Fall 2016 CSE 4233/6233 – Software Architecture & Design Paradigms

Trade Net Requirements Overview and Scenario Description

Company: Trade Net Brokerage and Financial Services

Scenario:

You have been asked to lead an internal software development team to build a system to enhance your company's market competitiveness. You and your team are employees of Trade Net Inc., an online discount brokerage firm looking to develop a software platform to enable customers to manage their accounts and execute stock trades. The project must balance reasonable system cost, development effort, timely delivery, software quality, and functionality.

Trade Net provides brokerage services (i.e., the ability to buy and sale shares of a company) to its customers. To enhance its offerings, Trade Net recently purchased a small Local Bank and their banking software, previously limited to checking and savings accounts. This software will be extended to encompass Trade Net brokerage accounts to easily allow customers to transfer funds for trading. Trade Net customers can use their brokerage accounts to execute buy and sale orders for stocks they are interested in trading. Trade Net is looking to gain market share from its major competitors, ETrade® and TDAmeritrade by creating a cross-platform software system that enables quick customer execution of stock trades on 2 major stock exchanges (NYSE & NASDAQ). Trade Net customers are charged a small commission for each execution and also pay for access to streaming stock data (e.g., price charts) and company financial information. Previously, customers had to phone in to speak with a Trade Net representative to execute orders.

Requirements Overview:

The requirements engineering team has yet to completely specify all system requirements. Many of the system's features have been determined (as described below), but the VP in charge of development wants the system incrementally developed as requirements are finalized. Here is what is currently known about the system's needs.

Trade Net's new trading application will allow access to the markets via a web browser, a standalone (Mac and PC) application, and both iOS and Android mobile apps. The client apps enable stock charting, technical analysis for strategy development, access to company fundamental data (e.g., financial reports), news, related twitter posts, brokerage account management, and trade execution. In addition to the client apps, Trade Net has to extend its existing legacy Account Management (AM) server, which will integrate with a new Trade Execution server. The Trade Execution (TE) server will access real-time market data via a direct stream to the stock exchanges' market data providers to obtain share price information and the best stock bid (buy) and ask (sell) prices for its customers. The TE server will also execute trades on the exchanges after verifying the availability of customer funds with the AM server. The TE server will authenticate customer login info using the AM server. Both servers will store their data in a central database. This data includes historical market data, all transactions, access logs required by federal regulators, and customer account information. Each client application relies on a third party charting and financial graphing library to generate charts and integrate charting technical analysis. The desktop/web client applications also allow direct access the Dow Jones newswires for real-time financial news updates and Twitter for social updates on the tickers they are researching.

Fall 2016 CSE 4233/6233 – Software Architecture & Design Paradigms

Trade Net Requirements Overview and Scenario Description

The AM server facilitates cash transfers using standard banking ACH and Wire Transfer protocols and can transfer funds with any other US-based bank implementing the protocols. Securing customer data and transactions is vital to company success. All communication between nodes should be conducted using encrypted transmissions and verified, open authentication protocols.

The Trade Net Application holds **Security** as its most pertinent quality attribute. The software should also be Flexible (**Maintainability**) for added features and additional types of client interactions and accounts to be added in the future. **Availability** is also important as customers require consistent uptime to place trades. These quality attributes should be considered and the architecture's appropriateness to meet these attributes documented.

Initial Requirements Backlog:

- Trade Net market data source —> Tradier - see https://developer.tradier.com
- Use mailboxlayer API to validate email address at login - https://mailboxlayer.com/
- Create web (or mobile or desktop) GUI client to access and manage Trade Net customer brokerage account information, trading interface, and trade executions
    - Clients app functions:
        - Stock data (ticker symbol, last price, exchange, daily net change, volume)
        - Company fundamental data access (Company name, financial grade, current rev & profit)
        - Trade Execution
            - Buy, Sell
        - Account Management
            - Account Balance
            - Total Profit and Loss (P&L)
            - Add Funds
        - Portfolio Information
            - Stocks owned
            - Purchase Price
            - Current Price
            - Current P&L per stock
            - Current P&L
        - List Transaction History
        - Financial / Stock News Headlines (API / source determined by dev team e.g., Associated Press, Tumblr, Reddit, others)
        - Twitter updates for ticker symbol (e.g., $AAPL, $MSFT) - https://dev.twitter.com/overview/api
    - Optional Initial Requirements
        - Integrate captcha to login
        - Use Quandl API to build daily price chart and display
        - oAuth Login using GitHub (or other account e.g., Google, Facebook)
    - Database Storage
        - SQLite - local storage

        - PostgreSQL / MySQL - soma.cse.msstate.edu
        - Other
    - Server Access (if needed)
        - Account on soma.cse.msstate.edu - Ubuntu Server 16.04 - open ports required and required software
        - Google Cloud Platform - Educational Credits
- Account Management Server (Database)
    - Access Customer account database (username, email, name)
    - Process and store customer portfolio information (stocks owned, prices used to calculate P&L)
    - Transaction history
- Trade Execution Server
    - Access Customer information and legacy data from Account Management Server
    - Communicate with Tradier Network for Market Data
    - Access market data for any listed security (equities only)
    - Execute trades using prices for the listed securities
    - Determine buying power and trade execution based on actual account info

It is expected that the software architects responsible for development will make assumptions regarding the appropriate implementation of the features described (e.g., derived requirements from the scenario description). The architect uses his/her previous experience and education to reason about requirements and determine the appropriate frameworks and technical approach needed to design and implement the system. Any assumptions should be stated in the architecture description document.