Milestone 1 (Planning): Aquatic Weeds

(Tracking water Hyacinth at Hartbeespoort Dam)

Nosipho Precious Donkrag

577354

13 July 2024

PMM381

(Version 2)

INDEX:

A. Executive Summary

Project Title: Tracking water Hyacinth at Hartbeespoort Dam

The 'Tracking water Hyacinth' project forms a subset of a larger programme called the Aquatic Weeds programme ran by the Centre of Biological Control (CBC) at Rhode's University. The project is created to support the university's endeavours in controlling and managing the growth and spread of the invasive Hyacinth species.

This project aims to predict the location and distribution of the Hyacinth mat on the Hartbeespoort Dam. As part of the requirements analysis it was gathered that the research team requires access to data for analysis purposes; and to monitor the effectiveness of their control strategies. Thus, part of the project includes a cohesive database as a deliverable.

This report forms part of the project's timeline; it is the deliverable for the planning phase. This report covers scope, stakeholder and requirements analysis, scheduling, high-level design, project management methodology, risk and resource management. The conclusion of this section allowed for the commencement of the design phase.

This report takes a practical approach over a theoretical one, and provides a step by step approach to managing the project. It utilises the following tools to plan and manage the project:

- Requirement traceability matrix;
- Work break-down structure diagram;
- Gantt chart and critical path analysis;
- Scrum implementation (with sprints);
- RACI matrix;
- Risk assessment matrix;
- Responsibility matrix; and
- Version control.

This structured outcome will ensure that the project remains on track, and its objectives are met.

B. Scope

The scope of the Water Hyacinth project will define the goals, deliverables and boundaries of the project. Setting clear goals and deliverables will reduce the chances of scope creep and ensure the project remains focused and on track, thus increasing the likelihood of the success of this project.

1.1. Goals

1.1.1. Research

- The initial goal of the Water Hyacinth project is to understand the ecological and environmental impact of the Hyacinth mat on the Hartbeespoort Dam located east of Pretoria; and to understand existing management strategies through extensive research.

- Identify the significance of successfully tracking and predicting the location and distribution of the Hyacinth mat and identify the impact of the success of this project on the shareholders (fisherman, community) of the project.

- Investigate projects of a similar nature that utilise machine learning algorithms and satellite imagining tracking hyacinth species across bodies of water, with the aim of drawing comparisons with the current project in order to gather valuable insights on how to approach this project.

- Identify variables that impact the movement, distribution and growth of the Hyacinth mat with the aim of identifying possible features to use to train the model(s).

1.1.2. Data collection

- Interface with Google Earth engine using an existing API(Application programming interface).
- Collect images using the API of the Hartbeespoort Dam over a period of one year.
- Collect these images from Landsat or Sentinel Satellites.
- Additionally collect weather data of the dam over a year.

1.1.3. Technological goals

- Decide on initial features required to train the model or models, such as mat size, distribution, weather information and location.
- Investigate and implement the required image pre-processing steps to highlight these features.
- Preform data transformation to prepare the data for the model.
- Decide and develop a machine learning model or models to train using the data, and train the model(s).
- Produce a final user centric application that will allow users to view and track the mat in real time (front-end application).
-

1.2. Deliverables

- **A project execution plan** highlighting the scope, objectives, requirements, timeline, and resource and risk management. The project plan will also detail the agile methodology that will be utilized to implement this project.
- A document fully describing the system design and analysis.
- A fully functioning user-centric prototype that predicts and tracks the movement of the Hyacinth mat.
- An end-to-end software document that explains how the system was implemented.
- A 15 minute power point presentation of the project.

1.3. Boundaries

- **Predict only:** The project will focus only on tracking and predicting the location and distribution of the Hyacinth mat on the Hartbeespoort Dam; it will not provide management solutions or eradication of the mat.
- **Data sources:** Only images collected through Google Earth Engine will be used; and weather data collected from the South African weather service will be used. No additional data sources will be integrated.
- **Geographical location:** the project will be restricted to the Hartbeespoort Dam located east of Pretoria South Africa.
- **Timeline:** The project will conclude on the 1st of November 2024.

C. Objectives

The objectives of a project form part of the scope, the objectives for the current project are listed below:

1. Research

Conduct extensive research to understand the environmental and ecological impact of the Hyacinth mat on the Hartbeespoort Dam.

Investigate similar projects that utilise machine learning and satellite imaging to track invasive species and draw comparisons with the current project to gather insights.

Identify factors affecting the movement, distribution and growth of the Hyacinth mat as these will be the parameters used to train the model.

2. Data collection

Interface with Google Earth to collect weather data and images of the dam over a period of one year. Process the collected data and store it appropriately.

3. Technological Objective

- Create a database suitable for storing the collected data and analysis by the research team.
- Identify initial features required to train the ml (machine learning) model.

- Develop and train the model using these selected features.
- Create a user centric web application that shareholders can utilise to track the Hyacinth mat.

4. User Adoption:

- Engage with the research team to ensure that the system meets their requirements.
- Gather user feedback to improve the system as required.

5. Presentation and documentation:

- Develop an extensive project execution plan detailing the scope, objective, timeline, risk and resource management as well as the chosen project management methodology.
- Create a document outlining the project's design (both high-level and lower-level (milestone 2) designs).
- Prepare a functioning prototype to demonstrate to the project owner on the 1st of November.
- Prepare a 15minute power point presentation summarising the Hyacinth project.

D. Requirements Analysis

1. Stakeholder analysis

The initial objective in stakeholder analysis is to identify the stakeholders of the Hyacinth project. The stakeholders will be the people or entities affected by the success or failure of the project.

Question:

Who will be the primary users of this system?

Answer:

- The Rhodes university Research team (Centre for Biological Control (CBC))
  The research team aims to find an encompassing approach to managing the growth and spread of the invasive plant. Through their bio-control program the team hopes to ensure ecosystem recovery and sustainability. The availability of data for their study will assist in managing and controlling the spread of the invasive species.

- The community
  The community is made up of the residents around the dam, boaters and fishermen and the on sight local community that works with the university; these are the local members that help combat the growth of the mat by releasing insects that feed on the mat.

Analysis:

The impact of the success of the project on the stakeholders of the project is discussed below:

CBC:

The success of this project will support the availability of the invasive plant population and weather data, that the university can use for analysis and to monitor the effectiveness of their recovery strategies.

Community:

The success of this project will ensure that the community members can view the current and predicted location of the mat. Being able to immediately view the location of the mat will help the community that works with the university; identify the attack point for their control measures thus saving them time.

As of 2020 there were 6 different insects that were released to combat the growth of the Hyacinth plant (University, 2020). One of these insects called the ‘Megamelus Scutellaris’ swarms the community in summer searching for Hyacinth plants to feed on. This negatively affects the residents around the dam. The system will allow

residents to view the current and predicted location of the mat; thus they may be able to plan ahead and prepare for any influx of this species.

Boaters and fishermen will be able to plan their routes to avoid dense mats they may damage their boats and get stuck in their gears. Fishermen will be able to identify locations where the mat is not present; as this will make fishing easier.

Concluding stakeholder analysis:
From the stakeholder analysis, it can be concluded that most of these stakeholders will interact with the system in the similar manner. However, the research team will require access to the data for analysis, thus a database will be required.

From the stakeholder analysis, future considerations can be placed on utilising the system to promote sustainable fishing practices. That is, the app can be further expanded to allow fishermen to view areas where fishing is allowed and areas where it is not, these would be areas that are still under recovery (natural habitat recovery) thus promoting sustainable fishing practices.

2. User stories

Research Team (CBC)

"As a researcher I would like to access real time data of the location and distribution of the Hyacinth mat so I can monitor the effectiveness of our bio-control strategies"

"As an analyst I want to be able to download historic datasets of the mat so I can perform analysis and evaluate the long-term impact of our intervention strategies"

Community member

"As a community member I want to view the current and predicted location of the plant so I can plan for the insect influx that follows during summer"

"As a community worker I want to see the location where the mat is most dense so I can effectively release my insects to control the plant"

Boaters

"As a boater I would like to see the location where the mat is most dense so I can plan my route, avoid damaging to my boat"

Fishermen

"As a fisherman I want to view the current location of the mat so I can avoid it, this will allow me to fish more efficiently without the plant hindering my experience."

3. Functional requirements

Functional requirement (FR) 1: View the Hyacinth distribution on the map

Question:
What are the primary functions that the system needs to perform?
Answer:
The system must allow users to be able to view the current and predicted location of the water Hyacinth mat on the Hartbeespoort Dam. It must also show users the wind and weather information as this affects the growth of the mat.

Requirements gathering for the log

View images of the dam:

FR1.1: The system should allow users to view the current distribution and the predicted distribution of the Hyacinth mat on the dam through a web application.

FR1.2: The system will provide a map interface that will display the current and predicted location of the mat.

View weather data:

FR1.3: The system will allow users to view the current wind and weather information through the web application.

FR1.4: The system will allow users to view the predicted wind and weather information through the same we application.

FR1.5: User Interface

The user interface, which will be the front end of the web application, will consist of 2 parts.

- The first part (the first page): Display the current weather data and the location of the Hyacinth mat on an integrated map.
- The second page: Display the prediction of the mat's location on the dam.

4. Non-Functional requirements

Non-functional requirements are specifications and criteria's that define how the system will behave. They also assist in highlighting the constraints and limitations of the system. This section will analyse the non-functional requirements of the Hyacinth project.

4.1. NFR1 (non-functional requirement 1): Performance requirement

Question:
What is the expected response time when the system processes satellite images and weather data?

Answer:

NFR1.1: The end-to-end interaction with the system; that is, viewing the current information and making a prediction must take less than or 60 seconds.

$$system\ response\ time\ (srt) \leq 60\ seconds$$

Question:
What performance benchmarks or metrics must the system meet?

Answer:

NFR1.2: The system must predict with an accuracy of at least 60%. That is, the employed machine learning model must have an accuracy score of 60% or above when predicting.

$$prediction\ accuracy(PA) \geq 60\%$$

4.2. NFR2: System behaviour

NFR2.1: The system shall receive input data that is weather and image data of that specific day and make a prediction for the following day (24 hours in advance predictions).

NFR2.2: The input data fed into the system shall come from a Google earthengine's API (application programming interface

NFR2.3: The system shall update its database on a monthly basis with the latest data for that specific month and the model will be retrained and tested on the updated data (monthly training and testing).

4.3. NFR3: Reliability and availability

NFR3.1: The system's database and its code must be backed up on a monthly basis. Backups will be stored on Github.

NFR3.2: Once live, the system will be available 99% of the time, this is excluding the time it will take to backup and update the system.

5. Assumptions and constraints

'A mistaken assumption can derail a project strategy' Alan Zucker. Project assumptions are required to understand the project's key factors and documenting them provides a starting point for future work. A project constraint is a limiting factor

for a project, constraints affect the quality of the project hence they need to be managed and documented.

5.1. Assumptions:

Availability of accurate data	The required satellite images and weather data from Google Earth is accurate, the weather data is up-to-date and available. The system will mainly use the weather data to train the model and make a prediction. Hence, the system's accuracy is dependent on the accuracy and completeness of the weather data.
Access to internet	The community and team members have reliable internet access to use the web application. An internet connection is a requirement for connecting to the web application.
Technical proficiency	Training will not be required to use the application as users have basic level of training.

5.2.Constraints:

Data Sources	The input data will come exclusively from Google Earth, no other external data sources will be used. The system will not be flexible enough to incorporate other data sources in the future should they become available.
Monthly backups and updates	The system will only be backed-up and updated on a monthly basis, in 30 days form the last recorded backup and update.
System availability	Once the application is live, it will be available 99% of the time, excluding the scheduled update and backup time. The chosen database hosting solution must support this.
Limited Technical support	The system must be designed to not require extensive technical support. At the end of the year the team working on it will not be available.

6. Requirement Traceability Matrix:

ID:	Requirement	Type	Priority	Source	Comment
FR1.1	View current and predicted location of the mat	Functional	**critical**	Stakeholder analysis, CBC	
FR1.2	Map interface	Functional	**medium**	Stakeholder analysis, CBC	Requirement not critical to the success of the project as the predicted location of the mat can always be written down if the map interface fails.
FR1.3	View current weather data	Functional	**Low**	Stakeholder analysis, CBC	
FR1.4	View predicted weather data	Functional	**High**	Stakeholder analysis, CBC	
NFR1.1	System response time less than or equal to 60 seconds.	Non-Functional	**Medium**	Performance requirement	The initial priority at this stage is to have the system respond.
NFR1.2	Prediction accuracy must be more than or equal to 60%.	Non-Functional	**Critical**	Performance requirement	The requirement is critical to the success of the project. If the prediction score is low, then users will not find value in the application.
NFR2.1	The input data that is weather and image data of that specific day.	Non-Functional	**Medium/high**	System Behaviour	Perhaps the model will perform better if it uses multiple days' data to make a prediction?
NFR2.2	Input data comes from Google earthengine's API	Non-Functional	**Critical**	System Behaviour	Standardized data sources
NFR2.3	Monthly database updates (differential updates)	Non-Functional	**Medium**	System Behaviour	
NFR3.1	Monthly backup of the entire system	Non-Functional	**Medium**	Reliability and availability	
NFR3.2	System availability (99%)	Non-Functional	**Critical**	Reliability and availability	

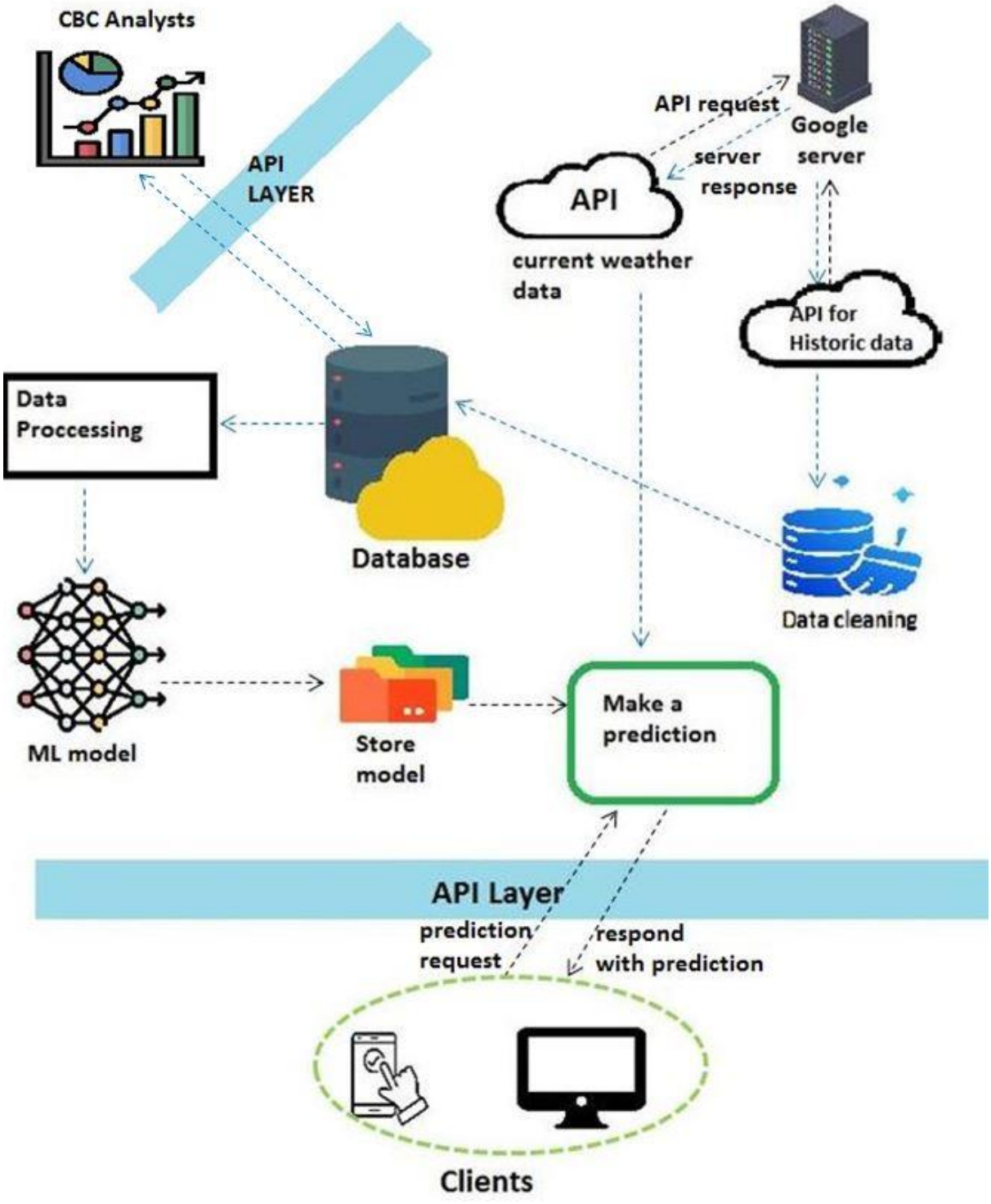
7. Validation and Acceptance criteria

Project acceptance criteria are criteria that must be met before the project deliverables are accepted. The validation criteria are benchmarks used to evaluate if the project meets the goals and objectives set.

Requirement	Validation criteria	Acceptance criteria
View current and predicted location of the Hyacinth mat	The system generates a real-time map of the current Hyacinth mat distribution. The system generates and displays the predictions of the mat distribution for the next 24 hours.	Users can view the current and predicted location of the distribution of the Hyacinth mat through the web application. The map interface is user-friendly and in the case of failures users can still access the predictions in text format.
View the current and predicted weather data	The web application will display the current weather data. The system will predict the weather data for the following day.	The web application will display the weather data (wind and temperature) in real-time, and make a prediction for the following day.
The prediction accuracy must be more than or equal to 60%	Test the prediction output against historical data. Regular testing of the model must result in predictions of 60% and above.	The model utilised consistently achieves accuracy of 60% and above when predicting.
Monthly database updates and backups	Ensure the system is updated and backed-up on a monthly basis. Check that the updates are differential in nature.	Monthly backups complete without any errors and the backed up data is retrievable for training. Monthly updates are successfully completed and the database has the most recent data available in it.
System availability (99%)	Monitor and record how long it will take to update and backup the entire system. This will help identify the system's uptime.	The system maintains an uptime of 99% excluding the period required for backups and updates.
Documentation	Produce a comprehensive software document for the source code (red the doc format?), the document should be available on Github.	The document is fully completed and available on Github. The document describes the full software implementation.

E. High-level design

A high-level design is an overall design of a system; it is not as intricate as the lower-level design but serves as an indication of how the overall components will interact with each other. The high level design for the Hyacinth project is given below and an overview of the design will follow.



High-Level Design Overview

Externa APIs:

- The external APIs will request data from the google servers; this data will be the weather data and images of the dam through a satellite interface. The first external API will request historical data over a period of one year.
- The second API will request current weather information for the specific day.

Data cleaning:

The historical data collected over a period of one year; will undergo data cleaning before it is loaded into the database, to ensure data consistency.

Data processing:

Once the data is loaded into the database it will then get processed. The processing steps could involve normalization and preparation for the ml model. Any required processing before training will occur here.

ML model and storage:

Once the data has been processed it will be sent to the chosen algorithm for training and testing. Once the model has been trained and the desired matrices have been met, the model will be stored in a '.pk' file.

The Main API Layer:

- The main API layer is there to facilitate communication between the clients, through the web application, and the backend application.
- The clients will make a request for a prediction, this request will be sent to the application that solely makes a prediction;
- The application will use the second external API to fetch the current weather data to make a prediction on.
- Then load the stored ml model and feed it the current weather data, so the ml model can make a prediction.
- The prediction is then sent back to the front-end (clients).

The second API layer:

This layer is created for the research team's analysts; this layer will allow for CRUD events between the analysts and the database; thus allowing for direct access to the database.

F. Timeline and Schedule plan

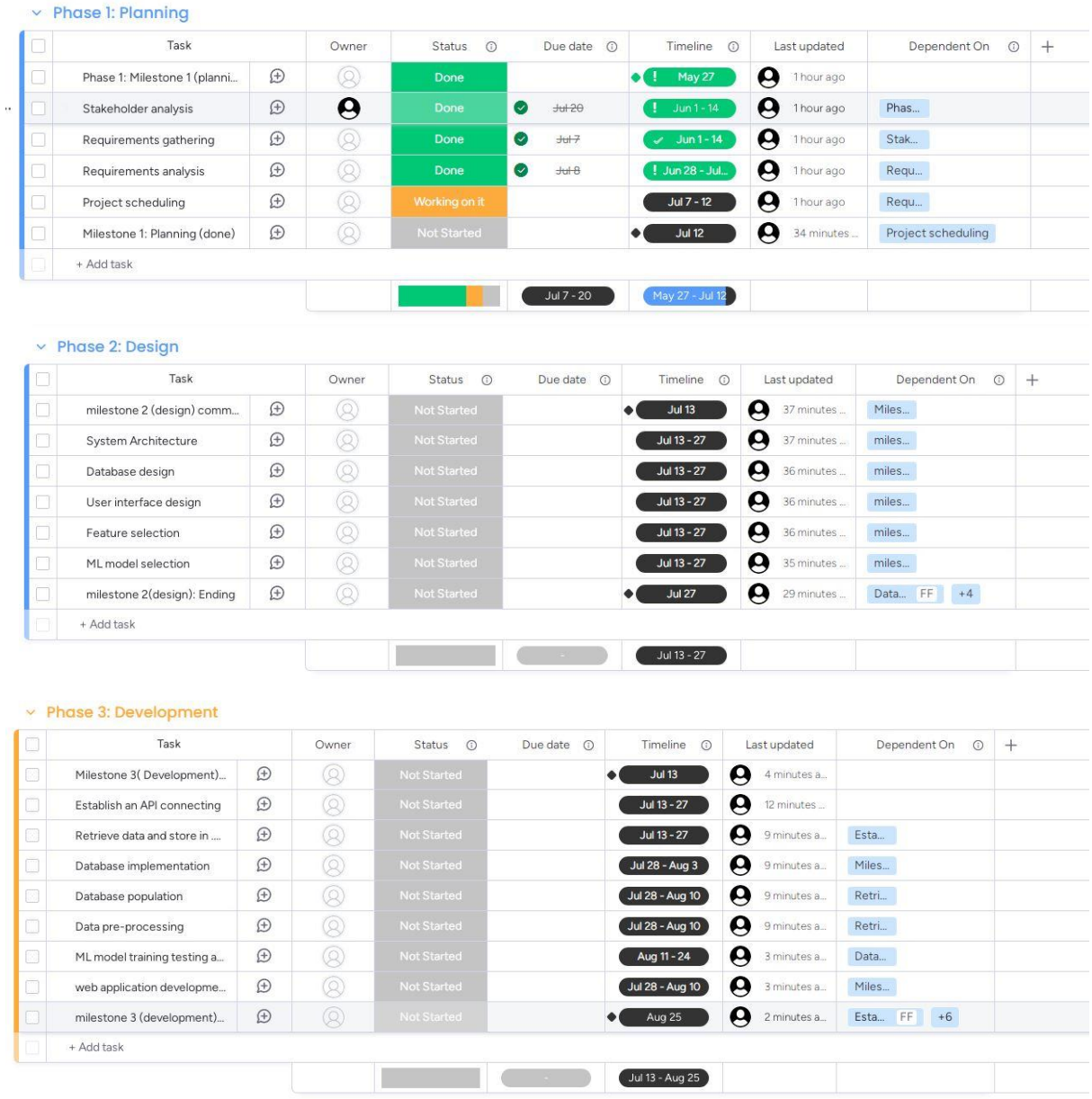
Deadlines:

Milestone1: Planning	21 July 2024
Milestone 2: Design	6 September 2024
Milestone 3: Implementation, testing and demonstration	1November 2024
Milestone 4: Expo	18 November 2024

The project must be completed in 14.71 weeks. The project will be broken down into the following tasks:

Phase 1: Planning	**Phase 2: Design**	**Phase 3: Development**
Stakeholder analysis	System Architecture	Google earth engine API integration (**store weather and image data in a csv file** while the database is being implemented so that the project is not halted – data pre-processing can continue)
Requirements gathering	Database design	Database Implementation and population
Requirements analysis	User interface design	Data preparation and pre-processing
Project Scheduling	Features to train model on selection	Ml model training, testing and storing.
Phase 4: User Acceptance Testing (iterative)	ML model selection	Web application development
Meet with supervisor to give a pre-demo before the final demo.	**Phase 5: Documentation**	Continuous tuning and testing of the ML model.
Gather feedback from supervisor	Create full software documentations and save it on Github.	
Update project as per feedback	Prepare a 15min power point presentation.	

The tasks required for phase 1 to 3 have been tabulated in Monday.com, it was not possible to place phase 4 and 5 into tables because they require input from the supervisor. Hence at the end of phase 3 a meeting will be request with the supervisor and a timeline will be set for the two remaining phases.



Phase 1: Planning

Task	Owner	Status	Due date	Timeline	Last updated	Dependent On
Phase 1: Milestone 1 (planni...		Done		May 27	1 hour ago	
Stakeholder analysis		Done	~~Jul 20~~	Jun 1 - 14	1 hour ago	Phas...
Requirements gathering		Done	~~Jul 7~~	Jun 1 - 14	1 hour ago	Stak...
Requirements analysis		Done	~~Jul 8~~	Jun 28 - Jul...	1 hour ago	Requ...
Project scheduling		Working on it		Jul 7 - 12	1 hour ago	Requ...
Milestone 1: Planning (done)		Not Started		Jul 12	34 minutes ...	Project scheduling
+ Add task						
			Jul 7 - 20	May 27 - Jul 12		

Phase 2: Design

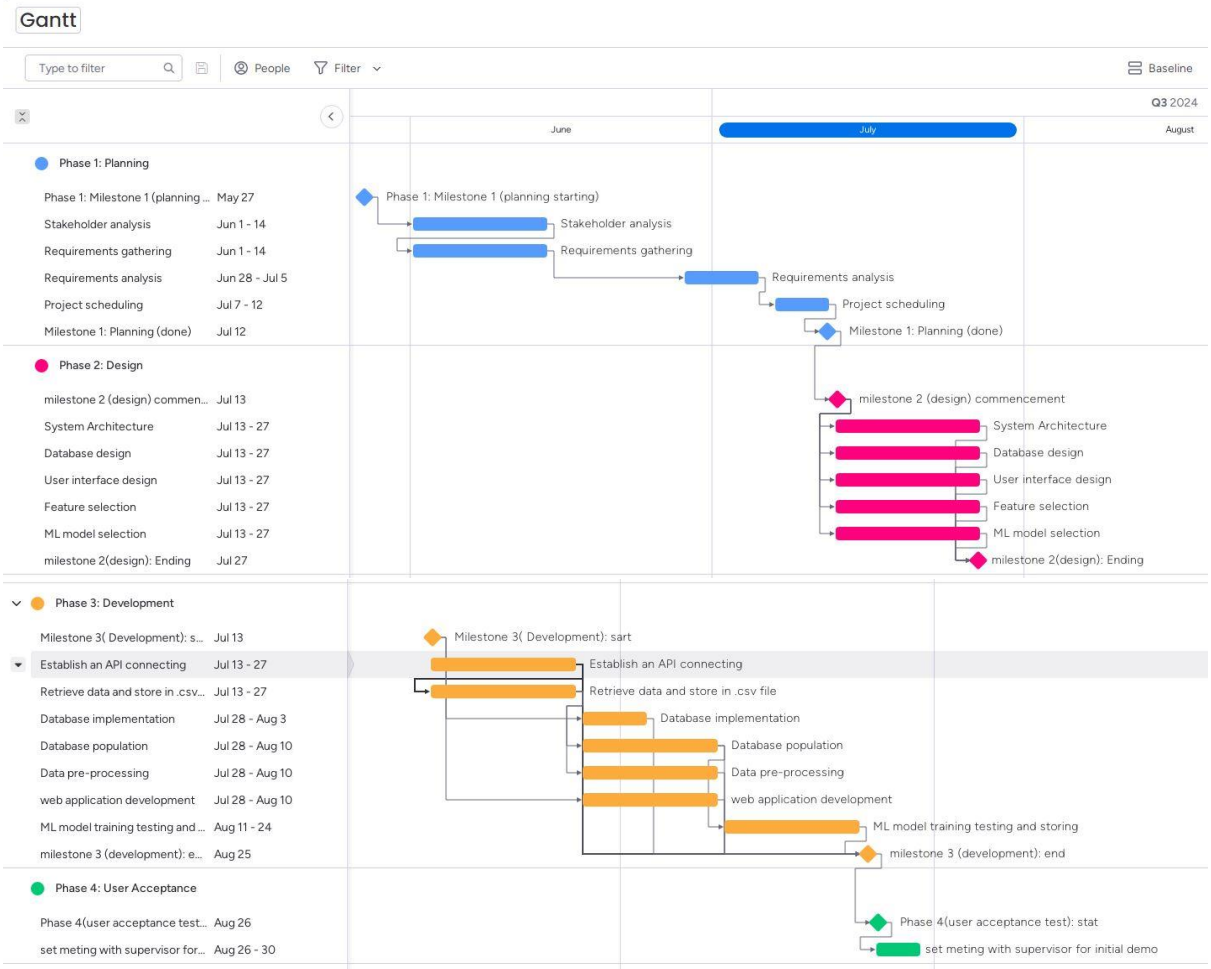
Task	Owner	Status	Due date	Timeline	Last updated	Dependent On
milestone 2 (design) comm...		Not Started		Jul 13	37 minutes ...	Miles...
System Architecture		Not Started		Jul 13 - 27	37 minutes ...	miles...
Database design		Not Started		Jul 13 - 27	36 minutes ...	miles...
User interface design		Not Started		Jul 13 - 27	36 minutes ...	miles...
Feature selection		Not Started		Jul 13 - 27	36 minutes ...	miles...
ML model selection		Not Started		Jul 13 - 27	35 minutes ...	miles...
milestone 2(design): Ending		Not Started		Jul 27	29 minutes ...	Data... FF +4
+ Add task						
			-	Jul 13 - 27		

Phase 3: Development

Task	Owner	Status	Due date	Timeline	Last updated	Dependent On
Milestone 3(Development)...		Not Started		Jul 13	4 minutes a...	
Establish an API connecting		Not Started		Jul 13 - 27	12 minutes ...	
Retrieve data and store in		Not Started		Jul 13 - 27	9 minutes a...	Esta...
Database implementation		Not Started		Jul 28 - Aug 3	9 minutes a...	Miles...
Database population		Not Started		Jul 28 - Aug 10	9 minutes a...	Retri...
Data pre-processing		Not Started		Jul 28 - Aug 10	9 minutes a...	Retri...
ML model training testing a...		Not Started		Aug 11 - 24	3 minutes a...	Data...
web application developme...		Not Started		Jul 28 - Aug 10	3 minutes a...	Miles...
milestone 3 (development)...		Not Started		Aug 25	2 minutes a...	Esta... FF +6
+ Add task						
			-	Jul 13 - Aug 25		

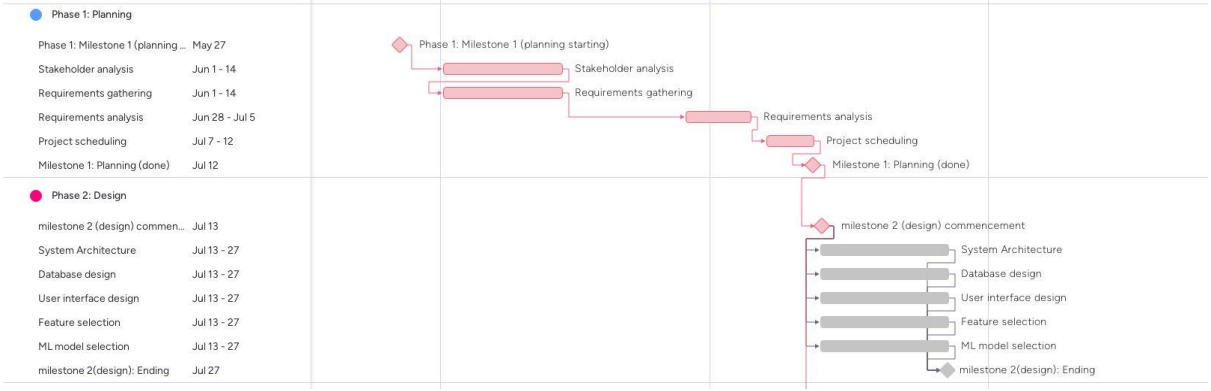
Gantt chart:

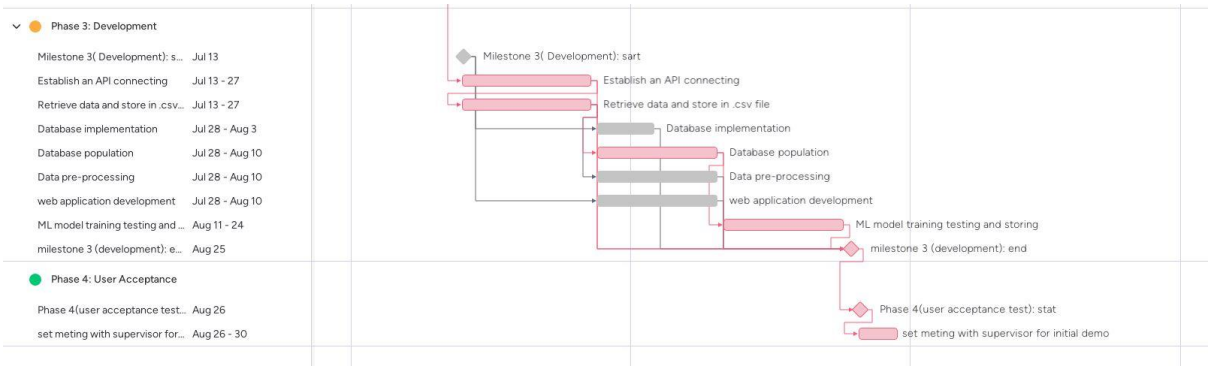
A Gantt chart is a project management tool that will assist in planning and scheduling of tasks that are required to be done in order for the project to complete successfully. Below is a Gantt chart of the already tabulated tasks:



Critical path:

The critical path is the longest sequence of tasks that must be completed for the project to succeed. The tasks (highlighted in red) are critical, which means should they be delayed the whole project will be delayed.

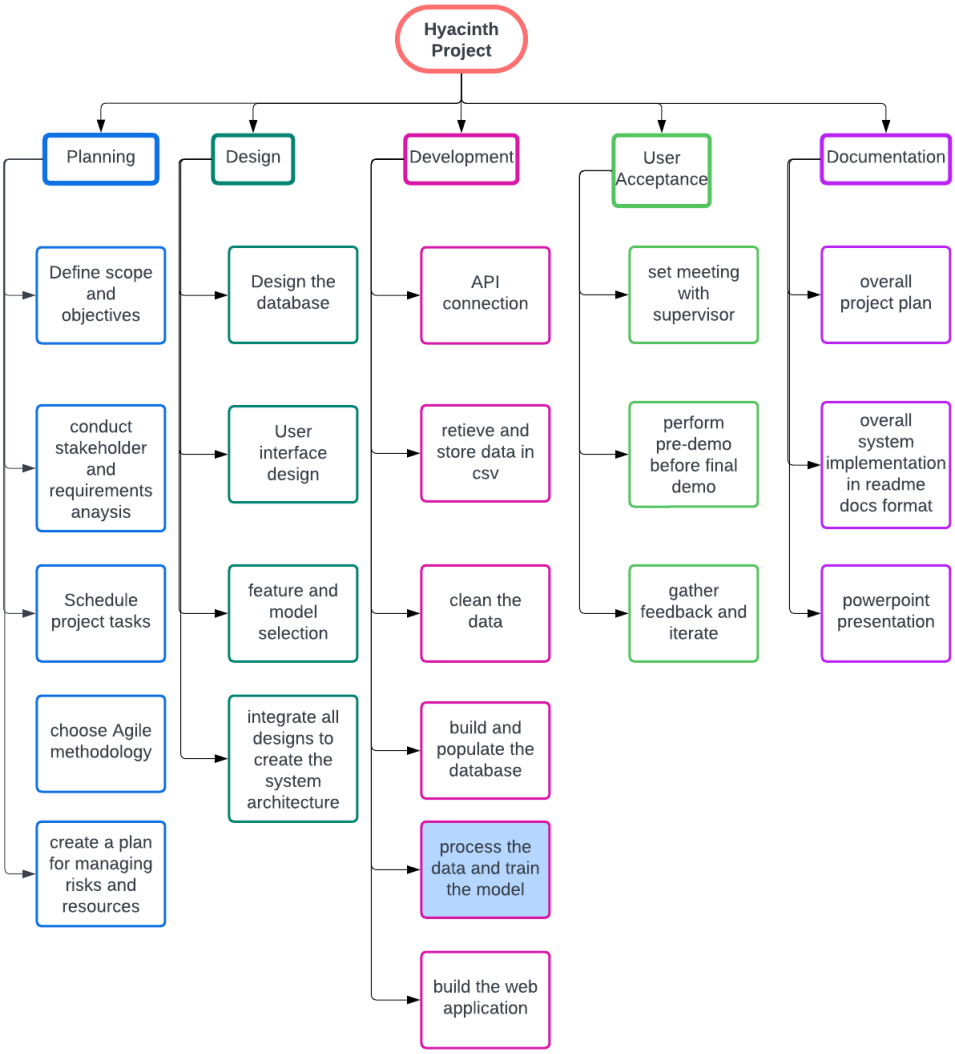




Having a thoroughly documented plan is critical to the success of the project; other critical tasks are 'establishing an API connection' and retrieving and storing data. Without the API connection data cannot be retrieved, the ml model cannot be trained and the whole project fails; hence it is utmost important to start with establishing and API connection as early as possible that is when phase 2 commences.

Work Breakdown structure

The overall work breakdown structure for the Hyacinth project is below:



G. Agile Methodology (Scrum)

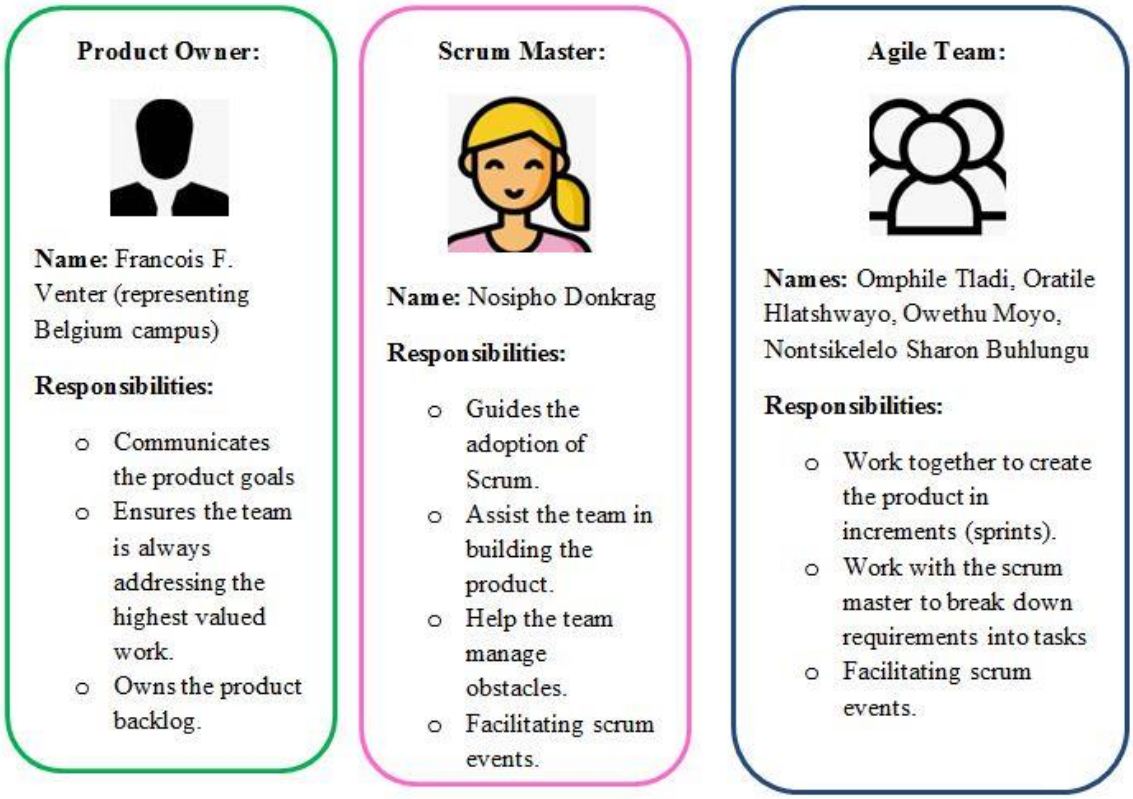
1. Overview

Scrum is an agile project management framework; agile refers to a set of principles and values for managing a project and these principals are based on the agile manifesto (Ram Srinivasan, 2024). The choice to use the Scum methodology is due to its ability to adapt to changing user requirements and conditions.

This section will not give an in depth theoretical analysis of the methodology but rather a practical application of it. Through the application of the methodology, brief descriptions of the components of Scum will be given.

2. Scrum accountabilities (roles)

The scrum methodology consists of three main roles to ensure work is managed and shared effectively:

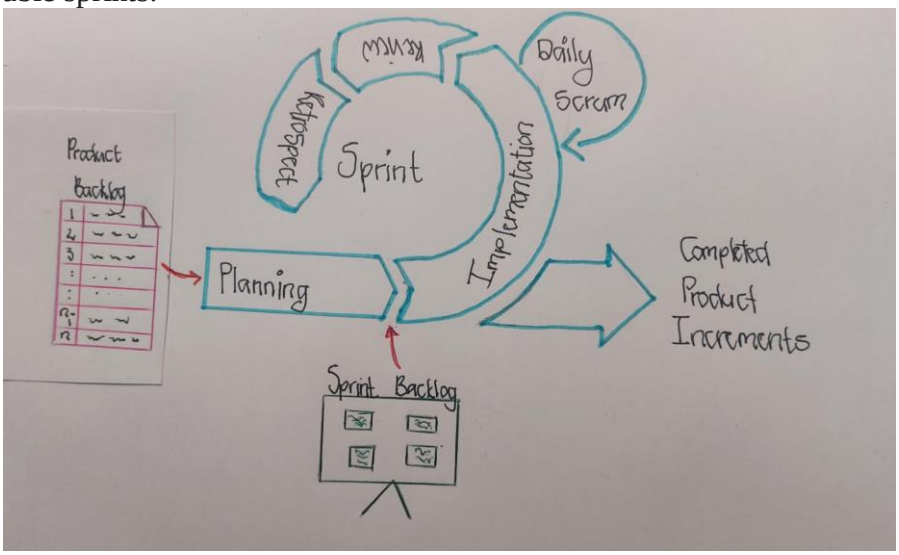


- The initial meeting with the product owner was held on the 27th of May 2024; in this kick off meeting the project goals and expectations were discussed. The product owner provided a quick overview of the project's importance and its stakeholders. This provided sufficient information to commence the researching phase of the project.

- The following meeting that was held with the product owner was to gather the requirements. Conducting prior research before gathering the requirements with the product owner; resulted in an effective and efficient meeting for gathering the requirements.

3. Product Backlog

The product backlog of the entire project is the input into the sprint planning. The tasks to be completed on the product backlog are broken down into smaller achievable sprints:



The Product backlog for the Hyacinth project:

The product backlog for the project is first shown under the schedule plan that was created using Monday.com. The backlog is broken down into phases and these phases are then broken down into tasks, from these tasks the sprints were then planned.

Phase 1: Planning (milestone 1)

Task	Owner	Status	Due	Timeline	Dependent On
Stakeholder Analysis	Nosipho, Sharon	**Done**	7 July	1 June – 7 July	
Requirements gathering	Nosipho, Sharon	**Done**	7 July	1 June – 7 July	
Requirements Analysis	Nosipho, Sharon	**Done**	7 July	1 June – 7 July	Stakeholder Analysis
Scheduling	Nosipho	**Done**	13 July	7 June – 13 July	
Methodology	Owethu	**Working on it**	13 July	1 June – 13 July	Scheduling
Resource Management	Thandeka	**Done**	13 July	1 June – 13 July	Requirements Analysis
Risk Management	Omphile	**Done**	13 July	1 June – 13 July	Requirements Analysis

Phase 2: Design (milestone 2)

Task	Owner	Status	Due	Timeline	Dependent On
System Architecture	Everyone	**Working on it**	27 July	13 – 27 July	Milestone1 ending
Database Design	Owethu	**Working on it**	27 July	13 – 27 July	Milestone1 ending
User Interface	Sharon	Not started	27 July	13 – 27 July	Milestone1 ending
Feature selection	Nosipho, Omphile	Not started	27 July	13 – 27 July	Milestone1 ending
ML model selection	Nosipho, Omphile	Not started	27 July	13 – 27 July	Milestone1 ending

Phase 3: Development

Task	Owner	Status	Due	Timeline	Dependent On
Establish an API connection	Thandeka	**Working on it**	27 July	13 July – 27 July	Milestone 1 ending
Retrieve and store data in a csv file	Thandeka	**Working on it**	27 July	13 July – 27 July	Establishing and API connection
Database implementation	Owethu	**Not started**	3 August	28 July – 3 August	Milestone 2 ending
Database population	Owethu	**Not started**	3 August	28 July – 3 August	Retrieving data and storing it
Data pre-processing	Omphile, Nosipho	**Not started**	10 August	28 July – 10 August	Retrieving data and storing it
ML model training and tuning	Nosipho, Omphile	**Not started**	24 August	11 August – 24 August	Data pre-processing
Web application development	Sharon	**Not started**	24 August	11 August – 24 August	Milestone 2 completion
Integrate database with ml model then with web app	Owethu, Nosipho	**Not started**	1 September	25 August – 1 September	Web app, ml model, database population

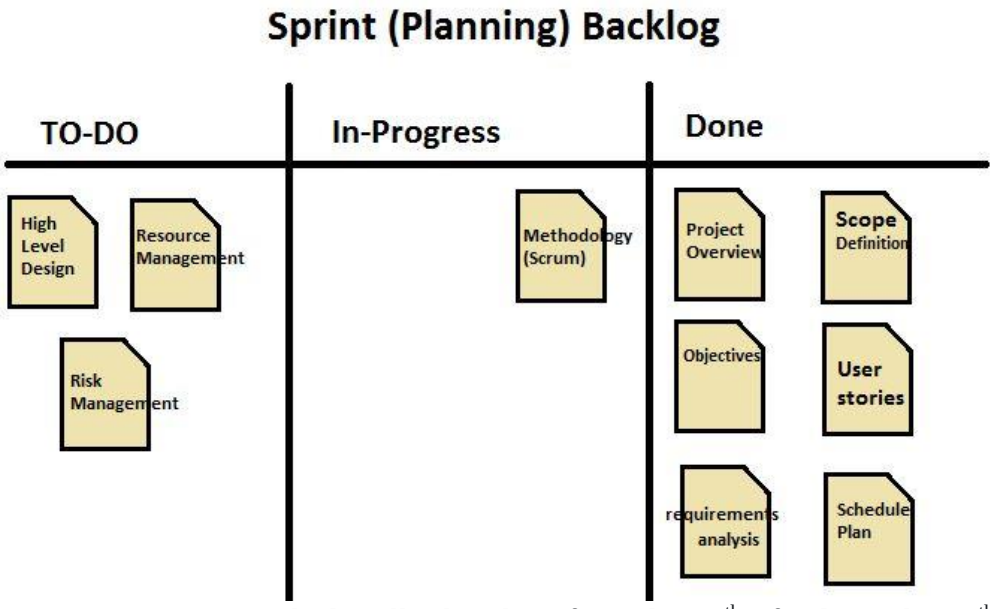
Phase 4: User Acceptance testing					
Task	**Owner**	**Status**	**Due**	**Timeline**	**Dependent On**
Set meeting with supervisor	Nosipho	**Not started**	26 August	26 August – 1 September	Phase 3 completion
Demo the prototype	Everyone	**Not started**	1 September	26 August – 1 September	Phase 3 completion

4. Scrum Events

This section will discuss the sprints set out for the project. A sprint is a time-boxed event (1- 4 weeks) where the team focuses on completing the sprint's goal. The sprint goal is made up of tasks that need to be completed for the project; these are broken down manageable tasks. The sprint lifecycle consists of sprint planning, sprint review, sprint retrospect and daily scrum. Each of these components will be discussed in the section.

Sprint Planning

The current sprint for milestone 1 (planning) is shown below and the deadline for this sprint is the 12th of July.



The upcoming sprint, which will take place from the 13th of July to the 27th of July, is the design and connection sprint. The tasks to be completed and the member responsible for each task are shown below:

Design and connection sprint	
Tasks:	**Responsibility:**
System Architecture design	Everyone
Database design	Owethu
User interface design	Oratile, Sharon
Feature model selection	Nosipho, Omphile
ML model selection	Nosipho, Omphile
Establish an API connection	Thandeka
Timeline: 13th July to 27th July 2024	

The following sprint will be the development sprint which will include testing; this sprint is projected to be longest sprint spanning almost a month.

Daily scrum

One of the components of the scrum methodology is the daily scrum meetings that last a period of about 15 minutes for each member. The team has decided to have daily check-ins on WhatsApp where we will be updating each other on our respective work; however these daily meetings will only commence at the start of the second sprint this is due to the fact that milestone 1 was mostly research, no technical work was required.

Layout of daily scrum meeting:

Sprint 2: Design and Connection			
	What did you do today?	**What did you do yesterday?**	**Where there any obstacles?**
Member 1	Each member provides a brief summary of the tasks completed for the day.	Each member provides a brief summary of the tasks completed yesterday.	Each member identifies any obstacles they have encountered and how they are tackling them.

The daily check-ins will occur in the afternoon; because it was unanimously decided that mornings are the most productive period for the team and obstructing their focus will decrease their productivity.

Sprint Review

Once a sprint has completed a meeting with the product owner (supervisor) will be set to assess if the sprint has met the specific goals it set out to meet; if the sprint has not met the sprint goal it will go through an iterative process till it has. This forms part of the acceptance testing.

The upcoming sprint review will occur between the 14th and the 20th of July; the object for this sprint review is to show case the integrated work for milestone 1 (planning) and gather feedback from the product owner.

Project Management Tools:

The tools that will be utilised to manage the Hyacinth project are given below:

Monday.com	- Keep track of project sprints - Creation of a Gantt chart to identify critical tasks - Manage the team's work and keep track of the project's progress - Assign responsibilities to tasks
Python and JavaScript	- Data manipulation and exploration - ML model training - Front end dash board creation
Oracle online database	- Create a cohesive database for researchers.

H. Resource Management

Resource management in project management is estimating the required resources to complete the project; allocating the correct resources and utilising them efficiently to meet the project goals.
The first step will be to identify the required resources to meet each goal. The type or resources available for the Hyacinth project are:

- Human resources (the development team, community and researchers);
- Technical resources (software and software tools for project management);
- Time resources (deadlines and milestones).

Human resources (development team)

The **RACI (Responsibility Accountability Consulted and Informed) matrix** below will clarify roles assigned to the team for phase 2 and phase 3 of the project; this will form part of resource planning.

Project Task	Data Analyst (Nosipho)	Data Analyst (Sharon)	Data Analyst (Owethu)	Software engineer (Thandeka)	Software engineer (Oratile)	Software engineer (Omphile)
Database implementation	I	C	R	I	I	I
User interface	I	R	I	C	I	I
Ml model	R	C	I	I	I	A
API implementation	I	I	C	A	R	I
Data collection	A	I	C	R	I	I
Data processing	A	I	C	I	I	R
Web application development	I	R	I	A	I	I

	Responsibility: Individuals responsible for the development of the task and its completion.
	Accountability: Individuals accountable for the deadline of the tasks and anything that may go wrong related to the task.
	Consulted: Individuals whose options should be considered at every step of the development of the task at hand.
	Informed: Individuals that need to be kept in the loop of communication throughout the development of the task.

Resource allocation Plan

Task	Assigned To	Start Date	End Date	Resource Needed
Database implementation	Owethu, Sharon	13 July 2024	10 August 2024	MongoDB, Monday.com, Github, Microsoft Teams
User interface	Sharon, Thandeka	13 July 2024	10 August 2024	Javascript, python, bootstrap, Monday.com, Github, Microsoft Teams
Ml model	Nosipho, Omphile	13 July 2024	24 August 2024	Dataset, Python, Tensorflow, sklearn library, Monday.com, Github, Microsoft Teams
API implementation	Thandeka, Oratile	13 July 2024	27 July 2024	FastAPI, Postman, Database access, Monday.com, Github, Microsoft Teams
Data collection	Thandeka, Owethu	13 July 2024	3 August 2024	Database access, Python, Monday.com, Github, Microsoft Teams
Data processing	Omphile, Nosipho	13 July 2024	10 August 2024	ETL tools, SQL, Pandas library, Visualization Tools, Monday.com, Github, Microsoft Teams
Web application development	Sharon, Thandeka	13 July 2024	25 August 2024	React/Angular, Node.js, Deployment platform, Monday.com, Microsoft Teams Github

Equipment Required:

- Desktop computer or laptop that can support the development environment.
- Internet connection.
- External storage for database backup.

I. Risk Management

Risk management is the process of identifying what could go wrong in a project; responding to any risk that may arise; and planning a response strategy should the risk arise. Risk management is not only reactive in nature but proactive too.

In identifying and analysing risks for the Hyacinth project a proactive approach is employed; while the reactive approach will be employed for any unplanned risk that occur. The first step in this section will be to identify the risks (projectmanager, 2024).

1. What could go wrong?

a. Technical risks:

- **Data Inaccuracy:** Incomplete or inaccurate data during data collection will result in inaccurate model predictions. This will also impact the data analyst's work for the CBC research team; it will result in inaccurate analysis thus the data access requirement will not be met.

- **System Downtime:** during differential database updates and model retraining the system will be down; the research team will not be able to access the database for their research and analysis. The remaining shareholders will not be able to access the system to view the location and prediction of the mat.

- **API:** Should the API not be integrated with the system or it fails in any way; manual data collection will have to be employed; this will have a cascading effect that will result in the project being delayed.

b. Project Management Risks:

- **Delays**: These are schedule delays that would result in the project's timeline being compromised.

- **Scope creep:** Additional features and requirements being added during the projects lifecycle could result in a work overload for the teams; and compromise essential requirements if not managed efficiently. The chosen agile methodology (Scrum) is one of the safe guards for scope creep. It allows for the addition of requirements and features during iterations, while maintaining a hierarchy of tasks to be completed.

c. Operational risks

- **User adoption:** resistance of the community or the research team in using and adopting this solution to help meet their project goals. The impact of this risk would be "a solution created as a futile gesture".

The project does not have environmental or external risks, as those will be beyond the scope of this project and are limited to the research team and their biocontrol strategies.

2. Risk Assessment Matrix

Risk ID	Risk	Impact	Likelihood	(Impact * Probability)
R1.1	Data Inaccuracy	**Significant**	**possible**	**Medium High**
R1.2	System Downtime	**Minor**	**Very Likely**	**Low Medium**
R1.3	API Failure	**Sever**	**Likely**	**High**
R2.1	Delays	**Moderate**	**Likely**	**Medium**
R2.2	Scope Creep	**Significant**	**Possible**	**Medium High**
R3	User Adoption	**Negligible**	**Possible**	**Low**

Analysis of risks:

Data Inaccuracies
Although the impact of inaccurate data is significant to the success of the project; its likelihood is not high. This is because Google weather readings (which will be the main parameters for training the model) have safeguards against inaccurate data and can be cross referenced with other data sources from weather reports.

System Downtime:
The impact of the system down time is minor because it will amount to less than 1% (<1%) of the entire system's operational time. Hence the system will be operational 99% of the time. Should there be unexpected downtime, safeguards will be in place.

API Failure:
The impact of the API failure is server to the project's outcome; without the API connection the data required to proceed with the project cannot be obtained. The likelihood of this occurring is very likely; this is due to the fact that the team does not have experience working with an external APIs as of yet.

Scope Creep:
Scope creep is not necessarily bad for a project, but if not handled efficiently it can be detrimental to the success of the project hence its impact is server. The scope of the project must be continuously monitored before the start of a new sprint.

User Adoption:
It would be unfortunate if the system is not utilised, however the project can still conclude successfully given that its goals and requirements are met.

Mitigating strategies:

Risk ID	Risk	Mitigating strategy
R1.1	Data Inaccuracy	Cross reference the collected weather data from Google earth engine with that from the international site"weather.com".
R1.2	System Downtime	Backup the whole system and database onto the cloud. Thus should the system's down time run longer than expected; the system's downtime can always be aborted during the day while the system returns to its prior-backup state. This will allow for trouble shooting during the evening without inconveniencing the users.
R1.3	API Failure	Perform extensive research on the implementation of the APIs; and commence with the API integration as soon as possible in order to identify problems early on.
R2.1	Delays	Develop a detailed project plan with some buffer time between tasks.
R2.2	Scope Creep	Utilise the Scrum methodology to continuously prioritise and keep track of tasks.
R3	User Adoption	Engage with stakeholders earlier on; and collect feedback on the system; use this feedback to adjust the project accordingly. Include stakeholders in the project's life cycle as this will provide them with a sense of ownership over the final product, thus making them more likely to use it.

Risk Responsibility:

Risk ID	Risk	Responsibility
R1.1	Data Inaccuracy	Nosipho Omphile
R1.2	System Downtime	Owethu
R1.3	API Failure	Thandeka
R2.1	Delays	Nosipho Thandeka
R2.2	Scope Creep	Sharon
R3	User Adoption	Oratile

J. Version control

Version	Date	Author	Changes made	Approved by
1.0	1 June 2024	Nosipho	Initial doc	Team
2.0	**12 July 2024**	**Nosipho**	**Addition of user stories**	**Team**

K. Bibliography

projectmanager, 2024. *Risk management.* [Online]
Available at: https://www.projectmanager.com/blog/risk-management-process-steps
[Accessed 11 July 2024].

Ram Srinivasan, B. M., 2024. *ScrumAlliance.* [Online]
Available at: https://www.scrumalliance.org/about-scrum
[Accessed 7 July 2024].

University, R., 2020. Bugs are winning the war against Hartbeespoort's hyacinths. *Rhodes University.*