CITYBYTES

Software Code Documentation

Team Members:-

Nirav Shah	**nshah28**
Vishwa Gandhi	**vgandhi**
Pradyumna Khawas	**ppkhawas**
Vrushanki Patel	**vpatel25**
Priya Saroj	**pbsaroj**

INTRODUCTION:

Moving to a new location can be a daunting endeavor, especially when you have the entire United States to choose from. Finding a new home from scratch while prioritizing certain aspects might be very challenging given the variety of nations and cities. However, with the advancement of technology, information from earlier times can now be leveraged to offer a number of vital insights about a certain location. Our project succeeds in one of those objectives. We seek to present that information in our project because there are many other elements that are taken into consideration when choosing a place to reside, such as weather, temperature, entertainment options, landmark locations, education, and many more. The project is totally created using a variety of technologies, including some of the accessible APIs that are utilized to fetch real-time data.

Although this project is still in its early phases of development, it can be expanded up even further by including multiple features that can benefit society in a variety of different ways. This article offers a critical viewpoint that users can use to comprehend the project, adopt it as open source software, and add further features before releasing it to the market. The document also serves as a starting point for the project and helps developers understand the code.

The technologies listed below were used to build the entire project, and it is advised that the group of developers who take on this project in the future retain these tools on hand.

- ➢ Python3
- ➢ Django
- ➢ Pytest

- ➢ HTML
- ➢ CSS
- ➢ JavaScript
- ➢ BootStrap

Although we have used HTML, CSS and Bootstrap for the frontend logic the user can use any technologies and combine it with backend such as Angular, React etc.

CODE FUNCTIONALITIES:

The project basically uses Django for the backend logic. Django is a Python framework that is useful to build websites. It uses model view and templates architecture. Some of the files used in Django are as follows:-

- ➢ Views.py:- Backend logic is being written in this file. It takes http requests as arguments, imports relevant models and figures out what data needs to be sent to the template and returns the final result.
- ➢ Models.py:- Django has Object Relational Mapping which makes it easier to work with databases.The model provides data from the database.
- ➢ templates/:- They are used to store templates for frontend development and html files.
- ➢ Urls.py:- They are used to set the urls of the django project and apps. The endpoints created are stored in the urls files.
- ➢ Tests.py:- They are used to write the test cases for the code. They are used to check the code.
- ➢ Settings.py:- It has the basic information about the various api's used and initializes the environment required for the project.
- ➢ CityByte code functionalities:- Firstly, there is a project that needs to be created in Django named CityByte. Then there are

two apps named as info and search in the directory. The urls are set in the urls.py file. Both the apps use utils and helper functions which are used to call the api to fetch the real time data. The frontend logic is written using the JavaScript, HTML , CSS and Bootstrap in the templates directory. Most of the backend logic is written in the views.py file which uses utils and helpers to give the desired output. We will be talking about different endpoints which are rendered using functions in the views file. Hence this section of the documentation will further explain various functions written in the views file.

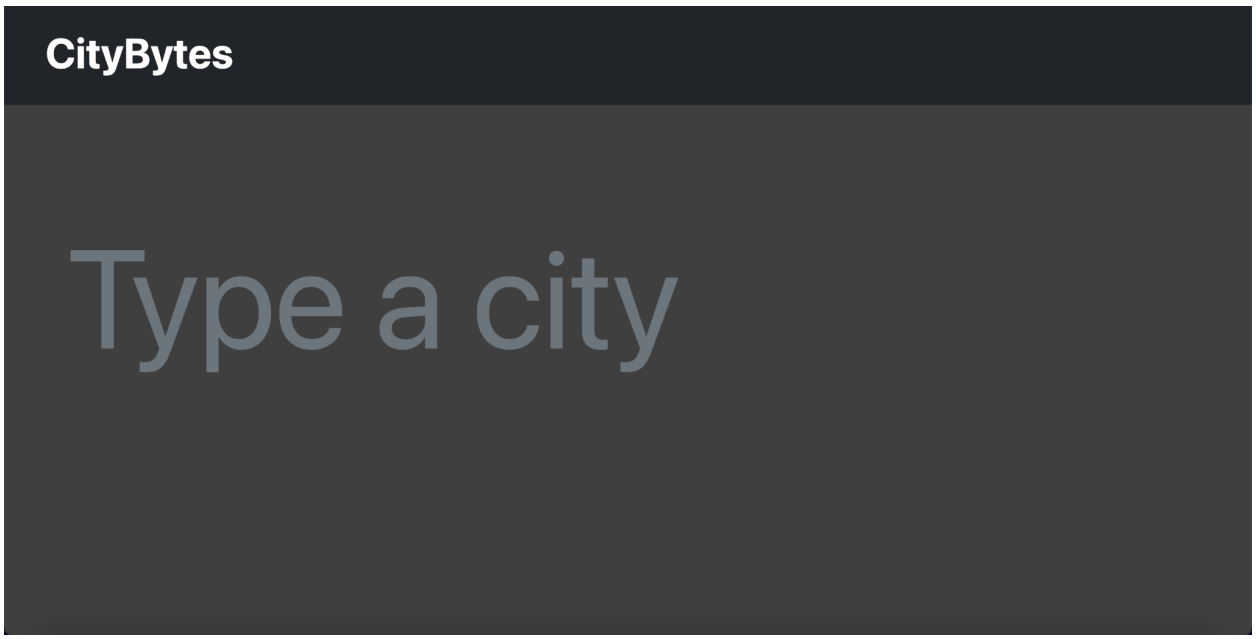
I. def main_page: Main page is used to render the HTML template which has some front-end logic written in javascript. It is used to run the first page of our website.
II. def info_page: Info page takes two request params namely city and country, which provides city and country respectively. The “weather info” variable includes the WeatherBitHelper function which uses WeatherUtilBase to get the data of city and country. The weather_info delivers the information of sunrise, sunset, and timezone of that particular city. The “dining info” variable takes the FourSquarePlacesHelper function that uses FourSquarePlacesUtilBase to get the dining information of that city. The variable “outdoor info” receives the information about places to visit outdoors from the FourSquarePlacesHelper function that uses FourSquarePlacesUtilBase. Similarly, “airport info” and “arts info” variable stores the information regarding airport and arts places respectively, from the FourSquarePlacesHelper function that uses FourSquarePlacesUtilBase. The “photo link” variable generates the photo of the entered city using the UnplashCityPhotoHelper function which uses PhotoUtilBase in the Unsplash class. All the following variables is passed to context using jinja templating.

III. def city_suggestions: City suggestion function receives the user city input, and request to the GenericDBSearchAutoCompleteHelper function that uses SearchAutoCompleteHelperBase class, which collects all the necessary information of that particular city and provide JSON response, which is reflected in the first page of our website.

IV. def city_photo: The City photo function uses the “photo link” variable that receives the information from UnplashCityPhotoHelper, and returns the JSON response which includes the path of the photo link and not the photo itself.

V. def place_photo: The function place photo has the “photo link” variable that uses fsq_id unique ID for that particular place and renders the photo link.

The below screenshots give the glance of the working of our project:

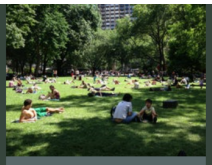


CityBytes

← CityBytes

Temperature	RealFeel Temperature	Pressure	Wind Speed	Wind Direction	Cloud Coverage	Precipitation	UV Index	Sunrise	Sunset
13.7 °C	13.8 °C	1010.9 mb	1.7 m/s	47 °	54 %	0 mm/hr	0	06:00 AM	05:35 PM

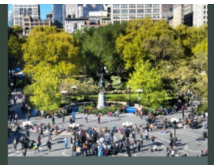
Top Rated Dining Spots



Madison Square Park

Madison Avenue (btwn E 23rd & E 26th St), New York, NY 10010

Park



Union Square Park

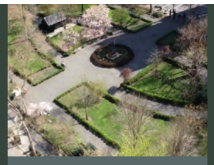
E 14TH St (btwn E 14th & E 17th St), New York, NY 10003

Playground

Washington Square Park

5 Avenue A (btwn MacDougal St & University Pl), New York, NY 10009

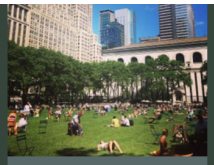
Park



Gramercy Park

Lexington Ave (btwn Gramercy Park E & W), New York, NY 10010

Park



Bryant Park

E 42ND St (btwn 5th & 6th Ave), New York, NY 10018

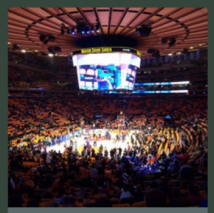
Park

Top Entertainment Spots

Fotografiska

281 Park Ave S (btwn E 21st & E 22nd St), New York, NY 10010

Arts and Entertainment



Madison Square Garden

Two Pennsylvania Plaza (btwn 7th & 8th Ave), New York, NY 10121

Music Venue

IFC Center

323 Avenue of the Americas (at W 3rd St), New York, NY 10014

Drive-in Theater

Lyric Theatre

213 W 42nd St (btwn 7th & 8th Ave), New York, NY 10036

Music Venue

Barcade

148 W 24th St (7th Ave), New York, NY 10011

Arcade

Top Rated Dining Spots

The Raines Law Room

48 W 17th St (btwn 5th & 6th Ave), New York, NY 10011

Cocktail Bar

Gramercy Tavern

42 E 20th St (btwn Broadway & Park Ave), New York, NY 10003

Bar

Daily Provisions

103 E 19th St (btwn Park Avenue S & Irving Pl), New York, NY 10003

Bakery

Caffè Panna

77 Irving Pl, New York, NY 10003

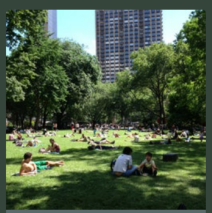
Coffee Shop

Stumptown Coffee Roasters

18 W 29th St (at Broadway), New York, NY 10001

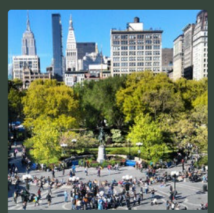
Burger Joint

Top Landmark Spots



Madison Square Park

Madison Avenue (btwn E 23rd &

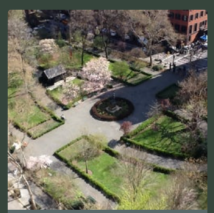


Union Square Park

E 14TH St (btwn E 14th & E 17th

Washington Square Park

5 Avenue A (btwn MacDougal St



Gramercy Park

Lexington Ave (btwn Gramercy

Bryant Park

E 42ND St (btwn 5th & 6th Ave),