**SERVICE DIRECTORY- TEAM 5**

| Service Directory | | | | |
|---|---|---|---|---|
| **Address:**  http://webstrar5.fulton.asu.edu/index.html | | | | |
| **Team 5** | | | | |
| **Provider Name** | **Service Name** | **Try It** | **Service Description** | **Planned resources to implement service** |
| Rishti Gupta | **Top10Words:** **Input:** A web page url string **Output:**An array of strings that contains the ten most-frequently occurred words in descending order of their frequencies | **http://webstrar 5.fulton.asu.edu /page8/Top10W ords.aspx** | Analyze the webpage at a given url and return the ten most-frequently occurred words in the webpage. Return the words in the descending order of their appearing frequencies | Write my own code and use local components. |
| Rishti Gupta | **Find the Nearest Store:** **Input:**two strings **Output:**string message | **http://webstrar 5.fulton.asu.edu /page8/NearestS tores.aspx** | Find the provided storeName closest to the zip code and return the address. If no store is found, return an error message | **API Used:** 1). Getting latitude and longitude from a given zip code: Geocoding API (Google) - https://console.clou d.google.com/ 2). Getting the address of the nearest store taking the store name as the parameter: Google Places API- https://console.clou d.google.com/ |

| | | | | |
|---|---|---|---|---|
| Rishti Gupta | **Weather Service:** **Input:** a U.S. zip code **Output:** An array (or list) of strings, storing 5-day weather forecast for the given zip code location | **http://webstrar5.fulton.asu.edu/page8/FindDistanceAndDuration.aspx** | Get a 5-day weather forecast service of zip code location | Retrieve information from: https://openweathermap.org/api |
| Rishti Gupta | **FindDistanceAndDuration:** **Input:** Address/zip for origin and destination **Output:** Distance and Travel Duration between between origin and destination | **http://webstrar5.fulton.asu.edu/page8/WeatherConditions.aspx** | Get the distance and travel duration between the origin and destination by taking in either the address or zip code of origin and destination. | **API Used:** 1). Getting latitude and longitude from a given zip code: Geocoding API (Google) - https://console.cloud.google.com/ 2). Getting the distance and travel duration information from Distance Matrix Google API: https://console.cloud.google.com/ |
| Arun Deepak Chandrasekar | **NewsFocus:** **Input:** a list of topics or key words **Output:** A list of URLs in which the given topics are reported | **http://webstrar5.fulton.asu.edu/page9/NewsFocusTryIt.aspx** | Find news links on specific topics | Retrieve information from : http://content.guardianapis.com/search?q= |
| Arun Deepak Chandrasekar | **NaturalHazards:** **Input:** latitude and longitude **Output:** a number reflecting the natural hazards at the location | **http://webstrar5.fulton.asu.edu/page9/NaturalHazardTryIt.aspx** | Create a service that returns the natural hazards (Tsunamis, earthquake, volcanoes) index of a given | Use the service from USGS site: https://earthquake.usgs.gov/ws/ |

| | | | position using latitude and longitude | |
|---|---|---|---|---|
| Arun Deepak Chandrasekar | **GasStationService:** **Input:** address string **Output:** get nearby gas station details as list of strings- station name, street address, fuel type, station type | **http://webstrar5.fulton.asu.edu/page9/GasStationTryIt.aspx** | The restful service takes an address as input and returns details of nearby gas stations including gas station name, street address, fuel type eg. LPG, CNG, electric, station type eg. grocery store. | 1.Getting latitude and longitude from a given address using: Geocoding API (Google) - https://console.cloud.google.com/ 2.Retrieve information from: https://developer.nrel.gov/docs/transportation/alt-fuel-stations-v1/nearest/ |
| Arun Deepak Chandrasekar | **UserRegistrationService:** **Input:** string username and password **Output: -** boolean value representing successful user registration | **http://webstrar5.fulton.asu.edu/page9/UserRegistrationServiceTryIt.aspx** | The WSDL/SOAP service registers the users by storing user credentials in a JSON file named database.json in the server. | Write my own code and use local components. |