datTopherPack

Kyle Lee | Jessica Yu | Vedant Kothari | Will Nzeuton
TARGET SHIP DATE: 2024-12-18

Description:

Our project is essentially a news site, such as the New York Times, which displays data from a variety of topics including weather, current events, sports, and arts. Each of these pages contain information from reliable APOs, which can all be found in our API knowledge base. In the arts page specifically, we utilize the Google Search API so that you can search up any artwork you find interesting and read a description on the web about it. Google Search API will also be used on the sports page to search for specific sports and game sessions for those sports. By logging in to the page, each user can 'favorite' their posts to view later, but this feature is not possible if you are not logged in.

Components and Interactions:

Database:

- **Description:** This module handles the back-end user information and their corresponding favorites
- **Interactions:**
  - With User Authentication to verify log-in credentials and create new users.
  - With User Authentication to manage favorites

User Authentication:

- **Description:** This module will handle user registration (logging in, signing up, etc.) and verify user permissions. Manages sessions.
- **Interactions:**
  - With FEF for logging in and signing up.
  - With Database to verify log-in credentials and create new users.
  - With Database to allow users to manage their favorites

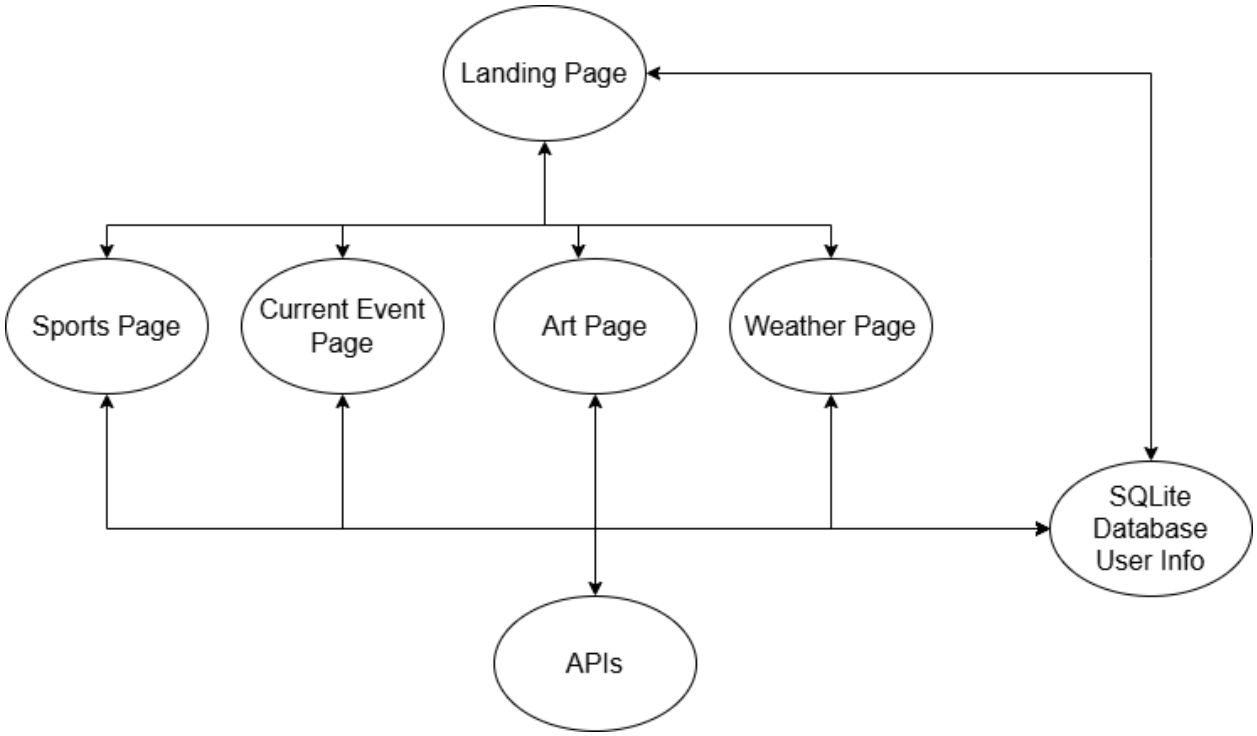
Front-end framework:

- **Description:** This includes all of the styling through Foundation and CSS, and essentially ensuring a smooth interaction for the user on the website.
- **Interactions:**
  - With objects on the screen (scroll bar, navbar, etc.)

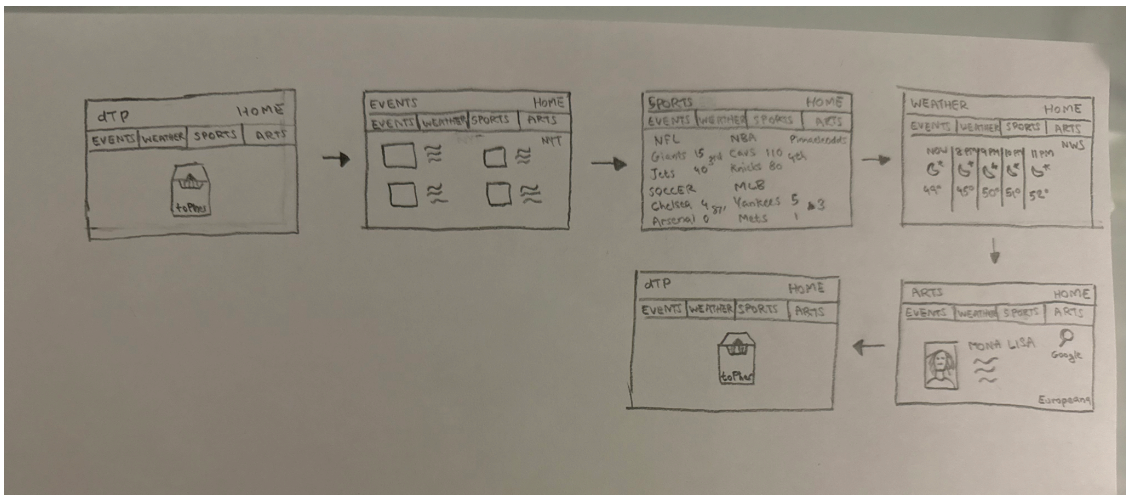
API Integration Module:

- **Description:** This module handles the incorporation of various APIs into the website.
- **Interactions:**
  - With Database to store user information that is collected
  - With flask to switch between web pages that show the different APIs
  - With frontend to display the contents of the API

Map:



Site Map:



APIs of Choice:

Google Fonts API
Pinnacle Odds (Sports)
National Weather Service (Weather)
Europeana API (Arts)
SearchAPI (Google Search)

Front-end Framework:

We will be utilizing Foundation for this project because we realized during k23 that Foundation is easy to implement yet contains a ton of flexibility and customization. We think it might be better to use Foundation for our project if we want to make it look unique, due to the things we can do with certain features such as the navbar and scrolling features of our website.

Database Organization:

Tables

users:

id	username	password	email
#	TEXT	STRING	TEXT

Note: The table below is solely for user "favoriting" functionality and is not for the stories, art, and other information displayed on the home pages, those stories will instead be pulled directly from the APIs and connected to the FEF through our Python middleware; not passing through the database.

favorites*:

id	user_id	content_type	metadata	created_at
#	#	STRING	JSON	DATE

*Each row of **favorites** refers to a single story, art, or sport event (**content_type**) favorited by **user_id**. **metadata** contains more detailed information about each story, art, or sport event and is stored as JSON to allow for flexibility and variation in the type of information stored in each post (images, text, descriptions, titles, scores, etc.)

TASK ASSIGNMENTS:

TASK	Kyle Lee	Jessica Yu	Vedant Kothari	Will Nzeuton
Set up Flask and SQLite3 environment	X			X
Build User Authentication Functionality				X
API Configuration and Connections		X		
Middleware configuration	X	X		
Build Database				X
Frontend (HTML Templates)	X		X	
Frontend (CSS + FEF)			X	
Final Testing and Bug Fixing	X	X	X	X