
<Group 05>

<SoulNote>
Software Development Plan
Version **<1.1>**

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

Revision History

Date	Version	Description	Author
30/05/2025	1.0	Introduction, Project Overview	Nguyen Tan Van
30/05/2025	1.0	Project Organization	Nguyen Le Quang
30/05/2025	1.0	Phase and Iteration Plan Releases	Ly Quoc Thanh
30/05/2025	1.0	Project Schedule	Pham Quang Thinh
30/05/2025	1.0	Project Monitoring and Control	Ly Quoc Thanh
09/06/2020	1.1	Add Project Estimates, Risk Management	Pham Quang Thinh
09/06/2020	1.1	Revise Project Schedule, Configuration Management	Pham Quang Thinh

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

Table of Contents

- 1. Introduction.....4**
- 2. Project Overview.....4**
 - 2.1 Project Purpose, Scope, and Objectives..... 4
 - 2.2 Assumptions and Constraints..... 5
 - 2.3 Project Deliverables.....5
- 3. Project Organization.....6**
 - 3.1 Organizational Structure..... 6
 - 3.2 Roles and Responsibilities.....6
- 4. Management Process.....7**
 - 4.1 Project Estimates.....7
 - 4.2 Project Plan..... 9
 - 4.2.1 Phase and Iteration Plan.....9
 - 4.2.2 Releases..... 10
 - 4.2.3 Project Schedule..... 15
 - 4.3 Project Monitoring and Control..... 18
 - 4.3.1 Reporting..... 18
 - 4.3.2 Risk Management..... 18
 - 4.3.3 Configuration Management.....19

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

Software Development Plan

1. Introduction

This Software Development Plan outlines the overall approach for developing the software system. It defines the purpose and scope of the project, introduces relevant terms and references, and provides a brief overview of the plan's structure. The document serves as a guide for managing and coordinating all development activities throughout the software lifecycle.

2. Project Overview

2.1 Project Purpose, Scope, and Objectives

Purpose:

The SoulNote project is designed to meet the growing need for a personal, emotionally enriched journaling platform that allows users to preserve and reflect on meaningful life experiences. The reasons for undertaking this project include:

- **Promoting Emotional Well-being:** Helping users process and revisit memories tied to emotions, fostering self-awareness and emotional resilience.
- **Preserving Personal History:** Enabling individuals to document important life moments through text, images, voice recordings, and emotional tags.
- **Encouraging Reflection and Growth:** Providing tools such as emotion-based filtering and timeline navigation to allow users to track emotional journeys over time.
- **Supporting Organization and Categorization:** Offering custom tags and themes for intuitive memory organization and retrieval.

Scope:

The scope of the SoulNote project includes the design, development, and deployment of a full-featured web-based memory journaling application. Key functionalities include:

- User account registration and authentication
- Creating and managing memory entries with text, photos, voice recordings, moods, and tags
- Viewing memories via calendar views, emotional filters, and visual timelines
- Tag management and personalized memory organization

Objectives:

The objectives of the SoulNote project are to:

- Develop a user-friendly web application that supports multimedia journaling, including text, images, and voice recordings.
- Implement emotion tagging and filtering features to help users reflect on their emotional journeys.
- Enable secure user authentication and personalized account management.
- Design intuitive interfaces for creating, organizing, and retrieving memories through tags, moods, and timeline navigation.
- Ensure responsive, accessible, and cross-platform compatibility across modern web browsers.
- Support data privacy and user control over their personal content.

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

2.2 Assumptions and Constraints

Assumptions:

- The project team will have continuous access to necessary development tools and resources.
- All team members are proficient in the required technologies and methodologies.
- Stakeholders will provide timely feedback and approvals throughout the project lifecycle.
- The platform will be accessible through major web browsers and mobile devices.
- Adequate documentation and guidelines will be provided for all third-party tools and libraries used.
- End users will be willing to engage with the platform and provide valuable feedback for improvements.

Constraints:

- The project must be completed within a fixed timeline of 12 weeks.
- This is a zero-budget project with no external funding or paid services.
- The team consists of 5 members, and no additional personnel will be added during the project.
- Only free or open-source development tools and platforms (e.g., Firebase, GitHub) may be used.
- All development and testing must be conducted within the academic semester timeframe.

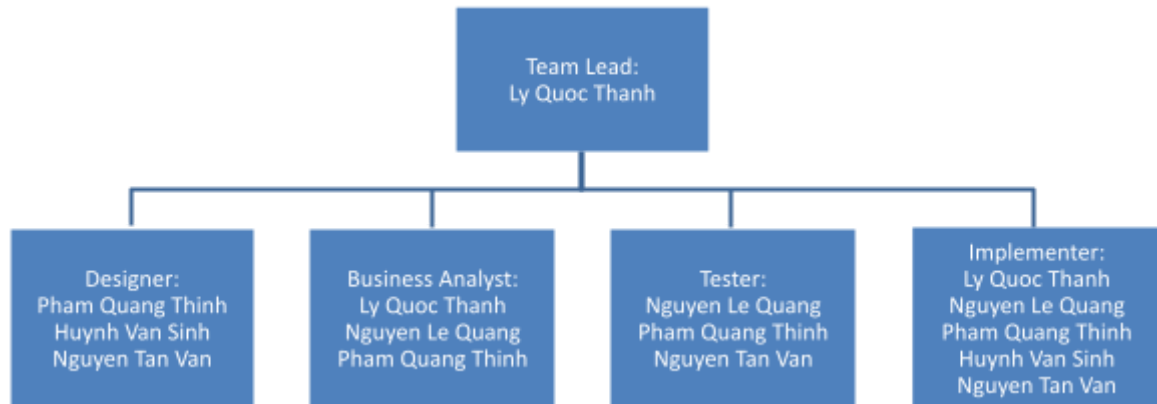
2.3 Project Deliverables

Deliverables for each project phase are identified in the Development Case. Deliverables are delivered towards the end of the iteration, as specified in section [4.2.3 Project Schedule](#).

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

3. Project Organization

3.1 Organizational Structure



3.2 Roles and Responsibilities

Role	Person	Responsibilities
Team Lead	Ly Quoc Thanh	<ul style="list-style-type: none"> - Write the project plan (including task assignments) - Monitor and report project status weekly - Coordinate meetings - Main contact point for the project
Business Analyst	Ly Quoc Thanh, Nguyen Le Quang, Pham Quang Thinh	<ul style="list-style-type: none"> - Document requirements (e.g., interview customers) - Communicate requirements with team members - Coordinate to review requirements
Designer	Pham Quang Thinh, Huynh Van Sinh, Nguyen Tan Van	<ul style="list-style-type: none"> - Design the system (components, classes, databases, etc.) - Design the user interface - Create the Software Architecture Document
Tester	Nguyen Le Quang, Pham Quang Thinh, Nguyen Tan Van	<ul style="list-style-type: none"> - Write test plans and test cases - Perform system testing
Implementer	Ly Quoc Thanh, Nguyen Le Quang, Pham Quang Thinh, Huynh Van Sinh, Nguyen Tan Van	<ul style="list-style-type: none"> - Write source code - Perform unit tests - Review source code

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

4. Management Process

4.1 Project Estimates

Implementation:

Feature	Implementation Time	Implementation Cost
Memory Creation & Upload	5 days	\$0
Timeline & Memory Browsing	3 days	\$0
Advanced Search & Filtering	4 days	\$0
Memory Editing & Deletion	2 days	\$0
Memory Sharing	3 days	\$0
Emotion-Based Visualization	4 days	\$0
User Authentication	3 days	\$0
Account Management	2 days	\$0
Location Integration (Map View)	4 days	\$0
Testing & Debugging	4 days	\$0
UI/UX Design	4 days	\$0
Backend/API Setup	5 days	\$0

Tools:

Tool	Feature	Usage Estimate (Student Project)	Cost
Firebase Hosting	Static website hosting (frontend)	~200 MB for static pages and assets	~\$0 (<i>Free for up to 1 GB</i>)
	Data transfer	< 1 GB/month (development + demo traffic)	~\$0 (<i>Free for 10 GB/month</i>)
	SSL/HTTPS	Always enabled	Free

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

Firebase Cloud Firestore	Stored data (text, metadata)	< 1 GiB total	~\$0 (<i>Free for 1 GiB</i>)
	Document writes	~5K/day (estimated usage)	~\$0 (<i>Free up to 20K/day</i>)
	Document reads	~10K/day	~\$0 (<i>Free up to 50K/day</i>)
	Document deletes	< 1K/day	~\$0 (<i>Free up to 20K/day</i>)
Firebase Cloud Storage	Image/audio uploads	~200 MB total media uploads	~\$0 (<i>Free up to 5 GB</i>)
	Media downloads	< 500 MB/month	~\$0 (<i>Free up to 1 GB/day</i>)
	Upload/download operations	< 5K/day	~\$0 (<i>Free up to 20–50K/day</i>)
GitHub	Code repository	Used for code collaboration and version control	Free
Trello	Task management	Free version used	Free
Google Docs/Drive	Collaborative documentation	Shared among 5 members, < 1 GB used	Free (Google Workspace EDU)
Google Meet	Team meetings / reviews	~2–3 meetings/week	Free
Map Integration (e.g., Leaflet.js + OpenStreetMap)	Display memory location on map	Open source / no cost	Free

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

4.2 Project Plan

4.2.1 Phase and Iteration Plan

- Phase 1: Inception (contains 1 sprint)

Duration: 2 weeks

Starting date: 19/05/2025

Ending date: 01/06/2025

Sprint	Objectives
1 (19/05/2025 - 01/06/2025)	<ul style="list-style-type: none"> - Group registration - Project Proposal - Tools creation - Done planing most of the project - Vision document - 2 weekly reports on group works

- Phase 2: Elaboration (contains 2 sprints)

Duration: 4 weeks

Starting date: 02/05/2025

Ending date: 29/06/2025

Sprint	Objectives
2 (02/06/2025 - 15/06/2025)	<ul style="list-style-type: none"> - Revise and improve project documents - Schedule Sprint 2 backlog - Create use-case models - Write detailed use-case specifications - 2 weekly reports on group works
3 (16/06/2025 - 29/06/2025)	<ul style="list-style-type: none"> - Revise and extend use-case specifications - Define software architecture - Model class diagrams - Design database - 2 weekly reports on group works

-

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

- Phase 3: Construction (contains 2 sprints)

Duration: 4 weeks

Starting date: 14/07/2025

Ending date: 10/08/2025

Sprint	Objectives
4 (14/07/2025 - 27/07/2025)	<ul style="list-style-type: none"> - Revise Software Architecture Document - Design UI prototype - Deliver working software - Prepare test plan and test cases - 2 weekly reports on group works
5 (28/07/2025 - 10/08/2025)	<ul style="list-style-type: none"> - Implementation - Manual Testing - Automated Testing - 2 weekly reports on group works

4.2.2 Releases

Sprint 1 – PA0 & PA1: Group Registration, Tool Setup & Project Proposal:

Objectives:

- Form project teams and complete group registration.
- Propose and define a software product idea.
- Set up essential collaboration and development tools: Slack, Trello, GitHub, JIRA, Moodle.

Deliverables:

- Completed Group Registration Form (Google Sheet).
- **Project Proposal** including:
 - Product description and motivation.
 - Target users and platforms.
 - Key features.
 - Evidence of interviews with at least 5 potential users (e.g., recordings, notes, profiles).
- **Tooling Setup:** Team Slack channel, JIRA board, GitHub repository with correct folder structure.

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

Outcomes:

- The team was successfully registered with all required member information.
- The project idea was clearly defined and documented.
- All required tools were fully set up and actively used.

Sprint 2 – PA2: Requirement Specification and Design Foundation:

Objectives:

- Refine the project plan and vision document based on feedback.
- Specify detailed requirements and use-case modeling.
- Schedule Sprint 2 tasks using JIRA with proper time estimation and task assignment.

Deliverables:

- **Revised Project Plan & Vision Document**
 - Incorporated TA feedback and added more detailed functional/non-functional requirements.
 - Added analysis of competitors, alternatives, and user environments.
 - Revision history and changes highlighted in blue.
- **Sprint 2 Backlog**
 - Backlog items clearly defined and estimated.
 - All team members assigned at least one task.
- **Use-Case Model**
 - Diagrams created using tools like StarUML or Visio.
 - Covered all major user interactions based on functional requirements.
- **Use-Case Specifications**
 - Detailed scenarios for each use-case including basic and alternative flows, pre/post conditions.

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

- **Weekly Reports**

- Updated team progress for each week during the sprint.

Outcomes:

- Requirements were clearly documented and validated through use-case modeling.
- Initial design preparation established a solid foundation for later stages.
- All team members were actively involved in planning and requirement analysis.

Sprint 3 – PA3: Architecture and Design Foundation

Objectives:

- Finalize the use-case model and specifications.
- Define the system's software architecture.
- Create class and database models based on current requirements.

Deliverables:

- **Use-Case Specification – 2nd Submission**

- Refined and expanded use-cases based on TA feedback and new requirement updates.
- Revision history updated; changes highlighted for clarity.

- **Software Architecture Document**

- Architecture follows the selected style (e.g., MVC, n-tier, microservices).
- Identified key components, their interactions, and technologies used (e.g., languages, frameworks).
- Answered how components are connected and deployed.

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

- **Class Diagrams**

- Modeled core classes based on architectural components.
- Included attributes, methods, and relationships among classes.
- Updated Section 4.x in the architecture document.

- **Database Design**

- Designed the ER model with entities, attributes, and relationships.
- Integrated into the DB section of the architecture document.

- **Weekly Reports**

- Progress logged from the end of PA2 to submission time using the provided template.

Outcomes:

- The team established a clear architectural foundation.
- System structure and components were well-defined for the next development phase.
- Database and class-level designs aligned with system requirements.

Sprint 4 – PA4: UI Design, Deployment & Testing

Objectives:

- Finalize system design and structure.
- Implement core functionality.
- Build UI mockups for key scenarios.
- Plan, execute, and report on functional testing.

Deliverables:

- **Revised Software Architecture Document (SAD)**

- Updated Sections 1–4 based on TA feedback and new project insights.
- Added Section 5: UML deployment diagram describing system nodes and network connections (e.g., client-server or Android-only setups).

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

- Added Section 6: Code folder structures for components such as Web server, client, etc.
- **UI Prototype**
 - Designed the main screen and at least 3 key scenario screens using tools like Figma, Moqups, or Visily.
 - Documented all screens and usage descriptions.
- **Working Software**
 - Developed and delivered a working version of at least one main use-case.
 - Demonstrated functionality aligned with previous specifications and requirements.
- **Test Plan & Test Cases**
 - Completed a comprehensive test plan including all required sections.
 - Selected 3 use-cases and created 15+ functional test cases (minimum 5 per use-case).
 - Executed test cases and recorded outcomes in a test report:
- **Weekly Reports**
 - Weekly progress from the end of PA3 to submission time documented using the provided template.

Outcomes:

- System design is now complete with clear deployment and folder structure.
- Main use-case implementation verified through hands-on testing.
- UI design closely reflects use-case flows, improving usability and planning for the next sprint.
- Software quality ensured through testing and documentation.

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

4.2.3 Project Schedule

Phase	Interaction	Start date - Due date	Task	Result	Releases
Inception	Sprint 1	19/05/2025 - 01/06/2025 (2 weeks)	<ul style="list-style-type: none"> - Group registration - Write project proposal - Set up working tools - Plan initial project timeline and tasks - Create initial vision document - Write 2 weekly group reports 	<ul style="list-style-type: none"> - Project proposal - Vision document - Project plan - Weekly report 1 - Weekly report 2 - PA0 and PA1 	<ul style="list-style-type: none"> - Project proposal - Vision document
		02/06/2025 - 15/06/2025 (2 weeks)	<ul style="list-style-type: none"> - Revise and improve project documents - Create and schedule Sprint 2 backlog - Create use-case models - Write detailed use-case specifications - Learning and training technology, tools - Write 2 weekly group reports 	<ul style="list-style-type: none"> - Updated project proposal - Revised project plan - Revised vision document - Use-case diagram - Use-case specifications - Weekly report 3 - Weekly report 4 - PA2 	<ul style="list-style-type: none"> - Use-case diagram - Use-case specification document
Elaboration	Sprint 2	16/06/2025 - 29/06/2025 (2 weeks)	<ul style="list-style-type: none"> - Revise and extend use-case specifications - Define software architecture - Model class diagrams - Learning technology and implement code - Design project database - Write 2 weekly group reports 	<ul style="list-style-type: none"> - Extended use-case specs - Software architecture diagram - Class diagram - Initial database design - Weekly report 5 - Weekly report 6 - PA3 	<ul style="list-style-type: none"> - Class diagram - Software architecture design - Database design document
		14/07/2025 - 27/07/2025 (2 weeks)	<ul style="list-style-type: none"> - Revise software architecture document - Design UI prototype - Deliver first working version of software - Prepare test plan and test cases - Write 2 weekly reports 	<ul style="list-style-type: none"> - Updated software architecture document - UI prototype - Working software prototype - Test plan and test cases - Weekly report 7 - Weekly report 8 - PA4 	<ul style="list-style-type: none"> - Software prototype v1 - UI prototype - Test plan document
Construction	Sprint 3	28/07/2025 -	<ul style="list-style-type: none"> - Implement core features - Perform manual testing 	<ul style="list-style-type: none"> - Implemented features - Test reports (manual & 	<ul style="list-style-type: none"> - Working software version

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

		10/08/2025 (2 weeks)	- Perform automated testing - Prepare for final submission - Write 2 weekly reports	automated) - Weekly report 9 - Weekly report 10 - Final submission - PA5	- Test result reports
--	--	-------------------------	---	---	-----------------------

● **PA0 (19/05/2025 - 25/05/2025):**

	Status	%	Dur	+	May	17	18	19	20	21	22	23	24	25	26	27	28	29
▼ Project Management			12w	+														
▼ Sprint 1 Week 1			1w 1d	+														
TASK-001 Brainstorm id...	Completed	100%	1w 1d	+														
TASK-002 Interview	Completed	100%	1w 1d	+														
TASK-003 Google Shee...	Completed	100%	1w 1d	+														
TASK-004 Create tools	Completed	100%	1w 1d	+														
DOC-001 Project descri...	Completed	100%	1w 1d	+														
RP-001 Weekly report 1	Completed	100%	1w	+														
TASK-007 Project Prop...	Completed	100%	1w 1d	+														

● **PA1 (26/05/2025 - 01/06/2025):**

	Status	%	Dur	+	May	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10
▼ Project Management			12w	+																		
▶ Sprint 1 Week 1			1w 1d	+																		
▼ Sprint 1 Week 2			1w 1d	+																		
DOC-002 Vision Docum...	Completed	100%	1w 1d	+																		
DOC-003 Software Dev...	Completed	100%	1w 4h	+																		
RP-002 Weekly Report 2	Completed	100%	1w	+																		
RP-003 Planning Report...	Completed	100%	1w	+																		
RP-004 Review Report ...	Completed	100%	1w	+																		

● **PA2 (02/06/2025 - 15/06/2025):**

	Status	%	Dur	+	June	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
▼ Sprint 2 Week 1			1w 1d	+																									
TASK-005 Use-case mo...	Completed	100%	1w 1d	+																									
DOC-004 Use-case sp...	Completed	100%	1w	+																									
RP-005 Weekly Report 3	Completed	100%	1w	+																									
▼ Sprint 2 Week 2			1w	+																									
RP-006 Review Report ...	Completed	100%	1w	+																									
RP-007 Do Planning Re...	Completed	100%	1w	+																									
DOC-005 Sprint 2's Bac...	Completed	100%	1w	+																									
DOC-006 Revise Vision...	Completed	100%	1w	+																									
DOC-007 Revise Softw...	Completed	100%	1d	+																									
RP-008 Weekly Report 4	Completed	100%	1w	+																									

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

4.3 Project Monitoring and Control

4.3.1 Reporting

- **Weekly Meeting:** Our team holds one main meeting in the middle of each week after all members have read and understood the assignment requirements. During this meeting, we assign tasks and clarify responsibilities. Additionally, we conduct one or two meetings at the end of the week to review the progress, check the completed work, and make necessary adjustments based on team discussions.
- **Weekly Status Report:** At the end of our weekend meeting, we upload all completed tasks to GitHub. These tasks are reviewed by the team leader and the TA before being officially used. After that, we prepare a weekly status report, which includes: achievements, challenges and next week's goals.
- **Informal Chats:** We use Messenger group chat and Discord for quick updates, questions, and informal discussions between team members. This helps us stay connected and resolve minor issues without waiting for official meetings.

4.3.2 Risk Management

<i>Risk ID</i>	<i>Risk Description</i>	<i>Probability</i>	<i>Impact</i>	<i>Risk Exposure</i>	<i>Priority</i>	<i>Mitigation Strategy or Contingency Plan</i>
R001	Members lack experience with full-stack web development (frontend + backend).	80	85	6800	High	Assign tasks based on each member's strength. Provide quick workshops or learning resources. Encourage pair programming.
R002	Miscommunication between team members leading to misunderstanding of requirements.	75	80	6000	High	Set up short, frequent meetings. Use shared docs (Google Docs, Trello). Confirm all tasks and responsibilities clearly.
R003	Incomplete or vague use-case specifications cause implementation errors later.	70	85	5950	High	Review and finalize use-case specs together. Request TA support for unclear requirements. Assign reviewer roles.
R004	Time pressure due to other academic courses and deliverables.	65	80	5200	Medium	Allocate buffer time in schedule. Prioritize important features. Reassign members when delays happen.
R005	Uploaded media files (images/audio) may exceed storage or cause performance issues.	60	77	4500	Medium	Limit file size at upload. Compress files client-side. Conduct media upload testing.
R006	Poor internet connection during group collaboration or online demo sessions.	60	70	4200	Medium	Prepare a demo video as backup. Ensure all team members have access to offline files. Use cloud syncing.

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

R007	Members are unfamiliar with Git, leading to merge conflicts and lost work.	65	65	4225	Medium	Provide Git workflow training. Use clear commit rules and branch naming. Assign a Git lead to support teammates.
R008	Backend server crashes due to too many simultaneous requests or memory leaks.	55	85	4675	Medium	Monitor backend performance early. Use lightweight frameworks. Conduct load testing before deployment.
R009	Users forget to tag memories, reducing the usefulness of timeline filters.	70	60	4200	Medium	Add reminders or hints for tagging. Set default tags or use AI to suggest tags (future enhancement).
R010	Sensitive data (e.g., personal memories) may not be properly secured.	60	90	5400	High	Use HTTPS, encrypt sensitive data, and implement proper access control. Regularly review for vulnerabilities.

4.3.3 Configuration Management

To ensure effective collaboration, task management, version control, and document sharing throughout the SoulNote project, our team utilizes a set of tools and platforms tailored to specific needs as follows:

- **Google Drive:** Google Drive is used to store and organize all project-related documents. It provides secure, cloud-based access for all team members, enabling real-time collaboration, version tracking, and structured file storage.
- **Google Docs:** Google Docs is used for collaborative writing and editing of documents such as the Vision Document, Software Development Plan, and Use-case Specifications. All members can comment, suggest changes, and edit simultaneously, helping ensure accuracy and consistency.
- **Git & GitHub:** Git is used for version control of the source code, while GitHub serves as the team's remote repository and collaboration hub.
 - GitHub repository structure:
 - **Folder 'src':** contains all source code for the SoulNote web application, including frontend and backend.
 - **Folder 'docs':** stores finalized documents such as the Project Description, Vision Document, and other official submissions.
 - **Folder 'pa':** holds PA-related folders and files, organized per submission cycle.
 - GitHub is also used for:
 - Commit history tracking
 - Branch management for parallel development
 - Backup and synchronization of documentation and code
 - Shared access among team members and instructors

SoulNote	Version: <1.1>
Revised Software Development Plan	Date: 11/06/2025
<document identifier>	

- **Trello:** Trello is used for sprint and task management. The team uses lists such as "Backlog", "To Do", "In Progress", and "Done" to visualize and monitor progress. Each task card includes:
 - Description
 - Assigned member
 - Due date
 - Label for sprint or priority
- **Google Meet / Facebook Messenger:** The team leader sends Google Meet links and agenda through Messenger ahead of meetings.
 - **Google Meet** is used for weekly meetings, sprint reviews, and demos.
 - **Facebook Messenger** is the team's primary channel for day-to-day communication, reminders, and quick clarifications