



Bilkent University

Department of Computer Engineering

Senior Design Project

Musync

Project Specifications Report

Ahmet Çandırođlu, Anıl Erken, Berk Mandıracıođlu, Halil İbrahim Azak

Supervisor: Asst. Prof. Dr. M. Mustafa Özdal

Jury Members: Assoc. Prof. Dr. Özcan Öztürk, Prof. Dr. Cevdet Aykanat

Project Specifications Report Oct
15, 2018

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the requirements of the Senior Design Project course CS491/2.

Contents

1. Introduction	3
1.1 Description	4
1.2 Constraints	5
1.2.1 Economic Constraints	5
1.2.2 Ethical Constraints	5
1.2.3 Implementation Constraints	5
1.2.4 Sustainability Constraints	6
1.3 Professional and Ethical Issues	6
2. Requirements	6
2.1 Functional Requirements	6
2.1.1 Place Owner Specific Requirements	6
2.1.2 Patron Specific Requirements	7
2.1.3 Data Resources	7
2.2 Non-Functional Requirements	8
2.2.1 Usability	8
2.2.2 Supportability	8
2.2.3 Reliability	8
2.2.4 Efficiency	8
2.2.5 Security	8
2.2.6 Scalability	9
3. References	10

1. Introduction

Music is a common interest that many people enjoy and rest their souls. It may become a cumbersome procedure to find music that many people enjoy. In our daily lives, we are surrounded with music by means of our environment such as restaurants, cafes, bars, etc. Therefore, it is even more important to create a suitable playlist to satisfy majority based on their collective music taste. However, majority can not be satisfied every time, hence it is also necessary to recommend new places that users might enjoy the general music taste and even discover new songs.

Musync aims to facilitate creation of playlists that are dynamically modified both by analysing the music tastes of users and their feedback on the current playlist. Musync is basically a digital jukebox which collects data about their music tastes from its current users and initialises a playlist. Moreover, users are able to add songs to the playlist and manipulate the currently played songs by the power of bidding. Users have different values in our system, thus not every user has the same bidding power. Users that are regular customers of the place have more power in the creation of the playlist in that place.

In this report, first we will provide a description of the project. After that, constraints of the project will be examined under the appropriate sections. Then, we will discuss professional and ethical issues that may arise. After those, requirements of the project will be given under appropriate sections grouped under functional and non-functional requirements.

1.1 Description

The project aims to open up the control of the ambient music in public settings such as cafe and restaurants to the patrons via a mobile website. The proposed solution allows the patrons to influence the order of the playlist in the location as well as the content of the playlist by bidding system. The solution allows the owners to worry less about finding type of music their patrons will enjoy and has the power to put together an appropriate playlist automatically based on the listening habits of the visitors.

Musync shines out with its extremely simple user experience. Patrons are not required to create accounts to use the service, the patrons enter a simple 4-digit PIN system backed by geo locations to identify cafes/restaurants, patrons do not need to be on the cafe's wifi network, patronage is used as a weight factor in influencing the playlists, and the owners have a simple console to manage their ambient music experience backed by Spotify.

Moreover, our service allows patrons to discover new places based on their music tastes. For example, a customer that enjoys rock music will be recommended to visit a rock bar. The places that play similar genre that patrons enjoy, can be viewed on the map. Patrons are able to select a place on this map to view the currently played song and the general genres of the place.

1.2 Constraints

1.2.1 Economic Constraints

- The application will have a monthly/yearly subscription price for place owners.
- Place owners and users who want to integrate their playlists with the joint playlist will need to have their own Spotify account. They may have extra costs if they choose to have a premium account.

1.2.2 Ethical Constraints

- Code of Ethics by the NSPE will be followed in the development of the project [1].
- Users will be able to share their playlists with the application if they want to. In that case they will be informed that Spotify data will be used to create a joint playlist. However the application will not store and use this data for another purpose without informing the user and asking for permission.
- Location of users will be used to determine the place they are in. This data will not be stored or used without informing user and asking for permission. Stored location data will be encrypted and will not be distributed.

1.2.3 Implementation Constraints

- Application will be a web application developed with both desktop and mobile browsers in mind.
- To track the development, Github will be used as a source control tool.
- The services provided by Spotify Web API will be used.

- Application will be developed by OOP paradigm, prioritizing user experience.

1.2.4 Sustainability Constraints

- The application will provide service using Spotify Web API. This will allow it to provide good number of up to date tracks.

1.3 Professional and Ethical Issues

We will not gather any data without user's permission. If user permits, information of their Spotify activity and places they visit will be stored in order to contribute to the collective playlists, to gain loyalty points from the cafe and to recommend new places according to user's music taste. However these data will not be distributed, will not be used for other purposes and will be encrypted and anonymous.

Any open source code used will be indicated and licence issues will be handled properly.

2. Requirements

2.1 Functional Requirements

2.1.1 Place Owner Specific Requirements

- Place owners have to create an account with an email and password.
- Anyone with a Spotify account will be able to subscribe to our system. There will be two types of accounts: free trial and paid subscription.

- Free trial allows up to 5 days of usage in a month. This account type is suitable for individuals who want to use the application occasionally for their home parties etc.
- Paid subscription allows unlimited usage of our system. These users are mostly owners of a place (cafe, bar, restaurant, etc) who are willing to give their customers best music experience.
- Owners can limit song requests to specific genres.

2.1.2 Patron Specific Requirements

- Patrons don't have to create an account to use the application. A unique account will be created automatically on first use.
- Patrons can add new songs to the common playlist if it is suitable to genre set by the owner.
- Patrons will be able to bid for songs to play them next.
- Patrons don't have to link their Spotify account to use add songs or bid.
- Patrons can link their Spotify account to contribute to the common playlist.
- Patrons can earn tokens which can be used to add/bid for songs by being a regular customer of a place.
- Patrons can receive place recommendations according to their music tastes and location, and view them in a map.
- Patrons can view all places with their preferred genre and currently played song in a map.
- Patrons should agree to share their GPS locations to our system in order to use it.

2.1.3 Data Resources

- The application will use Spotify Web API to gather user's Spotify data.
- Google Maps API will be used to offer and display new places to user.

2.2 Non-Functional Requirements

2.2.1 Usability

- The user interface should be simple and user-friendly.
- The users should have no difficulty to start using the system.

2.2.2 Supportability

- The system should be able to work in any web browser.
- The system should be compatible with third party APIs such as Google Maps and Spotify.

2.2.3 Reliability

- System should be able to create joint playlists that will satisfy the majority, independent of the number of concurrent users.

2.2.4 Efficiency

- The delay between a user requesting a song and it being added to the playlist should not exceed 10 seconds.
- Backend response time of the website (the time starting when an HTTP request taken and ending when the server starts to send frontend data) should not exceed 200 milliseconds.

2.2.5 Security

- Information gathered from users and authentication related tokens should be stored securely.
- Money transactions should be performed without any information leak to third parties.

2.2.6 Scalability

- The service provided to each place and its customers should not be affected by the amount of places and active users.
- System should be able to handle multiple song requests coming from different users simultaneously.

3. References

- [1] Code of Ethics National Society of Professional Engineers, nspe.org, 2016. <https://www.nspe.org/resources/ethics/code-ethics/>. Accessed: 2018-10-14.