|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| S\_No | P\_Id | P\_Name | E\_Discount | C\_Discount | F\_Discount | Description | Imp | Operations |
| 1 | 1 | Diwali Scale | 10% | 20% | 15% | Diwali Scale |  | Edit  Delete |
| 2 | 2 | Pongal Scale | 15% | 15% | 20% | Pongal Scale |  | Edit  Delete |
| 3 | 3 | Christmas Scale | 30% | 25% | 15% | Christmas Scale |  | Edit  Delete |

**DESIGN DOCUMENTATION – UI**

**Group**

Add Promotion

AAdd

Promotion ID

Promotion Name

Electronics Discount

Cloths Discount

Furniture Discount

Description

Image

**Promotion API Documentation**

**Promotion API Documentation**

**Overview**

The Promotion API allows users to manage promotions within the system. Promotions can be created, retrieved, updated, and deleted using this API. This documentation provides detailed information on how to interact with the Promotion API.

HTTP API Methods**: POST,GET, PUT,DELETE**

1. **CREATE PROMOTION**

* **POST** : http://localhost:8080/promotions
* Create a new promotion with the provided details.
* Request Body:

JSON

{

"name": "Promotion Name",

"type": "Promotion Type"

}

* Response Body:

{

"id": "promotion\_id",

"name": "Promotion Name",

"type": "Promotion Type"

}

**Example:**

**POST**:

{

"name": "Winter Sale",

"type": "Discount"

}

1. **RETRIEVE PROMOTION**

**2.1 Get all promotions**

* **GET**: http://localhost:8080/promotions/getAll
* Retrieve details of a specific promotion using id from database.
* Response

Status : 200 OK

Body:

{

"id": "promotion\_id",

"name": "Promotion Name",

"type": "Promotion Type"

}

**Example:**

**GET**:

{

"id": "123456",

"name": "Winter Sale",

"type": "Discount"

}

**2.2 Get promotion using id**

* **GET**: http://localhost:8080/promotions/getById/{p\_id}
* Retrieve details of a specific promotion using id from database.
* Response

Status : 200 OK

Body:

{

"id": "promotion\_id",

"name": "Promotion Name",

"type": "Promotion Type"

}

**Example:**

**GET**:

{

"id": "123456",

"name": "Winter Sale",

"type": "Discount"

}

1. **UPDATE PROMOTION**

* **PUT**: http://localhost:8080/promotions/update/{p\_id}
* Update details of an exisiting promotion in database.
* Request Body:

{

"name": "New Promotion Name",

"type": "New Promotion Type"

}

* Response:

Status : 200 OK

Body:

{

"id": "promotion\_id",

"name": "New Promotion Name",

"type": "New Promotion Type"

}

**Example:**

PUT:

{

"name": "Winter Sale",

"type": "Discount"

}

1. **DELETE PROMOTION**

* **DELETE**: http://localhost:8080/promotions/delete/{p\_id}
* Delete a promotion from the system.
* Response

Status : Deleted Successfully.

**Example:**

1.DELETE:

{}

**Architecture for Promotion**

**http request**

Back End

Postman

**POST** : http://localhost:8080/promotions

* **RETURN**: Added Data(JSON{})
* **GET**: http://localhost:8080/promotions/getById/{p\_id}

**RETURN**: JSO **RETURN**:Display data using id

* **PUT**: http://localhost:8080/promotions/update/{p\_id}

**RETURN**:Display updated data

**DELETE**: http://localhost:8080/promotions/delete/{p\_id}

**RETURN**:Delete data using id

**http response failure**

# (**Example**: **404 Not Found error)**

Success

# 

Database

**SCHEMA DIAGRAM**

|  |  |
| --- | --- |
| **PROMOTIONS** | |
| S\_NO  P\_ID  P\_NAME  E\_DISCOUNT  C\_DISCOUNT  F\_DISCOUNT  DESCRIPTIONS  IMG | INTEGER  INTEGER PK AI  VARCHAR(45) NN  INTEGER  INTEGER  INTEGER  VARCHAR(45) NN  BLOB |