

Vaultify

"Vaulting music memories."



Sprint 1 Deliverables

Software Design and Documentation

1:30 PM - 3:35 PM Section

Professor John Sturman

The Band:

Matthew Bui, Michael Lam, Dillon Li, Michelle Li, Thomas Orifici

14 June 2024

Vaultifyinquiries@gmail.com

Executive Summary

Elevator Statement

At its core, Vaultify is a web application that aims to transform the listening experiences of Spotify users by providing a “vault” containing each user’s favorite songs and memories. Vaultify curates highly customizable monthly playlists containing the user’s favorite songs and automatically adds them to the user’s Spotify library for easy playback. In addition, the platform acts as a hub for musical reminiscion, providing a listening timeline that allows users to insert personal notes and customizable AI playlist covers to keep track of their journey with music. Vaultify aspires to portray the personal story that each user’s listening history tells.

Problem and Proposed Solution

For many Spotify users, the process of curating a playlist can be daunting. It’s hard to keep track of which songs you’ve been enjoying the most, and many users struggle to curate their favorite songs. To add on to this, Spotify does not offer any way to track and revisit your listening history for a given month, leaving users with no means to reminisce and rediscover their old music tastes.

Vaultify aims to solve these problems by generating monthly playlists based on the user’s monthly listening history and exporting those playlists directly into the user’s Spotify account. This will provide users with a playlist perfectly catered to their current tastes, while also giving them a record of how their listening has changed over time. Additionally, the Vaultify website provides a musical vault that allows users to dive into their previous playlists and add personalized notes that they can look back on later.

Overview of Development Process

Vaultify will be adhering to an Agile development process. This means that developers will work across the entire project in a Scrum team and that development will be split into iterative sprints. Additionally, Vaultify will be adhering to development best practices by utilizing a GitHub repository for version control and bug/task tracking, implementing eslint to uphold JavaScript coding standards, following object-oriented design methodologies, and structuring code using tried and true design patterns.

In terms of technologies, Vaultify will be utilizing HTML, CSS, JavaScript, Node.js, React, Express.js, and Mongo database alongside the Spotify web API.

Overview of Features

Vaultify offers a range of features designed to enhance the music listening experience on Spotify for users. Some of the app's key features include authentication and Spotify API integration for seamless login, monthly playlist generation based on listening habits, and personal notes for annotating playlists and songs. The "Vault" page provides a timeline view of playlists and notes, while music emotional analysis and AI-generated playlist covers add a personalized and unique touch to every individual user's experience on Vaultify. Users can easily share and export playlists, receive email notifications for new monthly playlists, and enjoy gamification elements through "Achievements" on their Profile page for various listening milestones.

Overview of Business Case

Vaultify targets Spotify users who are passionate about music, wish to revisit old favorites, and are interested in detailed analysis of their listening habits. It stands out by offering unique features such as personalized notes, automated monthly playlist generation, and emotional analysis of music, differentiating it from competitors like Stats for Spotify, Stats.fm, Receiptify, and Obscurify. However, the product's success faces risks, including competition from established platforms, reliance on Spotify, and technical challenges in hosting and scaling the application. Despite these challenges, Vaultify aims to deliver a highly personalized and emotionally engaging music experience.

Overview of Project Schedule

The development of Vaultify will occur in 10 one-week-long sprints, with each sprint being dedicated to a different set of features. The development will start with basic platform functionality, including base UI, Spotify connectivity, and playlist generation. This will be followed by an interim release. Next, the team will focus on personalization, implementing a personalized vault, adding playlist customization features, and utilizing AI to generate custom playlist covers. This will be followed by a beta release. Development will conclude with the implementation of an extensive testing suite and a final release.

Overview of Project Risks

Despite the confidence of the development team, proceeding with this project does present several risks, both for the project and the product. Risks for the project include the team's unfamiliarity with the project's technologies and the project's short development cycle. Risks for the product are finding a host for the site, the site's dependence on Spotify, and standing out in comparison to other Spotify tools.

Features

Feature Name	Description of Function
Authentication and Spotify API Integration	Users will log in via Spotify API/OAuth, and login will be saved for future usage.
Monthly Playlist Generation	Vaultify generates a playlist based on the user's most listened-to songs of the month, with a new playlist for each month (e.g., January 2024, February 2024), and allows you to export the playlist to the user's Spotify library.
Personal Notes	Users can annotate their playlists or songs with personal notes and memories, reminiscent of a diary.
“Vault” Page	Users will have access to a “history” page where they can view their playlists and personal notes that have been generated by the app. This page will be structured like a timeline.
Music Emotional Analysis	The app determines the mood of the songs users listen to most frequently that month and displays it to the user.
AI-Generated Playlist Covers	Using the musical emotional analysis results, the app will generate a custom AI playlist cover. Users will have the option to input their own thematic preference for the generated playlist cover (i.e., abstract art, a specific color, cats, landscape).
Share and Export Playlist	Users have multiple options for sharing their playlist, such as saving an image of their playlist, generating a link, and exporting the album back onto Spotify.
Email Notification Reminders	Vaultify will allow users to activate automatic playlist generation and email reminders that inform users to check out their new playlist each month.
“Achievements” on Profile Page	As Vaultify strives to make the music experience fun and refreshing, it provides gamification elements where users can earn achievements for various listening milestones or challenges (i.e., listening to a new genre, discovering new artists that month, etc.).

Business Case

Niche

At its face and basic functionality, Vaultify's user base will be composed of Spotify users. Within that niche, Vaultify targets users who are particularly passionate about music and keen on revisiting old favorites they may have forgotten. The platform also caters to users who want to listen to their current favorite songs without having to manually curate a playlist. Vaultify also caters to nostalgic music listeners through features like the vault page and personalized notes. Finally, it attracts users who like to get into the statistics of things, as Vaultify provides detailed analysis and insights into their listening habits and preferences.

Competition

1. [Stats for Spotify](#)

Stats for Spotify provides several features revolving around the user's listening history. Users can view their most listened-to tracks, artists, and genres for the last month, 6 months, or year. Users can also export a playlist from their personal charts. Stats for Spotify has an advantage over our app as they give the user more types of data that they would be interested in than Vaultify does not directly provide, such as more detail regarding information about a user's top artists and genres. Our Vaultify app differs in the fact that we can save the history of our users in the form of playlists automatically rather than having to consistently go back and relog a new playlist every month, as well as that we can look at music listening history in the past.

2. [stats.fm](#)

Stats.fm displays your Spotify listening statistics in a similar manner to Stats for Spotify. However, Stats.fm emphasizes user profiles and sharing. There is also a unique import process that brings in your complete streaming history for viewing. The way we differ from the history that is logged in Stats.fm is that Stats.fm provides numerical data and stats while our app provides a more emotional connection towards the user, through the usage of notes that can be written and looked back upon on the user's profile page as well as their very own Vault page.

3. [Receiptify](#)

Receiptify displays statistics about your listening history such as top tracks or artists in a cute receipt format. Other than that, Receiptify doesn't provide much else besides a sharable format of song history. It also doesn't display anything else besides the intervals: the past month, 6 months, and year. Our Vaultify plans

on being able to view a greater variety of time intervals such as 2 months ago, 3 months ago, and more as we are planning to make it a more customizable experience that appeals to a larger user base. This also allows our app to cover a more holistic and accurate Vault history for our users as they would have their own past music listening history data to compare to and keep track of.

4. [Obscurify](#)

Obscurify gives users information about how unique/obscure their music tastes are. It displays a graph of your obscurity levels over time and displays unique statistics such as the mood and danceability of your music preferences. Obscurify has a very basic tracker for songs, recording only when you last logged into the website. Because our Vaultify app's functionality and technical setup keeps the user logged in, it updates with more real time statistics and gets rid of the hassle of logging in every single time on our user's end.

How Does Vaultify Differ?

While other platforms provide extensive information on a user's listening history, Vaultify sets itself apart by displaying how users' tastes have changed over time through the vault. Unlike Vaultify, other platforms also don't provide the extensive personalization features that Vaultify will provide, such as custom notes and playlist covers that make each user's vault appear and feel special. On top of this, while you have the option to generate playlists based on your habits on other platforms, none offer the unique monthly playlist generation that Vaultify provides. This, combined with the ability for users to share their vaults and playlists with friends, will keep users engaged with the site over long periods of time.

Value for the Effort

Vaultify provides value for users by giving them a highly personalized platform to look back on their journey with music. Additionally, by generating tailored playlists, it ensures that users always have access to music that resonates with their current tastes and moods. At its core, Vaultify aims to foster creativity and provide personality to the users' library. This can be seen through features such as emotional analysis of songs and attaching personal notes and memories to them. This transforms playlists into personal narratives that users can share with their friends and loved ones.

The level of customization and emotional engagement shown through the features of the Vaultify web application makes music more of a meaningful and enjoyable experience for users.

Risks of Product Success

While Vaultify shows great promise as a product, there are several risks to its success. One such risk is the existence of other platforms that analyze users' Spotify libraries. If Vaultify doesn't do enough to differentiate its platform and provide unique features, users may choose to stick with other platforms that have existed for longer. Another risk to the product is the development team's ability to find a host for its website. While RPI servers are available, using one would significantly limit the scope and user base of the application. In addition to this, Vaultify as a platform is reliant on Spotify and its relevance in the market.

Project Stakeholders

Michelle Li

Product Owner

Michelle is a junior at RPI dual majoring in Computer Science and Information Technology & Web Science. Over the past two years, she has taken many classes related to both her majors to develop her technical skills. During the current Arch summer semester, she is enrolled in one other Computer Science course, Operating Systems, to further her expertise in her field. Michelle has a rich background in music, having played the piano and violin, and also enjoys singing in her free time. Her passion for music drives her to create an application that caters to both those who are nostalgic about their musical past and those eager to explore new music. Michelle thrives on the challenges presented by programming and web development and is eager to gain valuable insights alongside her team. She is dedicated to sharing her love of music through innovative technology, aiming to develop a product that resonates with all music enthusiasts.

Thomas Orifici

Developer

Thomas is a junior at RPI majoring in Computer Science. He's planning on minoring in music and plays the guitar and bass. When Thomas heard Michelle's pitch for Vaultify, he immediately knew that he wanted to be a part of the team. He's felt the need for such a service to organize his own musical library and is excited to work on a project that aligns so closely with his own interests and needs.

Michael Lam

Developer

Michael is a junior at RPI majoring in Computer Science. Coming to RPI unequipped with knowledge of Computer Science, he has developed many skills since his arrival through a numerous number of courses provided at RPI. Although his experience with Vaultify's stack is minimal, he knew that working on this project would further develop his skills as a software engineer.

Dillon Li

Developer

Dillon is a junior at RPI majoring in Computer Science and through this project, Dillon aims to enhance his programming and team-collaboration skills. Besides programming, he is also passionate about music. As a daily Spotify user, he is very interested in developing and using features that are not typically accessible to general users.

Matthew Bui

Developer

Matthew is a junior at RPI majoring in Computer Science. As an avid music listener and dancer, Matt often utilizes Spotify on a daily basis and lives life through music. In addition to this, Matt is trying to get involved with more coding projects to help deepen his knowledge and enhance his abilities as a software developer. When Matt was pitched with the idea of Vaultify, he knew instantly that he wanted to work on it as it aligns with both of his interests.

John Sturman

Professor

John Sturman is a professor at Rensselaer Polytechnic Institute, where he teaches the course Software Design and Documentation to two medium-sized classes of around 40 students each. With certifications in Agile technology and other learning technologies, Professor Sturman brings a wealth of knowledge and experience to the project. His expertise will be invaluable in ensuring the successful rollout and adoption of Vaultify, providing critical insights into software design principles and best practices.

Teaching Assistants

Kushal Bhandari, Sharmishtha Dutta, Zhenhan Huang, and Aitazaz Khan

Kushal Bhandari, Sharmishtha Dutta, Zhenhan Huang, and Aitazaz Khan all serve as teaching assistants for the Software Design and Documentation course. Each of them is an experienced professional with a deep understanding of software development processes. Their combined expertise will be instrumental in the development of Vaultify, offering support, guidance, and technical knowledge to the team throughout the project.

Mentors

Dong Lin and Chris McCarthy

Dong Lin and Chris McCarthy are senior students at Rensselaer Polytechnic Institute and are mentors for the Software Design and Documentation course. Their prior experience with the course and their extensive knowledge in software engineering and programming will provide valuable guidance and mentorship to Vaultify's development team. Their support will help ensure that the Vaultify team adheres to best practices and meets project milestones effectively, each contributing significantly to the project's overall success.

Project Risks

During the development of Vaultify, several risks pose a threat to the success of the project. One such risk is the developers' unfamiliarity with technologies such as React, Node.js, HTML, and CSS. While the team is confident in their learning abilities, if they can't overcome this challenge, the team will not be able to turn their vision into a coded reality. Another such risk is the potential for scope creep due to the team's enthusiasm and numerous ideas for the project. Although this passion can be a positive aspect, it might also cause the project to extend beyond its original scope, leading to uncontrolled continuous expansion. This can result in delays and resource strain, ultimately jeopardizing the project's completion.

On top of that, the risk of inadequate project management and communication among team members could result in misaligned goals, missed deadlines, and reduced productivity. In order to avoid this problem, the team will need to proactively implement effective risk mitigation strategies, such as clear project planning, regular progress reviews, and efficient communication channels.

Additionally, the development team for Vaultify has not worked together before. If they fail to come together as a team, efficiency could be reduced, leading to a worse final product.

Another risk is the shortened development period that the summer semester presents. Since the summer semester is shorter than usual, the team will have less time to implement their product. This may lead to a reduction in the number of features that get implemented.

Finally, one last risk is the team's unfamiliarity with Agile methodology and the development process. If the team isn't able to properly adhere to these principles, their progress may become less organized, which would be reflected in the product presented at the end of the semester.

User Scenarios



Suzy Brown, Business Systems Analyst

Suzy is a 23 year old white-collar office worker. Last year, Suzy finished her Bachelor's degree at New York University Stern School of Business in Business Analytics. Now, she works at JP Morgan Chase & Co as a Business System Analyst, making 160k a year. She lives in New York City in a small apartment in Tribeca and listens to music on her daily commute to and from work. Suzy is known by her friends and coworkers for being very disciplined, reliable, and career-oriented. She already makes a very good living, but she places a lot of value in her career and wants to do the best job that she can.

In her day-to-day life, Suzy works a 9 to 5 and typically goes to the gym right after work. She's usually very tired after this and spends the rest of her evening relaxing, spending time with her family and friends, or winding down with a good Netflix show. Oftentimes on her commute to work, Suzy doesn't know which music she wants to listen to and is dissatisfied with her library, so she occasionally shares this frustration of hers with her close friends. Despite her love for consistency in her schedule, having been influenced by her college friends, Suzy has begun to develop a taste for getting out more on the weekends and trying new things with them. She's now always looking for a new restaurant or activity to try in her downtime, and above all else, she loves the feeling of exploration and expanding her worldview.

Goals:

1. Suzy wants a quick and easy way to generate a playlist with her current favorite songs to accommodate her busy lifestyle.
2. Suzy wants this playlist to be added to her Spotify library.
3. Suzy wants a new playlist to be automatically generated for her every month to reflect her changing tastes.
4. Suzy wants to be able to adjust the length of her generated playlists to match the length of her commute.
5. Suzy wants to be reminded when a new playlist has been added to her library so that she knows when to check it out.

User Scenario:

Suzy decides to use Vaultify to curate a playlist of her favorite songs from the past month. She visits the website and is greeted by an introductory screen that welcomes Suzy to Vaultify. It provides a brief description of the playlist curation services Vaultify provides and features a green “Login with Spotify” button underneath the text. She clicks the button and is automatically redirected to Spotify’s official login page. She fills in the username and password fields and presses the green “Login” button. She is then prompted to give Vaultify permission to view and modify her Spotify library. She clicks the green “Agree” button. This redirects her to a new Vaultify page, labeled “Home”. The page displays her basic user information such as her username, email, and profile picture. A gray navigation bar is vertically displayed on the left side of the screen, displaying the Vaultify logo and buttons labeled “Home,” “Playlist Generation,” “The Vault,” and “About Us.”

Suzy presses the “Playlist Generation” button on the navigation bar and is redirected to the associated page. This page features several customization fields in a vertical list format followed by a green “Generate Playlist” button. The first field reads “Length (minutes):” followed by a text box. She inputs 45 minutes, the length of her commute. The next field reads “Cover Theme:” followed by a dropdown menu. “None” is already selected. She leaves this selection. The next fields are “Activate Monthly Generation” and “Activate Email Notifications” and both are followed by a checkbox. She checks both boxes. She then presses the green “Create Playlist” button. This redirects Suzy to a “Playlist Successfully Generated” page, which shows a success message and a clickable cover for a playlist labeled May 2024. She clicks on the playlist and is redirected to a Spotify page containing her monthly playlist, where she can listen to and try out her newly generated playlist.



John Carson, Musician

John Carson is a 34-year-old full-time music producer and recording artist. As a dedicated father, John balances his family life with a highly demanding and fast-paced career in the rap and hip-hop music industry. Known for his dark and edgy production style, he has carved out a niche for himself in the industry, producing music for some of the best charting artists nowadays. His most notable work includes tracks for the hit rap artist DRE Lane. Professionally, John Carson charges between \$75,000 and \$150,000 per track, a reflection of his high demand and the quality of his work. His innovative and out-of-the-box style has made him one of the

most influential producers in his niche. He currently owns and lives in a luxurious penthouse in Manhattan, New York with his wife Camila and son Lucas. Outside of his professional life, John enjoys a few personal hobbies that provide him much-needed rest from his intense work schedule. When he's not working, John enjoys playing tennis and feeding the birds in the park. Camila often joins him in these activities, and together, they enjoy exploring the city's cultural scene, attending art exhibitions, and dining at new restaurants. Their son, Lucas, is a bright and energetic 8-year-old who shares his father's love for music. Lucas is learning to play the piano and often spends time in John's home studio, where he observes his father at work and experiments with creating his own tunes.

Goals:

1. John wants to efficiently organize his top songs at any given time in order to keep track of his musical inspirations.
2. John wants to be able to mark notes down on specific songs that particularly struck him.
3. John wants to share his new playlists and notes with his friends DJ Ace, Lil Trey, and Tay Kim. This will enhance collaboration amongst his closest group of rapper friends and allow him to get easily accessible feedback and ideas from his peers.
4. As John admits he is not good at picking images for his playlists, he wants an accessible way to stylize his playlists with covers.
5. John wants his listening history to be organized and display information in even increments.

User Scenario:

John decides to try Vaultify one day to streamline his music curation and production process. John follows the same initial process as Suzy, first viewing the welcome page, then logging into his account via Spotify OAuth, then navigating to the “Playlist Generation” page from the home page. However, John chooses some different customization options. He inputs 100 minutes for length, as he wants to keep track of as much music as he can. Additionally, John chooses a cover theme for his playlist by selecting “Swag” from the “Cover Theme” field dropdown menu. Like Suzy, he opts for monthly generation and email notifications. John then clicks the “Generate Playlist” button at the very bottom of the page. On the “Playlist Successfully Generated” page, he is greeted with a playlist with a cover picturing a dog wearing sunglasses and the title of the playlist labeled as “May 2024.”

After generating his playlist, instead of jumping right into Spotify, John navigates to the “Vault” page through the navigation bar on the left. Here, he is greeted with a vertical timeline of his generated playlists, with the most recent being at the top. Since he’s only generated one playlist, it’s a pretty short timeline. The timeline has one point on it, and it’s labeled “May 2024.” Next to this text is a clickable picture of the playlist cover for that month’s playlist. John clicks on the cover and is redirected to a page displaying the songs of the playlist in a vertical list. Next to each song name, there is a “+” icon, and hovering over the icon brings up the text “Add Note.” John clicks the “+” next to a song that inspired him that month, and the list below the song shifts down as a text box appears under the song. The text box is offset from the rest of the list and a lighter color in order to distinguish it from being a song. John types, “LOOOVVVEEEDD this one” and presses his Enter key when done. The text cursor disappears, and an alert appears on the top of the window saying “Note Saved.” John reloads the page, and the note is still there. Deciding that he wants to delete the note, he hovers over it and notices an “X” on the top right of the note. He clicks the “X,” and the note disappears. An alert appears reading “Note Deleted.”

Project Schedule

Sprint Number	Tasks
Sprint 1 6/4 @ 1:00 PM - 6/11 @ 11:00 AM	The team will complete our Sprint 1 Deliverables, including their vision statement, user scenarios, and schedule. Additionally, they will create a GitHub Repository with a skeleton React project and confirm that all team members' development environments are working. They will also establish coding standards for the project.
Sprint 2 6/11 @ 1:00 PM - 6/18 @ 11:00 AM	The team will create a welcome page with the project name, logo, description, and a “Login with Spotify” button. This button will allow users to log in to their Spotify accounts using the Spotify web API. They will also create a “home” page that displays user information and settings.
Sprint 3 6/18 @ 1:00 PM - 6/25 @ 11:00 AM	The team will complete their Sprint 4 Deliverables, including their user stories, product interface mockup, supplemental specification, deployment diagram, use cases, work breakdown structure, and updated project schedule. They will also work on implementing an “About Us” page and create a mock up for the playlist generation page.
Sprint 4 6/25 @ 1:00 PM - 7/2 @ 11:00 AM	The team will create a playlist generation page displaying a “Generate Playlist” button. They will also use the Spotify API to collect the user’s top 50 songs for the past month, generate a playlist for the user with these songs, and add the playlist to the user’s library.
Sprint 5 7/2 @ 1:00 PM - 7/9 @ 11:00 AM	The team will create a “Playlist Successfully Generated” page, with the playlist name, the playlist cover, and a success message. Clicking on the playlist name or cover will redirect the user to the playlist on Spotify. They will also create a database to store user email addresses and Spotify authorization codes.
Sprint 6 7/9 @ 1:00 PM - 7/16 @ 11:00 AM	The team will add UI to the playlist generation page to allow users to opt-in to monthly emails and playlist generation. They will then configure the app so that playlists automatically generate each month for opted-in users. They will also integrate email notifications for opted-in users.
Sprint 7 7/16 @ 1:00 PM -	The team will create a “Vault” page where users can view, add notes to, and interact with their generated playlists. They will also complete the

7/23 @ 11:00 AM	documentation for the interim release, including their design approach, sequence diagrams, CRC cards, static class diagram, and mock up interface test plan.
Sprint 8 7/23 @ 1:00 PM - 7/30 @ 11:00 AM	The team will integrate custom timeframes for playlist generation (weekly, biweekly, monthly, seasonal, yearly). They will also allow users to choose a custom playlist length (10 songs, 25 songs, 50 songs, 100 songs) and add a “New Songs Only” option for playlist generation, where only songs that are new to your top songs are added to the playlist. They will add these options into the UI of the playlist generation page.
Sprint 9 7/30 @ 1:00 PM - 8/6 @ 11:00 AM	The team will integrate AI playlist cover generation with multiple theme options based on the month and the songs that are in the playlist. They will also implement mood analysis for generated playlists. Additionally, they will complete the testing documentation for the beta release.
Sprint 10 8/6 @ 1:00 PM - 8/13 @ 11:00 AM	The team will add a “Share” feature that allows users to share their listening habits with their friends. They will also add achievements that users can unlock by using the app for consecutive months. In addition, they will fully test the product and prepare documentation for the final release.

Interim Release (7/26)

The interim release of Vaultify will contain all of the features that the platform requires to function, including automatic playlist generation with the user’s Top 50 songs, email notifications, a “Vault” page featuring previously generated playlists in a timeline, the ability to add notes to your generated playlists, and a simple playlist sharing function. The UI will include a “Welcome” page, “Home” page, “Playlist Generation” page, “Playlist Successfully Generated” page, a “Vault” page, and an “About Us” page.

Beta Release (8/9)

The beta release of Vaultify will add additional features to the interim release such as new options for playlist generation (“New tracks only,” song count, and length), AI playlist cover generation based on several selectable themes, mood analysis of generated playlists, and a share feature. The UI will be updated to account for these features and to account for feedback from the interim release.

Project Status Report

Since the creation of the Vaultify team (“The Band”), we have made substantial progress. We've established a solid foundation by getting to know each other and creating our team structure, including a team name, motto, and nicknames. We organized our communication through a Discord server and set up a GitHub repository for our code base. Our technical work includes creating skeleton code on GitHub using React/Node.js/Express.js, developing a working version of our logo, and successfully completing our deliverables for Sprint 1. In terms of technical progress, we've also linked the Spotify API and Spotify's user OAuth to access basic user information and created a skeleton frontend with a navigation bar and placeholder pages for future features. Additionally, we have solidified our project name of “Vaultify,” selected peer review metrics, crafted user scenarios, drafted a vision statement, developed a working version of our logo, established a preliminary schedule to keep us on track, and outlined our project's technical infrastructure. In class, we have also begun developing a general layout of our website architecture, as well as started to consider our design themes and website colors for both light and dark modes to ensure an enjoyable and pleasant user interface.

Our current risks involve learning the intricacies of the Spotify API, as well as ensuring all team members are proficient in JavaScript, HTML, CSS, React, Node.js, and Express.js.

Moving forward, we plan to collaboratively build on the design of the web application as a team, create mockups, and implement our core features first to ensure that we have a functioning website. Additionally, we will continue our ongoing research on how the Spotify API will allow us to edit and create playlists to enhance our application's functionality.

Contribution Summary

Matthew Bui:

Matthew did research regarding the technical stack and functionality of the Spotify API. In terms of Sprint 1 deliverables, he logged information about the competition as compared to Vaultify. He also pitched an additional feature to the app in the later stages of development with the “Most Streamed Artists” feature by playlist.

Michael Lam:

Michael developed a slogan for Vaultify and helped to develop the user personas with the team. Additionally, in terms of Sprint 1 deliverables, he outlined and completed various different product and project risks that were posed towards the project. Lastly, he developed the first working authentication page for the app, which had a bare skeleton page for a user’s general information.

Dillon Li:

Dillon contributed to the brainstorming and the development of the product’s name and logo design. Dillon also contributed to the different user scenarios for the product. In terms of Sprint 1 deliverables, he completed writing the stakeholder descriptions for the professor, teaching assistants, and mentors and added detailed information for each of the user personas.

Michelle Li:

Michelle participated in brainstorming up name ideas and collaborated with the team to select an appropriate name for the product, designed the product’s logo, and organized the GitHub repository and website skeleton for the team. As the product manager, she played a role in leading the team members to achieve their collective goals by setting up a Discord server for team communications and keeping track of completed tasks in the team’s “Status Report” document. In terms of Sprint 1 deliverables, she primarily focused on implementing the names and descriptions of the product features and descriptions, completed the project status report section, and contributed to the business case and project risks sections.

Thomas Orifici:

Thomas had a profound impact on the production of the schedule, executive summary, business case, and personas for this assignment. Additionally, he’s contributed to the organization of the team, helping to delegate tasks and cooperate to finalize the vision

for the product. He's also been pivotal in deciding which features to prioritize in development.