

Project Soloma

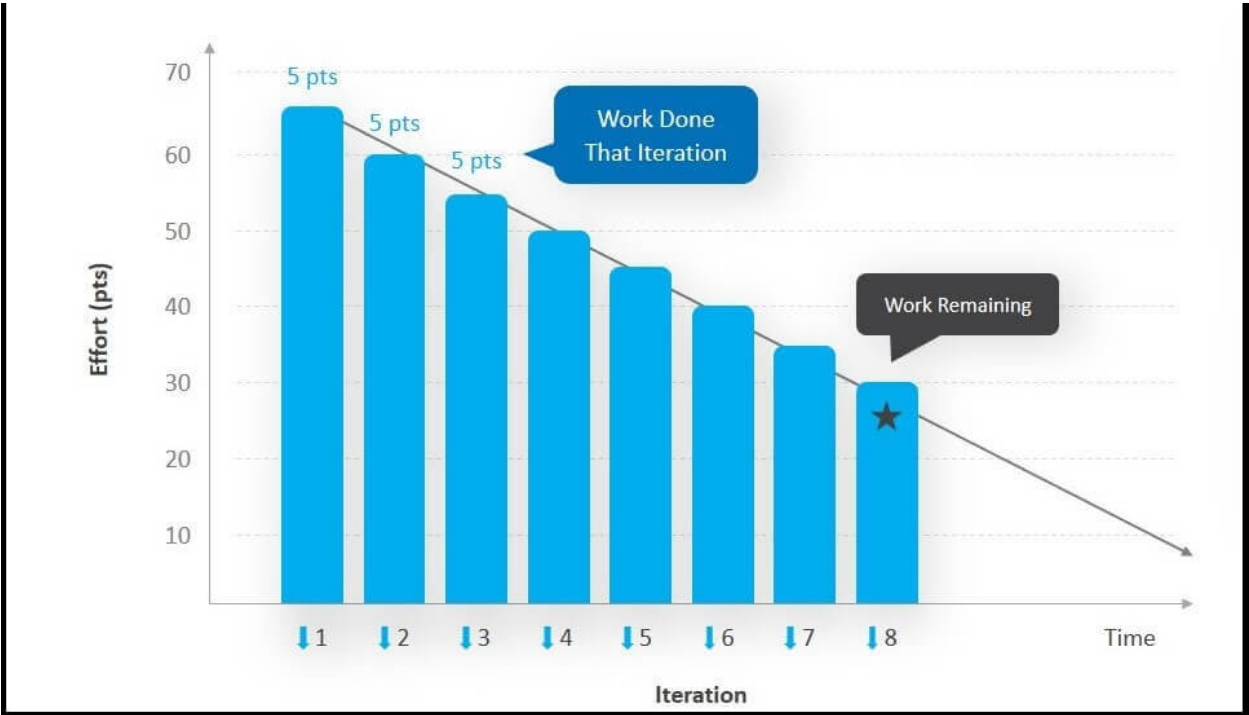
System Proposal Part 1+ Part 2

Andres Gaspar Gonzalez

CSC3150 System Design

Prof. Andy Cameron

May 25, 2024



(Jones, 2022)

Table of Contents

<i>Executive Summary</i>	4
Problem Statement	4
Project Vision and Scope	4
Requirements Summary	4
Stakeholders and Their Interests	5
Expected Costs and Benefits	5
Constraints	5
Document Overview	6
2.0 <i>System Initiation</i>	7
Project Initiation Request (PIR)	7
3.0 <i>Feasibility Assessment</i>	12
Introduction	12
Feasibility Analysis	12
Conclusion	13
4.0 <i>Requirements Definition</i>	14
Introduction	14
Functional Requirements	14
Non-functional Requirements	15
5.0 <i>Requirements Model</i>	16
Use-Case Diagram	16
Use-Case Descriptions	17
Use Case Name: Process Log-in.....	17
ID: UC-2.....	17
Importance:	17
Must Have	17
Use Case Name: Create Vision.....	18
ID:	18
UC-4	18
Importance:	18
Must Have	18
Use Case Name: Build Sprint	19
ID:	19
UC-5	19
Importance:	19
Must Have	19
Use Case Name: Maintain project	20
ID:	20
UC-7	20
Importance: Must have	20
Use Case Name: Process SCRUM review	21

ID:	21
UC-8	21
Importance: Must have	21
Use Case Name: Deploy Project	22
ID:	22
UC-9	22
Importance: Must have	22
6.0 Sys.....	22
tem Evolution	23
7.0 Conclusions and Recommendations.	23
Appendices.....	24
Glossary.....	24
Bibliography.....	25

Executive Summary

My name is Andres Orlando Gaspar Gonzalez, and I am a third-year student at Seattle Pacific University. I am majoring in Computer Science and aspire to become a Software Engineer. I have a strong passion for software development and the many possibilities that come with it. I am working on providing a system to help developers like me focus more on the task rather than the other things that come with being a developer. Such things are mainly about communication. During a conference held at SPU called the Erikson conference, a situation many teams had to go through further inspired me. I saw this situation as lacking a structured platform where a big part of the software development process could occur. Thus, Project Soloma aims to be that platform that will enhance communication by tracking and following the SCRUM framework. This project's primary goal is to limit the dependency on several applications and reduce the project's complexity.

1.0 Introduction and Overview

Problem Statement

Recognized as one of the most popular frameworks, SCRUM aids product teams by incorporating three essential principles: transparency, inspection, and adaptation. Teams using this framework expect to be collaborative, timelier at go/no-go decisions, and able to realize product value sooner. Yet, during the software development lifecycle (SDLC), communication and guidance often struggle. The effects of such struggles can range from deadlines not met to introducing high-stake risks, such as user unsatisfactory features that harm the product. Teams that combat these struggles would become more efficient and less prone to errors. Here, the opportunity represented is the need for a service that guides teams and allows them to communicate more transparently.

Project Vision and Scope

By continuously aiming to work on the software development life cycle, product teams can provide user-satisfactory services. To do so (or maybe in contribution), project Soloma seeks to give the teams a digital SCRUM-based project management platform. It aims to aid with projects expected to be delivered quickly and efficiently. Some boundaries of this software will strictly aim to adapt scrum to all the applications of UI features.

Requirements Summary

Abstract: This list below contains features and capabilities that project Soloma will have.

Primary Business Requirements:

- A daily scrum feature that must summarize the scrum goals and any updates for the day.
- A product backlog feature that is implementable. You can add, delete, select, and move a task.
- A product burndown chart feature shows when the work will be completed.
- An implementable sprint backlog features like the product backlog.
- A sprint burndown chart feature like the product burndown chart.
- A vision feature that begins the project process.
- A sprint review feature at the end of a sprint.
- A feature that assigns roles to team members of a project.
- A chat feature for communication throughout the team.
- A calendar feature that sets up meeting times.
- Security sign-in feature to safely keep project information.

Primary User Requirements:

This is a list of the users' proposed use cases:

Used to:

- Create scrum projects.
- Guide users through the scrum process.
- Track project process.
- Set up meetings.
- Communicate between team members.

Stakeholders and Their Interests

- Development team: A digitalized system that guides this team through the development process aims to improve efficiency and transparency. Thus, the development team would have a high interest in this project.
- User experience team: With the system being more transparent, the UX team would be highly interested in this project, as user stories will guide the development team.
- Security: Since this platform would contain valuable project information, the security team would have a high influence on the project but a low interest as it introduces risks for the company.
- Finance: This team would have easier budget approximations, as all a project's information would be available to them on the platform. Thus, they would be highly interested in the project.
- Project owners: Project owners would be more transparent about their projects' progress, increasing their interest.
- Business owners: The system aims to push more projects to the shipment at a lower risk rate. Thus, their interest in the system would be high.
- Scrum master: As part of the development team, having a digital platform to visualize the process of a project would make a scrum master's job easier; thus, their interest would be high.

Expected Costs and Benefits

Business Benefits:

- Growth in project efficiency
- Stakeholders have peace of mind knowing what is going on during the project.
- Leads to growth in revenue.
- Customers are more loyal based on the expected project completion.
- The team is more satisfied with having guidance throughout the process.
- Increases team value.
- Enhances system processes.

Cost areas:

- funding is required.
- Training for users is required.

Constraints

Since we are basing this system on the SCRUM framework, it cannot fit another approach to product development.

Recommendation:

After reading this document, I want the reader to approve Project Soloma and then allocate the resources to begin development. Next, build a team based on the Scrum framework, assigning roles such as the product owner, scrum master, and a team of 5-10 developers. Once the team building is done, the first meeting can take place to develop and agree on a vision. Finally, monitor the project and provide feedback by the set finalization date to see how well the project did based on the system proposal.

Document Overview

As for the following, section 2.0 will contain the Project Initiation Request (PIR) form. Section 3.0 will include the project's feasibility assessment, including an introduction, the feasibility analysis, additional comments, and a conclusion. Section 4.0 contains the definition of requirements, including an introduction and functional requirements. Finally, at the end of this document, there will be three subsections: Appendices, Glossary, and a Bibliography.

2.0 System Initiation

Include copies of the document that began this process – The PIR. No introduction or other text is needed.

Project Initiation Request (PIR)

PIR-00000 [PIR Number to be assigned by the Project Office]

Project Initiation Request (PIR) –

Level1 v6.0

Project Name: Project Soloma

Student Name: Andres Gaspar Gonzalez

This Project Initiation Request (PIR) is to be completed for all requests expected to require over 40 hours of effort or over 4 weeks of total duration. For larger requests requiring over 40 person-days or estimated project costs greater than \$5,000, this template is used to assess the product's feasibility and get approval to scope and plan the proposed project.

If approved, the Level 2 template (System Proposal: Part 1 and Part 2) must be completed.

NOTE: Sections 0-4 are required. Section 5 is optional, but any ideas on estimating costs should be included. **Replace the *italic* prompts with your answers/information.** [Expand each section in this template as needed.]

0. General Project Information

Project Name:	Project Soloma
Two Sentence Request Description:	The business result of this project is more transparency and accessibility, and projects are more efficient.
Requested Launch Date(s):	June 6, 2024
Department(s) Affected By Project:	Development, UX, Security, Finance, Sales.
Project's Customers:	This project intends to help companies digitalize product management. Thus, the customers could be any company that develops in the scrum framework or wants to try it. The affected would be developers, stakeholders, and product owners.
Date Request Submitted:	April 29, 2024

1. Project Sponsor and Manager

Project Sponsor

Name:	Andy Cameron
Title:	Professor
Department:	Computer Science - SPU
Email:	acameron@spu.edu

Business Project Manager & Requestor

Name:	Andres Gaspar Gonzalez
Title:	Student
Department:	Computer Science - SPU
Email:	gaspargonzaa@spu.edu

2. Business Problem or Opportunity: The motivation for this request

Describe the problem or opportunity that you would like to solve. Include a simple, high-level description of this request's business problems or opportunities. Focus on the problem or opportunity, not the solution. Be sure to include any date or deadline-related dependencies or needs related to the project.

The opportunity to digitalize the framework for businesses already using the scrum framework in their software development lifecycle arises. Though one principle of this framework is based on transparency, the lack of a digital platform doesn't allow for anytime access or communication on the project. Thus, the project is a digital platform that provides users with a graphic user interface that helps guide them through the scrum process and keeps them updated 24/7.

3. Justification, Impact, and Importance

What is the financial impact and justification for this request? How will the investment of time, resources, and capital be returned to our company? (Please note any contractual or regulatory requirements associated with the request. If you have an NPV, IRR, or ROI calculation, please provide the link(s) in this section.)

Assumptions

- *Include at least two. Add more rows to each table as needed.*
- The target audience has access to capable operating systems.
- The target audience has access to the internet.
-

Competitive Landscape / Context

- *Include at least two.*
- Zoho
- Jira Software

Tangible Return, Opportunity, or Value

One Time On-Going

- | | One Time | On-Going |
|---|----------|----------|
| ▪ <i>Include at least two. Estimate the best you can.</i> | \$ 0 | \$ 0 |
| ▪ Projects are more efficient. | \$ 0 | \$ 0 |
| ▪ Team members understand their part more in-depth and thus are prone to fewer errors | | |
| ▪ If errors occur, it won't affect the entire project; it will just affect that one part. | | |

Intangible Benefits

Impact or Value

- | | Impact or Value |
|---|-----------------|
| ▪ <i>Include at least two.</i> | \$ 0 |
| ▪ Project stakeholders have more peace of mind. | \$ 0 |
| ▪ Users learn to do things in a structured framework. | |

4. Product Requirements

The Project team will gather detailed requirements once the project is approved. Use this section to articulate the critical solution components to help scope the project's size and complexity. Do not describe how the solution will be implemented; instead, only list the functionality or results you expect to receive when the product is complete/delivered.

4.1. Must Haves

- | | |
|---------|---|
| 4.1.1. | <i>Include at least two. Add more rows to each table as needed.</i> |
| 4.1.2. | A daily scrum feature must summarize the scrum goals and any updates. |
| 4.1.3. | Must have a product backlog feature that is implementable. |
| 4.1.4. | Must have a product burndown chart feature keeps track of how the product backlog is going |
| 4.1.5. | It must have an implementable sprint backlog feature. |
| 4.1.6. | Must have a sprint burndown chart to track the current sprint. |
| 4.1.7. | A vision feature must begin the project process. |
| 4.1.8. | It must have a sprint review feature. Here, we can also add the option to finish the project. |
| 4.1.9. | It must have an assigned feature that assigns roles to users. |
| 4.1.10. | Must have a chat feature such as direct or group messaging. |
| 4.1.11. | You must have a security feature before logging in. |

4.2. Could Haves (Nice to Haves)

- | | |
|--------|--|
| 4.2.1. | <i>Include at least two.</i> |
| 4.2.2. | It could have a video chat feature. |
| 4.2.3. | We could have a suggestion bot that suggests information based on the scrum framework process. |
| 4.2.4. | Could we have a library dedicated to all the information on Scrum? |

4.3. Won't Haves (Don't Do's, aka Out of Scope)

- | | |
|--------|--|
| 4.3.1. | <i>Include at least two.</i> |
| 4.3.2. | We won't have a developing platform or feature that allows any programming related to a project. |
| 4.3.3. | It won't have a video playback feature. I won't be able to play videos. |

5. Project Costs (Operating and Capital: Onetime and Recurring) [Optional]

*This section is typically fleshed out after the requestor has submitted a PIR and received approval for the initial scoping effort. It captures the effort estimates, capital expenditures, and other costs associated with performing this work and creating the product/solution. If the submitter has thoughts or estimates on these costs or suggestions on how they might be estimated, please include those here. Add brief descriptions as needed. **Include at least two comments on your thinking around these items, even if you don't have specifics yet.***

Labor Costs

Type	Team(s) Affected	Low (hrs)	High (hrs)
Analysis & Design		0	0
Development		0	0
Testing and Quality Assurance		0	0
Systems Integration		0	0
Deployment		0	0
Support and Maintenance		0	0
Sales and Marketing		0	0
Total		0	0

Comments: *Include notes here on the costs or how they can be estimated. (optional)*

Capital Costs (Equipment, Software, Licenses, ...)

Description	Quantity	Cost (\$)
<i>Item 1</i>		\$ 0
<i>Item 2</i>		\$ 0
Total		\$ 0

Comments: *Include notes here on what these are or how they can be estimated. (optional)*

Type	Hours / Month Low	Hours / Month High	Maintenance Costs (Costs after the product is live)
System / User Support	0	0	
Business / Process Support	0	0	
Total Support & Maintenance	0	0	

Comments: *Optional.*

3.0 Feasibility Assessment

The attached example (at the end of this document) shows one layout and the level of required detail. However, your sections will be slightly different (remember that we added some feasibility sections), AND the complete assessment is not intended to fit on a single page.

Introduction

This section contains five areas of feasibility that will help conclude the feasibility and risk of the project. The rating scale will be low, medium, and ideal for feasibility and low, medium, and high for risk.

Feasibility Analysis

*Include complete analysis for five areas of feasibility - **Technical, Resource, Schedule, Organizational, Legal, and Contractual.***

Technical Feasibility

The project management system is feasible, yet there is some risk.

- This system should not cause technical problems, as SCRUM is already one of the most popular frameworks for application development. So, with user familiarity, the system is **feasible and has low risk** because of possible training requirements.
- Outsourcing of experienced developers in SCRUM will be needed to ensure the application follows all SCRUM rules and processes. This area makes analysts' familiarity **risky** because of the high complexity of the system.
- Like the framework, this system has many subsections, which we will develop step-by-step. The project size feasibility is **feasible** here. The project team will most likely include around five people.
- Developers will set this system's primary features and structure on a proven framework. Therefore, the project structure feasibility is **ideal**.

Resource Feasibility

- Project Soloma will only run on MacOS, so the company would need to provide additional hardware to members who do not have a work Mac. This area is of **low feasibility and has a medium risk** because of users' rejection of MacOS.
- Users will need access to the cloud and a subscription; therefore, this area is of **medium feasibility with low risk**, as this is standard practice for businesses today.

Schedule Feasibility

- This project aims to help teams through guidance and transparency at the service of a digital platform. However, Soloma does not have this option to develop it; therefore, this area is of **low feasibility and high risk** because of communication and timely errors.
- Maybe resources will not be available when needed, making the **feasibility of this area low and high risk**.
- This system has no product launch date, so this area is **feasible with shallow risk**.

Legal Feasibility

- Since this application will monitor work and work will be done using it, we must adhere to the FLSA. This area is **low risk**.
- No payments, health, or accounting will take place in his environment. Thus, there is **a low risk**.

Contractual Feasibility

- There will be a contract regarding subscription usage for licensing rights, which involves **medium risk**.
- Usage for frameworks and licenses will be needed, but this is **a low risk**.

Organizational Feasibility

- Stakeholders unfamiliar with digital platforms may need some training or help when first presented to the platform. This will use company resources thus **feasibility is medium with low risk**.
- Expected benefits from this system will help those reluctant to become adapters which makes **feasibility medium with low risk**

Additional Comments

- There are no additional comments.

Conclusion

This system aims to provide a digital platform for the development process of company projects. the switch, a feasibility assessment is conducted to measure whether such a project proves helpful. Based on the evaluation above, Project Soloma is medium feasibility with low risk. To mitigate some project risks, company goals and values will be at the forefront of the project process.

4.0 Requirements Definition

Provide a text overview of system services and behavioral properties. It will have a combined audience, including technical (developer) and non-technical (user/client) readers.

Introduction

This section is about Project Solomas' requirements. It contains both functional requirements and non-functional requirements. For instance, a functional requirement of Project Soloma is that it guides users through the SCRUM development process. A non-functional requirement for this project is how much downtime the system can handle.

Functional Requirements

1. Create a vision.
 - 1.1. The system shall create a vision (project) to store information such as the product backlog, goals, and purpose.
 - 1.2. The system shall produce daily "daily scrum" updates to output information based on project progress.
 - 1.3. The system will keep track of project progress based on user input and a schedule.
 - 1.4. Maybe user stories
2. Commit sprint.
 - 2.1. The system shall create a sprint plan.
 - 2.2. The system shall provide product backlog information so users can review it before committing.
 - 2.3. The system shall clearly define the sprint goal.
 - 2.4. Estimate/ create sprint backlog.
 - 2.5. Commit sprint.
 - 2.6. Create acceptance criteria.
3. Maintain project.
 - 3.1. Daily Scrum
 - 3.2. Show progress.
 - 3.3. Show the burndown chart for both sprint and product backlog
 - 3.4.
4. Review/Deploy project.
 - 4.1. Sprint review
 - 4.2. Discuss what is done and not done.
 - 4.3. Sprint Retrospective
 - 4.4. Inspect and adapt.
 - 4.5. Ship product or move to next sprint.
 - 4.6. Run through acceptance criteria.

Data Requirements:

1. Vision (project) data that the system will save includes product backlog, goals, and purpose.
 - 1.1. The system will save all three aspects in name, description, and ID format.
2. The system must display the daily scrum from 1.2 of functional requirements in a notification manner.
3. The data that the system will save from Functional Requirements 1.3 will be in description format, and updates to progress in time format hh:mm:ss using a 24-hour clock and a calendar.
4. For functional requirement 2.1, data that the system will save includes a sprint backlog as item name, description, and ID format.
5. For functional requirement 2.5, the system shall begin an instance of a sprint set on a schedule. The output of progress would be in a 24-hour clock and calendar format.
6. For functional requirement 2.6, the system should save the acceptance criteria in a description manner that contains conditionals, which would determine if a sprint is marked as finished.
7. Data the system will save when maintaining a project includes yes or no answers affecting conditionals and a description of progress made or to be made.
8. Data to be saved and implemented for the Review/deploy project will be sprint data, product backlog data, and data regarding inspection and adaptation. The system will store the data in a description format.

Non-functional Requirements

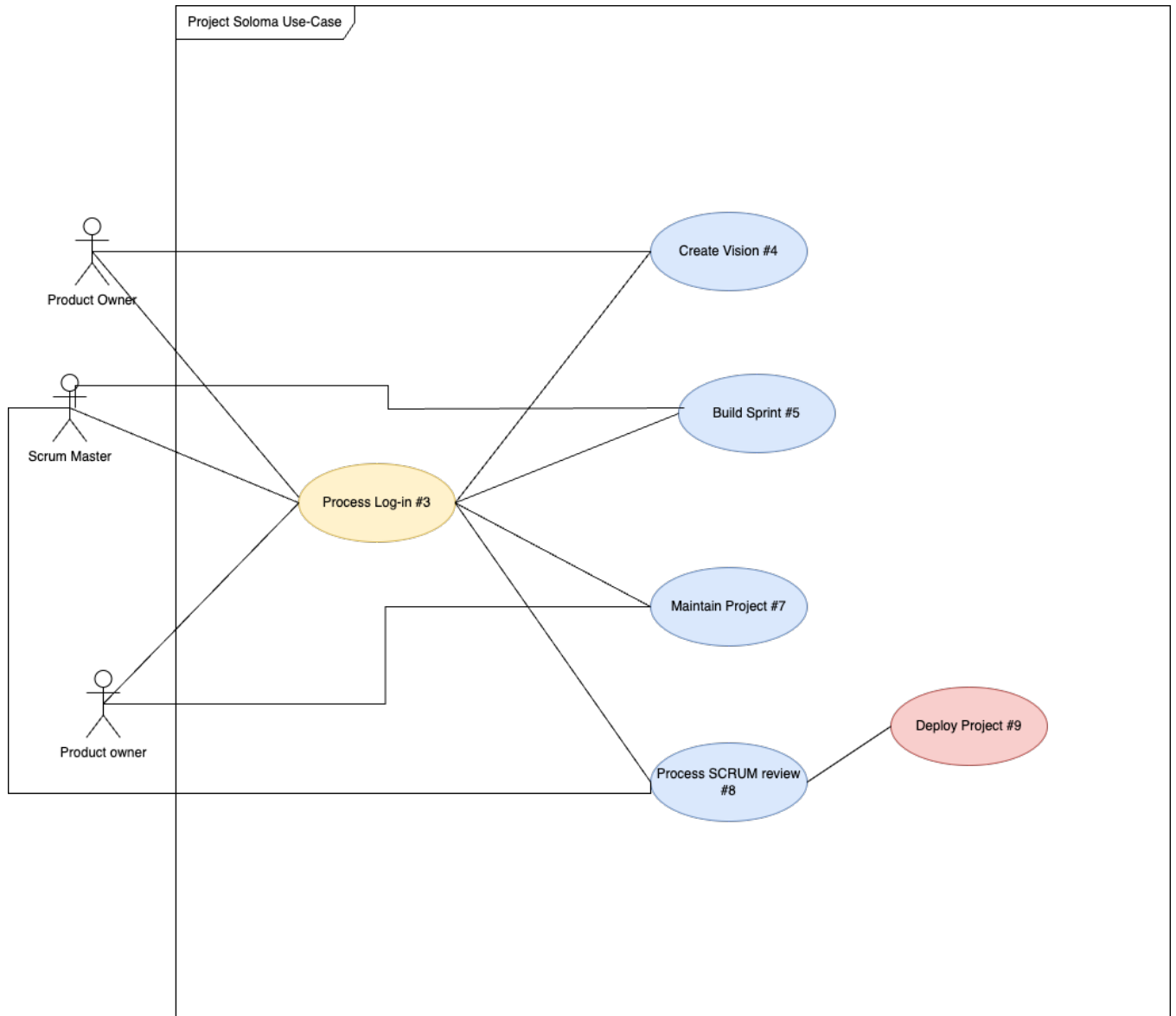
1. Operational requirements
 - 1.1. The system shall be built for macOS to begin with. Other versions capable of running on another platform will be released once 1st version is successful.
 - 1.2. The user interface shall be user-friendly and be expected to receive excellent reviews.
2. Performance requirements
 - 2.1. The system graphics must show information that meets current market standards for runtime.
 - 2.2. The system shall aim to have a minimal downtime of 90%.
3. Security requirements
 - 3.1. The system shall have the user sign in and 2-step verification.
 - 3.2. Passwords shall renew every 90 days.
4. Cultural requirements
 - 4.1. None.

5.0 Requirements Model

Introduction:

In this section, a use-case diagram and several use-case descriptions will show both graphically and descriptively how the projects aim to handle user interactions with product features. The use-case diagram is based on the UML (unified modeling language) format to provide a greater outreach. The use-case descriptions give an in-depth analysis of each use case and how they relate to the end-user.

Use-Case Diagram



Use-Case Descriptions

Use Case Name: Process Log-in		ID: UC-2	Importance: Must Have
Primary Actor: User		Use Case Type: Overview, essential	
Supporting Actors: Supervising Manager			
Stakeholders and Interests: Company, CEO, Project team.			
Brief Description: Before being given access to the system, a user must log in			
Trigger: The user starts the system, and the system displays the log-in page.			
Type (mark one): <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal			
Relationships: Association: Every user will have to log in when starting the system. Include: N/A Extend: UC-4, UC-5, UC-7, UC-8 Generalization: N/A			
The Normal Flow of Events: <ol style="list-style-type: none"> 1. The system will ask the user for an email and password. 2. The system will take the user to the main page. 			
Sub-flows: The system will ask the user to provide a different email if the email is already in use. If the system denies the user access, the user will receive a notification in the provided email and be notified on the screen. If the user does not receive approval, then the user does not have access to the home screen and is instead sent back to the sign-in page.			
Alternate/Exceptional Flows: <ol style="list-style-type: none"> 1.2. If the user is new, they must create an account. <ol style="list-style-type: none"> 1.2.1. The system would ask the user for an email username and password. 1.2.2. The system grants an account and access to the system's main page to the user. 1.2.3 If the user forgot the username or password, the user can select "forgot username or password" and will receive an email resetting the password or verifying the username. 1.2.4 If not, the system asks the user to redo steps 1.2-1.2.2. 			
Special Requirements: <ul style="list-style-type: none"> - The manager must approve the account if it requests access to a created project. - Approval must take no longer than 30 minutes. 			
To do/Issues: Security measures, Supervising manager unavailability.			

Use Case Name: Create Vision		ID: UC-4	Importance: Must Have
Primary Actor: Product Owner		Use Case Type: Overview, essential	
Supporting Actors: Developer, Scrum Master			
Stakeholders and Interests: Product owner, developer, scrum master, product user.			
Brief Description: When beginning a new project, the team will create a vision (project) instance containing critical information, such as the product backlog, goals, and purpose.			
Trigger: The user accesses the vision page after selecting visions from the menu bar, and then the user chooses to create a new vision from the provided options.			
Type (mark one): <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal			
Relationships: Association: The product owner is to propose the vision of a project and thus communicate with the system vision feature Include: N/A Extend: N/A Generalization: Since the project's vision creates the project layout, all other use cases will connect because they contain the product backlog and goals. This feature will help users through the SCRUM process.			
The Normal Flow of Events: When proposing a new project, the team develops a vision. After doing so, the product owner inputs that vision into the system. Once the vision is complete, any team member can ask the system for a project layout that follows the SCRUM process.			
Sub-flows: N/A			
Alternate/Exceptional Flows: N/A			
Special Requirements: The project must have a purpose, a product backlog, and goals.			
To do/Issues: N/A			

Use Case Name: Build Sprint		ID: UC-5	Importance: Must Have
Primary Actor: Scrum Master		Use Case Type: Overview, essential	
Supporting Actors: Developers, Product owners.			
Stakeholders and Interests: Developers, Scrum Master, Product owner, end-user.			
Brief Description: After creating a vision, the development team can commit to a sprint. This use case marks an aspect of the SCRUM process.			
Trigger: The developer selects the "sprints" option under the sprint menu.			
Type (mark one): <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal			
Relationships: Association: The project team agrees to a sprint, and the scrum master inputs the sprint information into the system under the sprints category. Include: N/A Extend: N/A Generalization: Will share information with UC-2, 4, and 7.			
The Normal Flow of Events: When creating a new vision, the next step is to build a sprint. First, the system will ask the user to review the product backlog, and from there, the user can create a sprint. Second, the system will ask the user to define a sprint goal and acceptance criteria. Finally, the user can commit the sprint and begin the sprint process.			
Sub-flows: The project team will review the backlog, and the Scrum Master will input the information. The categories that the team implements are the sprint backlog, the product backlog, and the product calendar.			
Alternate/Exceptional Flows: N/A			
Special Requirements: To commit to a sprint, 1. There must be a clear and defined goal. 2. There must be acceptance criteria for that sprint. 3. There must be a sprint backlog.			
To do/Issues: Must find a reasonable method of timing the sprint.			

Use Case Name: Maintain project		ID: UC-7	Importance: Must have
Primary Actor: Developer		Use Case Type: Overview, essential	
Supporting Actors: Scrum master, product owner.			
Stakeholders and Interests: Scrum master, product owner, developers, business owner, end-users.			
Brief Description: During the Scrum process, there is a repetitive section regarding the daily tasks that the team must do. Therefore, a daily scrum must happen.			
Trigger: Based on a timer, the system will ask users to perform a daily scrum at the beginning of each project day. Type (mark one): <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal (rare)			
Relationships: Association: The developer must complete the daily scrum, and the system or manager must approve. Include: N/A Extend: N/A Generalization: N/A			
The Normal Flow of Events: At the beginning of each day, the system will ask each user to commit to a daily SCRUM by displaying each sprint's progress and ensuring they are on task.			
Sub-flows: When it is time for a daily scrum, the system will ask each team member to commit to the daily scrum, which the scrum master will approve			
Alternate/Exceptional Flows:			
Special Requirements: <ol style="list-style-type: none"> 1. Must ask users about daily scrum in the daily 2. Must receive valid input from users, which system or project manager decides if the team can go forward for the day. 			
To do/Issues: N/A			

Use Case Name: Process SCRUM review		ID: UC-8	Importance: Must have
Primary Actor: Scrum Master		Use Case Type: Overview essential	
Supporting Actors: Developer, product owner			
Stakeholders and Interests: Product owner, user, and development team.			
Brief Description: When finishing a sprint, the system will ask the scrum master to perform a scrum review.			
Trigger: The scrum burndown chart has shown that the sprint is over or the backlog is marked as completed.			
Type (mark one): <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal (rare)			
Relationships: <p>Association: When a sprint is complete, the system asks the Scrum Master to perform a scrum review.</p> <p>Include:</p> <p>Extend: If the project is incomplete, the progress on the product backlog or the scrum master decides when the scrum review occurs. After this, the system will shift back to UC-5.</p> <p>Generalization:</p>			
The Normal Flow of Events: Once a sprint is complete, either a user marking the sprint backlog as complete or the time is running out will notify the system. Next, the system will provide the user with review questions that the user must complete before moving on to a new sprint or marking a project as complete.			
Sub-flows: When the sprint is complete, the system will ask the user to input data on the sprint goal and analyze what the sprint was like. Also, the system will ask the product owner to implement the other project categories, such as the product backlog.			
Alternate/Exceptional Flows: N/A			
Special Requirements: <ol style="list-style-type: none"> 1. The product backlog must be marked as complete if time has not been up or time runs out. 2. The user must complete the review before moving to a new sprint or deploying the product. 			
To do/Issues: N/A			

Use Case Name: Deploy Project		ID: UC-9	Importance: Must have
Primary Actor: Product Owner		Use Case Type: Overview essential	
Supporting Actors: Developer, Scrum master			
Stakeholders and Interests: Product owner, end-user, and development team.			
Brief Description: After the product backlog is complete, the team can deploy the product.			
Trigger: The product owner selects the finished project during a sprint review			
Type (mark one): <input checked="" type="checkbox"/> External <input type="checkbox"/> Temporal (rare)			
Relationships: Association: When a sprint review is complete and the application is deployable, the product owner tells the system to mark the project as deployed. Include: N/A Extend: N/A Generalization: N/A			
The Normal Flow of Events: Once a SCRUM review is complete and the project is deployable, the product owner can tell the system that the project is complete. The system will respond with questions and tasks that the user must implement so that the system can mark the project as deployed.			
Sub-flows: While deploying a project, the system will ask the product owner if the product is meeting stakeholder needs.			
Alternate/Exceptional Flows: If the product does not meet stakeholder needs or there is disagreement, the project must go through the SCRUM process again.			
Special Requirements: 1. For the product to be deployable, the product must meet all goals, criteria, and stakeholder requirements.			
To do/Issues: N/A			

6.0 System Evolution:

After project Soloma has launched, some goals still need to be met. Ultimately, Project Soloma aims to be a platform that plays a big part in the development process. From team meetings via video calls and chats, the desired features aim to minimize dependency on various applications. This way, companies can not only focus more on the project but also track the project's progress more efficiently. Still, the project does not intend to incorporate any actual software programming. It will not become an IDE; instead, it will focus on the design aspect of the software development life cycle.

7.0 Conclusions and Recommendations.

In conclusion, Project Soloma aims to improve project management efficiency by providing a structured approach to the SCRUM development process. The product aims to help teams work through the SCRUM process efficiently, ensuring all requirements, deadlines, and user stories are incorporated. As for the stakeholders and benefits, various stakeholders, such as the project manager, developers, and end-users, benefit from the system as it enhances efficiency, reduces downtime, and allows for better communication. A Project Initiation request is part of this proposal, as it aims to show a high-level understanding of the project's goals, requirements, and justification. After a thorough feasibility analysis, the product, which is a project management system, is feasible, yet there is some risk. Factors of the feasible conclusion are that the system should not cause technical problems, the project size is not huge, and its implementation will be on the SCRUM framework. As for the risk analysis, one risk regards how we will source experienced Scrum developers, since the project must have the correct layout. Otherwise, it would fail in its purpose of being a Scrum-based project manager. Finally, a requirements model with a use-case diagram and several use-case descriptions shows the connections between the end-users and system features. Some desired features and goals of Project Soloma are that it becomes the sole driver of the design process of a project and that it will not host any functional programming.

Based on this proposal's findings, we recommend the following to ensure the implementation of Project Soloma is successful. First, Project Soloma must have an experienced SCRUM developer who will lead the team to ensure the product follows all SCRUM rules. Second, since the implementation will follow agile, the opportunity to perform various tests is present, so we must take advantage to produce a risk-free product. Third, ongoing feedback is necessary, as this project is very complex. Finally, developers must fulfill data requirements to fine-tune details because the system will provide access to company information. It would be risky not to secure this information.

Appendices

Include (at least) a copy of any completed questionnaires or surveys, plus any other applicable reference materials.

Glossary

- Feasibility: Measure of how beneficial or practical a new system will be to stakeholders.
- Feasibility analysis: Process by which feasibility is measured.
- SCRUM: An Agile methodology framework.

Bibliography

Cameron, A. (2024). *CSC 3150*. Lecture, Seattle ; Washington.

Larman, C. (2005). *Applying UML and patterns: An introduction to object-oriented analysis and design and the unified proces*. Prentice-Hall.

Jones, J. (2022). *sprint burn down chart*. What is a Sprint Burndown Chart & What is its Significance? UNICHROME. Retrieved May 10, 2024, from <https://unichrone.com/blog/agile/sprint-burndown-chart/>.

Alder, G., & Koreng, I. (2005). Draw.io. *draw.io*. computer software, JGraph Ltd. Retrieved 2024, from <https://www.drawio.com/>.