

# **ONLINE DONATION TO NGO**

# ONLINE DONATION TO NGO

## A Project Report

Submitted in partial fulfillment of the  
Requirements for the award of the Degree of

**BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)**

By

**Ram Gajanan Barapatre**

Seat Number

Under the esteemed guidance of

**Miss. Lochan Handure**



**DEPARTMENT OF INFORMATION TECHNOLOGY  
MODEL COLLEGE OF SCIENCE & COMMERCE**

*University of Mumbai*

**KALYAN (E)-421306**

**MAHARASHTRA**

**2019-2020.**

**PROFORMA FOR THE APPROVAL PROJECT PROPOSAL**

## ONLINE DONATION TO NGO

---

*(Note: All entries of the proforma of approval should be filled up with appropriate and complete information. Incomplete proforma of approval in any respect will be summarily rejected.)*

PNR No.: .....Roll no: \_\_\_\_\_

1. Name of the Student : Ram Gajanan Barapatre

2. Title of the Project : Online Donation To NGO

3. Name of the Guide : Lochan Handure

4. Teaching experience of the Guide : 3 Years

5. Is this your first submission? Yes ☐ No ☐

Signature of the Student

Date: .....

Signature of the Guide

Date: .....

Signature of the coordinator

Date: .....

# ONLINE DONATION TO NGO

---

**MODEL COLLEGE OF SVIENCE & COOMMERCE**

*(Affiliated to University of Mumbai)*

**DEPARTMENT OF INFORMATION TECHNOLOGY**



## **CERTIFICATE**

This is to certify that the project entitled, "**ONLINE DONATION TO NGO** ", is bonafied work of **Ram Gajanan Barapatre** bearing Seat No: ( ) submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

**Internal Guide**

**Coordinator**

**External Examiner**

**Date:**

**College Seal**

**DECLARATION**

I RAM GAJANAN BARAPATRE STUDENT OF MODEL COLLEGE OF SCI. & COM. RAJBHAR NAGAR CHINHPADA ROAD, KATEMANAVALI, KALYAN (EAST) 421306. STUDYING IN 3<sup>RD</sup> YEAR BSC INFORMATION TECHNOLOGY HAVE BY DECLARE THAT I HAVE COMPLETED THIS PROJECT ON SOCIAL NETWORKING SITE DURING THE ACADEMIC YEAR 2019-20

THE INFORMATION SUBMITTED IS TRUE AND ORIGINAL TRUE THE BEST OF MY KNOWLEDGE

DATE:

PLACE:

**Ram Gajanan Barapatre**

### **ACKNOWLEDGEMENT**

I WISH TO EXPRESS MY SINCERE GRATITUDE TO PROJECT **MR. K.S BRAMHAWALE** PRINCIPAL OF MODEL COLLEGE OF SCI. & COM. FOR PROVIDING ME AND OPPORTUNITY TO DO ANY MY WEB PROJECT WORK ON ONLINE DONATION TO NGO IN SINCERELY THANK MY PROJECT GUIDE **MISS.LOCHAN HANDURE** FOR GUIDENCE AND ENCOURAGMENT IN CARRYING OUT THIS PROJECT WORK.

SPECIAL THANKS TO ALL THE LAB SYSTEMS FOR SEEMINGLY SMALL BUT VALUABLE HELP FOR TIMELY INTERNET ACCESS AND LAB ACCESS

**TABLE OF CONTENTS**

Sr. No.	INDEX	Pg. No.	Sign
<b>CHAPTER 1:</b>	<b>INTRODUCTION</b>		
1.1	Background		
1.2	Objectives		
1.3	Purpose, Scope & Applicability		
1.3.1	Purpose		
1.3.2	Scope		
1.3.3	Applicability		
1.4	Organization of Report		
<b>CHAPTER 2:</b>	<b>SURVEY OF TECHNOLOGIES</b>		
<b>CHAPTER 3:</b>	<b>REQUIREMENTS AND ANALYSIS</b>		
3.1	Problem Definition		
3.2	Requirements Specification		
3.3	Planning and Scheduling		
3.4	Software and Hardware Requirements		
3.5	Conceptual Models		
<b>CHAPTER 4:</b>	<b>SYSTEM DESIGN</b>		
4.1	Basic Modules		
4.2	Data Design		
4.2.1	Schema Design		
4.2.2	Data Integrity and Constraints		
4.3	Procedural Design		
4.3.1	Logic Diagrams		
4.3.2	Algorithm Design		
4.4	User Interface Design		
4.5	Security Issues		
4.6	Test Case Design		

### **ABSTRACT**

The “Online Donation to NGO” system is developed to nongovernmental organization. To manage donor registration and user maintenance. It is an only system which only can be access or valid in Maharashtra state. The application will be processed by the administrator and each donor will receive the feedback about their donation status. Furthermore, the authorized user’s account will be maintained by the administrator.

This online donation to NGO system will help to improve the performance of current situation and overcome the problems that rise today’s.



### **CHAPTER 1:- INTRODUCTION**

A non-governmental organization (NGO) is a non-profit, citizen-based group that functions independently of government. NGOs are organized on local, national and international levels to serve specific social or political purposes.

As non-profit organizations, NGOs rely on a variety of sources for funding projects, operations, salaries and other overhead costs. Because the annual budget of an NGO can be in the hundreds of millions (or even billions) of dollars, fundraising efforts are important for the NGO's existence and success. Funding sources include membership dues, the sale of goods and services, private sector for-profit companies, philanthropic foundations, grants from local, state and federal agencies, and private donations.

## **1.1 BACKGROUND**

**Non-governmental organizations, nongovernmental organizations, or nongovernment organizations,** commonly referred to as **NGOs**, are usually non-profit and sometimes international organizations independent of governments and international governmental organizations (though often funded by governments) that are active in humanitarian, educational, health care, public policy, social, human rights, environmental, and other areas to affect changes according to their objectives. They are thus a subgroup of all organizations founded by citizens, which include clubs and other associations that provide services, benefits, and premises only to members. Sometimes the term is used as a synonym of "civil society organization" to refer to any association founded by citizens but this is not how the term is normally used in the media or everyday language, as recorded by major dictionaries. The explanation of the term by NGO.org (the non-governmental organizations associated with the United Nations) is ambivalent.

### **1.2 OBJECTIVES**

The Objective of the system are as follows:-

- To provide Social, Educational and Economic Empowerment to Women and Children.
- To safeguard the rights of the Child and provide for the well being of Children in need, more particularly,
- To mentor and provide guidance to the youth.
- To preserve and protect the environment and propagate the significance of sustainable development for maintaining ecological balance.
- To conduct study on various social issues from time to time, gather data, and carry out research.

### **1.3.1 PURPOSE**

NGOs are organized on local, national and international levels to serve specific social or political purposes. As non-profit organizations, NGOs rely on a variety of sources for funding projects, operations, salaries and other overhead costs.

Purpose of the System

- A. To generate the quick reports
- B. To provide proper information
- C. To provide data security
- D. To provide huge maintenance of records

After understanding the existing system and understanding the need for developing a new system different people involved in the related activities have been consulted. The data needed for the study has been collected from company records.

### **1.3.2 SCOPE**

Non Government Organizations are the non-profit voluntary groups established at local, national or international level. They perform different tasks for solving problems and development of society. NGOs are connected with government or private sector firms. They deal with some social issues like women empowerment, girl child, gender issues, education, pollution, street children, slum dwellers, health, urban development, human rights, concerns of less privileged etc. NGOs bring up people's concerns and issues to the government and policy makers non-profit making, voluntary, service-oriented/development oriented organization, either for the benefit of members or of other members of the population.

### **1.3.3 APPLICABILITY**

External funding is often needed for the implementation of projects, especially in the case of the water and sanitation sector, where a mix of financial mechanisms is needed to cover the software and hardware components involved in a project.

## **1.4 ORGANISATION OF REPORT**

- **SURVEY OF TECHNOLOGIES** : In this chapter we should discuss the student should awareness and understanding of available technologies related to the topic
- **REQUIREMENTS AND ANALYSIS**: In this chapter we should discuss the requirements specification of the system i.e hardware and software problem definition planning and scheduling.
- **SYSTEM DESIGN**: In this chapter we discuss the or describes features and operation in detail, including screen layout , business rule, process diagram, pseudo code and other documentation.
- The chapter 5 to 7 include the **IMPLEMENTATION AND TESTING** ,**RESULTS AND DISCUSSION** ,**CONCLUSIONS**,**REFERENCES** should be submitted in next semester i.e VI
- **IMPLEMENTATION AND TESTING**: inside the implementation and testing we should discuss Coding Details and Code Efficiency, types of testing, testing approaches, Modifications and Improvements of there projects.

## ONLINE DONATION TO NGO

---

- **RESULTS AND DISCUSSION** :we should discuss the test reports and user documentation
- **CONCLUSIONS**: The conclusions can be summarised in a fairly short chapter (2 or 3 pages). This chapter brings together many of the points that would have made in the other chapters.
- **REFERENCES**: In this topic we should discuss the bibliography and website used to create the project.

## **CHAPTER 2:-SURVEY OF TECHNOLOGIES**

### **Front End - ASP.Net**

As we need to develop Web Application for ONLINE DONATION TO NGO

We will use ASP.Net as it is the new Microsoft technology to develop any application

which support and integrate other server product for the internet.

### **C#**

.NET is a "Software Platform". It is a language-neutral environment for developing rich

.NET experiences and building applications that can easily and securely operate within it.

### **Back End - SQL server 2008**

SQL database is the world's most popular open source database because of its fast performance, high reliability, ease of use, and dramatic cost savings.

SQL is a database. A database is a data storage feature. It can be used to store, sort, arrange, and display information.

SQL Server 2008 main features is

1. Support for Multiple Platforms
2. Ultra-fast load utilities
3. High Performance
4. Scalability

## **CHAPTER 3 :-REQUIREMENTS AND ANALYSIS**

### **3.1 PROBLEM DEFINATION**

There are several problems that identified in current manual system. The proposed enhancement system can prevent and overcome the existing problems. First of all, the public are less publicity and knowledge about NGO due to the un attractive websites. Moreover , the current system is lack security. It is not protected and responsible by certain people.

Present system is manual. The Project Metrics has to enter all the details of project, documents, and tasks. It also maintenance the team information and also efforts estimation. For this purpose the organization maintain the size of the document, source code and update the information about team member's details manually. Which is much of time consuming process and more importantly it is error prone. Limitations Of the Manual system

1. It is time consuming
2. It leads to error prone results
3. It consumes lot of manpower to better results
4. It lacks of data security
5. Retrieval of data takes lot of time
6. Percentage of accuracy is less
7. Reports take time to produce

Hence Computerization of the existing system is proposed. The new system completely removes all manual burdens and provide efficient on the entry system.



### **3.2 REQUIREMENTS SPECIFICATION**

The project has been planned to be having the view of distributed architecture, with centralized storage of the database. The application for the storage of the data has been planned. Using the constructs of MS-SQLServer2000 and all the user interfaces have been designed using the ASP.Net technologies. The database connectivity is planned using the “SQL Connection” methodology. The standards of security and data protective mechanism have been given a big choice for proper usage. The application takes care of different modules and their associated reports, which are produced as per the applicable strategies and standards that are put forwarded by the administrative staff

### **3.3 PLANNING AND SCHEDULING**

#### **➤ PLANNING**

Project planning defines the project activities and end products that will be performed and describes how the activities will be accomplished. The purpose of project planning is to define each major task, estimate the time and resources required, and provide a framework for management review and control. The project planning activities and goals include defining:

- ❖ The specific work to be performed and goals that define and bind the project.
- ❖ Estimates to be documented for planning, tracking, and controlling the project.
- ❖ Commitments that are planned, documented, and agreed to by affected groups.
- ❖ Project alternatives, assumptions, and constraints.

#### **➤ SCHEDULING:**

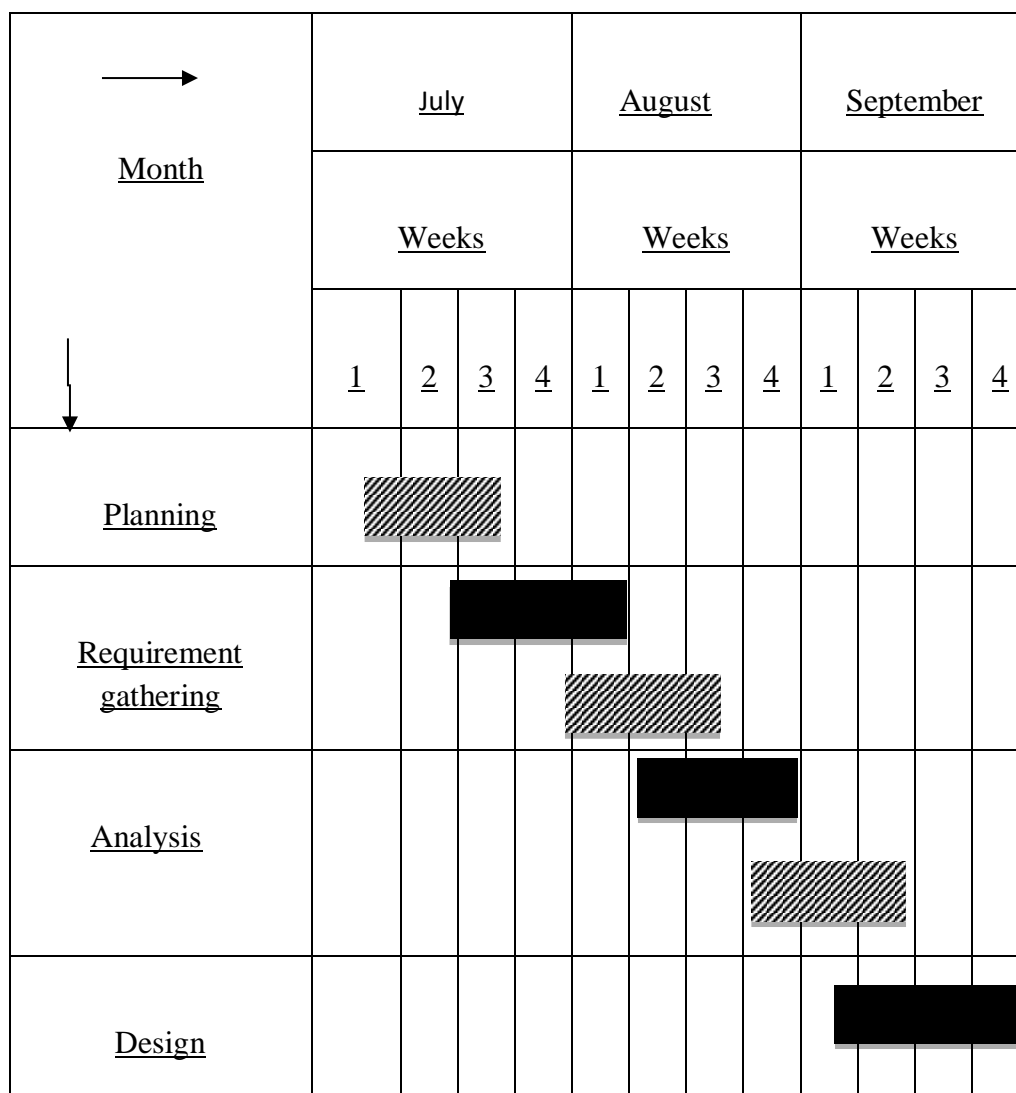
The project schedule provides a graphical representation of predicted tasks, milestones, dependencies, resource requirements, task duration, and deadlines. The project's master schedule interrelates all tasks on a common time scale. The project schedule should be detailed enough to show each WB Stask to be performed, the name of the person responsible for completing the task, the start and end date of each task, and the expected duration of the task.

- ❖ Define the type of schedule
- ❖ Define precise and measurable milestones
- ❖ Estimate task duration
- ❖ Define priorities
- ❖ Define the critical path
- ❖ Document assumptions
- ❖ Identify risks

#### **GANTT CHART**

## ONLINE DONATION TO NGO

- A Gantt chart is a horizontal bar chart used in project management as a tool for graphically representing the schedule of a set of specific activities or tasks. The horizontal bars indicate the length of time allocated to each activity, so the x-axis of a Gantt chart is subdivided into equal units of time, e.g., days, weeks, months. The y-axis of a Gantt chart, on the other hand, simply lists all the activities or tasks being monitored by the Gantt chart. A simple look at a Gantt chart should enable its user to determine which tasks take the longest time to complete, which tasks are overlapping with each other, etc.



### **3.4 HARDWARE AND SOFTWARE REQUIREMENTS**

#### **HARDWARE:**

Processor	:	Intel Core 4, 2.0GHz or above
Ram	:	2GB or above
Hard Disk	:	320GB
Monitor	:	LCD
Keyboard	:	Normal or Multimedia
Mouse	:	Compatible Mouse

#### **SOFTWARE:**

Front end	:	ASP.NET
Back end	:	SQL
Operating System	:	WINDOWS 7/8/10

### 3.5 CONCEPTUAL MODLES

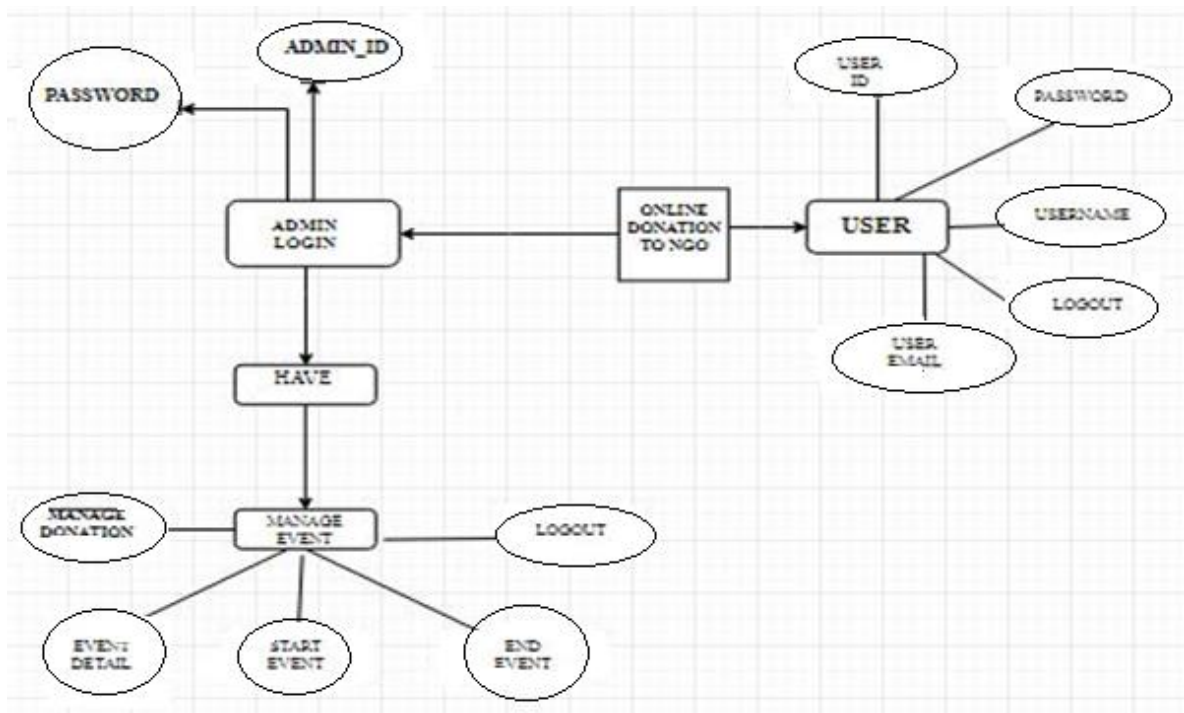
**ER DIAGRAMS:** A graphical model of the data needed by a system, including things about which information is stored & the relationships among them, produced in structured analysis & information engineering. ER Diagram represents entities or tables and their relationships with one another.

#### **Symbols:**



Connection between components. An arrow generally indicates a flow from one component to another

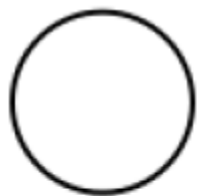
## ER DIAGRAM



### DATA FLOW DIAGRAM

- A **data flow diagram (DFD)** is a graphical representation of the "flow" of data through an information system.
- DFDs can also be used for the visualization of data processing (structured design).
- It views a system as a function that transform the input into desired output.

SYMBOLS:



Function



File/Database



Input/Output

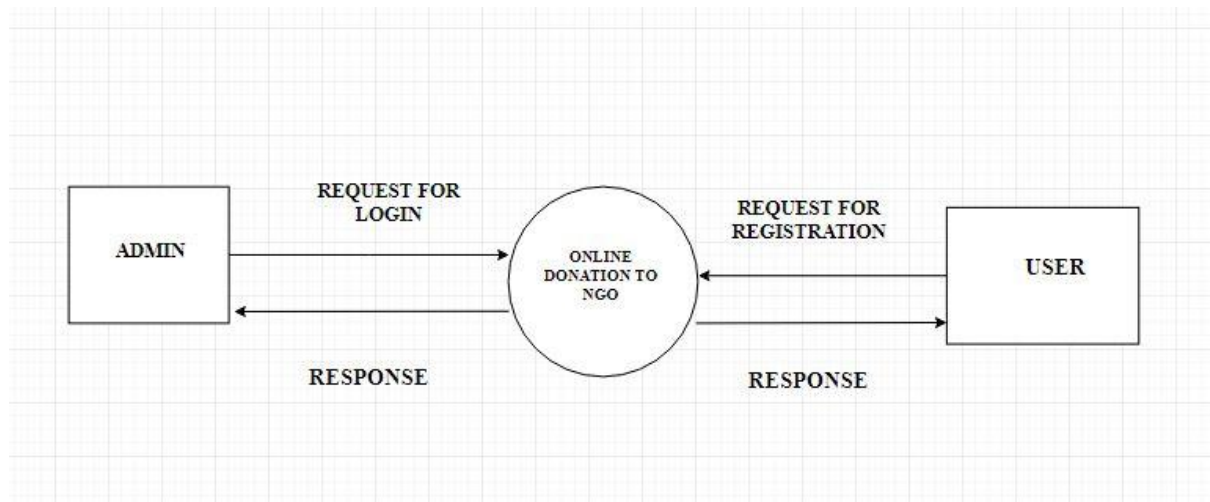


Flow

# ONLINE DONATION TO NGO

---

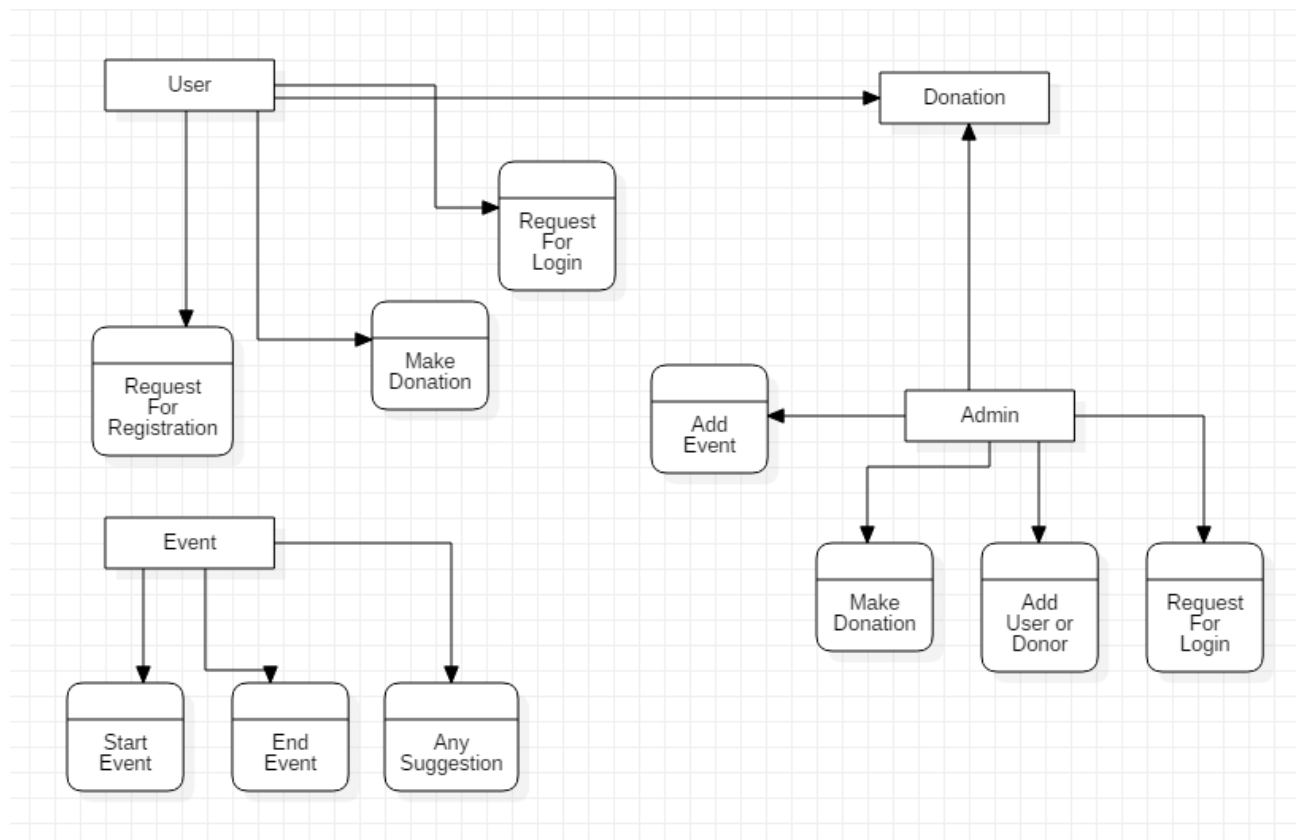
## DFD LEVEL -0



## DFD LEVEL -1



## ONLINE DONATION TO NGO

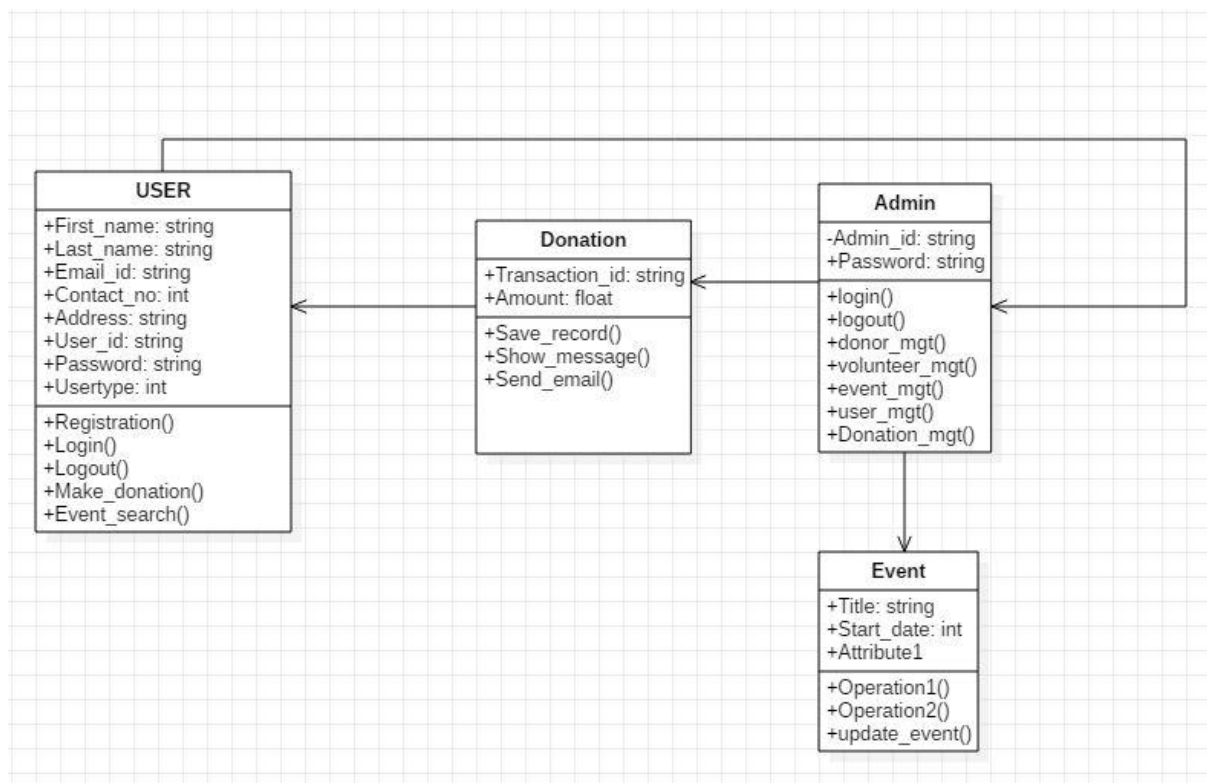


## OBJECT ORIENTED DIAGRAM

### CLASS DIAGRAM

## ONLINE DONATION TO NGO

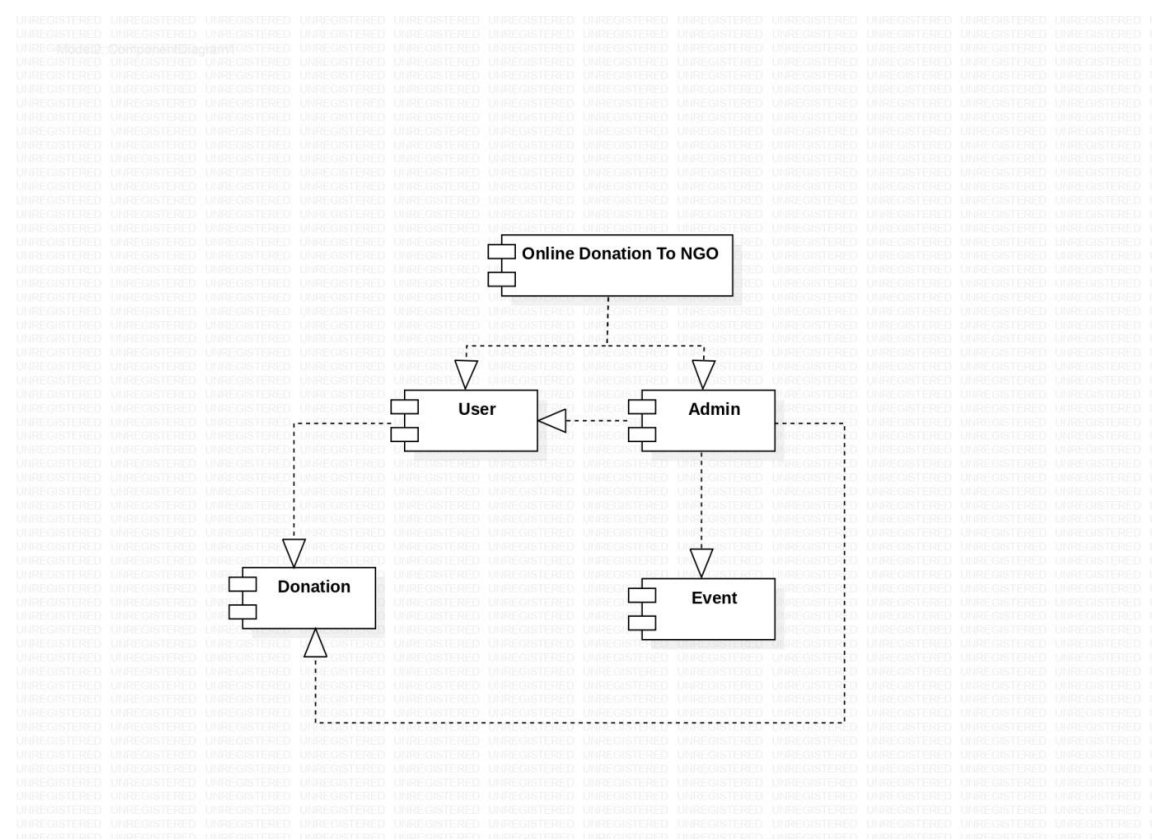
A class diagram is an illustration of the relationships and source code dependencies among classes in the Unified Modeling Language (UML). In this context, a class defines the methods and variables in an object, which is a specific entity in a program or the unit of code representing that entity. Class diagrams are useful in all forms of object-oriented programming (OOP).



### Component Diagram

# ONLINE DONATION TO NGO

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system. Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required functions is covered by planned development.



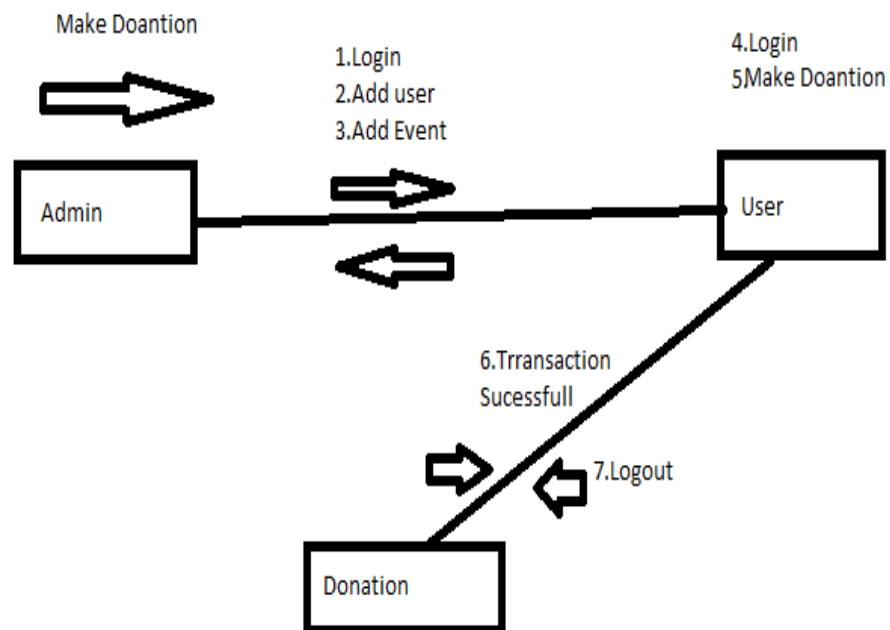
## Collaboration Diagram

A collaboration diagram, also known as a communication diagram, is an illustration of the relationships and interactions among software objects in the Unified Modeling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular **use case** and define the role of each object.

Collaboration diagrams are created by first identifying the structural elements required to carry out the functionality of an interaction. A model is then built using the relationships between those elements. Several vendors offer software for creating and editing collaboration diagrams.

## ONLINE DONATION TO NGO

---

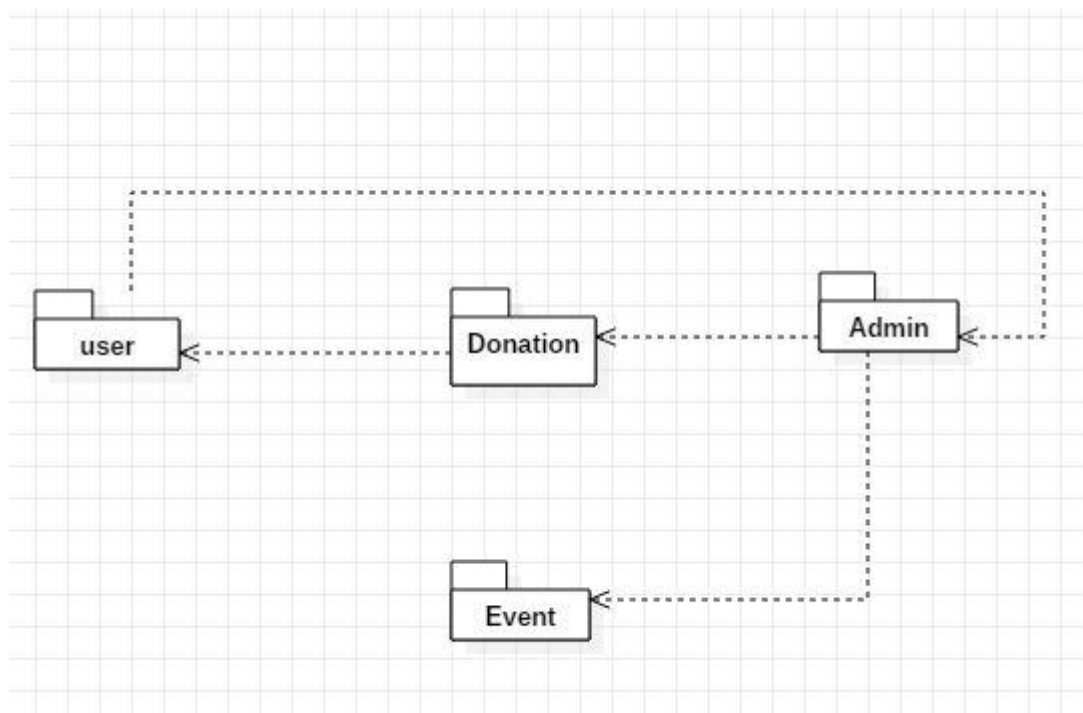


### **PACKAGE DIAGRAM:-**

Package diagram is UML structure diagram which shows packages and dependencies between the packages.

Model diagrams allow to show different views of a system, for example, as multi-layered (aka multi-tiered) application - multi-layered application model.

The following nodes and edges are typically drawn in a package diagram: package, packageable element, dependency, element import, package import, package merge.



## **CHAPTER 4 :-SYSTEM DESIGN**

### **4.1 BASIC MODULES**

BASIC MODULES: -

Ø Donor module

Ø Administrators module

**Donors module:** This module manages the details of all the donors who have come forward to participate in the charity funding. It also integrates itself to the charity request expenditures, and the charity clients information. This module also manages all the transactions and operations that relate to the donars.

**Administrative module:** This module takes the overall responsibility upon the systems transactions and operations. The administrative module takes the sole responsibility of managing the reliability and consistency of the data. The system takes care of the data collection, updatation and deletion of the actual data at the centralized level.

## ONLINE DONATION TO NGO

---

### 4.2 DATA DESIGN

#### EVENT TABLE

<u>Event</u>	<u>Trigger</u>	<u>Source</u>	<u>Activity</u>	<u>Response</u>	<u>Destination</u>
Admin Login	Login request sent	Admin	Validate password & login name	Login successful or denied	Admin
Admin creates program for donation	Donation program created.	Admin	Validate Donation Program Details	Donation Program Created	Admin
Admin deletes member account	Deletion request sent	Admin	Verifies and deletes account	Verified and deleted account	Admin
Admin creates event program	Add Event details	Admin	Add Event details successfully	Event Details Added	Admin
Admin delete Event program	Delete Event Details	Admin	Deleted Event details Successfully	Deleted Event details.	Admin

### **4.2.1 SCHEMA DESIGN**

A database system is an overall collection of different database software components and database containing the parts viz. Database application programs, front-end components, Database Management Systems, and Databases.

#### Normalization

Normalization is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency. Redundant data wastes disk space and creates maintenance problems .If data that exists in more than

one place must be changed , the data must be changed in exactly the same way in all locations. A buyer's or promoter's address change is much easier to implement if that data is stored only in the buyer's or promoter's table and nowhere else in the database. There are a few rules for database normalization. Each rule is called a "normal form

Data structuring is defined through a process called normalization. Data are grouped in the simplest way possible so that later changes can be made with a minimum of impact on the data structure.

There are different forms of normal forms

- First normal form(1NF)
- Second normal form(2NF)
- Third normal form(3NF)
- Boyce code normal form(BCNF)
- Fourth Normal form(4NF)
- Fifth Normal Form(5NF)



## ONLINE DONATION TO NGO

---

**User\_ DeTails Table:**

Field Name	Data Type	Null	Key
U_id	Varchar(50)	Not null	Primary key
Fname	Varchar(50)		
Lname	Varchar(50)		
Mobile	Varchar(50)		
Country	Varchar(50)		
City	Varchar(50)		

**Admin \_details Table :-**

Field Name	Data type	Null	Key
Admin_id	String	Not Null	Primary Key
Password	String	Not null	

**Event\_details Table**

Field name	Data Type	Null	Key
Event_id	String	Not null	Primary key
Event_title	String	Not null	
Event_startdate	int	NotNull	
Event_end date	Int	Not null	
Event_details	String	Not Null	

### **4.2.2 DATA INTEGRITY AND CONSTRAINTS**

The term data integrity refers to the accuracy and consistency of data.

When creating databases, attention needs to be given to data integrity and how to maintain it. A good database will enforce data integrity whenever possible.

For example, a user could accidentally try to enter a phone number into a date field. If the system enforces data integrity, it will prevent the user from making these mistakes.

Maintaining data integrity means making sure the data remains intact and unchanged throughout its entire life cycle. This includes the capture of the data, storage, updates, transfers, backups, etc. Every time data is processed there's a risk that it could get corrupted (whether accidentally or maliciously).

#### **Risks to Data Integrity**

Some more examples of where data integrity is at risk:

- A user tries to enter a date outside an acceptable range.
- A user tries to enter a phone number in the wrong format.
- A bug in an application attempts to delete the wrong record.
- While transferring data between two databases, the developer accidentally tries to insert the data into the wrong table.
- While transferring data between two databases, the network went down.
- A user tries to delete a record in a table, but another table is referencing that record as part of a relationship.

### 4 Types of Data Integrity

In the database world, data integrity is often placed into the following types:

- ❖ Entity integrity
- ❖ Referential integrity
- ❖ Domain integrity
- ❖ User-defined integrity

#### Entity Integrity

Entity integrity defines each row to be unique within its table. No two rows can be the same. To achieve this, a primary key can be defined. The primary key field contains a unique identifier – no two rows can contain the same unique identifier.

#### Referential Integrity

Referential integrity is concerned with relationships. When two or more tables have a relationship, we have to ensure that the foreign key value matches the primary key value at all times. We don't want to have a situation where a foreign key value has no matching primary key value in the primary table. This would result in an orphaned record.

So referential integrity will prevent users from:

- Adding records to a related table if there is no associated record in the primary table.
- Changing values in a primary table that result in orphaned records in a related table.
- Deleting records from a primary table if there are matching related records.

#### Domain Integrity

Domain integrity concerns the validity of entries for a given column. Selecting the appropriate data type for a column is the first step in maintaining domain integrity. Other steps could

include, setting up appropriate constraints and rules to define the data format and/or restricting the range of possible values.

### **User-Defined Integrity**

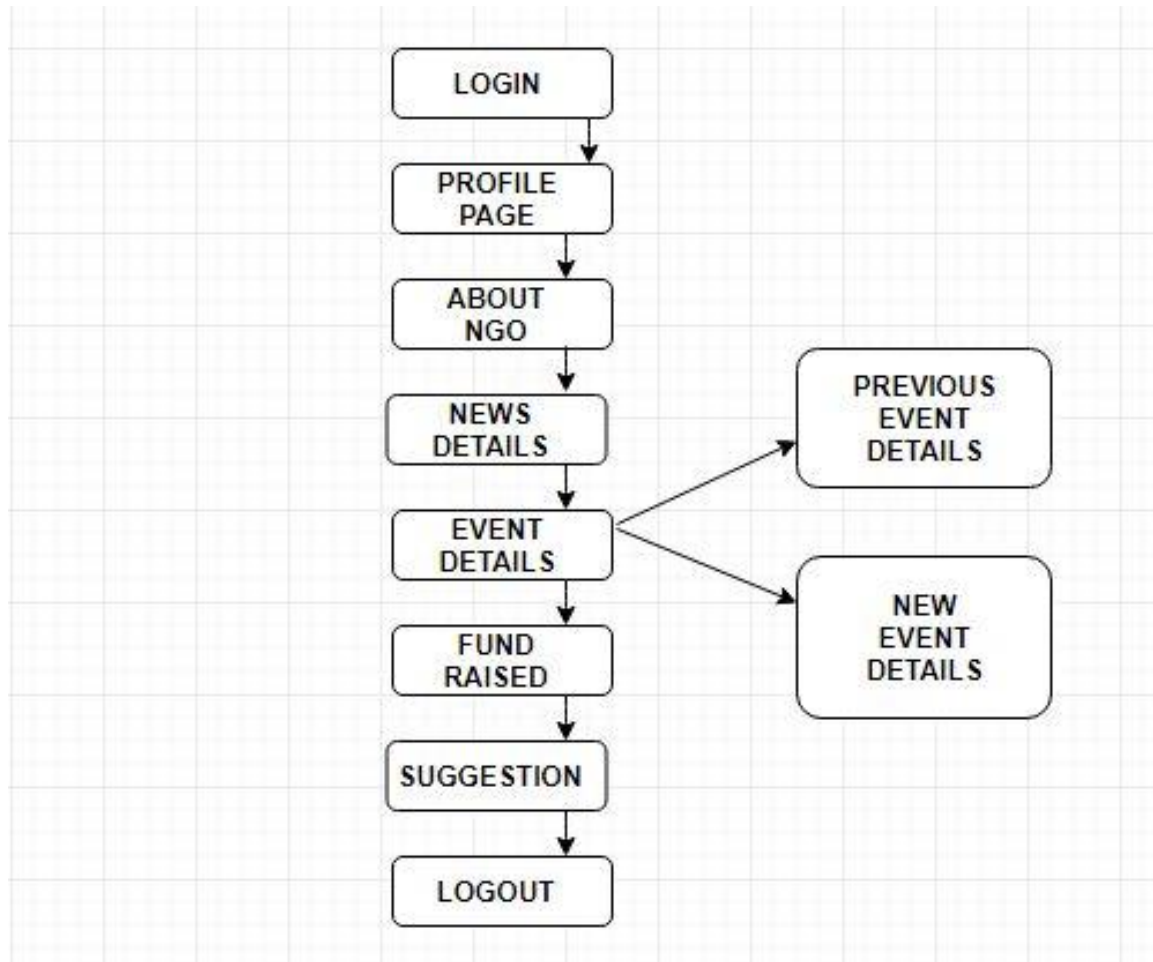
User-defined integrity allows the user to apply business rules to the database that aren't covered by any of the other three data integrity types

Constraints enforce limits to the data or type of data that can be inserted/updated/deleted from a table. The whole purpose of constraints is to maintain the **data integrity** during an update/delete/insert into a table. In this tutorial we will learn several types of constraints that can be created in RDBMS.

### **Types of constraints**

- NOT NULL
- UNIQUE
- DEFAULT
- CHECK
- Key Constraints – PRIMARY KEY, FOREIGN KEY
- Domain constraints
- Mapping constraints

### 4.3 PROCEDURAL DESIGN:-



### **4.3.1 LOGIC DIAGRAM:-**

The diagram control is used for creating diagrams like flow charts, organizational charts, mind maps, floor plans, and swim lane diagrams either through code or through a visual interface.

#### **Nodes**

Nodes are used to host graphical objects like paths and controls that can be arranged and manipulated on a diagram page.

Many predefined standard shapes are included.

- Custom shapes can also be created and added easily.
- A node's appearance can be fully customized.
- A node's UI can also be templated and re-used across multiple nodes.

#### **Connectors :-**

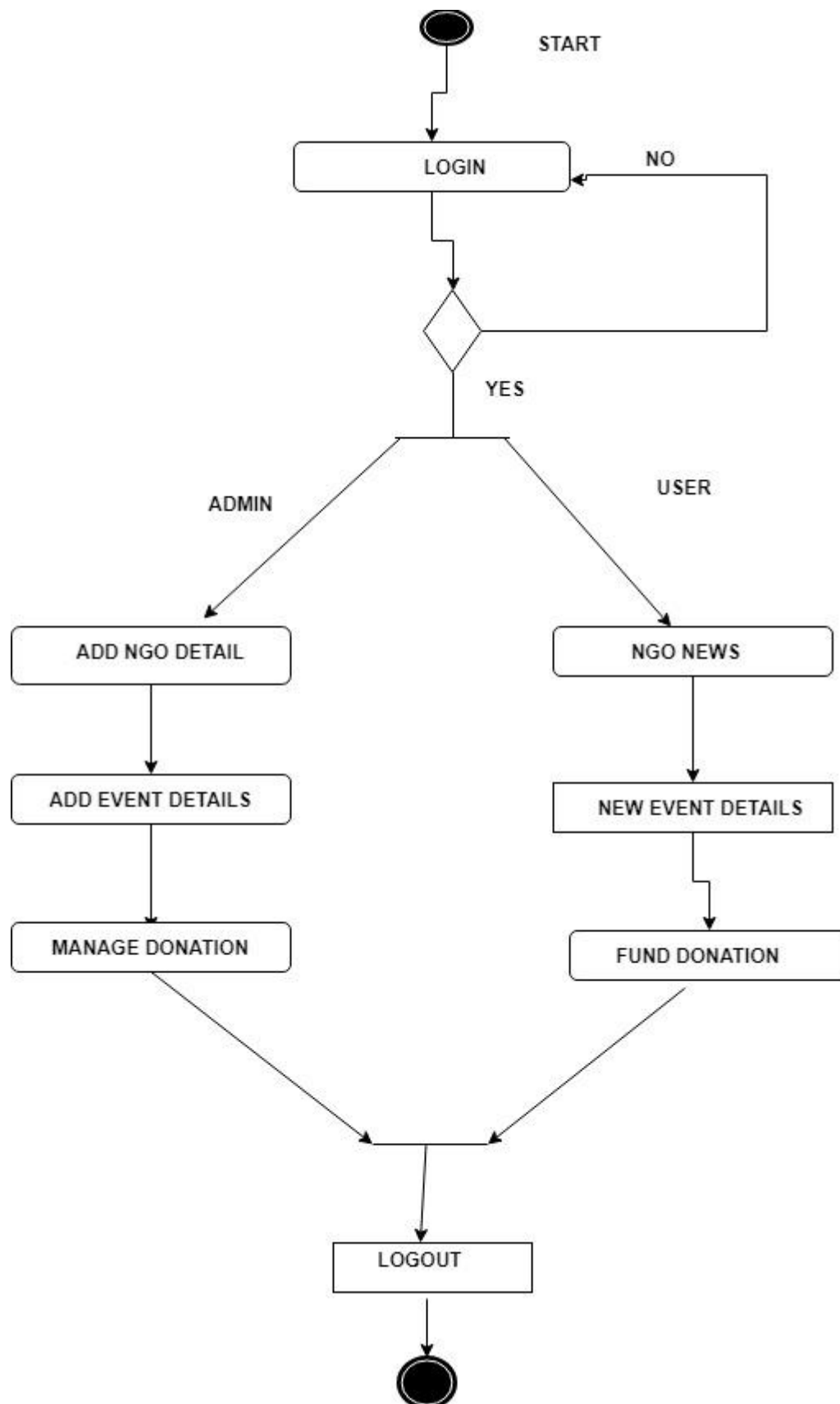
The relationship between two nodes is represented using a connector. Multiple instances of nodes and connectors form a diagram.

### **ACTIVITY DIAGRAM**

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc

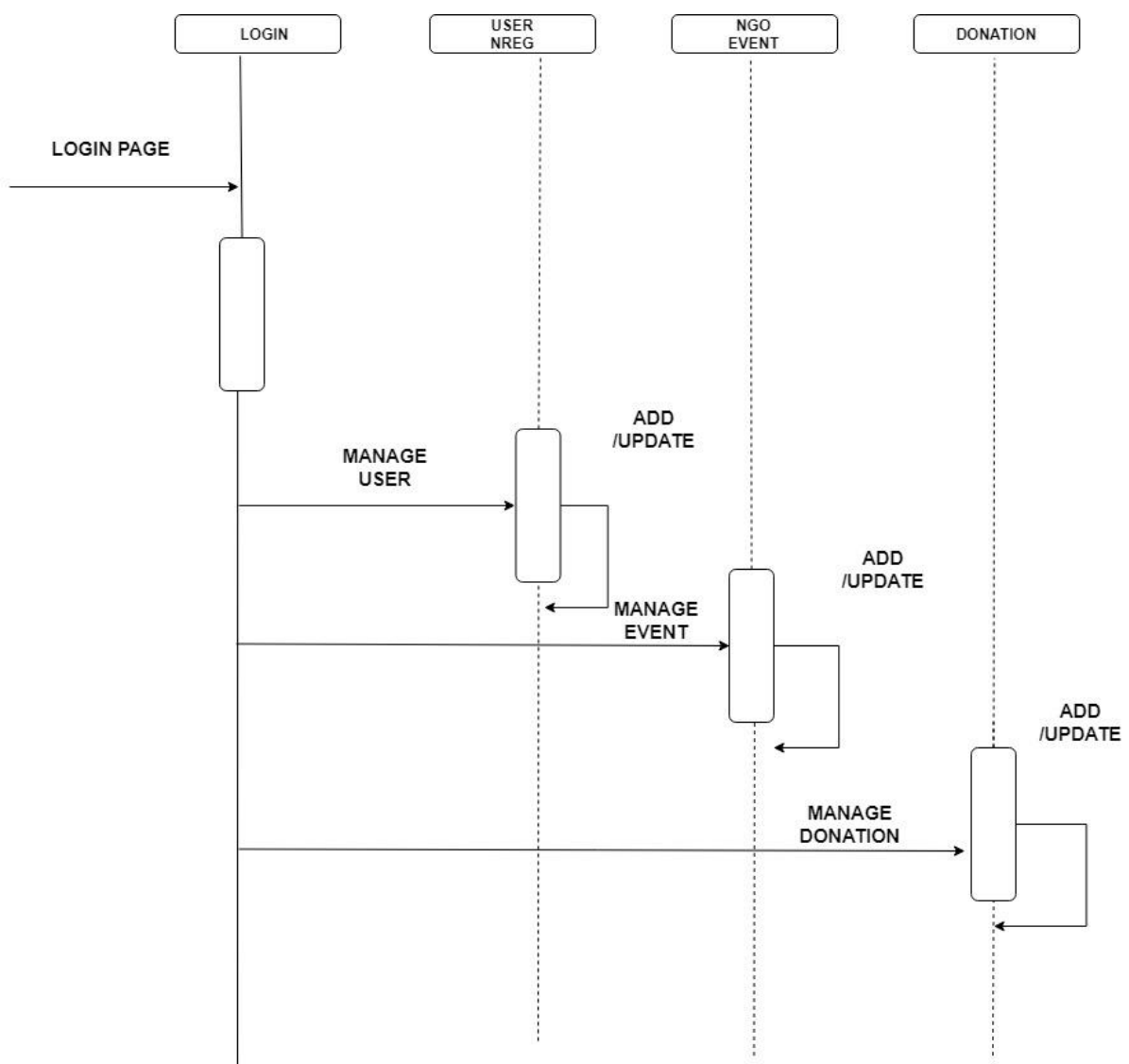
## ONLINE DONATION TO NGO





## SEQUENCE DIAGRAM

Sequence diagrams describe interactions among classes in terms of an exchange of messages over time. They're also called event diagrams. A sequence diagram is a good way to visualize and validate various runtime scenarios. These can help to predict how a system will behave and to discover responsibilities a class may need to have in the process of modeling a new system.



### **4.3.2 ALGORITHM DESIGN :-**

An algorithm is a finite set of instructions or logic, written in order, to accomplish a certain predefined task. Algorithm is not the complete code or program, it is just the core logic(solution) of a problem, which can be expressed either as an informal high level description as **pseudo code** or using a **flowchart**.

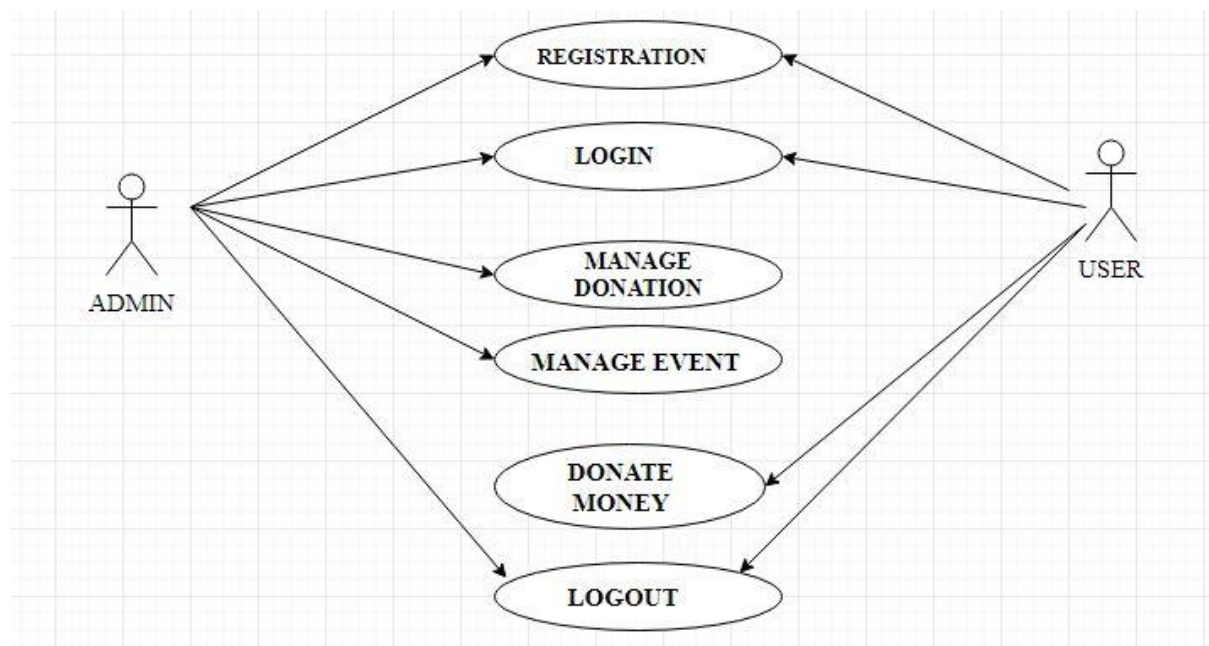
Every Algorithm must satisfy the following properties:

1. **Input**- There should be 0 or more inputs supplied externally to the algorithm.
2. **Output**- There should be at least 1 output obtained.
3. **Definiteness**- Every step of the algorithm should be clear and well defined.
4. **Finiteness**- The algorithm should have finite number of steps.
5. **Correctness**- Every step of the algorithm must generate a correct output.

### **4.4 USER INTERFACE DIAGRAM**

#### **USE CASE DIAGRAM**

Use case diagrams are usually referred to as behavior diagrams used to describe a set of actions (use cases) that some system or systems (subject) should or can perform in collaboration with one or more external users of the system (actors). Each use case should provide some observable and valuable result to the actors or other stakeholders of the system. UML 2.0 to 2.4 specifications also described use case diagram as a specialization of a class diagram, and class diagram is a structure diagram. Use case diagrams are in fact twofold - they are both behavior diagrams, because they describe behavior of the system, and they are also structure diagrams - as a special case of class diagrams where classifiers are restricted to be either actors or use cases related to each other with associations.



### **4.5 SECURITY ISSUE**

Security is a vital responsibility of all NGO personnel; while absolute security can never be guaranteed, threats and their associated risks can be mitigated and vulnerabilities reduced once identified and assessed. A **formalized** security risk management process is the key to accomplishing this vital function.

## ONLINE DONATION TO NGO

### 4.6 TEST CASES

Test Case ID	Test Scenario	Test steps	Test data	Expected Result	Actual Result	Pass/Fail
1	Check Admin login with a valid data/login page must be display	1 Enter username 2 Enter password 3 click login button	Username= Admin Password= Admin	User should login into application	User login into application	Pass
2	Check valid Username with invalid Password then invalid login page must be display	1 Enter username 2 Enter password 3 click login button	Invalid Data Username= Admin Password= Admin	Error msg (Plz Enter The Correct details)	Error msg display	Pass
3	Check invalid Username with valid Password then invalid login page must be display	1 Enter username 2 Enter password 3 click login button	Invalid Data Username= Admin123 Password= Admin	Error msg (login invalid)	Error msg display	Pass
4	Invalid both	1 Enter username 2 Enter password 3 click login button	Invalid both Username= Amin Password= Adm	Invalid both	Error msg display	Pass
5	Both are null	1 Enter username 2 Enter password 3 click login button	Username= Password=	Both are null	Error msg display	Pass