



Course ID: CPS5995

Project - CookBook

**Software Design Document (SDD)
Version 1.0**

Prepared by: Dharmay Prajapati

Reviewed by : Jose Serra

Instructor : Dr. Jing-Chiou Liou

Date:18Feb2023

Table of Contents

1 . Introduction.....	3	1.1
Purpose.....	3	1.2
Scope.....	3	1.3
Goals and Objectives.....	3	1.4
Acronyms and Abbreviations.....	4	
Definitions and Acronyms.....	4	
2. System Design.....	4	2.1
System Architecture.....	4	2.2
Context Diagram.....	6	2.3
Use case Diagram.....	6	
3. System Functional Model.....	10	3.1
Representation of functional Modules	10	3.2
Software Process Model	10	
4. Data Design.....	11	4.1
Database Schema Diagram.....	11	
5. Design Constraints, Restrictions and Limitations.....	14	5.1
Design Constraints	14	5.2
System Restrictions and Limitations	14	
6. Tools and References.....	14	6.1
Tools used to create Diagrams	14	6.2
Documentation Tool.....	14	6.3
Reference Materials.....	14	

1. Introduction

This software design document is **meant** to provide a low-level description of the CookBook Network, providing insight into the structure and design of the intended system. This document provides the narrative and graphical representation of the software design including System Architecture, use-case diagrams and other supporting requirement information.

1.1 Purpose

The purpose of this software design documentation is to provide the initial-sketch of the intended software that leads to start with the development of the software. Since this project is carried on using Agile methodology (which will be explained in the upcoming section), changes may occur during the development stage as per the consumer feedback and System Engineer's decision based on management objectives.

1.2 Scope

This Software Design Document is for a base level system which will work as a proof of concept for the use of building a system that provides a base level of functionality. This Software Design is focused on the base level system and critical parts of the system which can be improved later.

The following are the details provided in this document.

- System Architecture
- Context Diagram
- Use-case Diagrams
- Representation of functional modules
- Database schema Diagram

1.3 Goals and Objectives

This project is meant to provide an independent CookBook which has its own database and access control that facilitates Engage users in cooking at home behaviors through a web application that will gather information from local food markets, and user input of at home cooking inventory to deliver recipe recommendations, and retail resources for purchasing ingredients to cook meals.

To create an online cooking social platform that promotes growth/interest/engagement of culinary skills/needs within the average American by providing quick access to information, a friendly-approachable environment, and an optimal UX.

3

Cookbook Network's target audience is those seeking to reduce disposable income on processed food with high sodium, trans fat, and other unhealthy nutrients. Also our target is to reduce the percentage from 28% to 0% who do not know how to cook. This system will help to increase customer saving by 10% and aim to reduce food waste by 20%.

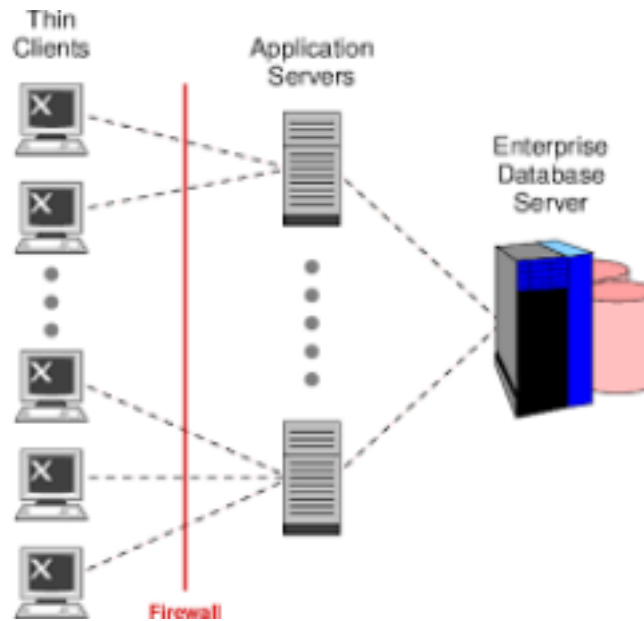
1.4 Acronyms and Abbreviations

- SDD - Software Design Document
- CBN - CookBook Network
- UI - User Interface
- PHP - Personal Home Page, a server-side scripting Language
- HTML - HyperText Markup Language, for web browser UI
- CSS - Cascading Style Sheets, for UI

2. System Design

2.1 System Architecture

Being a web application, this system follows a typical client-server model with three-tier architecture. Three-tier architecture is a well-established software application architecture that organizes applications into three logical and physical computing tiers: the presentation tier, or user interface; the application tier, where data is processed; and the data tier, where the data associated with the application is stored and managed.



4

Figure 1 Three Tier Architecture

- The Presentation tier - the part of the application which is visible to the user; it enables the input of requirements and the presentation of results. It is dependent on the platform (e.g. web applications, Windows applications, Android applications, etc.). It may therefore be different for different devices or platforms.
- The Application tier (also functional) - the middle layer of the model (middleware), assures the calculations and operations performed between input-output requirements and data. Also known as the application server.
- The Data tier (also database) - the lowest layer of the model, ensures all operations with data, i.e. database management system and basic data-base operations for functional storage, selection, aggregation, processing, integrity, and data audit.

Architecture of CBN

Technical details behind the layers

Presentation layer : HTML and CSS are used for coding. Any web browser can serve as a presentation layer. Eg. Google Chrome, Mozilla Firefox, Safari, Edge

Application layer : PHP is used for coding. Web Server is eve.kean.edu

Data tier : Mysql Database. Database Server is imc.kean.edu

This architectural model and languages are chosen to ensure better security of the software. For example, PHP is a server-side scripting language that hides the business logic/work-flow to the presentation layer which presents only the content that the user is authorized to view. In addition, only the application layer can communicate with the database and retrieve the requested data to provide it to the presentation layer.

2.2 Context diagram :

This diagram represents the entities which interact with the OES system.

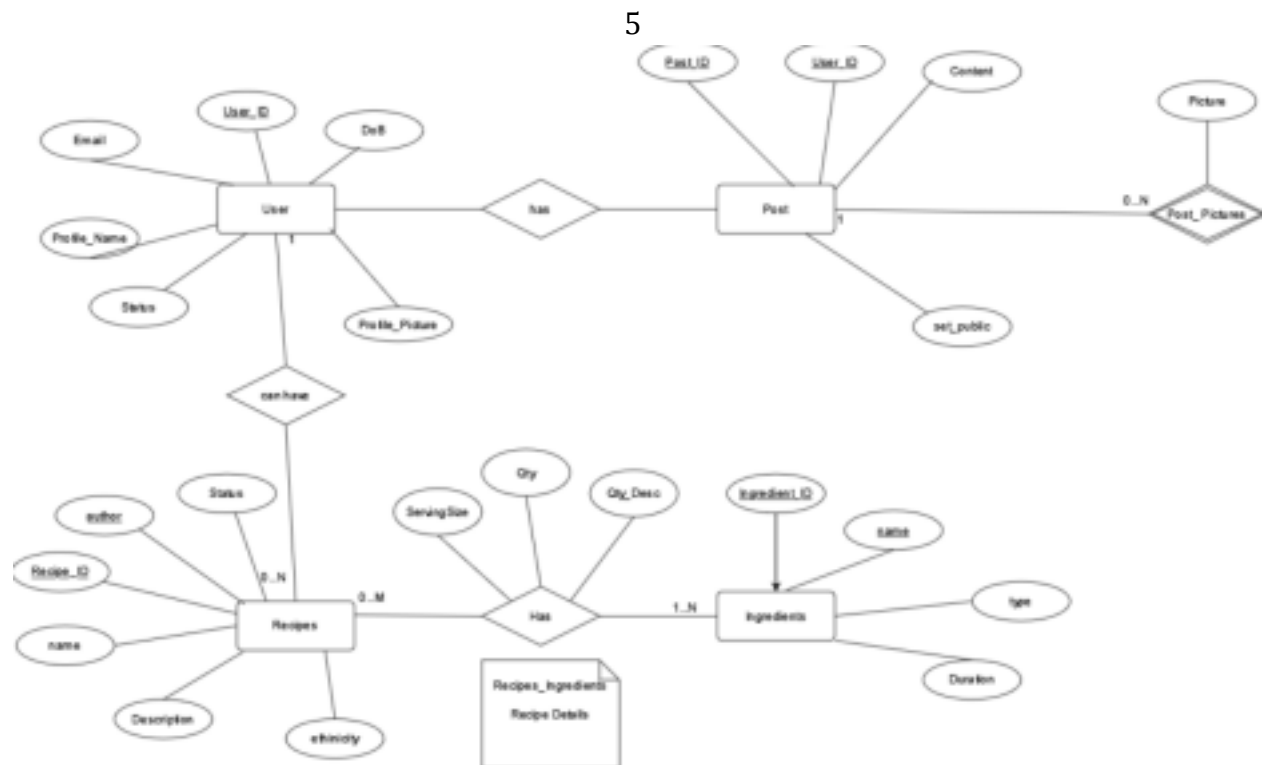


Figure 3 Context Diagram of CBN

Admin : Admin can create and maintain the accounts of Test conductors. Also there is only one admin account whose credentials will be shared with the system support people(via document) and therefore no user interface provided to alter the credentials of this account which can be done only by the system support people via back end.

Test Conductor : Test conductor can create and maintain the accounts of Students. He can create and conduct the test for the respective students. He can view the respective test results. He can change his password.

Student : Student can attend the designated test for him and view the result in case of automated evaluation. He can change his password.

2.3 Use case Diagram

A Use case Diagram is a pictorial representation of a user's interaction with the system. People who interact with the intended software system are called Actors and represented by human symbols. Use cases are represented by an eclipse symbol and the software system is represented by a rectangle.

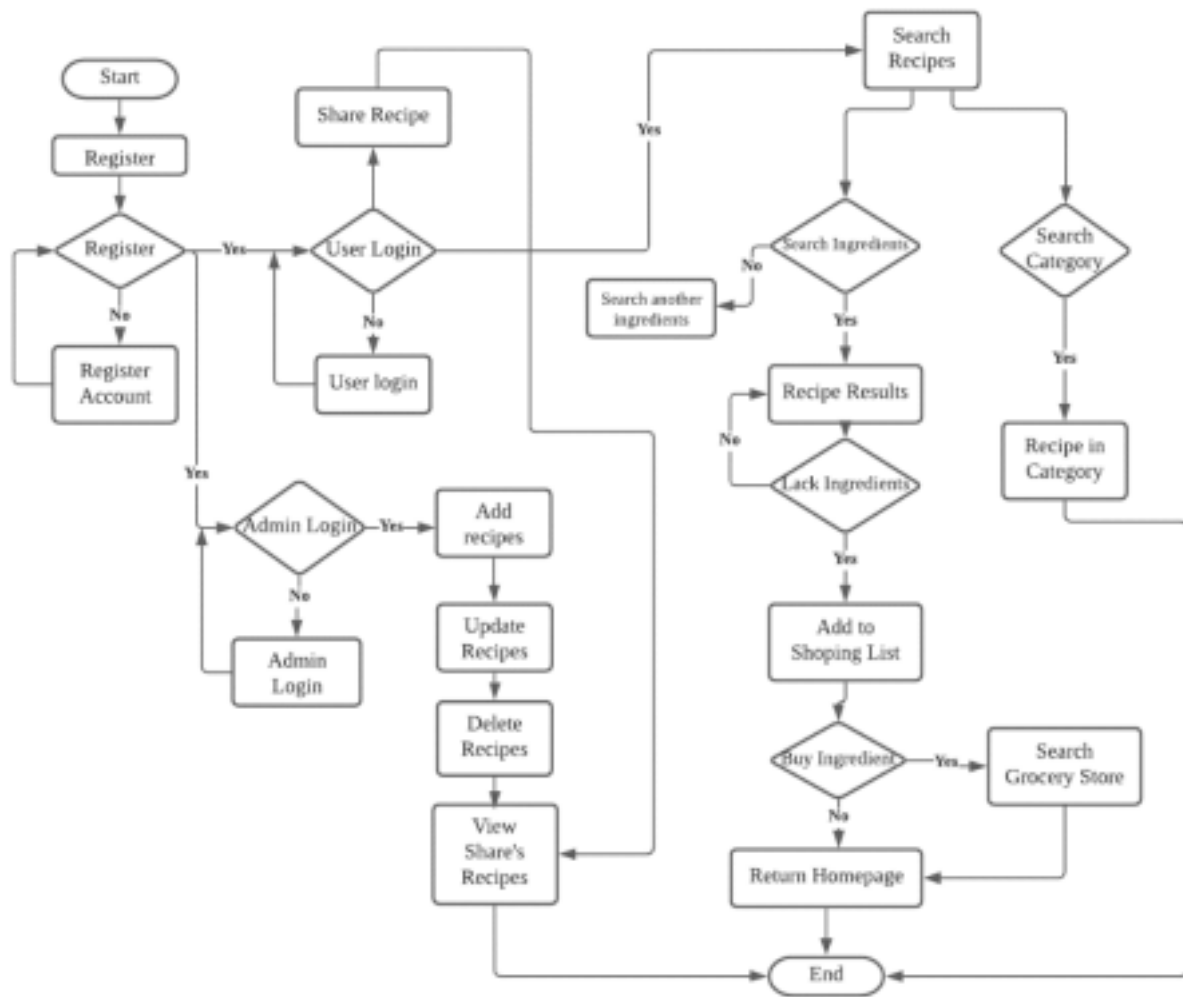


Figure 5 Data Flow Diagram for CBN

3.1 Representation of functional Modules:

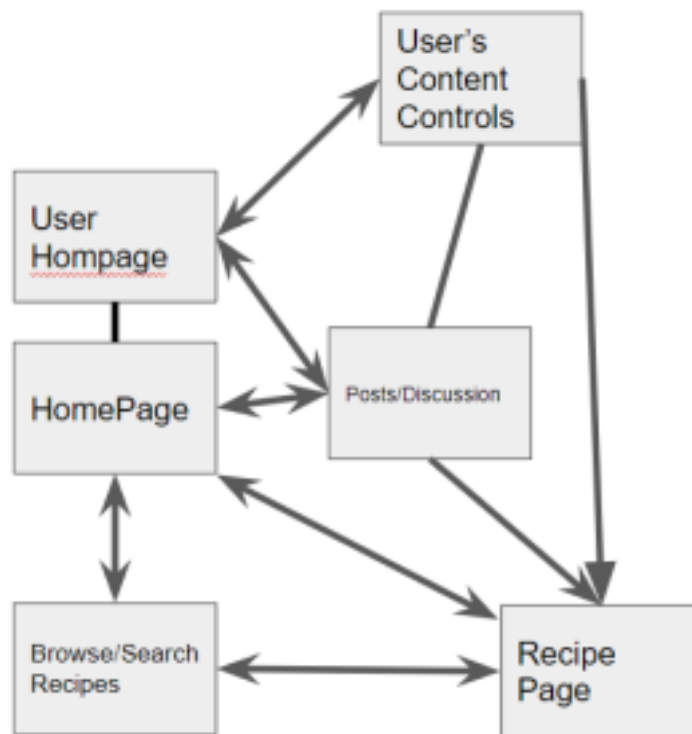


Figure 7 Representation of functional Modules of CookBook

3.2 Software Process Model

Agile methodology is followed to develop this software system as it is well-suited for this type of web application. In addition, this process model provides a working software at any stage with room for improvements.

Agile methodology is also called iterative/incremental approach where the main purpose of the intended system is identified and initial sketch of software design is planned. Starting with the basic requirements development work will be started and after completion of every deliverable, the same is presented to get client feedback for any change requirement. This is preceded by the next level of requirements in further deliverables. Every deliverable is integrated with the previous deliverable and tested and hence providing a working software with added functionalities at every iteration.

Here is the illustration of this process model.

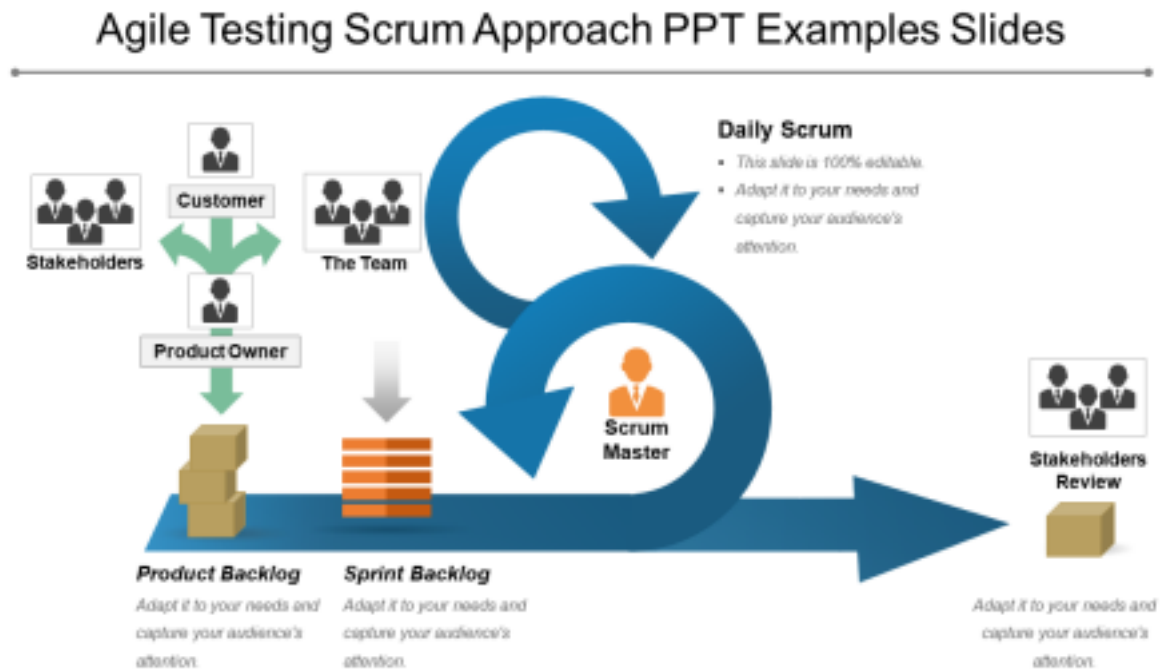


Figure 8 Agile - Software Process Model

This system is planned to be developed in three sprints/deliverables.

Sprint 1 :

- Front-End Design of sign-up, login pages, and landing pages of the project.
- Test Case and Plan developed for the Front-End Design

Sprint 2 :

- Back-End Design of user login pages
- Test Case and Plan developed for Back-End Design

Sprint 3 :

- Summary report feature
- Final Test Case and Plan is developed for full project

4. Data Design

4.1 Database Schema Diagram :

A database schema defines its entities and the relationship among them. It contains a descriptive detail of the database, which can be depicted by means of schema diagrams.

Mysql Database is used in this system. All Database tables are structured to conform to the third normal form to ensure a better database structure.

The Database schema diagram represented here is based on the initial design of the software and therefore might be changed/updated in forth-coming iterations as per agile methodology.

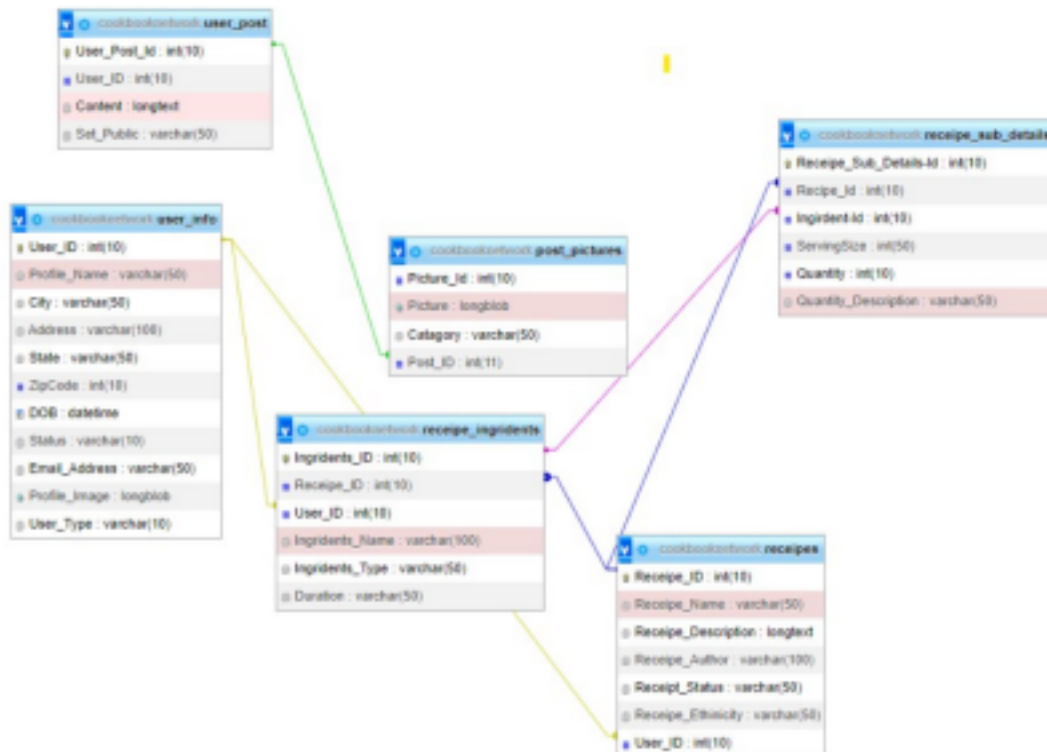


Figure 9 Database Schema Diagram of CBN

5. Design constraints, Restrictions, limitations

5.1 Design Constraints :

Being a short time project of 3 months, this System is designed in two weeks to facilitate the team with the basic sketch of the intended software system.

5.2 System Restrictions and Limitations

This software system is a web application and therefore needs internet service, client system (for user interaction) , web server (to deploy the software) and a database server (to store the application data).

- This System is meant for people with basic knowledge on computer usage.

6. Tools and References

6.1 Tools used to create Diagrams

- Online tool, https://creately.com/app/?tempID=gc7qvpsj1&login_type=demo# is used to create Use-case Diagrams
- Database schema Diagram is generated using Mysql workbench.
- General conceptual Diagrams like Agile process models are obtained from web resources.
- Lucid chart is used to create other project specific pictorial representations.

6.2 Documentation Tool

- Google Docs is used to create this documentation.

6.3 Reference Materials

- MyFridgeFood
- Zomato Case Studies
- Bhoj Deal Success Stories
- Panda services
- Internal as well as Internet research