

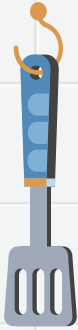


RECIPE ROULETTE

Carmen Samedi, Maria Lopez & Alessandra Bonilla
Group 101-2



TABLE OF CONTENTS



1

PROJECT DOMAIN

- Background
- Functional requirements
- Non-functional requirements

2

PROJECT SOLUTION

- Use case diagram
- Architecture diagram
- Class diagram
- Activity & Sequence diagram

3

DEVELOPMENT

4

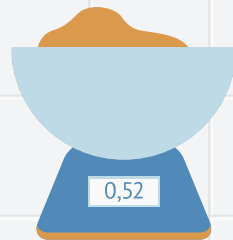
FUTURE ENHANCEMENTS & CONCLUSIONS

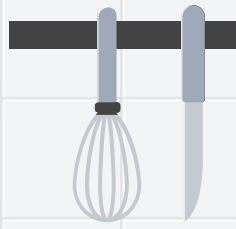
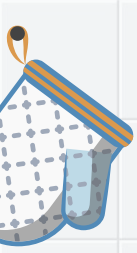


1



PROBLEM DOMAIN



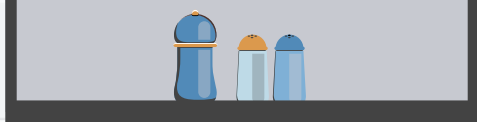


BACKGROUND

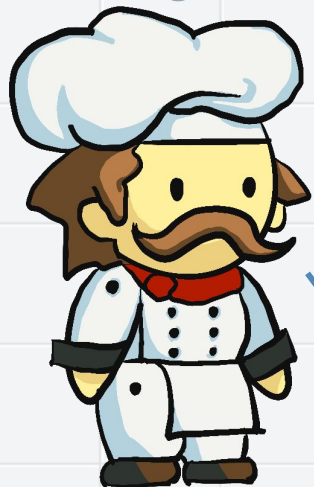


- **PRODUCT SCOPE:**
 - Turn ingredients into recipes
 - Reduce waste, enhance culinary skills
- **PURPOSE:** Make cooking accessible and enjoyable for all ages.
- **TARGET MARKET:** anyone old enough to cook (12-80)

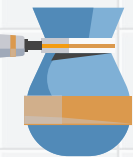




?



HOW DOES IT WORK?





REQUIREMENTS

FUNCTIONAL REQUIREMENTS

UFR-1: User registration

UFR-2: User Login/Authentication

UFR-3: Delete Profile

UFR-4: Search Recipes by Ingredients

UFR-5: Search Recipes by Dish Name

UFR-6: Filter Recipes

UFR-7: Recipe Management

UFR-8: Gain Points

UFR-9: Check Reward Status

UFR-10: Local Database Management

UFR-11: Search In Local Database

MFR-12: Management of Software

MFR-13: User Verification Management

CFR-14: Approve recipe

CFR-15: Send Notification

CFR-16: Delete recipe

CFR-17: Revise Content



UFR-1

User registration

Version: v2.1

Description:

Users access registration, enter personal info including name, email, phone, address, gender, and age. Password creation and confirmation complete registration. First login verifies info for account creation

Comments:

If the person selects the role wrongly they can go back to select a different role.

- The password should follow the following requirements: 8 to 10 characters
- A combination of uppercase letters, lowercase letters, numbers, and symbols.

If the information is invalid the user will return to the registration page where they have to enter their personal information to correct any errors.

Relationships: UFR-2, UFR-3, MFR-12, CFR-15, CFR-14



REQUIREMENTS

NON FUNCTIONAL REQUIREMENTS

NFR-1: Data Encryption



NFR-2: Fast Response Time

NFR-3: Easy Code Maintainability

NFR-4: Large User Load Handling

NFR-5: Intuitive Interface

NFR-6: Regular Backups

NFR-7: Constant Availability

NFR-8: Cross Platform Compatibility

NFR-9: Performance Optimization

NF-1

Data encryption

Version: v2.1

Type /Subtype: Security

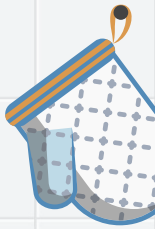
Description:

All sensitive user data, including passwords, personal information, and recipe details, should be encrypted both during transit and at rest to prevent unauthorized access or data breaches.

Comments:

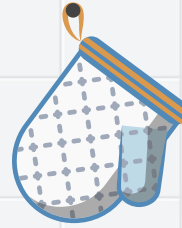
Utilize industry-standard encryption algorithms (e.g., AES) and protocols (e.g., TLS) to ensure the confidentiality and integrity of user data. Implement secure storage mechanisms and key management practices to protect encryption keys from unauthorized access. Regularly audit and update encryption protocols to address emerging security threats and vulnerabilities.

Relationships: UFR-1, UFR-2

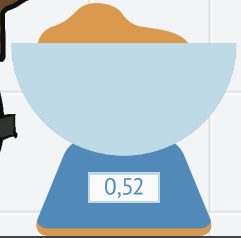
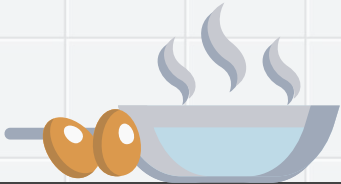




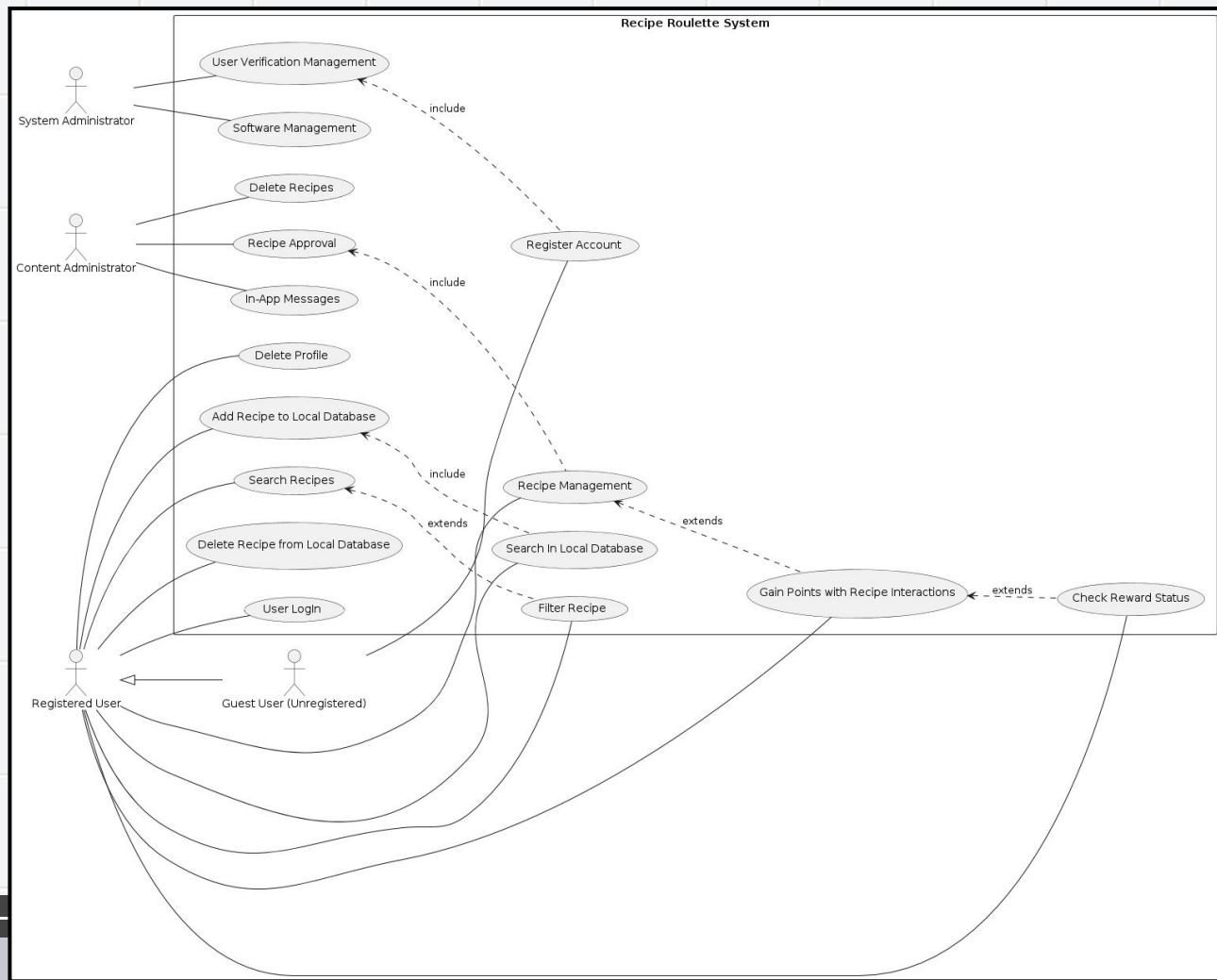
2



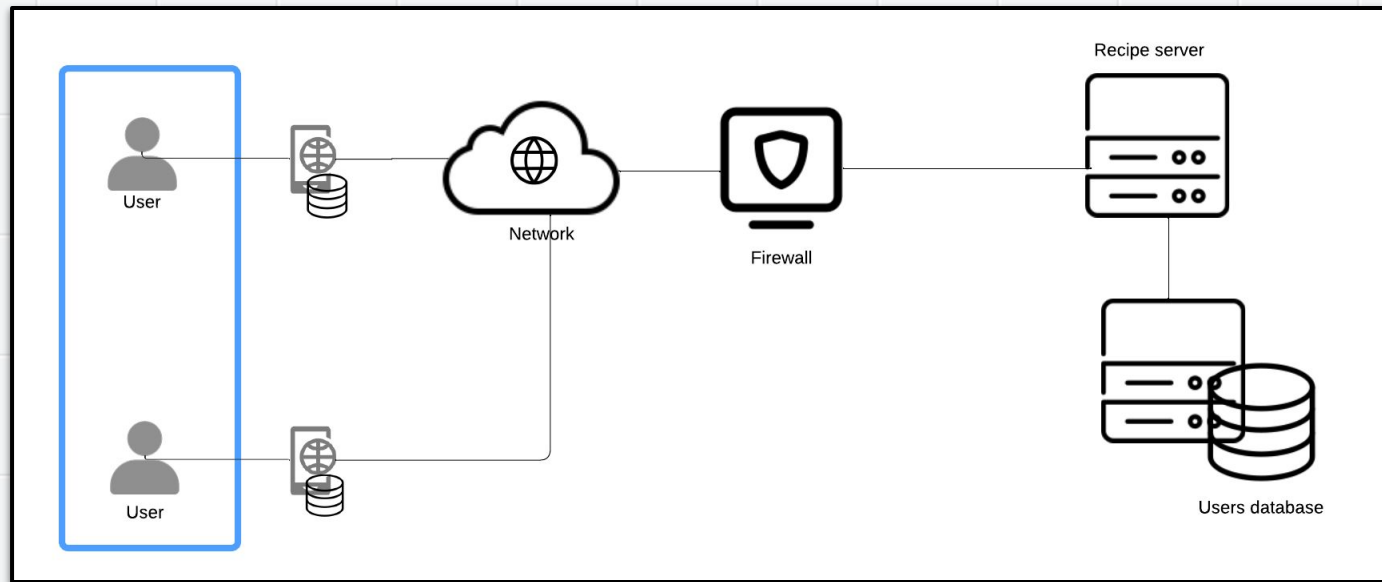
PROBLEM SOLUTION



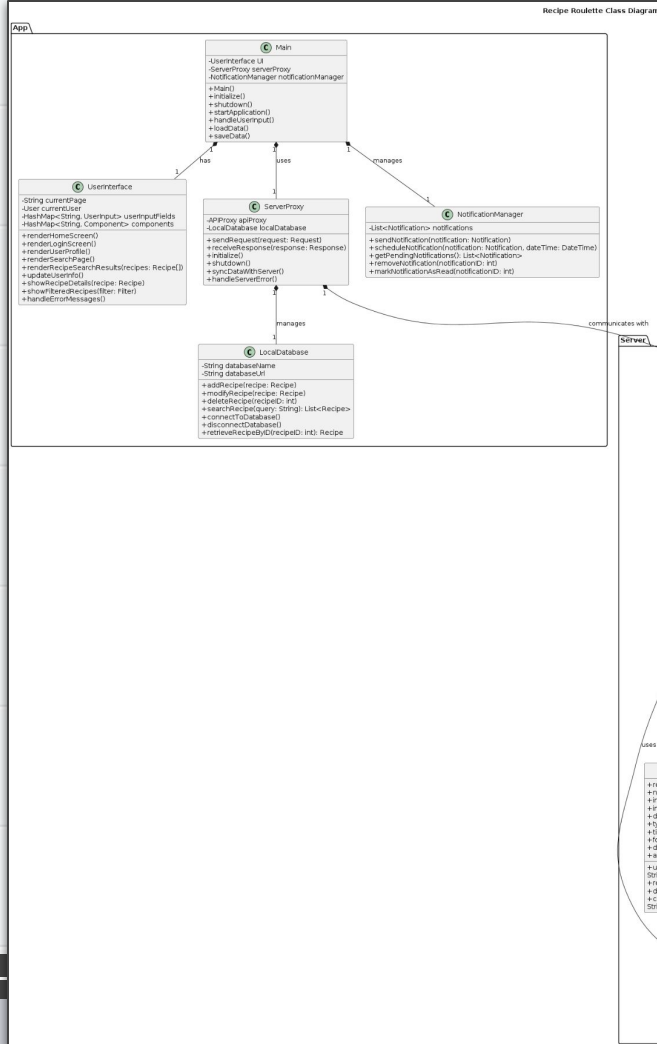
USE CASE DIAGRAM



ARCHITECTURE DIAGRAM

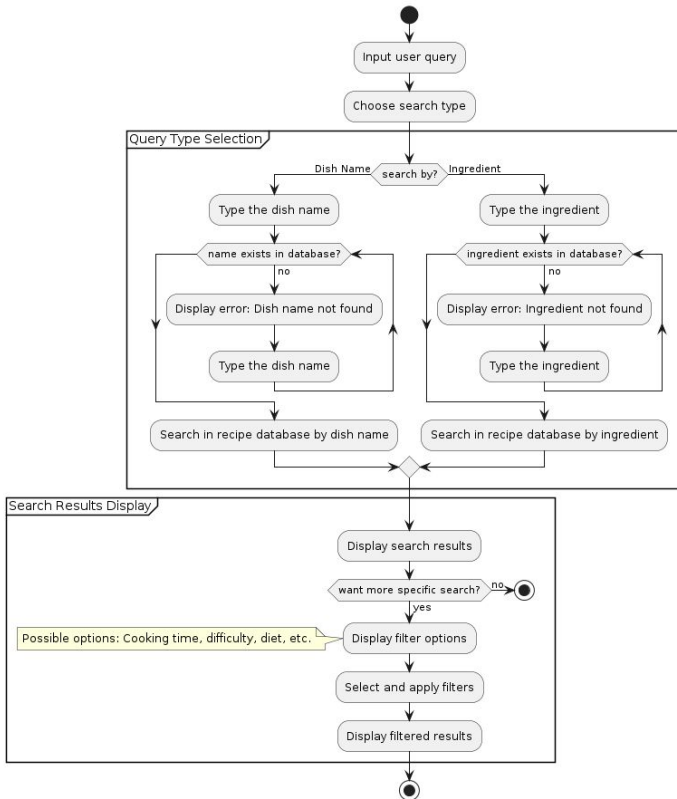


CLASS DIAGRAM

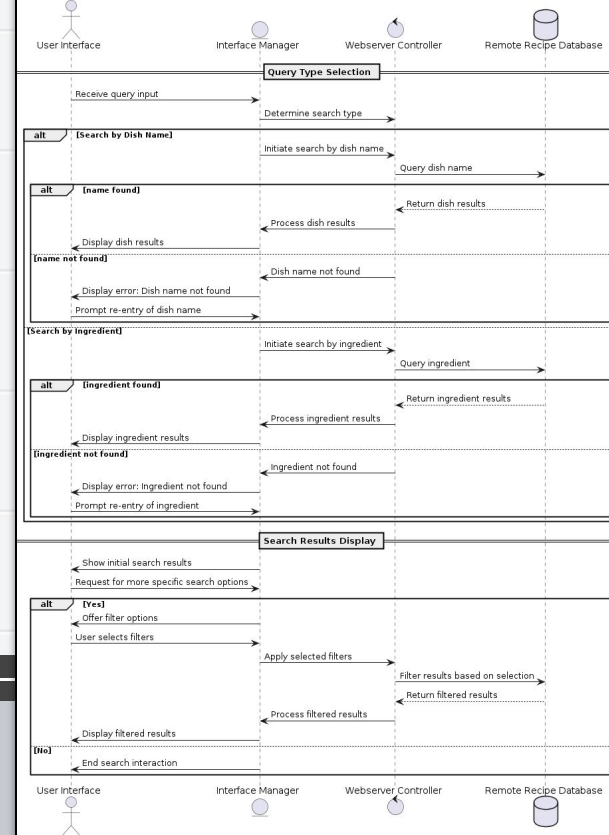


ACTIVITY/SEQUENCE DIAGRAM: UC2

UC2: Search Recipes - Activity Diagram



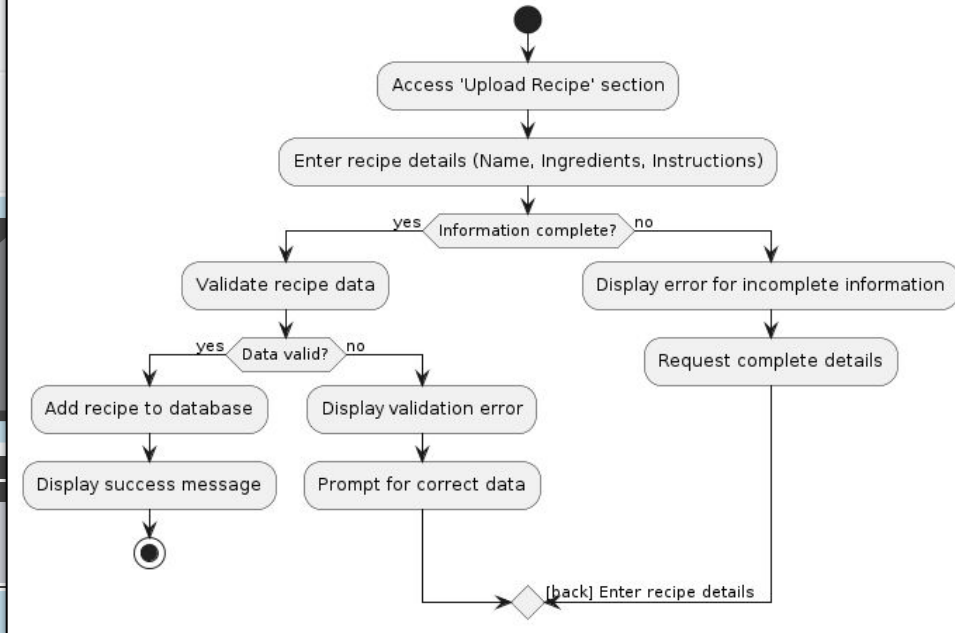
UC2: Search Recipes



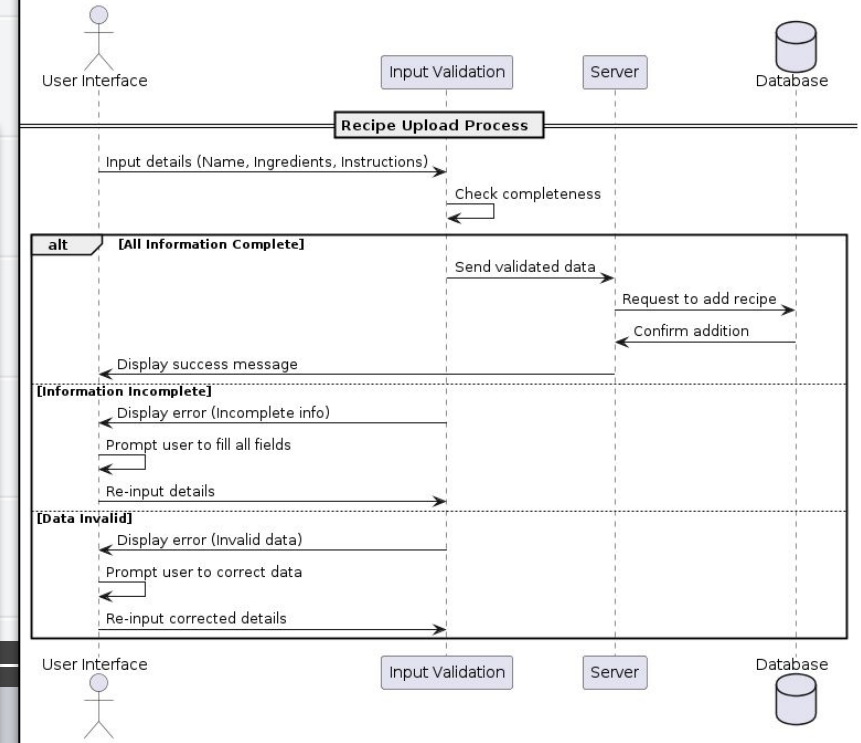
ACTIVITY/SEQUENCE DIAGRAM: UC3

Preconditions: User must be logged in.
Postconditions: The recipe is added to the database and available for others to view.

UC3: Upload Recipe - Activity Diagram

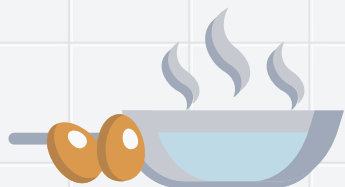


UC3: Upload Recipe - Sequence Diagram

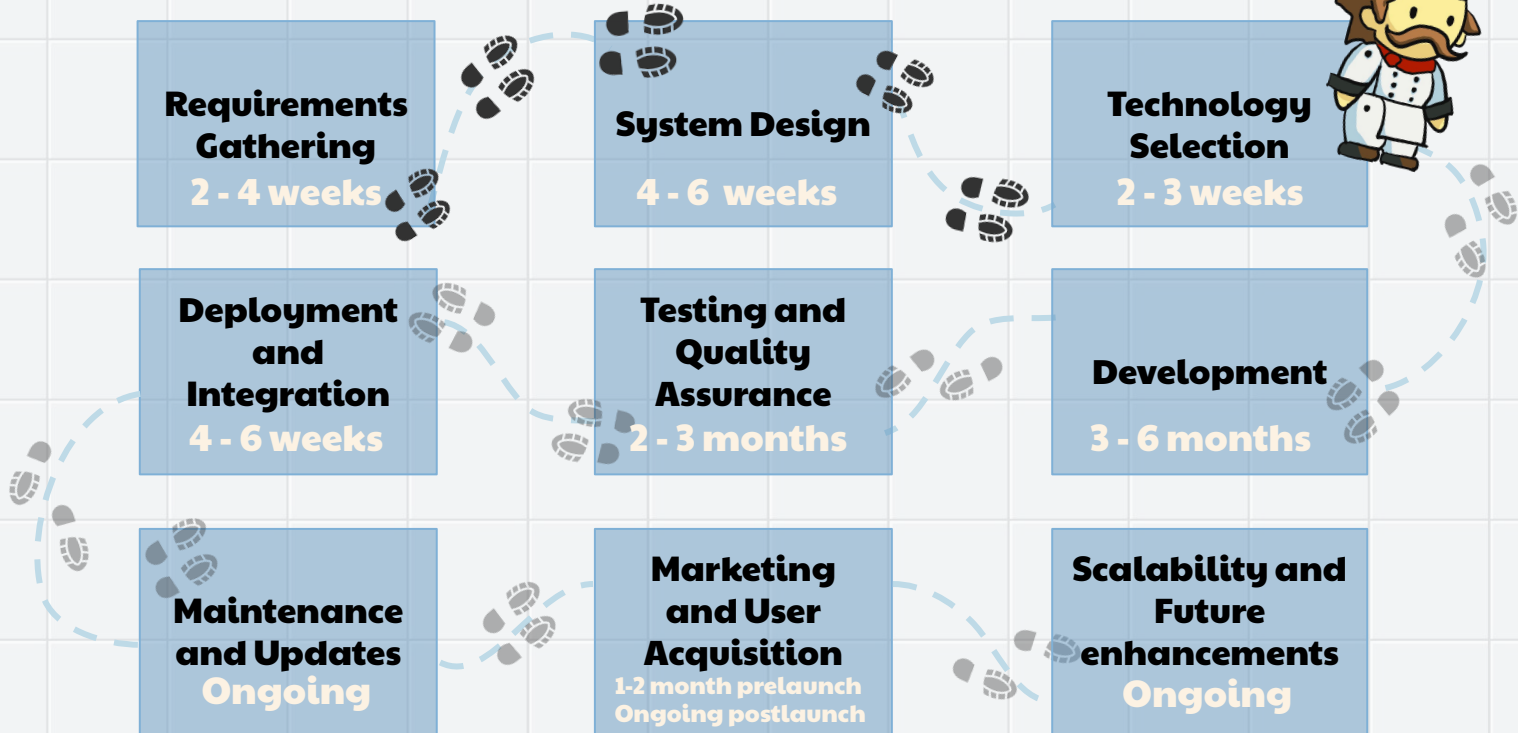


3

DEVELOPMENT



WORKFLOW





DEVELOPMENT

Iterative development

Agile methodologies → flexible & adaptive
Employ **sprint cycles** → increment features

Backend Development

User management
Recipe management
Server communication

Frontend Development

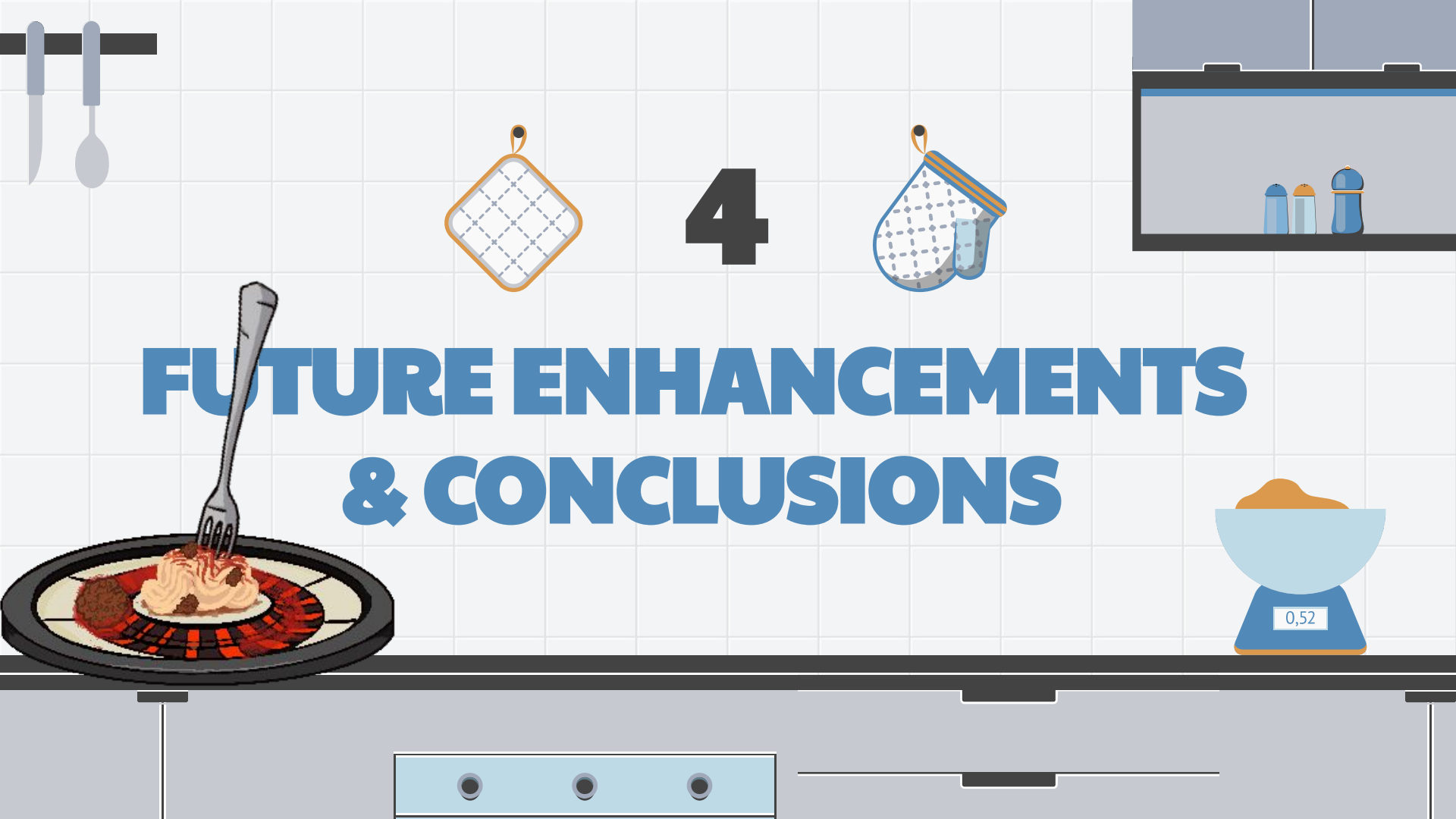
Responsive design
User Interface
Interactivity

Continuous Integration and testing

CI tools to automate testing and deployment
Perform regular code reviews and maintain coding standards

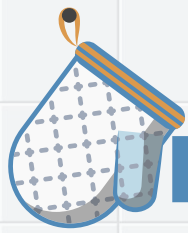
Continue with collaboration tools

GitHub for overall management
Trello for tracking process and issues



FUTURE ENHANCEMENTS & CONCLUSIONS

4



FUTURE ENHANCEMENTS

Ingredients delivery

- Delivery staff
- Association with supermarkets
- Points become discounts
- In-app messaging
- Payment gateways
- Maps

Expand languages available

- Increase accessibility and overall well-being of the user

More flexible design

Natural Language Processing (NLP)

- Use NLP to understand user queries better, allowing for more flexible and intuitive searches.

Incorporate a forum so users can share tips and experiences.



CONCLUSIONS

INNOVATIVE CULINARY ASSISTANT

- Minimize food waste
- Enhance cooking experience
- For novice and expert cooks

ENHANCED USER ENGAGEMENT

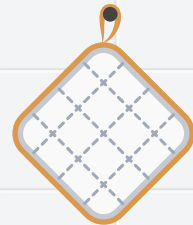
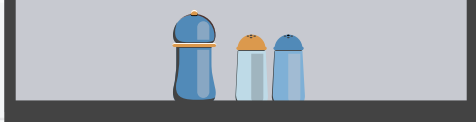
- Recipe uploads
- User-generated content
- Reward systems → active participation

ROBUST SYSTEM ARCHITECTURE

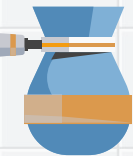
- Secure, scalable and efficient operations
- Capable of handling large user loads
- Consistent user experience

FUTURE GROWTH AND INTEGRATION

- Online grocery services and smart kitchen devices
- Expand its functionality



<https://github.com/carmensat/RECIPE-ROULETTE.git>



 SCAN ME



THANKS!

