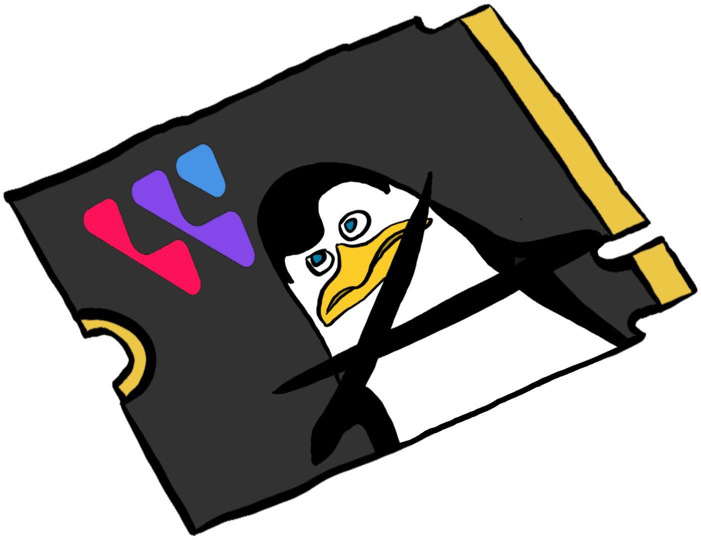
Requirements Documentation
V. 1.1
10/5/2023

Team
Kowalski

Sponser
Western Digital

Team Mentor
Saisri Muttineni

Team Members
Erick Salazar, Bailey McCauslin, Jake Borneman, Nick Wiltshire



Version 1
Accepted as baseline requirements for the project
For the client:______ Date:_____
For the team:______ Date:_____

Introduction

Western Digital is a company that specializes in the manufacturing of different data storage devices. Part of their manufacturing process is the analysis of each of their individual devices, whose entire data is stored in different data centers. Analyzing all the data is a time consuming operation, considering that every analysis has to be hard coded for each independent device that needs to be analyzed. Having this in mind, Western Digital has proposed a project that dives further into the creation of a system that is capable of automating this process.

The general process of development entails accessing the data in the data centers to probe it and obtain any wanted information for a particular device. Then restructuring that data into a file that can be uploaded to a database to be sanitized and formatted before displaying it on a visual dashboard. This process must have automated capabilities that allow it to run at particular times of the day, as well as, having synchronizability where several devices can be analyzed simultaneously.

As such, Western Digital partakes in the Technology industry, specifically the data storage and data solutions sector. Overall, the data storage industry encompasses a broad range of products and services related to storing, managing and accessing digital data. It strongly emphasizes the development, manufacturing and distribution of these technologies to provide better and faster hardware and software solutions.

Under the storage devices umbrella, be it hardware or software, we have large ramifications for different solutions. For instance, on the hardware side, we have Hard-Disk Drives(HDDs) and Solid-State Drives(SSDs), which are physical devices capable of storing and accessing data at different time efficiencies. And on the software side, we have things like Cloud Storage Services, such as One Drive and Google Drive capable of storing massive amounts of data on a web server.

As an overall industry, it has a massive market value nearing that of tens of billions of dollars annually considering it is a still continuous growing market. It houses a vast and diverse user base ranging from individual users to major large corporations and governments. As a whole, the industry itself handles exabytes (1 exabyte = 1 billion gigabytes) to zettabytes (1 zettabytes = 1 trillion gigabytes) of data, with its volume constantly expanding as per the needs of the users, and the fast development of digital content and the means necessary to store them.

Following is a brief overview of who Western Digital is. As previously mentioned, Western Digital works in the data storage and data solutions sectors in the technology industry. They are one of several companies that focus on the development, manufacturing and distribution of different data storage solutions. As a company, Western Digital is a Fortune 500 company with a vast annual revenue. They have thousands of employees globally working in areas of research and development, manufacturing, sales, and customer support.

They hold a widespread international presence having offices, manufacturing facilities, and distribution centers in various countries. They have 2 main sources of revenue, through hardware sales (selling data storage hardware like HDDs and SSDs) and by providing services and solutions through subscriptions (data storage and management solutions for enterprise clients like data backup, recovery, and analytics). Their general workflow extends across several sectors but they mainly focus on research and development, and manufacturing. They invest huge amounts of money into the innovation and creation of new advanced storage technologies, as well as, the improvement and merging of existing technologies.

Having this in mind, our sponsor/client for this project is Rajpal Singh, an employee at Western Digital working as an AWS Solutions Architect. His main job is using AWS services to analyze incoming data from the data centers and providing visual representations of them to improve the overall understanding of what the collected data is trying to show.

Problem Statement

Western Digital is a lead manufacturer of storage devices. They produce these storage devices, test them, and send them out for consumers to buy. How Western Digital creates their storage drives is first being created at a warehouse through a factory line. As each of the storage products are created, they need to test the appropriate performance of the product. The tests involve any error checking of the device, benchmarks that need to be achieved, and stress testing the products to observe any faulty and hidden performance issues. Once the product is given the green light in all areas, it would then be packaged and ready to be shipped for selling. This is the same process for all Hard drive and Solid State drives that Western Digital creates. Each hardware had to be manually tested at the warehouse, slowing the process down overall and preventing their focus on innovation.

The big issue in this workflow is that the company doesn't have an efficient way to test these devices. Each device is tested individually for any particular data the company wishes to examine and has no automation involved for it. With that being said it's very difficult to gather any sort of long term data, consistent data among all devices. This leads to an inefficient solution to an ongoing problem. That being said, it leads to several issues that needs to be handled. Examples of these are:

- **Detection of Silent Data Errors:** Are undetectable issues in computer systems where data is read or written incorrectly without any apparent warning or error message. They can only be detected by amassing long term data that would then point to an error in the device

- **Time Expenditures of Manual Testing:** Time is a valuable resource to the company, hence having no automation to analyze data coming from a device will result in a waste in man hours that could be allocated somewhere else

- **Lack of Data Consistency:** Each device being tested individually leads to a lack in consistency of the data which in turn makes it more challenging to analyze trends, identify common issues, and make data-driven decisions for device improvements

- **Limited Long Performance Monitoring:** The lack of standardized testing and consistent data prevents Western Digital from understanding how their products perform over extended periods of time, further hindering their ability to provide reliable products to their customers.

- **No Automation:** Having automation in the system will decrease the consumption of man hours when developing a data analysis for a particular device

Solution Vision

In order to tackle Western Digital's problem, we will be building a platform that condenses and automates their existing data analytics workflow. With our platform in place, engineers will be able to easily view important performance data and alerts, without having to individually collect, store, or query any data manually. Data will also be organized and stored in a way that facilitates temporal analysis, allowing engineers to identify issues that are hidden in real time.

Our solution will include these specific features:

- **Versatile SSD Probing**
  - Any drive running on a Windows or Linux virtual machine can be easily probed, without any extra scripting.
- **Flexible Data Collection**
  - The specific data that is collected from probed drives can be quickly configured to include only what is essential.
- **Automated VM to Cloud Pipeline**
  - Collected data can automatically be sent to AWS S3 cloud storage at any desired frequency.
- **Organized Cloud Storage**
  - AWS S3 buckets will be automatically organized to specification, whether by drive type, time, location, etc.
- **Master and Transactional Data Querying**
  - Master and transactional data will be automatically queried separately by AWS Glue and Athena, creating tables and databases containing only essential information.
- **Malleable Data Analytics System**
  - Data will be analyzed and compartmentalized, using AWS OpenSearch, in a way that can be configured differently in the future, as needs change.
- **Essential Data Observability Dashboard**
  - Data will be displayed in a way that highlights vital statistics and important thresholds, providing a visual summary of drive performance currently, and over time.

As mentioned above, the data that is being obtained is taken from the linux kernel level. This is where the storage products can be observed on their performance in which the probes will inject information for the storage drives to read and write all while collecting the information about the process for the product. The generated data collected from this process is the speed of which the injected data was written in milliseconds. It doesn't sound much important as this would only be one run through the data collection but when repeating the process hundreds to thousands of times, it would greatly assist in finding the previously mentioned silent errors. Having this program running in the warehouses for the employees would greatly increase their efficiency in the product testing as the program would automatically observe, collect and send off the data without human interaction. Though, this is also a tradeoff as the program would be automated, there is no need for an operator of the program unless it is to fix the program, possibly leading to less employment in this area and also potential errors the program could miss while a human observing could catch the error easily. To prevent this type of problem to occur, it is likely that the program would have to go through a tedious and repetitive debugging phase to ensure such errors would not fall under the radar.

Project Requirements

The overall general solution should automate data collection from different storage devices, making it easy for users to configure and probe drives without manual effort. Collected data should be sent to AWS cloud storage and organized efficiently. Users should be able to analyze the data easily using a user-friendly dashboard that provides a visual representation of the analyzed data. The system should handle errors automatically and ensure data security.

I. Functional Requirements

1) Automated Data Collection

- The system should automatically collect data from storage devices at the Linux kernel level without manual intervention.
- Data collection should include information about data read and write speeds, with the ability to repeat the process multiple times for accuracy.

Use Case For Automated Data Collection

The user/owner of the program will likely want to be away without complete monitoring of the system. With the program's purpose, it must be automated at a certain rate to ensure data collection is still performed when the user is not present. Upon the user returning, they would have repeated collections of data from the system over the different periods of time that the user had assigned to the program. Such as they want the program to collect every 30 minutes and the user leaves for 3 hours, the program would have collected 6 different data collections

2) Versatile SSD Probing

- The system should support probing of any drive running on Windows or Linux virtual machines without requiring additional scripting.
- It should be able to probe various types of storage devices such as HDDs (Hard-Disk Drives) and SSDs (Solid-State Drives)

Use Case For Versatile SSD Probing

Whenever the script is being used, the user can apply any type of hardware that the program can access and probe. In a situation, a user in a manufacturer wants to know the performances of three different products affecting the same hardware. All that would be needed from the user is to properly apply the hardware to the system. This then lets the program access the hardware with no problem even though the hardware has changed. The user would then be capable of seeing the results from the three different products they wanted to test.

3) Flexible Data Collection

- Users should have the ability to configure the specific data they wish to be collected, allowing them to only include information they deem essential to the analysis

Use Case For Flexible Data Collection

Upon the user's wishes/plans, the program would be easily accessible and readable for the user to modify. An example would be a hardware company that is responsible for creating different types of hardware. The user would then want to observe different aspects of the different hardware. The user would then be capable of accessing and modifying the program with ease.

4) **Automated Data Transmission**
   - Collected data should be automatically transmitted to AWS S3 cloud storage at specified intervals without manual intervention.
   - The system should establish a reliable pipeline for data transfer from Virtual Machines to the cloud.

Use Case For Automated Data Transmission

After activating the program, the user would receive a data file that was collected from the kernel level. This goes in hand with the “Automated Data Collection” as the program would send the data to the servers without the user present. The user would only need to properly setup the correct database that the data would be stored in. After accomplishing this, the program would continually send the information to the database without the user’s presence

5) **Cloud Storage Organization**
   - The system should automatically organize data in AWS S3 buckets based on predefined specifications such as drive type, timestamp, or location.
   - Proper organization of data will facilitate easy retrieval and analysis when needed.

Use Case For Cloud Storage Organization

For ease of access and readability of the files and data generated, the program would organize the data sent to the database servers. There would be a preset condition but upon the user’s desires, they could modify the collection and organizing of the data within the program.

6) **Data Querying and Processing**
   - The system should employ AWS Glue and Athena to automatically query both master and transactional data separately, creating tables and databases containing essential information.
   - Data querying should be efficient and capable of handling large volumes of data.

Use Case For Data Querying and Processing

Table schemas inside the respective databases will be created through the processing, ensuring a safe separation between essential drive information. This further enhances the ease of access to query each independent database to link any necessary information regarding a specific device that was probed.

7) Malleable Data Analytics

- The system should use AWS OpenSearch for data analysis and compartmentalization, allowing for flexible configurations to accommodate future changes in analysis requirements
- The Data Analytics should provide meaningful insights into drive performance and highlight any anomalies or issues occurring to it.

Use Case For Malleable Data Analytics

There is a variety of information and data that the user would like to specifically analyze, after the data collection process has been accomplished. The program would have the capacity to assist the user in analyzing the information they would like. With the requests of the user via by directly modifying the program or through the AWS database. The user would then receive their modified analysis.

8) Data Observability Dashboard

- The system should generate a user-friendly dashboard displaying vital statistics and thresholds that were configured in the probing stage
- The dashboard should offer real-time monitoring and visualization of drive performance, allowing users to quickly identify trends and patterns.

Use Case For Data Observability Dashboard

After the program has been completed, the dashboard would present all data in a compiled and cleaned format. This allows the user to observe and review the data the program has collected. The user would be capable of selecting, filtering, and displaying the data depending on the requests assigned. This development would give the user easier ideas as the display would give detailed information such as graphs, file locations, and raw data.

9) Automated Error Handling and Debugging

- The system should have an automated error handling mechanism that detects and reports on any issues in the data collection, transmission and analysis processes.

- ○ Automated debugging routines should be in place to identify and resolve any errors without any manual intervention

Use Case For Automated Error Handling and Debugging

This returns a robust data quality assurance system with automated error handling and debugging. This ensures that data collected, transmitted, and analyzed is reliable, leading to more confident decision-making and improved overall data-driven operations.

10) Security and Access Control

- ○ The system should ensure secure transmission and storage of sensitive data through the use of industry-standard encryption protocols
- ○ Role-based access control should be implemented to restrict access unauthorized personnel, ensuring data security and privacy

Use Case For Security and Access Control

As the data is very sensitive as the products being analyzed are normally in development or contain information the company wants to protect, security is involved. The user would need a secured connection between the transmission of the data from the program to the database storage. That way when the data is being transferred, the data is delivered safely.

II. Performance Requirements

- Data Collection Speed
  - ○ The system should collect and probe data from storage devices within seconds, ensuring minimal impact on device performance.
  - ○ The data collected about the system should be expected not to be above 8000 milliseconds (8 seconds) for a test on a probe. Anymore would be perceived as either a hardware problem or a program problem.

- Data Transmission Rate
  - ○ Data transmission to AWS S3 cloud storage should occur at a high speed, ensuring real-time or near-real-time updates for analysis.
  - ○ In the case of transmission failure from either timeout or transmission errors, a reattempt and backup of the file should occur on the local system to retry the transmission

- Data Querying Efficiency
  - The system should be capable of querying large volumes of data quickly and efficiently, providing rapid responses to user queries and short wait times for any future queries.
  - Also applies to the filtering and analyzing of data.

- Dashboard Responsiveness
  - The user interface should respond swiftly to user interactions, providing real-time updates and visualizations without delays. Any interaction should take no time to show any changes.

- Error Handling Time
  - Automated error detection and handling mechanisms should identify and resolve issues promptly to minimize system downtime and improve overall system efficiency and quality.

- Scalability
  - The system should be able to handle an increasing number of storage devices and data volumes without showing any significant degradations to system performance.

- Security Response Time
  - Any security protocols, including encryption and access control, should operate without any noticeable delays, ensuring data security without compromising speed.

- Time Taken to Analyze Data
  - All data analytics processes should be performed efficiently, providing timely and accurate insights to the users.

- Data Visualization Rendering
  - All visualizations on the dashboard should render quickly, allowing users to interpret trends and patterns without waiting for extended periods of time

- Automated Task Execution
  - All automated tasks should execute promptly per their predefined schedules without causing any hiccups in system speed and performance

III. Environmental Requirements

There are several restraints that have been imposed on us by the client, relating to the technologies we will be using throughout the development of this system. This are the following restrictions:

- **Development Environment:**
  - The system will be developed through a Linux based system (Ubuntu-based preferably) be it installed as a main driver for a pc, or used as a Virtual Machine.
  - Testing will be handle in both Linux and Windows to ensure reliability and consistency between pc systems

- **Supported/Acceptables Languages:**
  - Python3 (Used for the data analysis portion of the system)
  - Node.js/TypeScript (Use TBD)
  - Angular (Use TBD)
  - C/C++ (eBPF is a C/C++ syntax based language that will be used for probing)

- **Analysis/Sanitization Systems:**
  - AWS Open Search (Will take transactional data from noSQL database to analyze and organize it)
  - AWS Athena (will take master data from mySQL database to analyze and organize it)
  - AWS Quick Sight Dashboard (all organized, analyzed and sanitized data will be displayed as a visual analytic of the essential information the drives were probed for)
  - AWS Kibana (Use TBD)
  - AWS Glue (Data will be sanitized and organized using Glue)

- **Storage Services:**
  - AWS S3 (All Python analyzed data will be sent to S3 buckets to be stored, before the organization and sanitization processes begin)

Potential Risks:

Data Level Risks:

- ***Data Collection Level:***
  - Risks at kernel level during probe attachment.
  - Possible data loss if probes fail, impacting intended operations.
  - Logical errors in collection programs can corrupt kernel-level data.

- ***Data Analysis Level:***
  - Risks of wrongly interpreted data and incorrect analysis.
  - Possibility of data being organized in incorrect locations.
  - Storage risks such as primary/foreign key issues and data type errors.

Data Storage Risks:

- ***Database Cloud Storage:***
  - Accidental deletion or dropping of data when accessing or modifying databases.
  - Risk of separating master and transactional data incorrectly.
  - Potential for data duplications leading to readability issues.

Display Risks:

- ***Data Visualization:***
  - Incorrect proportions in graphs and charts can mislead interpretation.
  - Importance of scaling visuals appropriately for large datasets.

- ***User Interface (UI) and Experience:***
  - Risk of a bad UI impacting user experience.
  - Importance of clean and concise representation for effective project communication.
  - Regular UI development monitoring and feedback collection.

Project Plan

The project execution plan will include implementing an automated data analytics workflow for Western Digital's storage devices. We will develop a system that collects, organizes and analyzes data from storage devices that have been probed, automating the entire process from data collection to visualization. The system will be built on a Linux-based platform, utilizing technologies like Python3, Node.js and C/C++ for various aspects of the project. The collected data will be transmitted to an AWS S3 cloud storage, where it will be organized and processed using AWS Glue, Athena, and OpenSearch. The final insights will be displayed on a user-friendly dashboard created with AWS QuickSight.

Minimum Viable Product (MVP):

1. **Capturing Logs:**
   a. Collect data like Quality of Service (QoS), Latency, and Bandwidth from various Validation Test Suites.
   b. Gather data at both system level and per Virtual Machine level.
   c. Cover different Operating Systems (OS) and Workload scenarios.
2. **Storage:**
   a. Save these logs in JSON format on our on-premises Network Attached Storage (NAS) system.
3. **Data Transformation and Publishing:**
   a. Use eBPF tools to transform and publish logs into AWS OpenSearch.
   b. Utilize REST APIs for writing and reading data.
   c. Structure data in JSON or CSV format suitable for analytical and visual purposes.
4. **Visualization:**
   a. Enable log analytics and search capabilities on AWS Kibana.
   b. Create visualizations on AWS QuickSight dashboard based on the transformed logs.
5. **Monitoring and Alerts:**
   a. Implement monitoring for system resources such as CPU, RAM, IOPS, Bandwidth, and Latency.
   b. Set configurable alerts to capture and report any unusual events, especially when provisioning resources or running different

workloads on Virtual Machines. This could be accomplished by the user self inserting their thresholds into the program where key identifiers are observed or as a stretch goal, implement machine learning and set the alerts based on that.

In essence, the goal is to efficiently collect, store, transform, and visualize performance data from various tests and system levels. This process includes setting up alerts to promptly address any abnormal usage patterns or thresholds in system resources. All technologies mentioned are written in stone, but are still open to change.

Expected Milestones:

1. **Project Initialization**
    - Project planning and scope finalization
    - Meeting with client to clarify requirements and obtain assignments

2. **Data Collection and Probing Implementation**
    - Develop and implement scripts necessary for versatile SSD probing on Linux and Windows VMs
    - Implement error handling mechanisms during data collection
    - Implement automated data collection from different storage devices

3. **Data Transmission to AWS S3**
    - Implement automated data transmission to AWS S3 cloud storage
    - Implement reliable pipeline for real-time and near-real-time data transmissions
    - Implement backups and retry mechanisms for transmission failures

4. **Data Querying and Processing**
    - Set up AWS Glue and Athena for automated querying of master and transactional data
    - Implement/Revise/Ensure efficient data querying and processing capabilities (Ensure best performance possible)
    - Implement data organization capabilities based on predefined specifications coming from the user

5. **Data Analytics and Visualization**
    - Integrate AWS OpenSearch in the system for data analysis and compartmentalization
    - Develop a malleable data analytics system to provide the proper meaningful insights
    - Create a user-friendly dashboard using AWS QuickSight for visualization of analyzed data

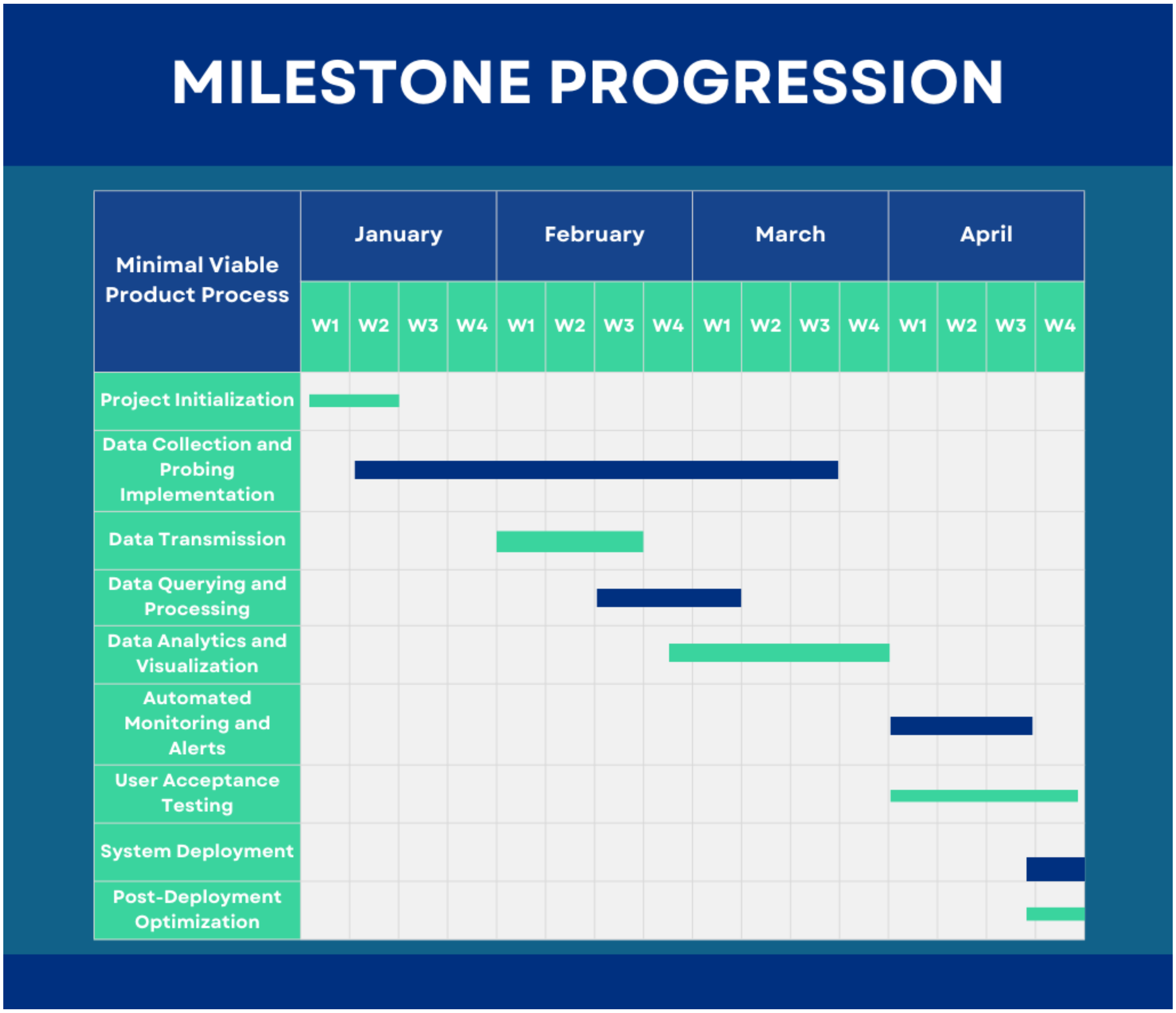
6. **Automated Monitoring and Alerts**
    - Implement proper monitoring for system resources including CPU, RAM, IOPS, Bandwidth and Latency
    - Set up configurable alerts for any abnormal usage patterns, behaviors and threshold events
    - Implement automated error handling and debugging routines

7. **User Acceptance Testing**
    - Conduct/Implement proper testing mechanisms to ensure system integrity and functionality
    - Address any issues or bugs identified during testing
    - Gather feedback from users on capabilities of the system or concerns they've developed

8. **System Deployment**
    - Deploy system as part of the production environment
    - Finalize system documentation and manuals

9. **Post-Deployment Optimization**
    - Provide support mechanisms to obtain any feedback addressing issues or concerns
    - Optimize system performance based on user feedback and noticed usage tendencies
    - Keep continuous monitoring of system to further enhance it if necessary

Gantt Chart of Expected Milestone Progression



MILESTONE PROGRESSION

Minimal Viable Product Process	January				February				March				April			
	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4	W1	W2	W3	W4
Project Initialization	🟩	🟩														
Data Collection and Probing Implementation		🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦	🟦					
Data Transmission					🟩	🟩	🟩									
Data Querying and Processing							🟦	🟦	🟦							
Data Analytics and Visualization								🟩	🟩	🟩	🟩	🟩				
Automated Monitoring and Alerts													🟦	🟦	🟦	
User Acceptance Testing													🟩	🟩	🟩	🟩
System Deployment																🟦
Post-Deployment Optimization																🟩

In the provided Gantt chart, this is the timeline the expected milestones are intended to be completed. As an outlier from the rest of the milestones, the collection and probing section is the longest as that is the most important piece in the Minimal Viable Product (MVP). Further in, some milestones are expected to start midway within another milestone such as Data Transmission, querying and processing, and analytics. These can be worked without one another as dummy data can be used but in the end, the deployment must have these accomplished. Although, the automated monitoring and alerts and the user acceptance test must start together as the previous parts of the milestones need to be accomplished and the data pipeline and analysis is working properly. Without these milestones, the rest of the development is not capable of starting. Leading to the final part, This would be where the MVP would be ready for deployment, including the Post-Deployment Optimization where the stretch goals and refining of the program would start.

Conclusion

In conclusion, Western Digital lacks an automated process for stress testing and analyzing several of their data storage devices leading to a waste in manpower and money for the company. To solve this problem, we will fabricate an automated system capable of fulfilling the automated needs for this system. This will be done through the implementation of a system that can take data from WD data centers, probe/stress test it with eBPF, grab and analyze eBPF output logs with Python, upload analyzed data to S3 buckets to be sanitized with Glue and finally display sanitized data on a visual dashboard. This system will be automated to run at any desired time and for as many devices as wanted.

The solution had already been laid out by the client which made it easier to understand the requirements of the project. However, to fully understand what is required, we had to create the documentation for it. Creating a document that specializes on describing these requirements helps progress the project by providing a guideline that details what is expected for the system to include. Therefore, if any of the members ever has some question about what is being expected as part of the development, they can refer to this document.

Several key insights can be derived from the analysis and planning of the project through this document. Our system will allow adaptability to different hardware configurations and operating systems, providing Western Digital with a comprehensive solution for their diverse range of devices. By automating the entire process, from the data collection in WD data centers to the visualization of sanitized data, we are creating a streamline, efficient and reliable solution.

As we move forward, we are confident that this project will meet the MVP (Minimal Viable product) requirements and further exceed expectations set forth by our client. The structured approach outlined in this document, combined with our commitment and determination, ensures a positive trajectory to its development. We recognize that changes and challenges are inherent to any project. As we anticipate these shifts and potential hurdles, we remain agile and prepared to adapt, ensuring a proactive environment to address any arising issues and concerns. Through our collaborative efforts, we are committed to delivering a cutting-edge solution that will revolutionize Western Digital's testing process, setting new standards for efficiency and innovation in the industry.

Glossaries/Appendices

Definitions:

1. **eBPF (Extended Berkeley Packet Filter):**
   - A highly flexible and efficient in-kernel virtual machine that allows the execution of bytecode within the Linux kernel. Used for various purposes, including network packet filtering and performance monitoring.

2. **AWS (Amazon Web Services):**
   - A comprehensive cloud computing platform provided by Amazon. AWS offers a wide range of services, including computing power, storage, databases, machine learning, and analytics.

3. **AWS S3 (Amazon Simple Storage Service):**
   - A scalable object storage service provided by AWS, designed to store and retrieve any amount of data from anywhere on the web.

4. **AWS Glue:**
   - A fully managed extract, transform, and load (ETL) service that makes it easy to prepare and load data for analysis. It provides capabilities for data cleaning, transformation, and organization.

5. **AWS Athena:**
   - An interactive query service that makes it easy to analyze data in Amazon S3 using standard SQL. It allows users to analyze large datasets stored in S3 without the need for complex ETL processes.

6. **AWS QuickSight:**
   - A business analytics service provided by AWS for creating visualizations, performing ad-hoc analysis, and quickly getting insights from data.

7. **AWS OpenSearch:**
    - A search and analytics engine derived from Elasticsearch that allows organizations to perform various data analysis tasks, including log analysis, full-text search, and real-time application monitoring.

8. **Gantt Chart:**
    - A visual representation of a project schedule that shows the start and finish dates of elements of a project. It includes tasks, milestones, dependencies, and the overall timeline.
9. **Sanitization:**
    - The process of cleaning up information. In this case, the cleaning of data that would be deemed useless or unnecessary in the program's objectives.
10. **Probing:**
    - The action of gathering objects/information. In this case, The collection of kernel system information.
11. **Probes:**
    - Probes are objects that attach themselves to the Kernel level of the system for various reasons such as debugging, tracing, fault injections and such. When a probe attaches itself to the kernel, it would begin probing (Collecting) information based on the reason the probe is attached..
12. **Querying:**
    - A request given to the system that would manipulate or pull the information for the user.

Resources:

- **eBPF Resources:**
    - eBPF Documentation: Official documentation providing an overview of eBPF and its capabilities.
    - eBPF Tracing Tools: A collection of eBPF tracing tools curated by Brendan Gregg.
    - BPF Performance Tools: Resources on performance tools using BPF, recommended by Brendan Gregg.
- **AWS Resources:**
    - AWS Documentation: Official documentation for various AWS services, providing in-depth information on usage and configuration.