

IT314-Software Engineering

Lab 3 : User Stories

Lab Group : 3

Project Group 13 :

Project : The Naive Baker

Group Member's Name and ID :-

- 202101186 - Nikunj Sheer
- 202101231 - Rohan Mistry
- 202101170 - Abhishek Dalsaniya
- 202101211 - Prashuk Jain
- 202101184 - Devansh Dalwadi
- 202101195 - Ustav Maru
- 202101183 - Harsh Patel
- 202101214 - Harikrushna Suhagaya
- 202101218 - Swet Lakhani
- 202101228 - Harshkumar Chavda

Actor	User Stories	Success Scenarios	Failures Scenarios
Busy Professional	As a busy professional, I want to quickly find recipes using the ingredients I have, to prepare a meal in a short amount of time. So that I can be punctual with the office work.	I input my available pantry items and select a time filter. I find a recipe that matches my criteria and time constraints, allowing me to cook efficiently.	The system's ingredient suggestions are inaccurate or not relevant to my pantry.
Adventurous Cook	As an adventurous cook, I want to discover new ingredient combinations to experiment with. So that I can create unique dishes.	The system suggests an ingredient I hadn't considered before. I use it in a recipe, and the resulting dish is a delightful surprise, sparking my culinary creativity.	The ingredient suggestions are repetitive or unappetizing.
Novice Cook	As a novice cook, I need clear and detailed instructions to successfully prepare dishes, helping me gain confidence in the kitchen.	I follow the step-by-step instructions provided by the system, and my dish turns out well. The clear guidance boosts my cooking skills and self-assurance.	The recipe instructions are confusing or incomplete, causing me to struggle while cooking and potentially ruin the dish.
Home Chef	As a home chef, I want to save and organize my favorite recipes for future use, creating a personalized collection of go-to dishes. So that I	I save recipes I love and organize them within the system. Whenever I want to cook something I enjoyed, I can easily	The system's recipe-saving feature is unreliable or difficult to use. I lose access to my saved recipes.

	can reuse the receipes.	find the recipe and recreate the dish.	
Simple User	As a simple user, I want a list of recipes so that I can prepare recipes in less time.	I get a list of recipes as per my filters.	Recipe suggestions are inaccurate and take more time than the mentioned time in the filter.
	As a simple user, I can give feedback on the recipes so that other users get an idea about the recipe.	I am able to enter feedback to the system successfully and other users are successfully able to see the feedback.	I am not able to give feedback to the system.
Registered User	As a registered user, I want to add my own recipes so that people can see my recipes	I can successfully add my own recipe.	I am not able to add my recipe properly. It shows added but when I search for it then it does not show up.
Fitness freak	As a fitness freak, I want that system gives nutrients lists of that specific recipe.	I can successfully get the nutrients lists of the particular recipe and according to that I can choose my recipe.	System doesn't show correct nutrients lists of that specific recipe.

Functional Requirements :-

- 1) User Registration and Authentication :-
 - User can create his/her account and password
 - User can login with the credentials
- 2) Recipe catalog :-
 - User can search recipe with name
 - User can search recipe by applying filter(preparation time, Ingredients, meal type, category)
 - User can upload recipe
 - User can give feedback for a particular recipe
- 3) Notification system :-
 - User will be notified about their queries submitted to us
 - User will also be informed for their recipe uploading
- 4) User Profile(only for admin) :-
 - Admin can manage all profiles for contacting the user
 - Admin can address the issues by referring from database

Non-Functional Requirements :-

- 1) Performance :-
 - The system should load quickly, even during high traffic periods
 - Response time for searching and uploading recipes should be less.
- 2) Security :-
 - There should be no loss of data during their session.
 - User data should be stored securely and encrypted.
- 3) Scalability :-
 - The system should be able to handle for more number of users
- 4) Reliability :-
 - The information provided should be well reliable and well understood by the user.
- 5) Usability :-
 - The user interface should be user-friendly and should be responsive on different device.
- 6) Load testing :-
 - Testing should be done on the number of users increasing every time and network traffic on the website should be maintained.

USE CASES:

1) Login:

Use Case: Login, Database

Actors:User

Description:This use case describes the process of a user logging into the system.

Main Flow:

- The user opens the application or website.
- The system presents a login screen.
- The user enters their username or email address.
- The user enters their password.
- The user clicks the "Login" button.
- The system validates the user's credentials using database.
- If the credentials are valid:
- The system logs the user into their account.
- The system records the login event.

Alternate Flow:

- If the system detects too many failed login attempts:
- If the credentials are invalid then the system displays an error message and the user is prompted to re-enter their credentials..
- The system may temporarily lock the user's account for security reasons.
- The user is informed and may need to follow a password reset process to regain access.

Login Preconditions:

- User Existence: There must be a registered user account with the provided username or email address.
- Valid Input: The username/email and password provided must meet certain criteria, such as minimum length, character requirements, or format constraints.
- Authentication State: The user must not already be authenticated (not logged in)

Login Postconditions:

Authentication Success: If the provided username/email and password match the stored credentials, the user is considered authenticated (logged in).

Authentication Failure: If the provided username/email and password do not match the stored credentials, the login attempt fails, and the user remains unauthenticated.

Session Management: A session is established for the authenticated user, allowing them to access protected resources or perform authorized actions.

Logging and Auditing: Successful or failed login attempts should be logged for security and auditing purposes.

2) Sign up

Use Case Name: Sign-up

Primary Actor: New User, Database

Preconditions:

- The user is not currently registered within the system.
- The system is accessible and operational.

Main Flow:

- User navigates by clicking on a "Sign-up" button.
- User is presented with a sign-up form, which typically includes fields for:
 - Username or email address
 - Password
 - Confirm password
 - First name
 - Last name (optional)
 - Gender (optional)
- User fills out the sign-up form with the required information.
- User clicks a "Sign-up" or "Create Account" button.
- The system validates the user's input, checking for errors or missing information.
- If the provided information is valid, the system creates a new user account.
- The system sends a confirmation email to the user's provided email address (if email verification is required).
- User receives the confirmation email and clicks on the verification link (if applicable).
- The system confirms the user's email address and activates the user's account.
- This new entry of user will be stored in database.
- User is redirected to a logged in page automatically.

Alternate Flow:

Invalid Input: If the user enters invalid or incomplete information, the system displays error messages and prompts the user to correct the issues before resubmission.

Exception Flows: If there are technical issues during the sign-up process (e.g., server errors, network problems), the system displays an error message and may request the user to try signing up again later.

Postconditions:

- A new user account is created within the system.
- The user can log in using their newly created credentials.
- The user may have to complete additional profile setup steps, depending on the system's requirements.

3) Search Recipe:

Use case Name: Search Recipe

Actor : Simple User, Database

Preconditions:

User Authentication: The user must be logged into their account on the website with appropriate permissions to upload a recipe.

Input Availability: There must be a valid input for the search. This could be a search query, keywords, ingredients, or any other relevant information used to search for a recipe.

Recipe Database: A database or source of recipes must be available and accessible. Without a source of recipes, there can be no search operation.

Main Flow:

- The user will go to the Web page of Search recipes.
- User can search recipes with two methods: user can directly search with the recipe name and with a filter.
- User will click on search button after entering search details.
- This search request goes into database.
- Database will give list of recipes to the user according to search request.
- User can select a particular recipe and can save that recipe as favorite recipe and also can give feedback on that recipe.

- If user give feedback then this change will be stored in database and database will send acknowledgement.

Alternative Flow:

- The filter in the search couldn't find the recipe correctly.
- The searched keyword doesn't exist in the database.
- Users enter Invalid input string.
- Site shows recipes but when user click on more details about recipes, it doesn't show properly.

Postconditions:

Search Results: The system returns a list of recipes that match the search criteria. If no matching recipes are found, it should indicate this to the user.

Display of Results: The search results should be displayed to the user in a clear and organized manner, making it easy for them to select a recipe of interest.

Navigation and Interaction: The user should be able to interact with the search results (e.g., view details, save recipes, print recipes) as intended by the application.

4) Upload Recipe:

Use case Name: Upload Recipe

Actor : Simple User, Database

Preconditions:

User Authentication: The user must be logged into their account on the website with appropriate permissions to upload a recipe.

Valid Recipe Data: The user should provide valid and complete recipe information, including a recipe title, ingredients, cooking instructions, and any other required fields.

No Duplicate Titles: The website should ensure that the recipe title is unique and does not conflict with existing recipes.

Valid Category and types: If the website categorizes recipes or uses tags, the user should select appropriate categories and types for the recipe.

Main Flow:

- The user will go to the Web page of Upload recipes.
- The User provides details of recipes like steps to make it, time taken to make the recipe, ingredients ,type of recipe.
- User will click on upload recipe button.
- This request of upload recipe goes to database.
- Database verifies the recipe details and sends acknowledgement back to user.
- User can see the acknowledgement message saying 'Recipe Uploaded Successfully'.

Alternative Flow:

- If the system breaks down while uploading the recipe then data will be lost.
- Recipe uploaded successfully but not visible in actual data.
- Due to server error the recipe doesn't show up.

Postconditions:

Successful Upload: After meeting all preconditions, the recipe is successfully uploaded to the website's database. The recipe should be displayed on the website with all the provided details, including ingredients, cooking instructions, images, and any associated categories or tags.

Visibility: The uploaded recipe should be visible to the user end, if applicable, to other users on the website.

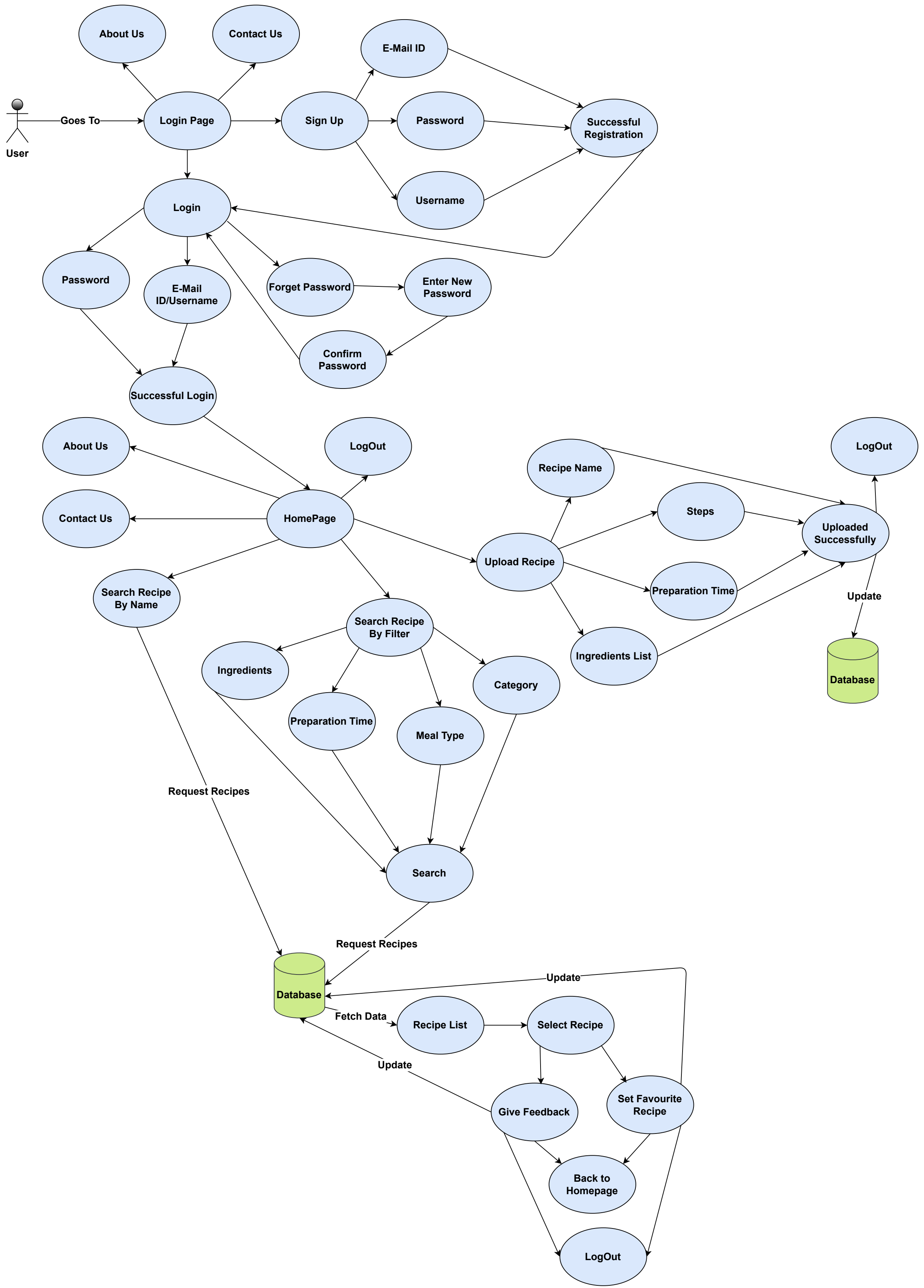
Edit and Delete Options: The user who uploaded the recipe should have the option to edit or delete their own recipe.

Searchable: The uploaded recipe should be searchable by other users on the website, based on keywords, categories, or tags.

Comments and Ratings: If the website supports it, other users should be able to leave comments or ratings on the uploaded recipe.

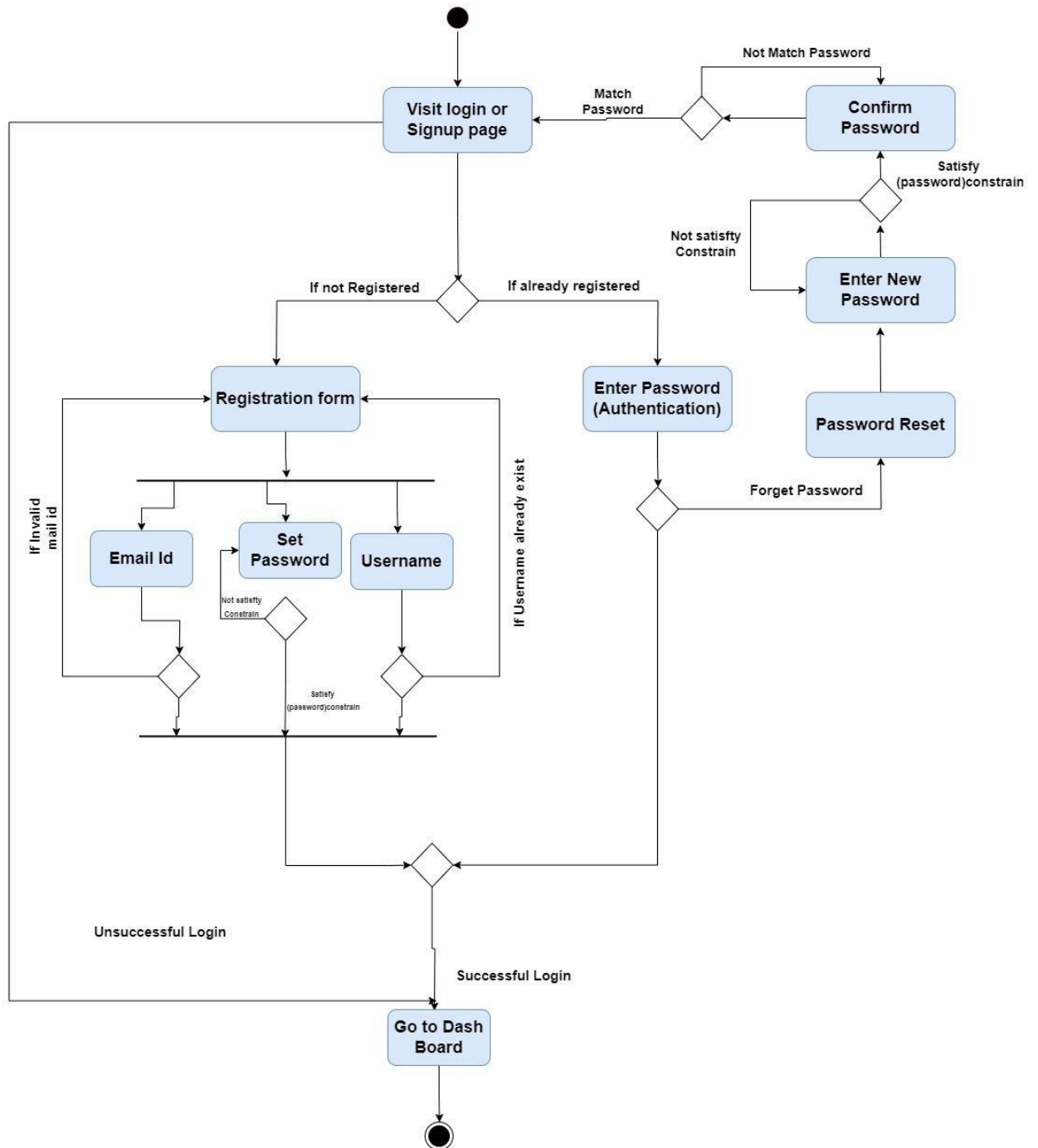
Notification: The user who uploaded the recipe may receive a confirmation or notification of successful upload.

Concept Mapping

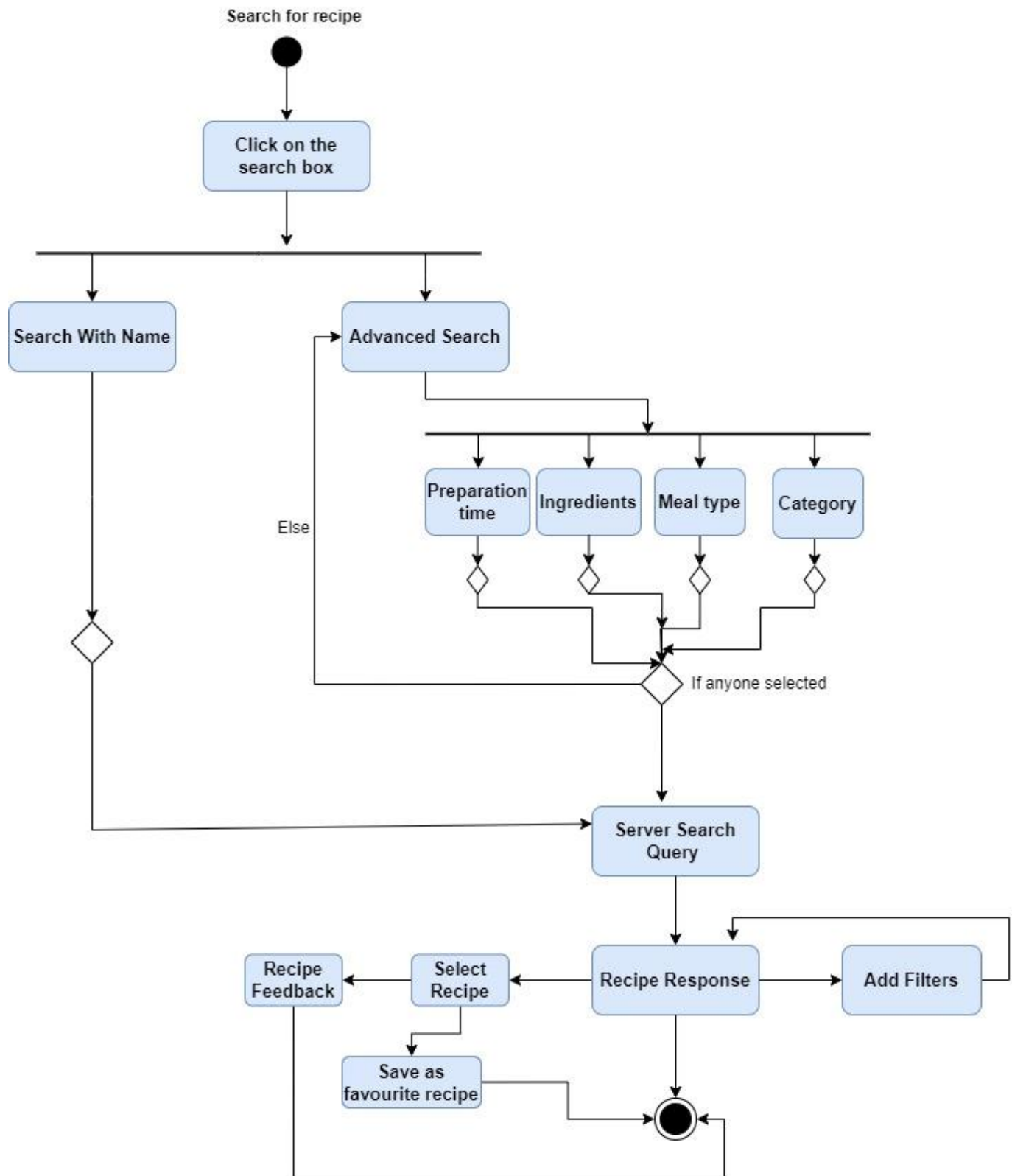


Activity Diagrams:

1) Login-SignUp:



2) Search for Recipe:

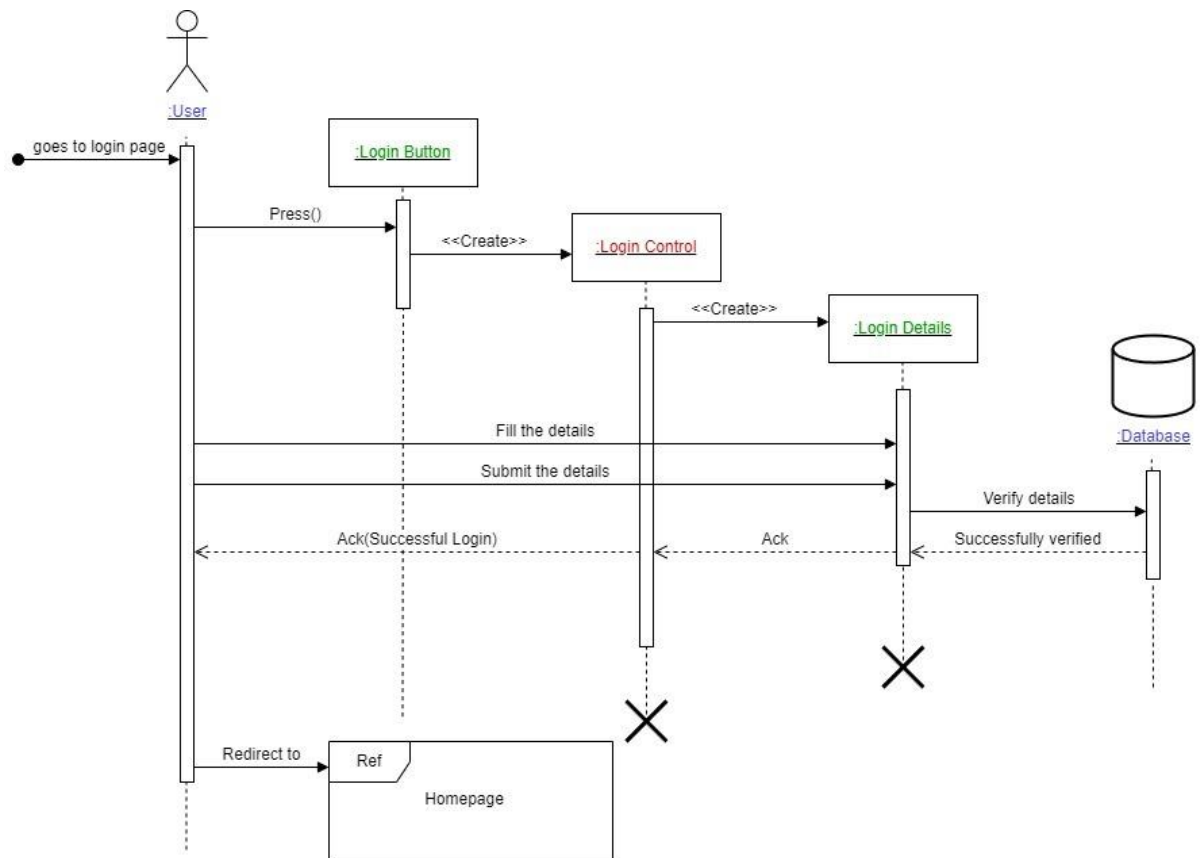


3) Upload Recipe:

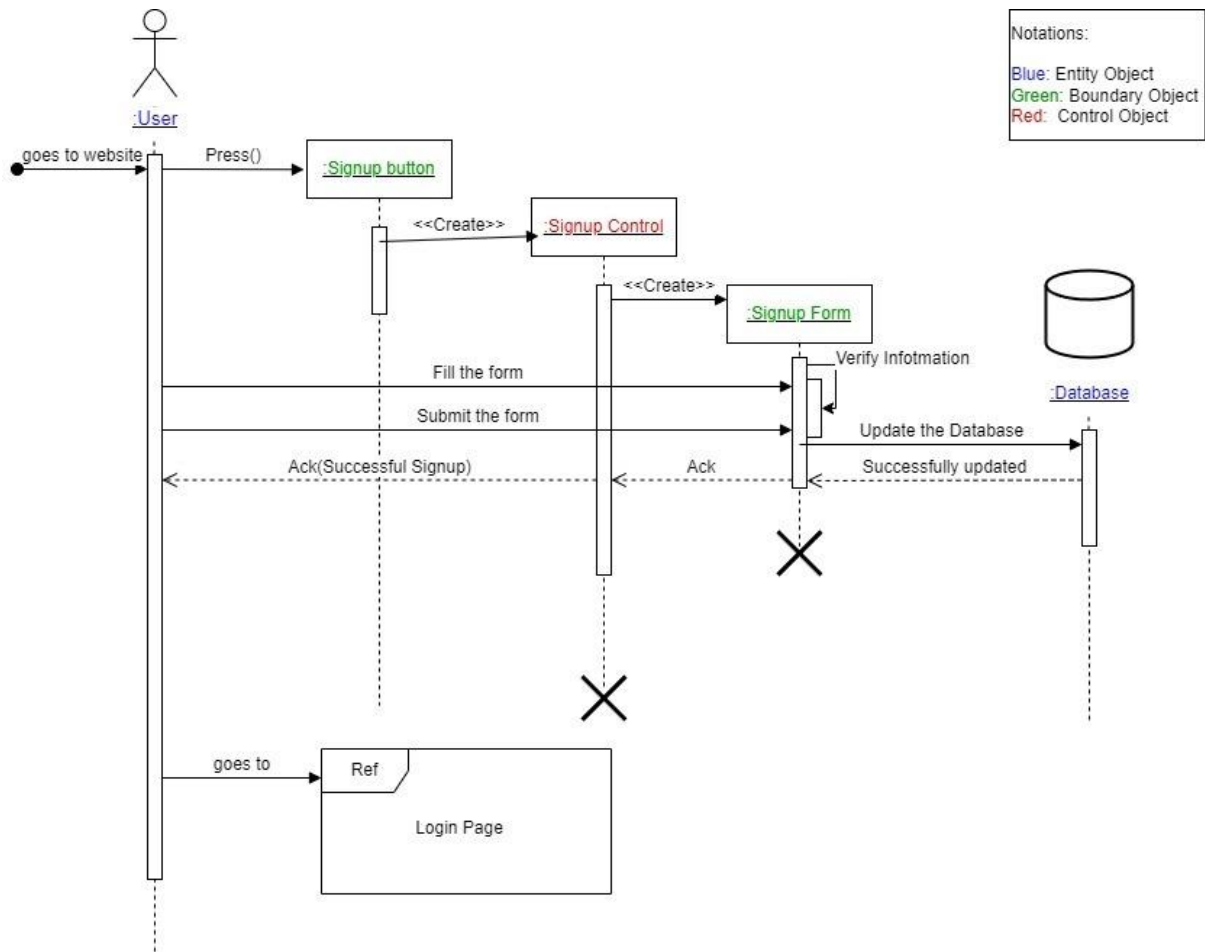


SEQUENCE DIAGRAM

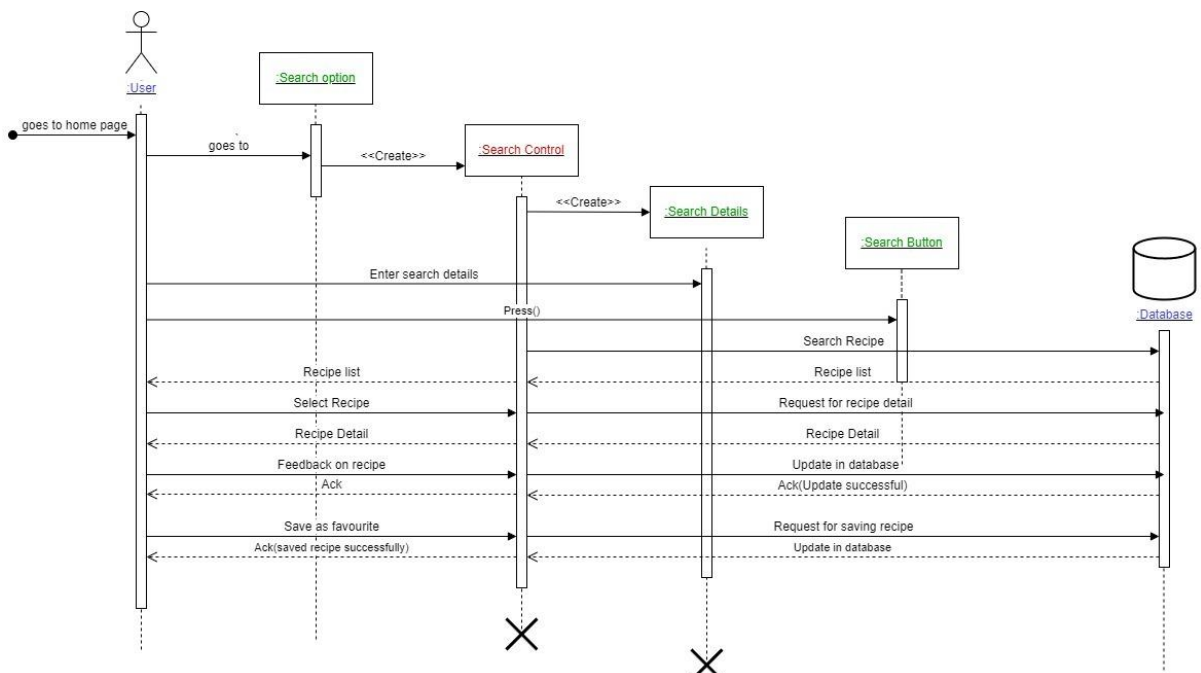
Login



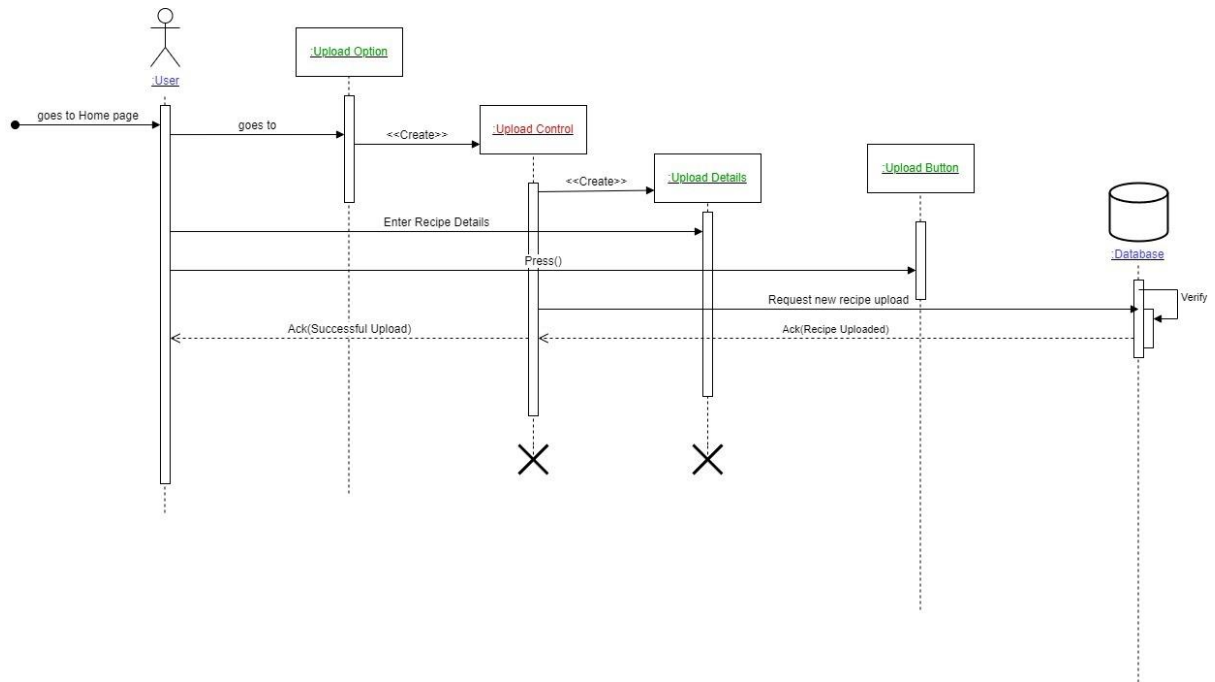
Signup



Search



Upload



CLASS DIAGRAM

