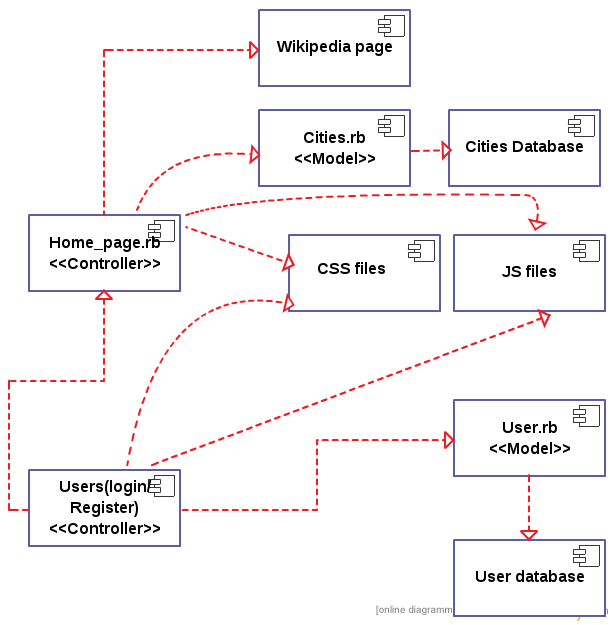
**Product Design**

|  |  |
| --- | --- |
|  | **Team 46: Akanksha, Arushi Vashist, Ayush Tripathi, Dhruv Das, Siddharth Bhatore** |
|  |  |

# **Architectural Model**

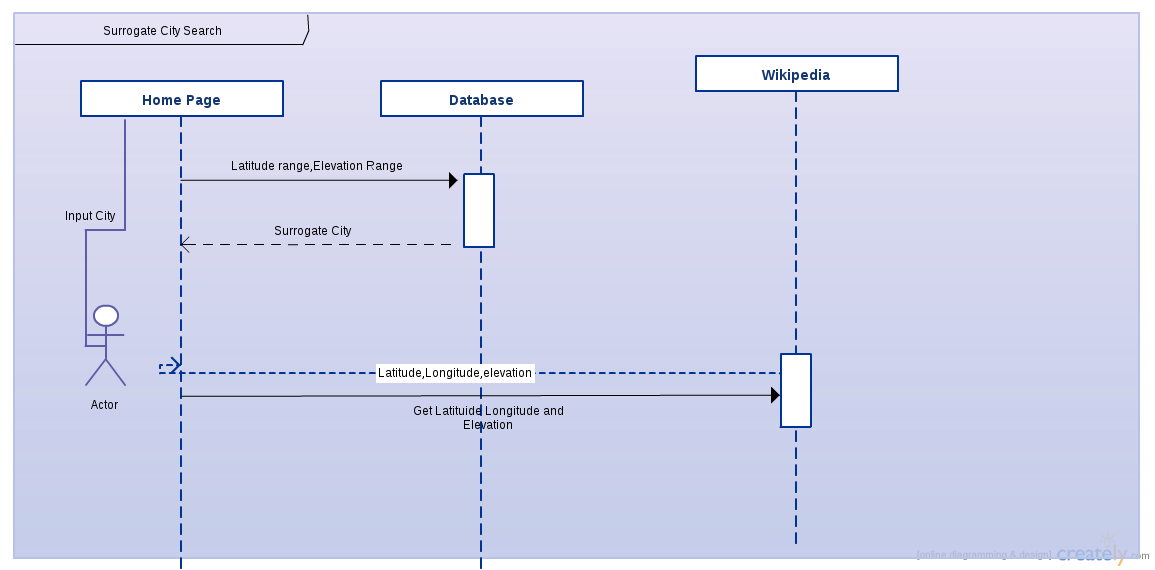


|  |  |
| --- | --- |
| User (Login/Register)  <<controller>> | Component state   * The controller that has the functions/methods which control the user database, allows existing users to login and use the app and also allows new user to register.   Component Behaviour   * User controller interacts with the user by taking input from the login page and the register page. * The controller is used to update or query the Users database during the Login and registration process |
| User Model and Database | Component state   * This contains the information on registered Users.   Component behavior   * The tool is allowed to be used only if the User is logged in. To login to the tool a user must be registered. That is the user must exist in the database. The Login page and Register page sends data to or queries data from the User model and Database. |
| Cities Model and Database | Component state   * The various cities for which Weather Data files are available with us are stored in the Cities database. The various parameters that we pull from the files necessary for our tool like Latitude, Longitude, Elevation, Min-Max Temperature etc are stored with the cities in the database.Also the link to the weather data file for the city is stored in the database.   Component behavior   * This Database is queried when the User requires the surrogate city for the city they have input in the search bar on Home\_Page. The queries are according to the parameters like latitude, elevation etc. |
| Home\_Page.rb  <<controller>> | Component state   * The main functionality of the tool is to take in the user input of a city and output the link of the weather data file if it exists or recommend a surrogate city whos climate matches the climate of the queried city most closley * The controller contains functions and methods like Searching the database using name or searching for the cities lying within a particular range of llatitude and longitude. Search within a radius or by elevation and also matching Min/Max temperatures. * To get the information like latitude , longitude,elevation and temperature profiles of cities not exiting in city database the controller will fetch data by parsing wikipedia pages of the cities queried.   Component behavior   * The controller will interact with the Home Page view where it will take user input for the city, latitude range, radius ranges, temperature range and elevation range. * It will interact with the database to query and search for information * It will interact with wikipedia pages for queried cities whos information does not exist in database. |
| CSS And JS Files | Component state   * These contain defined CSS classes and the javascript model used by views of the site.   Component behavior   * The view involves these component to help improve the user interface of the website. |

# **Class Diagram(s)**

This diagram is present in the diagrams folder of the repo.

# **Sequence Diagram(s)**



# **Design Rationale**

The problems we have been facing:

1. To mark, cities(15000 calculated) whose weather data files are present, on the map.
2. Time taken to download and extract information from the epw files is large.