Software Requirements Specifications: SingAlong

Alex Oller, 243398
David Pérez, 241375
Eneko Treviño, 241679
Guillem Escriba, 242123
Max Frigols, 241156

Introducing SingAlong - The Ultimate Music Experience

SingAlong is an innovative web-based application designed to transform how music is distributed and consumed. That is the ultimate music experience for music lovers.

Whether you are a casual listener or a die-hard music fan, SingAlong brings something for everyone.

With SingAlong, you can access millions of songs from all genres, artists and eras. You can create playlists and discover new music based on your preferences. The app’s intuitive design makes it easy to navigate and find the music you love.

Features of SingAlong

SingAlong offers a range of features that makes it stand out from other music apps. SingaAlong intends to offer a close user-artist interaction where users benefit from lower expenses while artists can earn more in a common and more sustainable music ecosystem.

In other words, artists can upload easily their music content without intermediaries while users can have a fully-personalized experience with many different features, including personalized mixed playlists or direct feedback from their favourite artists, among others.

User’s Experience

Fully Personalized Playlists

Plenty of Social Features

Endless Albums to Experience

Culture of Rewarding Artists

Stakeholders

Stakeholders

1.- Regular user: The potential customer of the application. Purchase and listen to songs, albums, make donations...

Who?

- Individuals
- Companies and stores for background ambientance
- Gyms for workouts
- DJs at discoteques

Stakeholders

2.- Artist The composer of the uploaded songs. Publish songs and albums, set their prices, upload promotional material...

Who?

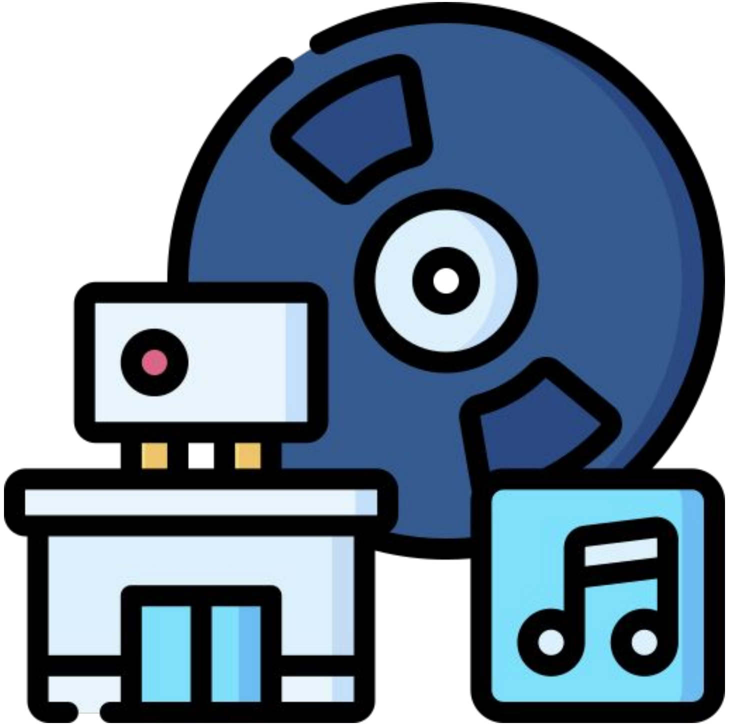
- Musicians from different genres
- Signers specialized in different styles
- Songwriters
- Bands

Stakeholders

3.- Record label: The producer of the artists songs. Take a commission from production, specializes in technical aspects.

Who?

- Major record labels
- Independent record labels specialized in specific genres
- Regional labels promoting local artists

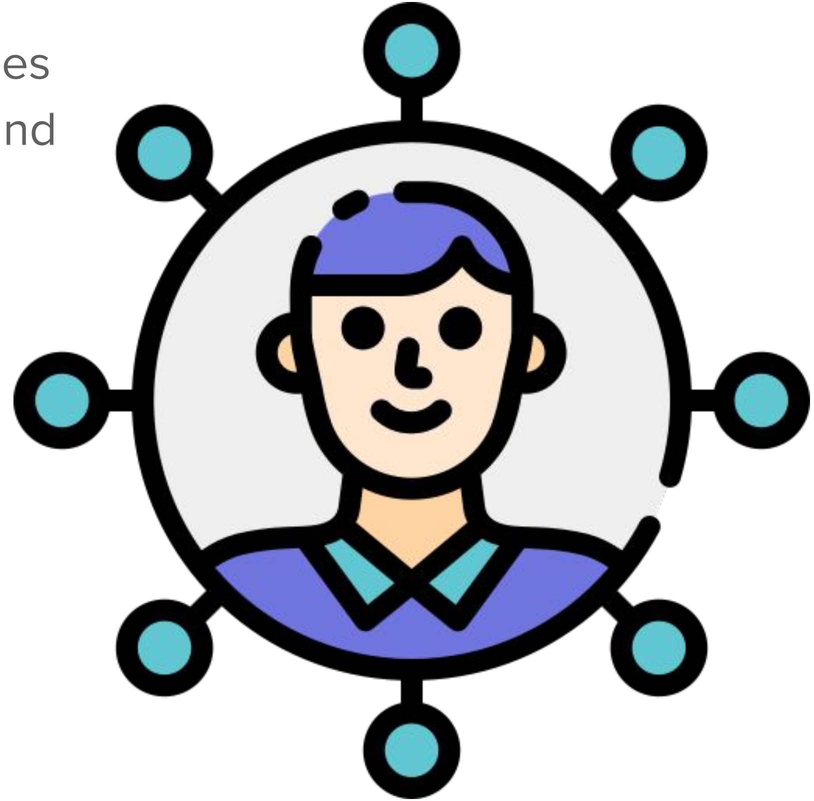


Stakeholders

4.- SingAlong Community Manager: Manages the social media. Gather user feedback and musical trends, valuable for the product.

Who?

- Social media Managers
- Online Community Moderators
- Engagement specialists managing

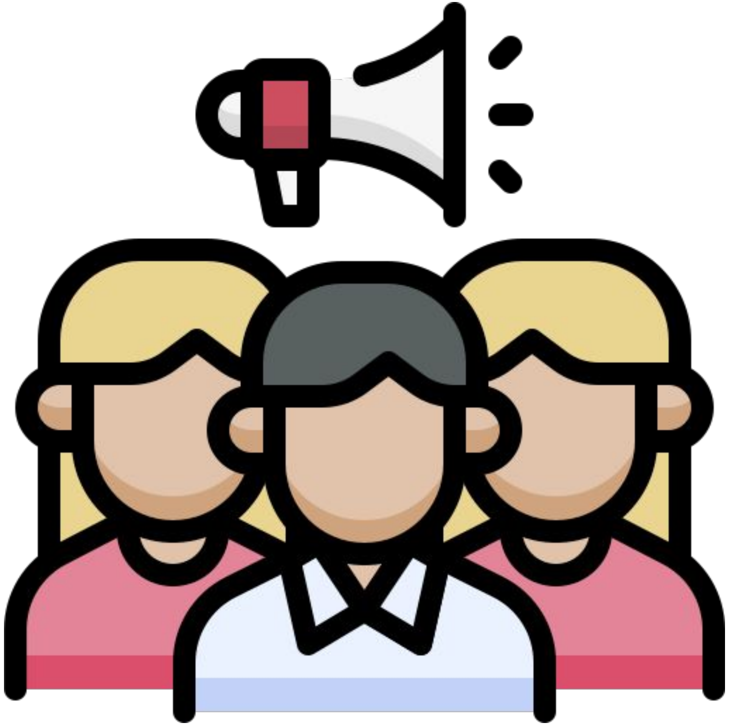


Stakeholders

5- Promotional Team: Creates marketing campaigns to reach the target users and uploads the promotional material from the artists.

Who?

- Marketing specialists
- Advertising professionals
- Campaign managers
- Technical team for uploading promotional material from artists



Stakeholders

6- Lawyer Team: Responsible for the Intellectual Property conflicts and other legislative issues.

Who?

- Intellectual property lawyers
- Copyright experts
- Legal advisors ensuring compliance with local industry regulations

Stakeholders

7- Maintenance and Support Team: Responsible for solving problems both about the app environment and the customers issues.

Who?

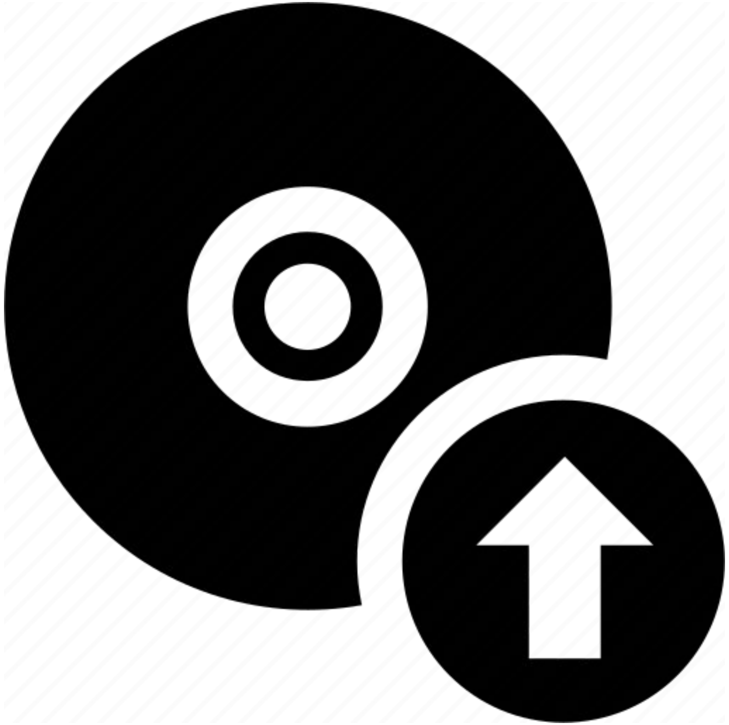
- Customer Support representatives
- Technical Support Staff for Businesses
- Helpdesk personnel resolving music playback or account management issues

Stakeholders

8- Human Operators: Responsible for sending and burning the CDs for the physical copies and uploading the heavy song files sent in CDs.

Who?

- Shipping CDs staff
- CD duplication technicians
- Large files uploading and processing operators

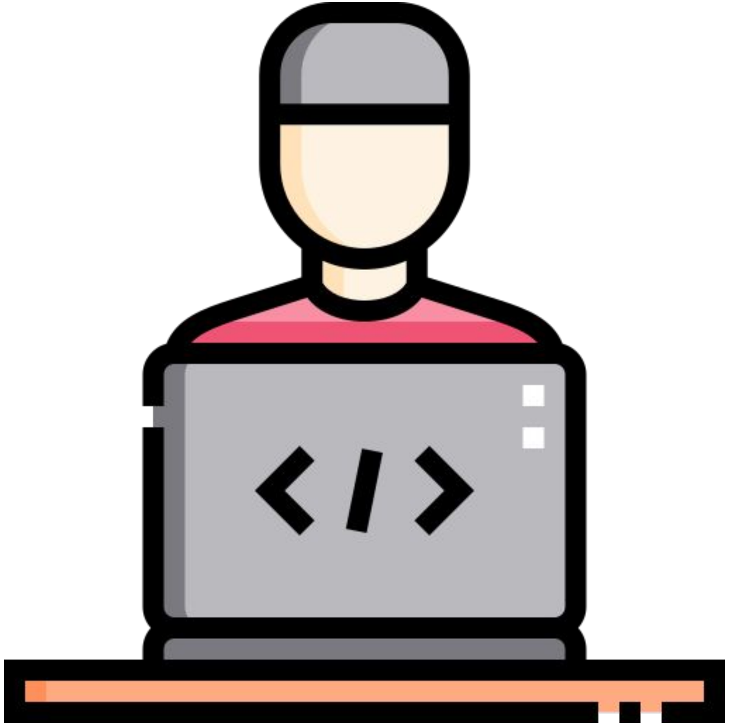


Stakeholders

9- Specialists for external software: Responsible for integrating external features to the system, such as the PayPal paying method.

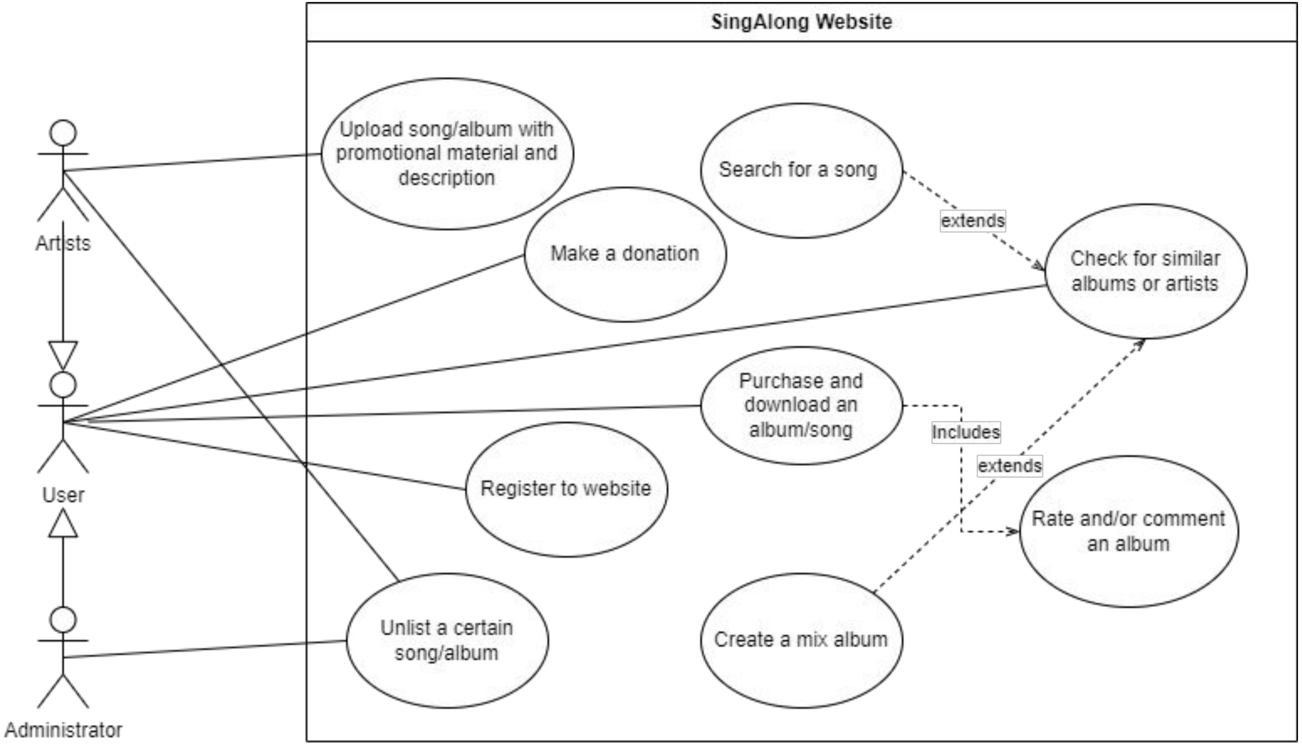
Who?

- Software engineers integrating the platform with external systems
- Technical consultants for implementing new features and technologies
- PayPal software engineers integrating payment method

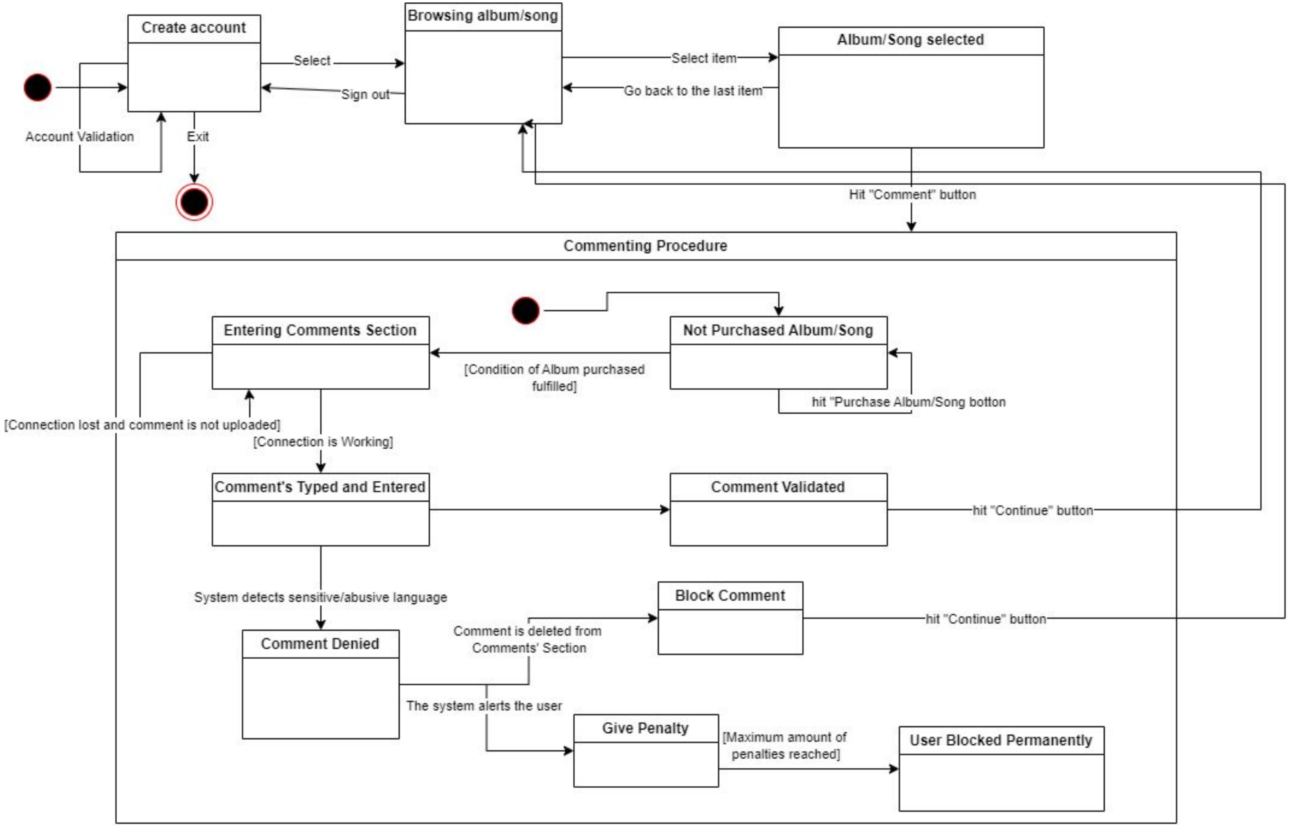


Diagrams about the Application

Use Case Diagram of the Full System


SingAlong Website
Artists
User
Administrator
Upload song/album with promotional material and description
Make a donation
Search for a song
extends
Check for similar albums or artists
Purchase and download an album/song
Includes
extends
Rate and/or comment an album
Register to website
Unlist a certain song/album
Create a mix album

State Diagram about “Commenting an Album”



Obscure Points and Functional Requirements

Obscure Points

“However, we want registration to be simple: the system only asks for a username and password. Optionally, a user may register their real name and email address. Registering emails enables the system to notify users of novel content that may interest them.“ … “Each user can add a single vote or comment, but only if the user bought or downloaded the given song or album.”

Issue: Since the personal information is not compulsory it lets malicious users create several accounts to bias some song, album or artist.

Functional Requirements

1.- User Verification: To prevent fraudulent behaviors, such as the creation of multi-accounts in order to manipulate the voting system, the application should need extra information during user registration, such as phone numbers or email addresses. This would also enable the system to notify users when new content is available, that can be an incentive to users to share personal data at the same time as we ensure the privacy of the user.

Obscure Points

“The system should incentivize the culture of making donations.”

Issue: It is not specified how the company wants to incentivize the users to make donations, it could be interpreted in many different ways, from a heavy spam campaign to the users to just making the incentivization option look better in the interface.

Functional Requirements

2.- Donation Promotion: It should be possible to present the donations in an accessible way. More information on the impact of donations on artists should be supplied to motivate donations. This feature wants to promote a supportive atmosphere for musicians within the community of the platform.

Obscure Points

“It is expected, by nature of the open and unfiltered system, that many new artists will register and as a consequence, that many songs will be of poor quality. By itself, this is not a problem, but it is very important that the user can “separate the wheat from the chaff” (i.e. quickly find successful artists with high-quality songs). Otherwise, if a user hears several songs that they do not like they may not be motivated to return to the Web service in the future. “

Issue: It is not very clear how we should classify the quality of each song, even if songs are usually rated by the overall performance, the new ones and the less known songs may be biased and it is possible that high quality songs could be hidden by a biased classification.

Functional Requirements

3.- Quality control and track detection: Since the system is open, the app should include a way to distinguish high quality from low quality music. This could include a reward for unique performance for new artists or voting elements for experienced musicians. This tool should facilitate the discovery of high-quality music and successful artists, increasing user engagement and retention.

Obscure Points

“The main page of each user will have three top-10 lists (although in the future, 10 could be changed for another arbitrary number): the 10 artists with highest performance overall, the 10 artists with highest performance over the last 6 months, and the 10 artists with highest performance over the last 15 days.”

Issue: We want to know if the artists of the tops can be included in more than one list, i.e., in case the top 1 artist overall is also the top 1 artist over 6 months it should be in both or we prioritize other artists to provide more diversity.

Functional Requirements

4.- Dynamic Artist Performance Lists: On each user's home page, the system should present three separate lists of the top 10 performers. These lists should show the top 10 performers overall, over the previous six months, and over the last 15 days. A musician may appear on multiple lists based on their performance over different time periods. The performance is computed as is explained in a previous point. These lists should be customizable, with the ability to change the number of artists included by some kind of administrator.

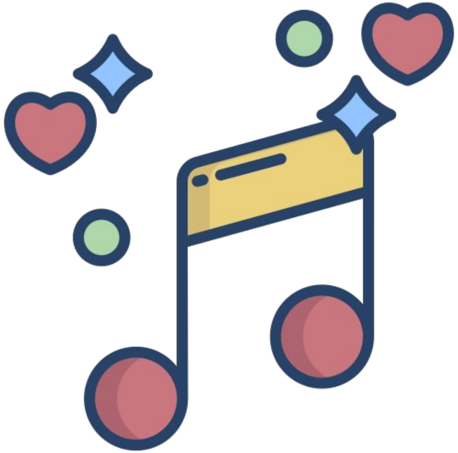
Obscure Points

“An artist can, in theory, add the same song to different albums. However, this should usually be avoided in order not to confuse users that follow this artist.”

Issue: Since the own SingAlong company wants this option to be avoided, it seems counterproductive to implement a functionality that it is expected to be ignored that could imply some cost both about the budget and the developers time.

Functional Requirements

5.- Song Allocation to Albums: The system should prevent artists from including the same song across multiple albums. This rule should be set during the song upload stage to ensure that each song belongs to only one album. This is in order to help the customers to avoid getting confused during the process of purchasing.



Obscure Points

“A customer can also purchase personalized albums, selecting songs by an artist of choice. If the artist permits, it is also possible to mix songs of different artists. In this case the total sale price is computed on the basis of the price of each individual song.”

Issue: As the company is interested in attracting the maximum number of regular users, it should be possible for the users to create their own albums with the songs they like without needing the authorization of the artist to mix songs.

Functional Requirements

6.- Customized Album Creation: Users should be allowed to make customized albums by choosing music from various artists. The entire cost of such an album should be estimated by adding the individual expenses of the songs chosen. This feature should improve user satisfaction and engagement.

Obscure Points

“The songs on physical devices will be entered into the system by a human operator, who needs access to the system via a graphical interface (not a Web browser).” “Optionally, a user can choose to purchase an album on a CD which is burned and subsequently mailed to the user’s address. In this case, the sales price includes the cost of buying, burning and mailing the CD.”

Issue: We want to know if the human operator responsible for uploading the songs on physical devices into the system would be the same as the person responsible for the mailing of the sales CD to the users. If this is not the case, we would like to know who is going to be responsible for this last function.

Functional Requirements

7.- Physical Device Management: The system should provide a separate interface for a human operator to input music from physical devices into the system, rather than through a web browser. This operator should also be responsible for burning and sending CDs to customers who purchase them. The number of operator accounts created should be proportional to the workload and these accounts should be easily created and deleted as required.

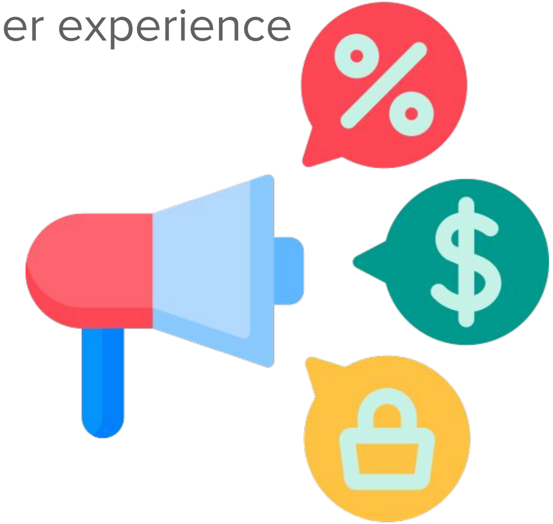
Obscure Points

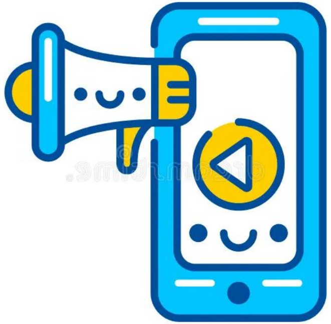
“The system is in charge of managing promotional material and distributing it to users when they purchase songs and/or albums.”

Issue: It is not specified how the web page should manage this case, as several options are implied. For example, when the artist uploads to the web the images, this have to be automatically attached to the artist songs or a promotional/marketing team will manage this.

Functional Requirements

8.- Manage promotional content: When promotional material is uploaded, the system should immediately link it to the artist's songs. There should also be a review system in place to guarantee that the promotional content fits the platform's criteria. By making it easier to get promotional material, this capability should improve the overall user experience



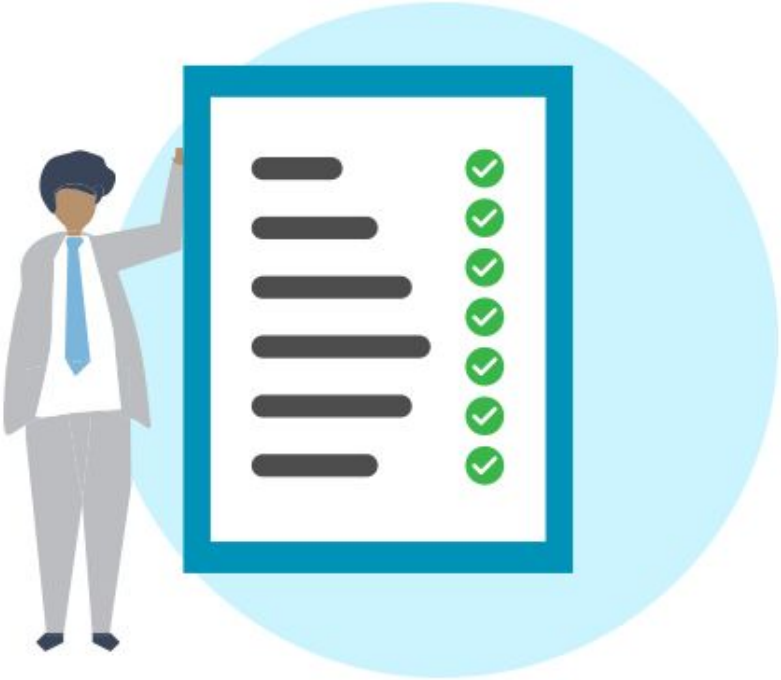


Additional Functional Requirements

The following requirements are less relevant requirements or too clear to explain more details about them.

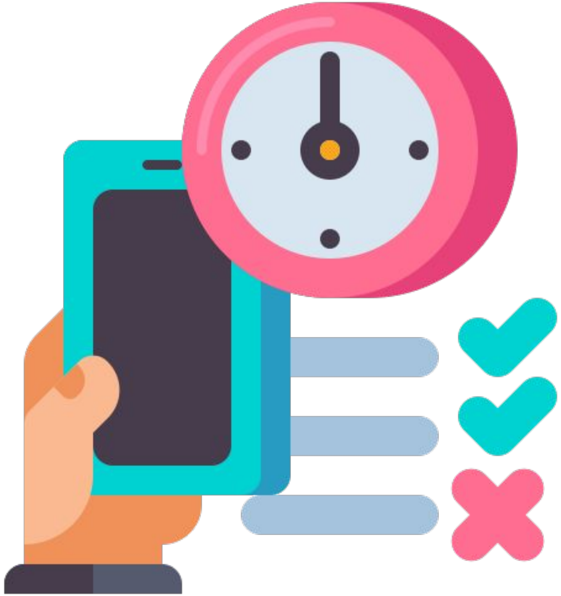
These requirements are not a direct result of a obscure point.

Are relevant but not essential.



Non-functional Requirements

Non-functional Requirements: Product



1.- Usability: Easy and interpetable app interface for any user. To achieve a fully personalized experience, the main search toolbar allows to search by songs, albums or artist, as well as different music styles or age. Moreover, a settings menu must be included where different functionalities will be displayed. It’s important to have a good relation with users, so a FAQ section and real-time support section should also be included.

Non-functional Requirements: Product

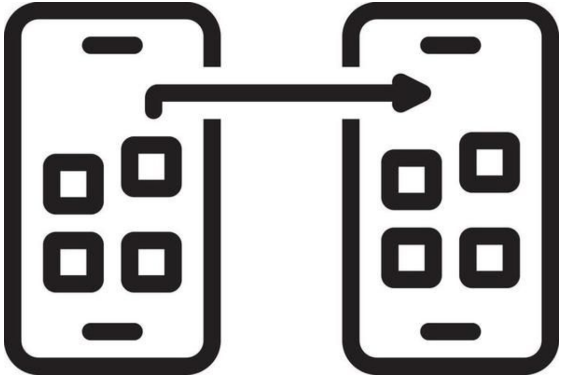
2.- Efficiency: To achieve the best experience, the purchase and download of the music must be fast, preventing the user from having to wait. This reaction time should be implemented also in any other action inside the website, which will also include a refresh button. To achieve efficiency, the app’s size should be strictly limited to 100MB and consume at most 500MB of memory. Moreover, the user experience should not be affected by late deliveries for burned CDs, which shouldn’t last longer than 2 weeks.

Non-functional Requirements: Product

3.- Robustness: To avoid full system failure, the company should have backup servers to act in case of shutdown of the main ones. In unfortunate case of shutdown, the recovery time shouldn’t be longer than 10 seconds. In addition, to achieve maximal confidentiality of our data, several security measures should be implemented. About the app’s interface, it should have a decent error tolerance in order to handle common user errors, like invalid inputs.

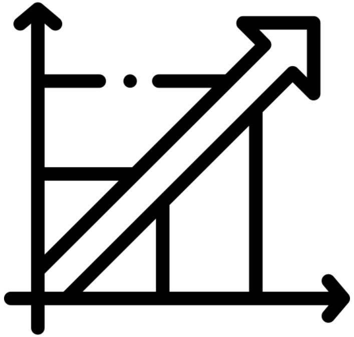
Non-functional Requirements: Product

4.- Portability: Extend the app’s accessibility from all types of electronic devices to achieve the best possible experience. That is, designing the system core functionality to be portable in different operating systems of the different platforms and major web browsers. In addition, the app should be seamlessly integrated with external systems, like payment gateways and artist platforms to support future expansions.

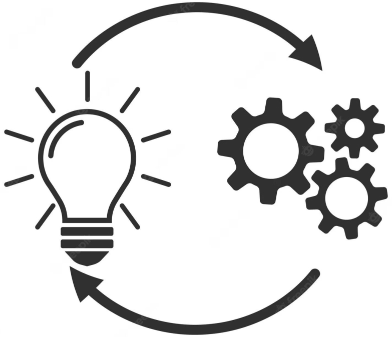


Non-functional Requirements: Product

5.- Scalability: The application should have a well design for an easy scalability to accommodate a growing user base and handle increased data volume without performance degradation. Moreover, since the community is oriented to be international, the app should provide support for multiple languages.



Non-functional Requirements: Organizational



1.- Deliverables and implementation: Deadlines should be implemented for delivering updates of the web to ensure timely enhancements of the system, as well as generating periodic reports to demonstrate progress towards project milestones. Moreover, it’s recommended for the low-level programming to be based on c++ language.

Non-functional Requirements: Organizational

2.- Standards: To give freedom to the artist and user, all songs will have different types of qualities, in a range from low to high. However, every song in this range, despite being in the lowest value, must have a minimum of quality that assures a good user experience. In addition, no matter the quality of the song, it must have several options for the users to download it in the format they prefer, inside the valid possibilities.

Non-functional Requirements: External

1. **Interoperability (regulatory):**

   Gives to an outside product like an external interface the ability to interact with them. The company has the ability to work with different social platforms and they can share the music that they are listening in our platform. Furthermore, gives the option to use the Paypal method to be compatible with the donation and payments of the user.

Non-functional Requirements: External

2. Ethics:

The platform should apply ethical restrictions to avoid disrespect. Consequently the system has to take care in the security and also with the banned personal information. Furthermore, the full percentage of the donations is for the artist.

Non-functional Requirements: External

3. Social:

The system will have notice of the social factors due to, filtering options in playlist of our platform and recommendations. changes in the system are not accepted. In addition, changes are not accepted, the system discards and delete songs with offensive content to any social collective penalizing the artists that upload songs with discriminative lyrics.

Non-functional Requirements: External

4. Legislative:

The copyright policy will be important, due to each country has different legislation and it is also important to have a license for music distribution. In addition, the privacy policy it is fundamental and the system must ensure the privacy and safety of the users.

REFERENCES

[1] Drawing on the successful model of Spotify, which requires around 130MB of space excluding downloaded music, we anticipate our application's size to be relatively similar. However, we aim to streamline our app further, striving to be closer to the average most-used app size of roughly 60MB, as outlined by statistics from **[Statista]**(https://www.statista.com/statistics/1296527/size-top-android-apps/).

[2] Turning once again to Spotify as a benchmark, discussions on **[Spotify's Community]**(https://community.spotify.com) reveal that the app consumes around 600MB on average with hardware acceleration enabled, and about 150MB without it. Notably, Spotify's RAM usage averages around 8% of a device's total memory - a proportion we intend to bear in mind for our software.

[3] With the average size of a song estimated at 5MB, our system will cater to the needs of artists by accommodating songs that are up to twice the average length. This decision is rooted in providing ample flexibility to the artists while ensuring efficient use of our system's resources.

[4] We've also considered the minimum bit amount required for human auditory perception. This consideration will influence the quality of the audio output of our application, ensuring that it's not just technically sound, but also delivers a great user experience.

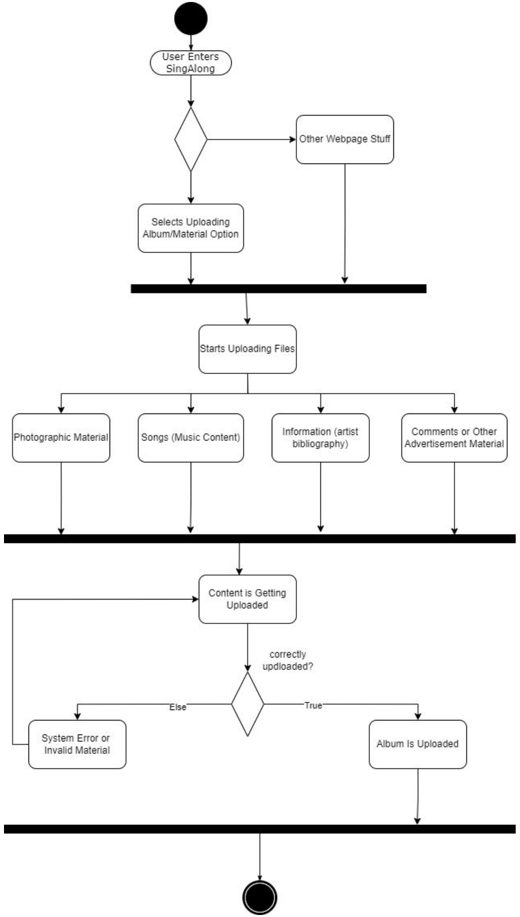
[5] In addition, we've considered the lowest optimal frequency required to cover the entire range of human hearing. This includes accounting for anti-aliasing, twice the range of human hearing (20Hz-20KHz). This decision is backed by **[Adobe's guidelines]**(https://www.adobe.com/uk/creativecloud/video/discover/audio-sampling.html#:~:text=The%20human%20ear%20can%20hear%20between%2020%20hertz%20(20Hz)%20and,rates%20to%20capture%20all%20sounds) on audio sampling, ensuring that our application will deliver high-quality audio, regardless of the range of sound

Software Requirements Specifications: SingAlong

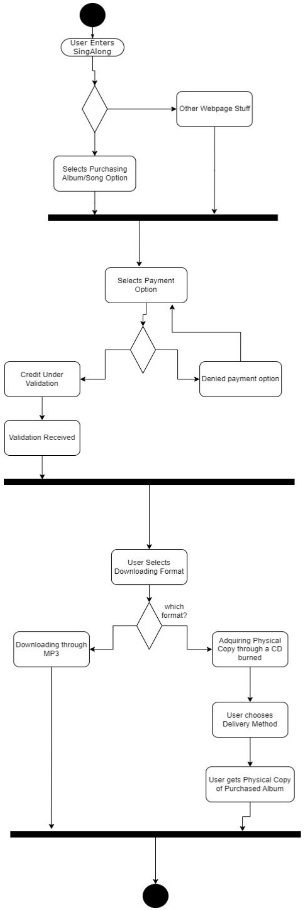
Alex Oller, 243398
David Pérez, 241375
Eneko Treviño, 241679
Guillem Escriba, 242123
Max Frigols, 241156

Backup Slides

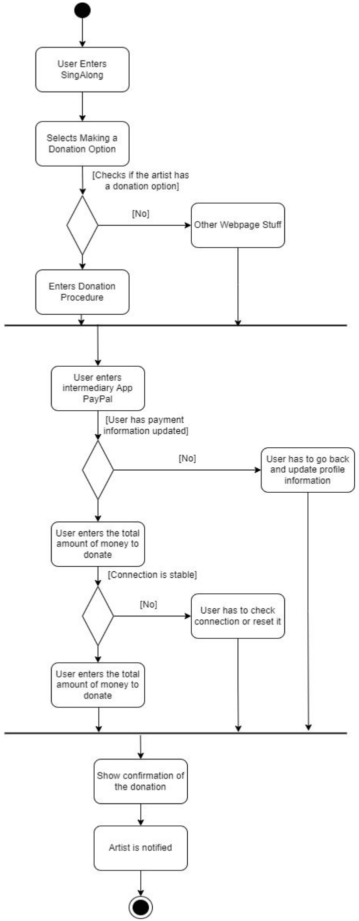
Activity Diagram of “Publishing an Album”


User Enters SingAlong
Other Webpage Stuff
Selects Uploading Album/Material Option
Starts Uploading Files
Photographic Material
Songs (Music Content)
Information (artist bibliography)
Comments or Other Advertisement Material
Content is Getting Uploaded
correctly updloaded?
Else
True
System Error or Invalid Material
Album Is Uploaded

Activity Diagram of “Purchase an Album”


User Enters SingAlong
Other Webpage Stuff
Selects Purchasing Album/Song Option
Selects Payment Option
Credit Under Validation
Denied payment option
Validation Received
User Selects Downloading Format
which format?
Downloading through MP3
Adquiring Physical Copy through a CD burned
User chooses Delivery Method
User gets Physical Copy of Purchased Album

Activity Diagram of “Make a donation”



User Enters SingAlong

Selects Making a Donation Option

[Checks if the artist has a donation option]

[No]

Other Webpage Stuff

Enters Donation Procedure

User enters intermediary App PayPal

[User has payment information updated]

[No]

User has to go back and update profile information

User enters the total amount of money to donate

[Connection is stable]

[No]

User has to check connection or reset it

User enters the total amount of money to donate

Show confirmation of the donation

Artist is notified