# Chapter 3: Design

Design is an important phase of development which determines how system should be built. It's the critical step in which logical and physical project preparation is conducted. It incorporates system functionality based on user requirements. Requirements in this stage are transformed into comprehensive and detailed system design specification.

## 3.1 Structural Modeling

The structural modeling shows my system's structural diagram by displaying the relationship from architecture map between classes, attributes, entity and process.

## 3.2 Final Class Diagram

The Class diagram are the class structure diagrams that is the part of unified UML language modeling that provides and defines an overview of the classes, relationship, attributes and methods between different objects and classes in a system.

**Justification for using class diagram:**

1. **Helps identify the class structure and how it interacts with each other before writing code that leads to more robust application development.**
2. **This helps to reduce maintenance time as the diagram gives an explanation of the layout of the specification.**
3. **Shows the type of classes and object that exists in the system and various kinds of static relationships.**

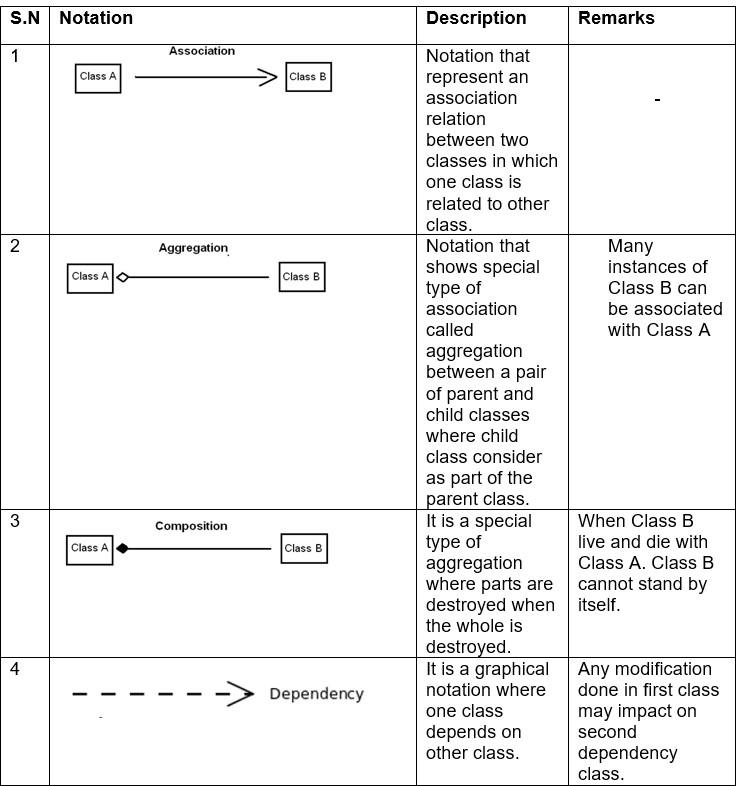
**Final class diagram:**

**A close up of text on a white background

Description automatically generated**

Figure 2 final class diagram

**Notations Used:**



## 3.3 Data Flow Diagram (DFD)

DFD is a visual representation of how information flows through a system or process of business information. Furthermore, it is represented how information is supplied and received from the system, source a well as destination of this information, and where this information is stored. It simplifies system implementation process by visually representing process and system that would be difficult to understand if represented in large portions of text documents.

Justification for using DFD:

* + - * Enables definition of system boundaries.
      * Assists in enhancing users’ knowledge of the system.
      * A basic graphical tool that is easy to use and recognize.
      * Presents comprehensive system description and its elements.

Notations for DFD:

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

DFD Explanation for user

1. External "User" entity provides login credentials for the "Login System" process, which is then checked with the Registration Table from the data store. If credentials are shown as valid dashboard otherwise error message will be sent to the user.
2. User enters the dashboard after valid credentials are issued which possesses donor management and campaign management processes.
3. For Manage donor Process:
   * Adds new donor information and data is stores in blood donor table.
   * The donor table displays the donor information to the user.
   * Further, user can update donor information that is store in blood donor table.
   * Similarly, user can delete donor information which is in turn deleted from data blood donor table.
4. For Manage campaign Process:
   * Adds new campaign information and data is stores in campaign table.
   * The campaign table displays the campaign information to the user.
   * Further, user can update campaign information that is store in campaign table.
   * Similarly, user can delete campaign information which is in turn deleted from data campaign table.

# 3.4 Behavioral Modelling

The behavioral modeling is a UML depiction that represents a system's internal behavior. This focuses on the system's dynamic view and not the way it's applied. This represents the workflows by means of sequence of communication between objects to provide functionality and data changes.

## 3.4.1 Activity Diagram:

Activity diagram shows the system flow control and the steps involved from start to finish point showing the various decision paths that occur when performing a use case activity Here the dynamic system can be interpreted as both the concurrent and the sequential processing of an operation.

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A close up of a map

Description automatically generated

Figure 3activity daigram

In the above activity diagram, the user of various organization, first register their credential then the system authenticates their registration information. Then the user can login to the system using his/her username and password, then after login the system present them with dashboard which has three 6 parallel option to add, edit, view, delete: donors and campaigns. Then all the input added or removed to the dashboard is either add or removed from the storage depending on the user action. Then there is the logout button from where they can logout out of the system.

## 3.4.2 Sequence Diagram

Sequence diagrams in UML are diagrams of interactions that describe how operations are performed. In the sense of a collaboration they capture the interaction between objects. It captures how different system objects interact with each other in collaboration. These diagrams are time-centric and visually represent the order of the interaction by using the diagram's vertical axis to represent what messages are being sent.

Notation used:

A screenshot of a cell phone

Description automatically generated

Diagram:

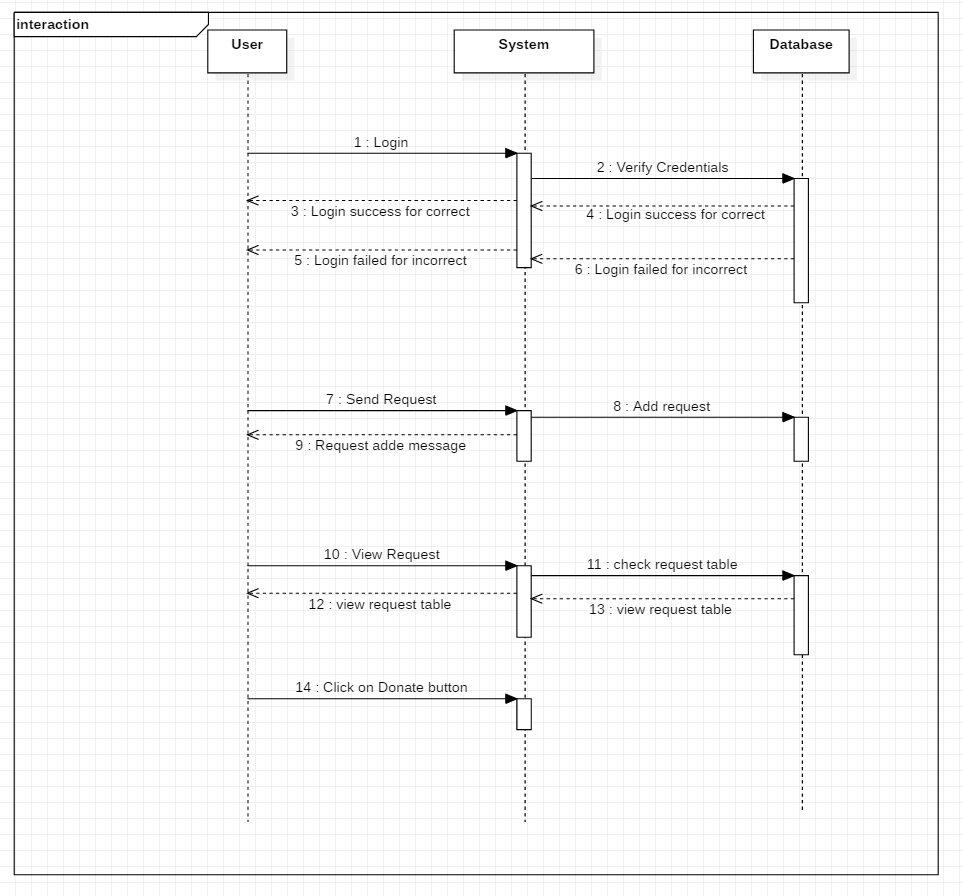


Figure 4 sequence daigram

From the example above, when a user logs in, they can only access their accounts if otherwise the credential given is right, they cannot. When the user sends request by filling out the form, the system adds a request to a database request table. Users send messages for viewing the requests, testing the system in the database, and responding as the database replied, and showing the table needed. When the user clicks the button on the table of requests the request status is updated.