Requestify Project Proposal

Team

Sean Cheema - Project Manager/Front End Developer

Kyle McClelland - Back End Developer

Joseph Parimucha - Back End Developer

Madhav Sagi - Front End Developer

Advisor

Dr. Alexandre Gomes de Sequeira

agomesdesiqueira@ufl.edu

September 8, 2024

Abstract

Requestify is a web application for song requests where users can send requests and add songs to a DJs playlist. The web application will be developed using react due to its dynamic content loading and practicality for a single page product. The Spotify Web API will also be utilized to improve the user experience by providing them access to Spotify's music library to search for and play songs. Example scenario: The user is at a club and requests a song via the Requestify web application. The DJ can either approve or deny the request; if the request is approved, the song will be added to the DJs playlist. Users can also send anonymous tips to DJs via Stripe's well-documented API for payment processing to support them, serving as an additional revenue stream for the DJ or venue.

Table of Contents

Introduction

Problem Formation

DJs are responsible for maintaining an engaging atmosphere at live music events. In order to create a lively atmosphere that everyone can enjoy, DJs can often be seen trying to take song requests from the attendees. However, the process of requesting songs can be disorganized, as it typically relies on verbal communication or handwritten/typed notes on devices. This can lead to misunderstandings, lack of interaction from DJs, and missed song requests.

Rationale of Project

Demand for personalized music is growing and both DJs and audience members are looking for more organized ways to interact. Creating a digital platform for personalized music requests can address these inefficiencies and provide a quicker and more organized experience. Additionally, a digital platform would aid in the success of DJs as well as the venues they perform at.

Proposed Solution

Requestify is a web application designed to simplify the song request process. Users at the venue can send song requests directly to the DJ. The DJ has the option to approve or deny the request. If a song is approved, it will be added to their playlist. Using Stripe API (for anonymous tipping), adds a new revenue stream for the DJ. One additional proposed feature for this web-application is a polling component, where users can vote on songs to be played. This would increase the interactivity the audience members have with the DJ, the other attendees, and the event itself. In turn, this would provide a more positive experience to everybody at the venue, allowing them to have a say in what music they are listening to.

Importance/Impact of Project

Requestify helps both DJs and users as it allows for event attendees to request songs in a quicker and more organized fashion. For the DJ, this can be seen as an opportunity to increase their earnings through tips and to increase their interaction with the target audience. This innovation has the potential to revolutionize how live music environments can be managed and monetized.

Literature Survey

Problem Context and Relevance

As mentioned previously, DJs are the main person responsible for keeping up an engaging atmosphere at a live entertainment event. These venues are a popular way to have fun for many students and other young adults living in college towns and cities. The University of Florida is no different, as there are numerous bars and clubs lining University Avenue, which are packed with the university's nearly 60,000 students at the end of the week (U.S. News & World Report, n.d.).

Traditionally, attendees would need to navigate through a loud and crowded venue and up to a DJ to request for a specific song to be played, whether verbally or written on paper or their phone. This is a cumbersome process and is one of the most difficult parts of the job of being a DJ (Burrett, 2023). At a club or bar, playing songs that people want to hear keeps them engaged and enjoying their experience, underscoring why this is a valuable issue to address (Burrett, 2023). Guests at live entertainment venues at U.S. based college towns and other major cities would benefit from a system like the one being proposed.

The increase of digital interconnectedness of the modern world offers a clear opportunity for web applications to enhance and simplify this process to improve the attendees experience after a long week of studying and/or working.

Existing Work and Similar Applications

There are various smartphone apps or web applications that offer similar functionality to what is being proposed.

One application named RequestNow, has the user scan a QR code which allows them to text song requests, send tips, and engage with the DJ (RequestNow, n.d.). The DJ is also provided with a software interface to assist with this process. It includes a variable fee depending on the use case.

Another similar application, limeDJ, also has the users scan a QR code and they are directed to the DJs show page through a web browser (Lime DJ, n.d.). Here, they have access to the DJs social media links, song request system, and tip system. All of this information is included in a web application, so there is no need for the user to download any software to their device.

The final application being discussed is called QueueDJ, which provides an interface for the DJ as well as an IOS or Android app for attendees (Queue, n.d.). Using the phone app, users can interact with the DJs, send song requests, video requests, shoutouts, and tips. While it seems to have more functionality than the other applications, it does require the users to download a software application.

In general, the existing solutions exemplified above work as intended, but each have their own design and/or operational flaws. Requiring the user to download an application while at an

event may be difficult to implement in practice. Attendees are in a loud and disorienting environment, which is typically not conducive to downloading a piece of software and setting up an account. While this is not an issue for every application mentioned, the others have issues varying from unnecessarily convoluted user interfaces to complicated tipping processes. This leaves a gap to be filled in this market area by a new application and brings up further research questions to be investigated and implemented in Requestify.

Frameworks, API's, and Technology

Spotify API

The Spotify API will allow Requestify to directly interact with the Spotify web application. This would enable users of the service to utilize Spotify's library of music to play songs or to search for song information. This functionality will improve their experience using the web application as they would not need to switch back and forth between other phone applications and the Requestify interface. This also helps to ensure the exact song the user wants is requested, since the link to the song is provided, as opposed to just the name. The Spotify API is a well established API with plenty of resources available for troubleshooting and further development (Spotify for Developers, n.d.).

Stripe

Stripe is a payment processing platform that allows users to accept payments. It can be integrated with React. The React Stripe.js library specifically has a plethora of helpful documentation and resources available for its implementation in a web application like Requestify (Stripe Documentation, n.d.).

React

React is a JavaScript library that can be utilized for building user interfaces with reusable components and it is particularly applicable in the situation of a single page web application, like Requestify (Alvarez Duran, 2023). It also has the ability to use dynamic content loading meaning the Requestify web application can update certain parts of the web page with new data without needing to reload the entire page (Mozilla, n.d.), increasing performance, and in turn, improving the user experience.

Literature Review Summary

Overall, the literature displays that there is a need for an application like the one being proposed. There are millions of students and young adults alike around the world who would benefit from a system like Requestify, allowing them to have a more enjoyable experience after a long week of hard work. While other similar applications exist, Requestify is an innovative solution to this issue providing users with an easy to use interface to interact with the DJ, request songs, and send tips without needing to download an application to their device. This solution addresses the issues of needing to download an application as well as overly complex interfaces of other products.

Proposed Work

The work for this project will be roughly divided into two sections, the front end and the back end, but will be fluid enough to allow members to work in other areas as needed. The project will be built as a web application utilizing the ReactJS framework, using javascript, html,

and css for the front end, and a combination of Golang for its efficiency, and Python for its compatibility with the APIs for the backend.

We are planning on using two different APIs for this project, the first being Spotify's API. Spotify's API allows developers to gain access to Spotify's database of music, and allows for playback through Spotify itself. Having that direct connection to Spotify will allow for an overall smoother and more engaging experience. The other API, Stripe, is a payment processing service that would be the backbone of the tipping feature.

As of writing this proposal, there are two main features we plan to have implemented by launch. First is the ability for a DJ or event host to accept song requests from event/venue attendees, and then play such requests directly from the web app using the previously mentioned Spotify API. Additionally, users can vote on requested songs communicating to the DJ what they want to be played. Second is the tipping functionality, which will allow attendees to directly send money to the DJ, allowing for an additional revenue stream for the DJ/venue. Due to the inclusion of tipping, we will include a user account system to help maintain security related to payment information. However, to keep tipping from becoming too much of an influence on which songs are played, we will keep the tipping and song playing anonymous and separate.

The target audience will mainly consist of college town bars, clubs, weddings, or any venue with a need for a DJ, but Requestify is something that could be used anywhere, regardless of the demographic or event type.

Future / Reach Goals

Given the constraints of this development process and course, there are reach goals that will be researched and developed if time permits. The current reach goal that has been

established is a social reward / incentive system for users. This goal would include the research and implementation (if it is found to be possible) of a reward system to encourage socially conscious and healthy behaviors. For instance, utilizing the Apple HealthKit or Google Fit API to reward users if they have walked a certain number of steps that day. This is just one example of a reward system, others may be discovered throughout the duration of this course.

Project Plan

TASK	PROGRESS	START	END
Software Specification			
Research	100%	8/26/2024	8/30/2024
Create Project Plan	100%	8/26/2024	8/30/2024
Complete Project Proposal	100%	8/26/2024	9/8/2024
Software Design and Implementation			
Create Project Overview	0%	9/9/2024	9/20/2024
Make Software Architecture Diagram	0%	9/9/2024	9/27/2024
Design User Interfaces	0%	9/9/2024	9/27/2024
Begin Software Development	0%	9/9/2024	10/6/2024
Create Presentation	0%	9/23/2024	10/6/2024
Present Design and Implementation	0%	9/30/2024	10/6/2024
Software Implementation and Validation			
Update Project Plan	0%	10/7/2024	10/18/2024
Create Project Walkthrough	0%	10/7/2024	10/25/2024
Make Test Plan	0%	10/7/2024	10/25/2024
Analyze Issues and Strategies	0%	10/7/2024	10/25/2024
Continue Developing Software	0%	10/7/2024	11/3/2024
Test Software	0%	10/7/2024	11/3/2024
Create Presentation	0%	10/21/2024	11/3/2024
Present Implementation and Validation	0%	10/28/2024	11/3/2024
Software Delivery			
Complete Software Validation	0%	11/4/2024	11/22/2024
Complete Project Build	0%	11/4/2024	11/22/2024
Create Project Poster	0%	11/18/2024	11/29/2024
Present Project	0%	11/25/2024	12/3/2024
Project Submission and Documentation			
Create Final Project Documentation	0%	11/25/2024	12/3/2024

Link to full plan: Requestify Project Plan.xlsx

Conclusion

Requestify leverages modern web technologies and integrates Stripe and Spotify's APIs to create a user-friendly solution to a common problem that DJs and their audiences face. This platform not only enhances the efficiency and organization, but also improves the ability for users to make song requests, and introduces a new stream of revenue for DJs. Requestify solves the logistical challenges of song requests and contributes to the evolution of the live entertainment industry by creating a more connected environment. This project exemplifies how modern technology can be harnessed to provide novel opportunities for both artists and their audiences.

References

Alvarez Duran, M. E. (2023, April 20). *Choosing react.js for web development: Top use cases and real-world examples*. React.js for Web Development: Top Use Cases & Examples. https://www.nan-labs.com/blog/reactjs-web-development/

Burrett, M. (2023, June 2). *The Ultimate Guide to Taking Song Requests*. DJ Tech Advice. https://djtechadvice.com/taking-song-requests/

Lime DJ - QR Code Song Request System for DJ's. Lime DJ. (n.d.). https://limedj.com/

Mozilla. (n.d.). *Fetching data from the server - learn web development: MDN*. MDN Web Docs. https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Fetching_data

Queue. (n.d.). *Queue - Song, Video and Shoutout Request for Everyone!* https://queuedj.com/

React stripe.js reference. Stripe Documentation. (n.d.). https://docs.stripe.com/stripe-js/react

RequestNow | The modern song request tool for DJs. RequestNow. (n.d.). https://www.requestnow.io/

University of Florida - profile, rankings and Data | US News Best Colleges. U.S. News & World Report. (n.d.). https://www.usnews.com/best-colleges/university-of-florida-1535

Web api. Web API | Spotify for Developers. (n.d.). https://developer.spotify.com/documentation/web-api