Recipe Sharing Assignment

Requirements analysis

Building APIs for a recipe sharing system that provides the following features for users.
● Sign up with personal details and upload profile pic
● Post recipes Each recipe contains:
○ A list of ingredients
○ A list of cooking steps
○ Photos
● Fetch all recipes on the platform
● Fetch recipes filtered by ingredients
● Like, and comment on others' recipes

Data Models

User

To store user data

Field Name	Data type	Description
id	UUID(String)	Unique identifier
firstName	String	First name of the user
lastName	String	Last name of the user
createdAt	Instant	Time of user creation
updatedAt	Instant	Time of user data last modified

UpdateUser

To store user's secret data

Field Name	Data type	Description
username	String	Unique for each user
password	String	Encoded passphrase

Recipe

To store recipe details

Field Name	Data type	Description
id	UUID	Unique identifier
username	String	Username of the recipe posted by user
ingredients	List<String>	Cooking ingredients used
cookingSteps	List<String>	Cooking steps
createdAt	Instant	Recipe added time

updatedAt	Instant	Recipe last modified time
likes	List<Action>	Likes obtained
Comments	List<Action>	Comments obtained

Action

To store action details like "likes" and "comments" on recipe

Field Name	Data type	Description
id	String	Unique identifier
username	String	Username of the action made by
recipeId	String	Recipe id on which the action was made
actionType	ActionType	Enum[LIKE, COMMENT]
comment	String	Comment added
createdAt	Instant	Action created time
updatedAt	Instant	Action last modified time

API Endpoints

User endpoints

API request	API response

POST /user/add
 	
{
   "firstName":<text>,
   "lastName":<text>
}
	{
	   "Message":"user created"
	   "Id": <userId>
	}

POST /user/{userId}/update
 	
{
   "username":<text>,
   "password":<text>
}
	{
	   "Message":"user credentials updated"
	   "Id": <userId>
	}

POST /user/{userId}/image/upload
RequestParam:
"Image": <image file>
	{
	   "Message":"profile pic uploaded"
	   "Id": <userId>
	}

GET /user/{userId}/image/download	{
	   <profilePic>
	}

Authentication Endpoints

API request	API response

POST /authenticate
{
   "username":<text>,
   "password":<text>
}
	{
	   "bearerToken":<bearer-token>
	}

Recipe Endpoints

API request	API response

POST /recipe/add
{
   "Ingredients": [<text>,<text>],
   "cookingSteps":   [<text>,<text>]
}
	{
	   "Message":"recipe created"
	   "Id": <recipeId>
	}

POST /recipe/{recipeId}/update
{
   "Ingredients": [\<text>,\<text>],
   "cookingSteps": [\<text>,\<text>]
}
	{
	   "Message":"recipe updated"
	   "Id": \<recipeId>
	}

PUT /recipe/{recipeId}/action
{
   "actionType":"LIKE"
}
	{
	   "Message":"recipe updated"
	   "Id": \<recipeId>
	}

PUT /recipe/{recipeId}/action
{
   "actionType":"COMMENT",
"comment":\<text>
}
	{
	   "Message":"recipe updated"
	   "Id": \<recipeId>
	}

POST
/recipe/{recipeId}/images/upload
RequestParam:
"Images" : \<Multiple images>
	{
	   "Message":"images uploaded"
	   "Id": \<recipeId>
	}

GET
/recipe/{recipeId}/images/download
	{
	  \<Images>
	}

GET /recipe/list	{
	
	  "data": [
	{
	 "Id" : \<UUID>
	  "Username": \<text>,
	  "ingredients":[\<text>],
	  "cookingSteps": [\<text>],
	  "createdAt" : \<DateTime>
	  "updatedAt": \<DateTime>,
	  "Like" : [Action],
	  "comments": [Action]
	}
	]
	}

POST /recipe/list_with_filter
{
  "Ingredients": [\<text>]
}
	{
	
	  "data": [
	{
	 "Id" : \<UUID>
	  "Username": \<text>,
	  "ingredients":[\<text>],
	  "cookingSteps": [\<text>],

	"createdAt" : <DateTime> "updatedAt": <DateTime>, "Like" : [Action], "comments": [Action] }] }

DELETE /recipe/{recipeId}	{
	"Message":"deleted recipe",
	"id": <recipeId>
	}

Database Tables

User

Field Name	Data type	Description
id	UUID(String)	Unique identifier
firstName	String	First name of the user
lastName	String	Last name of the user
username	String	Unique name
password	String	Encoded password string
createdAt	Instant	Time of user creation
updatedAt	Instant	Time of user data last modified

Recipe

Field Name	Data type	Description
id	UUID	Unique identifier

username	String	Username of the recipe posted by user
ingredients	List<String>	Cooking ingredients used
cookingSteps	List<String>	Cooking steps
createdAt	Instant	Recipe added time
updatedAt	Instant	Recipe last modified time

Action

Field Name	Data type	Description
id	String	Unique identifier
username	String	Username of the action made by
recipeId	String	Recipe id on which the action was made
actionType	ActionType	Enum[LIKE, COMMENT]
comment	String	Comment added
createdAt	Instant	Action created time
updatedAt	Instant	Action last modified time

ER Diagram

- User creates many recipes
- Actions are applied to Recipes
- Users post many actions to a recipe

User
id
first_name
last_name

Recipe
id
ingredients
cooking_steps

Action
id
action_type
comment

creates

makes

applied to