial news for companies that are in your portfolio, and companies that you may be tracking.

2.7 Email Updates

The project will facilitate a variety of email updates to the end user. Some examples are transaction confirmation, daily, weekly and monthly portfolio updates, league updates, and site updates. The user will be able to select which email updates they would like to suscribe to.

2.8 Educational Interfaces

The project plans to implement definition boxes for investment terms across the website. Certain terms (eg. P/E) will be highlighted and by mousing over them a dialogue box will pop up with a definition and links to external resources for a more in depth analysis.

3   Product Ownership

We have decided to divvy up the group in such a way that each function is claimed by a particular team member. In this way, we allow for easy traceability for the future reference and to determine accountability. When the customer is in need of finding the "go to" person for a specific function of the project, this will be the person to consult. Of course, this is not to say that each team member works in a vacuum. As each person's particular skills vary, one may contract out some part of his own function to another team member. In fact, as many functions will likely interface with one another, it will almost certainly be required for some cross-functional cooperation. However, it is important for the person with ownership to be very particular about the design and understanding of his function.

Each person is responsible for unit testing his own projects.

3.1   Portfolio Functionality

Nick Palumbo will be taking responsibility for the portfolio functionality of this project. One main aspect of this is the way a user is able to enact trades within their own portfolio, including basic transactions such as buy and sell as well as more advanced actions like short/cover and stop/limit. In addition, Nick will be responsible for the structure of leagues and the way in which users create and join leagues.

3.2   Social Functionality

Dario Rethage will be taking responsibility for the social functionality of this project. As this application will involve grouping users into leagues and competition, it is important for users to be able to interact with one another in the form of viewing one another's portfolios, leaderboards, and the live activity stream. Dario will be in charge of implementing these functions in a way that is intuitive and clean as well implementing a commenting/messaging system between users.

3.3   Interactive and Graphical Functionality

Val Red will be taking responsibility for the interactive and graphical functionality of this project. This includes being able to view one's own portfolio as well as the finances of his league mates in an easily maneuverable and manipulatable way. In addition, Val will be in charge of the way users follow the trends of a company's stock performance and view any other relevant data pertaining to a company, industry, or user.

3.4   Account Functionality

Eric Cuiffo will be taking responsibility for the account functionality of this project. He will be heading functions such as registration, Facebook integration, and user profile settings. Also, he will implement account management operations such as changing passwords or account deletion. In addition, he will create ways in which users can receive information about their accounts, such as downloading personal portfolio reports or the e-mail updates described above.

3.5   League Functionality

Jeff Rabinowitz will be taking responsibility for the league functionality of this project. Pertinent functions here will be the powers and actions available to the league manager of a given league, such as league settings and player management. Jeff will implement the different league settings available to a manager, as well the ability to create league-wide announcements and moderating comments within the league.

3.6   Administrative Functionality

Jeff Adler will be taking responsibility for the administrative functionality of this project. This essentially covers all the power available to a site administrator. He will be responsible for the ways in which an administrator can manage leagues, users, and user interactions. He will also generate the way in which site administrators can view statistics about the website, such as user count, active leagues, transactions, new accounts, popular stocks, and other relevant data that will dictate product decisions down the road.