Jadcup Digital Factory Automation

Project Proposal

Harshil Patel

Jane Jung

Joshua Ladowsky

Myles Hosken

Yeran Edmonds

Team Mentor: Matthew Kuo

Client: Daniel Li

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Author | Changes |
| 1 | 15/3/23 | Jane Jung | Created initial proposal structure |
|  | 20/3/23 | Jane Jung | Added to section 4,5,6, started section 8 and started adding completed plans to appendix |
|  | 22/3/23 | Jane Jung | Added Contents page |
|  | 23/03/23 | Josh | Added cost estimations |
|  | 23/03/23 | Myles Hosken | Terms of reference Project rationale |
|  | 23/03/23 | Yeran Edmonds | Added section 3 |

# Contents

Cover Page

Executive Summary

Terms of Reference

Project Rationale

Project Objective and Scope

Skills Analysis

Team Roles and Responsibilities

Team Schedule for Part 1

Project Management Methodology

Risk/Issue Management

Project Plan

Estimated Cost

Disclaimer

Appendix

* Skills Matrix
* Training Plan
* Team Contract
* Meeting Minutes
* Risk Register
* Issue Log
* Quality Assurance Plan
* Change Management Plan
* Change Request Form Template

# Executive Summary

Include a 1-page Executive Summary summarising the proposal and covering scope, time,   
method, risks and cost.

Scope

Time:

Method:

Risks:

Reduced time; team formed late.

Timetable issues

Upskilling limits

Communication issues w client

Can’t meet client often.

Cost:

# Section 1: Terms of Reference

Jadcup is a major manufacturer of sustainable and compostable disposable cups in New Zealand.   
Daniel Li, the founder of Jadcup, has identified that the current manufacturing process has room for improvement. While the process is automated, the monitoring and adjusting of the process can only be done using a built-in monitor connected to the machines. This limits the ability to monitor and optimize the process in real time, which can lead to inefficiencies, production downtime, and potential quality issues. This production process is known as Industry 3.0. In Industry 3.0, the process is automated using Programmable Logic Controllers (PLCs) and information technology. This process can often operate largely without human interference, but a human aspect still remains. To address the gap for improvement, Daniel has reached out to us:

* Joshua Ladowsky (Service request manager/developer)
* Myles Hosken (Service delivery manager/developer)
* Jane Jung (Developer)
* Yeran Edmonds (Developer)
* Harshil Patel (Developer)

with the guidance and mentorship of:

* Matthew Kuo(Mentor) and
* Tony Clear(Moderator)
* Professor Jaqcui Whalley (Course leader)
* Dr. Ramesh Lal (Course leader)

to develop and implement a remote monitoring system for the manufacturing process.

The proposed system will allow Jadcup to monitor the manufacturing process in real-time, change parameters, adjust, and detect any potential issues before they become major problems. The implementation of the remote monitoring system will enable Jadcup to increase their standing to Industry 4.0, also known as the fourth industrial revolution. Schwab, K (2016).Under industry 4.0 the process can be automated based on data and not by human input. For Jadcup to advance to Industry 4.0 a few of the following technologies will be implemented in conjunction with the remote monitoring system.

* **Internet of things (IoT) :**The Internet of things has been recognized as a key component in smart factories. Each machine can be installed with a low-cost sensor that houses an IP address. This address allows the machines to connect to internet-enabled devices. This connectivity allows large amounts of data and analytics to be collected and analysed. (IBM 2023).
* **Cloud Computing :**Cloud computing is a key component for an Industry 4.0 strategy. Cloud computing makes it possible to store large amounts of data to be processed more efficiently and cost-effectively. In order to unlock the full capabilities that smart manufacturing has to offer, it is necessary to connect aspects of the business such as, engineering, supply chain, production, sales and distribution. The use of cloud computing can facilitate this integration.
* **Edge Computing :**Edge computing is a term to describe where data is created. In this case Jadcup can implement edge computing by tracking their machine analytics right at the source, at the “edge”. Ultimately this minimizes latency and how fast the data is produced to when a response may be needed.
* **Digital Twin :**

In this project we will implement a digital twin of Jadcups manufacturing process. A digital twin is created by simulating a factory floor and pulling data from sensors, devices, and the current PLCs. Through the use of a 3D modelling software such as Factory I/O, we can simulate the production process to create workflows and test changes to improve capacity and minimize downtime.

**Schwab, K (2016).** [**https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/**](https://www.weforum.org/agenda/2016/01/the-fourth-industrial-revolution-what-it-means-and-how-to-respond/)

**(IBM 2023) https://www.ibm.com/topics/industry-4-0**

# Section 2: Project Rationale

By investing in this project, Daniel Li and Jadcup are demonstrating their commitment to continuously improving their product quality, enhancing customer satisfaction, and maintaining their position as a leader in sustainable and compostable disposable cups in New Zealand. The transition from Industry 3.0 to Industry 4.0 is a significant advancement for Jadcup and will provide a several benefits for the business:

* **Efficiency and productivity:**

Data-driven automation through the Internet of Things will allow Jadcup to streamline its operations, automate repetitive tasks, and fully optimize its processes.

* **Improved Quality:**

Industry 4.0 technologies can detect and resolve quality issues in real-time, resulting in higher production quality by detecting issues before they become a problem.

* **Cost saving:**

Automating processes under Industry 4.0 can allow Jadcup to reduce labour costs, minimize production downtime, and reduce waste all leading to cost savings. With data driving the production process, there will be less need for workers to be in the factory.

* **Enhanced Flexibility:**

Having access to Industry 4.0 technologies will ensure that Jadcup can respond quickly to changing market demands and customer needs by having the capability to change input and output production streams along with reconfiguration and customization faster.

* **Increased Safety:**

A higher stand of automation can help reduce the risk of workplace accidents ensuring the safety of employees.

* **Enhanced Data Collection:**

Jadcup under Industry 4.0 will have access to real-time data on their operations, giving them control and information to analyse performance and make informed decisions to improve their process.

* **Scalability:**

Once the remote monitoring system works for one machine, with the use of the Digital Twin and Industry 4.0, when the time comes for Jadcup to expand, the system will be modular and decoupled. Allowing Jadcup to expand effectively without many blockers.

* **Custom Manufacturing:**

A big part of the transition from Industry 3.0 to Industry 4.0 is the removal of “minimum batch requirements”. Industry 3.0 is all about mass production, utilizing the Digital Twin, Jadcup can easily create small batches of specialized orders for customers. Reducing waste and meeting customer needs in a modern way.

# Section 3: Project Objective and Scope

**Project Objective & Purpose**  
The objective of this project is to develop a prototype for a web-based FERN (Firebase, Express.js, React.js, Node.js) application that allows Jadcup to connect and interact with programmable logic controllers (PLCs).

The purpose of this prototype is to determine the viability of a web interface to manage PLCs on the Jadcup factory floor.

**Project Scope**

**Functional Requirements**

* The application must allow users to connect & control PLCs.
* The application must show data managed by PLCs.
* The application must be modular enough to support multiple, different PLCs.
* The application must provide secure user authentication and authorization.

Non-functional requirements

* The application must be responsive and have a fast load time.
* The application must be functional and user-friendly when run on a PC.
* The application UI must be designed with tablets and phones in mind for the future.

*Further reading: See 3.1 Scope Statement document in the appendix.*

**Key stakeholders**

* The primary stakeholder for this project is Jadcup who will use the FERN application to connect and interact with PLCs. eg stakeholders
* The project team, team mentor, team mediator and paper leads are also stakeholders in this project.

*Further reading: See 3.2 Stakeholder Management Strategy in the appendix.*

**Key risks and mitigation strategies**

* Technical upskilling redundancy: To mitigate the risk of team members unable to participate or unable to progress, each skill required for the prototype is being upskilled by two developers.
* Managed time efficiency: To mitigate time scheduling constraints, the team have concise meeting agendas and action items with set deadlines. Communication is also strong within the team in the case that an item cannot be completed.
* Minimum-viable product: To mitigate risks with unclear feature requirements & ‘scope creep’, the team have collaborated with the client and the mentor to identify minimum specifications to deliver an accurate prototype.

*Further reading: See 8.4 Change Management Plan, 8.2 Issue Log and 8.1 Risk Register in the appendix.*

**Infrastructure and human resource requirements**

The baseline software infrastructure for the server prototype will be a Local **FERN** server (**Firebase, ExpressJS, ReactJS, NodeJS**). As the prototype will be local and only needs to meet specific requirements for the MVP, we can leverage the team’s current knowledge of Javascript and the extensive global resources for React & Node to rapidly upskill and implement required features. Additionally, Firebase has support for authentication & authorization, which is a requirement of the MVP.

Additional tool upskilling required:

* **Git**: Industry standard version control and code management tool for collaboration. We will use Git in combination with Fork, a Github extension for visualizing branches and merging, to keep track of changes and version history.
* **FactoryIO**: Simulation software to replicate the factory environment. It will allow us to create a prototype factory and begin reading data from the simulated PLC.
* **NodeRED**: Open-source flow-based programming tool built on top of Node.js. Used to create a workflow for the PLC and facilitate direct communication with the dashboard within the same tech stack.

*Further reading: See 3.3 Identified Technical Architecture in the Appendix.*

Software technology considered:

* **LAMP** stack (Linux, Apache, MySQL, PHP)

Prototyping on a Linux server with Apache & PHP would allow us to better manage web traffic to the server. However, our prototype doesn’t have a requirement to be online so the LAMP stack would require more extensive upskilling. In addition, we quickly found NodeRED for interfacing with PLCs and, as a team, preferred to upskill in modern technologies.

* **Other NoSQL databases**

While MongoDB and DynamoDB were also considered, we as a team had more knowledge with firebase to better facilitate rapid iteration. Additionally, firebase has support for authentication & authorization. This comes at a known cost to data access, which is included in our budget.

Hardware technology considered:

* **Local Physical Server**

The prototype is being designed with a server in mind, however we deemed it out-of-budget for the prototype to be hosted on a local server. We expect to strictly use FactoryIO and the AUT-sourced PLC for testing the local server.

**Limitations & Constraints:**

It is currently beyond-scope for our MVP to directly interface with Jadcup’s PLCs. While this will allow for rapid iteration, it is possible that the prototype will need post-project support to work with Jadcup devices.

Technology provided by AUT:

* FactoryIO service keys
* Schneider Modicon M221

# Section 4: Skills Analysis

The development of the project deliverables requires the combination of several components, each of which involves knowledge and skills from different fields of software development.

The table below identifies the skills required for the project, as well as the skills missing that members of the team need to upskill before the development stage of the project can begin. Individual skill levels can be found in the appendix.

|  |  |
| --- | --- |
| Required Skills | Missing Skills |
| * Application Server   + Online Servers   + Server Administration * Component Technology   + PLC   + Factory I/O * Database   + NoSQL * Language/Development tools   + C#   + Git   + HTML/CSS   + Java   + NodeRed * Multimedia   + UI Design   + UX Design * Project Management   + Agile Methodology   + Client Management * Project Management tools   + Github   + Teams   + Trello * QA Expertise   + QA Tools   + Unit Testing * Web Technology   + Javascript   + Node.Js   + PHP   + React   + Visual Studio   + VS Code   + Web APIs | * JavaScript/React * PLC * Factory I/O * Web API (Factory I/O) * NodeRed/SCADA * Databases |

The upskilling phase for missing skills will commence immediately after the completion of the proposal (31st of March) and will be no longer than 2 weeks (see appendix for upskilling schedule). During this time, everyone in the team will be researching and familiarizing themselves with the relevant skills and tools. (See more details in Appendix- Training Plan). Once the upskilling phase is finished, team members are expected to have gained the relevant skills and abilities which can be applied to efficiently develop high-quality outcomes for the project.

# Section 5: Team Roles and Responsibilities

**Service Request Manager/ Developer:** Josh Ladowsky

* Managing the team
* Organizing team meetings
* Managing team schedules
* Delegating tasks between team members
* Communicating with mentor
* Resolving conflicts
* Delivering project requirements
* Completing delegated tasks
* Attending and actively contributing to meetings
* Communicating any issues/problems
* Documenting project processes

**Service Delivery Manager/ Developer:** Myles Hosken

* Representative of the team when communicating with client (e.g., email)
* Organizing client meetings
* Delivering team opinions to client during meetings or via online communication
* Answering client’s questions
* Delivering project requirements
* Completing delegated tasks
* Attending and actively contributing to meetings
* Communicating any issues/problems
* Documenting project processes

**Developer:** Harshil Patel

* Delivering project requirements
* Completing delegated tasks
* Attending and actively contributing to meetings
* Communicating any issues/problems
* Documenting project processes

**Developer:** Jane Jung

* Delivering project requirements
* Completing delegated tasks
* Attending and actively contributing to meetings
* Communicating any issues/problems
* Documenting project processes

**Developer:** Yeran Edmonds

* Delivering project requirements
* Completing delegated tasks
* Attending and actively contributing to meetings
* Communicating any issues/problems
* Documenting project processes

Every individual in the team is responsible ensuring that they follow the rules of the team (see appendix- Team contract) to maintain a healthy working environment for the duration of this project.

# Section 6: Team Schedule for Part 1

**Semester 1 availability**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Mon | Tue | Wed | Thu | Fri |  |  |
| 6:00 |  |  |  |  |  |  | Harshil unavailable |
| 7:00 |  |  |  |  |  |  |  |
| 8:00 |  |  |  |  |  |  | Jane unavailable |
| 9:00 |  |  |  |  |  |  |  |
| 10:00 |  |  |  |  |  |  | Joshua unavailable |
| 11:00 |  |  |  |  |  |  |  |
| 12:00 |  |  |  |  |  |  | Myles unavailable |
| 13:00 |  |  |  |  |  |  |  |
| 14:00 |  |  |  |  |  |  | Yeran unavailable |
| 15:00 |  |  |  |  |  |  |  |
| 16:00 |  |  |  |  |  |  | offsite |
| 17:00 |  |  |  |  |  |  |  |
| 18:00 |  |  |  |  |  |  | workshop |
| 19:00 |  |  |  |  |  |  |  |
| 20:00 |  |  |  |  |  |  |  |
| 21:00 |  |  |  |  |  |  |  |

**Meeting schedule**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Frequency** | **Date/Time** | **Location** |
| Offsite hours | Weekly | Tuesday 8am-2pm |  |
| Team meeting | Weekly | Tuesday 12pm | WZ701 |
| Makeup meeting |  | Saturday 9am | Online |
| Mentor meeting | Weekly | Tuesday 10am | WZ1101 |
| Client meeting | One-off | 16th March 1pm | Jadcup headquarters |

Due to the conflicting schedules between team members, we decided that we would hold the team meetings during offsite hours, and when required we would have a makeup meeting in the weekends.

**Meetings in semester 1**

|  |  |  |  |
| --- | --- | --- | --- |
| **Meeting type** | **Date** | **Time** | **Location** |
| Team meeting | Monday 6th March | 6pm | WZ701 |
| Mentor meeting | Tuesday 7th March | 11am | WZ1101 |
| Mentor meeting | Tuesday 14th March | 10am | WZ1101 |
| Team meeting | Tuesday 14th March | 11am | WZ601 |
| Client meeting | Thursday 16th March | 1pm | Jadcup headquarters |
| Team meeting | Saturday 18th March | 9am | Online |
| Team meeting | Tuesday 21st March | 9.30am | WZ701 |
| Mentor meeting | Tuesday 21st March | 10am | WZ1101 |
| Team meeting | Tuesday 21st March | 11am | WZ1101 |

[See appendix for further details of the meetings]/ portfolio

# Section 7: Project Management Methodology

**Summary**: As a team, we have decided to use Kanban as our project management methodology. With the wide range of Kanban tools available including Trello and the Agile Manifesto, we can leverage our team’s talents and experience with agile methodologies.

**Rationale**:

* **Flexible Development**  
  Our team and our client are continuing to discover the design intention of the Digital Twin prototype as we develop. While we have functional requirements, we’re still unclear on technical design and lack team knowledge both in design and technology stack. Because of this, we need rapid iteration through numerous meetings and highly visible iterables. For this reason, Kanban’s focus on user stories and granular work will be invaluable for development.
* **Industry Standard**  
  In addition to the practical applications of Kanban as a methodology, we also chose this agile methodology as a supplement to our own learning. Agile practices are becoming common in the wider workforce, and we as a team chose Kanban so that we can better fit employer requirements. We have also had strong experience with a past paper and lecturer that took the lead on teaching us strong agile practices.
* **Strong Agile Standards**  
  With only a limited amount of development time available (15 hours per week) we need to make sure our ways of working are strong and rapidly improved. Kanban’s agile working practices can be supplemented with retrospectives, reviews and to weekly planning allow us to reflect and iterate on the way we plan our development. This allows our development to become more lean and streamlined as time goes on.

**Justification**:

* **Waterfall**  
  While waterfall would be invaluable for designing the system from the start, we are in a position of unknown requirements, iterations and skills. Because of this, we don’t have the luxury of dedicating specific engineers to tasks for long lengths of time and create the perfect prototype. We are discovering as we go, and waterfall’s rigid, process-by-process structure would be unsuitable for our workflow and the project.
* **Scrum**  
  It should be noted that we are using Scrum’s framework for various meetings and ways of working to keep everyone informed and organized. However, scrum has a product owner and scrum master for keeping things organized and on-track. We do not have the luxury of a product owner for providing feature requests nor prioritizing our work, and we need to do that ourselves. Additionally, Scrum doesn’t provide a clear methodology for iterating on work like Kanban does in the form of visual tasks that proceed along development steps.

Check appendix for WBS

# Section 8: Risk/Issue Management

Potential project risks (see appendix- risk register) that could pose a threat during the project development were identified and evaluated, as well as the potential responses that could be made by the team to prevent the risks from happening or reduce the damage as much as possible.

The threat levels of the risks were based on the probability of the risk occurring, as well as the impact of the risk if it were to occur. The team members assigned on risk documentation will be regularly updating the risk register during team stand-up meetings.

The issue log shows records of unexpected problems the team has faced so far during the project, and the potential solutions to mitigate the issues, as well as the measures taken to prevent reoccurrence of similar issues in the future. The issues are prioritized between low/medium/high threat levels, which corresponds to how soon we attempt to find a solution.

The change management plan is in place to establish a structured and systematic approach to manage changes in the project. If a change is needed, the client and relevant team members will need to collaborate and produce a change request form. The change request form will be specific to the change.

For Details about Change Management plan see Appendix 8.4: Change Management Plan

For Details about Change Request Form see Appendix 8.5: Change Request Form

Using our Quality Assurance Plan we will be able to minimise our outlined risks and produce a project of higher quality that will have a reduced risk of damaging the client’s equipment. This is done by ensuring any all code developed is tested.

To ensure the quality of our tests we will utilise a variety of metrics measuring our overall testing and bug distribution. As well as designing systems for reporting false QA test outcomes and non-compliance of test outcomes.

Should non-compliance or false QA testing be discovered then the Audit and review procedures are outlined to ensure any testing done is to a professional standard.

For Details about procedures see Appendix 8.3: Quality Assurance Plans

# Section 9: Project Schedule

[Milestone Report]

The project management framework we’re incorporating into our project management is Agile, which breaks down a large project into smaller and more manageable tasks and provides flexibility to develop multiple requirements at once in repeated updating cycles. Out of the Agile frameworks, we chose to go with the Kanban approach which focuses on continuous development on a set of independent tasks without a set hierarchy of order of the tasks. Kanban also makes scheduling more flexible, as there are no fixed time limits for a task to be completed.

Our project schedule is split into the following:

**Phase 1- Planning**

During the planning phase, we lay out the foundations for our project development and start gathering information on the project’s requirements, the tech stack and the methods we will use to develop a healthy and productive working strategy. This involves:

* Meeting and becoming acquainted with team members through initial team meeting.
* Initial team meeting with mentor to familiarize ourselves with the project and client background.
* Discovering and researching requirements for project.
* Meeting with client to establish details of project, identify project scope and gain insight on factory layout.
* Breaking down project into tasks and establishing user stories.
* Complete and present project proposal.

**Phase 2- Upskilling**

Due to the wide range of skills required for the project, we require an official time period before we develop the outcome to ensure that we have the skills and knowledge necessary to efficiently provide the stakeholders with a high-quality outcome. During this phase, we will be following the upskilling schedule and will be individually upskilling on different fields depending on the tasks we are responsible for, as well as fields that require the proficiency for all team members (e.g., GitHub, Quality assurance strategies).

*See appendix for Upskilling schedule.*

**Phase 3- Development/Monitoring**

During this phase, we repeat cycles where we implement and release small versions of the user stories assigned in the planning stage, which is then tested/reviewed then released again with relevant updates. Because there are multiple components required for the delivery of this project, the team decided that developing and monitoring tasks would be more efficient than pushing out all requirements at once then monitoring a large batch of components.

The default cycle length is 14 days but can be adjusted dynamically throughout the project based on the progression rate of development.

**Phase 4- Closing**

This phase is used to prepare for the completion and handover of the project deliverables. During this time the team will be completing the final release of the project deliverables and the handover to client and start reviewing the project documentation to prepare for the review of the project.

**Phase 5- Project Review**

The final phase of the project outlines the assessments for the team project post-handover, and marks estimated deadlines for all requirements of the project. During this time the team will be finalizing the project portfolio, as well as the individual reflective reports.

*For a detailed timeline of the project, please refer to the chart in the appendix- “Project schedule”.*

# Section 10: Cost

The following outlines the estimated costs of the project from now till the end of the project due to further research is required to find pricing of Database and Dashboard hosting as many services such as firebase and google cloud use systems to pay as you use so precise estimations can only be made when the system is in further development.

Furthermore, currently it is expected that Jadcup’s existing hardware is adequate for the systems to be developed and no additional modifications or modules will need to be added into the environment.

Senior Management costs are based on AUT’s estimated hourly pay for one academic hour, while the Development Team’s wage is based on the lower end of a junior developer’s hourly wage. Trainee cost is assumed from the average warehouse worker wage in NZ.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| WBS items | # Units/Hrs | Cost/Unit/Hr | Subtotals | WBS Level 1 Totals $ of Total | % of Total |
| 1. Project Management |  |  |  | $9,699.00 | 73.25% |
| 1.1 Senior Management | 30 | $163.30 | $4,899.00 |  |  |
| 1.2 Development Team | 150 | $32.00 | $4,800.00 |  |  |
| 3. Software |  |  |  | $1,052.28 | 7.95% |
| 3.1 Factory I/O | 7,333.33 | $0.08 | $550.00 |  |  |
| 3.2 DB/ Dashboard hosting (Estimated cost) | 3,666.67 | $0.14 | $502.28 |  |  |
| 4. Testing (10% of software) |  |  |  | $105.23 | 0.79% |
| 5. Training |  |  |  | $178.00 | 1.34% |
| 5.1 Trainee cost | 2 | $25.00 | $50.00 |  |  |
| 5.3 Project team members | 4 | $32.00 | $128.00 |  |  |
| 6. Reserves (20% of total estimates) |  |  |  | $2,206.90 | 16.67% |
| Total project cost estimate |  |  |  | $13,241.41 |  |

# Section 11: Disclaimer

Attach a disclaimer as appendix, clarifying the nature of the relationship involved (see   
the Standard Disclaimer on Canvas).

# Appendix

## 3.1 Scope Statement

**Prepared by:** Yeran Edmonds

|  |
| --- |
| **Project Title: Jadcup Digital Twin**  **Date: 08/03/2023Prepared by:** Jadcup team |
| **Project Justification:**  Jadcup wants to modernize their factory processes and pull their production into Industry 4.    This project is a prototype to determine the viability of an interface to manage PLCs on the Jadcup factory floor. |
| **Purpose and Objectives:**  The purpose of this project is to develop a web-based FERN (Firebase, Express.js, React.js, Node.js) application that allows Jadcup to connect and interact with programmable logic controllers (PLCs).  The objective of this project is to provide Jadcup with a prototype for a user-friendly interface to connect and control PLCs remotely. |
| **Product Characteristics and Requirements:**   1. Web interface 2. Factory I/O Simulation (strictly for the purposes of this prototype) 3. Read/Modify PLC Data |
| **Functional Requirements:**   * The application must allow users to connect & control PLCs. * The application must show data managed by PLCs. * The application must be modular enough to support multiple, different PLCs. * The application must provide secure user authentication and authorization.   **Non-Functional Requirements:**   * The application must be responsive and have a fast load time. * The application must be functional and user-friendly when run on a PC. |
| **Summary of Project Deliverables**   1. A functioning FERN web application prototype for interfacing with PLCs 2. A design document for the application 3. A presentation for the web application |
| **Assumptions & Constraints:**   * The application will be developed using Kanban methodology. * The development team will consist of five developers, one of which is a project manager, and one of which is a scrum master. * The team will be managed & report to a Team Mentor and a Client * The application will be hosted on a cloud-based platform. * The development will be completed by the end of October, 2023. * The application will be a prototype. |
| **Stakeholders:**   * The primary stakeholder for this project is Jadcup who will use the FERN application to connect and interact with PLCs. * The project team, team mentor, team mediator and paper leads are also stakeholders in this project. |
| **Project Success Criteria:**   * The application must be delivered by the end of October, 2023. * The application must be fully functional and meet all the functional and non-functional requirements. * The application must have a high level of user satisfaction, as measured by feedback received from the client. |

## 3.2 Stakeholder Management Strategy Register

**Prepared by:** Yeran Edmonds

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Name** | **Position** | **Project Role** | **Level of Interest** | **Level of Influence** | **Potential Management Strategies** |
| Matthew Kuo | AUT Lecturer | Team Mentor | Medium | Medium | Keep informed, do his clear requirements. He likes seeing work done and his team keeping ahead and on-schedule. Has a preference for strong technical knowledge. |
| Yeran Edmonds | AUT Student | Team Member | High | High | Prefers clear tasks. Can be bribed with snacks. |
| Harshil Patel | AUT Student | Team Member | High | High | Keep informed, get opinion on development. Prefers to be directly given tasks. |
| Jane Jung | AUT Student | Team Member | High | High | Keep informed. Likes to keep on top of work and stay ahead. |
| Myles Hosken | AUT Student | Team Member | High | High | Prefers to get work done asap to not work weekends. Likes to stay ahead as much as possible. |
| Joshua Ladowsky | AUT Student | Team Member | High | High | Prioritizes Factory IO development and managing the PLCs. Strong technical background. |
| Tony Clear | AUT Lecturer | Team Mediator | Medium | Medium | Likes Agile practices and clear, readable documentation. |
| Ramesh | AUT Lecturer | Paper Leader | Low | Medium | Very low communication/influence but prefers Agile and nicely presented information (coloured documents). |
| Jacqui | AUT Lecturer | Paper Leader | Low | Medium | Very low communication/influence. |
| Client | Jadcup Project Manager | Team Client | Medium | High | Very busy. Prefers to keep informed via email |

## 3.3 Identified Technical Architecture

**Prepared by:** Myles Hosken

|  |  |
| --- | --- |
| **Technology** | **Purpose** |
| Git | Industry standard version control and code management tool for collaboration. We will use Git in combination with Fork, a Github extension for visualizing branches and merging, to keep track of changes and version history. |
| JavaScript / React | Dynamic and responsive user interface capability for the web dashboard that can be customized according to the client's needs. Third-party libraries can facilitate communication between the factory's Programmable Logic Controllers (PLCs) and the dashboard. |
| Programmable Logic Controllers (PLCs) | Industry standard for machine programming and automation, used to control the factory machinery. |
| Factoryio | Simulation software to replicate the factory environment. It will allow us to create a prototype factory and begin reading data from the simulated PLC. |
| Web API’s | Application Programming Interface (API) for real-time communication between the factory and dashboard. |
| NodeRED | Open-source flow-based programming tool built on top of Node.js. Used to create a workflow for the PLC and facilitate direct communication with the dashboard within the same tech stack. |
| Online Database | Online database to store information securely and reliably, such as client login credentials. Firebase has been considered as it offers 1GB of data transfer per month, which should be sufficient for the MVP and the client's needs. |
| Online Server | Hosts the web dashboard. Selection of hosting provider, operating system, and necessary software is required. Reliable free services may not be available, and the cost of the server may need to be factored into the project budget. |
| IDE | An Integrated Development Environment (IDE) for JavaScript development, such as Visual Studio Code or WebStorm, would be suitable for developing the web dashboard using JavaScript/React. Additionally, an IDE or development environment for programming the PLCs may be required depending on the specific hardware and software used. |

## 4.1 Skills Matrix

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Group | Skill | Harshil ISTP-A | | | | | Jane-ESTP-A | | | | | Josh - INTP-T | | | | | Myles - ISFJ-T | | | | | Yeran - ISFJ-A | | | | | Total |
| 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 5 |
| Application Server | Online Servers |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |  | 6 |
| Server Administration | 1 |  |  |  |  |  | 2 |  |  |  | 1 |  |  |  |  | 1 |  |  |  |  |  |  | 3 |  |  | 8 |
| Component Technology | PLC |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  | 2 |
| Factory I/O | 1 |  |  |  |  | 1 |  |  |  |  |  |  | 3 |  |  | 1 |  |  |  |  | 1 |  |  |  |  | 7 |
| Database | NoSQL |  | 2 |  |  |  |  | 2 |  |  |  | 1 |  |  |  |  |  | 2 |  |  |  |  |  |  | 4 |  | 11 |
| SQL |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |  | 6 |
| Online Database |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |  | 6 |
| Language/Development Tools | C# |  |  |  | 4 |  |  |  |  |  | 5 |  |  |  |  | 5 |  |  | 3 |  |  |  |  | 3 |  |  | 20 |
| Git |  |  | 3 |  |  |  |  |  | 4 |  |  |  |  | 4 |  |  |  | 3 |  |  |  |  | 3 |  |  | 17 |
| HTML/CSS |  |  | 3 |  |  |  |  |  |  | 5 |  |  | 3 |  |  |  |  | 3 |  |  |  | 2 |  |  |  | 16 |
| IDEs |  |  |  |  |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 4 |  | 8 |
| Java |  |  | 3 |  |  |  |  |  |  | 5 |  | 2 |  |  |  |  |  |  | 4 |  |  |  | 3 |  |  | 17 |
| NodeRed/SCADA |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  | 2 |
| Visual Studio |  |  |  |  |  |  |  |  |  | 5 |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |  | 8 |
| Vs Code |  |  |  |  |  |  |  |  |  | 5 |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |  | 8 |
| Multimedia | Adobe Illustrator |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  | 4 |
| UI Design |  |  |  |  |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  | 5 |
| UX Design |  |  |  |  |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  | 1 |  |  |  |  | 5 |
| Project Management | Agile Methodology Knowledge |  |  | 3 |  |  |  |  | 3 |  |  |  |  | 3 |  |  |  |  | 3 |  |  |  |  | 3 |  |  | 15 |
| Client Management |  | 2 |  |  |  |  |  | 3 |  |  |  | 2 |  |  |  |  |  |  | 4 |  |  |  | 3 |  |  | 14 |
| Project Management Tools | Github |  |  |  |  |  |  |  |  | 4 |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 4 |  | 8 |
| Teams |  |  |  |  |  |  |  |  |  | 5 |  |  |  |  |  |  |  |  |  |  |  | 2 |  |  |  | 7 |
| Trello |  |  |  |  |  |  |  |  |  | 5 |  |  |  |  |  |  |  |  |  |  |  |  | 3 |  |  | 8 |
| QA Expertise | QA Tools |  |  |  |  |  | 1 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 2 |  |  |  | 3 |
| Web Technology | JavaScript |  |  | 3 |  |  |  |  |  | 4 |  |  | 2 |  |  |  |  |  | 3 |  |  |  |  |  | 4 |  | 16 |
| Node.Js |  | 2 |  |  |  | 1 |  |  |  |  |  | 2 |  |  |  |  | 2 |  |  |  |  |  | 3 |  |  | 10 |
| PHP | 1 |  |  |  |  |  | 2 |  |  |  |  | 2 |  |  |  |  | 2 |  |  |  | 1 |  |  |  |  | 8 |
| React | 1 |  |  |  |  |  | 2 |  |  |  | 1 |  |  |  |  |  |  | 3 |  |  |  | 2 |  |  |  | 9 |
| Web APIs |  |  |  |  |  |  |  | 3 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | 4 |  | 7 |

## 4.2 Upskilling Schedule

**Prepared by:** Jane Jung

Table

Description automatically generated A picture containing waterfall chart

Description automatically generated

Note: Upskilling phase 1 is individual upskilling in different fields depending on responsible task, Upskilling phase 2 prioritizes skills lacking which are required by all team members.

## 4.3 Training Plan

The below is a training plan targeted towards industrial automation in conjunction with a web dashboard. The identified skills are generalized as some members skills are higher in some areas than others.

1. **Kanban Methodology**. Kanban is the chosen methodology for the project members will need to be familiar and up to date with:

* The principles of Kanban. Workflow, visualization, learning how to prioritize and limit work in progress (WIP).
* Understanding of the Kanban board.
* Introducing Kanban into the project.
* Utilizing Kanban alongside a visual board such as Jira or Trello.

1. **JavaScript / React**. As the main deliverable is a web dashboard that is responsible for reading and controlling Programmable Logic Controllers, we have identified that JavaScript / React will be a suitable language for this.

* Transferring existing programming knowledge into JavaScript. Basics and advanced programming techniques.
* If react is needed, then understanding react architecture, react components, events, built-in hooks, and using in-built libraries.
* Implementing a simple React application.

1. **Factory I/O.** As professionally suggested, factoryio has been decided as the 3D simulation tool to replicate a factory.

-      Familiarity with factoryio and what it is capable of.

-      Factory simulation and creation as best as can be replicated to the client’s factory.

-      Connecting factoryio to a Programmable Logic Controller (PLC).

-      Simulating a production line according to the client’s requirements.

1. **Programmable Logic Controllers (PLCs).** PLCs are a new area of upskilling that the team will need to be familiar and capable of working with.

* Being familiar with common industrial protocols such as Ethernet/IP, OPC UA and Modbus.
* Configuring PLCs to communicate and transfer data with other devices.
* Familiarity with common debugging and troubleshooting techniques and issues.
* Coding PLCs to read and write data using a protocol.

1. **Factory I/O Web APIs.** Connecting factory I/O with a web dashboard is going to require communication over an API to read and write data.

* Understanding how to connect factory I/O with a web socket and web API.
* Creating and sending HTTP requests which may read and write data to either Factory I/O and the dashboard.

1. **NodeRed and SCADA.** NodeRed and SCADA are common tools used in industrial settings. To simulate a factory alongside PLCs, knowledge for NodeRed and SCADA may need to be investigated. SCADA systems consist of hardware and soft-ware which analyze and collect data from sensors. This data is processed and sent to control devices.

* NodeRed and SCADA familiarity
* Understanding, creating and managing flows.
* Integration with SCADA and NodeRed.

[*https://www.allaboutcircuits.com/technical-articles/an-introduction-to-scada-systems/*](https://www.allaboutcircuits.com/technical-articles/an-introduction-to-scada-systems/)

1. **Databases.** A database may be needed to hold and store machine and user login information.

* Recap on Database syntax.
* Understanding database schemas.
* Connecting the database to the web dashboard and PLC/computer.
* Hosting a database either locally or cloud based.

This training plan encompasses a diverse range of topics that are pertinent to a research and development project in industrial automation. It offers a thorough comprehension of crucial areas including Kanban methodology, Javascript / React, PLC protocols, Factory I/O, Factory I/O Web API’s, NodeRED and SCADA, and databases. Upon finishing the training, participants will possess the requisite abilities and expertise to build a virtual factory, manipulate it using various software tools, and manage and archive data associated with the factory.

## Team Contract

1. **Commitments**

As a team we will:

* Agree to attempt work that we have the ability for.
* Be realistic when planning and making schedules.
* Highlight any potential problems and work to fix them before they happen.
* Keep other team members informed and communicate effectively.
* Keep information regarding the client confidential.
* Focus on the project.
* Proceed to see the project through to its completion.

1. **Team meetings**

As a team we will:

* Meet at the agreed scheduled time.
* Notify team members if you will be absent with adequate notice.
* Give every member the opportunity to contribute.
* Be open and listen to all ideas that people present.
* Not place blame on any single member of the team.
* Keep meeting conversation to the topic at hand, further clarification can be taken offline.

1. **Problem Solving**

As a team we will:

* Encourage all team member’s input.
* Use each other’s ideas and build off them.

1. **Conflicts**

As a team we will:

* View any conflicts as part of working in a team and an opportunity to grow as a team.
* Seek to understand each other’s opinions.
* Clarify with each other what we understand and communicate if it is correct.
* Acknowledge valid points that the other person has made.
* Points of view will be stated in a non-judgemental and non-attacking manner.
* Seek to find common ground.
* If a resolution cannot be found, third party support can step in.

1. **Meeting Guidelines**

* The allocated offsite hours on **Tuesday 8am-2pm** will stay and be reserved for the project meetings and work the team needs to complete.
* Team members will attend meetings prepared.
* Meeting discussions will stay on topic.
* A brief for the next week's meeting will be discussed or planned throughout the week before the meeting.

1. **Documentation**

* Use OneDrive to host final submission.
* Keep working documents on Teams for mentor visibility.
* Use Trello to organize work required and work results.

## 6.1 Meeting Minutes

*Sike refer to portfolio please.*

## 7.1 Work Breakdown Structure

## 8.1 Risk Register

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Risk Register for Jadcup Digital Factory Automation** | | | | | | | | | | |
| **Prepared by:** | | Yeran Edmonds & Harshil Patel | |  |  |  | **Date:** | 13-Mar |  |  |
| **No.** | **Threat** | **Risk** | **Description** | **Root Cause** | **Triggers** | **Potential Responses** | **Risk Owner** | **Probability** | **Impact** | **Status** |
| 1 | Medium | Unclear Requirements | The design and necessary features of the project may be and remain unclear throughout the project | The project is as new for the client as it is for the team |  | Keep open and clear communication with the client's requirements | Client Liaison | 40% | 60% | Watching |
| 2 | Low-Medium | Incorrect work performed | With uncertainty of outcomes or requirements, work may be wasted in unnecessary areas | Unclear, undocumented team direction |  | Clarify work with the team, mentor, and client regularly | Team Manager | 20% | 40% | Unresolved |
| 3 | Medium | Scope Creep | The work required for the project may gradually increase and overwhelm the team | Designing or discussing features that don't align with the prototype MVP |  | Check each feature aligns with the MVP | Whole Team | 30% | 50% | Unresolved |
|  | Low-Medium | Mismanaged feature prioritization | Features incorrectly prioritized may result in delays if one system requires on another, undelivered system (AKA, the web frontend relying on the backend & database) | Poorly managed and discussed features. Improper discussion on slow-to-implement systems |  | Weekly meetings to discuss prioritization and delays | Whole Team | 40% | 30% | Unresolved |
| 5 | Low-Medium | Poor product design | The tech stack or user design may be irrelevant to the prototype's end goals | Research and discussion not properly performed between the team and client |  | Be as thorough as the team can be with tech stack research and keep discussion open with the team. | Whole Team | 20% | 60% | Watching |
| 6 | High | Poor communication with the client | Poor communication with clients can lead to problem like dissatisfied client and unscalable | The client first language isn't English and has trouble speaking English |  | Is to check with the client properly and double check it if the client is happy with it | Whole Team | 55% | 90% | Unresolved |
| 7 | Medium | poor time-management | The standard and quality of the prototype could be impacted by improper time management for the entire project. | Not finishing task on schedule can lead to more time catching up |  | Finish your work and task on schedule. Stay focus. | Whole Team | 60% | 50% | Watching |
| 8 | Low-Medium | Workplace injury (Health and Safety) | Not knowing the health and safety hazards of the environment | The project in a factory environment is a first for most of the team |  | Researching and learning the basic health and safety rule of the workshop | Whole Team | 10% | 60% | Watching |
| 9 | High | Legal Issue | The clients need to tell the team their legal or regulation requirements, so that we don’t break the client requirements | By not knowing the client's legal or regulation requirements |  | Ask the client if there is any legal or regulation requirements | Team Manager | 5% | 90% | Watching |
| 10 | high | Poor communication with other members of the team | Not letting your team members know about problems or struggle that you are having. | Trying to solve issue/problem yourself, or struggling in finishing up your task |  | Letting the team know your issue or problems that you may been having. | Whole Team | 70% | 40% | Watching |
| 11 | high | team member unavailable to participate | Someone might have had an issue come up or some sort of personal reason that is preventing them from coming. | Overwhelmed with everyday problem/obstacle |  | Letting the team and mentor know about the situation. | Team Members | 30% | 60% | Watching |
| 12 | Medium-high | lockdown happening again | Another global pandemic issue according | Not caring about yourself or others during pandemic |  | Staying home when you are not feeling well | Whole Team | 5% | 75% | Watching |
| 13 | high | data loss | Not able to gather data due to it being lost | poor quality of coding, |  |  |  |  |  | Watching |
| 14 | high | requirements beyond scope |  |  |  |  |  |  |  | Watching |

## 8.2 Issue Log

**Prepared by: Jane Jung**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Issue Description** | **Impact on Project** | **Report date** | **Reported by** | **Assigned To** | **Priority**  **(M/H/L)** | **Resolve date** | **Status** | **Comments** |
| 1 | Late team formation | Reduced amount of time for the project by a week | 27/3/23 | Jane | Jane | High |  | Active | Will be working during weekends if necessary |
| 2 | Scheduling issue | Everyone’s timetables during the week clash which prevents the team from having meetings | 27/3/23 | Jane | Everyone | High | 13/3/23 | Closed | Everyone agreed to meet during the weekend (Saturday) |
| 3 | Communication with client | Client prefers Mandarin over English which none of us can speak | 28/3/23 | Matthew | Matthew | Medium | 16/3/23 | Closed | Matthew acted as the translator during the meeting, and the client agreed to give us his email if we needed further clarifications |
| 4 | Behind schedule | Lack of information of the project delayed the start of the proposal | 14/3/23 | Matthew | Jane | High | 15/3/23 | Active | Wrote out the proposal’s template, will be updating regularly with completed plans |
| 5 | Lots of requirements | The project requires lots of different components that all require different skillsets | 16/3/23 | Jane | Everyone | High |  | Active | Will designate tasks between members based on their skills, as we may start upskilling before submitting proposal.  Will designate tasks between members based on their skills, as we will start upskilling before submitting proposal |

## 8.3 Quality Assurance Plan

**Prepared by:** Josh Ladowsky

1. **Draft Quality Assurance Plan**
   1. **Introduction**

While developing a dashboard for integration in industrial settings, it is vital that we ensure that there are no issues that could cause damage to the product, production, or equipment involved. To do this we must ensure that our software is tested timely and regularly to find any compatibility errors.

* 1. **Purpose**

The following Plan will outline the procedure we will undertake to assure our product is both stable and meets the requirements of our client as well as outline team members roles for the purpose of Quality Assurance.

* 1. **Policy Statement**

We will have the following systems in place to ensure our products are of a quality that will is of a professional standard:

* Regular meetings regarding feedback.
* Stakeholder feedback gathering.
* Internal testing of any feature pushed.
* Regular tests of software
  1. **Scope**

This plan will focus quality assurance for the development of both backend and frontend software created for the purpose of developing a dashboard for Jadcup’s cup production procedure.

1. **Management**
   1. **Organisational Structure**

Quality Assurance will be the development team's responsibility, with a developer requesting a code test by a member that did not assist in the writing of the code.

* 1. **Roles and Responsibilities**
     1. **Technical Monitor/Senior Management**

The Technical Monitor for this project is Matthew Kuo. He may evaluate finalized features but will not be involved in feature testing at lower levels.

* + 1. **Task Leader**

For any given task a Task Lead will be assigned, this person is responsible for completing the code for the task and submitting it to be tested.

* + 1. **Quality Assurance Team**

Quality Assurance will be done internally by the development team as we do not have the staffing to hire an independent team.

* + 1. **Technical Staff**

As With the QA Team technical weaknesses in code will be assessed by members of the Development team.

1. **Required Documentation** 
   1. **Quality Assurance Procedures**

Upon the completion of code, a Developer will post a Git push request to be reviewed by a member of the team that was not involved in the code will evaluate the code and should any errors be found then the push will be denied. Should there be no errors found the push request will pass. In both situations the Developer will be alerted to the status of the push request along with any additional information that is required

* 1. **Walkthrough Procedure**

Code Walkthroughs will be done with the whole development team with the possibility of additional stakeholders joining the procedure. During this walkthrough, the goals will be to identify issues and ensure the feature meets any goals outlined.

* 1. **Review Process**

QA reviews can be held if members of the team find that the quality of accepted code is not up to the expected standards outlined above.

**Review Procedures**

In the code of a QA Review, a third-party member of the developer team will investigate code that has been pushed along with any additional notes and discussions that took place during the process.

If the reviewer does not find that the code should have been accepted, then further review of the members' previous QA contributions will be in order in the form of an audit.

* 1. **Audit Process**

If a member of the QA team is found to have been producing poor reviews of code; accepting code that is faulty or denying code that is acceptable. Then an audit of previous QA procedures will have to occur.

**Audit Procedures**

During a QA audit the review of previously accepted push requests will occur with the other members of the QA

* 1. **Evaluation Process**

Evaluation of the product’s overall health will occur regularly to assess the quality and ensure it will not damage the client’s existing systems. Should any issues be discovered it could result in a review or audit.

* 1. **Process Improvement**

Should any process outlined above is unnecessary, after discussion and review by the QA team members along with Senior Management will review the process and decide if it needs to be altered.

1. **Problem Reporting Procedures**

In the event of an issue found during the testing of code, be that a bug or the code is not achieving any goals outlined then the developer should be alerted, and the push request must be denied with the reason for denial in the push response.

**Noncompliance Reporting Procedures**

Should a reported problem not be properly addressed after the code is submitted for testing when it has been denied once already, the code should be denied again, and the team lead or senior management should be informed of this noncompliance. Should this noncompliance occur regularly then it may result in an audit of the user's code.

1. **Quality Assurance Metrics**

The following metrics will be used to measure the quality of our QA testing and to assist in ensuring our code is of professional standard.

* Test Execution:
  + Number of tests completed and their results.
* Defect distribution:
  + Types of flaws discovered during testing.
* Time to test:
  + Average time to test a selection of code.

**Quality Assurance Check List Forms**

* Code is free of bugs.
* Code achieves goals outlined.
* System does not negatively interfere with other systems.
* Code runs effectively and efficiently.

## 8.4 Change Management Plan

**Prepared by: Myles Hosken**

**Purpose of Change Management Plan:**

* This document aims to establish a structured and systematic approach to manage changes in the project.
* It should include details on how to request, review, communicate, and implement changes in the project.

**Change Request Submission:**

* Change requests can be submitted by contacting the project manager to develop and implement an appropriate form for the change.
* The form should include a detailed description of the requested change, the impact it will have on the project, and the reason behind the change.

**Change Request Review:**

* The project manager and technical lead (area specific) will review the change request.
* The review will consider the size, impact, and complexity of the change, as well as the project's schedule and budget.

**Change Communication:**

* The project manager will communicate the change and its outcome to all stakeholders through email or project management platform.
* The communication will include the reason for the change, its impact on the project, and any potential risks.

**Change Implementation Plan:**

* If the change is approved, the project team will develop an implementation plan.
* The plan will include details on how to implement the change, such as the scope, schedule, cost, labour, and technology requirements.
* If the implementation plan is expected to affect scheduling, this will be communicated to the client to be accepted. If it effects deliverable dates, this will also be established in the plan and communicated to the client before final approval.

**Change Result Review:**

* The project manager and technical lead will review the results of the change to ensure it has met its objectives and delivered the expected changes.
* The review will include an assessment of the change's impact on the project, its stakeholders, and any other relevant factors.

## 8.5 Change Request Form Template

**Prepared by: Myles Hosken**

Project Name: Digital Twin

Date Request Submitted:

Title of Change Request:

Change Order Number:

Submitted by:

Change Category:

* Scope
* Schedule
* Cost
* Technology
* Other

Description of change requested:

Events that made this change necessary or desirable:

Justification for the change/why it is needed/desired to continue/complete the project:

Impact of the proposed change on:

Scope: [Explain how the change will impact the scope of the project]

Schedule: [Explain how the change will impact the project schedule]

Cost: [Explain how the change will impact the project cost]

Staffing: [Explain how the change will impact project staffing]

Risk: [Explain any new risks that may arise from the change]

Other: [Explain any other impacts of the proposed change]

Suggested implementation if the change request is approved:

## 9.1 Milestone Report

**Prepared by:** Harshil Patel **Date: 20/03/2023**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Milestone** | **Date** | **Status** | **Responsible** | **Issues/Comments** |
| Proposal Presentation | 07/04/2023 | Running | Whole Team |  |
| Project Proposal Deadline | 07/04/2023 | Running | Whole Team |  |
| Mid-Semester 1 Break Starts | 07/04/2023 |  |  | The start of mid- semester break |
| Mid-Semester 1 Break Ends | 21/04/2023 |  |  | The end of mid- semester break |
| Mid-term Review | 02/06/2023 | Running | Whole Team |  |
| Semester 1 Break Starts | 23/06/2023 |  |  | The start of semester break |
| Semester 1 Break Ends | 17/07/2023 |  |  | The end of semester break |
| Mid-Semester 2 Break Starts | 04/09/2023 |  |  | The start of mid- semester break |
| Mid-Semester 2 Break Ends | 15/09/2023 |  |  | The end of mid- semester break |
| Project Deliverable Handover | Week 11 | Running | Whole Team | Exact deadline unspecified |
| Client Feedback | Week 12 | Running | Clients | Exact deadline unspecified |
| Mentor Feedback | Week 13 | Running | Mentor | Exact deadline unspecified |
| Team Poster | Week 13 | Running | Whole Team | Exact deadline unspecified |
| Team Portfolio | Week 14 | Running | Whole Team | Exact deadline unspecified |
| Reflective Report | Week 14 | Running | Individual | Exact deadline unspecified |

## 9.2 Project Schedule

**Prepared by:** Jane Jung

Sections highlighted in green mark the mid-term break for semester 1, semester break and the mid-term break for section 2.

Graphical user interface, application, table, Excel

Description automatically generated

Graphical user interface, application, table, Excel

Description automatically generated

Graphical user interface, application, table, Excel

Description automatically generated

Table

Description automatically generated with medium confidence

A picture containing timeline

Description automatically generated

Chart, waterfall chart

Description automatically generated

Table

Description automatically generated

Table

Description automatically generated

Graphical user interface

Description automatically generated with low confidence