

API Documentation and Integration App Project Closeout Report

SOC09109 - Group Project

Group 49:

Hughes, Conor

Rossi, Benjamin

Sami, Mansour

Stankiewicz, Pawel

Taft, Luke (PM)

Table of Contents

1. Introduction	3
1.1 Client Brief: Coastr.....	3
1.2 Project Brief	3
2. Project End State	4
2.1 Initial Project Deliverables	5
Figure 1: Initial MoSCoW Requirements Prioritization	5
Figure 2: Project Initiation Report: Deliverables Map	6
2.2 Final Project Deliverables	6
Figure 3: Project Closeout MoSCoW Requirements	7
Figure 4: Coastr "Reservation API" Documentation	8
Figure 5: Coastr "Booking API" Functions	8
Figure 6: Coastr API Documentation Landing Page	9
3. Conclusion and Further Development	10
Figure 6: Voyager: Logical Topology Exploration.....	11
4. Appendices	12
Appendix I: Client Approval	12
Appendix II: Stakeholder Satisfaction Report.....	13
Appendix III: Lessons Learned	14
Appendix IV: Peer Review	15

1. Introduction

1.1 Client Brief: Coastr

Coastr is a technology development company that works in the transportation sector, developing solutions and improvements for the car and van rental process. They maintain a fully digital vehicle rental system with the aim of creating a seamless vehicle rental process for their customers.

Coastr aims to bring a new age of convenience and competition to the vehicle rental arena, giving customers access to competitive rates and exemplary service by creating an all-in-one platform that allows for digital control of vehicle fleets and business operations. These tools would allow Coastr customers to have a greater level of control over their vehicles while also receiving feedback and metrics for their business via advanced AI and telematics technologies.

1.2 Project Brief

The goal in the API Documentation and Integration App project (ADIA) is to build a client-facing app that serves as a central hub for Coastr customers integrating with their API. The app will provide clear visuals of API functions and thorough documentation, simplifying integration and decreasing the need for technical assistance. The app will assist customers in understanding, integrating, and using the Coastr API efficiently.

Anticipated advantages include enhanced customer integration experience, increased understanding of API features and limitations, and reduced need for direct technical assistance.

The cost and duration of the project have been estimated and are based on factors such as project scope, team capability, and the quality and availability of 3rd-party solutions. To identify relevant parties a dynamic stakeholder list will be maintained through the lifecycle of the project. The stakeholders, including the Coastr-appointed project representative, Customer Management team lead, development team members, and project manager, will identify and agree on requirements and quality expectations.

The objective of the ADIA project is to improve the customer experience and support customer success by providing a comprehensive, centralized, and user-friendly web application for accessing API documentation and integrating with the Coastr API. This is expected to increase

customer satisfaction, reduce the need for technical support, while also boosting efficiency in the integration process.

2. Project End State

The project closeout report aims to provide a comprehensive analysis of the project's end state, including an evaluation of its success against the initial expectations outlined in the Project Initiation Report (PIR). This assessment will involve a comparison of the project's objectives as defined in the PIR with the final product delivered to the client. By comparing the initial goals with the project's end state, the report aims to provide a clear indication of the project's success and the extent to which it has met the client's requirements.

In addition to the assessment of the project's deliverables, the closeout report will also evaluate stakeholder satisfaction. This will involve a review of feedback received from clients, project team members, and any other stakeholders involved in the project. By analysing stakeholder satisfaction, the report aims to provide insights into the effectiveness of the project's management and its impact on stakeholders.

Furthermore, the closeout report will include a detailed analysis of lessons learned throughout the project's lifecycle. This will involve identifying any challenges or areas for improvement, as well as highlighting any best practices or successful strategies employed during the project. By sharing these insights, the report aims to provide recommendations for future projects and to support ongoing improvements to project management practices.

Finally, a peer review will be conducted to assess the accuracy and completeness of the project closeout report. The review will involve an assessment of the report's methodology, findings, and conclusions, and will provide an opportunity to identify any gaps or areas for improvement.

This project report will provide a thorough analysis of the project's end state. This will include comparing the final product with initial deliverables, as well as evaluating stakeholder satisfaction, lessons learned and a peer review. These resources will provide guidance to both the client and the project development team by illuminating the factors that lead to project success and recognizing areas of improvement.

2.1 Initial Project Deliverables

The API Documentation and Integration App (ADIA) project aims to create a web application that provides Coastr customers with a centralized platform for accessing information regarding the use of the Coastr API. The application should be compatible with Coastr's backend, easily accessible to customers, provide visualizations of the API's functions, guidance on best use practices and integration techniques, and offer an extensible solution for future adaptations of the current API. Further details of prioritized project requirements can be seen in figure 1.

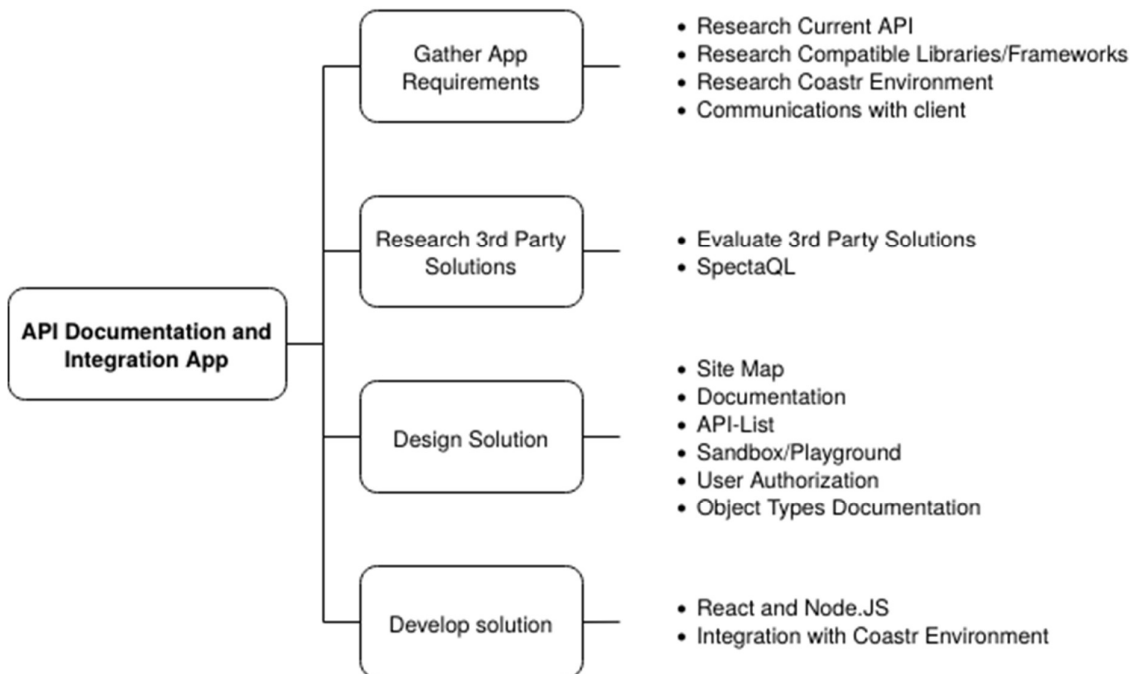
Figure 1: Initial MoSCoW Requirements Prioritization

Classification	Requirement
Must have	1. The app must be compatible with different operating systems and popular web browsers, while working seamlessly with Coastr's API
	2. The app must provide comprehensive documentation of integration with Coastr's API, including clear diagrams and illustrations of the various functions within the API
Should have	1. The app should be user-friendly, intuitive to the customer, and visually appealing
	2. The app should provide clear instructions for the customer as well as access to a knowledge base and customer support
Could have	1. The app could have security/authorization to authenticate customers and protect sensitive information regarding Coastr, their customers and their API
Will not have	1. The app will not be maintained by the university development team after its delivery to Coastr for use and maintenance in the Coastr production environment

The main purpose of the project is to streamline the process of integrating new customers into the Coastr API. This has the added benefit of reducing the amount of technical support required. The application will act as a repository of information regarding the use of the API as well as best use practices and demonstrations of the API's features. This is expected to increase customer satisfaction and provide valuable documentation to be used by both customers and Coastr staff.

The project deliverables will include the creation of a web application that meets the requirements identified through the MoSCoW process. Further details about these project deliverables are displayed in figure 2. The solution should contain "Must Have" requirements, be designed to be extensible for future adaptation, and be accompanied by quality documentation of the development process. The development team will maintain regular contact with the client to ensure the project is developing to meet their needs.

Figure 2: Project Initiation Report: Deliverables Map



2.2 Final Project Deliverables

The final project deliverables are a key metric used to evaluate the product delivered at the project's cessation. To ensure this product meets the prioritized requirements agreed upon by both parties a comparative analysis must be conducted. The primary goal of this analysis is to compare the product delivered to Coastr with the project requirements agreed upon during the project initiation stage. This critical evaluation is imperative to verify the final product suitability towards the client's expectations and requirements.

The critical analysis of the product will examine various aspects, including functionality, design, and usability, in relation to the project requirements. The analysis will also evaluate the product's quality and the level of client satisfaction to determine the project's overall success. Overall, this analysis of the product delivered to Coastr will provide valuable insights into the effectiveness of the project management processes and the project team's ability to meet the client's expectations and requirements. These insights will enable the identification of areas for improvement in future projects and emphasize the importance of prioritizing client satisfaction.

To best prioritize and measure the finished product, requirements were categorized as “must have”, “should have”, “could have” and “won’t have” requirements. The most important of these being the “must have” requirements. Without meeting these requirements, the project has failed to

API Documentation and Integration App | Group 49

deliver a minimum viable product. Please refer to figure 3 for the project closeout MoSCoW requirements and their status at the closure of the project. The first prioritized project requirement is that the product must be “compatible with different operating systems and popular web browsers, while working seamlessly with Coastr's API.” This was accomplished by building the app using a Node.js and React environment. Using this framework to generate the HTML for this web application ensured that the end-product would have compatibility with different operating systems and would be supported by popular web browsers. As an added benefit the product is compatible with mobile platforms via dynamic scaling. In its current state it lacks a navigation bar dropdown function that could impede mobile users, but further development of the app will be discussed in a later section. The product delivered meets and exceeds the requirement by providing a compatible and extensible platform.

Figure 3: Project Closeout MoSCoW Requirements

Classification	Requirement	Status
Must have	1. The app must be compatible with different operating systems and popular web browsers, while working seamlessly with Coastr's API	Completed
	2. The app must provide comprehensive documentation of integration with Coastr's API, including clear diagrams and illustrations of the various functions within the API	Completed
Should have	1. The app should be user-friendly, intuitive to the customer, and visually appealing	Completed
	2. The app should provide clear instructions for the customer as well as access to a knowledge base and customer support	Completed
Could have	1. The app could have security/authorization to authenticate customers and protect sensitive information regarding Coastr, their customers and their API	Relegated to Client
Will not have	1. The app will not be maintained by the university development team after its delivery to Coastr for use and maintenance in the Coastr production environment	Relegated to Client

The second prioritized project requirement states that the product must “provide comprehensive documentation of integration with Coastr's API, including clear diagrams and illustrations of the various functions within the API.” Figure 3 is a screenshot from the development environment where the web application was produced. It clearly displays the documentation generated for the “Reservation” API run by the client. The placeholder text in the image displays where individual API descriptions will be added once the application is handed over to the clients.

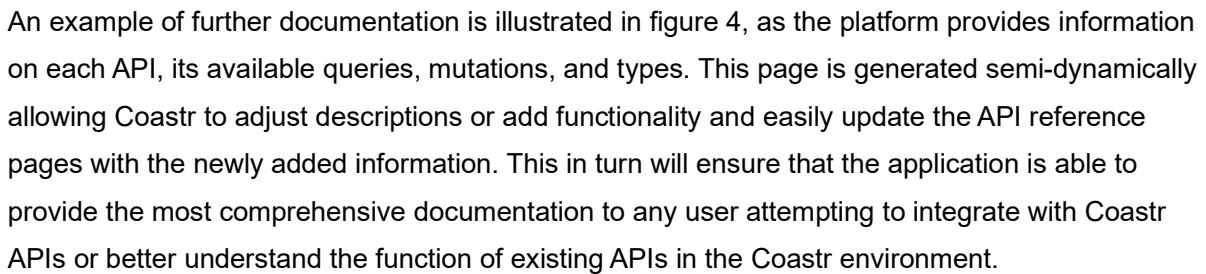
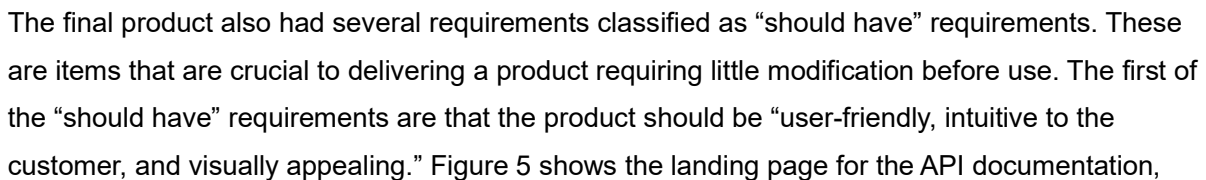
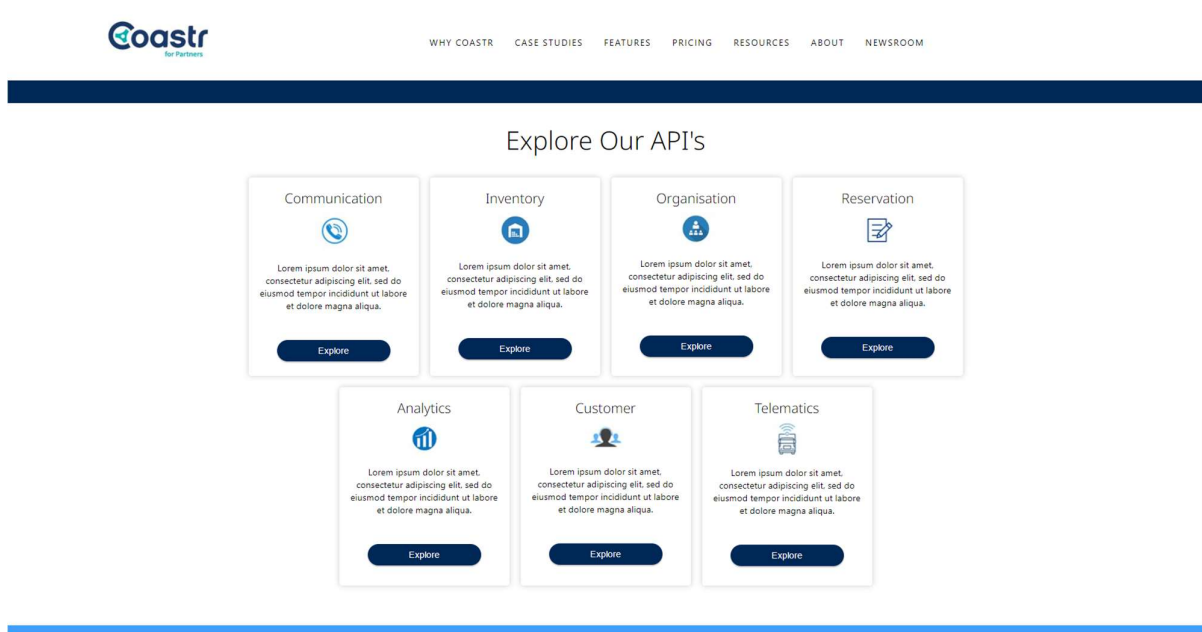


Figure 5: Coastr “Booking API” Functions



while also illustrating how the final product meets this requirement. From this landing page a user is free to explore each API with visual representations of the functions therein and how the APIs interact with each other. This is accomplished while also maintaining a clean, professional layout that uses the colours and logos advised in the Coastr branding guideline documentation.

Figure 6: Coastr API Documentation Landing Page



The second and last “should have” requirement states that the product should “provide clear instructions for the customer as well as access to a knowledge base and customer support.” This requirement is illustrated best under figure 4. The ability to browse each API’s functions, queries and mutations with the included descriptive text, provides instructions on the best use of Coastr APIs as well as acting as an in-depth knowledge base. While there is no direct link to customer support built into the product, it will have the standard Coastr footer once the application is running in the Coastr production environment. This will add further Coastr branding as well as pertinent company information and contact details.

The next section of prioritized requirements is “could have” items. In the case of this project the requirements states “The app could have security/authorization to authenticate customers and protect sensitive information regarding Coastr, their customers and their API.” This requirement was not met, due to time constraints. Security/authorization is a feature that will be developed by the client after the project’s conclusion and hand off.

Finally, the last class of requirement is items that the project will not have. In the case of this project the “won’t have” requirement maintains that “The app will not be maintained by the university development team after its delivery to Coastr for use and maintenance in the Coastr production environment.” This requirement remains unchanged, and the university team will not be

involved in the maintenance or further development of the web application once it has been handed off to the client: Coastr.

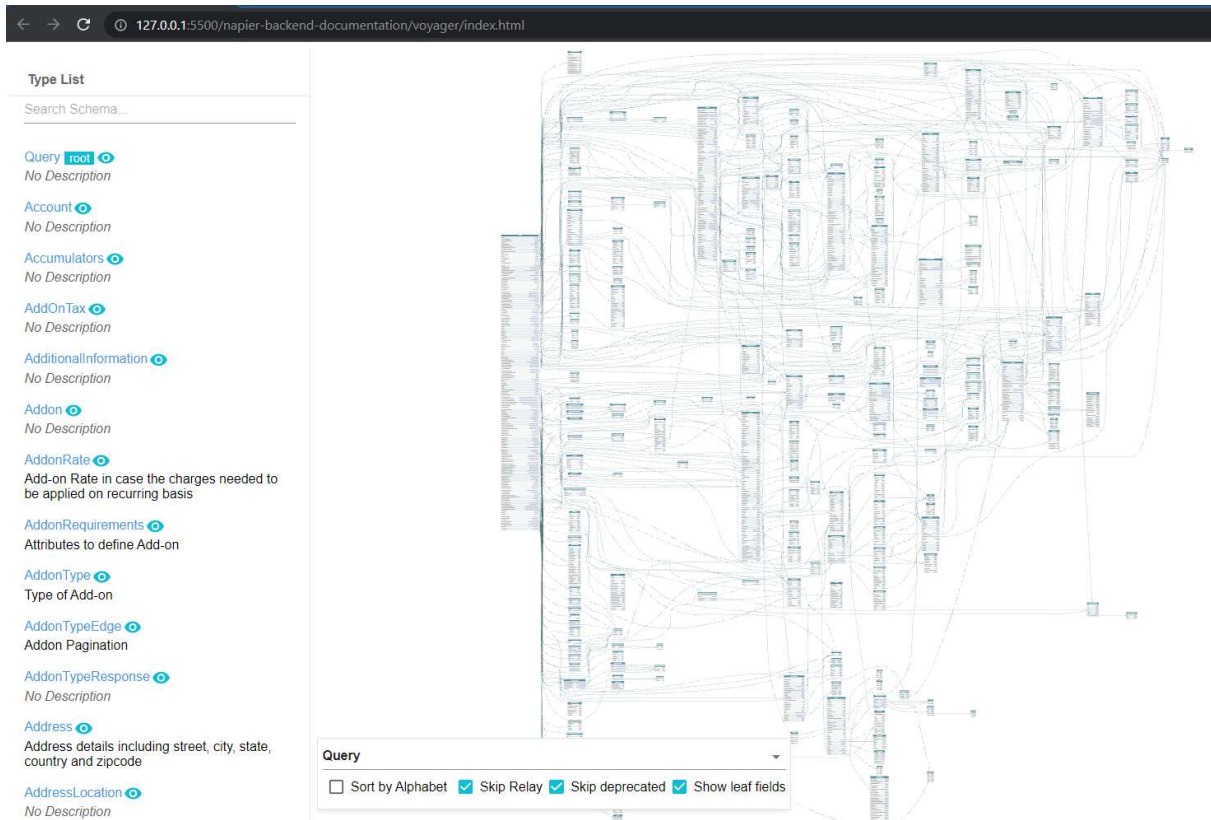
In summary, the ADIA project has been able to produce a product that meets its minimum viable product requirements. While the project was unable to achieve every requirement, the most important requirements were met and, in some cases, exceeded.

3. Conclusion and Further Development

In summary, the project closeout report presents a comprehensive analysis of the API Documentation and Integration App (ADIA) project's end state. This report has given a comparative analysis between the initial goals set in the Project Initiation Report (PIR), with the final product produced. Additionally, it provides a summarized stakeholder satisfaction rating, as well as an in-depth examination of lessons learned through the project's lifecycle. Moreover, this document will include a record of peer review to ensure the report's accuracy and completeness. This report provides valuable guidance to both the client and the development team by emphasizing the factors that led to the project's success and pinpointing areas that require improvement. The final project deliverables were assessed based on the prioritized requirements, with a particular focus on the "must have" requirements. The project's success is determined by its ability to meet the client's expectations and requirements while prioritizing client satisfaction. Ultimately, the ADIA project successfully delivered a high-quality web application that meets the client's needs and provides valuable documentation for both customers and Coastr staff.

The following paragraph will outline some further extensibility for the API documentation app. As was mentioned earlier in the project review, security/authentication for accessing the API documentation will be implemented by the Coastr security team. While it was a "could have" objective for the project, it ultimately fell outside the scope of the team's ability and time frame for development. Specifically, the addition of security would require the client to decide their security methodology with respect to where the application would be hosted. This process would fill the timeframe of the project on its own. In addition to this feature, the development team also looked briefly into a visualization of the logical topology of the Coastr APIs. This program, GraphQL Voyager (<https://github.com/IvanGoncharov/graphql-voyager>) works as a plugin for GraphQL APIs, allowing for a interactive graph. This plugin is easily added to the existing framework but was ultimately abandoned by the development team to produce a more focused product, delivered on time. Referring to figure 6 shows an early test deployment of the Voyager plugin.

Figure 7: Voyager: Logical Topology Exploration



Appendix I: Client Approval

Client sign-off

ADIA Project

Materials checked against PMIS

- API Documentation and Integration App

I confirm that the results from the project have been completed and delivered as agreed.

Signed:

Francis de Lima

Date:

21st April 2023

Appendix II: Stakeholder Satisfaction Report

Averaged Team Satisfaction		Feb	Mar	Apr
Values for totals are between 0 and 60 Values for Individual elements are between 0 and 10	Totals:	39.8	40.5	44.5
	Work load:	7.4	6.75	7.75
	Team Communication:	8.8	7.75	9.5
	Client Communication:	6.6	8.25	8.25
	Technical Progress:	7.2	8.25	9
	Teamwork:	8.2	7.5	8
	University Support:	1.6	2	2
Client Satisfaction		Feb	Mar	Apr
Values for totals are between 0 and 60 Values for Individual elements are between 0 and 10	Totals:	53	55	53
	Team Communications	9	9	9
	Technical Progress:	8	9	9
	Feedback Incorporation:	10	10	10
	Development Clarity	9	9	9
	Professionalism:	9	9	8
	State of the Project:	8	9	8

Client comments - February: I think the communication first month starting to find the balance and Luke has been great PM and also good at communicating any issues early and in good time. The technical team are also good at identifying which problems they can solve and which they need further information on e.g. lacking env variables. The next month will be interesting, as the development will have started

Client comments - March, April: N/A

API Documentation and Integration App | Group 49

Appendix III: Lessons Learned

Type	Cause	Effect	Im- pact	Likeli- hood	Im- portance	Re- sponse Type	Re- sponse	Custodian	Dates Occur- ring	Closure Outcome
R	<i>Lack of Clear Communication</i>	Without clear communication and coordination among team members, tasks can overlap, leading to confusion and decreased productivity.	80	30	2	Avoid	Miti- gate	PM	Ongoing	Efforts to refine commu- nications have been con- tinuous throughout the project
R	<i>Inconsistent Cod- ing Standards</i>	Different coding styles can lead to difficulties in integrating work and maintaining a consistent code- base.	50	30	2	Avoid	Testing	CH,BR,PS,MS	N/A	
R	<i>Difficulties using Node.js</i>	Node.js is a complex framework, team members may encounter technical difficulties while devel- oping the web app.	30	80	2	Avoid	Train- ing	CH,BR,PS,MS	Feb	Difficulties integrating into environment ex- panded deadline expec- tations
R	<i>Missed Deadlines</i>	If team members don't prioritize the project or if unexpected events arise, deadlines may not be met, leading to delays and added stress.	80	20	1	Avoid	Priori- tize	PM	10-Feb	Daily team standup- meetings to maximize work collaboration when required
R	<i>Version Control Conflicts</i>	When working on the same code- base conflicts can arise if versions do not match and changes are not properly managed.	60	10	3	Avoid	Miti- gate	CH,BR,PS,MS	N/A	
R	<i>Inadequate Docu- mentation</i>	Without proper documentation it can be difficult for team members to understand the purpose of dif- ferent parts of the code or how they should be modified.	80	30	2	Avoid	Miti- gate	CH,BR,PS,MS	Feb-Apr	Inadequate environ- ment documentation re- quired additional trou- bleshooting

Appendix IV: Project Closeout Peer Review

Reviewer: John Johnston **Team:** 13

Reviewee: Luke Taft **Team:** 49

Date of review: 21/04/2023

Project description

Reviewer's comments and recommendations

The report is clear about the nature of the project and the application's intended purpose. Having read the report from top to bottom, however, it is very much written with the intention of the project as opposed to what has been achieved. A direct comparison of the intended and agreed deliverables to the achieved outcomes is not easily arrived at without reading the entire report. The use of two MoSCoW tables, before and after, will solve this quickly and easily.

Response and actions taken:

A second figure containing a follow up MoSCoW table was added to allow the reader to quickly and easily ascertain the status of individual project requirements.

Deliverables map

Reviewer's comments and recommendations:

The original deliverables map is not present; therefore, no comparison can be made with the presented information.

Response and actions taken:

Deliverables map added to allow for comparison between presented information and visualized deliverables.

Follow-up register

Reviewer's comments and recommendations:

The follow-up register is apparent and is fully completed. However, there are no amendments listed where coding or strategies were changed. EG in our own project-specific technologies were started but changed due to ease of programming or compatibility issues. I am not saying you had them, but if individuals made changes, they should be reflected in the table.

Response and actions taken:

Added amendments to follow-up register to illustrate when issues were encountered and the outcome of their resolution.

Quality of document (clarity, presentation, etc.)

Reviewer's comments and recommendations:

The report is clear and concise, and it is well written and laid out very well. It is easy to read, and the table of contents directs the reader straight to the information. The evaluation of the stakeholder's satisfaction is a nice touch giving valuable insight into the success of the team and project development. A good report!

Response and actions taken:

Grateful for the kind words and glad the quality of the document is satisfactory.