**Name – Abhay Deomane**

**Food Price API Documentation**

**Project link –**

https://food-price-frontend.onrender.com/

**Project code-**

<https://github.com/Abhaydeomane/food-price-api>

**API Documentation**

**Base URL**

The base URL for all endpoints is **/**.

**Endpoints**

1. GET **/**

* **Description**: Check server status.
* **Request Type**: GET
* **Response**:
  + Status: 200 OK
  + Body: "server is running"

2. POST **/cost**

* **Description**: Calculate delivery costs.
* **Request Type**: POST
* **Request Body**:
  + zone: String (required) - The delivery zone.
  + organization\_id: Integer (required) - The ID of the organization.
  + total\_distance: Numeric (required) - The total distance for delivery.
  + item\_type: String (required) - The type of item.
* **Response**:
  + Status: 200 OK
  + Body:
    - total\_price: Numeric - The calculated total price for delivery.
  + Possible Errors:
    - 400 Bad Request: If any required parameter is missing.
    - 404 Not Found: If pricing information is not found for the specified organization and zone.
    - 500 Internal Server Error: If there is an internal server error.

**Test Suite**

A comprehensive test suite should cover the following scenarios:

1. Verify server status endpoint **/** returns "server is running".
2. Test the calculation of delivery costs using various scenarios:
   * Verify correct calculation when total distance is within the base distance.
   * Test correct calculation when total distance exceeds the base distance.
   * Ensure correct handling of missing parameters in the request.
   * Check handling of cases where pricing information is not found for the specified organization and zone.
   * Test for internal server errors during calculation.

**Setup Guide**

Postgresql database is created on ElephantSQL cloud.

node js server hosted on render.com

Frontend is also hosted on render.com

**To run the project just go on below link**

https://food-price-frontend.onrender.com/

**To run the project on local machine**

1. Install Node.js and npm if not already installed.
2. Run **npm install** to install project dependencies.
3. Set up PostgreSQL locally if not already installed.
4. Create a PostgreSQL database and note down the connection string.
5. Create a **.env** file in the project root directory and add the following variables:

PORT=5000

DB\_CONN=<your\_database\_connection\_string>

1. Modify the connection string (**<your\_database\_connection\_string>**) with your PostgreSQL database connection string.
2. Create the table and Enter the entries in PostgreSQL database
3. Run the database migration to create necessary tables by executing **npm run migrate**.
4. Start the server by running **npm start**.
5. The server should now be running at [**http://localhost:5000**](http://localhost:5000).
6. To run index.html properly you have to change the frontend.js file and update server api url to <http://localhost:5000/cost>