**CHAPTER THREE**

**Methodology and Design**

**3.1 Introduction**

A methodology is an approach to rigorous study or investigation, particularly to uncover new facts or information; hence, research methodology should be good enough to make the attainment of the established objectives attainable with certain components, such as methods of data collecting and design. This chapter includes the input/output specifications and system requirements for the development of a food recipe system, as well as the system modeling (use case, activity, and class diagrams).

**3.2 Methods of Data Collection**

It is crucial to acquire data and facts about the current system before implementing any system since one has to understand what is happening. Two techniques were used to conduct this study.

1. Primary
2. Secondary

**3.2.1 Primary Source of Information**

This comprises information that is collected directly or indirectly from target users without any alterations or ideas from other authors. The information from this primary source is deemed more accurate and reliable. Hence, the aim is to assimilate the information gathered from this source into the project in order to meet requirements. The chosen fact-finding techniques for the primary source data gathering are: interview and observation

**3.2.2 Secondary Source of Information**

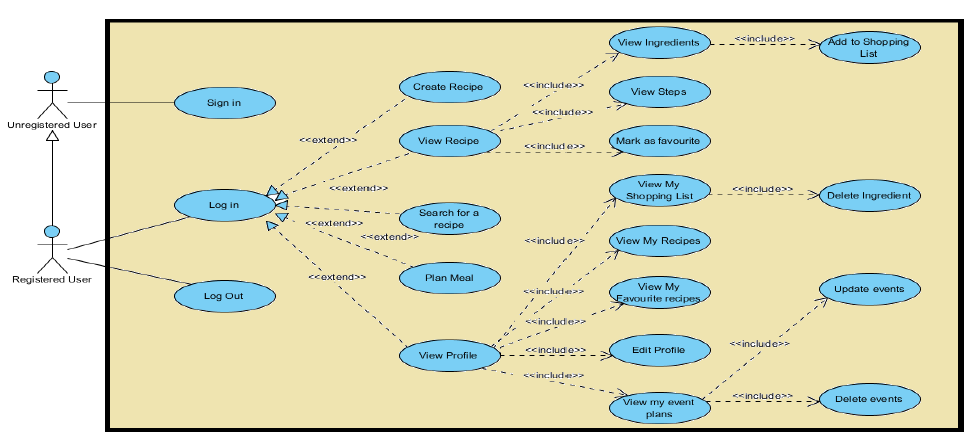
This basically comprises the totality of information someone is able to obtain from existing sources such as books, the internet, case study, articles, newsletter, and other valuable publications. The resources gathered from the internet specifically have been very relevant, various search engines especially Google made information finding very easy.

**3.3 System Modeling**

A system model is a conceptual model that describes and represents a system. Any interaction between a group of components that work together to accomplish a single goal is referred to as a system. Visual models of the object-oriented software-intensive systems can be made using a set of graphic notation techniques that are part of the Unified Modeling Language, which is employed in this modern system design. Use case diagrams, class diagrams, and activity diagrams are among the UML diagrams used in this new design.

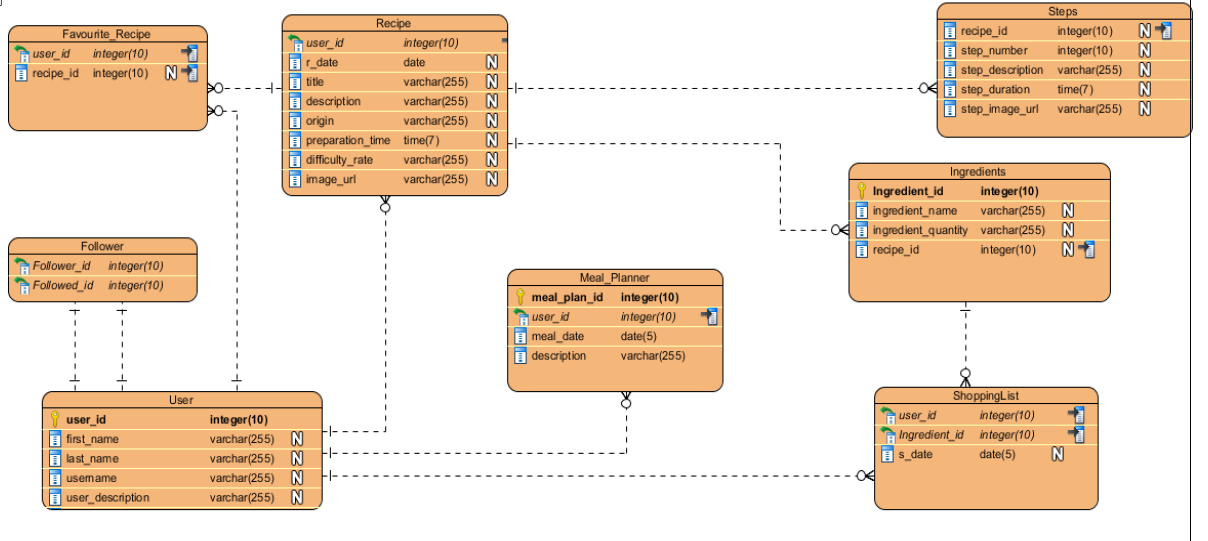
**3.3.1 Use Case Diagrams**

Use cases are collections of interactions between systems and users. Use case diagrams are used to visually summarize a system's functionality in terms of its actors, its goals (represented as use cases), and any dependencies between those use cases.



**Fig 3.1 System Use Case Diagram**

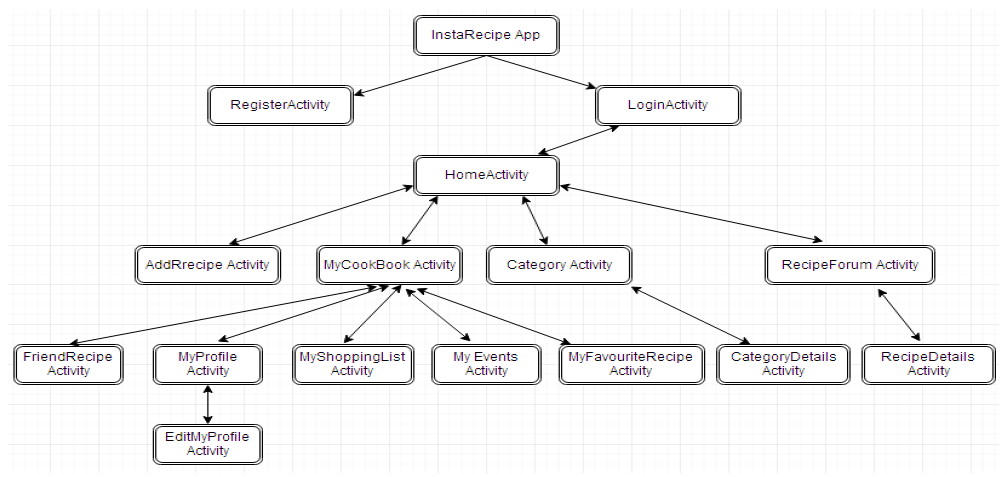
**3.3.2 Class Diagrams**

The Unified Modeling Language (UML) class diagram is an implementation of an independent view of how the system interface will be, with each class having its own properties and illustrating how they interact with one another. Class diagrams use the rules established by the Unified Modeling Language to visually depict the static structure and composition of a specific system (UML).

**Fig 3.2 System Class Diagram**

**3.3.3 Activity Diagrams**

Similar to a flowchart or a data flow diagram, an activity diagram visually depicts a sequence of events or the flow of control in a system, but it functions more like an advanced version of both.



**Fig 3.3 Activity Diagram**

**3.4 Database Design**

The logical explanation of how data is kept in the computer's memory is called input specification. The freedom experienced in using the system, as well as the convenience of retrieving and reading the data and assuring applicability across the internet, make SQL standards essential for ensuring that structured data is uniform and independent of applications. Some of the input specifications employed in this project work are presented below.

1. Users Table: contains basic information about all system users.
2. Recipe Table: contains every system recipe information.

**Table 3.1 Users Table input specification table**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| Email | String | 150 | Email for login (case sensitive) |
| Password | String | 150 | Access Code (case sensitive) |
| Firstname | String | 150 | User first name |
| Lastname | String | 150 | User last name |
| Phone | String | 150 | User phone number |
| Picture | String | - | User profile picture |
| acct\_id | String | 64 | A unique string for identifying users |

**Primary key:** acct\_id

**Table 3.2 Recipe Table input specification table**

|  |  |  |  |
| --- | --- | --- | --- |
| **FIELD NAME** | **DATA TYPE** | **LENGTH** | **DESCRIPTION** |
| Recipe\_id | String | 150 | A unique string for identifying tricycles |
| Recipe\_title | String | 150 | Recipe title |
| Recipe\_desc | String | 500 | Detailed description of the recipe |
| Reg\_date | Date | 20 | Date when the recipe was registered |
| Preparation\_time | String | 150 | Duration of recipe |
| Difficult\_rate | String | 20 | Difficulty level of the recipe |
| Recipe\_image | Image | - | Image to identify the recipe |

**Primary key:** recipe\_id

**3.5 Output Design**

This declares and displays the outcome of the given input. The automated system's output is dependent on its input. The output specification is listed below.

**Table 3.3 Users Table Output Design**

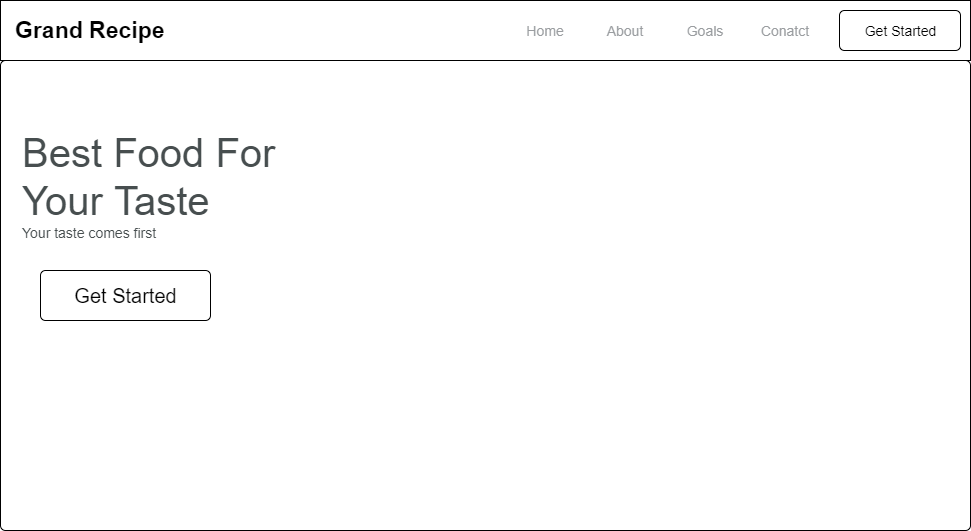
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Email** | **Password** | **Firstname** | **Lastname** | **Phone** | **Picture** | **Acct\_id** |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |

**Table 3.4 Recipe Table Output Design**

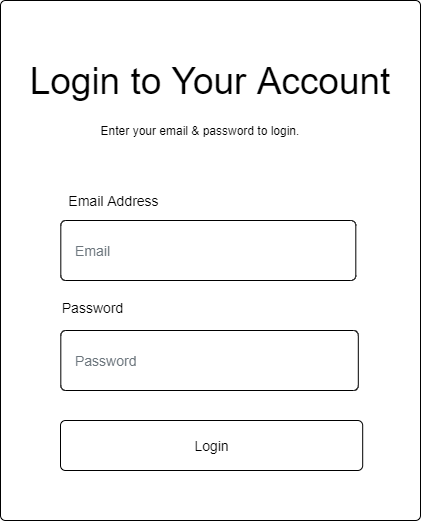
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Recipe\_id** | **Recipe\_title** | **Recipe\_desc** | **Reg\_date** | **Preparation\_time** | **Difficult\_rate** | **Recipe\_image** |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |
| XXXX | XXXX | XXXX | XXXX | XXXX | XXXX | XXXX |

**3.6 Input & User Interface Design**

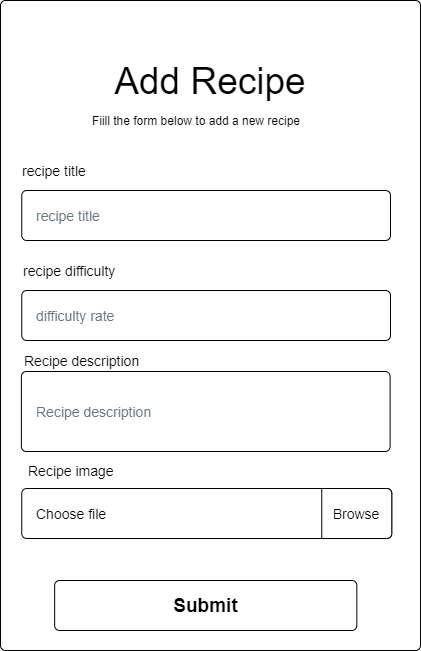
This displays the pictorial representation of the system interface, the interface is going to be designed in a way that it will be user friendly, responsive and attractive. It will also be well secured such that login will be required to access some level of contents. The designs are aided by a mid-fidelity wireframing tool called Draw.io



**Fig 3.6.1 Home Page Screen**



**Fig 3.6.2 Login Screen**



**Fig 3.6.3 Add Recipe Screen**

**3.7 System Requirement**

Every piece of software that is created has preset system requirements that it must meet in order to run at its best. However, the system requirements are the bare minimum hardware and software needed for the system's intended operation.

**3.7.1 Hardware Requirement**

System Hardware Requirement Include:

a. Minimum of 8 GB of RAM (Random Access Memory) installed.

b. Minimum of intel core i3 processor.

c. Minimum of 250GB HDD (Hard Disk Drive).

**3.7.1 Software Requirement**

The software requirements include:

a. At least windows 10 OS (Operating System).

b. Modern browsers such as Chrome, and Microsoft edge.

c. Vs. Code installation.

d. XAMMP installation.

**3.8 Choice of Programming Language**

The proposed design will be implemented using HTML, which is a markup language will be used for its user interface (frontend) while JavaScript will be used as the programming language for interacting with the database, MySQL will be used for its database due to its portability, the combination of the above modern technology forms the technology for this research work.