**API DOCUMENTATION**

**Allowed HTTPs requests:**

PUT : To create resource

POST : Update resource

GET : Get a resource or list of resources

DELETE : To delete resource

# Description Of Usual Server Responses:

* 200 OK - the request was successful (some API calls may return 201 instead).
* 201 Created - the request was successful and a resource was created.
* 400 Bad Request - the request could not be understood or was missing required parameters.
* 401 Unauthorized - authentication failed or user doesn't have permissions for requested operation.
* 404 Not Found - resource was not found.
* 405 Method Not Allowed - requested method is not supported for resource.
* 500 Internal Server Error – Server doesn’t support current request.

**Recipe:**

Represents recipe details.

**Recipe Table attributes:**

* Recipe Id(Number): unique identifier
* Recipe Name(String) : Name of recipe
* Time(time) : Time of creation of recipe
* Number Of People(Number): Number of servings
* Ingredients(String) : Ingredients required for the recipe
* Instructions(String) : Steps to be followed

**Authorization:**

* Admin Credentials: Username-‘admin’, password-‘admin123’

Access given-All endpoints

* User Credentials: Username-‘user’, password-‘user123’

Access given- Create, Get endpoints

**Recipe Collection:**

1. Create a recipe: Add a recipe

Method allowed – **POST**

Uri- <http://localhost:8100/recipe/add>

Body- {

  "recipeName":"pizza",

  "isVegeterian":"true",

  "noOfPeople":6,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

{

  "recipeName":"milkshake",

  "isVegeterian":"true",

  "noOfPeople":5,

  "ingredients":"milk,oreos,whipped cream",

    "instructions":"1.Fill up the jar with milk.2.Add oreos.3.Blend.4.Serve with whipped cream."

}

**Validations:**

* Recipe Name: Recipe name should contain only alphabets and spaces, and shouldn't start or end with whitespaces.

Error Code: 400

{

  "recipeName":" pizza",

  "isVegeterian":"true",

  "noOfPeople":5,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Number of People: Minimum number of people should be 1.

Error Code: 400

{

  "recipeName":"pizza",

  "isVegeterian":"true",

  "noOfPeople":0,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Duplicate Entry: Recipe name should not be same.

Error Code: 400

{

  "recipeName":"pizza",

  "isVegeterian":"true",

  "noOfPeople":7,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Missing attribute(Recipe name): Recipe name is null

Error Code: 400

{

  "isVegeterian":"true",

  "noOfPeople":7,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Missing attribute(Veg indicator) :

Error Code: 400

{

  "recipeName":"pasta",

  "noOfPeople":7,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Missing attribute(Number of people):

Error code: 400

{

  "recipeName":"taco",

  "isVegeterian":"true",

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Missing attribute(Ingredients):

Error code:400

{

  "recipeName":"pancakes",

  "isVegeterian":"true",

  "noOfPeople":7,

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Missing attribute(instructions):

Error code:400

{

  "recipeName":"milkshake",

  "isVegeterian":"true",

  "noOfPeople":7,

  "ingredients":"milk,oreo,whipped cream"

}

1. Update a recipe: Update attributes

Method Allowed- **PUT**

* Update number of people:

Uri: <http://localhost:8200/recipe/update/people/1>

{

  "noOfPeople":7

}

* Update recipe name:

Uri: <http://localhost:8200/recipe/update/name/1>

{

  "recipeName":"pasta"

}

* Update time:

Uri: <http://localhost:8200/recipe/update/time/1>

{

  "time":"2021-06-14T14:45:15"

}

* Update Veg Indicator:

Uri: http://localhost:8200/recipe/update/vegeterian/1

{

  "isVegeterian":"false"

}

* Update Ingredients:

Uri: <http://localhost:8200/recipe/update/ingredients/1>

{

  "ingredients":"milk,oreo,whipped cream"

}

* Update Instructions:

Uri: <http://localhost:8200/recipe/update/instructions/1>

{

  "instructions":"1.Fill up the jar with milk.2.Add oreos.3.Blend.4.Serve with whipped cream."

}

**Validations:**

* Recipe Name: Recipe name should contain only alphabets and spaces, and shouldn't start or end with whitespaces.

Error Code: 400

{

  "recipeName":" pizza",

  "isVegeterian":"true",

  "noOfPeople":5,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Number of People: Minimum number of people should be 1.

Error Code: 400

{

  "recipeName":"pizza",

  "isVegeterian":"true",

  "noOfPeople":0,

  "ingredients":"bread,sauce,cheese,toppings,spices and herbs",

  "instructions":"1.Make your own pizza dough.2.Keep the sauce and toppings simple.3. Bake it hot."

}

* Recipe does not exist: If no recipe exists for given Id, then exception will be thrown.

Uri:

<http://localhost:8200/recipe/update/people/4>

<http://localhost:8200/recipe/update/name/4>

<http://localhost:8200/recipe/update/time/4>

<http://localhost:8200/recipe/update/vegeterian/4>

<http://localhost:8200/recipe/update/instructions/4>

<http://localhost:8200/recipe/update/ingredients/4>

1. Delete a recipe:

Method Allowed- **DELETE**

Uri: <http://localhost:8300/recipe/delete/1>

**Validations:**

* Recipe does not exist: If no recipe exists for given Id, then exception will be thrown.

Uri:

<http://localhost:8300/recipe/delete/4>

1. Get All Recipes:

Method Allowed: **GET**

Uri: <http://localhost:8400/recipe/get>

* **Validations:** Recipe does not exist: If no recipe exists for given Id, then exception will be thrown.

Uri: http://localhost:8400/recipe/get