

IoT Frontend Manager Project Proposal

Prepared for:

Michael McGlynn, Adroit Environmental IoT

Prepared by:

Otis Wales, Team Leader - psg3398@autuni.ac.nz

Miguel Emmara – ryf2144@autuni.ac.nz

Glenn Neil - pxw9116@autuni.ac.nz

Munish Kumar - kby0857@autuni.ac.nz

Mentor:

Akbar Ghobakhlou - akbar.ghobakhlou@aut.ac.nz

Prepared on: **18-08-2023**

Version: 1.1

Table Of Contents

Terms of Reference.	3
Rationale for the Project.	5
Project Objective.....	6
Risks and Issues Management.	10
Risk Register.....	1
Skills Analysis.	1
Project Feasibility.....	2
Team Roles And Responsibilities.....	4
Project Management Methodology.....	6
Estimate Project Costs.	7
Project Schedule.....	1
Appendix.	1
References	2

Terms of Reference.

Adroit is a New Zealand company which does IoT-based (Internet of Things) environmental monitoring in New Zealand. This means that they capture data around New Zealand using their IoT sensors. They have various sensors that they use as they are involved in several types of environmental monitoring.

Adroit captures the data, maintains the sensors and sends their clients the data through a front-end application (Separate to the front-end manager which this R&D project will be dedicated to developing).

Adroit works across various sectors in New Zealand such as **environmental management, water catchment, worksite, and agriculture/aquaculture monitoring**. They cover an extensive range of clients such as the Department of Conservation, Fonterra, Mercury Energy, Auckland Council and many more. They were established in 2014 in Auckland and currently have an office in central Auckland. Spark is invested in Adroit and owns a significant percentage of the company's shares at 47% of ownership.

Adroit's expertise position in the realm of Environmental IoT Monitoring devices results from strategic collaboration with technology partners. Some Notable technology partners include Spark IoT, Amazon Web Services (AWS), and global equipment specialists like Libelium, S::CAN, In-Situ, and Hills Laboratories. Qrious, a prominent data verification and analysis entity, adds an essential layer of credibility to the data-driven insights that our client, Adroit provides.

The Problem

The issue that Adroit has is that they currently do not have a front-end application for the delivery team to use. They currently must view an email sent **every 6 hours** from AWS to see which devices are offline. Since Adroit has over **400 devices** this can be troublesome and tedious and means that at least one person must read the e-mail or else they will not know which devices are offline. This is the problem that we aim to solve by delivering a front-end web application for them.

This project aims to solve this problem by creating a front-end web application that contains a **dashboard** for the delivery team to monitor their sensors across New Zealand. From this application, the delivery team at Adroit should be able to see the status of the sensors i.e., if the status is offline/online, battery status, and identifying any faults. The sensors' status and information about the sensors will be displayed on a front-end web application. Encompassing factors like air quality, water levels, soil health, and more. By having this application, we believe the delivery team can efficiently monitor the status of sensors, interpret data trends, and swiftly respond to any anomalies.

Project Charter.

Project Title: IoT Adroit

Project Start Date: 17 July 2023

Projected Finish Date: 21 June 2024

Budget Information: \$73,326.30

Project Manager: Otis Wales, psg3398@autuni.ac.nz

The objective of this project is to send Adroit's data through RESTful API onto a responsive frontend web application dashboard for them to effectively monitor the devices in a reliable, real-time manner.

Main Project Success Criteria: The goal of this is to provide a comprehensive tool for the delivery team at Adroit to stay updated and quickly react to any changes with their sensors. They will be able to monitor sensors in real time.

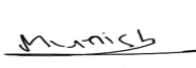
Approach: We will create a full-stack web app with features such as:

The features that we plan to incorporate into this application include sensor status, the end user should be able to view the devices in a table like format to see which devices are offline. The user should also be able to monitor the battery status of the device, ie if its green the health of the battery is good and if its orange the battery is decaying and if red it needs replacement. We also will have user authentication so that sensitive information can only be reviewed by authorised people.

Role	Name	Organization/ Position	Contact Information
Full Stack	Otis Wales	Team Leader	psg3398@autuni.ac.nz
Full Stack	Miguel Emmara	Team Member	ryf2144@autuni.ac.nz
Full Stack / UX	Glenn Neil	Team Member	pxw9116@autuni.ac.nz
Full Stack	Munish Kumar	Team Member	kby0857@autuni.ac.nz

Roles and Responsibilities

Sign-off: (Signatures of all above stakeholders. Can sign by their names in the table above.)

Miguel Emmara: Otis Wales: Munish Kumar Glenn Neil: 

Rationale for the Project.

Key Issues

Adroit currently lacks an in-house dashboard this means that they do not have a personal dashboard for the delivery team to use. Instead, the delivery team relies on reading e-mail notifications to view the status of which devices are offline. This method is inefficient as it means that at least one user must view the email every 6 hours if they want to know the status of the devices. If a device goes offline and then goes back online within the 6 hours (prior to next e-mail) then it is hard for the delivery team to currently know the status of the device, this shows that Adroit is currently lacking real-time monitoring.

The other issue that Adroit has is that this approach with e-mails means that there is currently no anomaly detection. For example, if the sensor reading is wrong how is Adroit meant to know that unless they manually look at the data and see that the data readings are 'off' / different to what is expected. For example, pH sensors can drift inaccurately and read values which are impossible, if this is to happen it means that Adroit is collecting incorrect data.

Rationale

Building a front-end dashboard would speed up the reaction time of the delivery team as instead of viewing an e-mail every 6 hours they can view the dashboard at any time and see the real time status of the devices. Having early detection of sensor issues will allow for Adroit to minimise downtime and prevent potential data loss this will improve client/business relationships and help elevate trust in the Adroit brand.

On completion of this project Adroit would have a user-friendly dashboard for their team to use. This dashboard would allow for them to have more control over their data. Given that we implement the minimum viable product, we plan to implement data visualisation in the dashboard so that the Adroit team can do complex monitoring we believe that data visualisation will be helpful to the delivery team so that they can monitor their data effectively.

Project Objective.

Aim: Project *Adroit's* data through RESTful API onto a responsive frontend web application dashboard for them to monitor the devices in a convenient and reliable manner.

The Adroit team should be able to easily see which devices are offline so that they can quickly respond to the issue by sending out a contractor to fix the device.

The goal of this is to provide a comprehensive tool for the delivery team at Adroit to stay updated and quickly react to any changes with their sensors. They will be able to monitor sensors in real-time.

Features:

View sensor status (offline/online).

The dashboard will offer a clear view of the status of each sensor, indicating whether they are online or offline.

Battery Health Monitoring

A traffic light system will visually convey battery status using colors: green for healthy batteries, orange for decaying battery health, and red for batteries needing replacement.

User Authentication.

A secure user authentication system will be implemented to ensure that only authorized personnel can access the dashboard and its functionalities.

Fault Identification:

- Algorithms will analyse sensor data to identify any anomalies or faults.
- Devices displaying unusual readings (e.g., pH levels beyond acceptable thresholds) will be flagged.
- The dashboard will have a dedicated section/category for devices deemed faulty, facilitating quick identification and action.

This will be the minimum viable product (**MVP**) which we aim to deliver, if time allows, we will also be focusing on adding additional features.

Additional features (Stretch Goals):***Heat Maps***

Introducing heat maps that visually represent sensor data patterns across different parameters, offering deeper insights briefly.

Statistics

Adding statistical analysis and visualizations to help Adroit's team gain a comprehensive understanding of sensor performance trends and patterns.

Fault Identification:

- Algorithms will analyse sensor data to identify any anomalies or faults.
- Devices displaying unusual readings (e.g., pH levels beyond acceptable thresholds) will be flagged.
- The dashboard will have a dedicated section/category for devices deemed faulty, facilitating quick identification and action.

Project Scope.

The project we develop involve a development of a responsive, fast, and easy to navigate dashboard with focus on both functional and non-functional requirement.

- **Functional Requirements**

Our functional requirements contain several key features.

- **Responsive design**

To ensure that the dashboard is compatible across mobile and various desktop displays. This feature aims to provide users with a seamless experience regardless of their chosen devices.

- **Offline Device table**

This table will clearly display IoT devices which are currently in an offline state. This feature is important for users to quickly identify devices that are not actively connected.

- **Traffic light system (colour) for battery health of devices.**

This will display next to each sensor so that the user can see what the battery's health. Enabling users to easily determine the status of device batteries.

- **User Authentication**

This security measure ensures that only authorized individuals can access the application and its functionalities.

- **Display all IoT sensors/devices on the dashboard**

Additionally, all IoT sensors and devices will be displayed on the dashboard, offering users a comprehensive overview of the network's status.

- **Non-Functional Requirements**

Our non-functional requirements contain several key features.

- **Comments only on complex code**

Complex code will be the focus of comments, streamlining comprehension and troubleshooting without overwhelming the codebase with unnecessary comments.

- **Documentation of pipeline, and codebase**

Thorough documentation of the entire pipeline and codebase will be maintained, facilitating understanding, onboarding, and future development.

- **Unit tests before merging to master branch.**

Prior to merging with the master branch, unit tests will be used. This quality assurance practice will identify and revise issues before they impact the stability of the entire application.

- **Security – Encryption through HTTPS**

To maintain the security even further, the system will be encrypted using SSL HTTPS protocol, ensuring the confidentiality and integrity of data transmitted between users and the application.

- **Maintainability**

For Adroit to extend the code base it should be easy to do so, this means programming must be done in a way which isn't confusing and is easy for them to extend.

Risks and Issues Management.

Adroit maintains the sensitive data of their clients therefore, we must secure the data and visualise the in-table format. There are some steps which we can take to identify the risk and produce a solution.

Scope Risks:

- **Risk: Michael (a key team member) becomes unavailable due to illness or other reasons.**
 - Likelihood: Medium
 - Impact: High (Potential delays, loss of expertise)
 - Mitigation: Regularly updating team documentation to ensure anyone can pick up where he left off.
 - Responsible Party: Project Manager
- **Risk: Unclear Requirements - Ambiguous or unspecified requirements can lead to misinterpretations and undesired outcomes.**
 - Mitigation: Work closely with stakeholders to define clear, detailed, and unambiguous requirements. Use techniques such as requirement elicitation sessions, user story mapping, or prototyping.
- **Risk: Scope Creep - Additional features or changes are added to the project without proper adjustments to time, budget, or resources.**
 - Mitigation: Establish a robust change management process. Any changes to the original scope should be formally reviewed, approved, and communicated.

Risk Register.

Risk Register for Adroit IoT Frontend Manager										
Prepared by: Otis Wales				Date: 18/08/2023						
No.	Risk/Impa	Name	Description	Category	Root Cause	Triggers	Potential Responses/Mitigation	Risk Owner	Probability	Status
1	Medium	Client sick	Michael (a key client) becomes unavailable due to illness or other reasons.	Scope Risk			Regularly updating team documentation to ensure anyone can pick up where he left off.	Team Leader	Medium to low	Active
2		Scope Creep	Scope Creep - Additional features or changes are added to the project without proper adjustments to time, budget, or resources.	Scope Risk		Client asking for alot of different items	Establish a robust change management process. Any changes to the original scope should be formally reviewed, approved, and communicated.	Munish	High	Active
3		Unclear Requirements	Unclear Requirements - Ambiguous or unspecified requirements can lead to misinterpretations and undesired outcomes.	Scope Risk		Gold Plating, Vague Requirements	Communicate with client deeply and ask more questions	Munish	Medium	Active
4	Medium	Sick team member	Team member gets sick		Stress and Burnout	Stress/Burnout, Personal Health issues	Need to make sure that everyone in the team is skilled to do the task that the other member was doing	Miguel	Low	Active
5	Critical	Client withdraws	Client decides they don't want to work with us anymore	Client Risk	Economic Downturn, communication problems	Client starts lacking communication	Not much can be done about this, but the chance of this happening is low as our client communicates with us	Team Leader	Low	Active
6	High	AWS Bankrupt	Amazon goes bankrupt or suddenly AWS disappears	Technical Risk	Economic Downturn.		Adroit would need to migrate to another platform. It would mean that we would need to use another platform to pull data from	Team Leader	Extremely low, near impossible	Active
7	High	Team conflict		Team Risk	Personal disagreements, work stress		Find a way to resolve the conflict, to prevent conflict we need to have a good team culture and respect each other and not micro manage	Team Leader	Low	Active
8	Low	Integration Failures	Different IoT devices might use different protocols or data formats. Integrating them onto one dashboard might pose challenges.	Technical Risk		Software Updates, Vendor Changes	We will need to look more into the data and see how different the formats will be and what is in common	Glenn	Medium to low	Active

For our risk register it is crucial that we keep it updated throughout the project so that we are aware of potential risks

Skills Analysis.

A	B	C	D	E	F	G	H	I	J
IT Skill Matrix (IoT Frontend Manager)									
Available Technical Resources							LOW = 1,2,3	MED = 4,5	HIGH = 6,7,8
Group	Skill Expertise	Munish Kumar	Otis Wales	Glenn Neil	Miguel Emmara	Total	EXPERT = 9,10		
Project Management Tools	GIT	8	6	6	7	27			
	Trello	9	8	9	9	35			
	BitBucket	1	1	1	1	4			
	Scrum	8	5	8	8	29			
	Microsoft Office Apps	7	4	5	7	23			
Backend	S3 Bucket	1	1	1	1	4			
	Lambda	1	1	1	1	4			
	Cognito	1	1	1	1	4			
	AWS Databases	3	3	3	3	12			
	Postman	5	5	5	5	20			
	Node Js	5	7	5	5	22			
	Restful API	3	3	3	4	13			
FRONT END	NEXT_JS	4	4	3	3	14			
	TYPESCRIPT	3	3	5	3	14			
	CSS	5	8	5	3	21			
	TAILWIND CSS	1	8	1	6	16			
	REACT	4	6	6	6	22			
	HTML	8	10	7	10	35			
TESTING	CI/CD	4	3	3	3	13			
	UNIT TESTING (JEST)	3	3	3	4	13			
DESIGN	FIGMA	3	5	4	3	15			
	MIRO	2	1	7	3	13			

Skill Matrix

As a team we have various levels of skills across various domains. From our brainstorming we have identified five categories of skill domains in which we must upskill. As a team it is crucial that we all have skills in the front-end and back-end as this will allow us to build the application.

1. Project Management Tools
2. Backend
3. Front-End
4. Testing
5. Design

From the Skill Matrix above we have identified skills that we are weak in. These include our AWS/Backend ability, our skills in using Next JS. These are the most crucial skills that we need to upskill as everyone in the team is working as a full-stack developer. It is crucial that we are competent in using Next JS as well as being confident in doing API calls from the AWS backend.

Though not as important, it is crucial that as a team, we understand how to deliver unit testing as we want to minimise the chance of unexpected behavior in the application.

IT-Specific Skills (BCIS Degree):

- Web Development with TypeScript: Proficiency in TypeScript and the ability to develop dynamic web applications using the chosen framework.
- Cloud Services and AWS: Understanding of Amazon Web Services (AWS) components, especially those related to IoT Core, Lambda, S3, DynamoDB, RDS, API Gateway, and Cognito.
- Front-End Development: Ability to design responsive and user-friendly front-end interfaces for web applications.

Personal, Professional, and Technical Capabilities:

- Team Collaboration and Communication: Effective communication and collaboration skills are essential to ensure seamless teamwork and project coordination.
- Problem-Solving and Troubleshooting: A strong problem-solving mindset to address technical challenges and unforeseen issues during development.
- User Experience (UX) Design: Knowledge of UX principles to create intuitive and user-friendly interfaces.
- Project Management: Understanding project management concepts to ensure timely execution and meeting of milestones.
- Version Control (Git): Proficiency in using version control tools for collaborative development.
- Security Awareness: Understanding of security best practices to ensure data protection and user privacy.
- Time Management: Effective time management to balance project work and upskilling efforts.

Acquiring Skills and Addressing Knowledge Gaps:

- TypeScript and Front-End Framework: Allocate time for individual upskilling using online resources, tutorials, and documentation specific to TypeScript and the chosen front-end framework (e.g., React, Angular, Vue).
- AWS Services: Explore AWS documentation, online courses, and tutorials related to IoT Core, Lambda, S3, DynamoDB, RDS, API Gateway, and Cognito.
- IoT Integration: Engage in hands-on labs, tutorials, and examples related to IoT device integration with AWS services.
- Cognito Integration: Study Amazon Cognito documentation and learn about user authentication and management.
- Security Best Practices: Study security guidelines and best practices for securing applications and data on AWS.
- Version Control (Git): Familiarize yourselves with Git and GitHub through tutorials and practical exercises.
- UX Design Principles: Study UX design principles through online resources and practical exercises.

Project Feasibility.

Infrastructure:

Adroit has a strong basis for this project thanks to its current Internet of Things infrastructure and connections with major global technology companies like Spark IoT and AWS.

Instruments and Technologies:

The following IT-specific abilities from a BCIS degree will be utilised by the project:

Knowledge of TypeScript and front-end frameworks.

To deliver this project, deep understanding in Typescript is needed as well as knowledge in popular front-end frameworks like React or Angular.

AWS Microservices.

Client current infrastructures lays on AWS services. Knowledge in AWS is needed to deliver this project. The use of AWS's IoT Core, Lambda, S3, DynamoDB, RDS, API Gateway, and Cognito services as well as other cloud services for backend support.

Roles and abilities:

The project will need:

- Frontend developers who are skilled in TypeScript, their preferred frontend framework, and UI/UX design.
- Project managers: To supervise the project, oversee milestones, and guarantee prompt execution.
- Engineers in quality assurance: To carry out extensive testing and find any flaws.

*Custom Development or Off the Shelf Software***Customised Development:**

The specific specifications and integration demand of Adroit are met by a customised solution. By using in-house custom apps, this will ensure a precise fit for their specific needs.

Off-the-Shelf Options:

Even if they are inexpensive, these might not be customised and may not offer smooth integration. Microservices vs. Monolithic Architecture

Monolithic:

Simplifies deployment and development, ideal for Adroit's present operational requirements and size.

Microservices Strategy:

While offering scalability and modularity, the microservices approach can introduce complexities during deployment.

Scalability Feasibility:

The front-end application needs to be scalable because there are over 400 devices and a wide variety of clients. As Adroit's operations expand, the frontend application's architecture must be designed to seamlessly accommodate growing user and device demands as Adroit's activities expand.

Team Roles And Responsibilities

Team Roles and Responsibilities

- **Team Leader** – Otis Wales

As **team leader**, Otis serves as the primary point of contact between our mentor and our client. This can include communications, addressing concern, and setting up meeting. Otis Also contributes as a **Frontend Developer**, leveraging his expertise to create beautiful and responsive user interfaces.

- **UX/UI Design** – Glenn Neil

Glenn is our **UX/UI Designer**. He leads the creation of wireframe and prototypes for the frontend application, transforming brainstorm concepts into visual representations to guide the development process.

- **Full Stack** – Munish Kumar

Munish takes part on both **Frontend** and **Backend**, including **Quality Assurance**. He will ensure a cohesive and functional web apps across all screen layers. Munish and Miguel will bring UX/UI designs to life.

- **Full Stack** – Miguel Emmara

Miguel takes part on both **Frontend** and **Backend**, including **Quality Assurance**. He will ensure a cohesive and functional web apps across all screen layers. Munish and Miguel will bring UX/UI designs to life.

Upskilling.

Several types of upskilling will be needed for this project. This can be broken up into three primary areas. **Front-end, backend and design.** Different members will do different upskilling as we all have different areas of skill and different areas that members will be focused on in the project.

AWS Upskilling

This is not the most important but will help with learning how AWS works so that we can utilise the data. For upskilling AWS there are many resources online as well as training through the AWS website.

Design

Having a solid foundation in UI/UX design is crucial for our project as we want to deliver a clean interface for the user to use. We do not want to create a confusing interface and want the navigation of the interface to be as simple as possible. To do this we will need to upskill design.

Design upskilling can be done through reading books on UI and UX design and purchasing of Udemy courses or reading case studies online.

Frontend

We can do upskilling on the frontend technology that we are using such as upskilling Next JS this can be done by watching YouTube tutorials, purchasing Udemy courses or by consuming the Next JS documentation online. As we are also using Tailwind CSS it is crucial that everyone in the team understands how to use Tailwind CSS this can also be done with the above methods.

Upskilling will be needed on this project in terms of understanding how the AWS works. For this we can upskill for free online it is crucial to upskill this as we are using the AWS for the backend

https://explore.skillbuilder.aws/learn/public/learning_plan/view/84/developer-learning-plan?la=sec&sec=lp

UI/UX will be part of the upskilling, we will need to learn about how to make the 'perfect design' as if the client has a bad experience with the interface, it may make it bad for them.

We can upskill through LinkedIn learning as it is free, and they have various content which will be applicable for us to use. We will be allocating 3 hours a week to upskilling.

Expected Work Behaviour and Practices

- All team members are expected to communicate openly, share ideas, and actively participate in discussions.
- Regular updates on individual tasks and overall project.
- Challenges and issues should be addressed within the team.

Project Management Methodology.

A dedication to quality and client satisfaction sits at the core of the Adroit initiative. We use the Agile technique as a foundation for direction as we work toward these objectives. Our project's dynamic nature perfectly aligns with the agile ideals of adaptability, iterative development, and client participation.

Agile gives us the ability to change the software development process into a never-ending cycle of improvement. By segmenting the project into reasonable iterations, we make sure that each stage is painstakingly created, tested, and improved. This iterative method makes it easier to receive frequent client feedback, allowing us to quickly adjust in response to shifting needs and ensuring that a solution develops in line with client expectations.

Pros:

Iterative Design and Evaluation:

Our ability to create the Adroit project incrementally and continuously solicit client feedback is made possible by Agile's iterative framework. The environment created by this dynamic method encourages the software to develop gradually under the guidance of insightful client feedback. The final output will precisely match the client's concept thanks to this tried-and-true technique.

Capacity for Change:

We are equipped to navigate the inherent risks in software development thanks to the agile approach. Agile offers the adaptability required to quickly adjust to new priorities, ensuring continuous progress in an environment where specifications can change.

Cons:**Scope Creep Possibility:**

The risk of scope creep exists because of Agila's emphasis on customer collaboration and changing requirements. To keep the project on track, it is crucial that we carefully monitor the scope of each iteration.

Client Availability Dependence:

Consistent customer involvement is key to the success of agile. The project's momentum could be affected by fluctuating client availability. To lessen this possible difficulty, efficient communication techniques will be used.

Estimate Project Costs.

We have estimated this project to cost \$73,326.30. This accounts for factors such as:

Name	Price	Hours	Subtotal
Mentor Cost	\$142/hr	24	\$3,408
Team labour	\$50/hr	1200	\$60,000
Software	\$114	0	\$114
Travelling Cost	\$12	20	\$240
Subtotal:			\$63,762
GST rate 15%			\$9,564.30
Total:			\$73,326.30

Adroit Front-end Manager Project Cost Estimate

Prepared by: Glenn Neil	Date:	8/08/2023		
	# Units/Hrs.	Cost/Unit/Hr.	Subtotals	WBS Level 1 Totals
WBS Items				
1. Project Management				
1.1 Project mentor	24	\$142	\$3,408	
1.2 Project team members	1200	\$50	\$60,000	
1.3 Project Leader/Manager			\$63,408	\$63,408
3. Software				
3.1 Iot Core	1	\$57.60		
3.2 Lambda	1	\$14		
3.3 S3 Bucket	1	\$20		
3.4 Dynamo DB and API Gateway		0		
3.5 Cognito		22		
Total		114	\$114	114
5. Support				
5.2 Travel cost/Uber	12	\$20	\$240	240
6. Tax rate 15%			\$9,564	\$9,564
Total project cost estimate				\$73,326

Project Cost Estimate Excel

Project Schedule.

We meet every week on Tuesdays from 9am-2pm. We also meet offsite and work through Discord on Sunday evenings at 6pm. This is the time in which we have our team meetings to discuss progress and what has been happening with the project.

We also meet our R&D mentor, Akbar Ghobakhlou, weekly to discuss what has been happening with the project.

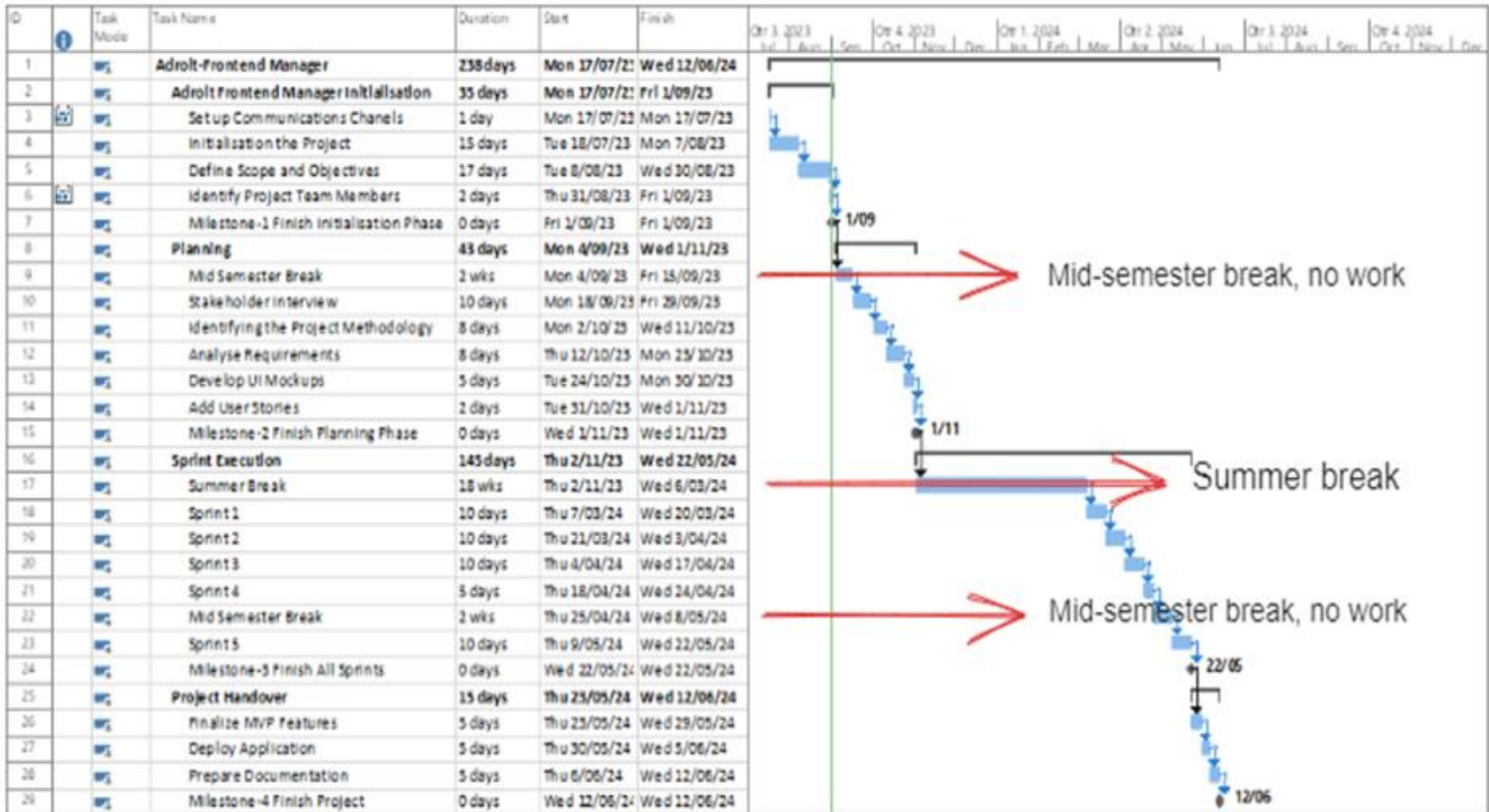
Every fortnight we have a meeting with our client Michael at the Adroit office to discuss the project and to get feedback.

Event	Frequency	Day/Time	Location/Platform
Team Meeting	Weekly	Tuesday, 9am - 2pm	Onsite
Teem Meeting	Weekly	Sundays, 6 PM	Discord
Meeting With Mentor (Akbar)	Weekly	TBD (Flexible)	TBD
Meeting With Client (Michael)	Fortnightly	TBD (Scheduled with client)	Adroit Office

For the first semester of the project (**Part 1**) we will be focused on research and planning of the project. This will involve interviewing the key stakeholders (Michael of Adroit & John) to find out the requirements and the problems that the Adroit team are faced with. We will try to address these problems through planning and breakdown the requirements into a feasible plan. This is what we entice to do on the first stage of the project.

For **Part 2** of the project the focus will be placed on development, testing and deployment of the application. In this stage of the project, we will be doing sprints in a fortnightly manner. In these sprints the focus will be on delivering multiple user-stories and testing the code. After each sprint we will be getting feedback from our client, Michael, in which he will review the code and give feedback.

Testing will take place in this stage of the project; this will involve unit testing and *TDD* (Test driven development) This stage will also involve deployment of the application.



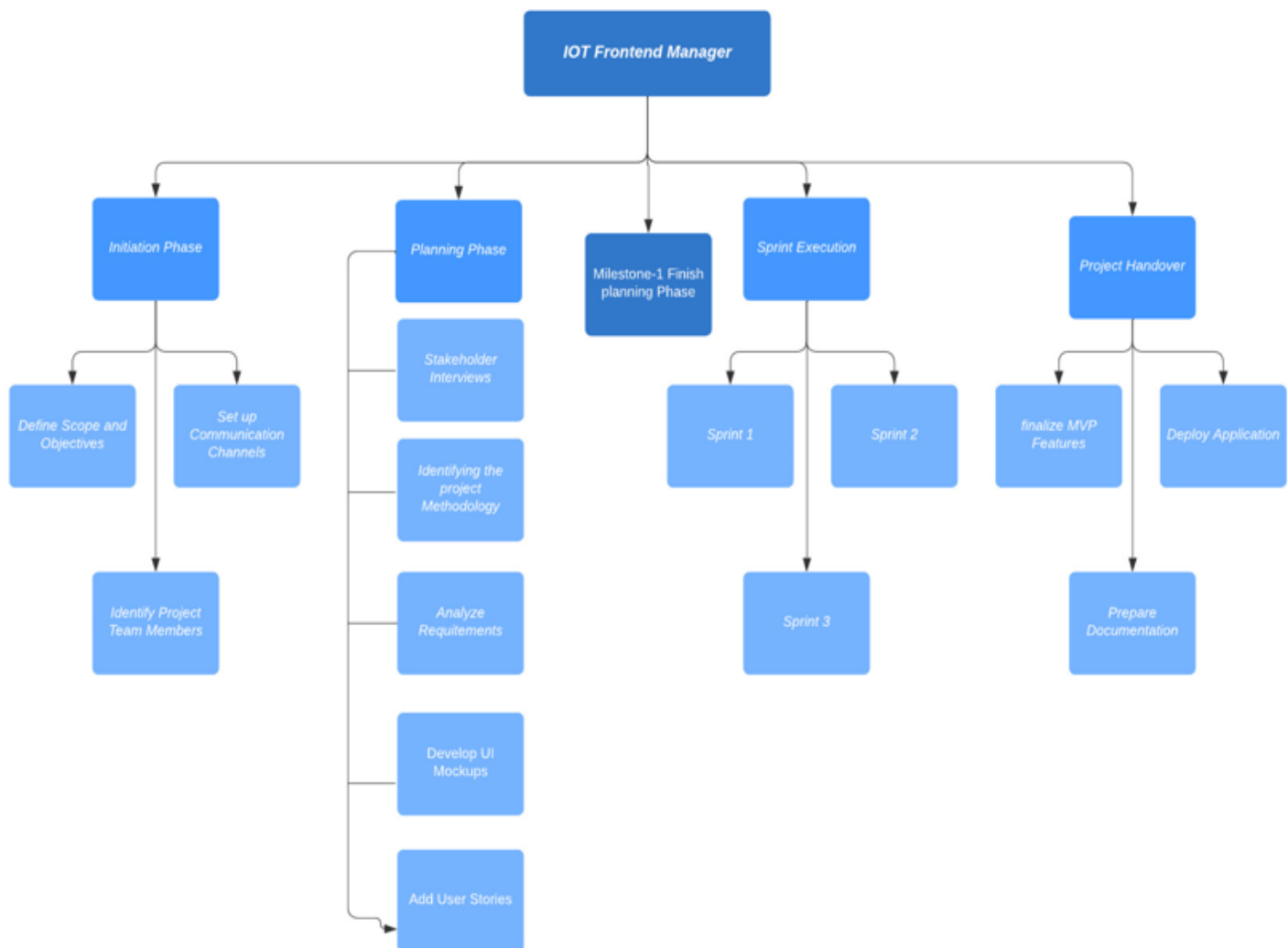
Gantt Chart

Milestone.

Since we are still in the planning stage of the project the current Milestone that we have identified is the MVP (Minimum viable product).

Deliverables

- Documentation
- MVP
- Testing of application.
- Comments in code base. (Only on complex functions)



Work Breakdown Structure

IoT Frontend Manager Team Agreement.

Goal: To excel as a team to develop IoT Frontend Manager software, while producing high-quality documentation to attain excellent marks and to improve our technical/collaborative skills in the process.

Rules & Expectations:

- The product leader shall have the final say on suggested ideas.
- We will communicate via:
 - Discord
 - WhatsApp
 - Email
 - Microsoft Teams
- Primary information communications within the team will be via Discord, and via WhatsApp if communications are imminent.
- Communicate respectfully with one another.
- Attend meetings.
 - Contact team members if you are unable to attend the meeting.
 - Cover for sick team members.
- Any new change must be communicated with the team before proceeding.
- Must reply with an acknowledgement/response to any email/Discord sent to you from the team within 24 hours.
- The Product leader is the main contact for communication with the client. He must immediately communicate important emails and messages with the team.
- Aim to complete assigned tasks within the time limit assigned.
- Everyone should have equal and fair opportunity to participate in the teams' decisions.

Procedures:

- The team will adhere to all agreed policy documents posted to Trello or Microsoft Project.
- The development team will do their best to adhere to the guidelines and conventions specified on Trello or Microsoft Project.

Meeting Guidelines:

- Meetings will occur at minimum weekly.
- Meetings with mentor will occur fortnightly or weekly as required.
- Meetings with a client will occur fortnightly or weekly as required.
- Meetings must start on time.
- Any apologies must be communicated to the team at least 5 hours BEFORE the meeting takes place.

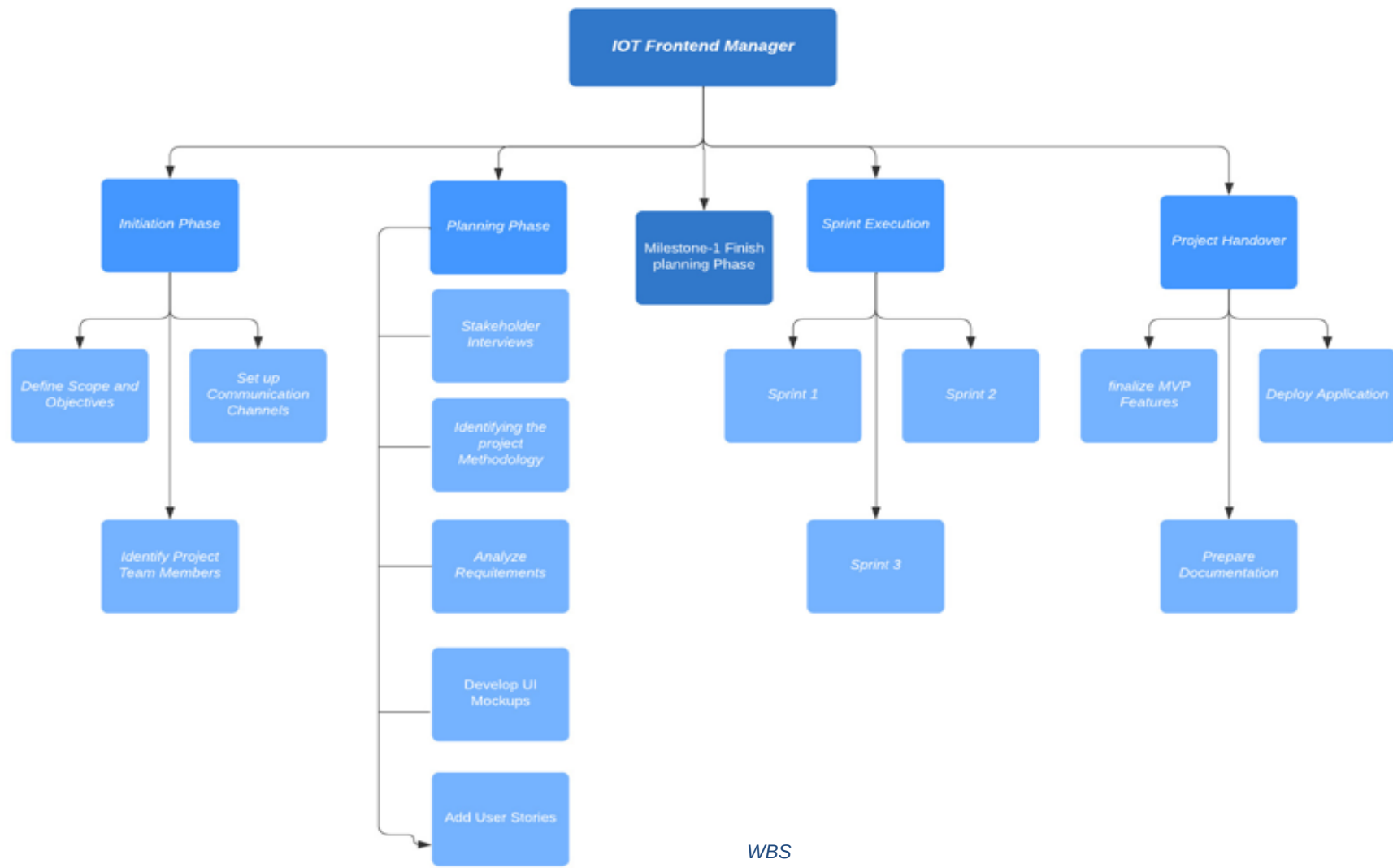
Conflict Resolutions:

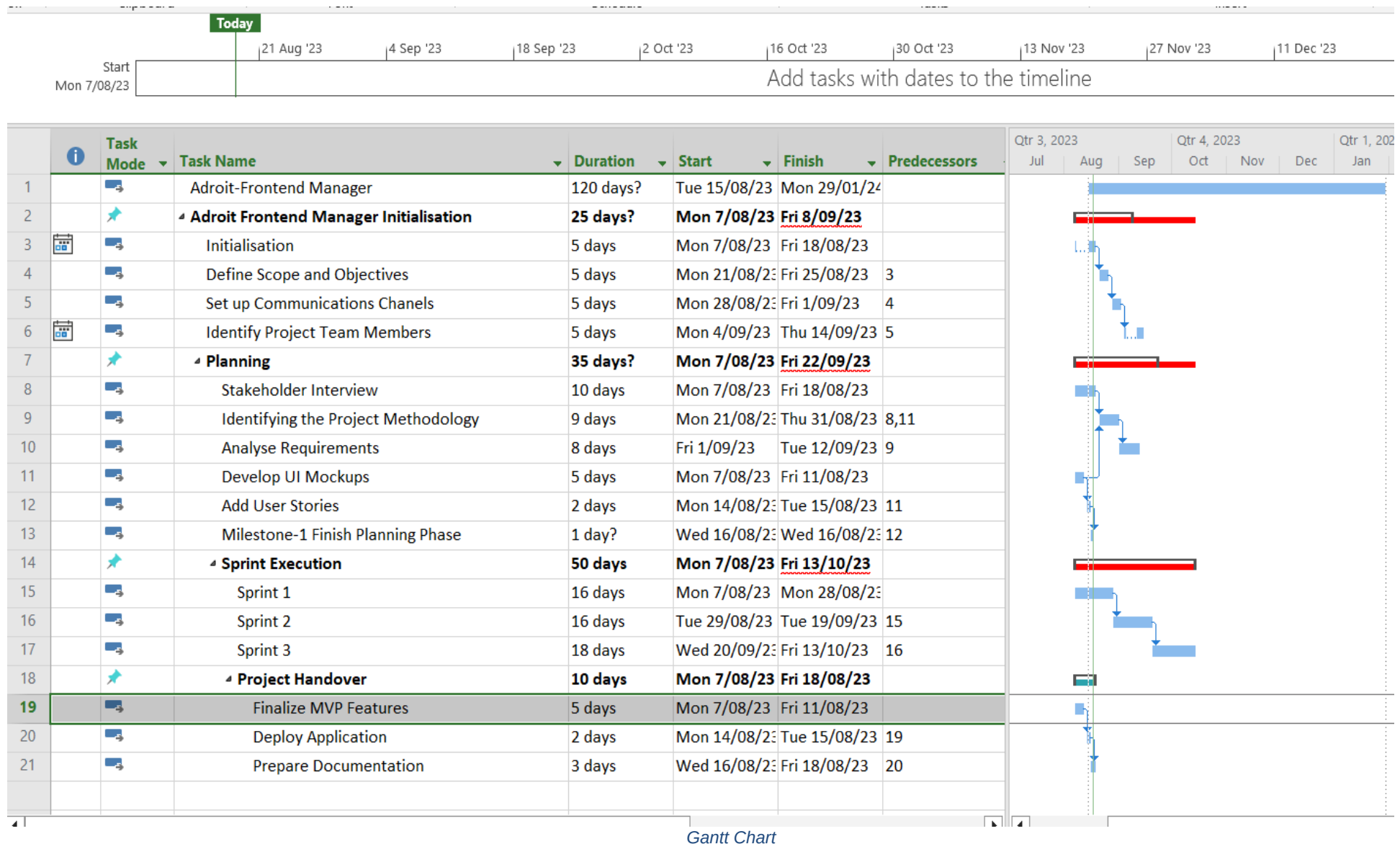
- Coin Flip
- Team Vote

Rule breaking Resolutions:

- Must shout the team pizza/chocolate.
- Apologise to the team.

Appendix.





Risk Register for Adroit IoT Frontend Manager

Prepared by: Otis Wales

Date: 18/08/2023

No.	Risk/Impa	Name	Description	Category	Root Cause	Triggers	Potential Responses/Mitigation	Risk Owner	Probability	Status
1	Medium	Client sick	Michael (a key client) becomes unavailable due to illness or other reasons.	Scope Risk			Regularly updating team documentation to ensure anyone can pick up where he left off.	Team Leader	Medium to low	Active
2		Scope Creep	Scope Creep - Additional features or changes are added to the project without proper adjustments to time, budget, or resources.	Scope Risk		Client asking for alot of different items	Establish a robust change management process. Any changes to the original scope should be formally reviewed, approved, and communicated.	Munish	High	Active
3		Unclear Requirements	Unclear Requirements - Ambiguous or unspecified requirements can lead to misinterpretations and undesired outcomes.	Scope Risk		Gold Plating, Vague Requirements	Communicate with client deeply and ask more questions	Munish	Medium	Active
4	Medium	Sick team member	Team member gets sick		Stress and Burnout	Stress/Burnout, Personal Health issues	Need to make sure that everyone in the team is skilled to do the task that the other member was doing	Miguel	Low	Active
5	Critical	Client withdraws	Client decides they don't want to work with us anymore	Client Risk	Economic Downturn, communication problems	Client starts lacking communication	Not much can be done about this, but the chance of this happening is low as our client communicates with us	Team Leader	Low	Active
6	High	AWS Bankrupt	Amazon goes bankrupt or suddenly AWS disappears	Technical Risk	Economic Downturn.		Adroit would need to migrate to another platform. It would mean that we would need to use another platform to pull data from	Team Leader	Extremely low, near impossible	Active
7	High	Team conflict		Team Risk	Personal disagreements, work stress		Find a way to resolve the conflict, to prevent conflict we need to have a good team culture and respect each other and not micro manage	Team Leader	Low	Active
8	Low	Integration Failures	Different IoT devices might use different protocols or data formats. Integrating them onto one dashboard might pose challenges.	Technical Risk		Software Updates, Vendor	We will need to look more into the data and see how different the formats will be and what is in	Glenn	Medium to low	Active

Risk Register

Adroit Front-end Manager Project Cost Estimate

Prepared by: Glenn Neil	Date:	8/08/2023		
	# Units/Hrs.	Cost/Unit/Hr.	Subtotals	WBS Level 1 Totals
WBS Items				
1. Project Management				
1.1 Project mentor	24	\$142	\$3,408	
1.2 Project team members	1200	\$50	\$60,000	
1.3 Project Leader/Manager			\$63,408	\$63,408
3. Software				
3.1 Iot Core	1	\$57.60		
3.2 Lambda	1	\$14		
3.3 S3 Bucket	1	\$20		
3.4 Dynamo DB and API Gateway		0		
3.5 Cognito		22		
Total		114	\$114	114
5. Support				
5.2 Travel cost/Uber	12	\$20	\$240	240
6. Tax rate 15%			\$9,564	\$9,564
Total project cost estimate				\$73,326

Project Cost Estimate Excel

A	B	C	D	E	F	G	H	I	J
IT Skill Matrix (IoT Frontend Manager)									
Available Technical Resources						LOW = 1,2,3	MED = 4,5	HIGH = 6,7,8	
Group	Skill Expertise	Munish Kumar	Otis Wales	Glenn Neil	Miguel Emmara	Total	EXPERT = 9,10		
Project Management Tools	GIT	8	6	6	7	27			
	Trello	9	8	9	9	35			
	BitBucket	1	1	1	1	4			
	Scrum	8	5	8	8	29			
	Microsoft Office Apps	7	4	5	7	23			
Backend	S3 Bucket	1	1	1	1	4			
	Lambda	1	1	1	1	4			
	Cognito	1	1	1	1	4			
	AWS Databases	3	3	3	3	12			
	Postman	5	5	5	5	20			
	Node Js	5	7	5	5	22			
	Restful API	3	3	3	4	13			
FRONT END	NEXT JS	4	4	3	3	14			
	TYPESCRIPT	3	3	5	3	14			
	CSS	5	8	5	3	21			
	TAILWIND CSS	1	8	1	6	16			
	REACT	4	6	6	6	22			
	HTML	8	10	7	10	35			
TESTING	CI/CD	4	3	3	3	13			
	UNIT TESTING (JEST)	3	3	3	4	13			
DESIGN	FIGMA	3	5	4	3	15			
	MIRO	2	1	7	3	13			

Skill Matrix

Project Schedule

Event	Frequency	Day/Time	Location/Platform
Team Meeting	Weekly	Tuesday, 9am - 2pm	Onsite
Teem Meeting	Weekly	Sundays, 6 PM	Discord
Meeting With Mentor (Akbar)	Weekly	TBD (Flexible)	TBD
Meeting With Client (Michael)	Fortnightly	TBD (Scheduled with client)	Adroit Office

Auckland University of Technology Bachelor of Computer & Information Sciences
Research & Development Project

Disclaimer:

Clients should note the general basis upon which the Auckland University of Technology undertakes its student projects on behalf of external sponsors:

While all due care and diligence will be expected to be taken by the students, (acting in software development, research, or other IT professional capacities), and the Auckland University of Technology, and student efforts will be supervised by experienced AUT lecturers, it must be recognised that these projects are undertaken during student instruction. There is therefore no guarantee that students will succeed in their efforts.

This inherently means that the client assumes a degree of risk. This is part of an arrangement, which is intended to be of mutual benefit. On completion of the project, it is hoped that the client will receive a professionally documented and soundly constructed working software application, some part thereof, or other appropriate set of IT artefacts, while the students are exposed to live external environments and problems, in a realistic project and customer context.

In consequence of the above, the students, acting in their assigned professional capacities and the Auckland University of Technology, disclaim responsibility and offer no warranty in respect of the “technology solution” or services delivered, (e.g. a “software application” and its associated documentation), both in relation to their use and results from their use.

References

Adroit. (n.d.). About Adroit. <https://adroit.nz/about-us/>

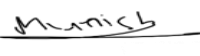
Suresh, M. (2013). Project management methodologies: A comparative analysis. *International Journal of Business and Management*, 8(10), 53-61. <https://doi.org/10.5539/ijbm.v8n10p53>

Hanssen, G. K., Stålhane, T., & Myklebust, T. (2018). SafeScrum® – Agile Development of Safety-Critical Software. *Springer Cham*. <https://doi.org/10.1007/978-3-319-99334-8>

Signatures

Miguel Emmara: 

Otis Wales: 

Munish Kumar: 

Glenn Neil: 

18 August 2023