COMP3850 Project Deliverable Certificate

Name of Deliverable	*Deliverable 3*
Date Submitted	*28 / 04 / 2022*
Project Group Number	*18*
Rubric stream being followed for this deliverable (highlight one) *Note: the feasibility study has the same rubric for all streams.*	*SOFTWARE Rubric* *GAMES Rubric* *CYBERSECURITY Rubric* *DATA SCIENCE Rubric*

We, the undersigned members of the above Project Group, collectively and individually certify that the above Project Deliverable, as submitted, **is entirely our own work**, other than where explicitly indicated in the deliverable documentation.

INITIALS	SURNAME	GIVEN NAME	STUDENT NUMBER	SIGNATURE *(IN-PERSON OR DIGITAL)*
EH	*Horvath*	*Erik*	*46042628*	
ST	*Torfeh Nejad*	*Sepehr*	*46042547*	
RS	*Shrestha*	*Rojwal*	*46137572*	
MI	*Ikeda*	*Marcus*	*45415064*	
LT	*Te*	*Lance*	*45581819*	

NB: please write all details clearly (if handwritten).

List of tasks completed for the deliverable and activities since last deliverable certificate with totals for each individual team member and whole team

Performed by *(Student Names)*	Duration *(hrs)*	Complexity *(L, M, H)*	Name of task	Checked by *(Initials)*
Lance Te	10	M	Writing my sections in D3 and editing others' sections	ST, EH
	2	H	Formatting D3	
	4	M	Designing our Figma MVP	
	12	H	Implement UI for MVP (login and persona page, logout)	ST
	3	L	Weekly team and client meeting	
	1	M	Presenting our MVP to client	
Total	32			
Sepehr Torfeh Nejad	3	H	Implement redux for frontend state management	LT
	1	M	Week 7 Meetings	
	3	H	MVP and testing Doc D3	LT
	4	M	Designing our Figma MVP	
	6	H	D3 documentation and MVP code implementation	LT
	6	H	Server code implementation	LT
	2	H	Finalising MVP doc and Test cases	EH
	4	H	Finalising MVP code for presentation	LT
	1	H	Presenting MVP to Faethm and weekly meeting	
Total	30			
Erik Horvath	2.5	H	Research Time for D3	
	2	M	D3 Planning	
	2	L	Team meeting time	
	1	L	Meeting with Marcus for D3 Task	
	5	M	Design Document Writing D3	LT
	1	M	Proofreading all D3 documentation	
	2	L	Faethm Meetings	
	1.5	M	Window Navigation Diagram	LT
Total	17			

Performed by *(Student Names)*	Duration *(hrs)*	Complexity *(L, M, H)*	Name of task	Checked by *(Initials)*
Marcus Ikeda	0.5	L	ER diagram	
	2	M	Activity diagrams	
Total	2.5			
Rojwal Shrestha	3	M	Research for D3	
	3	M	Writing for D3	
	1.5	M	Revising D2 document	
	1	L	Team Meetings	
Total	8.5			
Team Total	90			

Revision Table

ID	Document Version	Description	Date
1	Project Plan v1.0	Original submission for Deliverable 2.	31/03/22
2	SRS v1.0	Original submission for Deliverable 2.	31/03/22
3	Project Plan v2.0	Some work was carried out during the first week of holidays. Updated timelines & turnaround expectations.	12/04/22
4	SRS v2.0	Added a 'Contextual Diagram' to show the Onboarding UI in context of the entire Faethm application.	15/04/22
5	SRS v2.0	Listed the definitions in alphabetical order.	15/04/22
6	SRS v2.0	Considered user classes with accessibility requirements.	16/04/22
7	SRS v2.0	Clarified front-end user browser support.	16/04/22
8	SRS v2.0	Clarified what type of documents will be provided.	17/04/22
9	SRS v2.0	Numbered each sub dot-point.	17/04/22
10	SRS v2.0	Added 'Fit Criteria' for each functional and non-functional requirement.	17/04/22
11	Analysis Doc v1.0	Original submission for Deliverable 3.	28/04/22
12	Design Doc v1.0	Original submission for Deliverable 3.	28/04/22
13	Testing Doc v1.0	Original submission for Deliverable 3.	28/04/22

Project Plan

FÆTHM

Lance Te, Sepehr Torfeh Nejad, Marcus Ikeda, Erik Horvath, Rojwal Shrestha

COMP3850 Sem 1, 2022 Deliverable 3

Contents

1. Introduction 2
2. Statement of Purpose 2
3. Risk Management 3
4. Resource Management 4
   - 4.1. People 4
   - 4.2. Software 4
   - 4.3. Hardware 4
5. Team Organisation 5
6. Project Schedule 6
   - 6.1. Task / Activities / Phases 8
   - 6.2. Timeline 11
   - 6.3. Resources Allocated 13
7. Process Model 13
8. Documentation 14
9. Assumptions 14

1. Introduction

Faethm AI is a SaaS based AI powered platform that uses extensive datasets to provide insightful information to professionals and their industry. The team at Faethm has partnered up with Macquarie University to provide the opportunity for students to gain valuable industry experience. The project tasked to us is to create a personalised landing page for a new user to improve their onboarding experience.

2. Statement of Purpose

This project plan is the formal documentation providing an elaborate overview of the management and execution of the project as a whole. It is an important reference tool for all parties involved including the developers (Macquarie University students) and stakeholders (Faethm team). The project plan provides information about the scope of the project as well as the technical and management strategies of the project. This plan should be the primary resource for confirmation on the strategy for managing the project.

Project areas defined in this project plan include:

- Risk management strategies including a risk matrix outlining the risk probabilities, consequences, and the potential impact of the associated risk
- Mitigation strategies to help avoid the risks or reduce the overall impact of the potential risk
- Management of the project resources including people, hardware software and other resources used in the project
- Organisation of the group members into roles and responsibilities
- Scheduling of all tasks and activities
- Identifying the products and documentation required for the project and justification for its purpose
- Software Development Life Cycle
- Gantt Chart with planned tasks, timelines, deliverables, and resource allocations for the project

3. Risk Management

It is important to mitigate the risk of undertaking a project by planning for potential risks that may occur. By following a risk matrix, we can assess the severity of a potential risk as well as how important it is to mitigate that risk.

	Neglible	Minor	Moderate	Significant	Severe
Very Likely	Medium	Medium	Medium/High	High	High
Likely	Low	Medium	Medium	Medium/High	High
Possible	Low	Medium	Medium	Medium/High	Medium/High
Unlikely	Low	Medium	Medium	Medium	Medium/High
Very Unlikely	Low	Low	Medium	Medium	Medium

Risk	**Description**	**Likelihood**	**Impact**
Impact to Current System	The project causes problems to the current system and as a result it no longer functions.	Unlikely	Significant
Loss of Data	The project does not run correctly and causes Faethm's data to be lost.	Very Unlikely	Severe
Changes to Requirements	Faethm changes project requirements creating more work that needs to be done	Possible	Minor
Team Unable to Complete Work	A member of the team is not able to complete their allocated work	Possible	Moderate
Project Incomplete	The project is not finished within the development period	Possible	Minor
Bugs in Software	Software has bugs during development	Very Likely	Negligible

Risk	**Mitigation Strategy**
Impact to Current System	The system should be reverted back to a state where it was previously functioning or a state from before the project was started.
Loss of Data	Data should be backed up before running the software.
Changes to Requirements	Maintain good organisation and communication with Faethm.
Team Unable to Complete Work	Maintain good communication between team members and have someone else take over their work.
Project Incomplete	N/A.
Bugs in Software	Debugging.

4. Resource Management

The resources required for the successful execution of the project are outlined here. The main resources are broken down into three categories: people, hardware and software. Since this is a software development project with the Faethm AI application being a SaaS hosted platform, the need for physical resources such as material goods and capital expenses is eliminated. Further elaboration on the three project resources is provided below.

4.1. People

This resource consists of two categories including the developers and the Faethm team. The developers are the 5 members from COMP3850 Group 18:

- Lance Te
- Erik Horvath
- Sepher Torfeh Nejad
- Marcus Ikeda
- Rojwal Shrestha

The Faethm team is primarily involved to provide guidance and support for the development team. Since they are the primary stakeholders as well as providing us access to speak with members from their development team, they are both a technical and strategic, high level, resource.

4.2. Software

The Onboarding UI project will be built using the MERN (MongoDB, Express, React, Node) stack for ease of integration with the rest of the Faethm application. The different layers of the UI will be developed using technologies across the MERN stack:

- Frontend: React.js
- Server: Express.js, Node.js
- Database: MongoDB

4.3. Hardware

All members require their own computer set up with access to the internet to be able to contribute to the project.

5. Team Organisation

The members of COMP3850 Group 18 are broken up into specific roles to take advantage of team members strengths and create a clear delegation of duties. The following is a list of each member and their project role:

- **Lance Te:** Team Lead, Lead Formatter, Developer, Documentation Support
- **Sepher Torfeh Nejad:** Lead Developer, Documentation Support
- **Erik Horvath:** Editor, Documentation Officer, Developer
- **Marcus Ikeda:** Documentation Officer, Developer
- **Rojwal Shrestha:** Documentation Officer, Developer

These roles are distributed based on the agreed skillsets of the group members. The roles have clearly defined duties and the role descriptions outlined below:

- Team Lead: Main point of contact, in charge of all deliverables and project component submissions, organising group meetings and general team leader duties.
- Lead Formatter: Responsible for compiling all group contributions of deliverables and formatting documentation for presentation purposes.
- Lead Developer: Most experienced developer with ability to delegate tasks and advise other developers.
- Lead Editor: Looks for misspellings, incorrect grammar, missed punctuations, inconsistencies and general flow of the text.
- Documentation Officer: Responsible for creating the documentation and fill in all required information needed to satisfy deliverables.
- Developer: Responsible for the coding and development of the project.
- Documentation Support: Provide support to Documentation Officers with technical documentation requirements.

6. Project Schedule

ID	Task Name	Product	Duration	Start	Finish
1	Deliverable 01	Feasibility Study	7 days	Wed 3/2/22	Thu 3/10/22
2	Deliverable 02	Project Plan and Requirements/Scoping Document	12 days	Wed 3/16/22	Thu 3/31/22
2.1	Activity 01	Project Plan for Deliverable 2	4 days	Wed 3/23/22	Sun 3/27/22
2.3	Activity 02	Scoping Doc for Deliverable 2	7 days	Wed 3/16/22	Thu 3/24/22
2.4	Activity 03	Quality Manual for Deliverable 2	5 days	Sun 3/27/22	Thu 3/31/22
3	Deliverable 03	Update Deliverable 2, Prototype/MVP, Design, Test Cases	17 days	Wed 4/6/22	Thu 4/28/22
4	Deliverable 04	Update Deliverable 2, User/Training Manual	12 days	Wed 5/4/22	Thu 5/19/22
5	Deliverable 05	Final Group Reflective Report	12 days	Wed 5/18/22	Thu 6/2/22
6	Deliverable 06	Project Presentation/Demonstration	12 days	Wed 5/18/22	Thu 6/2/22
7	Deliverable 07	Final Web Application Delivery	11 days	Wed 5/18/22	Wed 6/1/22
8	Deliverable 08	Final Exam	1 day	Tue 6/7/22	Tue 6/7/22
9	Checkpoint 01	Individual Contribution Form Submission (First Half)	4 days	Wed 3/30/22	Mon 4/4/22
10	Checkpoint 02	Individual Contribution Form Submission (First Half)	4 days	Wed 6/1/22	Sat 6/4/22
11	Functional Requirement 01	Login (FR1)	11 days	Wed 4/6/22	Wed 4/20/22
11.1	Activity 04	FR1 Login Functionality	4 days	Wed 4/6/22	Sat 4/9/22
11.2	Activity 05	FR1 Error/Unsuccessful Login Handling	4 days	Sat 4/9/22	Wed 4/13/22

11.3	Activity 06	FR1 Questionnaire Display	4 days	Wed 4/13/22	Sat 4/16/22
11.4	Activity 07	Testing for FR1	3 days	Sat 4/16/22	Tue 4/19/22
12	Functional Requirement 02	User Persona (FR2)	11 days	Wed 4/20/22	Wed 5/4/22
12.1	Activity 08	FR2 Persona Classification	4 days	Wed 4/20/22	Sat 4/23/22
12.2	Activity 09	FR2 Pre-defined Persona Display	3 days	Sat 4/23/22	Tue 4/26/22
12.3	Activity 10	FR2 User Input Capture Functionality	3 days	Tue 4/26/22	Thu 4/28/22
12.4	Activity 11	FR2 Store User Input Functionality	3 days	Thu 4/28/22	Mon 5/2/22
12.5	Activity 12	Testing for FR2	2 days	Mon 5/2/22	Tue 5/3/22
13	Functional Requirement 03	Use Case (FR3)	5 days	Wed 5/4/22	Tue 5/10/22
13.1	Activity 13	FR3 User's Use Case Determination from Answers/From FR2	4 days	Wed 5/4/22	Sat 5/7/22
13.2	Activity 14	Testing for FR3	3 days	Sat 5/7/22	Tue 5/10/22
14	Functional Requirement 04	Personalisation (FR4)	9 days	Sun 5/8/22	Wed 5/18/22
14.1	Activity 15	FR4 Welcome Functionality	4 days	Sun 5/8/22	Wed 5/11/22
14.2	Activity 16	FR4 Relevant Features Display	5 days	Wed 5/11/22	Tue 5/17/22
14.3	Activity 17	Testing for FR4	3 days	Sat 5/14/22	Tue 5/17/22
15	Activity 18	Testing For the Entire Web Application	10 days	Wed 5/18/22	Tue 5/31/22

6.1. Task / Activities / Phases

In order to deliver the project on time and to balance the workload amongst the team members, the project can be divided into tasks and activities.

1. **Deliverable 01**
   Product: Feasibility Study
   Summary: The task is to write an official document regarding the practicality of the project based the team's capability and experience as well as the project functional requirements
2. **Deliverable 02**
   Product: Project Plan, Scoping Document and Quality Manual
   This deliverable is divided in to 3 activities:
   2.1. **Activity 01:** Project Plan
   Summary: Write a document specifying risk and resource management also including the project schedule
   Participants: Erik, Marcus, Roj
   2.2. **Activity 02:** Quality Manual
   Summary: Write a document showing how quality is assessed and maintained throughout the project
   Participants: All Members
   2.3. **Activity 03:** Scoping Document
   Summary: Write a document providing an overview of the context and functionality of the project
   Participants: Sepehr, Lance
3. **Deliverable 03**
   Product:
   - Revised Project Plan, Quality Manual, Scoping Document
   - Analysis and Design Document
   - Testing Document
   - Prototype/MVP

   Summary: The task is to update the previously written document according to the new developments in the project. The Project Plan, Quality Manual and Scoping Document will be revised in a manner similar to Deliverable 2. The workload for Analysis & Design and Testing Document is yet to be decided.

4. **Deliverable 04**
   Product: Revised Project Plan, Quality Manual, Scoping Document along with a User Manual
   Summary: The task is to update the previously written document according to the new developments in the project. The Project Plan, Quality Manual and Scoping Document will be revised in a manner similar to Deliverable 2. The workload for User Manual is yet to be decided.
   Participants: All Members
5. **Deliverable 05**
   Product: Final Group Reflective Report
   Summary: a document reviewing all aspects of the project
   Participants: All Members

6. **Deliverable 06**
   Product: Project Presentation/Demonstration
   Summary: the group will present their system to the sponsors, academics and students
   Participants: All Members
7. **Deliverable 07**
   Product: Final Group Reflective Report
   Summary: a document reviewing all aspects of the project
   Participants: All Members
8. **Deliverable 08**
   Product: Final Exam
9. **Checkpoint 01**
   Product: Individual Contribution Form (First Half)
   Summary: each member is required to fill out a form assessing other member's contributions in the project
   Participants: All Members
10. **Checkpoint 02**
    Product: Individual Contribution Form (Second Half)
    Summary: each member is required to fill out a form assessing other member's contributions in the project
    Participants: All Members
11. **Functional Requirement 01**
    Product: Login (FR1)
    Summary: this functional requirement ensures only registered and authenticated users are allowed onto the application.
    Participants: All Members
    The task is split into activities focusing on fundamental aspects of the requirement and testing.
    11.1.1. Activity 04: FR1 Login Functionality
    11.1.2. Activity 05: FR1 Error/Unsuccessful Login Handling
    11.1.3. Activity 06: FR1 Questionnaire Display
    11.1.4. Activity 07: Testing for FR1
12. **Functional Requirement 02**
    Product: User Persona (FR2)
    Summary: this functional requirement classifies users into persona in order to assist the app tailor its offerings according to the user's needs
    Participants: All Members
    The task is similarly split into activities focusing on fundamental aspects of the requirement and testing.
    12.1.1. Activity 08: FR2 Persona Classification
    12.1.2. Activity 09: FR2 Pre-defined Persona Display
    12.1.3. Activity 10: FR2 User Input Capture Functionality
    12.1.4. Activity 11: FR2 Store User Input Functionality
    12.1.5. Activity 12: Testing for FR2

13. **Functional Requirement 03**

    Product: Use Case (FR3)

    Summary: this functional requirement determines a user's use case

    Participants: All Members

    The task is similarly split into activities focusing on fundamental aspects of the requirement and testing.

    13.1.1. Activity 13: FR3 User's Use Case Determination from Answers/From FR2

    13.1.2. Activity 14: Testing for FR3

14. **Functional Requirement 04**

    Product: Personalisation (FR4)

    Summary: this functional requirement will display a personalised landing page to the user

    Participants: All Members

    The task is similarly split into activities focusing on fundamental aspects of the requirement and testing.

    14.1.1. Activity 15: FR4 Welcome Functionality

    14.1.2. Activity 16:FR4 Relevant Features Display

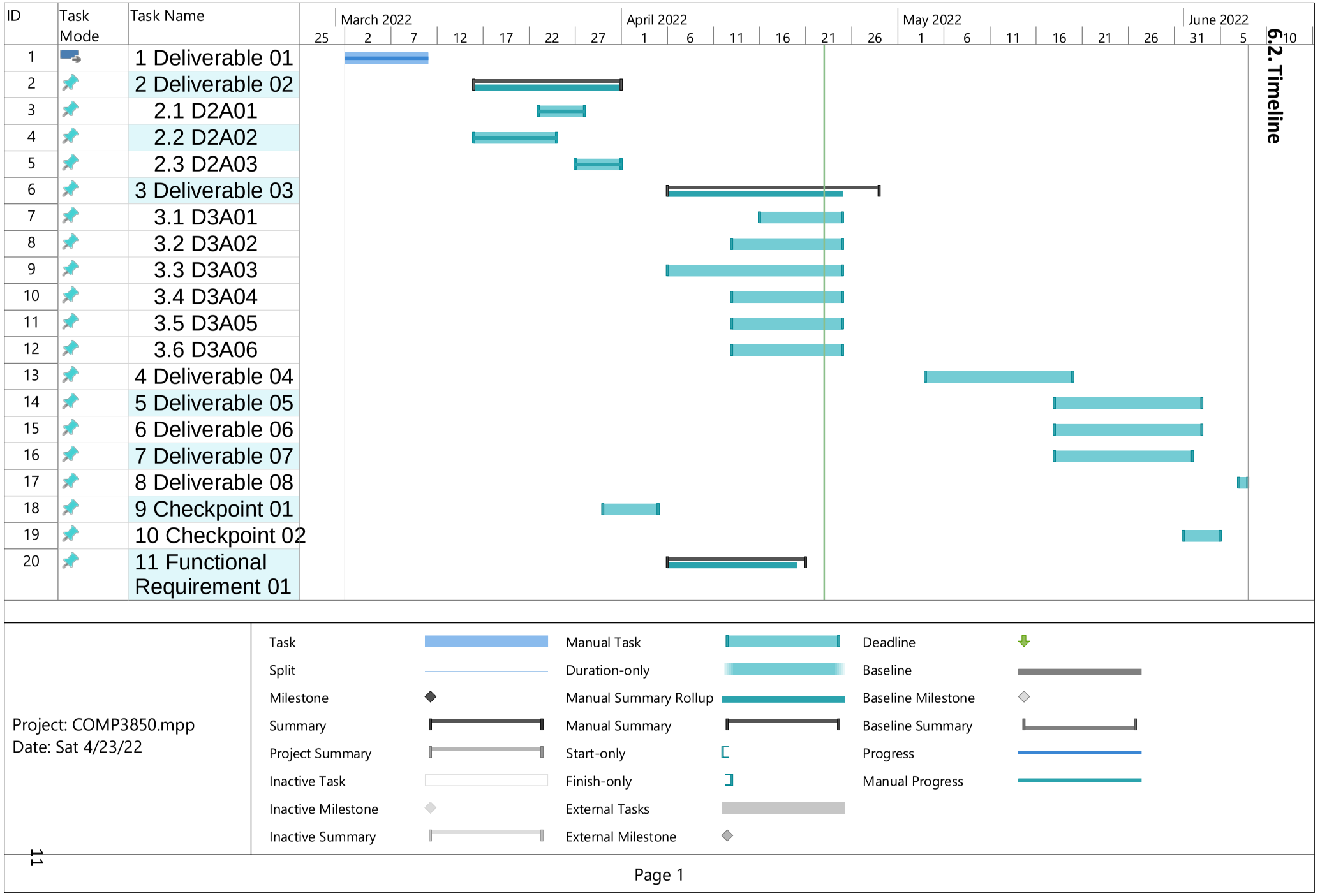
    14.1.3. Activity 17: Testing for FR4

15. **Final Testing**

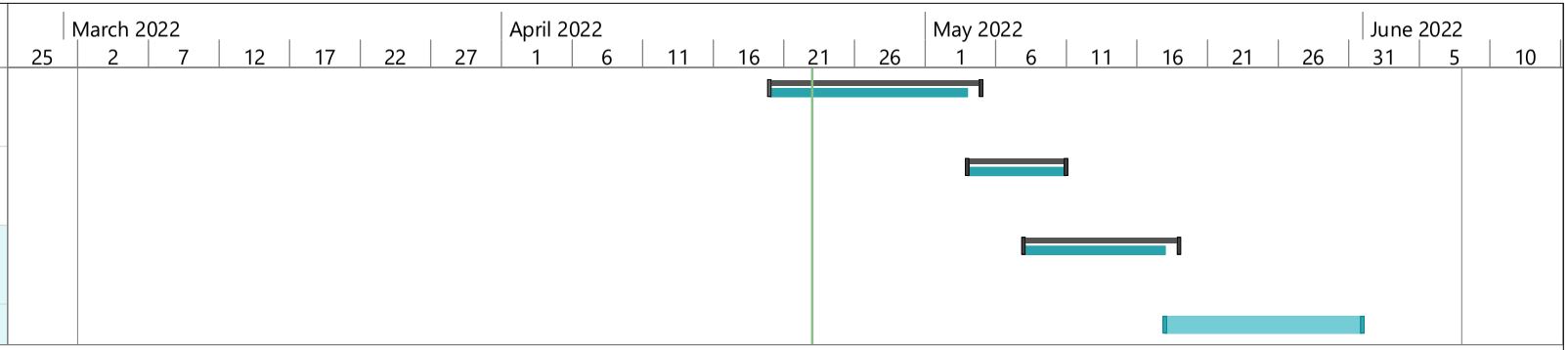
    Product: Testing for the Entire Web Application

    Summary: after all Functional Requirement are completed and tested, all modules will be integrated and tested to deliver a satisfactory product

    Participants: All Members


ID
Task Mode
Task Name
March 2022
April 2022
May 2022
June 2022
25 2 7 12 17 22 27 1 6 11 16 21 26 1 6 11 16 21 26 31 5 10
1 1 Deliverable 01
2 2 Deliverable 02
3 2.1 D2A01
4 2.2 D2A02
5 2.3 D2A03
6 3 Deliverable 03
7 3.1 D3A01
8 3.2 D3A02
9 3.3 D3A03
10 3.4 D3A04
11 3.5 D3A05
12 3.6 D3A06
13 4 Deliverable 04
14 5 Deliverable 05
15 6 Deliverable 06
16 7 Deliverable 07
17 8 Deliverable 08
18 9 Checkpoint 01
19 10 Checkpoint 02
20 11 Functional Requirement 01
6.2. Timeline
Project: COMP3850.mpp
Date: Sat 4/23/22
Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Baseline
Baseline Milestone
Baseline Summary
Progress
Manual Progress

ID	Task Mode	Task Name
25		12 Functional Requirement 02
31		13 Functional Requirement 03
34		14 Functional Requirement 04
38		15 Activity 18



Project: COMP3850.mpp
Date: Sat 4/23/22

Task	Manual Task	Deadline
Split	Duration-only	Baseline
Milestone	Manual Summary Rollup	Baseline Milestone
Summary	Manual Summary	Baseline Summary
Project Summary	Start-only	Progress
Inactive Task	Finish-only	Manual Progress
Inactive Milestone	External Tasks	
Inactive Summary	External Milestone	

6.3. Resources Allocated

The resources we have been allocated for this project can be categorised as personnel, hardware and software. The personnel resources that we have been allocated consists of all the individuals that are assigned to the project. This includes each of the group members that are working on the project, as well as any technical skills that each group member has, and the representatives that have been sent by Faethm to give us feedback and advice on the project. The hardware resources we have is any hardware resources that is owned by the group, i.e. laptops/computers and any hardware resources that we have permission to use from Macquarie University or Faethm. Since we are developing an online platform, we have mostly been allocated software resources by Faethm for development. Faethm has given us their website and their component library to use for development, and a Figma document where they have compiled their UI conventions as well as the personas for the user profiles that we need to categorise. Faethm has also compiled resources for us containing the basics of the software that we will be using to develop the project in (JavaScript, Typescript, ReactJS, NodeJS). We have also been provided with some of Faethm's insights, where they break down all the features of their platform and some diagrams explaining the workflow of the processes that are currently in place and have been planned.

All our code for the project is stored in our private GitHub repository, while all documentation is stored in a shared Google Drive folder. Any diagrams required are drawn using Lucid Chart. Documentation for this project is primarily done in Google Docs. The team has collectively agreed to use Discord as our main communication channel while we use Slack and Zoom for communicating without sponsors. We mainly use VSCode as an editor for projects.

7. Process Model

The process model that we are using for this project is agile project management. The agile method allows for Faethm to maintain more control over the project because they are more easily able to make changes to the software requirements. It will also allow them to provide input at regular intervals since the work is completed in a continuous stream and not all at once. This means that they will be able to provide feedback on completed work and request changes or additional features if necessary. The agile method is the ideal method for this project because it has the lowest chance of having the project fail due to the way that work is constantly being done over the development period, which as a result will make the progress of the project predictable. The software will continuously be improved during the entire development period with features and changes being made at the request of Faethm, which allows the project to maintain flexibility on the work that needs to be done and the features that need to be implemented for the project to be successful. This agile method also suits the project because the work that needs to be done is not confirmed and may be easily changed so any work that is done can be modified or built upon as development continues.

8. Documentation

The documentation required for this project include:

1. **Feasibility Study:** Discusses the viability of the project, what the nature of the problem is and what needs to be done to address the current situations. Possible solutions for the problem are addressed here and we determine what we need to do to make the project successful.
2. **Project Plan:** Addresses the organisation of the project i.e. Potential risks that could potentially prevent the project from succeeding, the tasks that need to be completed for the project to be successful and how the projects resources will be allocated to ensure the project runs smoothly. A timeline is also given to outline the progression of development. This will need to be updated if the requirements of the project change.
3. **Project Requirements & Scoping Document:** Goes over the purpose of the software and an overview of how it will function. This includes documentation of how the software will work as well as an overview of all of the product's functions. The design restrictions and constraints are also discussed in this document. The functional requirements can be updated throughout development if the need to change the project requirements comes up.
4. **Analysis Document:** Contains a case diagram showing the actors and use cases and the relationships between them. There should be a case diagram for each piece of functionality in the system.
5. **Design Document:** Contains the basic architecture of the system, explaining how processes in the system will be handled as well as explaining the design choices with justifications and trade-offs.
6. **Testing Documentation:** Contains plans for testing strategy, testing types, testing schedule, testing tools and resources assigned, testing milestones and test deliverables. This document should cover all the testing processes used.
7. **User Manual:** Contains documentation for users who are unfamiliar with the software. It should allow a moderately computer-literate user to fully utilise the software's functionality. Contents of the document should contain information that will benefit the user such as an installation guide, configuration settings, screenshots with example data, training, troubleshooting help etc.

9. Assumptions

This project plan was made under the assumptions that:

A1. We will be able to maintain contact with the representatives from Faethm for the duration of the project
A2. Faethm will provide us with the appropriate resources to complete the project.
A3. All group members have access to appropriate hardware and software to complete the project.
A4. The project's requirements can be changed by Faethm at any point in the project if it is required.

Scoping Document

FÆTHM

Lance Te, Sepehr Torfeh Nejad, Marcus Ikeda, Erik Horvath, Rojwal Shrestha

COMP3850 Sem 1, 2022 Deliverable 2

Contents

1. Introduction 2
   1.1 Purpose 2
   1.2 Scope 2
      1.2.1. Data Flow Diagram 3
      1.2.2. Logic Flow Diagram 3
      1.2.3. Context Diagram 3
   1.3 Definitions, Acronyms, and Abbreviations 4
   1.4 References 4
   1.5 Overview / Document Convention / Intended Audience 4
      1.5.1. Overview 4
      1.5.2. Intended Audience 4
2. Overall Description 5
   2.1 Product Perspective 5
   2.2 Product Functions 5
   2.3 User Classes and Characteristics 5
      2.3.1. Faethm Developers 5
      2.3.2. Users 5
   2.4 Operating Environment 6
   2.5 User Documentation 6
      2.5.1. Faethm Developers 6
      2.5.2. Users 6
3. Requirements 7
   3.1 Functional Requirements 7
      3.1.1. FR1 – Login 7
      3.1.2. FR2 – User Persona 7
      3.1.3. FR3 – Use Case 7
      3.1.4. FR4 – Personalisation 7
   3.2 Design and Implementation Requirements / Constraints 8
      3.2.1. Time 8
      3.2.2. Cost 8
      3.2.3. Technical 8
   3.3 Usability Requirements 9
      3.3.1. Accessibility 9
      3.3.2. Efficiency 9
      3.3.3. Intuitiveness 9
   3.4 Other Non-functional Requirements 10
4. Client Feedback 11
   4.1 Meeting Date and Time 11
   4.2 Feedback Received 11
   4.3 Team Response 11

1. Introduction

Faethm is a data driven, AI incorporated analytic company which provides a number of services on their platform for their users. They have partnered with Macquarie University to provide industry experience opportunities for students. Their aim is to develop a personalised landing page to provide new users a better onboarding experience when using the Faethm application for the first time.

1.1 Purpose

The purpose of this Software Requirements Specification (SRS) is to provide an overview of both the context and functionality of the Onboarding UI for the Faethm application.

This document serves as a medium between stakeholders (Faethm) and developers (Techpad Technology) to discuss and clarify the functionalities of the Onboarding UI to ensure both parties' expectations are met before any development begins. Given the preliminary nature of this document, both parties can make agreed changes throughout.

Once the development phase commences, the SRS document will then serve as a reference for the developer team to follow and stay on track while developing the Onboarding UI.

Thus, the SRS must be a collaborative effort between stakeholders and developers to produce a well-defined, clear and completely understood document.

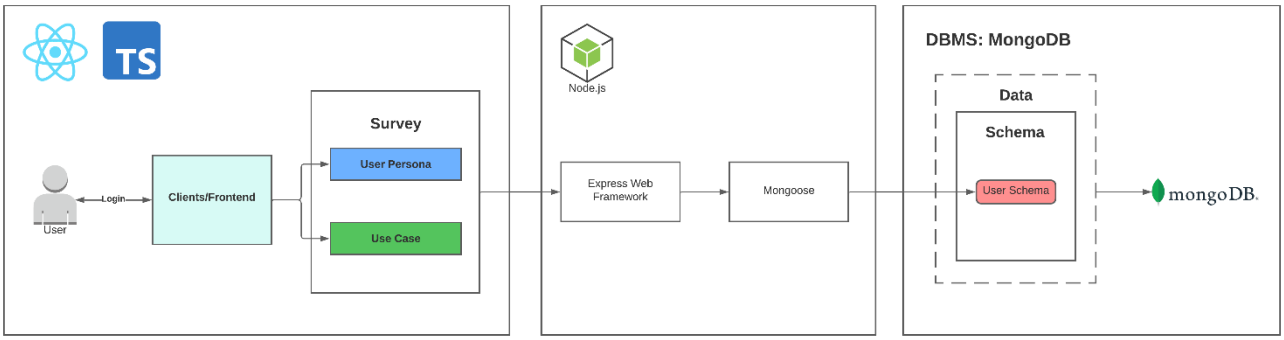
1.2 Scope

The Onboarding UI project is an additional data driven step for First Time User (FTU) which provides users with the opportunity to customise their experience on the Faethm application. The UI provides a tailored landing page for users and improves the user experience (UX) on the Faethm platform. The UI is a Single Page Web Application (SPA), with MERN framework (ReactJS, NodeJS, MongoDB) which may later be integrated with the Faethm web application.

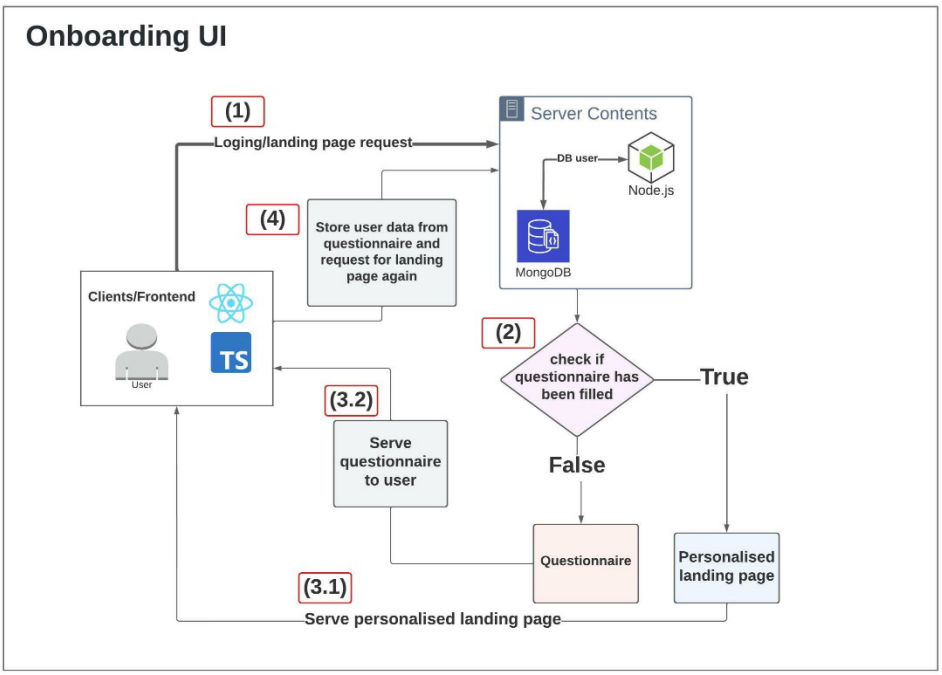
Comprising the first-time user experience (FTUE), the Onboarding UI is only applicable to users logging in for the first time. The UI **will** check if the user has previously logged in. For first time users, the UI serves a short set of questions. This data is stored for future reference and used to assign users a predefined persona and determine the most relevant services available on the platform to them. Ultimately, this enables the UI to provide a personalised landing page. The UI **will not** serve the short set of questions to users that have previously logged in. Instead, their personalised landing page will be served directly.

The Faethm platform offers a number of different services and they seek to enhance the overall user experience. By providing a personalised landing page, the Onboarding UI streamlines the user experience, offers Faethm greater insight into their user, and offers potential avenues for the development of future products.

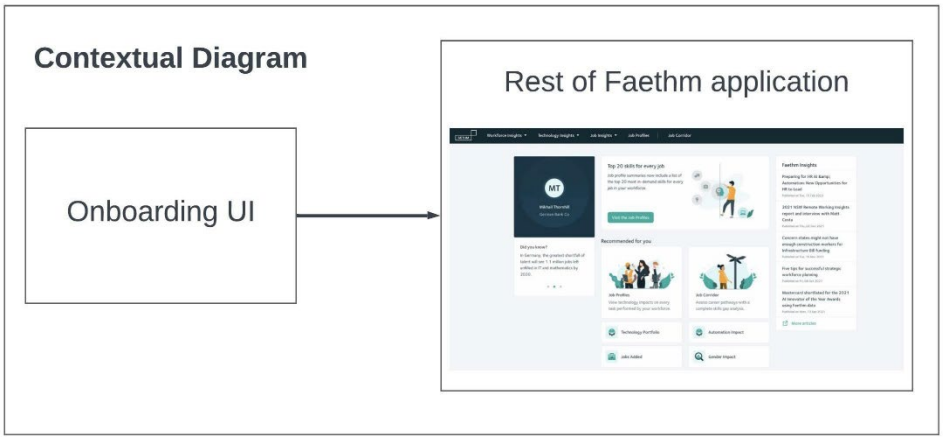
1.2.1. Data Flow Diagram



1.2.2. Logic Flow Diagram



1.2.3. Context Diagram



1.3 Definitions, Acronyms, and Abbreviations

Acronyms/Abbreviations	Definitions
AI	Artificial Intelligence
API	Application Programming Interface
CSRF	Cross-Site Request Forgery
FTUE	First Time User Experience
SPA	Single Page Web Application
SRS	Software Requirement Specification
UI	User Interface
UX	User Experience

1.4 References

Systems and software engineering – Life cycle processes – Requirements engineering (ISO/IEC/IEEE 291448, 2011). Obtained from: https://login.simsrad.net.ocs.mq.edu.au/login?qurl=https://ieeexplore.ieee.org%2fdocument%2f6146379

1.5 Overview / Document Convention / Intended Audience

1.5.1. Overview

In this document, product perspective and functions are described. It also contains user characteristics and the operating environment of the application.

Functional requirements of the UI, design and implementations requirements and usability of the application are included.

1.5.2. Intended Audience

Stakeholders (Faethm) and developers (Techpad Technology) are the intended audience of this document. It aims to inform the developers and the stakeholders of the purpose of this project and clarifies the requirements of the application for both parties.

2. Overall Description

2.1 Product Perspective

Our Onboarding UI will serve as the First Time User Experience (FTUE). This will be the first page a user interacts with upon logging into the Faethm application for the first time.

While we develop this feature, the interface will remain **independent** of the rest of the Faethm app. When implemented into the Faethm app, the interface will sit just behind the login screen and provide users with direct access to the tools and articles most relevant to their use case.

2.2 Product Functions

Our Onboarding UI will provide users with a personalised dashboard that caters to their needs and what they hope to gain out of the app. The UI offers two core functionalities:

- To determine the persona of a user (refer to *User Classes and Characteristics*)
- To determine a user's use case

By determining these from the user, the UI will provide the appropriate insights (workforce, technology, job) and articles that will align with their use case.

2.3 User Classes and Characteristics

There are two user classes that will interact with the Onboarding UI: Faethm developers and users.

2.3.1. Faethm Developers

The interaction of Faethm developers with the Onboarding UI will involve maintaining and updating the UI to meet their user's evolving needs. This could include updating personas to reflect changes in the type of employees in the corporate structure to general maintenance of the UI.

2.3.2. Users

Users will make up the primary group interacting with the Onboarding UI. They will answer a short set of questions the first time they login. This is used to classify users into one of nine broadly generalised personas based on their occupation and position within their company. These personas are:

1. Concerned executive
2. Head of the division
3. Data loving analyst
4. Overbooked manager
5. Lost employee
6. Learning manager
7. Workforce planner
8. Generalist HR manager
9. Automation engineer

Each type of user has varying levels of technical expertise, capabilities, and accessibility requirements. This is a strong consideration for the product's design requirements to ensure the UI is intuitive across all user types.

2.4 Operating Environment

Within the context of users, the Onboarding UI will operate through their web browser whether that be on a laptop or mobile device. This requires the UI, as with any modern UI, to be able to adapt to different screen sizes and aspect ratios.

Within the context of the Faethm developers, the Onboarding UI should operate and will thus be built using the MERN (MongoDB, Express, React, Node) stack for ease of integration with the rest of the Faethm application. The different layers of the UI will be developed using technologies across the MERN stack:

- Frontend: React.js
- Server: Express.js, Node.js
- Database: MongoDB

The frontend provides a short set of questions to determine their persona and use case. From there, the UI will display personalised content derived from a user's categorisation from the initial questionnaire. The server will handle requests from the frontend and serve the initial page to the user's browser. The database stores user information which includes their answers to the set of questions and the persona they are classified as.

The frontend will be designed to support all modern web-browsers and mobile devices. All existing browser-defined styles will be reset and built from the ground up to ensure a consistent UI experience across all devices.

2.5 User Documentation

2.5.1. Faethm Developers

Documentation of software features, and a map of the code will be provided to the Faethm developers to ensure a smooth handover. These documents will be in a PDF format.

2.5.2. Users

A help hint / tooltip will provide sufficient information for users to interact with the Onboarding UI.

3. Requirements

3.1 Functional Requirements

3.1.1. FR1 – Login

The login feature will serve as a barrier to ensure only registered and authenticated users are allowed entry into the Faethm application.

REQ 1. The system shall display the login form.
REQ 2. The system shall display an error message on unsuccessful login.
REQ 3. The system shall display a short set of questions upon successful login.

3.1.2. FR2 – User Persona

Classifying the user into a persona will assist the Faethm app to customise its offerings to better suit the needs of the user.

REQ 1. The system shall classify users into one of nine persona classes.
REQ 2. The system shall provide a user form displaying the nine pre-defined personas with an icon graphic and short description for each.
REQ 3. The system shall capture user input, and store that information in the database.

3.1.3. FR3 – Use Case

Determining a user's use case will assist the Faethm app to customise its offerings to cater more specifically to user needs.

REQ 1. system shall provide a short set of questions each with a set of possible answers.
REQ 2. The system shall determine the user's use case derived from their answers from these questions.

3.1.4. FR4 – Personalisation

The system shall use FR1 and FR2 to display a personalised landing page to the user.

REQ 1. The system shall provide a welcome message customised with the user's name and organisation.
REQ 2. The system shall provide tools relevant to the user.
REQ 3. The system shall provide articles relevant to the user.

3.2 Design and Implementation Requirements / Constraints

3.2.1. Time

Currently, the team is balancing part- or full-time work alongside this project. This was an important consideration when proposing a realistic timeframe. Ultimately, it was agreed that a reasonable timeframe to develop the documentation and final deliverable for the Onboarding UI ranges from 13 weeks to a maximum of 16 weeks. A breakdown of the deliverable's timeline is outlined under 'Project Schedule' in the Project Plan.

3.2.2. Cost

The development of the Onboard UI is an unpaid endeavour and thus there are no cost constraints applicable.

3.2.3. Technical

There are two technical considerations for the Onboarding UI.

1. The Faethm application adheres to a distinct style throughout which is clearly defined in their CSS. Thus, the Onboarding UI must continue to maintain a **consistent UI**.
2. The Faethm application was developed using the MERN stack. Thus, the Onboarding UI must be developed in the **MERN stack** to ensure integration compatibility.

3.3 Usability Requirements

3.3.1. Accessibility

The Onboarding UI should:

REQ 1. Offer a responsive design to serve users a consistent experience whether they are accessing the UI on a mobile device or laptop.
REQ 2. Be browser-neutral to ensure users are not limited to interacting with the UI on certain web browsers.

Fit Criteria: The web application should be accessible on all devices.

3.3.2. Efficiency

The Onboarding UI should maintain reasonable load times for efficiency, and to improve the user experience.

REQ 1. The frontend should make minimal requests to the server and have clear logic.
REQ 2. The server should be well structured and have a well-defined API.

Fit Criteria: The web application should be able to light-weight and load within a few milliseconds.

3.3.3. Intuitiveness

The Onboarding UI should:

REQ 1. Be user-friendly, that is, easy to learn and navigate. Additional help hints or tooltips may be implemented to increase user-friendliness.
REQ 2. Provide a low perceived workload to ensure a smooth learning experience while interacting with the UI.

Fit Criteria: The web application should be easy to navigate such that users should be able to appreciate its features within five minutes.

3.4 Other Non-functional Requirements

3.4.1. Performance

REQ 1. Support lower end devices
- Devices accessing the Onboarding UI should be JavaScript supported due to the nature of the software (React).
- The Onboarding UI shall be a relatively lightweight software so it can be accessed by lower end devices.

REQ 2. Support slower internet connections
- The Onboarding UI shall minimise the required API request by implementing clear and well-thought-out frontend logic to ensure a consistently smooth experience despite a slower internet connection.

REQ 3. Support an increasing numbers of simultaneous users
- The Onboarding UI shall be hosted on a server with appropriate infrastructure to support a growing user base.
- Load-balancing and intelligent scheduling algorithm is required during peak time.

REQ 4. Instantaneous load times
- The frontend React components shall be optimised to avoid long initial loading times of the UI.

Fit Criteria: The web application should be able to run on older devices. We have reasonably defined older devices as those that are up to 3 generations old.

3.4.2. Security

REQ 1. Authentication
- The Onboarding UI shall only allow registered users who provide the correct login credentials into the application.
- The Onboarding UI shall deny access to unregistered or unauthenticated users.

REQ 2. Authorisation / Access
- The Onboarding UI does not distinguish between admin users and general users. However, user access will be restricted to authorised workspaces.
- Users shall only be able to view their workspaces.

REQ 3. CSRF Prevention
- The Onboarding UI shall implement JSON Web Tokens (JWT) to mitigate the risks of Cross-Site Request Forgery (CSRF).

Fit Criteria: The web application should only allow authenticated users into the system, and only provide access to authorised parts of the application.

3.4.3. Scalability

As the Faethm platform continues to expand, the need for a scalable solution grows with it.

REQ 1. The Onboarding UI shall be developed in a manner that provides a level of convenience for the Faethm developers to expand or shrink the number of questions.

REQ 2. The Onboarding UI shall also be developed to handle a growing number of concurrent users.

Fit Criteria: The web application should be able to a large number of concurrent users.

4. Client Feedback

4.1 Meeting Date and Time

We submitted our *Scoping Document* for review via email and GitHub on Monday 28 March 2022, 9:00am AEDT. In addition, we submitted our entire *Deliverable 2* to provide context to the *Scoping Document*.

4.2 Feedback Received

The feedback we received was positive and confirmed our vision of the Onboarding UI aligned with those of the Faethm team.

4.3 Team Response

There are no further response / action points required at this stage.

Analysis Document

FÆTHM

Lance Te, Sepehr Torfeh Nejad, Marcus Ikeda, Erik Horvath, Rojwal Shrestha

COMP3850 Sem 1, 2022 Deliverable 3

Contents

1. Minimum Viable Prototype 2
   1.1 Login 2
   1.2 Survey 3
   1.3 Loading Page 5
2. Use Case Diagram 6
3. Use Case Descriptions and User Stories 7
   1.4 Logon 7
   1.5 Pick Persona 8
   1.6 Personalised Dashboard 9
4. Assumptions 10

1. Minimum Viable Prototype

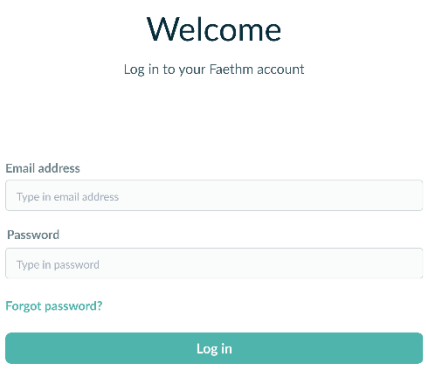
The screenshots included below are developed on Figma. Our Figma file can be accessed directly via: https://www.figma.com/proto/bCtAFQNPHas1c61rER3t0Z/Onboarding-UI?node-id=0%3A1

1.1 Login

In order to access any part of the Faethm AI application, including our Onboarding UI, a user will need to be authenticated. This is achieved through the login form which will allow the user to verify their identity. The login page is patterned off Faethm's existing login page to ensure a consistent user interface.

To login, users will need to provide two things: the email address and password used when they registered for their Faethm account. The email address input field will have validation rules that ensure that a valid email address is provided that includes the username, an @ symbol, domain name, a dot, and the domain. The email address and password will need to match those on record for a user to access the rest of the Faethm application.

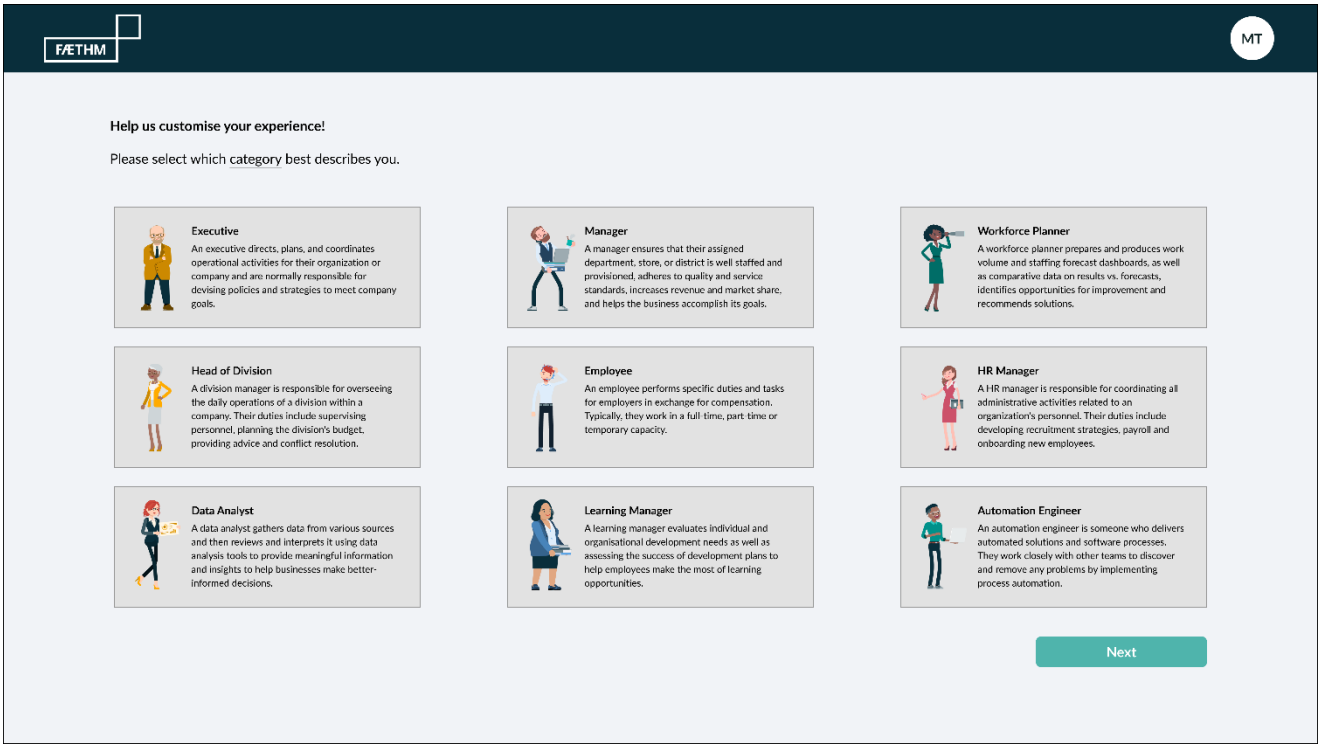
Once a user has provided a valid email address and password, and is authenticated as an existing user in the database, a JSON Web Token (JWT) is stored in both the database and the user's local storage. This is to allow constant checking to ensure that the user's session is still valid while they are using the Faethm application.



1.2 Survey

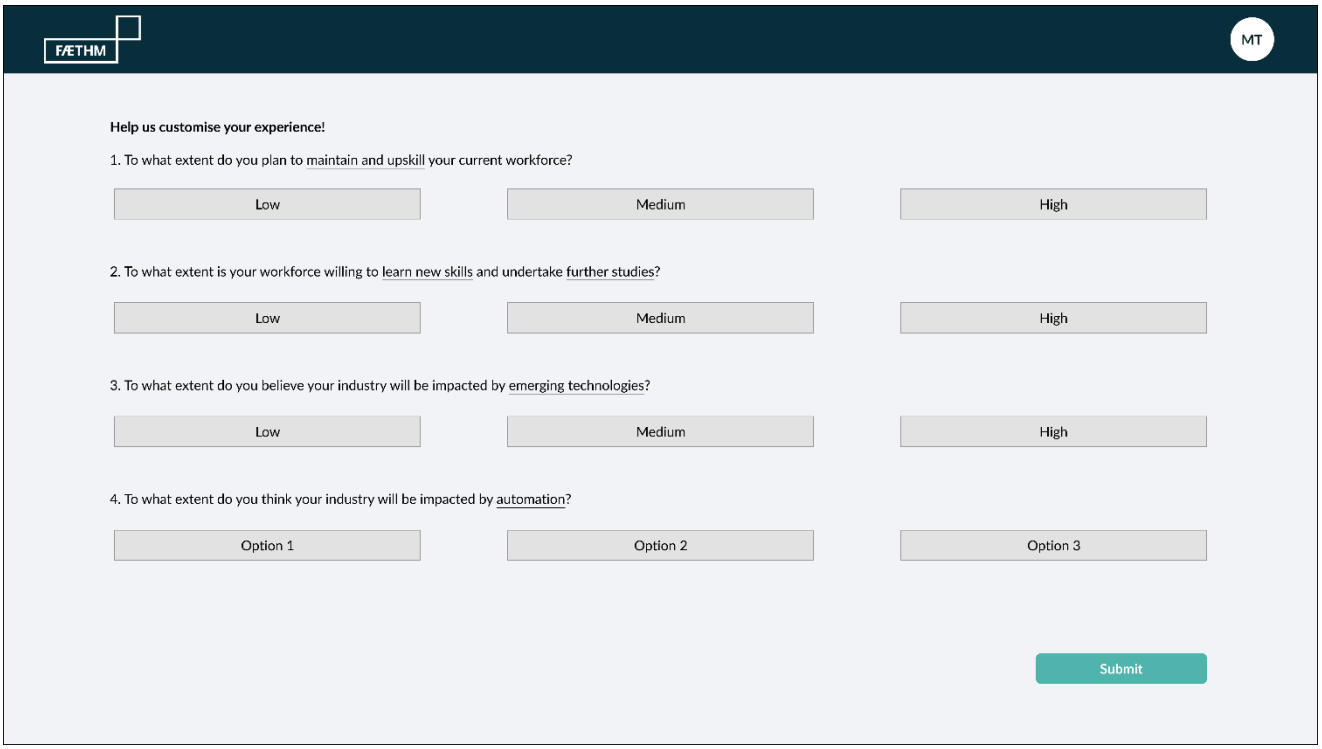
If a user is logging in for the first time, they will be presented with the Onboarding UI. The Onboarding UI includes a questionnaire that determines the category of employee a user falls under. The possible categories of employees are:

- Executive
- Head of Division
- Data Analyst
- Manager
- Employee
- Learning Manager
- Workforce Planner
- HR Manager
- Automation Engineer



In addition to categorising the user, further questions are asked to determine the user's use-case of the Faethm application to better understand and cater to their needs and requirements. While the exact questions have not been finalised, these are some that may be used in production:

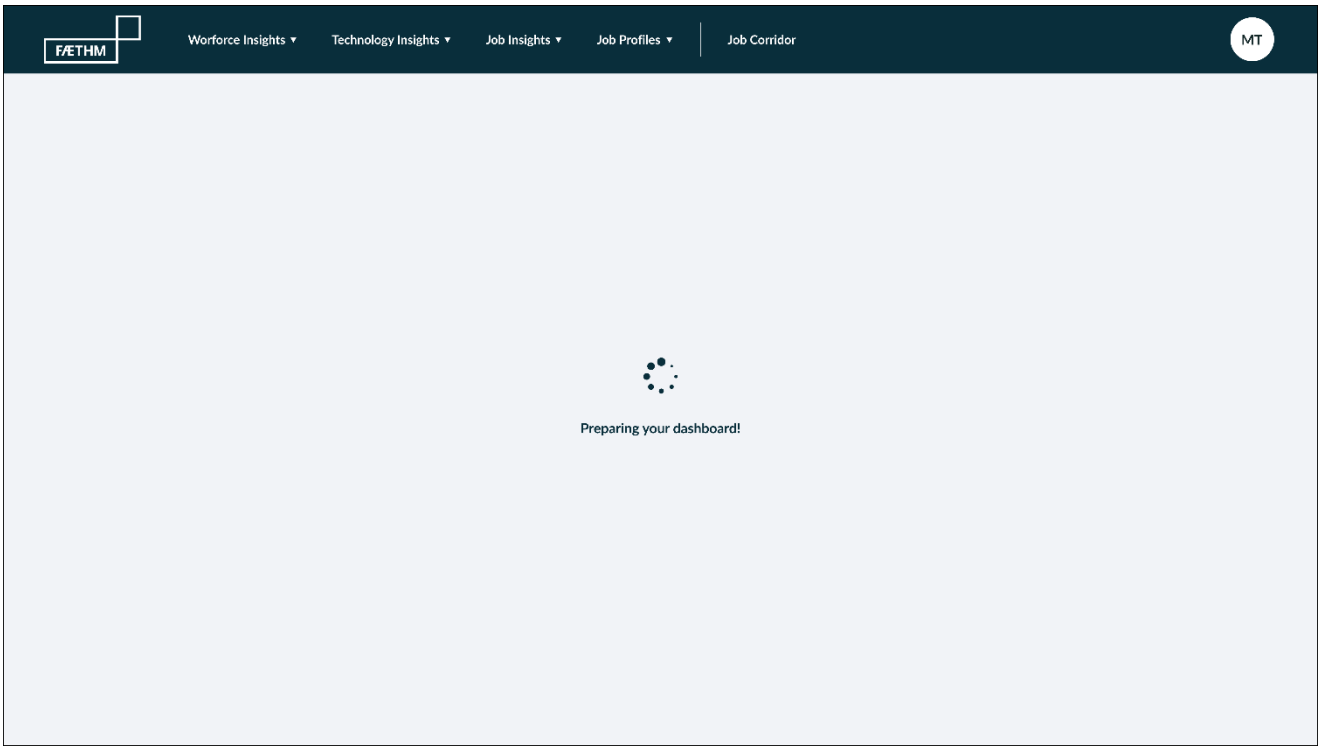
1. To what extent do you plan to maintain and upskill your current workforce?
2. To what extent is your workforce willing to learn new skills and undertake further studies?
3. To what extent do you believe your industry will be impacted by emerging technologies?
4. To what extent do you think your industry will be impacted by automation?



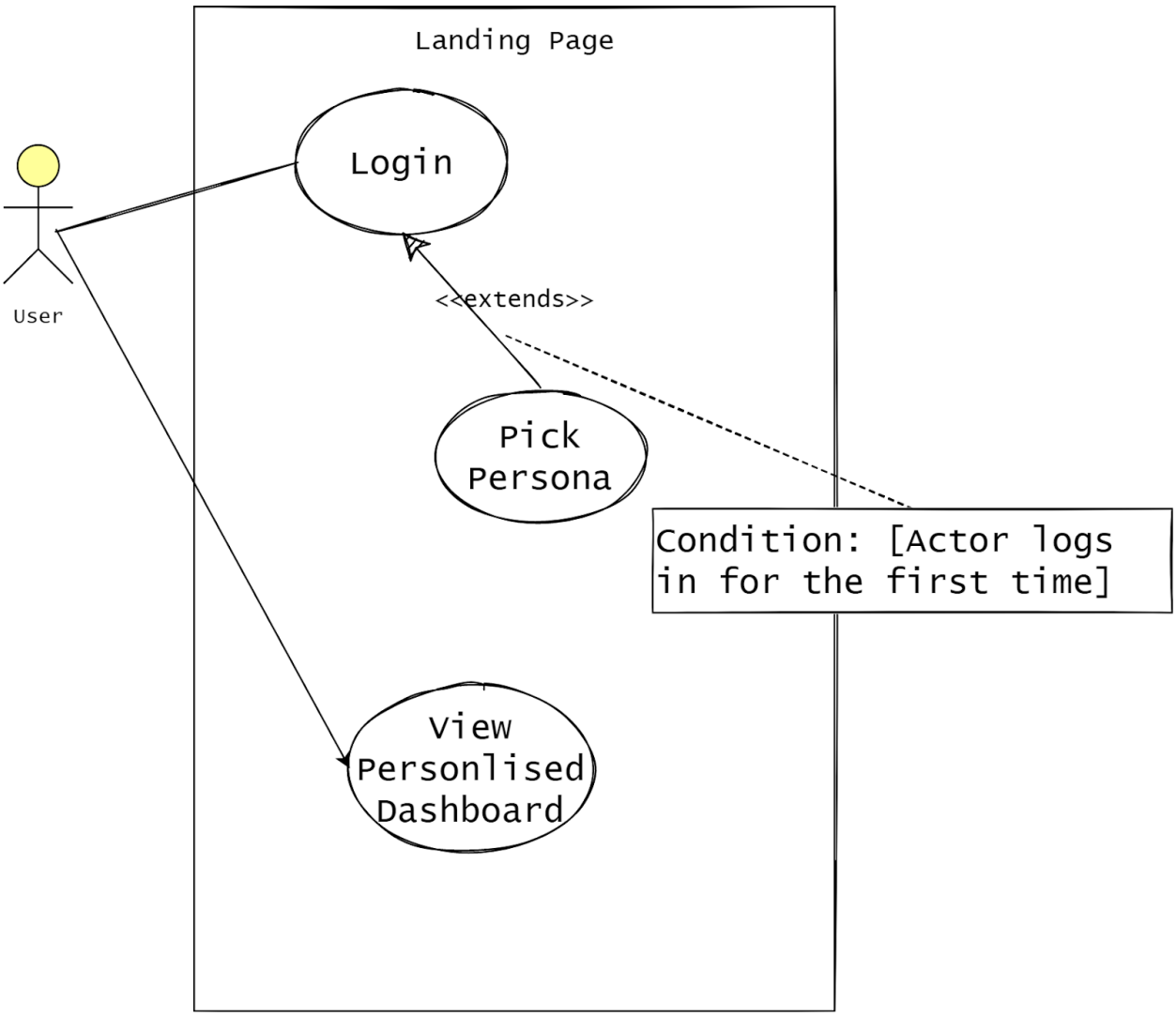
1.3 Loading Page

Once the user answers the questionnaire, their data is used to provide a personalised dashboard. Before the dashboard is provided to the user, an intermediate loading page is presented.

While the information being processed from the questionnaire is not large in size, as part of our design choice, the Onboarding UI will display a revolving loading icon. This delay is purposefully built-in to create anticipation and give the perception of added value to the user.



2. Use Case Diagram



3. Use Case Descriptions and User Stories

3.1 Logon

Use Case	LOGON
User Story	As a User, I want to login to the Faethm application.
Preconditions	The User has a valid email and password The CEO has internet access and a web browser.
Success End Condition	The User is able to login and access the Faethm application.
Failed End Condition	The User cannot login
Primary / Secondary Actors	Primary Actor - User Secondary Actor - Faethm Application
Trigger	The CEO launches the Faethm Application using their web browser
Description / Main Success Scenario	1. The Faethm application shows a landing page consisting of a login form. If the application is not available or fails to load, then refer to **Alternative Flow 1 "System Under Maintenance"**. 2. The user inputs their Email address and password. 3. The application authenticates the user if they entered valid credentials 4. If the application rejects the entered credentials, then refer to **Alternative Flow 2 "Invalid Credentials"**
Alternative Flows	1. System Under Maintenance a. Display an error message "System under maintenance". 2. Invalid Credentials a. Display an error message "Invalid Email or Password". b. Display option to reset password.

3.2 Pick Persona

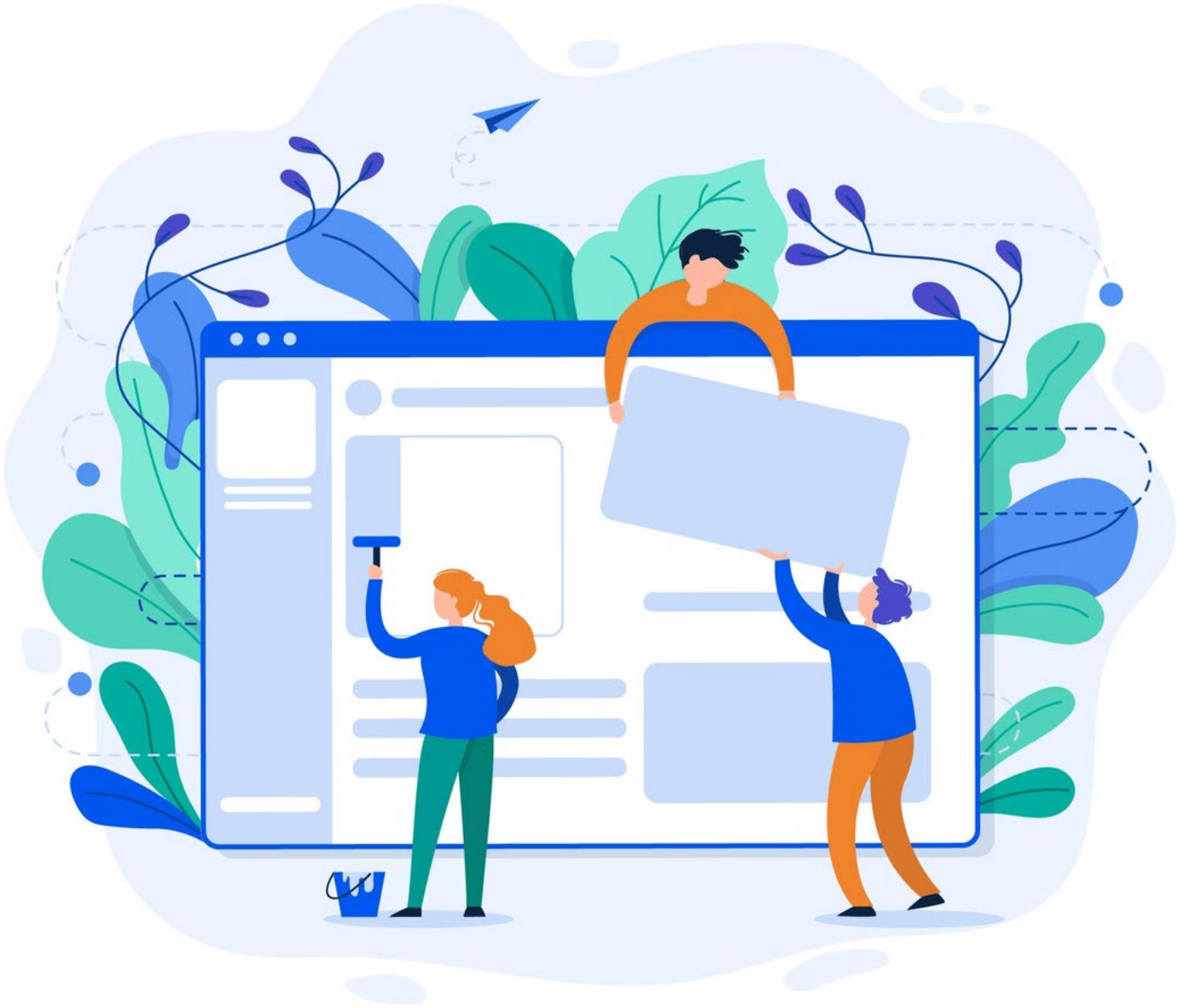
Use Case	Pick Persona
User Story	As a User, I want to check out the Faethm Application to see if it assists my business decisions in any way.
Preconditions	The User has a valid email and password The User has internet access and a web browser. The User is a new user.
Success End Condition	The User is able to pick a persona that matches their use case.
Failed End Condition	
Primary / Secondary Actors	Primary Actor - User Secondary Actor - Faethm Application
Trigger	The User launches the Faethm Application using their web browser
Description / Main Success Scenario	The User logs into the Faethm application. Refer to the use case description **LOGON** to view how a user logs in. The onboarding UI presents a questionnaire to determine the persona of the employee and the user's use-case. The User has to pick from the variety of personas presented.
Alternative Flows	

3.3 Personalised Dashboard

Use Case	View Personalised Dashboard
User Story	As a User, I want to review my company's numbers and make decisions in correspondence to achieve my goals and increase profit margins.
Preconditions	The User has a valid email and password The User has internet access and a web browser.
Success End Condition	The User views their personalized dashboard and access features provided by the application.
Failed End Condition	
Primary / Secondary Actors	Primary Actor - User Secondary Actor - Faethm Application
Trigger	The User launches the Faethm Application using their web browser
Description / Main Success Scenario	1. The User logs into the Faethm application. Refer to use case **LOGON** to view how a user logs in. 2. The User picks a persona if they are logging in for the first time 3. Review use case description 'Pick Persona' to see how persona is picked 4. The User will be presented with a loading page while the data from the questionnaire is processed. 5. After processing is complete, the User will be presented with select features of the Faethm application along with relevant articles, and statistics determined through the questionnaire.
Alternative Flows	

4. Assumptions

A1. We will be able to maintain contact with the representatives from Faethm for the duration of the project.
A2. Faethm will provide us with the appropriate resources to complete the project.
A3. All group members have access to appropriate hardware and software to complete the project.
A4. The project's requirements can be changed by Faethm at any point in the project if it is required.
A5. The project's requirements can be changed by Faethm at any point in the project if it is required.
A6. Access to the existing CSS styles to ensure a consistent UI with the rest of the app.
A7. While Faethm has the flexibility to accept a number of tech stacks, they ultimately prefer MERN (MongoDB, Express, React, Node). This report assumes that MERN will be used.
A8. The solution will be provided as a stand-alone single-page application (SPA) separate from the existing application.
A9. A laptop use case for most users has been assumed with mobile-friendliness, while ideal if achieved, as an added benefit.
A10. Access will not be granted to Faethm’s existing code base for security.
A11. Faethm will provide us with the assistance necessary to develop this solution.



Design Document

FÆTHM

Lance Te, Sepehr Torfeh Nejad, Marcus Ikeda, Erik Horvath, Rojwal Shrestha

COMP3850 Sem 1, 2022 Deliverable 3

Contents

1. System Design Document 2

1.1 System Architecture 2

1.2 Package Diagram 2

1.3 Storage/ Persistent Data Strategy 3

1.4 Concurrent Processes 3

1.5 User Interface Strategy 3

1.6 Design Choices and Trade-offs 3

2. User Interface Layouts 4

2.1 Login Page 4

2.2 User Category Page 4

2.3 Questionnaire Page 5

2.4 Loading Page 5

3. Window Navigation Diagram 6

4. Data Definitions / ER Diagram 7

5. Activity Diagrams 8

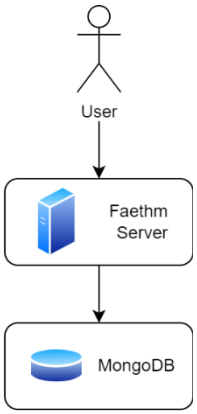
5.1 Login 8

5.2 Survey 8

5.3 Information Handling 9

1. System Design Document

1.1 System Architecture



The Onboarding UI has a simple, traditional system architecture. As it will sit in front of the current Faethm web application, it will be hosted on the Faethm servers.

When users access the Faethm app, they are served the login page where their identity is authenticated before being served the Onboarding UI. Each user's credentials are stored in the Faethm MongoDB.

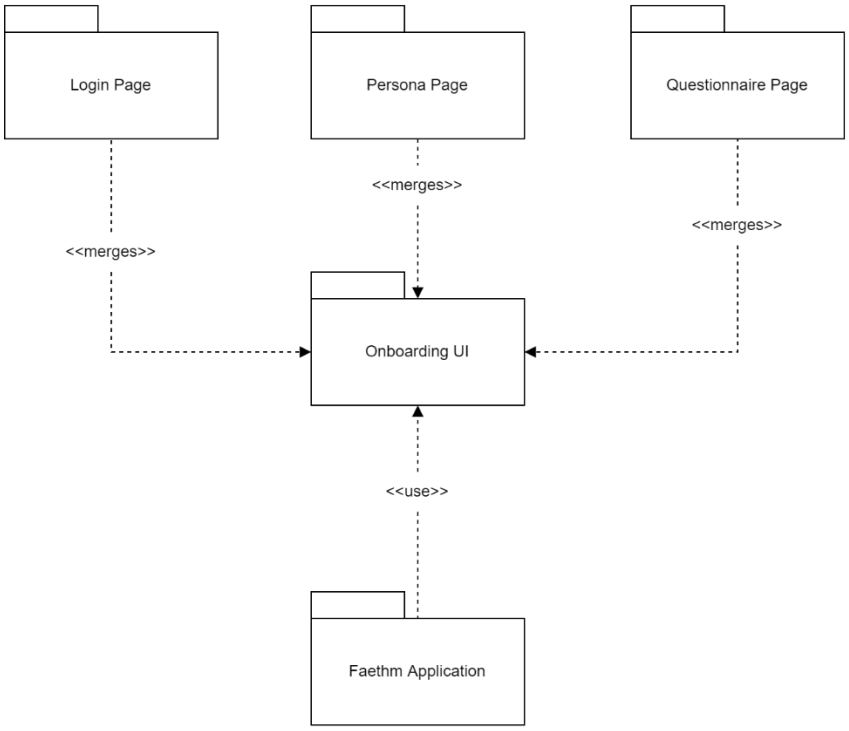
As users interact with the Onboarding UI, the information gathered will be sent back to the Faethm server and stored as part of their document record in the MongoDB.

1.2 Package Diagram

The package diagram shows the organisation and arrangement of the various models / elements / components of a software application in the form of packages. The Onboarding UI is the central model element that merges the:

- Login page - Processes user authentication
- Persona page - Determines the user's persona
- Questionnaire page - Determines the user's use case

Additionally, the remaining Faethm application uses the Onboarding UI as the first-time user experience (FTUE) before directing users to the primary application.



1.3 Storage/ Persistent Data Strategy

Our Onboarding UI collects and stores a range of persistent data that tracks user information. The initial user login credentials are stored upon creating an account and following this, the user is prompted to complete our onboarding UI. This process collects additional information about the user which is converted to data that is stored in the database. Our team has developed a process to manage the storage of this data and to strategically balance the durability of the data with the appropriate performance level when retrieving this data to ensure a seamless user experience.

We have decided to utilise the MongoDB cloud-based database application to both manage the database and to store the user data. MongoDB is an industry trusted SaaS application specialising in highly customisable document-oriented NoSQL database storage. MongoDB has pre-existing best practices and automates certain operations to ensure that data is kept secure, and the database runs smoothly. Cloud based storage means that it is fast to access anywhere as it reduces our hardware costs and running costs of on-premise servers and machines.

1.4 Concurrent Processes

The onboarding UI tool is a linear process that doesn't require any simultaneous processes. Multiple concurrent user connections are managed due to NodeJS being single threaded which means that it takes all simultaneous requests, but it will process each individually and one after the other. Multiple users accessing the database at the same time is managed automatically by MongoDB as it has been created to do so and is operating in a cloud environment.

1.5 User Interface Strategy

As our onboarding UI requires the user to interact with the application in order to gather information, we have developed a plan of action to improve the user experience and ensure efficient collection of data. Our application follows the JWS open standard to securely transmit information between the users and the application as well as utilising the Redux JavaScript library to manage the application state. The interactive GUI is created with a responsive CSS design that allows users to share their information by clicking on buttons once prompted.

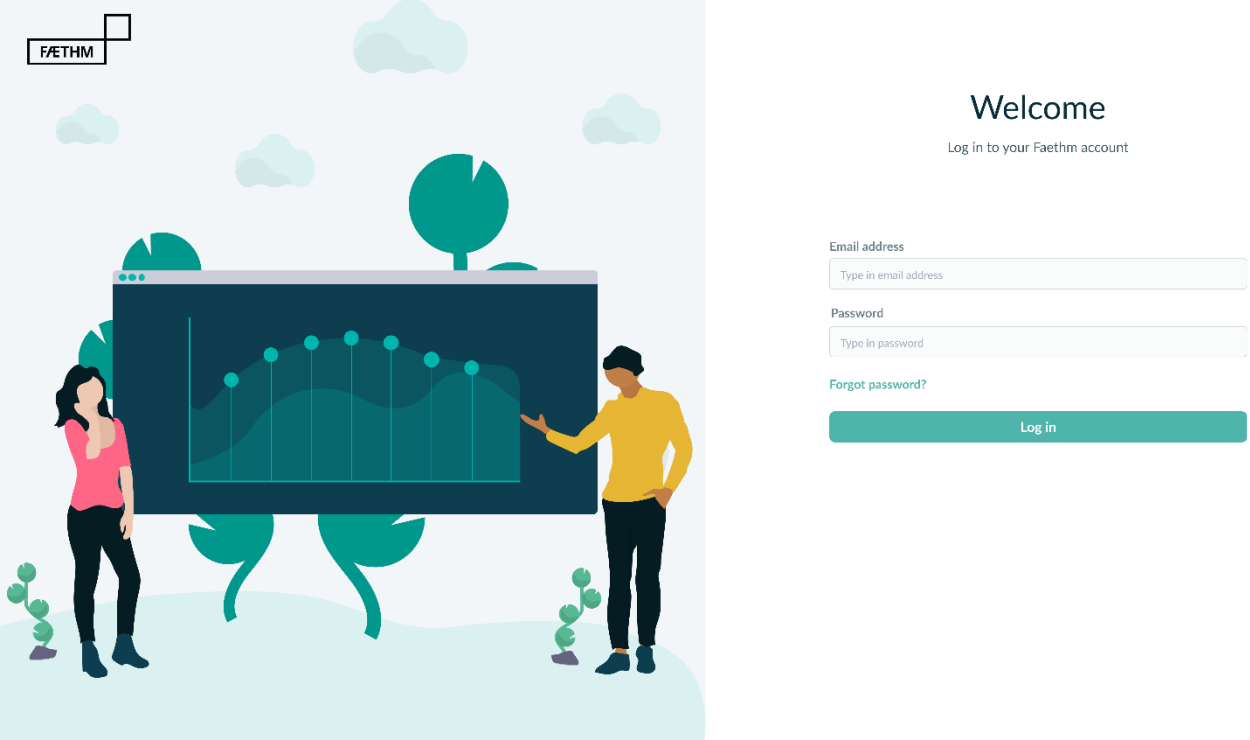
1.6 Design Choices and Trade-offs

Our application was designed using the MERN stack rather than a more traditional PHP + jQuery for easier implementation with existing Faethm AI application. Our colour schemes and icons are created to match the existing Faethm AI colour scheme and existing design format. The specific user categories were provided to us by the Faethm team and we built our application around this template. We have also added on an additional survey to further categorise the user which is designed to create a greater personalised experience to the user. The survey follows the previous pages design scheme and the landing page design scheme from the Faethm site.

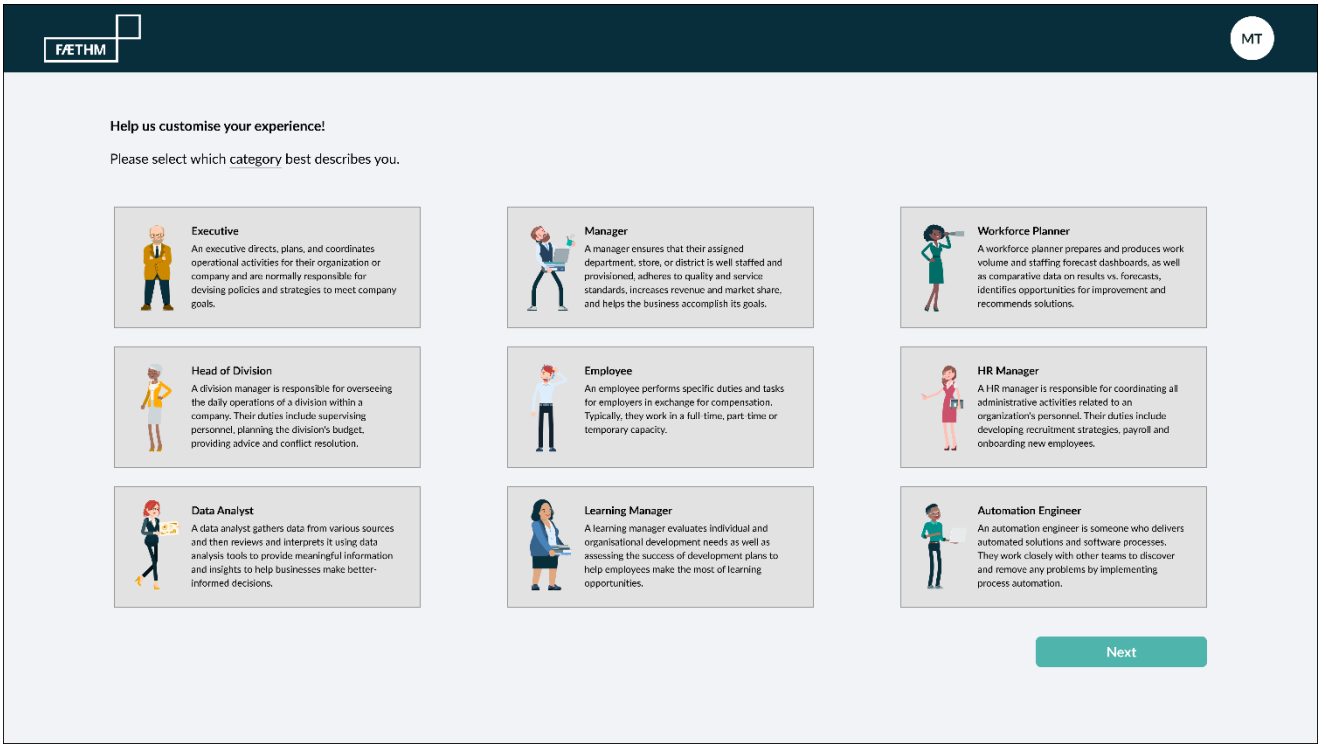
2. User Interface Layouts

The following images is the linear progression of the user Interface layouts as the user progresses through the onboarding process.

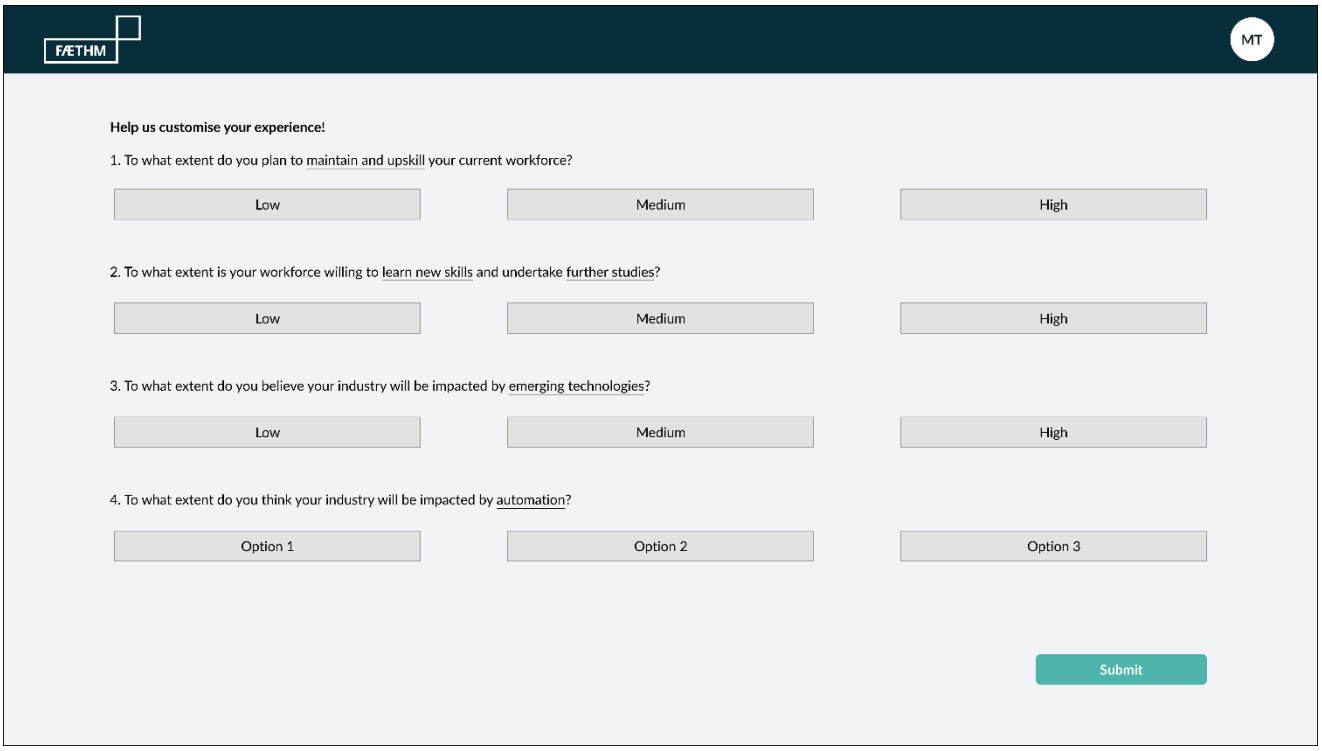
2.1 Login Page



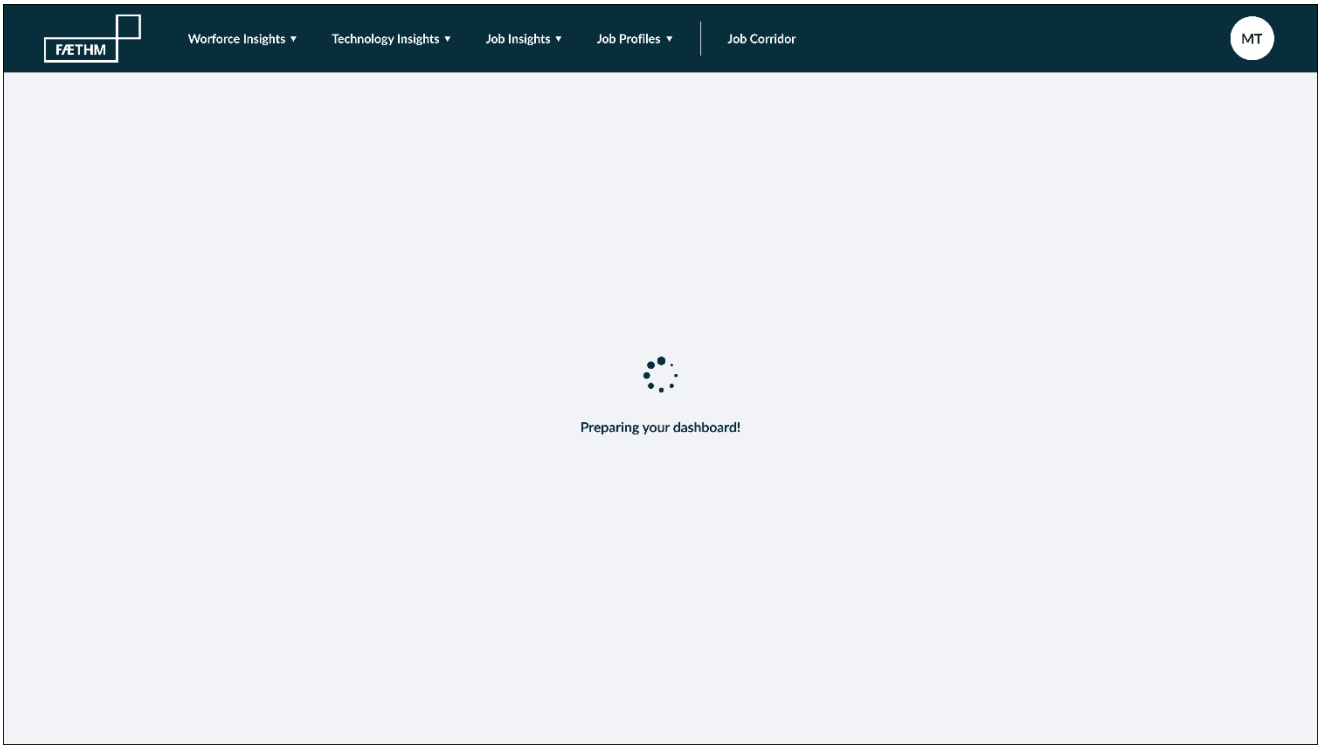
2.2 User Category Page



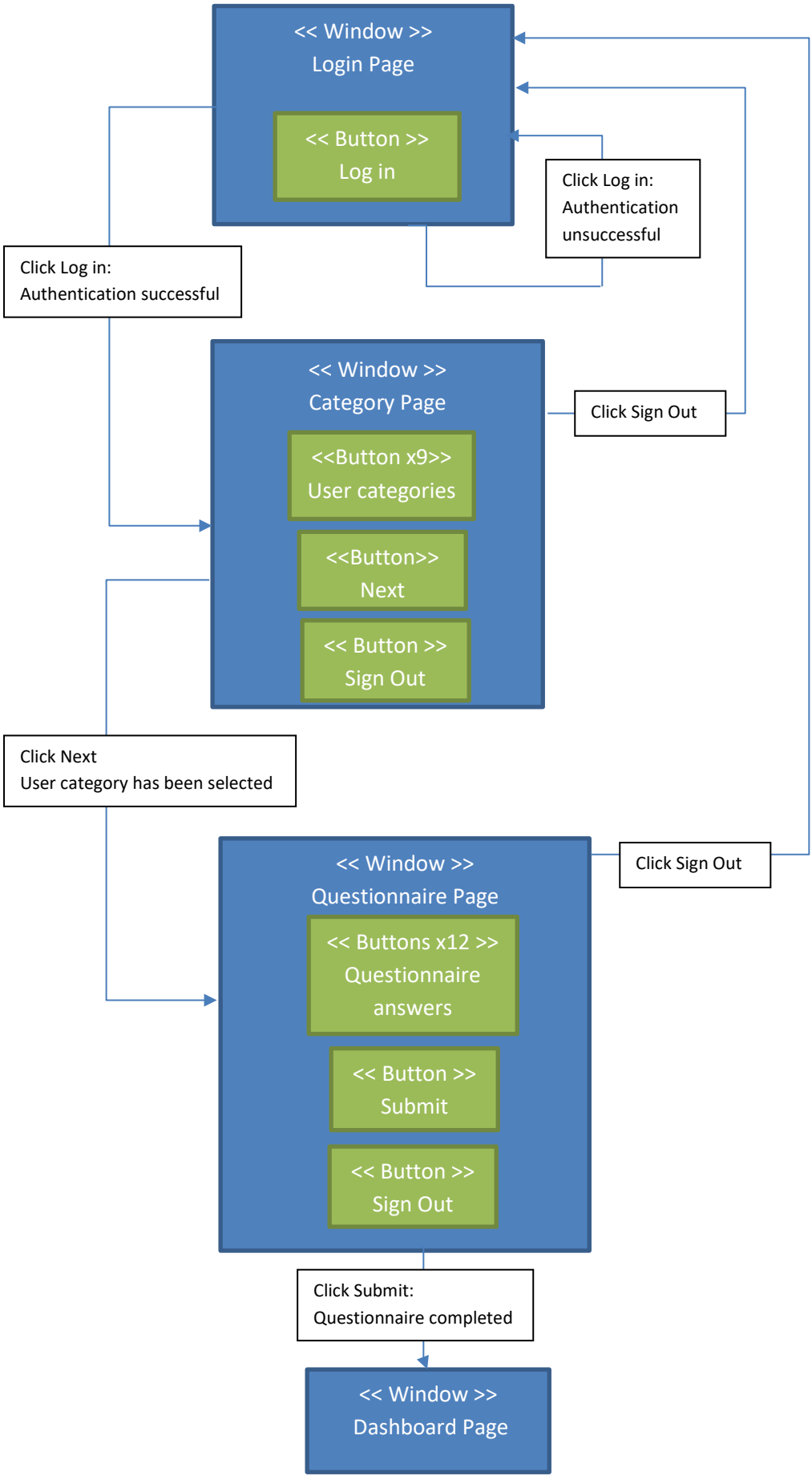
2.3 Questionnaire Page



2.4 Loading Page



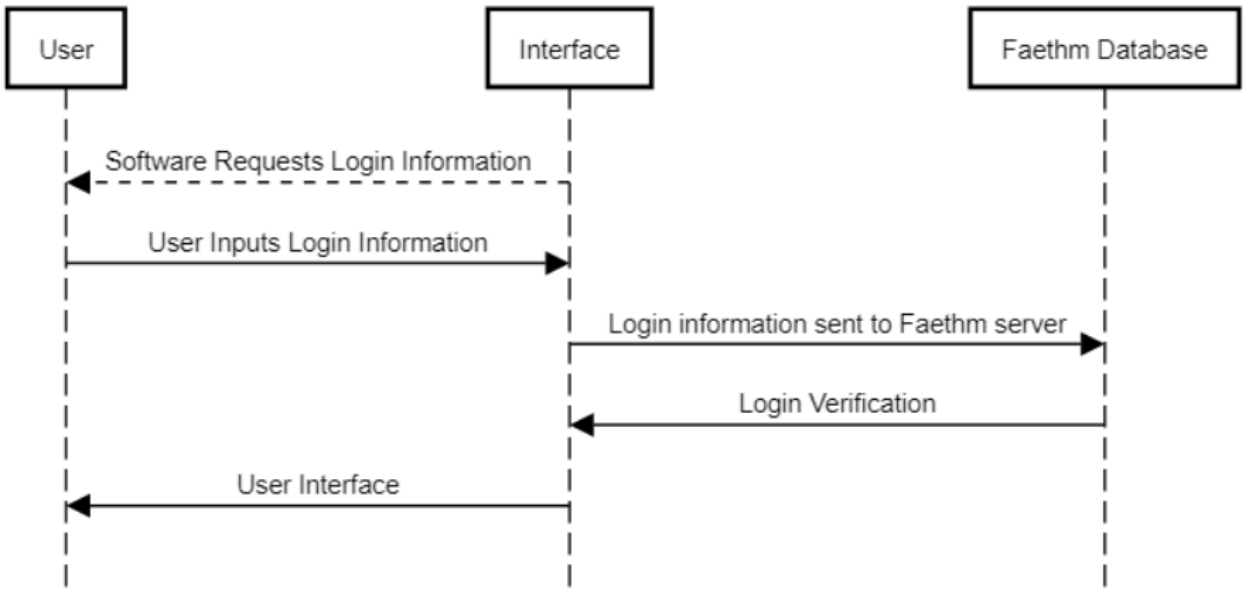
3. Window Navigation Diagram



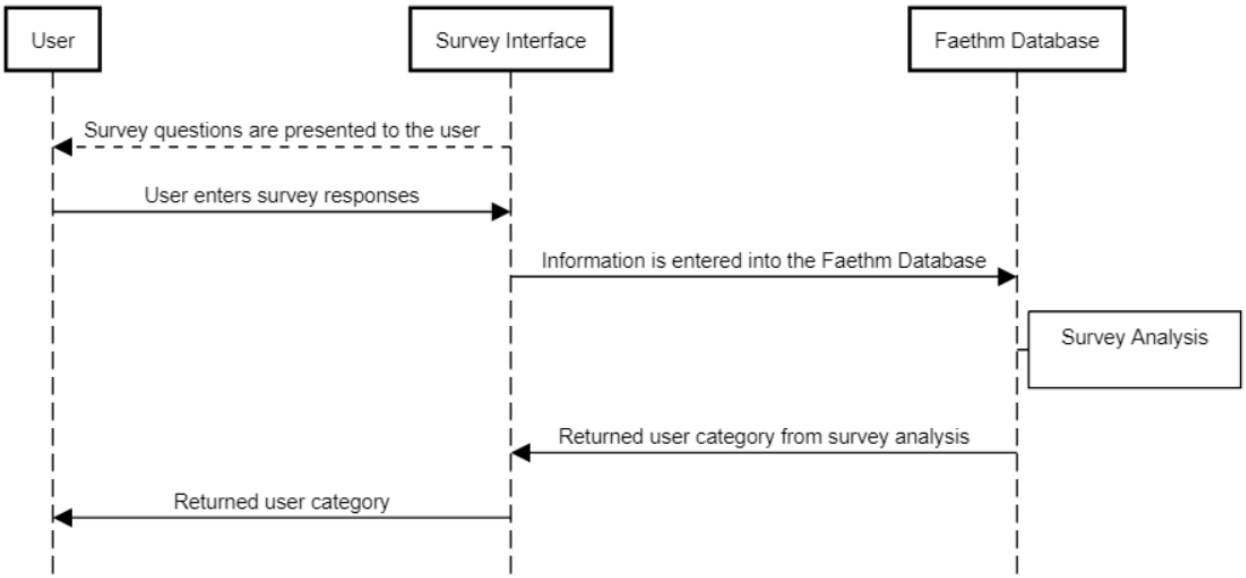
4. Data Definitions / ER Diagram

5. Activity Diagrams

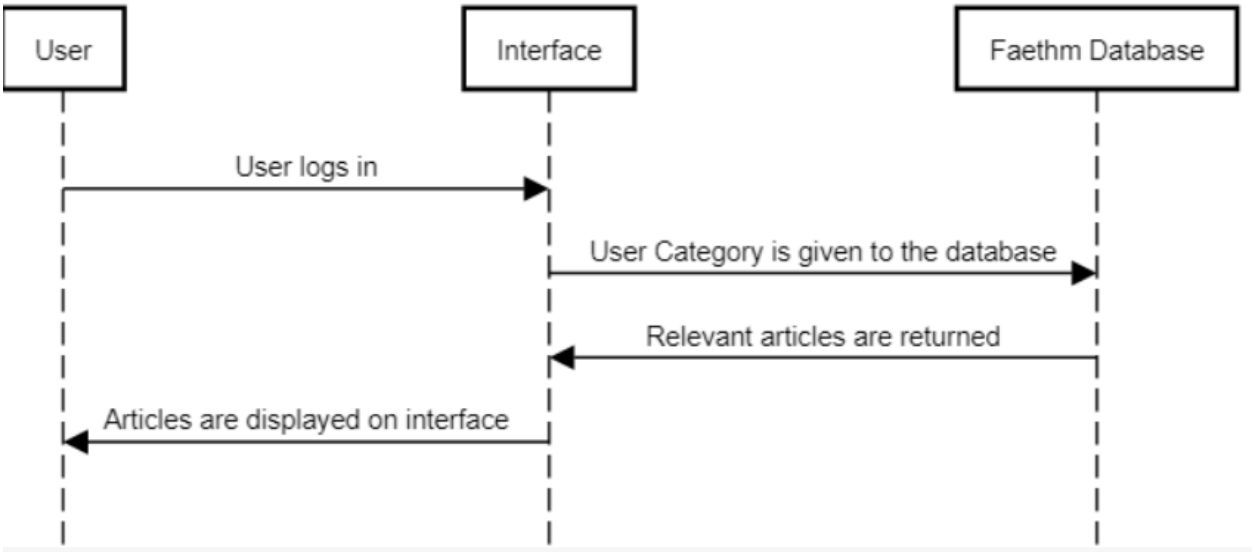
5.1 Login



5.2 Survey



5.3 Information Handling



Testing Document

FÆTHM

Lance Te, Sepehr Torfeh Nejad, Marcus Ikeda, Erik Horvath, Rojwal Shrestha

COMP3850 Sem 1, 2022 Deliverable 3

Contents

1. Test Plan 2

1.1 Testing Strategy 2

1.2 Test Types 2

1.3 Test Schedule 2

1.4 Test Tools and Resources 3

1.5 Test Deliverables 3

1.6 Test Resourcing 3

1.7 Test Milestones 3

2. Test Case Specifications 4

2.1 Login Functionality 4

2.2 Survey Functionality 6

3. Client Feedback 9

3.1 Meeting Date and Time 9

3.2 Feedback Received 9

3.3 Team Response 9

1. Test Plan

1.1 Testing Strategy

Our Onboarding UI is not hosted on the cloud, but instead hosted on GitHub and so testing will need to be carried out on each developer's local machine. Whenever a developer wants to test code, the updated version of the code base will need to be pulled and fetched from the remote repository. As bugs and errors are fixed, and new code is added, the developer will then push their local changes to the remote repository for other developers to perform further testing and verification. Every feature will be implemented on a separate branch in the remote repository to ensure the main code base remains stable at all times. Further, it mitigates the risk of errors and allows for easier debugging since any known errors following a merge with the main branch can be easily identified as coming from the feature branch.

When testing the Onboarding UI for the first time, developers need to clone the environment by"

1. Cloning the remote repository.
2. Using node package manager (npm) to install all dependencies on their local machine.
3. Setting up the .env file to define environment variables including port, MongoDB URL, and JWT secret.
4. Running the command npm run dev.

The developer will then be able interact with the Onboarding UI and perform the tests outlined in the Test Specification section below.

With the heavy user-interaction expected on the Onboarding UI, compared to its relatively low functionality, most tests will fall under black-box tests that are designed with the end-user in mind. White-box testing will also be performed on the back-end system to evaluate and examine the code; ultimately to verify the internal structures are working as intended.

1.2 Test Types

The test types will predominantly fall under system tests as the Onboarding UI will be providing the first-time user experience (FTUE) for the current Faethm web application.

However, as new features and functionalities are designed and implemented, the team will undergo rigorous regression testing to ensure that it works as intended and the existing features and functionalities are not broken.

Additionally, the Onboarding UI includes a login feature which requires penetration testing to ensure that users are authenticated and can only use features they are authorised to access.

1.3 Test Schedule

Given the agile methodology used throughout this project, work is delivered in small increments. Thus, the team does not have a specific test schedule. Instead, as the team implements new features and functionalities, testing will be done immediately after to ensure that it works as intended. This iterative, incremental approach means that our suite of test specifications will grow. These changes will be recorded and included for an updated Testing Document (Increment 2).

In the week before the presentation and handover of the Onboarding UI to Faethm on Thursday 2nd June 2022, the team will undergo a thorough test of the entire web application in accordance with the Testing Specifications.

1.4 Test Tools and Resources

There are a range of tools available for testing beyond those mentioned below. However, these are agreed minimum standards that all developers will adhere to. Any additional tools, browser extensions, IDE extensions, and other resources used is up to each developer's discretion.

- **Typescript:** While Typescript is not a replacement for tests, it is a good tool to catch errors and ensure static checking before reaching production.
- **React Dev Tools:** Provides the developer with the React component tree and an interface to inspect and interact with elements including components, states, and stores.
- **Jest:** Provides the developer with a JavaScript library designed specifically for React to create, run, and structure tests.
- **Seeder script:** Populates database tables/collections with test data.
- **npm run dev:** Runs the development environment locally for the developer to perform tests on the Onboarding UI.

1.5 Test Deliverables

The main test deliverable to the client is this Testing Document. However, as new features are implemented, the team will provide updated versions of this Testing Document.

1.6 Test Resourcing

The tests included in the Test Specification below outline which developers are responsible for which test(s).

1.7 Test Milestones

The discussion and addition of new features that results from the agile methodology will serve as a testing milestone. This involves implementing the feature and subsequently testing to ensure the feature works appropriately and interacts with the rest of the existing application cohesively.

The week before the presentation and handover of the Onboarding UI to Faethm on Thursday 2nd June 2022 remains the main test milestone the Onboarding UI team continues to work towards.

2. Test Case Specifications

2.1 Login Functionality

Test Case ID	LOGIN_001
Test Designed by	Sep
Test Reviewed by	Lance
Test Description	Test login page by verifying successful user login with valid username and password
Test Priority	Medium

Test Executed by	<Name>
Test Execution date	<Date>
Test Status	Not Executed
Test Scenario	Verify users can login successfully with valid credentials and JWT is created and returned upon successful user login and stored in local storage. Invalid user credentials will return an error and user can't login.
Preconditions	1. Tester has pulled the latest version from GitHub 2. Database is seeded using the seeder script 3. Application is running locally using npm run dev command 4. There is at least one valid user
Test Log	<Log>

Test Data		
ID	**Test Data Description**	**Test Data**
01	Valid username and password	Username: Sep@test.com Password: sep
02	Valid username with invalid password	Username: Sep@test.com Password: 123
03	Invalid username with valid password	Username: Sep.T@test.com Password: sep
04	Invalid username and password	Username: Sep.T@test.com Password: 123

Step	Test Steps	Expected Result	Actual Result	Status (P/F)
1	Navigate to http://localhost:3000/	Login page is rendered		
2	Enter Test_Data_01 Click Login button	1) Both username and password can be entered on the login form 2) User is successfully logged into the application and JWT is stored both on the database on local storage 3) User is directed to Onboarding UI page		
3	Clear local storage Navigate to http://localhost:3000/ again Enter Test_Data_02 Click Login button	1) Both username and password can be entered on the login form 2) User cannot login to the application 3) Error message “Invalid Email or Password” is displayed		
4	Enter Test_Data_03 Click Login button	1) Both username and password can be entered on the login form 2) User cannot login to the application 3) Error message “Invalid Email or Password” is displayed		
5	Enter Test_Data_04 Click Login button	1) Both username and password can be entered on the login form 2) User cannot login to the application 3) Error message “Invalid Email or Password” is displayed		

2.2 Survey Functionality

Test Case ID	QUESTION_001
Test Designed by	Sep
Test Reviewed by	Lance
Test Description	Test the questionnaire which user can choose a persona and answer more questions to determine most use case using a decision tree
Test Priority	High

Test Executed by	<Name>
Test Execution date	<Date>
Test Status	Not Executed
Test Scenario	User can choose a persona from the 9 predefined personas and answer some questions
Preconditions	1. Tester has pulled the latest version from GitHub 2. Database is seeded using the seeder script 3. Application is running locally using npm run dev command 4. There is at least one valid user 5. User is logging in for the first time (persona is set to None)
Test Log	<Log>

Step	Test Steps	Expected Result	Actual Result	Status (P/F)
1	Navigate to http://localhost:3000/	Login page is rendered		
2	Enter Test_Data_01 Click Login button	1) Both username and password can be entered on the login form 2) User is successfully logged into the application and JWT is stored both on the database on local storage 3) User is directed to Onboarding UI page		
3	Choose “Automation Engineer” Click Next button	1) Automation Engineer option is selected and highlighted 2) Survey page 2 is rendered with 4 questions		
4	Choose “Low” for all questions	1) 1) Selected values are highlighted		
5	Click Submit	1) User data is sent to backend 2) Loading page is rendered while new personalised landing page is being prepared		

2.3 Security Functionality

Test Case ID	SECURITY_001
Test Designed by	Sep
Test Reviewed by	Lance
Test Description	Testing protected routes on the application
Test Priority	Medium

Test Executed by	<Name>
Test Execution date	<Date>
Test Status	Not Executed
Test Scenario	Users can not access survey page without first being authenticated for security reasons. If a user tries to access http://localhost:3000/survey, they will be redirected to the login page.
Preconditions	1. Tester has pulled the latest version from GitHub 2. Application is running locally using npm run dev command
Test Log	<Log>

Step	**Test Steps**	**Expected Result**	**Actual Result**	**Status (P/F)**
1	Navigate to http://localhost:3000/	Login page is rendered		
2	Navigate to http://localhost:3000/survey Without logging into the application	1) Login page is rendered again 2) User cannot access survey page without authentication		

3. Client Feedback

3.1 Meeting Date and Time

We submitted our Figma MVP for review via Slack on Tuesday 26 April 2022, 3:00pm AEST and our code for review on Wednesday 27 April 2022. In addition, we will provide a live demonstration for the Faethm team in our weekly Wednesday meeting on 27 April 2022 from 4:00pm.

3.2 Feedback Received

In terms of functionality and general design, the Faethm team informed us that their expectations had been exceeded. One standout design choice was the purposefully built-in delay via a loading page to create anticipation and give the perception of added value to the user.

However, there was feedback in terms of more specific aspects of the design:

- Card background should be a grey from their Figma file.
- Less text on the persona cards perhaps limiting it to one sentence.
- Rounded elements for cards to make it easier on the eyes.
- Clearer instruction: singular or multiple selection.
- Utilise a larger contrast of font-sizes.
- Card flip animation.
- Use the font-family that the Faethm application uses.
- (Optional) landing page prior to the survey.

Overall, this is very encouraging and confirms that our vision of the Onboarding UI aligns with their vision.

3.3 Team Response

The team has taken the feedback on-board and discussed these desired changes in our post-client meeting. These changes will be implemented one at a time starting from the lowest hanging fruit, such as utilising a larger contrast of font sizes, and progressing to the more complex changes, such as a card flip animation.