# Chapter 3: Designing

## 3.1 Introduction to design

Design phase plays an important role to build a system. Here we can get the answer of “how” questions, arises during the development of system. GUI part and database design of the system development covered by this phase. To demonstrate the structure and workflow of the system, that is being developed I have used different models like structural model and behavioral model. It gives us an overview of the frontend and backend of the system. Design phase involves different diagrams so that non- technical user can know more about the system. The main purpose of the design phase is to transfer the user requirements into some acceptable form.

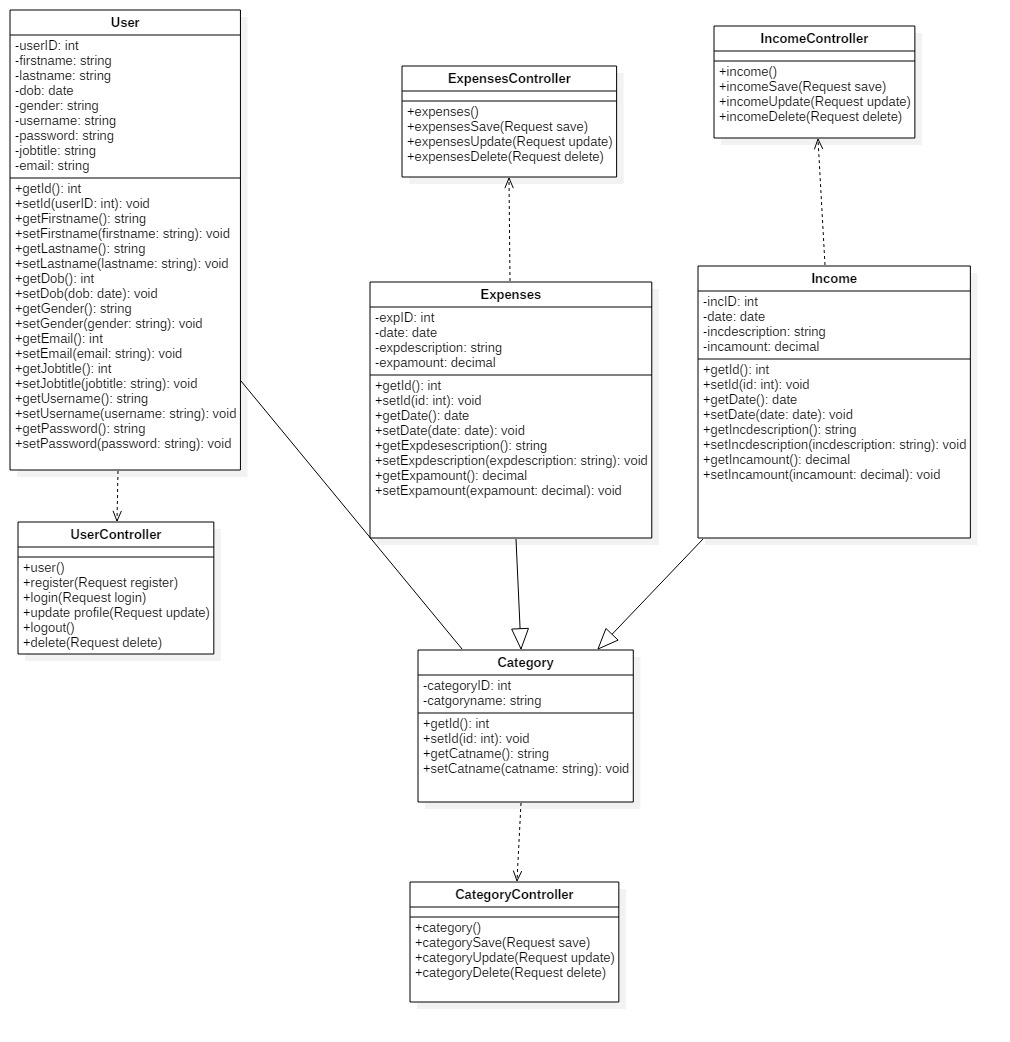
# 3.2 Structural Modelling

## 3.2.1 Final Class Diagram

In UML modelling, class diagram shows the relationship between classes, their attributes and operations. Class diagram is the backbone of an OO modelling and referred as blueprint of the system. It is the exception in UML Modelling that class diagram directly mapped with any object-oriented programming languages.

**Justification:**

I have chosen class diagram in comparison to other structural diagram because class diagram visualize, describe and document the different aspects of system. Class diagram is the only UML diagram that can directly mapped with object-oriented languages.



**Explanation:**

In design phase, final class diagram of expenses management system has created according to the MVC pattern. In MVC pattern, model is a non-visual object that holds all the data and behavior other than utilized for the User Interface (UI) design. Display of the model (i.e. data or information and behavior) in the User Interface represents View. Changes made to the data or information managed by third member called controller. Above class diagram shows the specific operations with their controller and view.

## 3.2.2 Context Diagram

Context diagram is the part of structural modelling which shows the overview of the complete system. The main purpose of this diagram is to show the expecting input and output values from the system, to and from various external entities. In the context diagram there is only one process to represents for entire system.

**Justification:**

I have chosen context diagram in comparison to other structural diagram because categories of context diagram is unique to other, describe the input, and output process of the system.

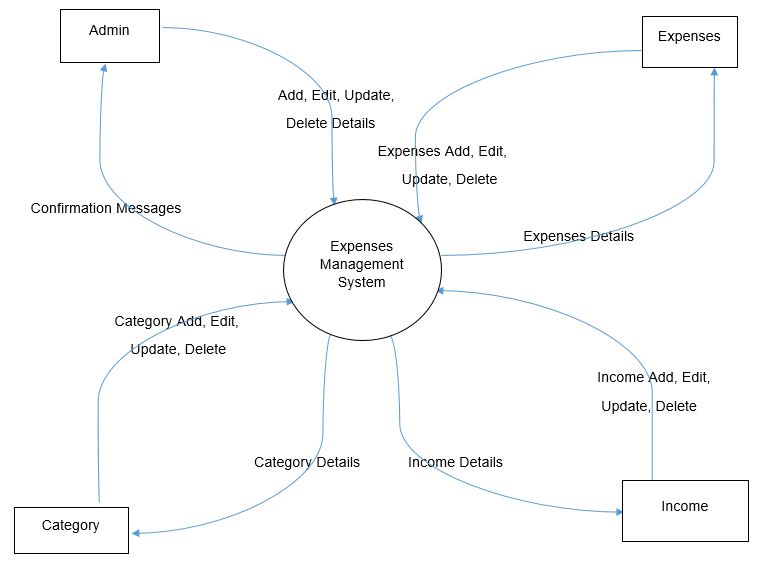
Categories of the context diagram include following points:

* Active: Dynamic to accomplish some objectives or reason
* Passive: represents external entities and interaction of those entities with the system
* Cooperative: to release some ideal result with the use of some external entities by the system
* Autonomous (Independent): Autonomous represents those external entities that are being separate from the system and indirectly affect the system by means of forced requirements.

**Notation used to draw Context Diagram:**

|  |  |  |
| --- | --- | --- |
| Notation Used | Name | Description |
|  | External Entity | Input data or retrieve data into and from the information system. |
|  | Processes | Action takes place on data and turning it into information. |
|  | Flow Line | Represents the flow of data (i.e. data is being sent or retrieved) |

**Context Diagram:**



**Explanation:**

Above context diagram represents the overview of the entire system. Admin can easily access into the user account and manipulate their data and returning confirmation message by providing what type of action take place. In case of expenses, income and category we can add, edit, update and delete data of those entities provide returning details message.