Tunes

Team Members: Kevin Alarcon, Munkh-Erdene Baatarsuren, Kyle Godinho, Damain Moquin, Evan Moore, Jack Nguyen

Github

Overview

Tunes is a social media platform that focuses on connecting users with similar music taste who live near each other. The app will allow you to connect your Spotify account and share information such as location, favorite songs, and playlists and offer robust features for browsing through users close to you and connecting with them such as event pages, user search, and an interactive map that displays your friends on the site, as well as others near you currently listening to music. Since our first iteration of the app, we've kept our design mostly the same, adding or removing some functionality as originally envisioned depending on our experience with integrating it into the Django framework.

Design Overview

Our data model includes the main classes of User and Event, which contain many fields which we use to populate user profile and event pages, as well as the people search page, map, and homepage parts of the application. Examples of such data include favorite song (or "anthem"), latitude and longitude for the map, time period for events, and profile picture/other personal information. For types of data that was reused often in our database (such as artists, albums, genres, songs, and locations) we created additional model fields with basic information such as art, name, or embed code when necessary.

We created views and URL mappings for each page which reflect the needs of each page, generally looping through a list of users and events when necessary to populate the page with each of their necessary data model information from the database. The URL mappings are created to link our pages together to facilitate easy navigation among the different sections of our site, no matter which page you are currently on. For applications like the homepage menus and the map, we made use of the absolute URL methods included in each class to easily link to unique user or event pages as we see fit.

Problems/Successes

Overall, our group was successful in expanding our application to include backend elements of a real webpage using Django. We continued using the work distribution of before where each was generally responsible for one aspect of the site and creating the necessary views/URL mappings for that page. We took a collaborative approach to planning the models and adding data to the admin site, which led to one of the few problems we encountered during the project involving the use of Github to host our changes. Since most of us were fairly new to Git, it has been a learning process getting used to the process of pushing and pulling changes from a single master branch, especially when adding data to the admin site separately, and we occasionally ran into the problem of some data being overwritten due to careless merging. Once we identified this issue, we began clearly communicating with each other via Slack when one of us would push our changes to the master branch, and have ceased running into such issues as our communication improves. Keeping this in mind for our work on the next section of the project, we anticipate that as long as we continue frequent communication with each other regarding our work on the project that we should not run into more issues, especially as we all become more familiar with the process of using Git in a collaborative, software engineering environment.

Individual Writeups

Kevin Alarcon

My work consisted of contributing to the user page detailing all the users information like what albums they've been listening to, their hand selected playlist, events they are attending, etc. I created the template "user_detail.html", which uses the User, Event, Album, and Playlist model. All the album artwork was properly placed and mapped using the admin site and downloading to images folder. As for the models, I helped identify attributes for the User model that were necessary for the User detail page, like rotation, playlists, and user bio

Munkh-Erdene Baatarsuren

My major focus was on how a user who is currently logged in sees his/her own profile page and settings of his/her account. Based on the model designed for "user", the two pages, my-profile and settings, use and present all data from the user who is currently logged in. Since user authentication is not included in this project, we assumed the user, Tim Richards, is currently logged in.
Presenting these two pages includes url mapping, retrieving data from the database and dynamically changing DOM of the my-profile.html and settings.html for each user.

Since, aside from individual tasks, we helped each other when there are some problems related to integrating the components of the project, everyone has some amount of contributions to other parts. I think we did a good work, evenly distributing the workload of the project to all members; therefore, my contribution was around 16% (⅙) of the total work.

Kyle Godinho

For this Project Submission I again worked on implementing the home page. To begin I setup the static directories for the css files and other static files we are using, the base template, and the proper url mappings to link to the homepage on server startup. Following that, I created the index.html template for the homepage and implemented the view for it to populate the user and event lists appropriately, leaving the map functionality to Evan. As the project went on I also helped with other issues as they arose.

I helped make sure the url mappings for the user and event pages worked using regular expressions for the IDs we are using and also helped with sorting out other media we were uploading to the database through the admin site. This was fixed by specifying MEDIA_ROOT/MEDIA_URL in the settings. Then I just updated the links on the navbar from the base template and the homepage as necessary as other url mappings were made. Overall, I believe I did a fair share of the work, around 16% (or 1/6) of the total. Due to the homepage not being overly complicated, I tried to pick up fixing those issues and anything else that could be done to help the team.

Damain Moquin

I contributed to the team's work for project 2 in a variety of ways, with my main focus being on the event page and its related models. I helped come up with the initial ideas for the models and corresponding fields, and decided on all of the fields related to the event model and page. With some help from other members, I implemented the event model and corresponding fields, and set up the url mapping for the event page as well as its view. I added some mock data for the event fields to the admin site. Most importantly, I created the template for the event detail page, using the fields corresponding to the event model in conjunction with my previous code for the mock event page, to create a unique detail page for each of the events. I also identified that some additional python packages would need to be downloaded by the graders to properly run the server.

It is hard to come up with a concrete percentage of work done in a project with so many dynamic pieces, but I would estimate my contribution to be around 16%. I would say that I was somewhere in the middle regarding contribution, doing a modest share of the work.

Evan Moore

My contributions to the project involved initially designing and coding much of our data model (such as the User, Event, Song, Album, Artist, and Genre classes) as well as adding some mock data for each of them. This included five users, five events, and five albums (which included an artist and genre for each.) Other team members added to the original models, as well as contributed more mock data to our admin site, as necessary for their sections of the site, as well as helping to assist with problems like images not saving correctly in my original implementation as they arose.

Continuing the work with the parts of the site I was responsible for in the mock UI, I implemented the map element of our site (on both the home page and dedicated map page) using the django-leaflet library. This involved adding to the index.html template that Kyle wrote as well as creating the maps.html templates as well as the necessary view and URL mapping for that page. Finally, I wrote the draft of our team write-up. I would estimate my contributions to be roughly 16%, or 1/6 of the total work done for the project, and feel comfortable with both my own as well as my fellow team members' contributions overall.

Jack Nguyen

The work I contributed for this project submission included creating the template for the people page, setting up the url mapping and views for the people page, and drawing the model diagram. I was able to accomplish all of this with the help of other team members. I used the fields in the user and song models created by my team members in order to create my people page template. I drew the model diagram based on the fields that were already implemented in the models by my team members.

Overall, I think everyone did a fair share of work and I would estimate my contribution to be around 16%.