

PANTRY2PLATE

A PROJECT REPORT

Submitted by

**TANUJ TAYAL
(2200290140157)**

**VIKAS KUMAR GUPTA
(2200290140176)**

**Submitted in partial fulfilment of the
Requirements for the Degree of**

MASTER OF COMPUTER APPLICATION

**Under the Supervision of
Dr. Ankit Verma
Assistant Professor**



Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS

KIET Group of Institutions, Ghaziabad Uttar Pradesh-201206

(Mar-2024)

DECLARATION

I hereby declare that the work presented in this report entitled “Pantry2Plate”, was carried out by me. I have not submitted the matter embodied in this report for the award of any other degree or diploma of any other University or Institute. I have given due credit to the original authors/sources for all the words, ideas, diagrams, graphics, computer programs, experiments, results, that are not my original contribution. I have used quotation marks to identify verbatim sentences and given credit to the original authors/sources. I affirm that no portion of my work is plagiarized, and the experiments and results reported in the report are not manipulated. In the event of a complaint of plagiarism and the manipulation of the experiments and results, I shall be fully responsible and answerable.

Name: Tanuj Tayal (2200290140157)
Vikas Kumar Gupta (2200290140176)

(Candidate Signature)

CERTIFICATE

Certified that **Tanuj Tayal (2200290140157)** and **Vikas Kumar Gupta (2200290140176)** have carried out the project work having "Pantry2Plate" for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU), Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself / herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Tanuj Tayal (2200290140157)

Vikas Kumar Gupta (2200290140176)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Dr. Ankit Verma

Assistant Professor

Department of Computer Applications

KIET Group of Institutions, Ghaziabad

ABSTRACT

Imagine a kitchen companion that transforms leftover ingredients into culinary masterpieces. Our recipe recommender does just that, empowering home cooks of all levels to discover new flavours and minimize food waste. Simply input your available ingredients, and our intelligent system scours a vast database to present the perfect recipe, tailored to your pantry and preferences.

But this isn't just a one-way street. Register and unleash your inner chef! Create, edit, and personalize your own recipes, adding secret ingredients and treasured family traditions. Share your culinary creations with the community, inspiring others and expanding your personal recipe library.

Our intuitive interface makes browsing and managing recipes a breeze. Visual recipe cards showcase key ingredients, preparation steps, and user ratings, while text search, voice commands, and even image recognition make adding ingredients to your list a snap.

More than just a recipe finder, we're fostering a collaborative community where food lovers can connect, share their passion, and learn from each other. Whether you're a kitchen novice seeking inspiration or a seasoned chef adding a new dish to your repertoire, join us and unlock a world of endless culinary possibilities.

So, ditch the meal-time rut and embrace the joy of culinary creativity. Let our ingredient-driven recommender be your guide, and turn your pantry into a playground of flavour!

ACKNOWLEDGEMENTS

Success in life is never attained single-handedly. My deepest gratitude goes to my thesis supervisor, **Dr. Ankit Verma** for his guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Arun Kumar Tripathi**, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me with moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Tanuj Tayal

Vikas Kumar Gupta

TABLE OF CONTENTS

Declaration		ii
Certificate		iii
Abstract		iv
Acknowledgement		v
Table of Contents		vi-vii
List of Figures		viii
List of Tables		ix
1	Introduction	1-3
1.1	Overview	1
1.2	Objective	2
1.3	Project Features	2-3
2	Literature Review	4-5
2.1	Introduction	4
2.2	Personalized Recipe Recommendations	4
2.3	User Generated Content	4
2.4	Authentication and Security in Recipe Platform	4
2.5	Technological Innovations in Recipe Discovery	5
2.6	Challenges and Opportunities in Recipe Platforms	5
3	Feasibility Study	6-7
3.1	Technical Feasibility	6
3.2	Economical Feasibility	6
3.3	Operational Feasibility	7
3.4	Scheduling Feasibility	7
3.5	Security and Privacy Feasibility	7
3.6	User Engagement and Feedback	7
4	Requirement Analysis	8-10
4.1	Functional Requirement	8-9

4.2	Non-Functional Requirement	9-10
4.3	Software Requirement	10
4.4	Hardware Requirement	10
5	System Architecture and Design	11-14
5.1	ER-Diagram	11
5.2	Case Study Diagram	12
5.3.1	Data Flow Diagram 0-level	13
5.3.2	Data Flow Diagram 1-level	14
6	Database Design	15-14
6.1	Introduction	15
6.2	Data Modeling	15
6.3	Use Cases	16
6.4	Schema Design	16
6.5	Link Related Data	17-18
6.6	Data Storage and Retrieval	18
7	Form Design	19-24
7.1	Landing Page	19
7.2	List Of Recipes	20
7.3	Login Page	21
7.4	Register Page	22
7.5	Adding Recipe Page	23
7.6	Search Recipe Page	24
8	Testing	25-26
8.1	Testing Case 1 (Login)	25
8.2	Testing Case 2 (Adding Recipes)	25-26
8.3	Testing Case 3 (Logout)	26
9	Bibliography	27

LIST OF FIGURES

Figure No.	Name of Figure	Page No.
5.1	ER Diagram	11
5.2	Case Study Diagram	12
5.3.1	Data Flow Diagram 0-level	13
5.3.2	Data Flow Diagram 1-level	14
6.5.1	Embedded Data	17
6.5.2	References	18
7.1	Landing Page	19
7.2	List Of Recipes	20
7.3	Login Page	21
7.4	Register Page	22
7.5	Adding Recipe Page	23
7.6	Search Recipe Page	24

LIST OF TABLES

Figure No.	Name of Figure	Page No.
4.3	Software Requirement for Pantry2Plate	10
4.4	Hardware Requirement for Pantry2Plate	10

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

In the bustling realm of culinary exploration, the intersection of convenience and creativity has found a new ally in Pantry2Plate. As the pace of modern life quickens and the demand for home-cooked meals rises, the need for an intuitive and efficient recipe search engine becomes paramount. Enter Pantry2Plate, a culinary companion designed to simplify the cooking experience by leveraging the ingredients readily available in your home.

Pantry2Plate is not just another recipe search engine; it's a culinary aide with a clear mission—to empower individuals in their kitchens with an easy-to-use search tool that transforms their existing pantry staples into delightful, wholesome meals. This innovative platform is crafted for those who cherish the joy of home cooking, providing a seamless bridge between the ingredients at hand and the vast world of culinary possibilities.

At the heart of Pantry2Plate lies the My Pantry feature, a user-friendly input system that invites individuals to showcase the contents of their home kitchen. Users simply list the ingredients they have readily available, and with a few clicks, Pantry2Plate's powerful algorithms curate a diverse array of recipes tailored to match those ingredients. This intelligent matching system goes beyond basic searches, breaking down the recipes into categories to suit varied tastes and preferences.

The culinary landscape is diverse, and so is Pantry2Plate's repertoire. Whether you're a seasoned chef or a novice in the kitchen, our platform strives to make the art of cooking accessible to all. No longer will the question of "What's for dinner?" be met with uncertainty; Pantry2Plate ensures that the answer is just a few clicks away.

Join us on a journey where your pantry transforms into a canvas for culinary artistry. Pantry2Plate is not merely a recipe search engine—it's a gateway to a world of flavors, a companion in your culinary escapades, and a testament to the belief that home-cooked meals can be both delightful and effortlessly accessible. Welcome to Pantry2Plate, where your ingredients are the key, and the possibilities are endless.

1.2 OBJECTIVE

Eradicating Food Waste: Dish craft champions the fight against food waste. By understanding your pantry's potential, it crafts recipes that utilize every morsel, saving you money and reducing environmental impact. No more wilting vegetables or forgotten leftovers, just a celebration of culinary resourcefulness.

Unleashing Culinary Creativity: Step outside your comfort zone and embark on a global culinary adventure, all fuelled by what you already have. Dish craft's intelligent algorithms unlock a universe of diverse recipes, from Thai stir-fries to Italian pastas, all waiting to be explored with the ingredients at your fingertips. Break free from the shackles of predictable meals and embrace the thrill of culinary discovery.

Personalizing Your Kitchen Experience: Dish craft is more than just a recipe finder; it's a digital cookbook tailored to your palate. Register and build your own culinary haven, filled with cherished family recipes, personal modifications, and even your own innovative creations. Experiment with flavours, document your culinary journey, and create a personalized library that reflects your unique culinary identity.

Fostering a Vibrant Community: Connect with fellow foodies, share your culinary triumphs and disasters, and discover hidden gems through Dish craft's intuitive social features. Leave reviews and comments on recipes, sparking conversations and enriching the user experience. Build a community where food unites, inspires, and fosters a culture of shared culinary passion.

1.3 PROJECT FEATURE

Ingredient-driven Search: Tell Kitchen Maestro what you have, be it leftover vegetables, a can of beans, or a lonely chicken breast, and watch it whip up a symphony of flavors, prioritizing recipes that best utilize your existing ingredients. Minimize waste, maximize flavor, and discover hidden potential in your pantry.

Smart Recipe Recommendations: Forget wading through irrelevant options. Kitchen Maestro's intelligent algorithms consider your ingredients, dietary restrictions, preferences, and even cooking skills to present a curated selection of perfectly matched recipes. No more recipe ruts, just endless possibilities waiting to be explored.

Detailed Recipe Cards: Get a clear picture of each dish, with step-by-step instructions, cooking times, nutritional information, and user ratings all readily available. Stunning visuals, intuitive menus, and interactive features make following recipes a breeze, even for novice cooks.

Personalization Haven: Register and build your own culinary sanctuary. Save your favorite recipes, add personal notes and modifications, and even create your own dishes to share with the world. Kitchen Maestro becomes your digital cookbook, evolving with your culinary journey and reflecting your unique palate.

Social Features: Connect with fellow food enthusiasts, share your culinary creations, discover hidden gems through ratings and reviews, and engage in lively discussions about all thing's food. The Kitchen Maestro community fosters a vibrant atmosphere where inspiration and knowledge flow freely, enriching the user experience for everyone.

Beyond the Core: Imagine seamless integrations with grocery delivery services, where the app can suggest recipes based on both your pantry and available deliveries. Dietary filters cater to specific needs and preferences, making healthy cooking effortless. Voice recognition capabilities further streamline the experience, allowing users to simply speak their ingredients and receive instant recipe recommendations. Gamification elements, like points for trying new recipes or sharing creations, keep users engaged and motivated.

CHAPTER 2

LITERATURE REVIEW

2.1 PANTRY2PLATE

Existing recipe search engines have paved the way for P2P. Platforms like Yummly and Allrecipes emphasize ingredient-based searches, but P2P aims to distinguish itself by prioritizing simplicity and accuracy in matching ingredients with recipes. Research suggests that user-friendly interfaces significantly impact user engagement in such platforms.

2.2 PERSONALIZED RECIPE RECOMMENDATIONS

Studies on personalized recommendation systems reveal the importance of algorithms in understanding user preferences. P2P's commitment to delivering the best recipes tailored to users' ingredients and dietary preferences aligns with the growing trend of personalization in online platforms, enhancing user satisfaction.

2.3 USER-GENERATED CONTENT IN CULINARY PLATFORMS

Examining successful platforms like Food Network's community and Epicurious, it becomes evident that user-generated content enhances the overall experience. By allowing users to contribute recipes, P2P taps into the collaborative spirit of cooking communities, fostering a sense of connection among users with shared culinary interests.

2.4 AUTHENTICATION AND SECURITY IN RECIPE PLATFORMS

Platforms that encourage user contributions necessitate robust authentication and security measures. Insights from research on secure user authentication and data protection underline the importance of P2P's decision to implement separate login credentials for recipe contributors, ensuring a secure and trustworthy environment.

2.5 TECHNOLOGICAL INNOVATIONS IN RECIPE DISCOVERY

Technological advancements, such as dynamic ingredient lists and real-time updates, have transformed the landscape of recipe discovery. P2P's utilization of a dynamic list for ingredient selection aligns with research advocating for intuitive and responsive interfaces in cooking-related applications.

2.6 CHALLENGES AND OPPORTUNITIES IN RECIPE PLATFORMS

By reviewing challenges faced by existing platforms, P2P can proactively address potential issues. The literature emphasizes the importance of responsiveness, scalability, and effective filtering mechanisms in recipe search engines, presenting opportunities for P2P to excel.

CHEPTEr 3

FEASIBILITY STUDY

After doing the project, study and analysing all the existing or required functionalities of the system, the next task is to do the feasibility study for the project. All projects are feasible-given unlimited resources and in finite time. Feasibility study includes consideration of all the possible ways to provide a solution to the given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can be easily done based on the future upcoming requirements. There are three parts in feasibility study

- a) Technical Feasibility
- b) Economic Feasibility
- c) Operational Feasibility
- d) Scheduling Feasibility
- e) Security and Privacy Feasibility
- f) User Engagement and Feedback

3.1 TECHNICAL FEASIBILITY

The technical feasibility of P2P is rooted in its ability to implement and maintain a sophisticated recipe search engine. The utilization of dynamic lists, real-time updates, and user authentication systems suggests a sound technical foundation. The study evaluates the availability of required technologies, the expertise of the development team, and the feasibility of incorporating innovative features.

3.2 ECONOMIC FEASIBILITY

Evaluating the economic viability of P2P involves assessing the costs associated with development, maintenance, and marketing against the anticipated benefits. The feasibility study considers budgetary constraints, potential revenue streams (such as

advertisements or premium features), and a projected return on investment. This analysis ensures that P2P is economically sustainable and aligns with market expectations.

3.3 OPERATIONAL FEASIBILITY

The operational feasibility of P2P hinges on its seamless integration into users' daily cooking routines. User-friendliness, efficient recipe categorization, and a responsive interface are critical factors. The study explores how well P2P aligns with user behaviours and whether it can be easily adopted and embraced by the target audience. Operational feasibility also considers the scalability of the platform to accommodate a growing user base.

3.4 SCHEDULING FEASIBILITY

Meeting deadlines and launching the platform within a reasonable timeframe is a key consideration. The feasibility study outlines the project timeline, identifies potential bottlenecks, and assesses the realistic attainment of milestones. This ensures that the development and deployment of P2P align with market demands and competitor timelines.

3.5 SECURITY AND PRIVACY FEASIBILITY

Given the nature of user-contributed content and personal data involved in the login process, ensuring robust security and privacy measures is crucial. The feasibility study delves into the implementation of secure authentication practices, data encryption, and measures to protect user information. This address concerns related to user trust and compliance with data protection regulations.

3.6 USER ENGAGEMENT AND FEEDBACK

Anticipating user engagement and collecting feedback during the development and beta testing phases is integral. The feasibility study explores strategies for soliciting user input, addressing potential challenges, and adapting the platform based on user responses. This iterative approach ensures that P2P remains responsive to user needs and preferences.

In conclusion, the feasibility study for P2P is a holistic evaluation encompassing technical, economic, operational, scheduling, security, and user engagement considerations. By thoroughly assessing these factors, the study provides valuable insights that guide the development process, ultimately increasing the likelihood of P2P's success as a user-centric.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Functional requirements specify what the system or software should do or the actions it should perform. They describe the intended functionality, features, and capabilities of the system. These requirements outline the system's behavior, inputs, outputs, and interactions with users or other systems. Functional requirements are typically specific, measurable, and verifiable. Examples include user authentication, data input validation, report generation, and system integration.

- **User Authentication**

Users must be able to create accounts with unique login credentials.

Secure authentication mechanisms should be implemented for user data protection.

- **Ingredient Input**

Users should have the option to input ingredients they have at home.

The system should support both dynamic list selection and manual entry.

- **Real-Time Recipe Updates**

Recipes should update instantly as users add or remove ingredients.

The system must provide a responsive and seamless user experience.

- **Recipe Categorization**

Recipes should be categorized into user-friendly sections for easy navigation.

Categories may include cuisine types, dietary restrictions, or meal types.

- **Personalized Recommendations**

The platform must analyse user preferences and dietary restrictions to offer personalized recipe recommendations.

Recommendation algorithms should be able to adapt to user feedback and evolving preferences.

- **Recipe Contribution**

Users interested in cooking can contribute recipes.

The contribution process should be straightforward, allowing users to share their culinary creations with the community.

- **Visibility Controls**

Contributors should be able to manage the visibility of their recipes.

Privacy settings should allow control over who can view and access the shared recipes.

- **Search Functionality**

The search tool should efficiently match input ingredients with available recipes.

Search results should be displayed in a user-friendly and organized manner.

4.2 NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements, also known as quality attributes or constraints, define the characteristics and constraints of the system beyond its functionality. These requirements describe how the system should perform, rather than what it should do. Non-functional requirements are often related to performance, reliability, security, usability, and other aspects that contribute to the overall system quality. Examples include response time, system availability, data encryption, user interface design, and regulatory compliance.

- **Performance**

The platform must provide quick response times, especially during real-time updates.

Load testing should be conducted to ensure optimal performance under varying user loads.

- **Security**

Robust security measures must be in place to protect user data and privacy.

Encryption protocols should be implemented for secure data transmission.

- **Scalability**

The system should be designed to handle a growing number of users and recipes.

Scalability testing should be performed to assess the platform's ability to expand.

- **Usability**

The user interface should be intuitive, accommodating users with varying levels of technical expertise.

User experience testing should be conducted to ensure ease of use.

- **Reliability**

The platform must be reliable, minimizing downtime and service interruptions.

Implementing backup and recovery mechanisms is essential for data integrity.

- **Regulatory Compliance**

The system must comply with relevant data protection and privacy regulations.

Transparency in terms of service and privacy policies is crucial for user trust.

- **Feedback Mechanism**

A user feedback mechanism should be implemented for continuous improvement.

Analytics tools must be in place to monitor user behaviour and preferences.

In conclusion, the functional requirements outline the specific features and functionalities of the P2P recipe search engine, while the non-functional requirements address aspects such as performance, security, scalability, usability, reliability, regulatory compliance, and user feedback mechanisms. Together, these requirements provide a comprehensive guide for the development and deployment of a successful and user-centric web page.

4.3 SOFTWARE REQUIREMENT

S. NO.	DESCRIPTION	TYPE
1	Operating System	Windows, MacOS
2	Language	HTML5, CSS3, JavaScript, Bootstrap, ExpressJS, NodeJS
3	IDE	VS Code
4	Database	MongoDB

Table 4.3 Software Requirement for Pantry2Plate

4.4 HARDWARE REQUIREMENTS

S. NO.	DESCRIPTION	TYPE
1	Hardware	I3 Processor
2	Clock Speed	3.0GHz
3	RAM	8GB
4	SSD	512GB

Table 4.4 Hardware Requirement for Pantry2Plate

CHEPTER 5

SYSTEM ARCHITECTURE AND DESIGN

5.1 ENTITY-RELATIONSHIP DIAGRAM

Entity-Relationship model stands for an ER model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system. It develops a conceptual design for the database. It also develops a very simple and easy to design view of data. In ER modelling, the database structure is portrayed as a diagram called an entity relationship diagram.

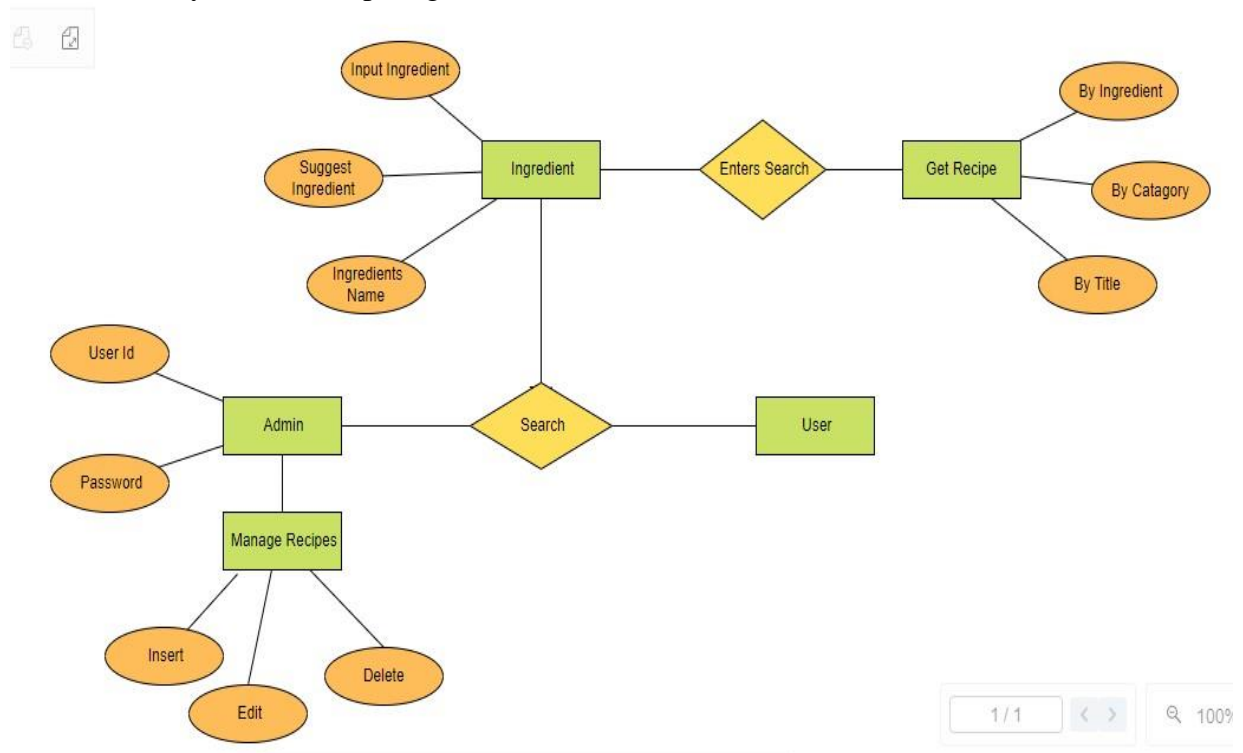


Fig 5.1 Entity-Relationship Diagram

5.2 CASE STUDY DIAGRAM

A case study in software engineering is an empirical research method that uses multiple sources of evidence to investigate a contemporary software engineering phenomenon. It's an in-depth examination of a specific real-world situation, project, problem, or success story within the field.

A case study can be a document, such as a video, white paper, or blog post, that outlines how a customer used a product to overcome a problem. It's real-world proof that a product works and gets results.

Case studies are commonly used in social, educational, clinical, and business research. In computer programming courses, case studies involve learners in activities of expert programmers such as identifying decisions, justifying choices among alternatives, and evaluating the consequences of these choices.

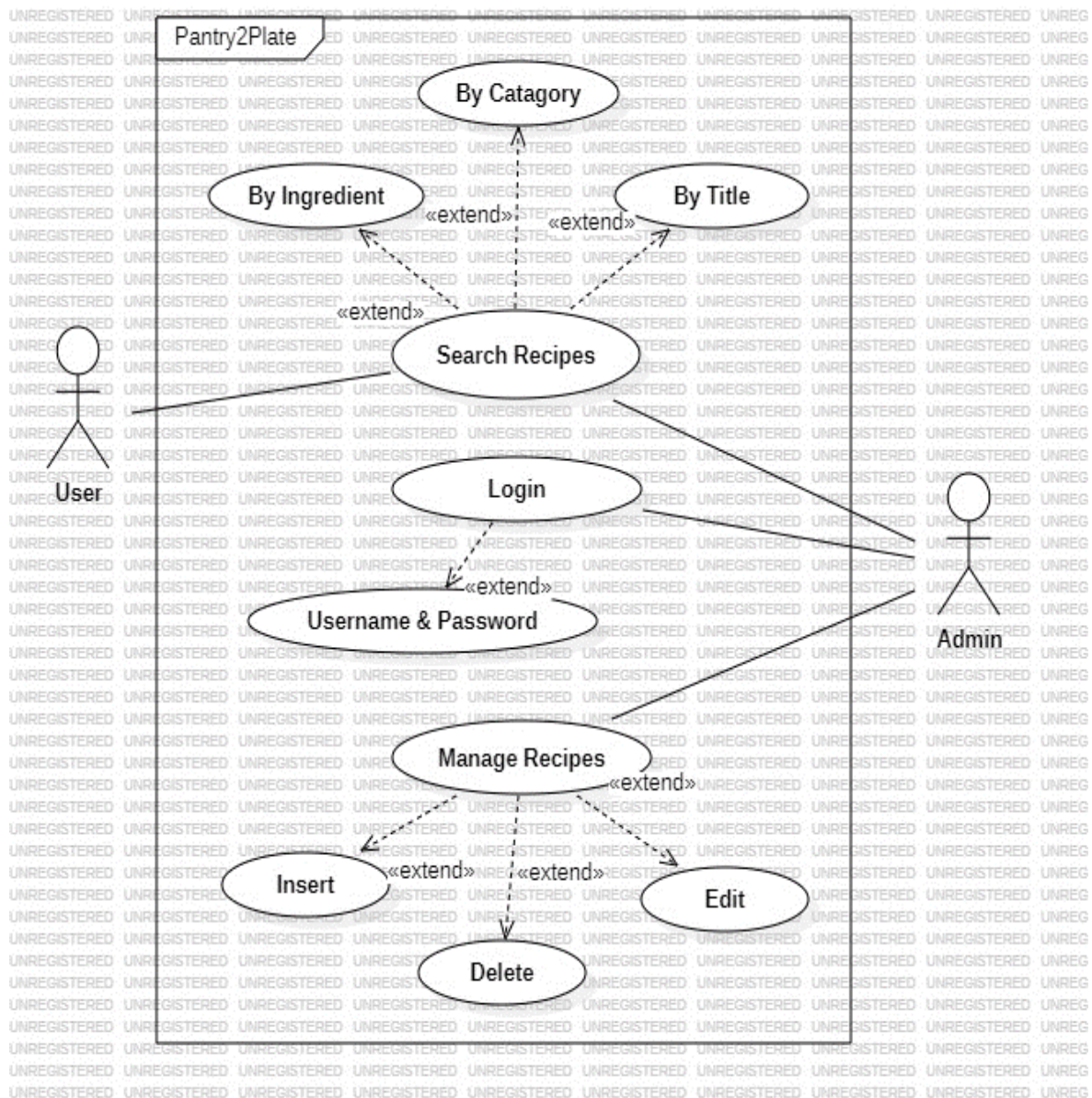


Fig 5.2 Case Study Diagram

5.3 DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of how data moves through a system or process. DFDs are a key tool in software development and business analysis.

DFDs have no control flow, meaning they don't have decision rules or loops. Instead, specific operations based on the data can be represented by a flowchart.

DFDs have four main components: Processes, Data flows, Data stores, External entities.

DFDs have two levels:

0-level DFD

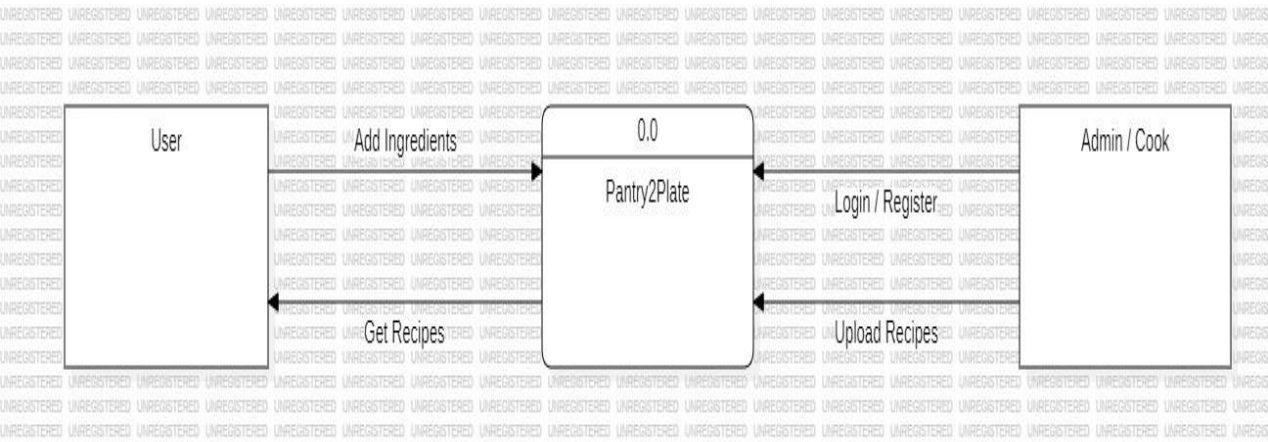


Fig 5.3.1 Data Flow Diagram 0-level

1-level DFD

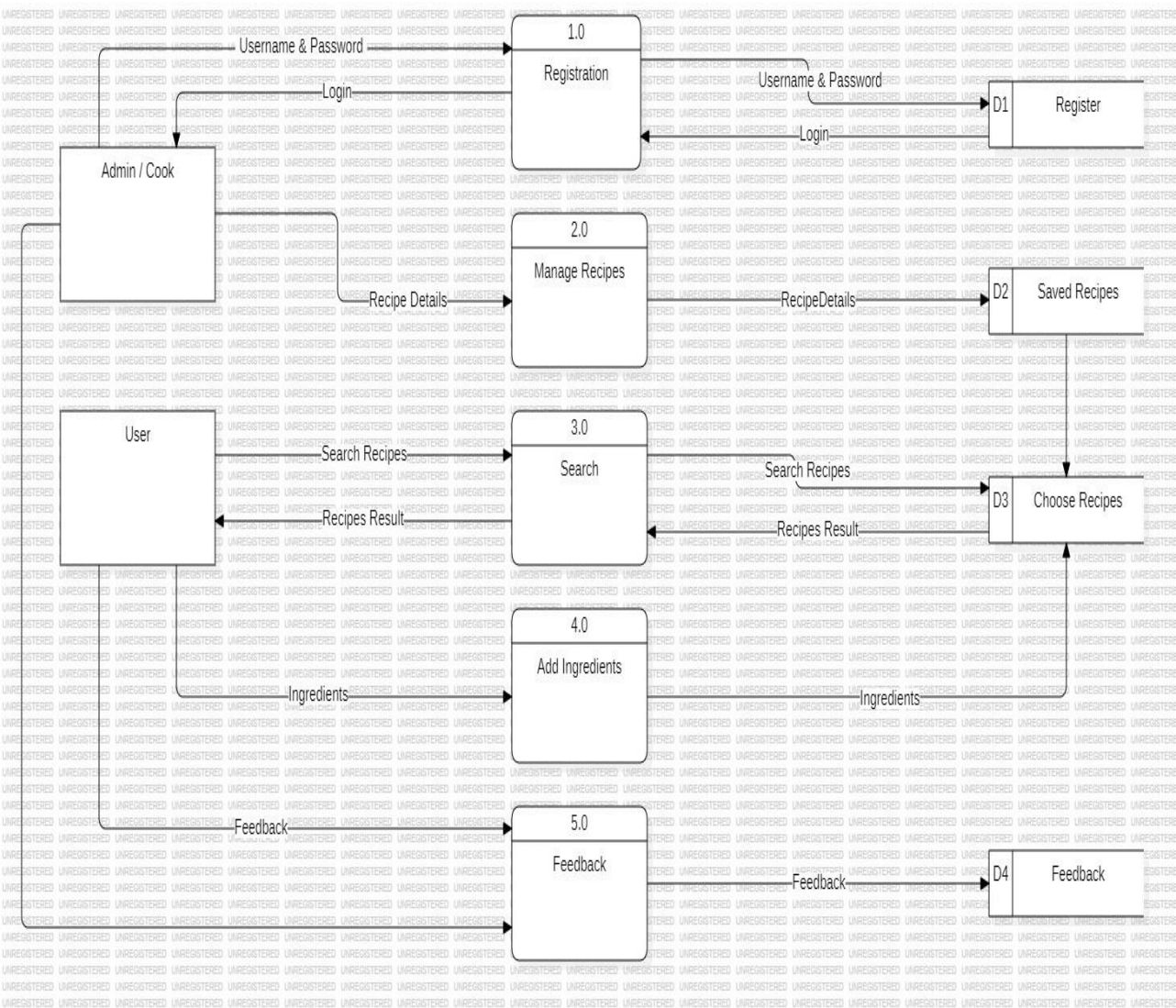


Fig 5.3.2 Data Fow Diagram 1-level

CHEPTER 6

DATABASE DESIGN

6.1 INTRODUCTION

MongoDB (link resides outside IBM) is an open source, nonrelational database management system (DBMS) that uses flexible documents instead of tables and rows to process and store various forms of data. As a NoSQL database solution, MongoDB does not require a relational database management system (RDBMS), so it provides an elastic data storage model that enables users to store and query multivariate data types with ease. This not only simplifies database management for developers but also creates a highly scalable environment for cross-platform applications and services.

MongoDB documents or collections of documents are the basic units of data. Formatted as Binary JSON (Java Script Object Notation), these documents can store various types of data and be distributed across multiple systems. Since MongoDB employs a dynamic schema design, users have unparalleled flexibility when creating data records, querying document collections through MongoDB aggregation and analysing large amounts of information.

6.2 DATA MODELING

Data modelling refers to the organization of data within a database and the links between related entities. Data in MongoDB has a **flexible schema model**, which means:

- Documents within a single collection are not required to have the same set of fields.
- A field's data type can differ between documents within a collection.

Generally, documents in a collection share a similar structure. To ensure consistency in your data model, you can create schema validation rules.

6.3 USE CASES

The flexible data model lets you organize your data to match your application's needs. MongoDB is a document database, meaning you can embed related data in object and array fields.

A flexible schema is useful in the following scenarios:

- Your company tracks which department each employee works in. You can embed department information inside of the `employee` collection to return relevant information in a single query.
- Your e-commerce application shows the five most recent reviews when displaying a product. You can store the recent reviews in the same collection as the product data, and store older reviews in a separate collection because the older reviews are not accessed as frequently.
- Your clothing store needs to create a single-page application for a product catalog. Different products have different attributes, and therefore use different document fields. However, you can store all of the products in the same collection.

6.4 SCHEMA DESIGN

When you design a schema for a document database like MongoDB, there are a couple of important differences from relational databases to consider.

Relational Database Behaviour

You must determine a table's schema before you insert data.

You often need to join data from several different tables to return the data needed by your application.

Document Database Behaviour

Your schema can change over time as the needs of your application change.

The flexible data model lets you store data to match the way your application returns data, and avoid joins. Avoiding joins across multiple collections improves performance and reduces your deployment's workload.

6.5 LINK RELATED DATA

When you design your data model in MongoDB, consider the structure of your documents and the ways your application uses data from related entities.

To link related data, you can either:

- Embed related data within a single document.
- Store related data in a separate collection and access it with a reference.

6.5.1 Embedded Data

Embedded documents store related data in a single document structure. A document can contain arrays and sub-documents with related data. These **denormalized** data models allow applications to retrieve related data in a single database operation.



Fig 6.5.1 Embedded Data

For many use cases in MongoDB, the denormalized data model is optimal.

To learn about the strengths and weaknesses of embedding documents, see Embedded Data Models.

6.5.2 References

References store relationships between data by including links, called **references**, from one document to another. For example, a `customerId` field in an `orders` collection indicates a reference to a document in a `customer's` collection.

Applications can resolve these references to access the related data. Broadly, these are normalized data models.

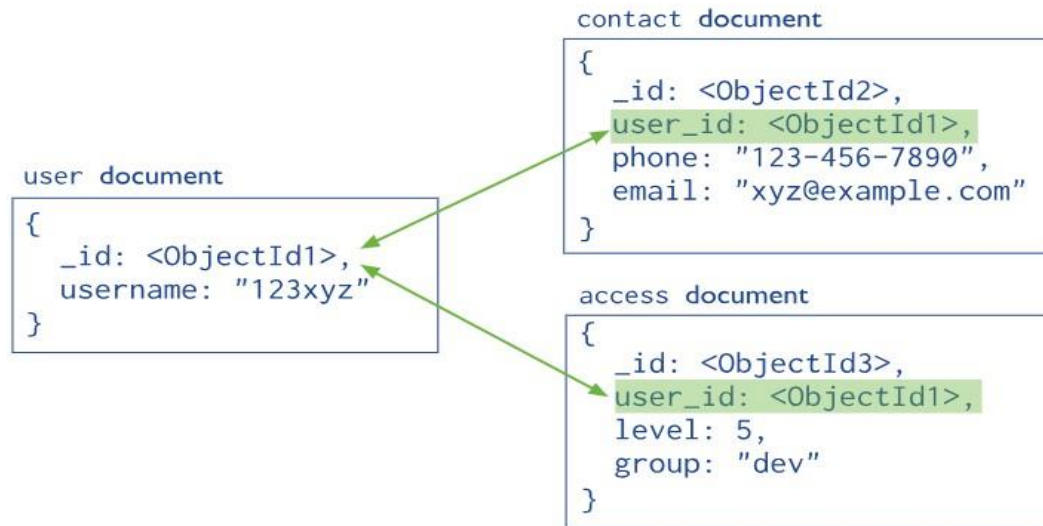


Fig 6.5.2 References

6.6 DATA STORAGE AND RETRIEVAL

The storage engine is the component of the database that is responsible for managing how data is stored, both in memory and on disk. MongoDB supports multiple storage engines, as different engines perform better for specific workloads. Choosing the appropriate storage engine for your use case can significantly impact the performance of your applications.

You can perform find operations to retrieve data from your MongoDB database. You can perform a find operation to match documents on a set of criteria by calling the `find()` or `findOne()` method.

CHEPTEK 7

FORM DESIGN

7.1 LANDING PAGE

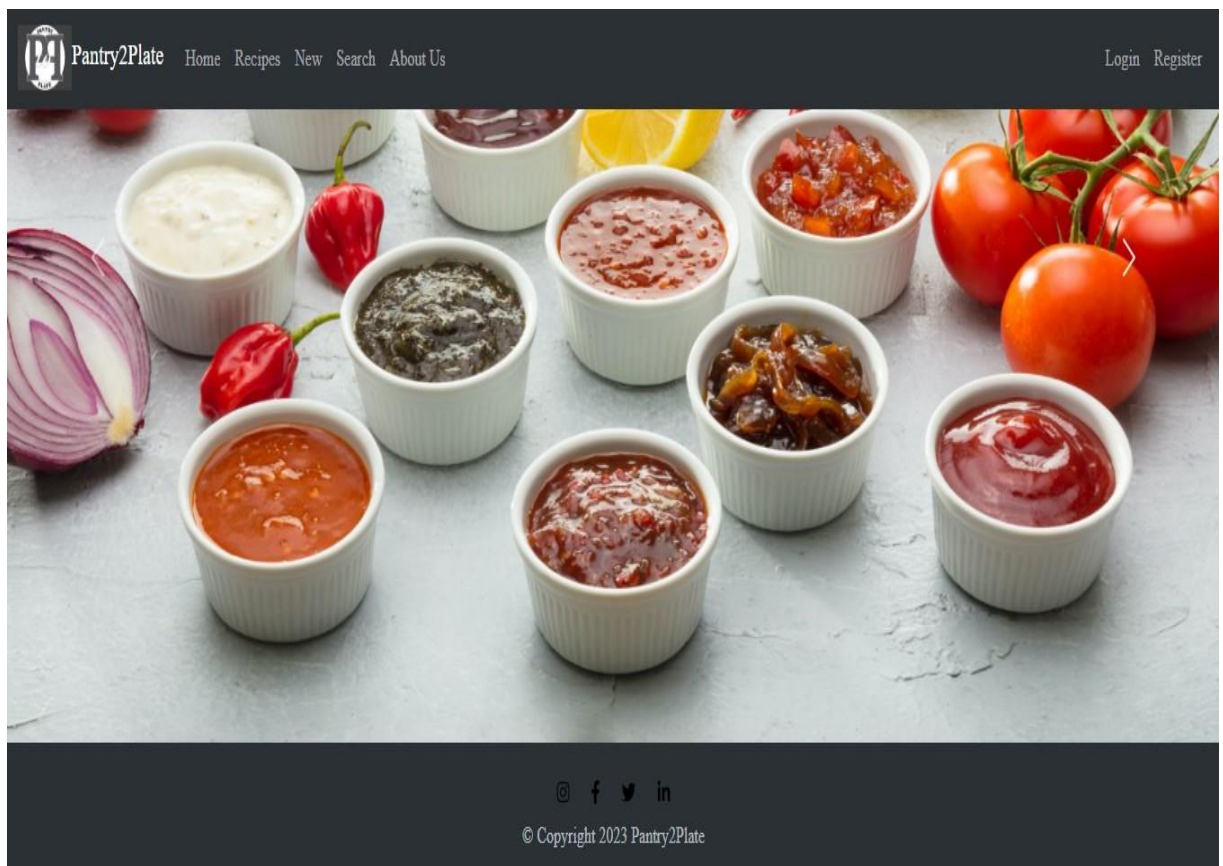




Fig 7.1 Landing Page

7.2 LIST OF RECIPES

Pantry2Plate [Home](#) [Recipes](#) [New](#) [Search](#) [About Us](#)

[Login](#) [Register](#)




Beetroot Halwa Recipe

35

This beetroot halwa is a winter delight. If you want, you can easily make this Indian dessert at home this Diwali.

[View Beetroot Halwa Recipe](#)




Green Moong Cutlet Recipe

30

This delicious cutlet is packed with the goodness of green moong (mung beans), Onions, potato and spices. Healthy and flavoured, this high-protein moong cutlet can be enjoyed even while dieting. You can choose to pan-fry, air-fry or bake it, according to your preference.

[View Green Moong Cutlet Recipe](#)



No-Oil Vegetable Pulao Recipe

25

Pulao is such a delicious recipe that we can make in few minutes. But today we are bring an amazing recipe of pulao which can be made without Oil and it can be good option for those people who are weight watchers.

[View No-Oil Vegetable Pulao Recipe](#)

Fig 7.2 List of Recipes

7.3 LOGIN PAGE

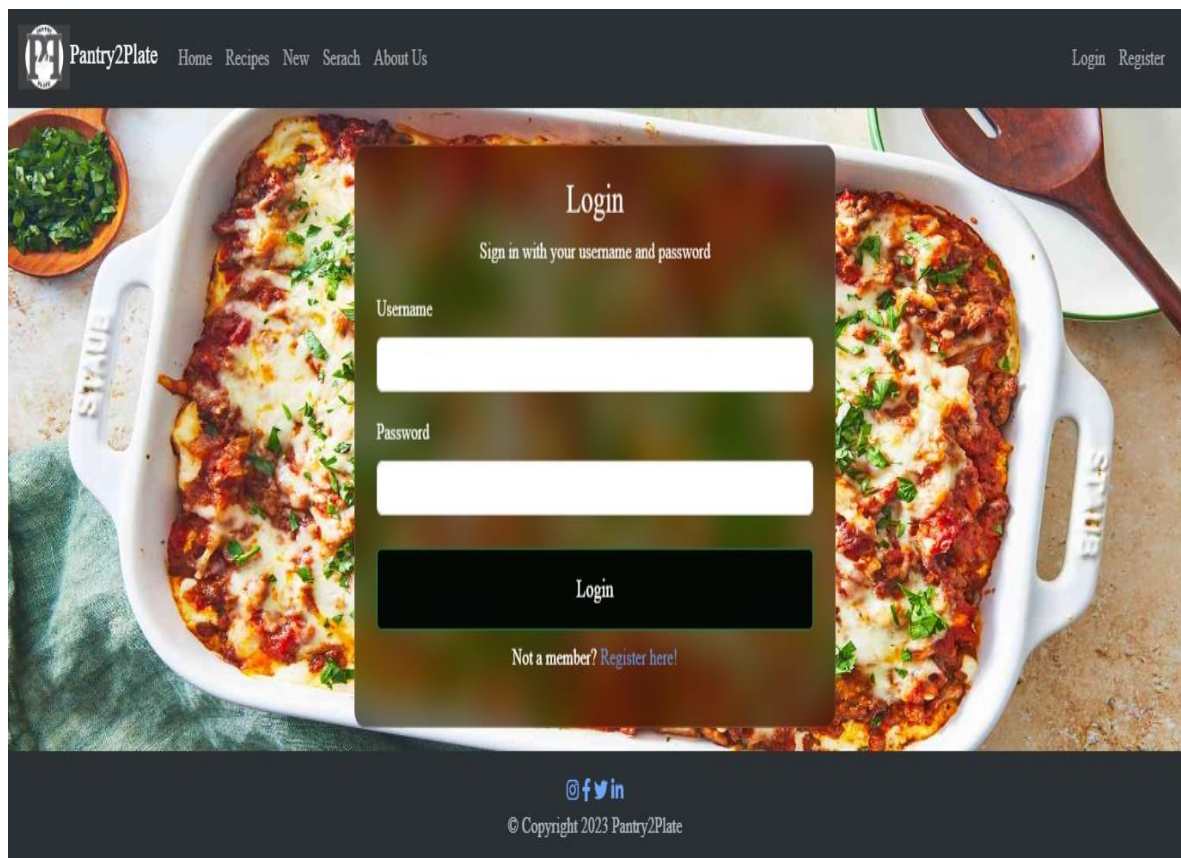


Fig 7.3 Login Page

7.4 REGISTER PAGE

Pantry2Plate Home Recipes New Search About Us Login Register

Register

Username

E-mail

Password

Register

Have an Account ? [Login Here](#)

@ f t in

© Copyright 2023 Pantry2Plate

Fig 7.4 Register Page

7.5 ADDING RECIPE PAGE

Pantry2Plate Home Recipes New Search About Us Logout

New Recipe

Name

Time

Image URL

Method

Ingredient

Description

[All Recipes](#)

[Add Recipe](#)

Fig 7.5 Adding Recipe Page

7.6 SEARCH RECIPE PAGE

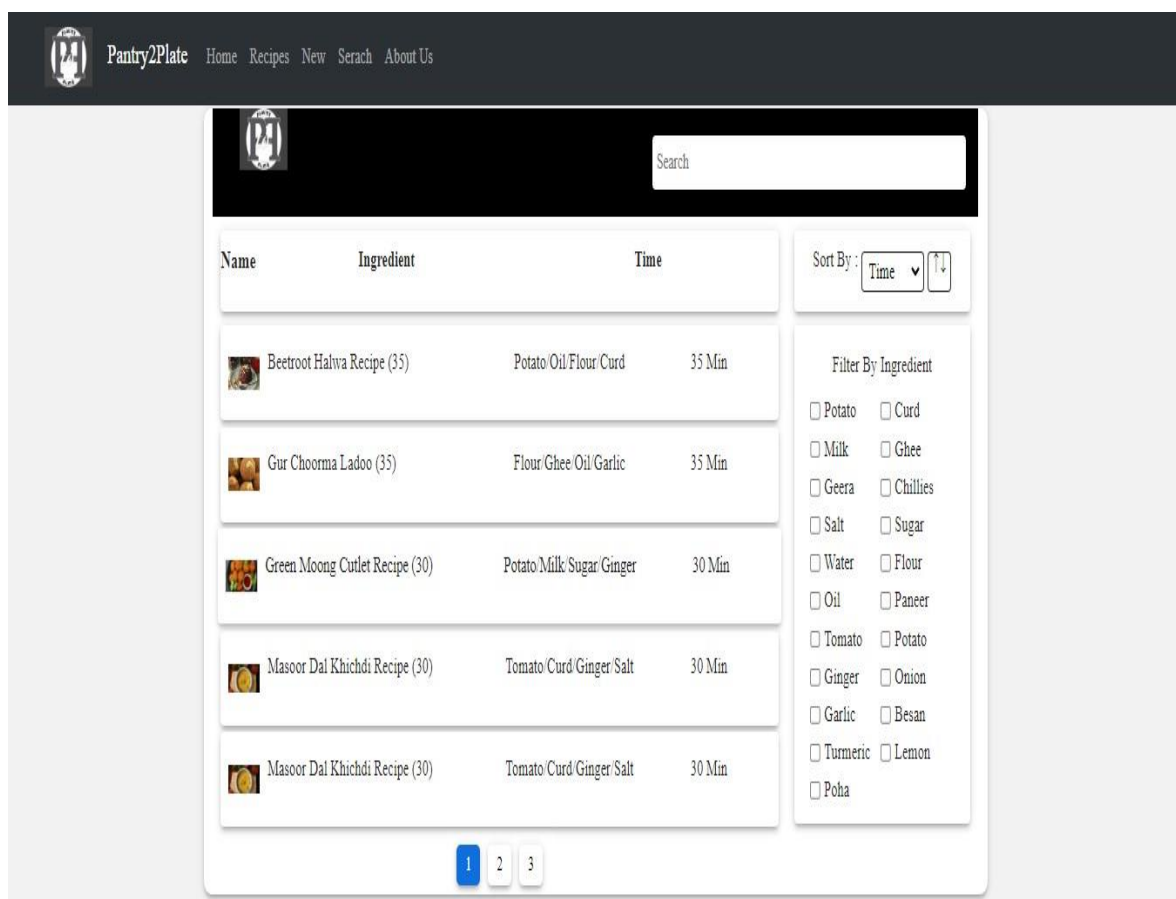


Fig 7.6 Search Recipe Page

CHEPTEr 8

TESTING

8.1 TESTING CASE 1 (Login)

8.1.1 Functional Test Cases-

- Verify if a user will be able to login with a valid username and valid password.
- Verify if a user cannot login with a valid username and an invalid password.
- Verify the login page for both, when the field is blank and Submit button is clicked.
- Verify the messages for invalid login.

8.1.2 Non-Functional Security Test Cases-

- Verify the time out functionality of the login session.
- Verify the login page by pressing 'Back button' of the browser. It should not allow you to enter the system once you log out.
- Verify if a user should not be allowed to log in with different credentials from the same browser at the same time.

8.2 TESTING CASE 2 (Adding Recipes)

8.2.1 Functional Test Cases-

- Verify that the all required fields are filled.
- Verify that the ingredients are properly field with commas.
- Verify that the adding recipe will update to the list.
- Verify that the adding recipe will be available for the only registered user or not.
- Verify that the only registered user can edit or delete the save recipe.

8.2.2 Non-Functional Security Test Cases-

- Verify that the all fields are visible to the user.
- Verify that the “nav bar” should be visible for direct reach to the “All recipes” tab for show updated recipe.

8.3 TESTING CASE 3 (Logout)

8.3.1 Functional Test Cases-

- Verify After successful login in Pantry2Plate, click on the profile icon to check logout button is visible or not.
- Verify by Clicking on the sign-out button without an internet connection and reconnecting to the internet to check if it's properly logout or not.
- Verify by clicking on the logout button, after successful logout on the login screen press the back button.
- Verify, login into more than two browser or mobiles and log out from anyone them and check all other account is properly working or all get logout.
- Verify after logout tries to re-login with the same or different account it's allowing or not.

8.3.2 Non-Functional Security Test Cases-

- Verify the logs for the login and logout sessions.
- Verify if the logs contain multiple IPs for a single ID at the same time.
- Verify if the logs contain a denial-of-service attack for the login or logout.
- Verify if the log has suspicious activity.

CHEPTER 9

BIBLIOGRAPHY

- Node.js v20.9.0 and express 4.18.2 – Modern Cross-Platform Development: Build applications with express JS, node JS web development framework, MongoDB 7.0.2 and using VS Code 1.85.1.
- C. J. Date, A. Kannan, and S. Swaminathan, An Introduction to Database Systems, Pearson Education, Eighth Edition, 2009.
- Abraham Silber Schatz, Henry F. Korte and S. Sudarshan, Database System Concepts, McGraw-Hill Education (Asia), Fifth Edition, 2006.
- Pressman, Somerville, Software Engineering: Design, Implementation, and Management.

APPLICATIONS

- www.google.com
- www.stackoverflow.com
- www.w3school.com
- www.udemy.com
- www.gitHub.com
- www.youtube.com
- <https://unsplash.com/>
- <https://hoppscotch.io/>