**Final Year Project Report**

**Chefu**

****

**Project Advisor:**

**MAM ARIFAH AZHAR**

**Submitted By:**

**ABDUL BASIT: 12002065106**

**M ZAIN MOHSIN: 12003065331**

**M SOHAIB ASHARAF: 13004065071**

**Session:**

**2012 -2017**

**University of Management and Technology**

**C-II Johar Town Lahore Pakistan**

**Dedication**

We dedicate this project to our MAM ARIFAH AZHAR due to her dedication and efforts we are able to complete this project and she guided us in the right path so we able to meet the deadline and manage the project completion.

## 

**Final Approval**

**Panel of Examiners**

* **Head of Department**  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Department of Computer Science

UMT Lahore

* **Program Director (Final Year Projects)** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Department of Computer Science

UMT Lahore

* **Supervisor** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Department of Computer Science

UMT Lahore

* **Co-Supervisor** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Acknowledgment**

In the journey of this project, there are many names after Allah Almighty who helped us a lot in achieving our goals and with their support and efforts we are able to manage the special tasks timely. We acknowledge the support of our resource advisor Mam Arifah Azhar and our co-advisor Aqsa Ali. They guided us in the right direction for timely completion of our tasks. Our university provided a platform to perform the project under the supervision of such an intellectual and intelligent resource that not only provided us in the road map but also provided us new horizons for our future ventures as well. We are thankful to our teachers and university members who helped us a lot in every step and guided us in the right way.

**Project Title:** Chefu

**Objective:** Providing a same platform to chef and user so they both can get benefit from each other

**Undertaken by:** Abdul Basit

M Zain Mohsin

M Sohaib Ashraf

**Supervised by:**  Mam Arifah Azhar

**Starting Date:** 3 April 2016

**Completion Date:** 24 May 2017

**Tools Used:**

* PHP
* HTML
* CSS
* JavaScript
* My SQL server
* AJEX

**Operating System:** Macintosh, Windows

**Documentation:** Provided in subsequent chapters

**Plagairism ReportAbstract**

A systematic and organized platform for chefs and users for better taste and better environment. Relevant information regarding each entity that needs to be recorded, maintained and tracked when necessary. All modules in the system are connected for easy data access based on user roles. From signup to buying, hiring chefs based on location, generating codes, managing payment module to a specific portal for viewers to view different recipes, search recipes, search chefs, client can send a comment to the chef.

Our solution is designed to reduce complexity of managing data. The interface is designed to be user friendly so that a non-technical person can easily add and maintain records. The solution is web based platform created using code igniter framework. Not only it is multi browser compatible, it also works on mobile device being responsive in nature. Both client/chef has the permission to add/edit/delete any account.

We will keep everybody informed without any hassle and in short, the designed solution will save a lot of time, effort and money in cooking.

Revision Chart

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Primary Author(s) | Description of Version | Date Completed |
| Draft | Abdul Basit, Zain Mohsin, Suhiab Ashraf | Checked and suggestions for improvements. Corrections of mistakes | 3 april-12june  2016 |
| Preliminary | Abdul Basit | Its pre-final draft, checked and resolved mistakes. Suggestions for improvements | 30 june |
| Final Phase 1 | Abdul Basit | First complete documentation | 25 July 2016 |
| Revision 1.1 | Abdul Basit | Revised according to suggestions. | 3 August |
| Revision 1.2 | Abdul Basit | Revised according to suggestions | 18 August |
| Revision 1.3 | Abdul Basit | Revised according to suggestion | 2 September |
| Revision 1.4 | Abdul Basit | Revised according to new format | 1 jan |
| Revision 1.5 | Abdul Basit, Zain Mohsin, Suhiab Ashraf | Revised according to new format | 23 Jan |
| Revision 1.6 | Abdul Basit | Revised according to new format | 8 Feb |
| Revision 2.1 | Abdul Basit, Zain Mohsin | Prototypes | 2 April |
| Revision 2.2 | Abdul Baist , Zain Mohsin | Revised According to Suggestions | 15 April |
| Revision 2.3 | Zain Moshin , Suhaib Ashraf | Start working on Phase 2 | 5th May |
| Revision 2.4 | Zain Moshin , Suhaib Ashraf | Revised According to Suggestions | August |
| Revision 2.5 | Zain Moshin , Suhaib Ashraf | Revised According to Suggestions | September |
| Revision 2.6 | Zain Moshin , Suhaib Ashraf | Revised According to Suggestions | November |
| Revision 2.7 | Zain Moshin , Suhaib Ashraf | Revised According to Suggestions | December |
| Revision 2.8 | Zain Moshin , Suhaib Ashraf | Revised According to Suggestions | 8 January |
| Final Phase 2 | Zain Moshin , Suhaib Ashraf | Revised According to Suggestions | 4 Feb |

# Contents

Contents

[Contents 1](#_Toc487903722)

[Definitions and Acronyms 4](#_Toc487903723)

[Table 1: Acronyms and definitions 5](#_Toc487903724)

[List of Figures 6](#_Toc487903725)

[List of Prototypes 9](#_Toc487903726)

[1. Introduction 10](#_Toc487903727)

[1.1 Motivations 10](#_Toc487903728)

[1.2 Project Overview 10](#_Toc487903729)

[1.3 Problem Statement 10](#_Toc487903730)

[1.4 Objectives 10](#_Toc487903731)

[2. Domain Analysis 11](#_Toc487903732)

[2.1 Customer 11](#_Toc487903733)

[2.2 Stakeholders 11](#_Toc487903734)

[2.3 Affected Groups with Social or Economic Impact 12](#_Toc487903735)

[2.4 Dependencies/ External Systems. 12](#_Toc487903736)

[2.5 Reference Documents 12](#_Toc487903737)

[2.5.1 Related Projects 12](#_Toc487903738)

[2.5.2 Feature Comparison 13](#_Toc487903739)

[3. Requirements analysis 14](#_Toc487903740)

[3.1 Requirements 14](#_Toc487903741)

[3.2 List of Actors 16](#_Toc487903742)

[3.3 List of Use Cases 16](#_Toc487903743)

[3.4 System use case diagram 17](#_Toc487903744)

[3.5 Extended Use Cases 20](#_Toc487903745)

[3.6 User interfaces (mock screens) 36](#_Toc487903746)

[4. Data flow diagram (optional) 41](#_Toc487903747)

[4.1 Data Flow Diagram Level 0 41](#_Toc487903748)

[4.2 Data Flow Diagram Level 1 42](#_Toc487903749)

[4.3 Data Flow Diagram Level 2 45](#_Toc487903750)

[5. System Design 48](#_Toc487903751)

[5.1 System Architecture Diagram 48](#_Toc487903752)

[5.2 Class Diagram 49](#_Toc487903753)

[5.3 Collaboration Diagrams 50](#_Toc487903754)

[5.4 Other UMLs 51](#_Toc487903756)

[5.5 ERD 53](#_Toc487903757)

[5.6 Data Dictionary 54](#_Toc487903759)

[6. Implementation details 55](#_Toc487903760)

[6.1 Development Setup 55](#_Toc487903761)

[6.2 Deployment setup 55](#_Toc487903762)

[6.3 Algorithms 56](#_Toc487903763)

[User Login 58](#_Toc487903764)

[6.4 Constraints 59](#_Toc487903765)

[6.4.1 Assumptions 59](#_Toc487903766)

[6.4.2 System constraints 59](#_Toc487903767)

[6.4.3 Limitations 59](#_Toc487903768)

[7. Testing 60](#_Toc487903769)

[7.1 Extended Test Cases 60](#_Toc487903770)

[7.2 Decision Table 76](#_Toc487903771)

[7.2.1 Code snippet 76](#_Toc487903772)

[7.2.2 Decision coverage table 76](#_Toc487903773)

[7.3 Traceability Matrix 79](#_Toc487903774)

[7.3.1 Prototypes (UCID vs RID)RID 79](#_Toc487903775)

[7.3.2 Prototypes (RID vs PID)ID 80](#_Toc487903776)

[7.3.3 Test Cases (RID vs TID) 81](#_Toc487903777)

[7.3.4 Coverage (UCID vs TID) 82](#_Toc487903778)

[8. Results/Output/Statistics 83](#_Toc487903779)

[8.1 % Completion 83](#_Toc487903780)

[8.2 % Accuracy 83](#_Toc487903781)

[8.3 % Correctness 83](#_Toc487903782)

[9. Conclusion 84](#_Toc487903783)

[10. Future work 85](#_Toc487903784)

[11. Bibliography 86](#_Toc487903785)

[11.1 Books 86](#_Toc487903786)

[11.2 Journals 86](#_Toc487903787)

[11.3 Articles 86](#_Toc487903788)

[11.4 Research papers 86](#_Toc487903789)

[11.5 Other References 86](#_Toc487903790)

[12. Appendix 87](#_Toc487903791)

[12.1 Glossary of terms 87](#_Toc487903792)

[12.2 Pre-requisites 87](#_Toc487903793)

## Definitions and Acronyms

|  |  |
| --- | --- |
| Acronym | Definition |
| Chefu | Name of our project |
| DFD | Data Flow Diagram |
| SAD | System Architecture Diagram |
| UCID | Use Case Identity |
| UC | Use Case |
| TC | Test Case |
| RID | Requirement Identity |
| PID | Prototype Identity |
| TID | Test case Identity |
| ERD | Entity Relation Diagram |

## Table 1: Acronyms and definitions

## List of Figures

Figure 1: System Use Case Diagram 15

Figure 2: UC Diagram for Chef……………………………………………………………………………...16

Figure 3: UC Diagram for Recipe……………………………………………………………………………17

Figure 4: DFD Level 0 39

Figure 5: DFD Level 1 For Login 40

Figure 6: DFD level 1 For Hiring chef…………………………………………………………....................41

Figure 7: DFD Level 1 for Buy Recipe……………………………………………………….......................42

Figure 8: DFD Level 2 For Login 43

Figure 9: DFD Level 2 for Chef………………………………………………………………………….….44

Figure 10: DFD Level for Recipe……………………………………………………………………………45

Figure 11: System Architecture Diagram 46

Figure 12: Class Diagram 47

Figure 13: Collaborative Diagram 48

Figure 14: Sequence Diagram for Registeration 49

Figure 15: Sequence Diagram for Hiring 49

Figure 16: Sequence Diagram for Buying 50

Figure 17: Sequence Diagram for Comments 50

Figure 18: ERD 51

List of Tables

[Table 1: Acronyms and definitions ………………………………………………………………3](#_Toc475198518)

Table 2: List of stake holders……………………………………………………….…………………………9

[Table 3: Feature Comparison ..11](#_Toc475198519)

Table 4: Functional Requirement………………………………………………………….……………...…12

Table 5: Non-Functional Requirement…………………………………………………….………………...13

Table 6: UC 1……………………………………………………………………………………………......18

Table 7: UC 2……………………………………………………………………………………….……….19

Table 8: UC 3……………………………………………………………………………………….……….20

Table 9: UC 4……………………………………………………………………………………….……….21

Table 10: UC 5……………………………………………………………………………………….……...22

Table 11: UC 6……………………………………………………………………………………….…...…23

Table 12: UC 7……….…………………………………………………………………………………...…24

Table 13: UC 8……….……………………………………………………………………………………...25

Table 14: UC 9……….……………………………………………………………………………………...26

Table 15: UC 10……….…………………………………………………………………………………….27

Table 16: UC 11……….………………………………………………………………………………….…28

Table 17: UC 12……….………………………………………………………………………………….…29

Table 18: UC 13……….…………………………………………………………………………………….30

Table 19: UC 14………………………………………………………...………………………………...…31

Table 20: UC 15………...………………………………………………………………………………...…32

Table 21: UC 16…………...……………………………………………………………………………...…33

Table 22: Data Dictionary …...…………………………………………………………………………...…52

Table 23: TC 1…………………………………………………………………………………………….…58

Table 24: TC 2…………………………………………………………………………………………….…59

Table 25: TC 3…………………………………………………………………………………………….…60

Table 26: TC 4……………….………………………………………………………………………………61

Table 27: TC 5…………………………………………………………………………………………….…62

Table 28: TC 6……………………………………………………………………………………………….63

Table 29: TC 7…………………………………………………………………………………………….…64

Table 30: TC 8…………………………………………………………………………………………….…65

Table 31: TC 9…………………………………………………………………………………………….…66

Table 32: TC 10……………………………………………………………………………...………………67

Table 33: TC 11……………………………………………………………………………………………...68

Table 34: TC 12…………………………………………………………………………………………...…69

Table 35: TC 13…………………………………………………………………………………………...…70

Table 36: TC 14…………………………………………………………………………………………...…71

Table 37: TC 15…………………………………………………………………………………………...…72

Table 38: TC 16…………………………………………………………………………………………...…73

Table 39: Decision Table1 ………………………………………………………………………………..…74

Table 40: Decision Table 2………………………………………………………………………………..…75

Table 41: Decision Table 3…………………………………………………………………………………..76

Table 42: Decision Table 4………………………………………………………………………………..…76

Table 43: RID vs UCID……………………………………………………………………………………...77

Table 44: RID vs PID………………………………………………………………………………………..78

Table 45: RID vs TID………………………………………………………………………………………..79

Table 46: UCID vs TID……….……………………………………………………………………………..80

## 

## List of Prototypes

Prototypes 1: User Interface 34

Prototypes 2: Login 34

Prototypes 3: Features 35

Prototypes 4: Chef Profile 35

Prototypes 5: Chef List 36

Prototypes 6: Manage Chef Profile 36

Prototypes 7: Recipies 37

Prototypes 8: Recipies list 37

Prototypes 9: Contact 38

# Introduction

## Motivations

Cooking is an art, and representation of a culture, but now a day it is developed in to a big profession and it is expanding day by day through its means and there are very big names in chefs which are known by their specific tastes that can be delivered only by them. The industry is getting huge like there are many cooking channels e.g. Masla, Zaouq and many more etc. Some of the chefs make their channels on the YouTube but how one can find best of best since there is no such a platform where the user and chef can interact with each other.  
The idea came in our mind that there should be some platform where a chef and the user can interact with each other so the user get the best taste and the chef also get rewarded.

## Project Overview

Nowadays the world is progressing very fast. Technologies are enhancing day by day and engineers are working on web technologies for the betterment of lifestyle, by the help of web technologies we can able to buy products online and interact to our relatives, friends & family easily. So social media is a big platform on which people interact and work on daily basis. We decided to make a web portal named “*Chefu****”*** on which audience can able to find a chef, book a chef, able to see the best recipes etc. There will be an option for the new account on portal for client and if there already exist an account; automatically its geo-locations will be taken and system will tell the nearby chefs availability on map; also, one can able to find the best chef by the rating. Now there is a chef portal on which chefs will be able to create new account and log-in to interact with users. Chefs also able to publish their recipes on portal. Now chefs are able to earn this way as well as they can charge the amount for sharing their recipes.

## 1.3 Problem Statement

Our main effort is to provide the platform on which user can get the best chef for their use and the user can also buy best recipe by best chef base on the ratting of that profile chef. There is no platform in Pakistan which provide the portfolio for chef and the addition of the unique function buy recipe so the user can also buy the recipe and enjoy the same taste at home.

## Objectives

We wish to provide a platform on which a chef and user both can interact with each other. User get the required taste what he wants and chef get their fare rewarded.

# Domain Analysis

## Customer

Chefu is a web-portal which provide the platform where the user and the chef can interact with each other on the safe side, user can hire a chef or buy a recipe just on a click away, the targeted customer of this platform is chef and the user who wants to hire a chef, like restaurants, house wife’s, cooking institutes, event organization and as well as students, it is a safe platform and will be globally interact the client.

## Stakeholders

|  |  |
| --- | --- |
| Stake holders | Role in system |
| Owners | They are responsible for maintaining the website |
| Chefs | He is responsible for adding and deleting the recipes in his dashboard. He is responsible for entering the details of the recipe in terms of ingredient and steps. He will request the system and select that weather the recipe is free or for sale. |
| Restaurants | They can hire a chef from our system. They can also buy different recipes as well. |
| Cooking institute | They can also hire a chef from our system. |
| House wife | They can buy different delicious recipes or hire a chef for events and party functions. |
| Students | Students can learn from our system by watching different free recipes. |
| Bank | Client can easily buy recipes by using PayPal module. |

Table 2: List of stakeholders

## Affected Groups with Social or Economic Impact

* Online users.
* Online buyers.

## Dependencies/ External Systems.

* Web Server.

Our system is based on client server architecture that’s why it is dependent on webserver.

* PayPal Account

Transactions will be handled by PayPal methods.

* Internet Access.  
   Since our system is an online web portal so it is necessary to have an internet access.

## Reference Documents

.

### Related Projects

* “Hire a chef” is a web site provided by the United States Personal Chef Association.
* “Chef Pencil” is a website which run by some professional chefs.

### Feature Comparison

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sr No.** | **Comparison Feature** | **Chef Pencil** | **Hire a Chef** | **Remarks** |
| 1 | Login | There is only login for clients | Only chefs can login | Chefu provides Separate logins for client and chef. |
| 2 | Recipes | There are some limited recipes which are shown. is | Only for chef hiring not for selling recipes | Chefu provide the platform for displaying and selling ofrecipe |
| 3 | Payment Method | Payment process is not supported by PayPal | Payment method is not defiend | Chefu provides payment method in the form of Paypal |

Table 3: Feature comparison

# Requirements analysis

## Requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| * **Ref #** | * **Functions** | * **Category** | * **Attribute** | **Details & Boundary Constraints** |
| * R1 | Visitors fill the sign up form.to become client. | * Functional(Evident) | Sign Up | * Information about user, save to data base. Provide legal information like name, username, password, location, City. |
| * R2 | * Client should Log in first to buy. | Functional | Log in | Provide correct Information. |
| * R3 | Chefs should fill the Sign-up Form. | Functional (Evident) | Sign Up | * Information about chef, save to data base. Provide legal information like name, username, password, location, City. |
| * R4 | * Chef should Log in first to add and manage profile. | Functional | Log in | Provide correct Information. |
| * R5 | Visitor/client can search by recipe. | Functional | Search | Search by the recipe and provided with the correct results. |
| * R6 | Visitor/client can search by chef | Functional | Search | Search buy chefs and provided with the results. |
| * R7 | Client can update their profile | Functional | Update | * Update the profile if the user |
| * R8 | Chef can update/manage the profile | Functional | Manage | Update/manage profile |
| * R9 | Client can hire the chef for their services and send comments. | Functional | Hire | Hire the chef for the users |
| * R10 | Chefs can add the recipe in their dashboard | Functional | * Addition | * Recipe added to the dash board of chef |
| * R11 | * Client can rate a chef | Functional | Rating | System should rate the chef. |
| * R12 | Client can rate a recipe | Functional | Rating | System should rate the recipe. |

Table 4: Functional requirements

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| * R1 | Minimum time for visitor/client search Result | * non-functional | * System Response time | * Visitor/client search Result will appear within 25 seconds |
| * R2 | Client credentials should be secure | * non-functional | * Security | System should secure the details. |
| * R3 | Easy to access the required chefs | * non-functional | Access Efficiency | * Easily can access or chat with another user. |
| * R4 | Interface must be efficient | * non-functional | * Usability | Easy to use and must be clear. |

Table 5: Non-functional requirements

## List of Actors

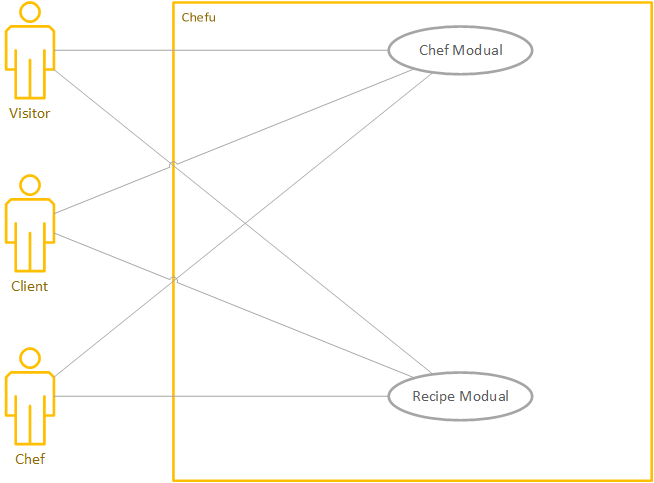
* Visitor
* Client
* Chef

## List of Use Cases

* Signup
* Login
* Forget password
* Search a chef
* Search a recipe
* Add recipe
* View recipe
* Delete recipe
* Update recipe
* Rate a chef
* Rate a recipe

## System use case diagram

System Use Case Diagram of Chefu



**Figure 1: System UC diagram**

UC Diagram of Chef Module

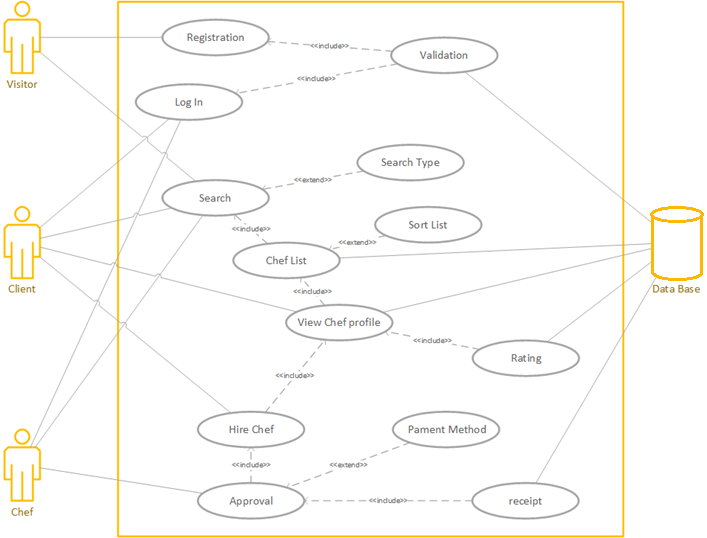
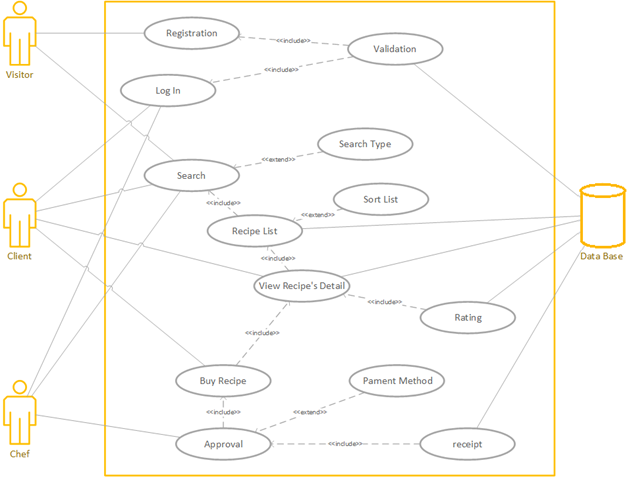


Figure 2: UC Diagram of Chef Module

UC Diagram of Recipe Module



**Figure 3: UC Diagram of Recipe Module**

## Extended Use Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-1 | | | |
| **UC C Name:** | Sign up | | | |
| **Created By:** | Abdul Basit | | **Last Updated By:** | Abdul Basit |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Visitor | | |
| **Description:** | | A Visitor will be able to sign up | | |
| **Trigger:** | | Go to the homepage of chefu | | |
| **Preconditions:** | | Internet connection, setup of application, server up | | |
| **Post condition:** | | Successful sign up | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Go to the homepage of Chefu | 2 | System execute the front page with (sign up /login) | |  |  | 3 | It gives sign up option and login | | 4 | Visitor select the signup option | 5 | The system gives the mandatory fields to fill. | | 6 | Visitor Enter information required field (Name, email, password, cell no, blood group, Interested games) |  |  | | 7 | Click on the submit button | 8 | Proceed the request | |  |  | 9 | Successful signup | | | | | |
| **Alternative Flows:** | | Step 2:  System crash and does not load the signup page  1-Check internet connection  2-Again go back to step 1  Step 8:  Incorrect field type or empty field  1-Show error message  2-Go back to step 5 | | |
| **Exceptions:** | | Internet Connection error, Server down, application error | | |
| **Includes:** | | Create profile | | |
| **Frequency of Use:** | | At least once per new user. | | |
| **Special Requirements:** | | Internet connection and web browser | | |
| **Assumptions:** | | User knows to use web applications and provide correct information | | |
| **Notes and Issues:** | | None | | |

**Table 6: UC 1**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-2 | | | |
| **UC Name:** | Login | | | |
| **Created By:** | Abdul Basit | | **Last Updated By:** | Abdul Basit |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | A Client will be able to login for hiring a chef or to buy a recipe | | |
| **Trigger:** | | Open Sign up page | | |
| **Preconditions:** | | Signup, Internet Connection | | |
| **Post conditions:** | | Login | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Enter client Name and Password | 2 | Verify | |  |  | 3 | Successful login | | | | | |
| **Alternative Flows:** | | Step 2:  Could not verify   1. Invalid information 2. Step back to 1   Step 3:  Invalid information or no internet connection  1-Show error message  2-Go back to step 1 | | |
| **Exceptions:** | | Internet Connection problem | | |
| **Includes:** | | No | | |
| **Frequency of Use:** | | One login requires for specific time | | |
| **Special Requirements:** | | Internet connection and web browser | | |
| **Assumptions:** | | User knows to use web applications and provide correct information | | |
| **Notes and Issues:** | | None | | |

**Table 7: UC 2**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-3 | | | |
| **UC Name:** | Forgot password | | | |
| **Created By:** | Abdul Basit | | **Last Updated By:** | Abdul Basit |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | Client will be able to change password | | |
| **Trigger:** | | Click on forgot password button | | |
| **Preconditions:** | | Must have signup | | |
| **Post conditions:** | | New password | | |
| **Normal Flow:**   |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client click on the forgot password | 2 | A new page password recovery open | | 3 | Client enter the client name | 4 | System Check the data | |  |  | 5 | System verifies | |  |  | 6 | System give the option for new password | | 7 | Client enter new password and select the submit button | 8 | Successfully change password | | | | | |
| **Alternative Flows:** | | Step 5:  Invalid user name or enter other data   1. Shows invalid data 2. Go back to step 2 | | |
| **Exceptions:** | | Client does not exist or resembles to many  Internet connection | | |
| **Includes:** | | Extend View Profile | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet connection and web browser | | |
| **Assumptions:** | | User knows to use web applications and provide correct information | | |
| **Notes and Issues:** | | None | | |

**Table 8: UC 3**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-4 | | | |
| **UC Name:** | Search chef | | | |
| **Created By:** | Abdul Basit | | **Last Updated By:** | Abdul Basit |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | A Client will find a chef | | |
| **Trigger:** | | Go to search Bar | | |
| **Preconditions:** | | Enable Location | | |
| **Post conditions:** | | Found and shows list | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Visitor/Client select the search bar option and search query. | 2 | Open page of Chefs list | |  |  |  |  | | 3 | Client select Chefs by their locations. | 4 | System shows the drop-down menu for Chefs by name. | |  |  |  |  | | 5 | Client select the Chef and view profile. | 6 | System shows the results of available Chefs. | |  |  |  |  | | | | | |
| **Alternative Flows:** | | Chef unavailable.  Go back to step no 1. | | |
| **Exceptions:** | | Internet connection problem or any system error | | |
| **Includes:** | | None | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet connection and web browser. | | |
| **Assumptions:** | | Client/Visitor knows to use web applications. | | |
| **Notes and Issues:** | | None | | |

**Table 9: UC 4**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-5 | | | |
| **UC Name:** | Search a recipe | | | |
| **Created By:** | Abdul | | **Last Updated By:** | Basit |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Visitor/Client | | |
| **Description:** | | When user wants to search a recipe | | |
| **Trigger:** | | Go to search bar | | |
| **Preconditions:** | | Connected to internet. | | |
| **Post condition:** | | Found and shows list | | |
| **Normal flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client/Visitor select the search bar for recipe and enter query. | 2 | System shows the recipe w.r.t to which user type. | | 3 | Client/Visitor select the recipe | 4 | System shows the recipe | | 5 | Client/Visitor saw the recipe |  |  | | | | | |
| **Alternative Flows:** | | Recipe unavailable if query is not correct.  Go back to step no 1. | | |
| **Exceptions:** | | Internet connection problem or any system error | | |
| **Includes:** | | None | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Android Phone, Global Positioning System | | |
| **Assumptions:** | | User are online and their locations switched on | | |
| **Notes and Issues:** | | None | | |

**Table 10: UC 5**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Use Case ID:** | UC-6 | | | |
| **Use Case Name:** | Add recipes | | | |
| **Created By:** | Abdul Basit | | **Last Updated By:** | Abdul Basit |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Chef | | |
| **Description:** | | Only chef can add the recipes to their dashboard | | |
| **Trigger:** | | Relevant “Add Recipe” button on the menu to add relevant user | | |
| **Preconditions:** | | User should be logged in | | |
| **Post condition:** | | Relevant recipe successfully added | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Clicks on add recipe button from dashboard | 2 | System execute the request and open the add recipe page | | 3 | On add recipe page, user is presented with 3 options to add new recipe   1. Type User click on the relevant type | 4 | System show the relevant form fields of add new recipe on the page to the user | | 5 | Chef fill the relevant form fields to add new teacher | 6 | System ensures all mandatory fields are filled | | 7 | Chef clicks on add recipe | 8 | The required recipe will successfully add | | | | | |
| **Alternative Flows:** | | Step 3:  1-User clicks on add relevant recipe  2-Go to step 4  3-User fill the relevant form fields to add new ingredient  4-Go to step 6  5-Go to step 7  6-Go to step 8 | | |
| **Exceptions:** | | Step 7a:  1-System crash or internet connection error  2-Check internet connection  3-Refresh the webpage  4-Go to step 1 again | | |
| **Includes:** | | UC-2 | | |
| **Frequency of Use:** | | Every time when chef want to add new recipe | | |
| **Special Requirements:** | | Internet connection and web browser | | |
| **Assumptions:** | | User knows to use web applications and provide correct information | | |
| **Notes and Issues:** | | None | | |

**Table 11: UC 6**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **UC ID:** | UC-7 | | | | | | | | |
| **UC Name:** | View recipe list | | | | | | | | |
| **Created By:** | | Abdul Basit | | | | **Last Updated By:** | | Abdul Basit |
|  | | 6 June | | | | **Last Revision Date:** | | 10 November |
| **Actors:** | | | | Chef | | | | |
| **Description:** | | | | Chef can view the list of students accordingly | | | | |
| **Trigger:** | | | | “Recipe” button in the menu of relevant chef | | | | |
| **Preconditions:** | | | | Recipe must be added there in | | | | |
| **Post condition:** | | | | User can view the recipe list accordingly to | | | | |
| **Normal Flow:** | | | | | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Chef clicks on recipe button in the menu | 2 | System execute the request and open the recipe page | | 3 | On recipe page user clicks on View recipe list | 4 | System execute the request and generate a recipe list according to the chef | |  |  | 5 | Chef successfully view recipe list | | | | | | | | | |
| **Alternative Flows:** | | | | Step 1a:  1-Chef clicks on recipe button  2-Go to step 2 again  3-Go to step 3 again  4-Go to step 4 again  5-Go to step 5 again | | | | |
| **Exceptions:** | | | | Step 3a:  1-System crash or internet connection error  2-Check internet connection  3-Refresh the webpage  4-Go to step 1 again | | | | |
| **Includes:** | | | | UC-2 | | | | |
| **Frequency of Use:** | | | | Every time when user clicks to view list | | | | |
| **Special Requirements:** | | | | Internet and web browser | | | | |
| **Assumptions:** | | | | User knows to use web applications | | | | |
| **Notes and Issues:** | | | | None | | | | |
| **Table 12: UC 7** | | | |  | | | | |
| **UC ID:** | UC-8 | | | | | | | | |
| **Use Case Name:** | Delete recipe | | | | | | | | |
| **Created By:** | Abdul Basit | | | | **Last Updated By:** | | Abdul Basit | | |
| **Date Created:** | 6 June | | | | **Last Revision Date:** | | 10 November | | |
| **Actors:** | | | Chef | | | | | | |
| **Description:** | | | Chef can delete a recipe from the recipe | | | | | | |
| **Trigger:** | | | “Delete” button in the student list | | | | | | |
| **Preconditions:** | | | Chef should be logged in to delete recipe from the list | | | | | | |
| **Post condition:** | | | Recipe successfully deleted from the list | | | | | | |
| **Normal Flow:** | | | | | | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Chefs clicks on Recipe list button | 2 | System execute their request and open the recipe list | | 3 | On recipe list user clicks on delete a recipe button | 4 | System execute the request and Delete the requested recipe from database and screen | |  |  | 5 | System update the database | |  |  | 6 | System refresh the recipe list page | |  |  | 7 | Requested recipe successfully deleted | | | | | | | | | | |
| **Alternative Flows:** | | | Step 3:  1-Relative id is not chef and can’t see delete button on screen  2-Logged in to the chef account  3- Go to step 1 | | | | | | |
| **Exceptions:** | | | Step 3a:  1-System crash or internet connection error  2-Check internet connection  3-Refresh the webpage  4-Go to step 1 again | | | | | | |
| **Includes:** | | | UC-1 | | | | | | |
| **Frequency of Use:** | | | Every time when user wants to deletes the recipe | | | | | | |
| **Special Requirements:** | | | Internet and web browser | | | | | | |
| **Assumptions:** | | | User knows to use web applications | | | | | | |
| **Notes and Issues:** | | | None | | | | | | |

**Table 13: UC 8**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-9 | | | |
| **UC Name:** | Update Profile | | | |
| **Created By:** | Abdul Basit | | **Last Updated By:** | Abdul Basit |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client, chef | | |
| **Description:** | | Client /chef can update his/her username, password. | | |
| **Trigger:** | | Click on Update Profile button | | |
| **Preconditions:** | | Already sign up | | |
| **Post conditions:** | | Successfully updated profile | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client/visitor enter into update profile | 2 | System shows the things he/she can update | | 3 | Client/visitor can change user name |  |  | | 4 | Click on the submit button | 5 | Successfully updated | | | | | |
| **Alternative Flows:** | | Step 2:  Invalid data   1. Go back to step 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | View profile | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | |  | | |
| **Assumptions:** | |  | | |
| **Notes and Issues:** | | None | | |

**Table 14: UC 9**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-10 | | | |
| **UC Name:** | Rate a chef | | | |
| **Created By:** | Zain Mohsin | | **Last Updated By:** | Zain Mohsin |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | Client can rate a chef | | |
| **Trigger:** | | Click to rate | | |
| **Preconditions:** | | Already sign up | | |
| **Post conditions:** | | Successfully rated the chef | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | User drag the mouse at the stars | 2 | System highlights the stars | | 3 | User click on the stars |  | System rated recipe | | 4 |  | 5 | Successfully rated | | | | | |
| **Alternative Flows:** | | Step 3:  User drag but not click.  1: Go back to step no 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | View rating | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet and web browser | | |
| **Assumptions:** | | User knows to use web applications | | |
| **Notes and Issues:** | | None | | |

**Table 15: UC 10**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-11 | | | |
| **UC Name:** | Rate a recipe | | | |
| **Created By:** | Zain Mohsin | | **Last Updated By:** | Zain Mohsin |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | Client can rate a chef | | |
| **Trigger:** | | Click to rate | | |
| **Preconditions:** | | Already sign up | | |
| **Post condition:** | | Successfully rated the chef | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client drag the mouse at the stars to rate | 2 | System highlights the stars | | 3 | Client click on the stars |  | System rated recipe | | 4 |  | 5 | Successfully rated | | | | | |
| **Alternative Flows:** | | Step 3:  User drag but not click.  1: Go back to step no 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | View rating | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet and web browser | | |
| **Assumptions:** | | User knows to use web applications | | |
| **Notes and Issues:** | | None | | |

**Table 16: UC 11**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-12 | | | |
| **UC Name:** | Comment | | | |
| **Created By:** | Zain Moshin | | **Last Updated By:** | Zain Mohsin |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | Client can comment on a chef | | |
| **Trigger:** | | Click to comment | | |
| **Preconditions:** | | Already sign up | | |
| **Post condition:** | | Successfully comment send. | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Write a comment | 2 | Comment appears | | 3 | Click on comment | 4 | Successfully commented | |  |  |  |  | | | | | |
| **Alternative Flows:** | | Step 3:  User drag mouse to send the comment.  1: Go back to step no 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | View comments | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet and web browser | | |
| **Assumptions:** | | User knows to use web applications | | |
| **Notes and Issues:** | | None | | |

**Table 17: UC 12**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-13 | | | |
| **UC Name:** | Minimum time for research result | | | |
| **Created By:** | Zain Moshin | | **Last Updated By:** | Zain Mohsin |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | Client /visitor can find on the search bar | | |
| **Trigger:** | | Click on Search bar | | |
| **Preconditions:** | | Already sign up | | |
| **Post condition:** | | Successfully completed the search | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client drag the mouse at the search bar | 2 | System blink the cursor on search bar | | 3 | Client can write down on search bar |  | System search the query | | 4 |  | 5 | Successfully searched | | | | | |
| **Alternative Flows:** | | Step 3:  User drag but not click.  1: Go back to step no 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | View Search result | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet and web browser | | |
| **Assumptions:** | | User knows to use web applications | | |
| **Notes and Issues:** | | None | | |

**Table 18: UC 13**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-14 | | | |
| **UC Name:** | Client credentials should be secured | | | |
| **Created By:** | Suhaib Ashraf | | **Last Updated By:** | Suhaib Ashraf |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | System | | |
| **Description:** | | The user credentials should be secured | | |
| **Trigger:** | | Click to rate | | |
| **Preconditions:** | | Already sign up | | |
| **Post condition:** | | Successfully credentials are secured | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client drag the mouse at the credentials | 2 | The cursor starts blinking at credentials | | 3 | Client enter his credentials |  | System secure the credentials | | 4 |  | 5 | Successfully secured | |  |  |  |  | |  |  |  |  | | | | | |
| **Alternative Flows:** | | Step 3:  User drag but not click.  1: Go back to step no 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | Securing of credentials | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet and web browser | | |
| **Assumptions:** | | User knows to use web applications | | |
| **Notes and Issues:** | | None | | |

**Table 19: UC 14**

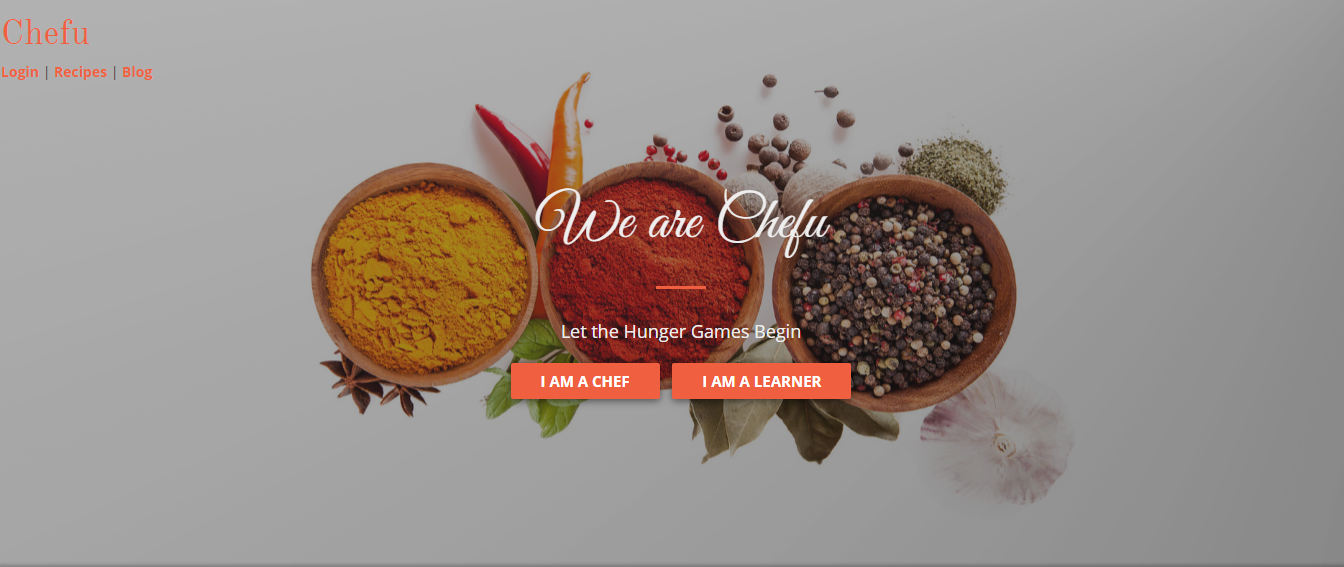
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-15 | | | |
| **UC Name:** | Ease of access | | | |
| **Created By:** | Abdul Basit | | **Last Updated By:** | Abdul Baist |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | Client can chat with chefs | | |
| **Trigger:** | | Click to Chat option | | |
| **Preconditions:** | | Already sign up | | |
| **Post condition:** | | Easily access to chefs | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client drag the mouse at the chat option and click | 2 | System open the chat window | | 3 | Client click on the chat window and type message |  | System typed the message | | 4 |  | 5 | Successfully access to chefs | | | | | |
| **Alternative Flows:** | | Step 3:  User drag but not click.  1: Go back to step no 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | Chat with chefs | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet and web browser | | |
| **Assumptions:** | | User knows to use web applications | | |
| **Notes and Issues:** | | None | | |

**Table 20: UC 15**

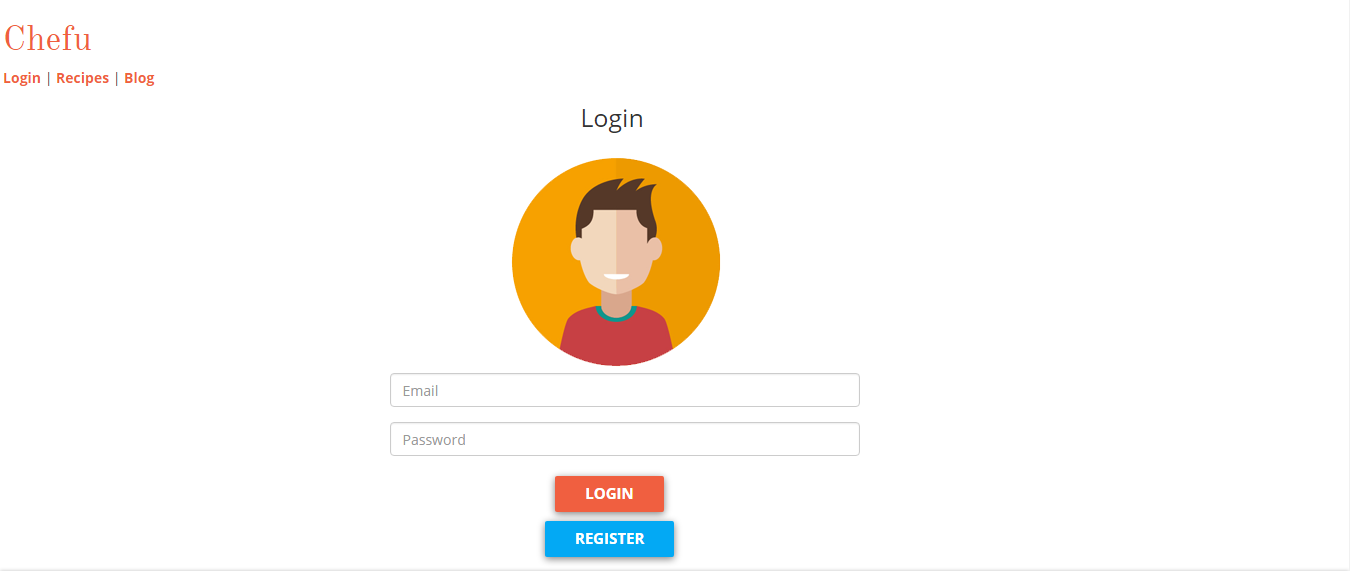
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **UC ID:** | UC-16 | | | |
| **UC Name:** | Usability | | | |
| **Created By:** | Zain Mohsin | | **Last Updated By:** | Zain Mohsin |
| **Date Created:** | 6 June | | **Last Revision Date:** | 10 November |
| **Actors:** | | Client | | |
| **Description:** | | Client use the system | | |
| **Trigger:** | | Click to any functionality of the system | | |
| **Preconditions:** | | Already sign up | | |
| **Post condition:** | | Successfully easily use the functionality of system | | |
| **Normal Flow:** | | | | |
| |  |  |  |  | | --- | --- | --- | --- | | Actor Action | | System Response | | | 1 | Client drag the mouse at any Functionality | 2 | System response to the function | |  |  | 3 | Successfully easily perform require function | |  |  |  |  | | | | | |
| **Alternative Flows:** | | Step 3:  User drag but not click.  1: Go back to step no 1 | | |
| **Exceptions:** | | Internet connection error | | |
| **Includes:** | | View rating | | |
| **Frequency of Use:** | | Unlimited | | |
| **Special Requirements:** | | Internet and web browser | | |
| **Assumptions:** | | User knows to use web applications | | |
| **Notes and Issues:** | | None | | |

**Table 21: UC 16**

## User interfaces (mock screens)



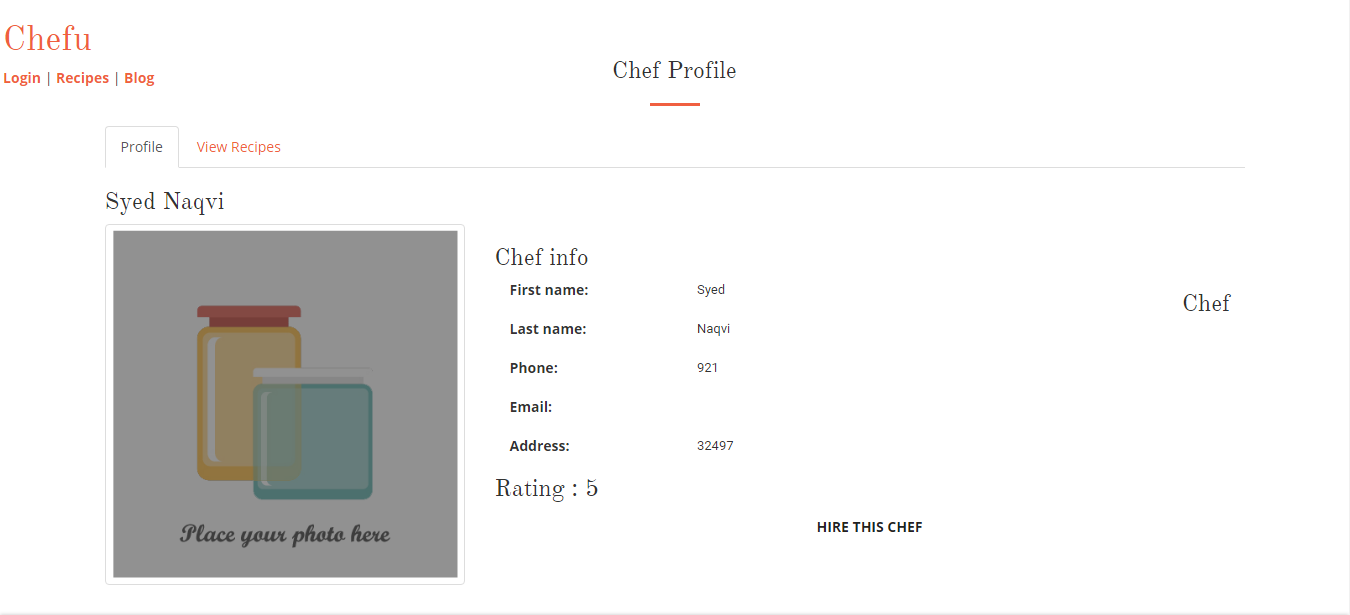
Prototype : User Interface



Prototype 2: Login



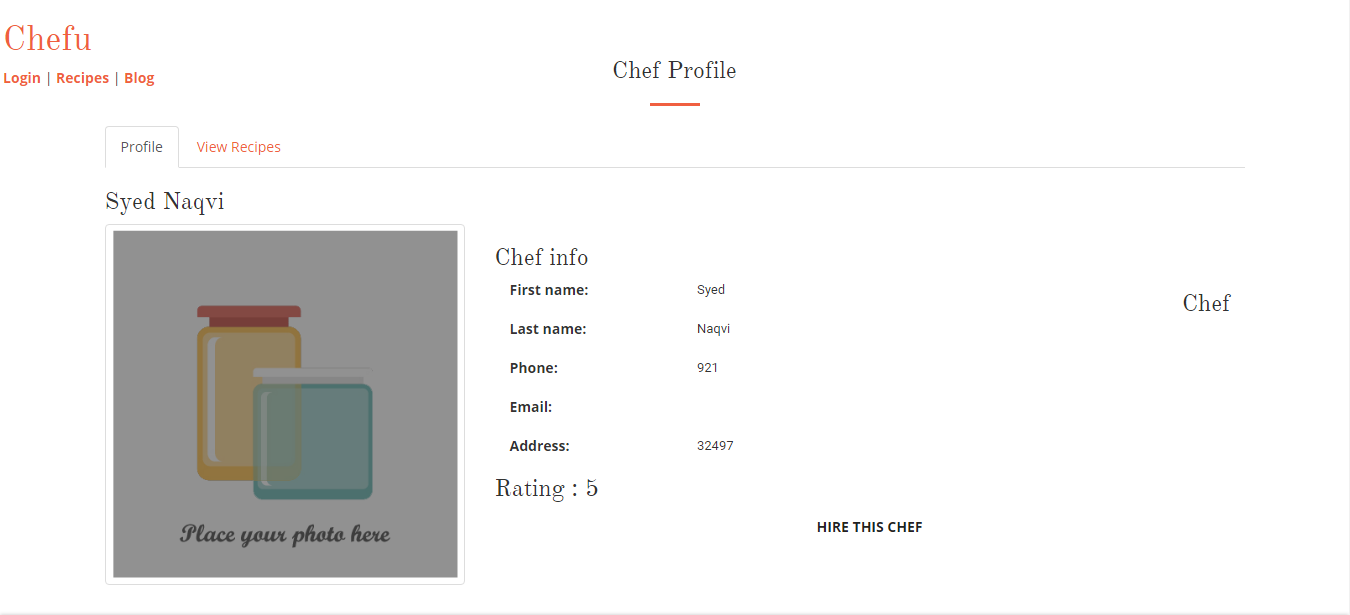
Prototype 3: Features



Prototype 4: Chef Profile



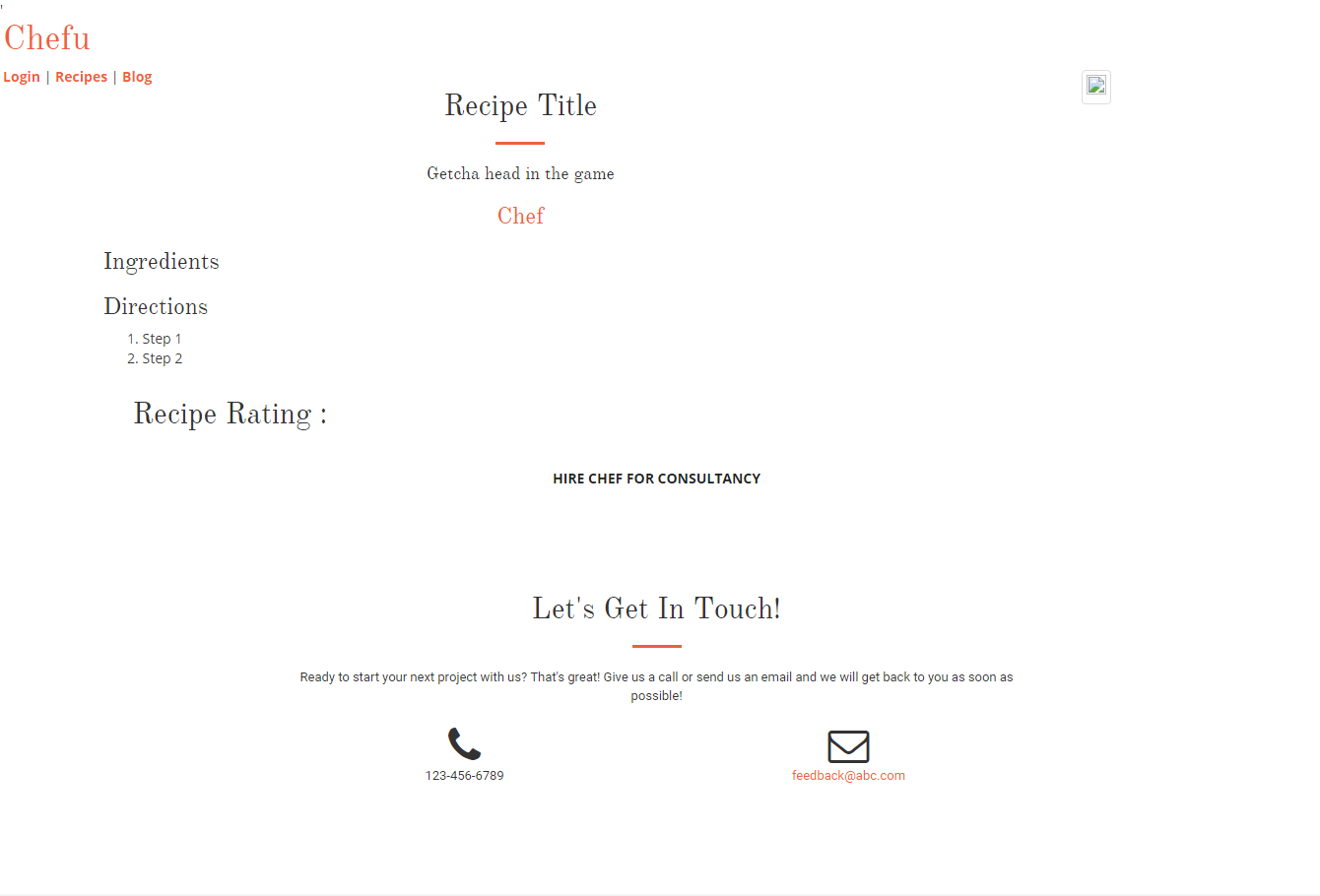
Prototype 5: List of Chefs



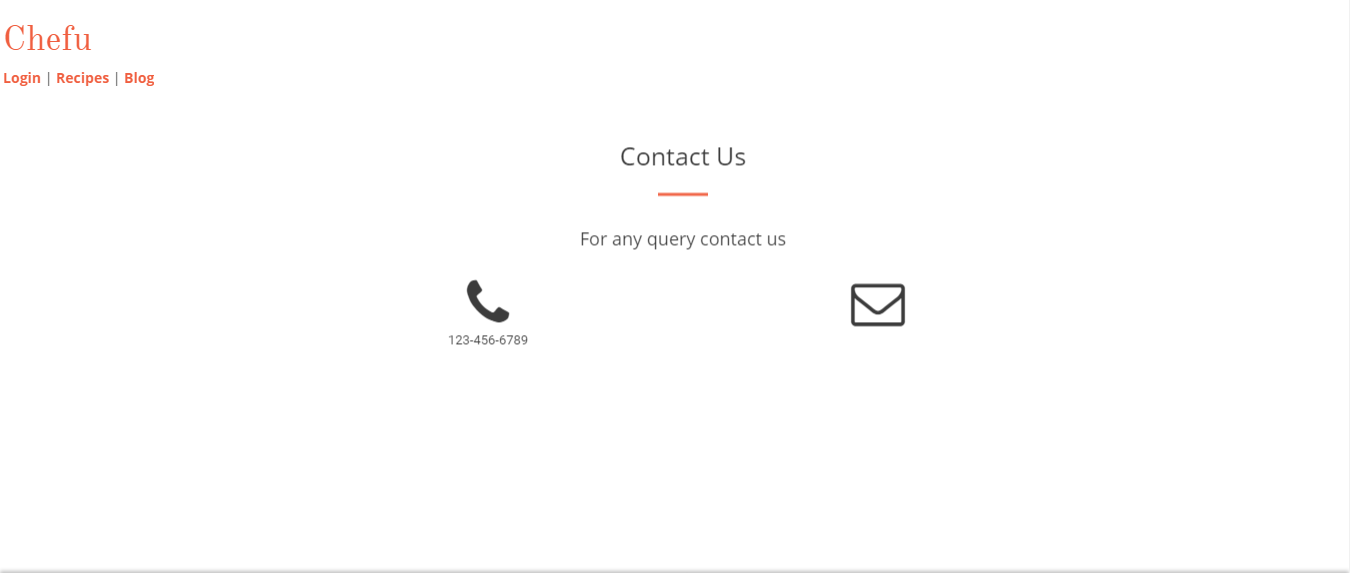
Prototype 6: Chefs Profile



Prototype 7: Recipes



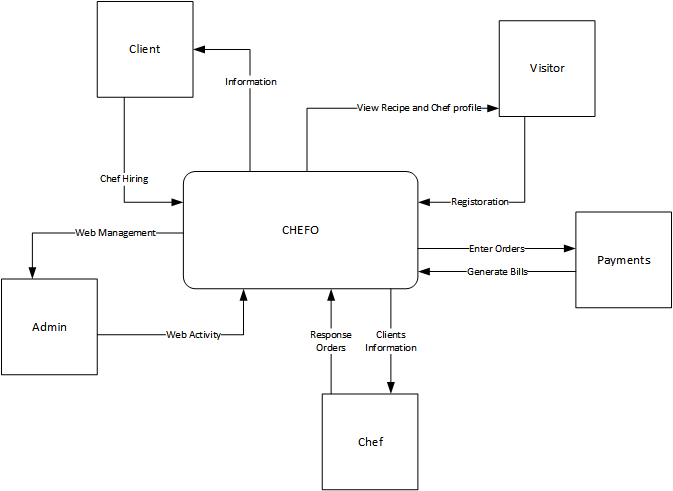
Prototype 8: Recipe details



Prototype 9: Contact

# Data flow diagram (optional)

## Data Flow Diagram Level 0



**Figure 4: DFD Level 0**

## Data Flow Diagram Level 1

**DFD L1 (Login and Registration)**

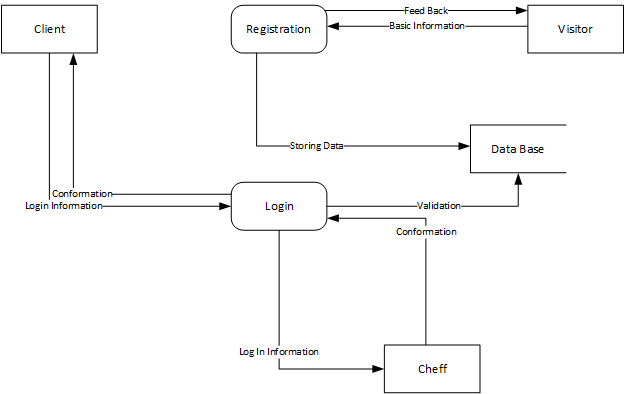


Figure 5: DFD Level 1 (Login and Registration)

**DFD L1 (Hiring a Chef)**

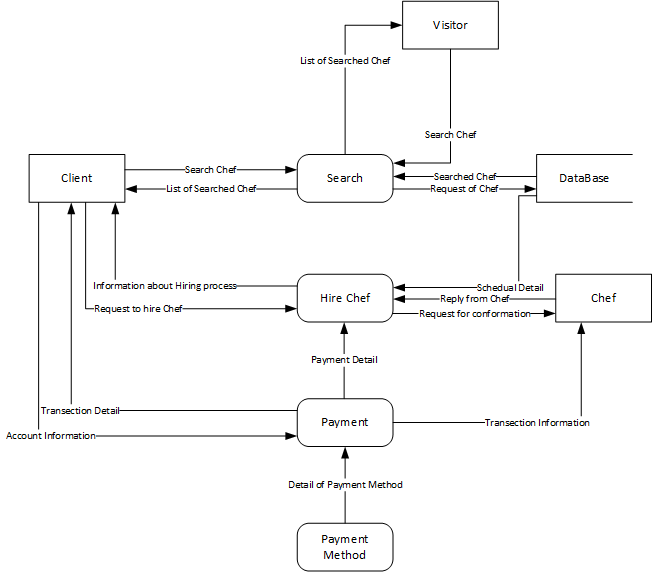


Figure 6: DFD Level 1 (Hiring a Chef)

**DFD L1 (Buying a Recipe)**

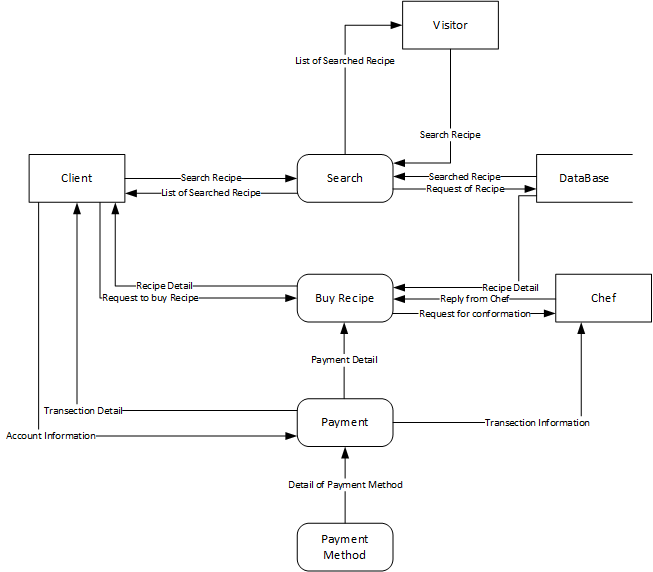


Figure 7: DFD Level 1 (Buying a Recipe)

## Data Flow Diagram Level 2

**DFD L2 (Login)**



**Figure 8: DFD Level 2 (Login)**

**DFD L2 (Chef)**



Figure 9: DFD Level 2 (Chef)

**DFD L2 (Recipe)**



Figure 10: DFD Level 2 (Recipe)

# System Design

.

## System Architecture Diagram

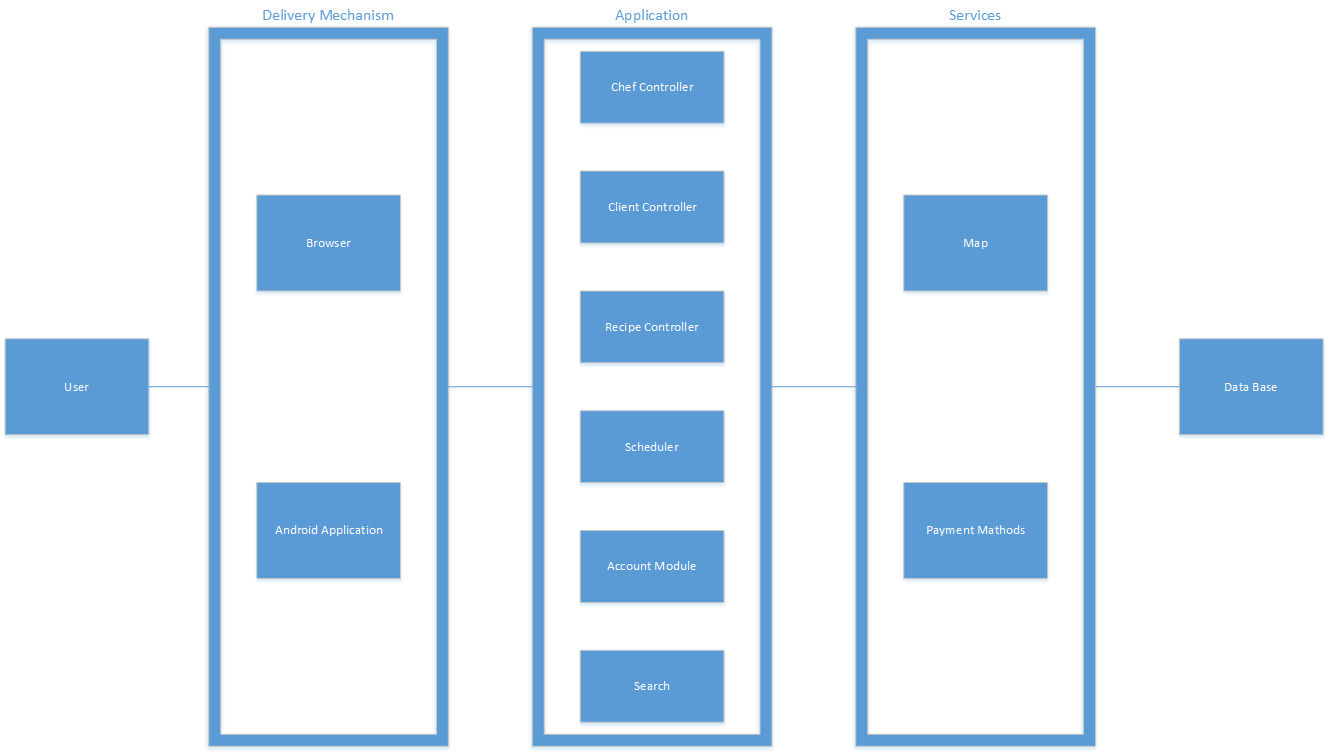


Figure 11: System Architecture Diagram

## Class Diagram

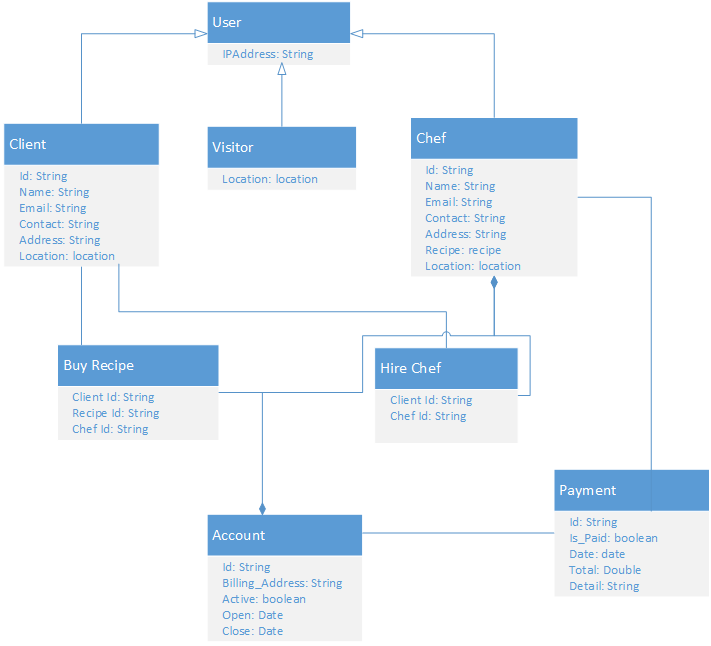


Figure 12: Class Diagram

## Collaboration Diagrams

## collabrationpng

## 

Figure 13: Collaboration Diagram

## Other UMLs

Sequence Diagram of Registration/Login:

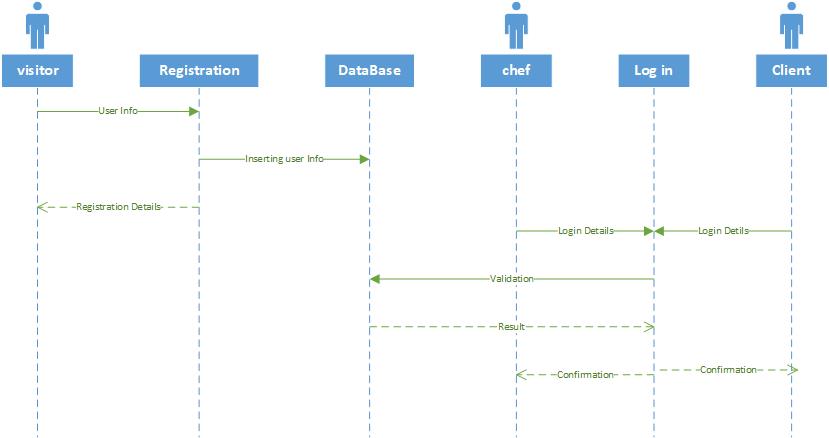


Figure 14: Sequence Diagram of Registration/Login

**Sequence Diagram of Hire a Chef:**

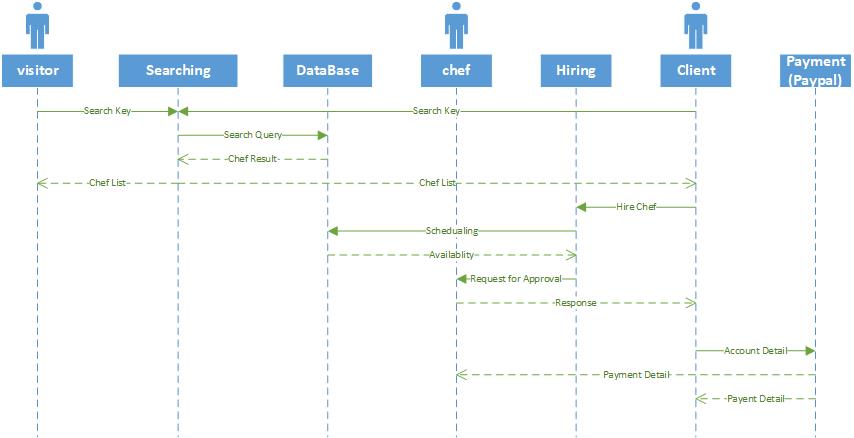
****

Figure 15: Sequence Diagram of Hire a Chef

**Sequence Diagram of Buy Recipe:**

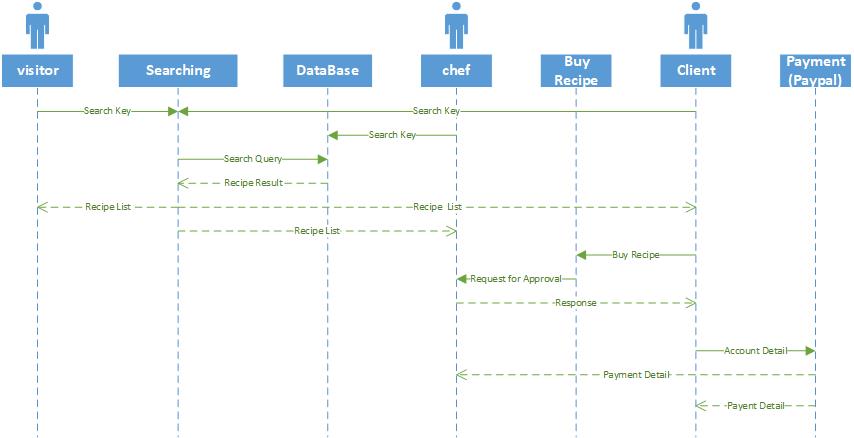


Figure 16: Sequence Diagram of Buy Recipe

**Sequence Diagram of Comments/Rating:**

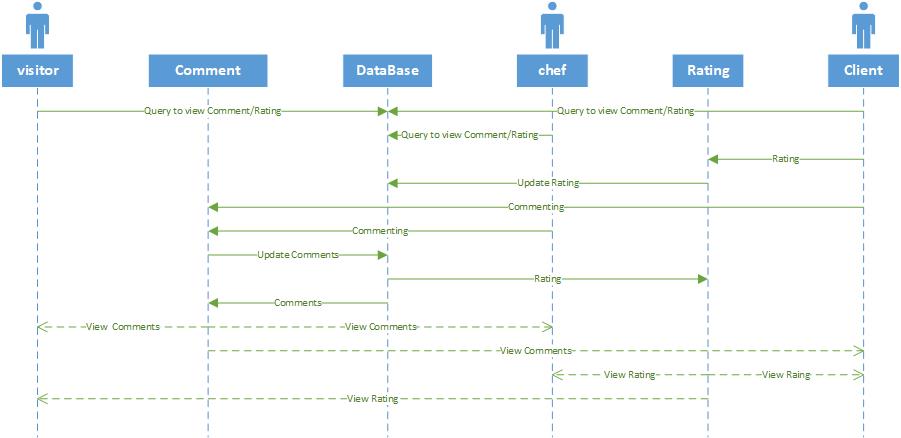


Figure 17: Sequence Diagram of Comment/Rating

## ERD

## 

Figure 16: ERD Diagram

## Data Dictionary

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Field Name** | **Data type** | **Constraint** | **Description** |
| 1 | User ID | Int | Primary Key | It stores Registration id. |
| 2 | User Name | Varchar (50) | Not Null | It stores Name. |
| 3 | User Password | Varchar (50) | Not Null | It stores user password |
| 4 | User email | Varchar (50) | Not Null | It stores user email. |
| 5 | User Phone | Varchar (50) | Not Null | It stores user phone number |
| 6 | Address | Varchar (50) | Not Null | It stores user address. |
| 7 | City | Varchar (50) | Not Null | It stores User City. |
| 8 | State | Varchar (50) | Not Null | It stores User State. |
| 9 | Zip | Int | Not Null | It stores City Zip. |
| 10 | Location | Text | Not Null | It saves user location |
| 11 | Image URL | Varchar | Not Null | It stores user image |
| 12 | User role | Varchar (50) | Not Null | It stores user role |

**Table 18: Data Dictionary**

# Implementation details

## Development Setup

* Php storm 2016.3.2, (Code igniter framework on server side and as well as for interface purpose. We use java as a language tool)
* Html, CSS, Bootstrap.
* JavaScript, JQuery, Jason.
* MySQL (Data Base to store Data)
* Net-Beans (compiler to build &execute the program)
* XAMP server.

## Deployment setup

* Web Browser.
* Web Server.

## Algorithms

#### Scheduling

Availabilty(obj, data[],modules)

query = this->get\_where(obj->chefId)

start\_new = time(obj->date.obj->start)

end\_new = time(obj->date.obj->end)

check = true

foreach query->result as row

old\_start = row->start

old\_end = row->end

if start\_new <= old\_end & old\_start >= end\_new)

check = false

return "Not Available"

else

check = true

if is\_check

data['chefId'] = obj->chefId

data['start'] = date(start\_new)

data['end'] = date(end\_new)

data['userId'] = obj->userId

data['status'] = 'pending'

this->load->module('Learner')

query = this->learner->get\_where(obj->userId)

foreach(query->result as row)

firstName = row->learnerFirstName

lastName = row->learnerLastName

requesteeId = row->userId

if this->get\_max()

data\_not['activity\_id'] = this->get\_max()

else

data\_not['activity\_id'] = '0'

this->load->module('Notifications')

this->notifications->\_insert(data\_not)

return "okay"

Accept(data[],modules)

data['status'] = 'approved'

this->\_update(\_GET['activityId'], data)

this->load->module("Notifications")

this->notifications->\_update(\_GET['notificationId'], dataNot)

schedule = this->get\_where\_id(\_GET['activityId'])

foreach(schedule->result as row)

startDate = row->start

endDate = row->end

chefId = row->chefId

this->load->module("Chef")

chef = this->chef->get\_where(chefId)

foreach chef->result as row

firstName = row->chefFirstName

lastName = row->chefLastName

randNo=random()

this->notifications->\_insert(dataNot)

Decline(data[],modules)

data['status'] = 'pending'

this->\_update(\_GET['activityId'], data)

this->load->module("Notifications")

this->notifications->\_update(\_GET['notificationId'], dataNot)

schedule = this->get\_where\_id(\_GET['activityId'])

this->load->module("Chef")

chef = this->chef->get\_where(chefId)

foreach chef->result as row

firstName = row->chefFirstName

lastName = row->chefLastName

this->notifications->\_insert(dataNotNew)

this->notifications->\_insert(dataNot)

## User Login

Login(data[],role)

data['role'] = role

Submit()

Submit(data[])

this->load->model("Mdl\_user")

if this->Mdl\_user->Pword\_check(data['username'], data['pword'])

this->logged\_in(data['username'])

else

return "Invalid Credentials!"

Pword\_check(data[], pword)

username = this->input->post('username' , TRUE)

result = this->mdl\_user->pword\_check(username, pword)

if (result == FALSE)

data[err\_message] = 'The username and/or password you entered is not correct'

return FALSE

else

logged\_in(data[user])

logged\_in(username)

query = this->db\_where\_custom('userName', username)

foreach(query->result as row)

user\_id = row->userId

username = row->userName

userRole = row->userRole

sessional[username] = username

sessional[userid] = userid

sessional[userrole] = userrole

if userRole == 'chef')

go to chef\_module

else

go to client\_module

## Constraints

### Assumptions

* Web portal shall be used frequently.
* Rules and regulations are predefined.
* User must be familiar with usage of online arena.

*.*

### System constraints

* System can’t be must faster, when 100 users are online at the same time.
* System can’t allow to add more than 500 recipes per chef in the system.

**Restrictions**

* + - Only chef can perfom the crud operations on recipe.
    - Only client can hire a chef.
    - Only client can comment a chef.
    - Only client can rate a chef.

### Limitations

* System does not provide the facility of chatting.
* System does not provide the facility of live guidance.

# Testing

## Extended Test Cases

|  |  |
| --- | --- |
| **TC: T1** | **Test Designed By: ZAIN** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 26-07-1 6 |
| **Module Name: login screen for client** | Test Executed By: BASIT |
| **Test Title: login** | Test Execution Date: 26-07-1 6 |
| **Description: Test the login** |  |
| **Pre-Condition: Client must be signup and register on the web portal** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to log in page |  |  |  | Pass |  | | **2** | Get client id | ahmed@gmail.com |  |  | | **3** | Get Password | ahmed |  |  | | **4** | Click on the login button |  | Client Should be able to login | Login Successful |   Post Condition: Client have account on use case and logged into account | |

**Table 23: TC 1**

|  |  |
| --- | --- |
| **TC ID: T2** | **Test Designed By: ZAIN** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 26-07-1 6 |
| **Module Name: login screen for chef** | Test Executed By: BASIT |
| **Test Title: login** | Test Execution Date: 26-07-1 6 |
| **Description: Test the login** |  |
| **Pre-Conditions: Chef must be signup and register on the web portal** |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go TO log in page |  |  |  | Pass |  | | **2** | Get chef id | zain@zain.com |  |  | | **3** | Get Password | Zain123 |  |  | | **4** | Click on the login button |  | Client Should be able to login | Login Successful |   Post Condition: Chef have account on use case and logged into account | |

**Table 24: TC 2**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Case ID: T3** | | | | **Test Designed By: Abdul Baist** | | | |
| **Test Priority (Low\Medium\High): med** | | | | Test Designed Date: 26-07-1 6 | | | |
| **Module Name: Sign up** | | | | Test Executed By: Zain | | | |
| **Test Title: Sign up validation** | | | | Test Execution Date: 26-07-1 6 | | | |
| **Description: Test the sign up** | | | |  | | | |
| **Pre-Conditions: check text field** | | | |  | | | |
| **Step** | **Test Step** | **Test Data** | **Expected Result** | | **Actual Result** | **Status** | **Notes** |
| **1** | Go to signup page |  |  | |  | Pass |  |
| **3** | Get user name | Basit@gmail.com |  | |  |
| **4** | Get password | 123456 |  | |  |
| **5** | Get Location | Lahore |  | |  |  |
| **8** | Click on submit button |  | Successful login | | Login Successful |
| Post Condition: Visitor now become a client | | | | | | | |
| **Table 25: TC 3** | | | | | | | |

|  |  |
| --- | --- |
| **TC ID: T4** | **Test Designed By: Zain** |
| **Test Priority (Low\Medium\High): High** | Test Designed Date: 26-07-1 6 |
| **Module Name: Search Recipe** | Test Executed By: **Basit** |
| **Test Title: Searching** | Test Execution Date: 26-07-1 6 |
| **Description: Test the search bar** |  |
|  |  |
| **Pre-Conditions: User enter in the search field** |  |
| **Dependencies:** |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go the recipe search bar. |  |  |  | Pass |  | | **2** | Enter recipe name |  |  |  | |  | | | |  | | **3** | Click on search button |  | Recipe found | Showed List of Recipes. | | |

**Table 26: TC 4**

|  |  |
| --- | --- |
| **Test Case ID: T5** | **Test Designed By: Zain Mohsin** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 26-07-1 6 |
| **Module Name: Search Chef** | Test Executed By: Abdul **Basit** |
| **Test Title: Searching** | Test Execution Date: 26-07-1 6 |
| **Description: Test the search bar** |  |
|  |  |
| **Pre-Conditions: User enter in the search field** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go the Chef search bar. |  |  |  | Pass |  | | **2** | Enter Chef name |  |  |  | |  | | | |  | | **3** | Click on search button |  | Chef found | Showed List of Chefs | | |

**Table 27: TC 5**

|  |  |
| --- | --- |
| **TC ID: T6** | **Test Designed By: Abdul Basit** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 08-07-16 |
| **Module Name: Comment** | Test Executed By: Zain |
| **Test Title: Commenting** | Test Execution Date: 08-07-16 |
| **Description: Test the comment box** |  |
|  |  |
| **Pre-Conditions: Other user comments must show into the comment box** |  |
| **Dependencies: acceptance of request** |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to the comment box screen |  |  |  | Pass |  | | **2** | Write message | Text of message |  |  | | **3** | Click on comment button |  | Comment will be sent successfully | Comment Send |   Post condition: Successfully sent Comment | |

**Table 28: TC 6**

|  |  |
| --- | --- |
| **TC ID: T7** | **Test Designed By: Zain Mohsin** |
| **Test Priority (Low\Medium\High): High** | Test Designed Date: 08-07-16 |
| **Module Name: Hire/Book a Chef** | Test Executed By: Abdul Basit |
| **Test Title: Hiring** | Test Execution Date: 08-07-16 |
| **Description: Test the Chef for hiring.** |  |
|  |  |
| **Pre-Conditions: Client must be logged in.** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to the Chef profile for hiring |  |  |  | Pass |  | | **2** | Click on hire a chef to start the session. |  |  |  | | **3** | Select date and time for hiring | Month/Date/ Time | Session saved |  | | **4** | Send notification to the chef for approval |  | Chef approved | Chef arrived |  |  |   Post condition: Successfully Hire a Chef | |

**Table 29: TC 7**

|  |  |
| --- | --- |
| **TC ID: T8** | **Test Designed By: Zain Mohsin** |
| **Test Priority (Low\Medium\High): High** | Test Designed Date: 08-07-16 |
| **Module Name: Add recipe** | Test Executed By: Abdul Basit |
| **Test Title: Recipe addition** | Test Execution Date: 08-07-16 |
| **Description: Test to add recipe** |  |
|  |  |
| **Pre-Conditions: Chef must be logged in.** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to the Chef profile for adding recipe |  |  |  | Pass |  | | **2** | Click on the upload recipe |  |  |  | | **3** | Fill detail for recipes to add | Picture and other detail against recipe added | Details saved |  | | **4** | Click on upload and publish |  | Recipe uploaded | Recipe added |  |  |   Post condition: Successfully recipe is added | |

**Table 30: TC 8**

|  |  |
| --- | --- |
| **TC ID: T9** | **Test Designed By: Zain Mohsin** |
| **Test Priority (Low\Medium\High): High** | Test Designed Date: 08-07-16 |
| **Module Name: Edit recipe** | Test Executed By: Abdul Basit |
| **Test Title: Recipe edition** | Test Execution Date: 08-07-16 |
| **Description: Test to edit the recipe** |  |
|  |  |
| **Pre-Conditions: Chef must be logged in.** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to the Chef profile for edit recipe |  |  |  | Pass |  | | **2** | Click on the manage recipe |  |  |  | | **3** | Click on edit | Enter details accordingly | Details saved |  | | **4** | Fill the details and click on save |  | Recipe saved | Recipe added |  |  |   Post condition: Successfully recipe edited | |

**Table 31: TC 9**

|  |  |
| --- | --- |
| **TC ID: T10** | **Test Designed By: Zain Mohsin** |
| **Test Priority (Low\Medium\High): High** | Test Designed Date: 08-07-16 |
| **Module Name: Rate a recipe** | Test Executed By: Abdul Basit |
| **Test Title: Rating** | Test Execution Date: 08-07-16 |
| **Description: Test to rate the recipe** |  |
| **Pre-Conditions: Client must be logged in.** |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **StepS** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to the recipe list |  |  |  | Pass |  | | **2** | Select recipe to rate | Drag mouse to rate the recipe |  |  | | **3** | Click on the start to rate | Fill stars accordingly | Rated stars filled |  |   Post condition: Successfully recipe rated | |

**Table 32: TC 10**

|  |  |
| --- | --- |
| **TC ID: T11** | **Test Designed By: Basit** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 08-07-16 |
| **Module Name: Rate a Chef** | Test Executed By: Zain |
| **Test Title: Rating** | Test Execution Date: 08-07-16 |
| **Description: Test to rate the chef** |  |
|  |  |
| **Pre-Conditions: Client must be logged in** |  |
| **Dependencies: acceptance of request** |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Steps** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to the chef list |  |  |  | Pass |  | | **2** | Select chef to rate | Drag mouse to rate the recipe. |  |  | | **3** | Click on the star to rate. |  | Stars filled accordingly. | Rated stars filled. |   Post condition: Successfully chef rated. | |

**Table 33 TC 11**

|  |  |
| --- | --- |
| **TC ID: T12** | **Test Designed By: Basit** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 08-07-16 |
| **Module Name: Update profile** | Test Executed By: Zain |
| **Test Title: update** | Test Execution Date: 08-07-16 |
| **Description: Test to update profile** |  |
|  |  |
| **Pre-Conditions: Client must be logged in** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Steps** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to login |  |  |  | Pass |  | | **2** | Select update profile | Edit profile |  |  | | **3** | Manage changes. |  | Update profile changes | Changes updated |   Post condition: Successfully handle password changes. | |

**Table 34: TC 12**

|  |  |
| --- | --- |
| **TC ID: T13** | **Test Designed By: Abdul Basit** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 08-07-16 |
| **Module Name: Minimum Time for search** | Test Executed By: Zain Moshin |
| **Test Title: Minimum search** | Test Execution Date: 08-07-16 |
| **Description: Test to check system response time** |  |
|  |  |
| **Pre-Conditions: Client must be logged in** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Steps** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to search |  |  |  | Pass |  | | **2** | Select the search bar | Enter the query |  |  | | **3** | Shows results |  | Show result within 25 second | Results shown within 25 seconds |   Post condition: Successfully handle search completed | |

**Table 34 TC 13**

|  |  |
| --- | --- |
| **TC ID: T14** | **Test Designed By: Abdul Basit** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 08-07-16 |
| **Module Name: Client credentials should be secured** | Test Executed By: Zain Mohsin |
| **Test Title: Security** | Test Execution Date: 08-07-16 |
| **Description: Test to secure the client credential** |  |
|  |  |
| **Pre-Conditions: Client must be sing up page** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Steps** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to sign up page |  |  |  | Pass |  | | **2** | Select the credentials | Enter data on credentials bar |  |  | | **3** | Manage to type data. |  | Secure the credentials | Credentials secured |   Post condition: Successfully handle credentials secured | |

**Table 32 TC 14**

|  |  |
| --- | --- |
| **TC ID: T15** | **Test Designed By: Abdul Basit** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 08-07-16 |
| **Module Name: Access to required chefs** | Test Executed By: Zain Moshsin |
| **Test Title: Access Efficiency** | Test Execution Date: 08-07-16 |
| **Description: Test to check the efficiency** |  |
|  |  |
| **Pre-Conditions: Client must be logged in** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Steps** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to login |  |  |  | Pass |  | | **2** | Select the chat option | Type on chat window |  |  | | **3** | Manage to type the message |  | Send the message to chef | Message sended |   Post condition: Successfully handle the message sending | |

**Table 33: TC 15**

|  |  |
| --- | --- |
| **TC ID: T16** | **Test Designed By: Abdul Basit** |
| **Test Priority (Low\Medium\High): med** | Test Designed Date: 08-07-16 |
| **Module Name: Interface must be efficient** | Test Executed By: Zain Moshin |
| **Test Title: Usability** | Test Execution Date: 08-07-16 |
| **Description: Test to check the efficiency of interface** |  |
|  |  |
| **Pre-Conditions: Client must be on the portal** |  |
|  |  |
| |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Steps** | **Test Step** | **Test Data** | **Expected Result** | **Actual Result** | **Status** | **Notes** | | **1** | Go to login |  |  |  | Pass |  | | **2** | Select update profile | Edit profile |  |  | | **3** | Manage changes. |  | Update profile changes | Changes updated |   Post condition: Successfully handle password | |

**Table 34: TC 16**

## 

## Decision Table

### Code snippet

### Decision coverage table

**Signup**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **R U L E S** | | | | | | | | | | | | | | | |
| **Conditions** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | |
| **Name (a)** | T | T | T | T | T | T | T | T | F | F | F | F | F | F | F | F | |
| **Password (b)** | T | T | T | T | F | F | F | F | T | T | T | T | F | F | F | F | |
| **City (c)** | T | T | F | F | T | T | F | F | T | T | F | F | T | T | F | F | |
| **Phone (d)** | T | F | T | F | T | F | T | F | T | F | T | F | T | F | T | F | |
| **Actions** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | |
| **Reports** | n | a+b+c | a+b+d | a+b | a+c+d | a+c | a+d | a | b+c+d | b+c | b+d | b | c +d | c | d | n | |

**Table 39: Decision Coverage Table 1**

**For Comment**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **R U L E S** | | | | | | | |
| **Conditions** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **Name(P)** | T | T | T | T | F | F | F | F |
| **Password(S)** | T | T | F | F | T | T | F | F |
| **Text(T)** | T | F | T | F | T | F | T | F |
| **Actions** |  |  |  |  |  |  |  |  |
| **Notifications** | N | P+S | P+T | P | S+T | S | T | N |

**Table 40: Decision Coverage Table 2**

**Search a Chef**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **R U L E S** | | | | | | | |
| **Conditions** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **Chef Name(c)** | T | T | T | T | F | F | F | F |
| **List of Chefs (l)** | T | F | T | F | T | F | T | F |
| **Actions** |  |  |  |  |  |  |  |  |
| **Reports** | N | C | N | C | L | N | L | N |

**Table 41: Decision Coverage Table 3**

**Search for Recipe**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **R U L E S** | | | | | | | |
| **Conditions** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **Recipe name(R)** | T | T | T | T | F | F | F | F |
| **List of Recipe (l)** | T | F | T | F | T | F | T | F |
| **Actions** |  |  |  |  |  |  |  |  |
| **Reports** | N | R | N | R | L | N | L | N |

**Table 42: Decision Coverage Table 4**

## Traceability Matrix

### Prototypes (UCID vs RID)RID

|  |  |
| --- | --- |
| **UCID** | **Requirements ID** |
| UC 1 | 1 |
| UC 2 | 2 |
| UC 3 | 2,4 |
| UC 4 | 6 |
| UC 5 | 10 |
| UC 6 | 5 |
| UC 7 | 5 |
| UC 8 | 8 |
| UC 9 | 8 |
| UC 10 | 11 |
| UC 11 | 12 |
| UC 12 | 9 |
| UC 13 | 13 |
| UC 14 | 14 |
| UC 15 | 15 |
| UC 16 | 16 |

**Table 43: Traceability Matrix 1**

### Prototypes (RID vs PID)ID

|  |  |
| --- | --- |
| **RID** | **Prototype ID** |
| RID 1 | Figure 1,3,6 |
| RID 2 | Figure 2 |
| RID 3 | Figure 4 |
| RID 4 | Figure 4 |
| RID 5 | Figure 5 |
| RID 6 | Figure 5 |
| RID 7 | Figure 4 |
| RID 8 | Figure 4 |
| RID 9 | Figure 8 |
| RID 10 | Figure 7 |
| RID 11 | Figure 8 |
| RID 12 | Figure 8 |
| RID 13 | Figure 3 |
| RID 14 | Figure 2 |
| RID 15 | Figure 4 |
| RID 16 | Figure 1 |

### 

**Table 44: Traceability Matrix 2**

### Test Cases (RID vs TID)

|  |  |
| --- | --- |
| **TID** | **RID** |
| TID 1 | 2 |
| TID 2 | 4 |
| TID 3 | 3 |
| TID 4 | 5 |
| TID 5 | 6 |
| TID 6 | 9 |
| TID 7 | 9 |
| TID 8 | 8 |
| TID 9 | 8 |
| TID 10 | 12 |
| TID 11 | 11 |
| TID 12 | 8,7 |
| TID 13 | 13 |
| TID 14 | 14 |
| TID 15 | 15 |
| TID 16 | 16 |

**Table 45: Traceability Matrix 3**

### Coverage (UCID vs TID)

|  |  |
| --- | --- |
| **TID** | **UCID** |
| TID 1 | 2 |
| TID 2 | 2 |
| TID 3 | 1 |
| TID 4 | 5 |
| TID 5 | 4 |
| TID 6 | 12 |
| TID 7 | 4 |
| TID 8 | 6 |
| TID 9 | 8 |
| TID 10 | 11 |
| TID 11 | 10 |
| TID 12 | 3 |
| TID 13 | 13 |
| TID 14 | 14 |
| TID 15 | 15 |
| TID 16 | 16 |

**Table 46: Traceability Matrix 4**

# Results/Output/Statistics

## % Completion

Total number of requirements = 16

Total number of use cases covering all requirements = 16

Completion = (16/16)\*100 = 100%

## % Accuracy

Total number of test cases = 16

Total number of requirements covering all text cases= 16

Accuracy = (16/16)\*100 = 100%

## % Correctness

Total number of use cases = 16

Total number of test cases = 16

Correctness = (16/16)\*100 = 100%

# Conclusion

**On completion of this project:**

* System will user friendly and easy to use.
* View recipes: User can able to view different recipes and their ingredients.
* View Chefs: user can able to view different chefs.
* Buy recipes: Client can able to buy recipes as well.
* Hire chefs: Client can book a chef.
* We can able to search chefs.
* We can search recipes.
* We can find the exact location of a chef.

# Future work

**We are planning to increase the scope of the project and also advertise this product**

* Advertise it to generate the profit
* Making of android application
* Making of iOS application

# Bibliography

Use IEEE or ACM format for citations

## Books

None

## Journals

None

## Articles

None

## Research papers

None

## Other References

Other chef’s websites are also available but they are not like Chefu

# Appendix

## Glossary of terms

|  |  |
| --- | --- |
| CHEFU | Chefu is our application name. |
| DFD | A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system modelling its process aspects. |
| SAD | Software architecture Diagram (SAD) involves the high-level structure of software system abstraction, by using decomposition and composition, with architectural style and quality attributes. |
| UCID | Use Case Identity |
| RID | Requirement identity |
| PID | Prototype identity |
| TID | Test case Identity |
| ERD | An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities. |

## Pre-requisites

We are using following tools and resources to develop our online system.

1. CodeIgniter: It is a PHP framework using MVC to develop. We customized the layout and styling to match the front-end interface based on our designs.
2. All the code written for the system is in PHP, JavaScript, jQuery etc.
3. Is used to provide the basic front end out layer for the system which is in HTML.
4. All color and design are given to website using CSS.
5. MySQL Server is used to save all data of the system.
6. JavaScript is used for quick response to user in the front end.
7. JQUERY is JavaScript library.