SpUStify

Vision Document

Version 1.0

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 24/06/2023 | 1.0 | Initial version | Dương Trường Bình |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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 introduction of the Vision document provides an overview of the entire document. It includes the purpose and references of this Vision document.

# Positioning

## Problem Statement

| The problem of | publishing, listening to, and downloading free songs online |
| --- | --- |
| affects | music artists who want to publish their songs and listeners who want to listen to or download songs online for free |
| the impact of which is | unpopular music artists who want to get famous but don’t have the budget to publish their songs on big music platforms while listeners on big music platforms have to pay so they can listen to or download songs. |
| a successful solution would be | helping unpopular music artists to publish their songs free and listeners to listen and download free songs by creating a free music platform |

## Product Position Statement

| For | artists and listeners |
| --- | --- |
| Who | publish and listen/download songs free |
| The (product name) | hailua.com.vn |
| That | helping unpopular music artists to publish their songs free and listeners to listen and download free songs |
| Unlike | the current music platforms, zingmp3.vn, open.sportify.com |
| Our product | allows people to publish and listen/download songs freely |

# Stakeholder and User Descriptions

## Stakeholder Summary

| **Name** | **Description** | **Responsibilities** |
| --- | --- | --- |
| Developers | Developers build a website that meets all user requirements | Researching, designing, implementing, and managing software programs. |
| Teachers | Teachers introduce the course and guide the software design processes | Teachers support students understand the methodology of Software Engineering |
| Students | Students studying at HCMUS | Giving feedbacks about architecture and interface of a website |
| Artists | Artists share their own music with the community through SpUStify | Contributing to building SpUStify's music store |

## User Summary

| **Name** | **Description** | **Responsibilities** | **Stakeholder** |
| --- | --- | --- | --- |
| Client | End consumers of the web, may be a member or not | View songs, playlists (only a feature for who is not a member)  Browse and search for playlists and songs  Add songs, playlists  Likes song and playlists | Self - represented |
| Administrator | Main managers of a website | View clients (Full Name, DOB , Email, Phone )  Add a song or a playlist to the website  Ensure the leaderboard on the website is up-to-date  Delete clients | Self - represented |

## User Environment

* SpUStify is a community-oriented music platform targeting educated, computer literate individuals who own personal computers. Users can visit the website to explore a wide range of songs and playlists, seeking flexibility in their music listening experience and sharing their own music with the community.
* The number of users is expected to grow significantly in the future, driven by factors such as staying up-to-date with current trends, implementing effective marketing campaigns, and leveraging social media platforms like Instagram, TikTok, and Twitter for music promotion and sharing.
* In addition, artists and individuals seeking self-development opportunities may utilize the website to discover new music.
* Initially, the website will have a limited release exclusively for university students. However, future plans involve marketing subsequent releases to schools, colleges, and universities. SpUStify is designed to be expandable, allowing for scalability and accommodating a growing user base. User community data will be organized in easily accessible table formats upon system installation, facilitating efficient management.
* The administrator will require a browser-enabled device to perform various tasks such as categorizing songs and playlists, monitoring system status, managing artists and clients, and generating reports.

## Alternatives and Competition

* Some general examples of alternatives and competitive choices that stakeholders might consider in the context of a music website like SpUStify:
  + Competitor's product: Stakeholders may consider purchasing or subscribing to a competitor's music website or streaming platform. Examples of competitors in this space include Spotify, Apple Music, Amazon Music, and YouTube Music. The strengths and weaknesses of each competitor can vary, but they typically include factors such as music catalog size, user interface, personalized recommendations, pricing, and exclusive content.
  + Building a homegrown solution: Stakeholders might consider developing their own music website from scratch, creating a customized platform tailored to their specific needs. This approach offers the advantage of complete control over features and functionality. However, it requires significant development resources, time, and ongoing maintenance costs.
  + Maintaining the status quo: Stakeholders may choose to stick with their current music website or platform, particularly if they perceive it to be meeting their needs adequately. This option avoids the potential risks and costs associated with switching to a new solution but may limit access to new features and improvements offered by competitors.
* It's important to note that the specific strengths and weaknesses of competitors will vary based on market dynamics, user preferences, and the specific features and capabilities offered by each platform. Stakeholders and end users would need to assess these factors based on their own priorities and requirements before making a decision.

# Product Features

| No. | Feature | Description | Priority |
| --- | --- | --- | --- |
| 1. | Browse and search | Users enter the song title and author's name to search for related works such as songs or playlists containing the song. | Medium |
| 2. | View detailed information | Users have the ability to access detailed information about the song and playlist (title, creation date, content), and some information about the author (name). | Low |
| 3. | Leaderboard (by likes, listens) | Based on user ratings (number of likes, number of listens), a ranking table is created for the songs. | High |
| 4. | Show the favorite songs and playlists of each account | List the songs in the playlists that the user has liked (handle the like button to avoid cases where the button is pressed multiple times and continues to loop). | High |
| 5. | Show the listening history of each account | List the songs in the user's browsing history (approximately 10 songs and playlists with the most recent access dates). | Medium |
| 6. | Upload new playlists or songs | Upload songs (song title, song file) and playlists (playlist name, list of songs) - handle cases where there are duplicate song or playlist names. | Medium |
| 7. | Update existing playlists | Update new songs (in the correct format) or delete old songs. | Medium |
| 8. | Delete playlists and songs | Delete songs or playlists with the user's permission (ask the user again when they press the delete button). | Medium |
| 9. | Playlist and song recommendations | Suggest to the user songs that they listen to frequently or that are on the ranking table. Also, suggest playlists with a high total number of song plays. | High |
| 10. | Calculate and generate reports | Calculate the number of registered user accounts, user access counts (monthly, weekly, etc.), number of returning users, etc. | High |
| 11. | Sign in / Sign up / Update password | Users can register/login using an account and password, and update their password. | Low |
| 12. | User profile | Users can update some information such as their full name, date of birth, email, phone number, etc. | Low |

# Non-Functional Requirements

Security:

* The website must have robust security measures to protect user information, including accounts, passwords, and personal data..

Speed and Performance:

* The website should have fast page loading times to ensure a smooth user experience.
* It should support a moderate number of concurrent users without compromising performance.

Maintenance and Scalability:

* The website should be easy to maintain and upgrade without causing disruptions to user activity.

Hardware Requirements:

* Computer or Mobile Device:
  + Desktop Computer: A personal computer with an equivalent Intel Core i3 or AMD processor, 4GB of RAM, and sufficient free storage space to store relevant files and data.
  + Mobile Device: A smartphone or tablet running iOS or Android operating system with the latest version of the operating system and web browser.
* Web Browser:
  + Popular web browsers such as Google Chrome, Mozilla Firefox, Safari, or Microsoft Edge. It is recommended to use the latest version of the web browser for the best experience and support for advanced web technologies.
* Network Connection:
  + The computer or mobile device needs to have an Internet connection. This can be a wired connection (such as Ethernet) or a wireless connection (such as Wi-Fi or 3G/4G).

Platform Requirements:

* The website must be compatible with popular web browsers such as Google Chrome, Mozilla Firefox, and Safari…

Environmental Requirements:

* The website should operate reliably in various server environments such as Windows, Linux, or macOS.
* It should function stably and reliably in different network environments, including LAN and the Internet.

Product Quality:

* Performance: The website must meet requirements for fast response times and high processing capability.
* Reliability: The website should be stable and free from frequent errors during regular usage.
* Fault Tolerance: The website should have the ability to recover from failures, such as network disruptions or software errors.
* Usability: The user interface must be user-friendly, intuitive, and easy to understand.

Design Constraints and Dependencies:

* The system must adhere to HTML, CSS standards, and web design principles to maintain consistency and compatibility.
* Identify and adhere to constraints related to integrating technologies such as HTML, Tailwind CSS, Django, and PostgreSQL.

Documentation Requirements:

* User Guide: Provide detailed documentation on how to use the system's features and functionalities.
* Online Help: Provide access to online help resources for users to search for answers to common questions.
* Installation Requirements: Provide detailed installation instructions for deploying the system in different environments.

Priority of Other Product Requirements: The descending order of priority

* Functionality: The primary priority is to ensure that the website functions properly and allows users to download music files seamlessly. This includes providing a user-friendly interface, search functionality, and the ability to preview and select music for download.
* Performance and Speed: The website should prioritize fast download speeds and efficient performance to enhance user experience. Optimizing file transfer speeds and minimizing latency will contribute to a smooth and responsive downloading process.
* Security: Ensuring the security of user data and protecting against unauthorized access is of utmost importance. Implementing secure authentication mechanisms, encryption protocols, and maintaining a secure infrastructure should be a priority.
* User Experience (UX): Prioritizing a positive user experience is crucial for engagement and retention. This includes intuitive navigation, responsive design for different devices, visually appealing layouts, and clear instructions throughout the downloading process.
* Compatibility: The website should be compatible with various web browsers, operating systems, and devices to ensure a wide user reach. Cross-platform compatibility and responsive design will enable users to access the website from different devices without issues.
* Accessibility: It is important to consider accessibility guidelines and ensure that the website is accessible to users with disabilities. This includes providing alternative text for images, keyboard navigation support, and other accessibility features.
* Scalability: Considering future growth and the ability to handle increased traffic and downloads is important. Designing a scalable architecture and implementing efficient backend processes will allow the website to handle high volumes of traffic and downloads without performance degradation.
* Documentation and Support: Providing comprehensive documentation, such as user manuals, online help, and FAQs, will assist users in navigating and utilizing the website effectively. Additionally, offering responsive customer support channels, such as email or live chat, can help address any user queries or issues.