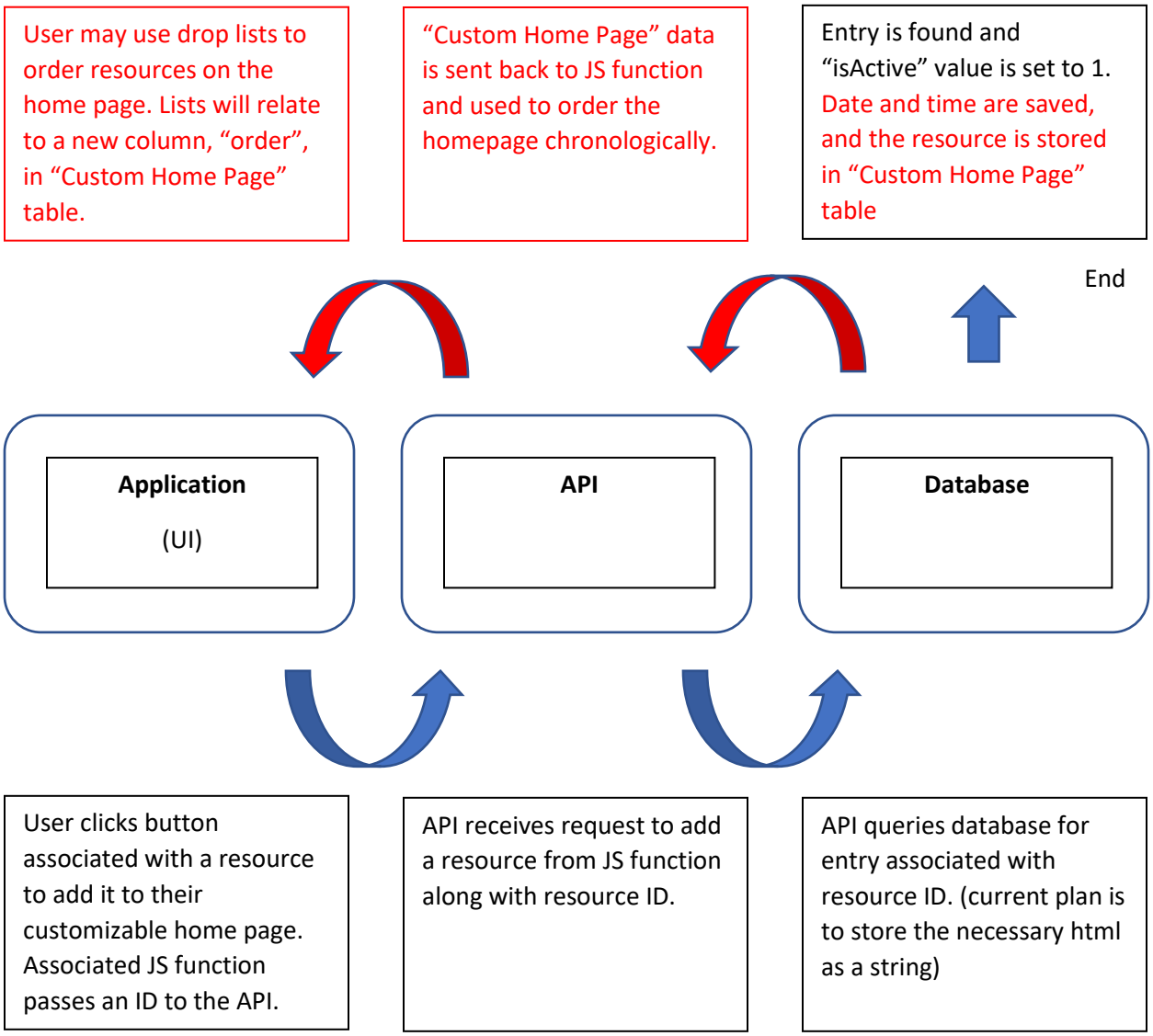
Connor Simmons

Music Theory Query

Service Layer Redesign

Below is a basic diagram of the service layer for the most important feature of the application. Below the diagram is a more detailed look at the service layer in general and breakdown of the diagram. The redesign means that fewer steps are required to prepare resources for insertion to the homepage. Steps in red indicate potential steps for stretch features I am considering.



Services Summary

The main service provided by the minimum viable product (MVP) of my application involves retrieving resources from the application database and displaying them on the user's customized homepage/dashboard. The process is initiated by buttons attached to the resources, which are displayed on separate pages from the home page. When the user finds a resource they want displayed on their home page they click on the appropriate button which will trigger a JavaScript function. The function will take an ID from the resource and pass it to the API to be used in querying the database. When the resource entry is found in the database its "isActive" value is changed to 1. Entries with an "isActive" value of 1 will be retrieved for the homepage during its onload function call. The onload function will query the database for any resources with an "isActive" value of 1. If any entries are found they will be loaded into the HTML of the home page. Additionally, the home page will offer user account services. The log in/account creation interface will include a button which will call a function taking the user input as parameters. This input will be passed through the API to query the database and authenticate the user.

Error handling/Additional services

Additional services may be included but are not part of the design of my MVP. In my current design there is little input taken directly from the user, rather they choose from lists of options which will be fed into functions. Because the options will be hard coded there should be no need for error messages in the final product save for user authentication. Error messages may however be used for testing and would return the ID of the resource that caused a problem. Errors from user authentication will follow a format like this: "Incorrect username and password combination, please try again" or "The username **** was not found, please try again".

Stretch Features

During the redesign process I have thought of a few new features that I would like to add to the application time permitting. These stretch features mostly deal with increasing the customization options of the homepage. For example I am considering using the "Custom Home Page" table, now mostly unnecessary for the MVP, to allow the user to order/rearrange the resources they have stored on their homepage.

Endpoints

Every page on the site will have a service endpoint. Pages displaying resources will all be capable of initiating customization services. The home page is the location many services will terminate, and user authentication and homepage customization services will also have an endpoint here.

- /home
- /home/login
- /home/createaccount
- /tuning
- /scales

- /intervalcalculator
- /chordbuilder