# Chapter 3: Design

# Introduction

Processing through the Software Development life cycle (SDLC), Design is one of the process which plays a distinct role to get the software been developed. According to the SDLC design falls under the third step of development. With the analysis progression and user requirement scheme the design part is undertaken with different techniques and tools defined like Activity Diagram, class diagram, sequence diagram, ER diagram, prototyping that indicated the work flow in this process, it helps in implementation process

Two different design models are used in the system design they are:

* Structural Model
* Behavioral Model

# Chapter 3.1- Structural design

Structural models of software display the organization of a system in terms of the components that make up that system and their relationships. Structural models may be static models, which show the structure of the system design, or dynamic models, which show the organization of the system when it is executing.

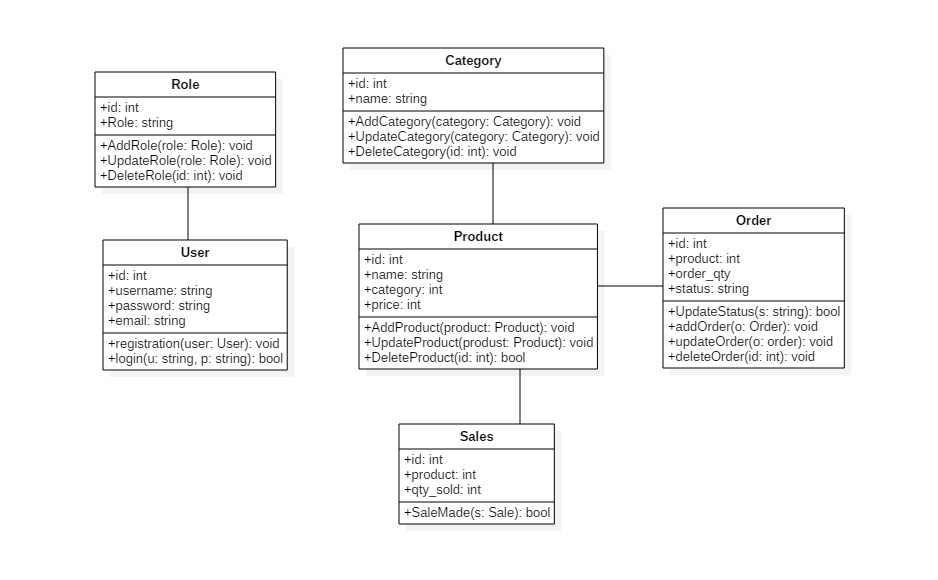
Class diagram and data flow diagram is created in this field of design.

# 3.1.1 Class diagram:

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

**Some reason of using Class diagram for our system are mentioned below:**

* Class Diagram Illustrates data models for even very complex information systems
* It provides an overview of how the application is structured before studying the actual code. This can easily reduce the maintenance time
* It helps for better understanding of general schematics of an application.
* Allows drawing detailed charts which highlights code required to be programmed
* Helpful for developers and other stakeholders.



# 3.1.2 Data flow diagram:

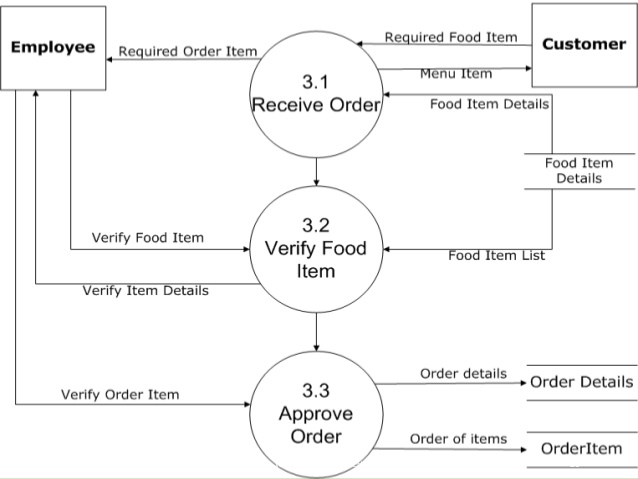
Data flow diagrams are used to graphically represent the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage and reports generation.

Data flow diagrams can be divided into logical and physical. The logical data flow diagram describes flow of data through a system to perform certain functionality of a business. The physical data flow diagram describes the implementation of the logical data flow.

**Some reason of using DFD diagram for our system are mentioned below:**

* It provides visual representation of how data moves between the system component.
* It helps to know the logic behind the data flow of the system.
* It is one of the most common modeling techniques used in design and development phase the system.

For Admin and user:



# Chapter 3.2- Behavioral diagram

Behavioral diagrams visualize, specify, construct, and document the dynamic aspects of a system. The behavioral diagrams are categorized as follows: use case diagrams, interaction diagrams, state–chart diagrams, and activity diagrams.

# 3.2.1 Activities diagram

An activity diagram depicts the flow of activities which are ongoing non-atomic operations in a state machine. Activities result in actions which are atomic operations.

**Some reason of using Activity diagram for our system are mentioned below:**

* It shows the activity flow between the system components in system.
* It is somewhat similar to flow chart and used to view logical view of system.
* It creates a helping hand for the developer while developing the system.

**For Login and registration**

Figure 8: Activities Diagram that shows login and registration flow

**For Admin**

Figure 9: Activities diagram that show Admin work flow

**For User**

Figure 10: Activity diagram that show user work flow

# 3.2.2 Sequence diagram

Sequence diagrams are interaction diagrams that illustrate the ordering of messages according to time.

**Some reason of using Sequence diagram for our system are mentioned below:**

* Interaction process between object can be viewed on the sequential order.
* It helps developer to know the interaction process of the system.
* High level of interaction modelling in the system.
* Object interaction between model with collaboration that realizes an operation.

For Admin:

Figure 11: Admin Sequence Diagram

**For user**

Figure 12: User sequence Diagram

# Chapter 3.3-Database design

# 3.3.1 Data dictionary

A data dictionary is a file or a set of files that contains a database's metadata. The data dictionary contains records about other objects in the database, such as data ownership, data relationships to other objects, and other data.

The data dictionary is a crucial component of any relational database. Ironically, because of its importance, it is invisible to most database users. Typically, only database administrators interact with the data dictionary.

Primary key and foreign key are the constraint used as data integrity enforced to be use in SQL Server table. We have built many data dictionary table including column name, data type, length, null, key.

# 3.3.2 ER diagram

An entity relationship diagram (ERD) shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. By defining the entities, their attributes, and showing the relationships between them, an ER diagram illustrates the logical structure of databases.

ER diagrams are used to sketch out the design of a database.

**Some reason of making ER diagram for our system are mentioned below:**

* It helps to view the entity in the system.
* Cardinality of relationship can be determined.
* It helps to find out relationship between entities.

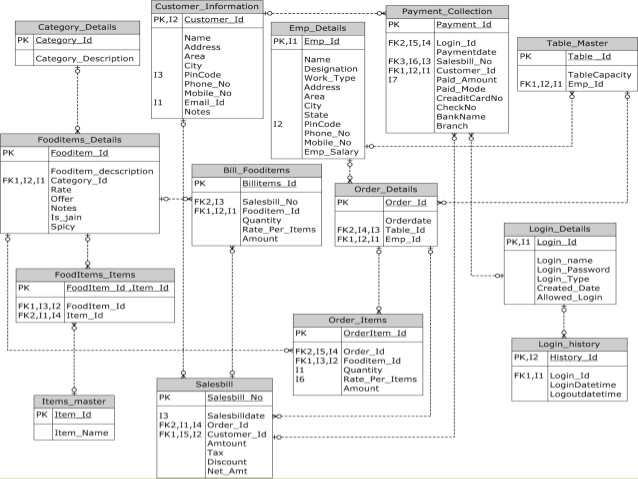


Figure 13: Entity Relationship (ER) diagram

# Chapter 3.4-UI Modeling/Design

UI (User Interface) is one of the most important aspect of software development. It contains Front-end part which users see within the application. It comprises the graphical elements such as text, audio, video, images and many more.

The prototype for the development system are:

# 3.4 Conclusion:

Finally, the design part is finished and further process of implementation is started as all the requirements and platform for coding is ready to be proceed i.e.: implementation.