Team HMKG

Project : StockWatch

Members : Mike Hutz, Mitch Moir,
Corah Krantz, and Jordan Gumby

Who is StockWatch?

Summary:

StockWatch is a web-based application intended to provide a platform for customers to monitor current values in individual stock sectors and allow for customization of their dashboard for which stocks they are viewing. By using the cloud, a safe location and record of the stock's current and previous values in a constant stream will be kept, a stable and effective monitoring tool will be provided, and historic stock values from a given timeframe can be stored. This cloud-based system will allow updates to the values of each stock consistently and quickly on the backend, to ensure customers get the most up to date information. StockWatch will provide customers with fast results, accurate information, and easy to read line charts of the stocks history. With efficient deployment and easy use, the program will be an easy for anyone to set up and use.

In this demo, a few set stocks running on a simulation style value generator are utilized. This simulation will be used on a small scale and may utilize a hard coded database for proof of concept. Hopefully, a timed execution that runs on the cloud to allow updates to happen at regular intervals can be created. HMKG looks forward to bringing you a polished product.

Chapter 1: Design
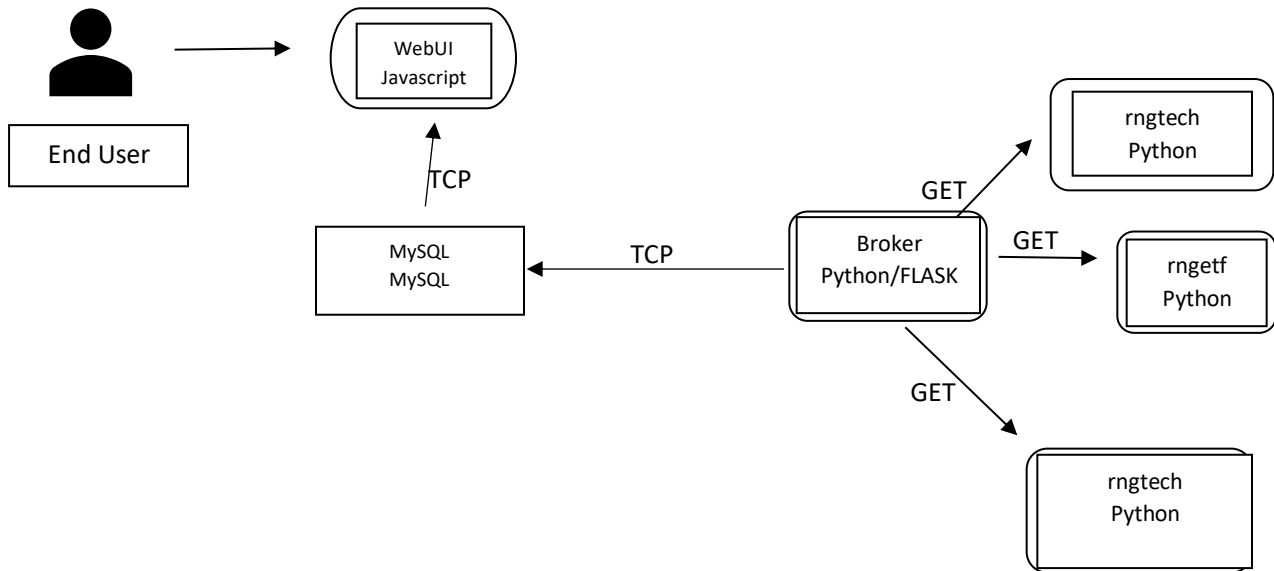
Chapter 2: Proposal

StockWatch intends to create a quick interface for customers to monitor, track, and research historic values of individual stocks in a live updating setting. Use of the cloud ensures that the most up to date information can constantly be available to the customer upon their request.

Utilizing Redis as the DBMS, the most up to date information can be tracked and reported, all while saving the historic values. This option was chosen due to group familiarity with this technology and the ability to interface and run a remote server. The Broker API will take on the bulk of the work. It will be running GET to three separate random number generation programs to obtain the weight of change for the stock. Each random number generator is tailored for the three types of stocks that will be available in the technical demo. These will each hold a different weight of change. The change will then be calculated, and the new value sent to the broker, and then passed to update the value of the stock in the database and in turn the WebUI.

The bulk of the program will be written in Python for both the Broker and RNG functions. HMKG has familiarity with the programming language. This consistency will allow us to be more thorough in our code and ensure that the team will be able to manage the cloud server.

Kubernetes and Jenkins create the pipeline for the deployment of the application. This allows for easy set up and deployment of the program's services and components. Random number elements of the program from our initial deployment will be kept in a pod. Then, the API will utilize those programs to allow for the constant updates After this, a Jenkins connection will be added for a full Kubernetes deployment. This will allow for a smooth deployment from the HMKG GitHub repository. There is not a need to run too many separate pods, since the

programs are similar and only contain minor changes. The three random number programs also utilize the same exact services, meaning it can be used across all three from a single build.

The original intention for our data and subjects was to use web scraping to ensure the most up to date information, but this has been removed until a final production program would be launched. HMKG intends for proof of concept to utilize a simulation that could easily be converted to a real-world implementation. The program has been built in such a way, that when it is time for the product to scale up into a larger sample and using real world information, it would be easy to make those changes.

HMKG looks forwards to bringing forward a polished technical demo and providing the best experience for future StockWatch users.

Chapter 3: Intermediate Milestone

HMKG has been working hard on the creation of a fully functioning and secure application. The development process has been particularly tricky due to the need to learn frontend development from the ground up, as no team member had prior experience in that area. A large amount of time has been spent developing JavaScript skills. Several working modules have been created to be integrated with the WebUI upon its completion, but the WebUI component is not currently fully functioning.

Regarding the backend development, learning how to launch a fully functioning API was another challenge. Currently, three random number generation programs have finished designs. These RNG programs will calculate weight of change and apply the change to the current stock values. They utilize a random number generator that uses separate ranges for each time of stock to simulate various levels of volatility. The RNG programs, simple in design, will be utilized by StockWatch's API, called Broker, to update and process the information to be sent back to the WebUI. An array will be used as the test format to store historical values. These values will allow for the user to view the trends and changes to the stock's price. In production, these values will be store in an individual table for each stock and allow for the customer to look up ranges of information to get the most detailed historical values. The API, will constantly run the GET requests for new values at a timed interval using a loop and will continue to run until the process is interrupted. This may be visualized as a dashboard for keeping track of stock prices in the WebUI. Additionally, a Redis image has been built. While Redis is StockWatch's DBMS now, it is possible it will be switched to MySQL for future iterations due to HMKG's belief that Redis is the better program for StockWatch's current scale, but that MySQL would be a more effective program for future iterations of StockWatch.

Proper usage of Kubernetes and Jenkins to deploy a functioning program using the pipelines was a large hurdle to overcome. After some research, the yaml and service yaml files were configured correctly, and now the project has a fully functioning pipeline. Each program was separated into its own pod to increase modularity of the program. This will allow for easier upgrades in future iterations.

Challenges face by the team have been briefly mentioned in this chapter but here they will be elaborated upon further. When StockWatch's development began, the team did not have any prior front end development experience. This required a significant investment in learning JavaScript and the surrounding utilities and has cut into development time. There was also an

issue with one of the intended services. Development began with MySQL as the DBMS, and it is still intended to be part of the final product, but due to configuration issues there has been no successful connection to the service. A switch was made to the Redis system to test proof of concept as a temporary fix. These are two of the larger challenges faced so far in the development process. Other challenges have included time constraints and scheduling.

A mock website for the program has been created. The test site we have made is a simpler version of the final product. The framework for the calculations and updating of values for the stock pricing is complete and is in the process of being integrated into the WebUI. The API is also finished and ready for testing once the WebUI program is running. Progress has been made and is on track based on the previous goals set.

The Kubernetes and Jenkins deployment worked and was able to deploy completely, but due to the nature of the program, it was not able to successfully execute the calculations through request functions. It is possible to post to a Redis image and observe the post, but the program itself does not perform the actions needed.

Although the beginning of development was a slow process, the team was able to adapt to the new tech stacks and create a solid foundation to build upon for StockWatch's future.

Chapter 4: Final State

The final state of the project has met its technical requirements with mixed results. As a development team, StockWatch has come a long way since the onset of the project but still has a way to go. The first such requirement is the interactive WebUI. This was the most difficult part of this project as no one on the development team had prior experience with front end programming. This was evident with the final state of our project not having a functioning WebUI available for it. Research was done on how to resolve this, and work began on development of mock-up's but we were ultimately unable to meet that goal as defined in the technical requirement. A module was working that would display the line graph but making the connection to the API was unsuccessful and there was not enough time to get that portion of StockWatch fully operational.

The second technical requirement was the use of a DBMS to read the information being sent from the API and store it for translation to the WebUI. Stockwatch had successful integration of Redis and could create a working image of it. This image was able to use a simple request to pull a hardcoded number to show that it does work and that it is able to take in the values being passed. Due to constraints with the RNG programs, while it was shown that the value is updating and cycling through loops of the API job, modified values couldn't be shown. The initial plan of MySQL was not implemented in the end. As a team, HMKG determined Redis was the right tool for our scale, and that if StockWatch were to grow in scale, and once a web scraper would be implemented, then an implementation using MySQL should be considered. With a working implementation of Redis in the pipeline that can connect to the API and be updated, this technical requirement appears to be accomplished. There are some bugs that need to be worked out in the RNG functions before it can be said with certainty.

The third requirement under review is the Broker AP, and whether it can successfully send the get requests and operate in a loop. The Broker API, running on the Flask framework, hinged around one loop that would execute a series of GET requests to the RNG programs. This loop

was functioning and successfully able to initialize the values needed, but when it attempted to execute the GET request that performed the calculation, it was unable to pass the correct variable. This is a less a restraint or issue on the API and more a technical issue with HMKG's knowledge of the requests function utilized. The API, once started, can execute in a loop until a termination signal is sent. This can be monitored using a log entry in Docker that allows verification of the number of loops completed. This loop runs on a 5 second delay and can stay in that loop for as long as needed. Were the requests functions working fully to access the RNG calculation, the customer could receive the most up to date values in a constant and reliable fashion. The technical requirement of developing a functioning API that is both reliable and consistent has been mostly met, with the only flaw being that the calculations were unable to be executed in their entirety.

The fourth requirement is the RNG programs that were intended to update the stock values in an appropriate rate of change. In testing, the command line version of all programs worked but ultimately Request in the API was used to send and receive values from the RNG programs. This was unable to send the values looping through to update, but that is more due to the nature of Request as mentioned before. Despite troubleshooting and testing different types and casting configurations, this issue was unable to be resolved. The initialize request was functioning and a connection created between the RNG functions and the broker API. Each RNG function was able to appropriately estimate and calculate changes based on a randomized and weighted value generated in each loop. This technical requirement was mostly completed, as there are three fully functioning programs, and most issues came from passing arguments to the RNG programs using the Request function in the API. As standalone programs, the RNG programs are fully functioning, and allow for a simulation environment for the stocks to prove the viability of StockWatch for stock trading but have some bugs that need resolving to fully function in the scope of the entire application.

The final technical requirement is the implementation of a cloud-based deployment that will allow for the platform to stay up and running and allow for the constant updates to the live stock price. The pipeline was built using Jenkins and Kubernetes and have a nearly fully functioning back-end program for the application. The service yaml and yaml files were difficult to figure out at first, but now all functioning parts are running in separate pods. Separate pods were used to ensure increased modularity of the program so that if one pod fails, the entire system would not go down. With this, StockWatch can prove that it is consistently able to keep running and processing the API loop. Cloud deployment was the most foreign concept that the team needed to tackle yet were successful in doing so. Comparing the results to the requirements set for the team, creating the foundation and groundwork for a consistently updating web-based application that is reliant on stability and consistency was a success. The team was able to publish and deploy the modules, and when running on Docker Compose, was able to keep the program looping for an entire twenty-four-hour period, which serves as an example of its reliability.

Overall, StockWatch has come a long way since the early stages of development. There is a functioning batch of rng programs, a functioning cloud deployment, and a functioning DMBS deployment in Redis. With those three elements working, the bulk of the requirements are met. Our broker API is mostly functioning, with just a few errors that need to be resolved for the application to function as expected. The WebUI is where the project fell short of the technical requirements. With extensive research and sandboxing, strides were made in the team's ability to create a front-end but were ultimately not enough to get a functioning and interactive WebUI

for StockWatch before the deadline. Overall, HMKG believes most of the technical requirements set by the team at the onset of the development cycle have been met, but still have a few aspects that need further refinement.

JORDAN GUMBY

jg905807@wcupa.edu | linkedin.com/in/jordangumby
(610) 883-7569 | Royersford, PA 19468

EDUCATION

West Chester University   **West Chester, PA**
Department of Computer Science   *Anticipated Graduation: May 2022*
Bachelor of Science in Computer Science, Computer Security Certificate
Term GPA: 3.59

TECHNICAL SKILLS

Java, C#, Web Design, Amazon Web Services, Python, Microsoft Office

WORK EXPERIENCE

Cisco
Remote
Technical Sales Engineer Intern   *June 2021-August 2021*
- Developing key insights into Cisco technologies.
- Working efficiently in teams to present key initiatives to System engineers.
- Participated in Hackathon to develop a custom security solution for a mock company.
- Led a discovery meeting in a roleplay scenario.
- Worked alongside with a field team to come up with a challenge project for high school students.

Self Employed   **Royersford, PA**
Science & Math Tutor   *August 2020 – May 2021*
- Assist students with Math assignments, mentor students on math skills needed for successful class completion.
- Utilizing creative approaches to assists students in finding solutions in an open and inclusive environment.
- Responsible for making sure student achieves learning outcomes.

YMCA   **Phoenixville, PA**
Camp Counselor   *June 2019 – August 2020*
- Planned and supervised activities associated with the assigned camps.
- Provided instruction to campers used in daily routines including wrap up activities.

RELEVANT COURSE WORK

- Computer Science I, II, III
- Data Structures and Algorithms 241
- Computer Security and Ethics 301, Malware Analysis
- Discrete Mathematics, Foundations of Computer Science
- Software Engineering

PROJECTS

DOE Challenge Project
- This summer I worked with the public sector state and local education-NYC. We presented a challenge project to high school students where we asked them to present to us a smart classroom using IoT devices. I helped other people on the team come up with a set of questions for the project as a guideline for the students. I am also in WebEx spaces with the kids for them to ask any questions and to provide any feedback needed.
- Reference: Carlos Aued caued@cisco.com

INTERPERSONAL SKILLS

Teamwork, Problem Solving, Leadership, Collaboration, Effective Social Skills, Responsible, Focused

Michael B. Hutz

Michael.b.hutz@gmail.com
114 Starr Road, Landenberg PA 19350
(610) 256-1588
www.linkedin.com/in/michael-hutz-wcucs/

EDUCATION:

West Chester University of Pennsylvania – West Chester, Pennsylvania
Graduation: Expected May 2022

- Bachelor of Science – Computer Science
- Computer Security Certificate
- GPA: 3.87
- Honors: Summa Cum Laude

University of Delaware – Newark, Delaware
Graduation: May 2018

Bachelor of Arts in Communications
- GPA: 3.52
- Minors: History and Advertising

Relevant University Coursework: Computer Science III, Data Structures & Algorithms, Systems & Design, Database Management, Cloud Computing, and Software Engineering

---

PROFESSIONAL EXPERIENCE:

Travelers – TLDP Intern   June 2021 – August 2021
Hunt Valley, Maryland

- Leading backend development of a tool designed to search and pull XML documents from an existing database
- Designing and building troubleshooting technical documentation to assist my team in current setup processes and for future employees
- Gaining experience in a variety of technical tools such as SQL, React, Angular, node.JS, Python, AWS, and Databricks
- Assisted in teaching fellow interns in various technical areas such as Git, CLI Interfaces, and Python.
- Took part in a hackathon and assisted in developing an updated contact directory for the company.

Apple Inc. – Technical Expert   August 2019 – Present
Newark, Delaware

- Provide hardware and software support and troubleshooting for customers of iPhones, Mac computers, iPads, and other Apple devices
- Received recognition from management as an exceptional team member for Q4 of 2020
- Have consistently maintained an evaluation score of 90 out of 100 over the last 5 quarters.
- Received a promotion to Technical Expert because of teamwork, leadership, and production in the store.
- Maintained inventory of existing repairs and necessary parts to complete future repairs.

Encore Capital Group – Corp. Communications Intern   June 2017-
August 2017
San Diego, CA

- Helped to manage Internal communications of a global company over a three-month span of time
- Led an audit of company's social media and media presence to ensure compliance restrictions were being followed
- Repaired portions of the website including broken links, UI upgrades, and content flaws
- Managed social media presence through web-based applications and platforms

---

PROJECTS:

Lexical Analyzer – Java
- Created a Lexical Analyzer to read a file and break the contents into readable components
- Analyzer utilized an arbitrary syntax framework to determine which symbols and patterns to read
- Utilized Linked List structure to read in and process characters from the input file

Insurance Database Search Engine – Python/SQL
- Utilizing an existing database in S3, created a web application to search for information using key search criteria and return the valid entries.
- Collaborative program written utilizing Python and Athena for the backend and Flask for the front end.
- Valid entries returned as a list view on the web application with an option to download the entry in an XML format.
- Individual contribution to the team was the lead backend designer. Developed the code to execute the database search, filter tool, and XML conversion.

Go Game – Java
- Java program utilizing Map structures to create a game board for Go, a strategy game where the goal is to surround your opponent's pieces.
- Game board automatically processes pieces and scoring locations to provide a final tally and winner of the game.
- Originally was created as a project in Data Structures lecture, shifted over into a personal project.

SKILLS
- Programming: Java, Python, React, SQL, Angular, Flask,
- Operating Systems: Windows (10,8,7), Mac OS, Linux, Powershell/Console
- Technical: Computer Hardware & Repairs, IT Servicing, Troubleshooting, Networks
- Data: SPSS Statistics
- General: Microsoft Word, Microsoft Excel, Microsoft Powerpoint

CORAH KRANTZ

570-614-9615| corah99@gmail.com | www.linkedin.com/in/corahkrantz

EDUCATION

West Chester University, West Chester, PA   **May 2022**
Bachelor of Science in Computer Science – *Security Certificate*   **GPA: 3.8/4.0**

WORK EXPERIENCE

Cisco Systems   June 2021 – Aug 2021
Technical Sales Engineering Intern – Global Enterprise Premier, Remote
- Interviewed each account team within the Premier Organization, created account specific technology heat maps and identified overall trends, presented findings to engineering leadership
- Learned about Cisco's solutions through technology trainings and shadowing opportunities
- Researched and presented concepts such as Kubernetes, Azure, and Terraform to Systems Engineers
- Developed key sales skills such as communication and active listening though simulated customer meetings

Justus Home and Garden   June 2019 – Aug 2020
Customer Service Representative, Justus, PA
- Consulted with customers to determine their needs and direct them to products or services
- Coordinated with team members from different departments to ensure timely delivery of services

PROJECT EXPERIENCE

Cyber Security Blog   Sept 2020 – May 2020
- Designed and maintained website using Wix, providing bi-weekly updates on cybersecurity trends
- Documented how cyber-attacks have increased with the onset of pandemic induced work from home policies
- Provided easy to read articles for my class to keep them informed on security incidents

Data Collection Research Project   Sept 2020 – Nov 2020
- Collaborated with teammates to research and present how businesses can use social media and other methods to collect data from customers and their motives behind it
- Researched data mining, cookies, and collaboration between companies and positive and negative impacts for both the business and the end user
- Group received a grade of 100% for our efforts, and educated the class on data collection

Business Plan Project   Oct 2019 – Dec 2019
- Determined a need for parking solutions due to congested student lots and brainstormed solutions with teammates
- Designed a high-level application to meet needs with teammates, acting as CTO for the duration of the project

INVOLVMENT & ACCOMPLISHMENTS

Varsity Sports Teams   Spring 2017 – Current
- Led Varsity Girls Track as a Team Captain for two years, encouraging a fun and respectful environment for all team members - 2 time state medalist in Pole Vault
- Current member of West Chester University Women's Track and Field team - Silver medalist in Pole Vault at 2020 PSAC Indoor Championship – Bronze medalist in Pole Vault at 2021 PSAC Outdoor Championship
- Participated in volunteer work with the track team by working other sports team events, completing tasks like running the concession stand and checking event tickets

INTERESTS & SKILLS

Technical Skills: Java, Python, Intellij, PyCharm, Azure, Terraform, Kubernetes, Excel, Security, Malware Analysis, Cisco Packet Tracer

Mitchel Moir

2650 Mount Road, Aston, PA 19014
Phone: 484-802-0168
E-Mail: mmoir893@gmail.com

Objective and Personal Statement:

I am seeking a fulltime position as a Software Engineer where I can leverage and grow my technical skills, knowledge, and passion. I am dedicated, motivated, and persistent, having worked fulltime while taking classes and paying for my own education.

Education:

West Chester University of Pennsylvania, West Chester, PA
- **Degree:** Bachelor of Science in Computer Science (with a Certificate in Computer Security)
- **GPA:** 3.81/4.0
- **Expected Graduation Date:** May 2022

Delaware County Community College, Media, PA
- **Degree:** Associate of Science in Computer Science
- **GPA:** 3.57/4.0
- **Graduated:** December 2019

Relevant Coursework:

- Data Structures & Algorithms, Intro to Cloud Computing, Software Engineering, Modern Malware Analysis, Data Communications and Networking, Database Management Systems Computer Security, Software Security

Work Experience:

Wolff's Apple House, Media, PA **2008-Present**
- Initially hired as a cashier and now a member of the management team for this local business
- Key contributor performing many functions including weekend shift manager, forklift operator, tractor operator, and commercial motor vehicle driver
- Responsible for traveling and procuring product for the business, vendor relationships
- Involved with reception, display, and inventory of various seasonal product
- Significant sales/customer service experience in a fast-paced environment

Technical Skills:

- **Programming Languages:** Java, Python, JavaScript, SQL, C, C++
- **Relevant Software:** Docker, Kubernetes, GDB Debugger, Wireshark, MATLAB, WinDBG

Personal Projects:

- **Programming:** Garden Generator, Leetcode Coding Competitions
- **Other:** Ongoing study of the Spanish language