<Memoir>

Software Development Plan

Version <1.2>

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Author** |
| 30/05/2025 | 1.0 | 1. Introduction  2. Positioning | Phạm Nam Huyên |
| 31/05/2025 | 1.1 | 4. Product Features  5. Non-Functional Requirements | Huỳnh Tuấn Phong |
| 31/05/2025 | 1.2 | 3. Stakeholder and User Description | Nguyễn Thị Trà My |

Table of Contents

1. Introduction 3

1.1 References 3

2. Positioning 3

2.1 Problem Statement 3

2.2 Product Position Statement 3

3. Stakeholder and User Descriptions 3

3.1 Stakeholder Summary 3

3.2 User Summary 3

3.3 User Environment 3

3.4 Summary of Key Stakeholder or User Needs 3

3.5 Alternatives and Competition 3

4. Product Overview 3

4.1 Product Perspective 3

4.2 Assumptions and Dependencies 3

5. Product Features 3

6. Other Product Requirements 3

Vision (Small Project)

# Introduction

* The purpose of this document is to collect, analyze, and define high-level needs and features of the Memoir app.
* It focuses on the capabilities needed by the stakeholders and the target users, and why these needs exist. The details of how the Memoir fulfills these needs are detailed in the use-case and supplementary specifications. It acts as a shared understanding among team members and stakeholders, ensuring everyone is aligned on the project's objectives and desired outcomes.

# Positioning

## Problem Statement

| The problem of | In today's fast-paced digital world, people maintain an increasing number of personal and professional relationships, yet there is no dedicated tool that helps them systematically remember, organize, and revisit meaningful details about these connections. |
| --- | --- |
| affects | forget important dates, shared experiences, or even the people themselves. |
| the impact of which is | weakened relationships, reduced emotional connection, and missed opportunities to build or maintain deeper bonds. |
| a successful solution would be | “Relationship Memory Assistant” – an application that helps users preserve, organize, and maintain deep, personalized social connections through emotional memory support and relationship context management. |

## Product Position Statement

|  |  |
| --- | --- |
| For | absent-minded people people with many relationships |
| Who | want to have the habit of keeping memories with their friends |
| The (product name) | **Memoir** |
| That | is an app that helps you remember and organize personal relationships by storing all meaningful details in one place. |
| Unlike | Capacities, Dextr, HumanNotes, Notion |
| Our product | Visualize social connections with graphs  Have clear ADMIN permissions  Focus on **personal relationships** – not just generic notes, by remembering emotional & contextual details |

# Stakeholder and User Descriptions

## Stakeholder Summary

|  |  |  |
| --- | --- | --- |
| **Name** | **Description** | **Responsibilities** |
| Mr. Mai Anh Tuan  Mr. Nguyen Van Vu | Supervisor\_oversees and supports the development team | * Monitor project progress, deadlines, and resource usage to keep the project on track. * Provide feedback and guidance during development. * Assist in solving technical or organizational issues. |
| Group 03 | Development Team \_student group developing the Memoir application | * Analyze requirements, design system architecture and software components. * Develop, test, debug, and deploy the application. * Write technical documentation, user manuals, and present progress and project outcomes. * Ensure the product meets the specified requirements, quality standards, and deadlines. * Continuously evaluate and improve the product based on user feedback and project supervisor. |

## 

## User Summary

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Description** | **Responsibilities** | **Stakeholder** |
| Group 03 | Admin\_system administrator responsible for coordinating and maintaining the stable operation of the Memoir application | * Manage user accounts (activation, locking, password resets). * Monitor usage statistics and storage capacity. * Handle user feedback and error reports. * Send system notifications and perform regular data backups. * Oversee system errors and application performance monitoring. | Group 03 |
| Individual User | Individuals who need support in managing and organizing information about their personal relationships, including friends, family, and colleagues. | * Store and update detailed information about each person (name, birthday, photo, meeting place) * Add personal notes about preferences, impressions, and type of relationship * Categorize contacts using a tagging system (e.g., #family, #work\_friends, #class12A1) * Set and manage reminders for important dates * Use the calendar feature to track special events * Synchronize data across devices | Self-represented |
| Professional User | Professionals such as teachers, HR staff, or managers who need to manage information about multiple people in a work environment. | * Manage detailed information about students/employees/colleagues * Track important work-related events * Use the relationship visualization feature to understand group structures * Create and manage tags appropriate to the work environment * Use the map feature to track meeting/gathering locations * Ensure data security and synchronization | Educational institutions and businesses |

## User Environment

* Most target users operate independently or as part of small teams, managing only their own contact information. However, users in specific roles—such as educators or HR staff—may need to manage contacts for larger groups, often ranging from 10 to more than 100 individuals.
* Users update information continuously and on an as-needed basis, such as when they meet a new person or obtain new details. A typical user may spend 5–15 minutes updating contact details or setting reminders, with occasional longer sessions dedicated to reviewing or organizing tags and managing relationship structures.
* The application is expected to be used primarily in mobile (Android). Users may access the system while at home, in the office, or on the move.
* Currently, users typically rely on separate basic tools such as contact lists, calendar apps, and simple note-taking applications. However, these tools lack integration and advanced features.
* The proposed application addresses these limitations by offering a unified platform with enhanced capabilities:
* Rich contact profiles with Markdown-formatted notes
* Automatic reminders and calendar view for important dates
* A tagging system to organize social groups
* Visual mapping of meeting locations
* Graph-based relationship visualization
* Cloud synchronization and backups for data consistency
* A well-organized UI that supports quick access to features and minimizes user effort

## Alternatives and Competition

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Competitor** | **Strength** | **Weakness** |
| **1** | Note Apps (Apple Notes, Google Keep) | * Easy to use, multi-platform synchronization (iOS, Android, Web) * Supports quick notes with images and audio * Deep integration with Apple and Google ecosystems | * Not specialized for managing personal relationships, lacks advanced tagging features * Missing birthday/event reminders * No map or relationship graph features |
| **2** | Obsidian | * Strong for managing linked Markdown notes, easy to create knowledge graphs * Highly customizable with plugins * Fully supports offline use | * Interface is not user-friendly for non-technical users * Lacks reminders and personal event management * Not designed for managing contacts or social relationships |
| **3** | Notion | * Versatile, allows creating complex databases and varied content types * Supports tagging, categorization, and inter-page linking * Suitable for project management, personal and team notes | * Quite complex interface, requires time to learn * Not optimized for birthday reminders or intuitive relationship management * No support for meeting location maps |
| **4** | Logseq | * Powerful graph-based note management, supports Markdown, easy to build idea networks * Open-source and offline support * Extensible with plugins and customization | * Complex interface for new users * Lacks reminder or personal event features * Not optimized for contact or social network management |
| **5** | Evernote | * User-friendly interface, easy to use * Multi-platform support, fast cloud sync * Supports various note types including images, documents, audio * Strong search capabilities | * Free version limits features * No advanced tagging system for relationship management * Missing birthday reminders and relationship map features |

# Product Features

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Feature** | **Description** | **Priority** |
| **1** | Notes | save and display information about the person users are noting. (markdown can be used) | High |
| **2** | Support image | Together with text, users can also add image into their note for easier visualizing | High |
| **3** | Calendar | Calendar displays timestamp data in notes (anniversaries, debts, etc.) | High |
| **4** | Notification | Send notification to remind user of important event mark in the calendar | High |
| **5** | Tag & Filtering | Users can group note together using tag for easy management (#workfriend, #classmate2023, #family, #enemy) | High |
| **6** | Map | The map displays the location data contents in the note. | Medium |
| **7** | Graph | Users display notes data in graph form. Represents connections between people. | Medium |
| **8** | Friend Statistics | Users statistics, e.g. how many people you know, how many people you met over the summer, how many more people at work than outside of work, the gender ratio, etc. | Medium |
| **9** | Full offline support | While not connected or sign-in, users can still fully use all the offline feature of the app | High |
| **10** | Registration | Users can create accounts by create username and password or using email, phone number or Google account. | High |
| **11** | Cloud Storage | Users can sign-in an account to store data on our server cloud | High |
| **12** | Data synchronize | When signed-in and connected, update on users note will get updated on the server automatically. | High |
| **13** | Publish | Users can choose a specific note to publish on the app domain as a form of profile or CV | Low |
| **14** | Feedback | Users can submit feedback forms to the app. Admin can then see these feedback | High |
| **15** | Server Management | Admin have full control over the server of the app | High |
| **16** | User Statistics | Admin can see statistic about many users | High |
| **17** | User Management | Admin manage user account deactivate, reset password, email user… | Medium |
| **18** | Receive server reports | Admin can read server status, log and crashes | Low |

# Non-Functional Requirements

* **Performance:**The application should deliver a smooth, responsive user experience in real time, optimized for low battery consumption across mobile platforms.
* **Platform Compatibility:**Development will begin with mobile platforms (Android/iOS), with potential expansion to other platforms if feasible.
* **Hardware Requirements:**The app should run efficiently on mid-range and older mobile devices, avoiding reliance on high-end hardware to ensure accessibility for a broader user base.
* **User Experience:**Memoir prioritizes quality and simplicity over feature bloat. It is not designed to be an all-in-one solution but rather a focused tool that works seamlessly out of the box—no login, complex setup, or onboarding steps required.
* **Ease of Use:**A clean and intuitive interface will guide users through notes, relationships, and event tracking with minimal effort, emphasizing a frictionless and accessible design.
* **Reliability:**The backend service should maintain a minimum of 99.9% uptime to ensure consistent availability and user trust.