INTRODUCTION

1.1 PROJECT PROFILE

Relief Fund Management System a public fund constituted for the purpose of giving relief in cases of privation and also to those affected by fire, flood, cyclone sea erosion and similar other calamities. Educational Cultural and Charitable Institutions of a public nature, which are affected by such calamities and whose financial position does not enable them to repair the damage caused to their property and bring them to a normal condition are also eligible for financial assistance from the Fund. Relief may also be provided from the Distress Relief Fund for the workers thrown out of employment for the reason beyond their control, in units not covered by Employees State Insurance Schemes or Employees Provident Fund Schemes in the traditional sector alone. Financial Assistance from this Fund will be restricted to cases not covered by other special relief programs of Government. The term "Privation" does not refer to ordinary cases of poverty, but refers only to exceptional cases of real hardship due to sudden loss of income where the District Collector is satisfied that financial assistance would be necessary.

1.2 ORGANIZATION OVERVIEW

COMPANY PROFILE

D'Soft Solutions is a Software Development and Research Centre of De Paul Education Trust, a registered charitable Trust of the Vincentian fathers who are committed to the integral development of the less privileged members' of the society. It has long years of sound experience in software development field.

D'Soft Solutions has the prime objective of supplying efficient software solutions to various enterprises to make the best use of the offerings of Information Technology. It is organized in such a way as to provide customized and tailored solutions to specific industries, business concerns and other service institutions. The qualified and experienced personnel will provide training for operating the software and our after-sale service is guaranteed.

SYSTEM SPECIFICATION

2.1 SOFTWARE SPECIFICATION

Operating System	:	Windows 10
Front End	:	PHP, HTML,JavaScript
Back End	:	MySQL
Server	:	Wamp64
Software Used	:	Notepad++, WAMP Server 2.6

2.2 RECOMMENDED HARDWARE SPECIFICATION

Processor	:	Intel CORE i3 with 2GHz or higher Memory
Primary	:	512.0MB RAM or higher
Secondary	:	2.0GB hard disc or higher
Monitor	:	14”CRT or TFT or higher
Keyboard	:	104 K
Pointing device	:	2 or 3 button mouse
Printer	:	Dot Matrix or Ink Jet or Laser Printer

DEVELOPING TOOL

3.1 FRONT END

3.1.1 OVERVIEW OF PHP

PHP (recursive acronym for PHP: Hypertext Pre-processor) is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. Originally created by RasmusLerdorf in 1994, the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive backronym PHP: Hypertext Pre-processor.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management system and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.

The PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as a de facto standard. Since 2014 work has been on-going to create a formal PHP specification.

3.1.2 WAMP SERVER64

WampServer refers to a software stack for the Microsoft Windows Operating System, created by Romain Bourdon and consisting of the Apache web, OpenSSL for SSL support, MySQL database and PHP programming language.

WAMP Stands for "Windows, Apache, MySQL, and PHP." WAMP is a variation of LAMP for Windows systems and is often installed as a software bundle (Apache, MySQL, and PHP).It is often used for web development and internal testing, but

may also be used to serve live websites. The most important part of the WAMP package is Apache (or "Apache HTTP Server") which is used run the web server within Windows. By running a local Apache web server on a Windows machine, a web developer can test webpages in a web browser without publishing them live on the Internet.

WAMP also includes MySQL and PHP, which are two of the most common technologies used for creating dynamic websites. MySQL is a high-speed database, while PHP is a scripting language that can be used to access data from the database. By installing these two components locally, a developer can build and test a dynamic website before publishing it to a public web server. While Apache, MySQL, and PHP are open source components that can be installed individually, they are usually installed together. One popular package is called "WampServer," which provides a user-friendly way to install and configure the "AMP" components on Windows.

The acronym WAMP refers to a set of free (open source) applications, combined with Microsoft Windows, which are commonly used in Web server environments. The WAMP stack provides developers with the four key elements of a Web server: an operating system, database, Web server and Web scripting software. The combined usage of these programs is called a server stack. In this stack, Microsoft Windows is the operating system (OS), Apache is the Web server, MySQL handles the database components, while PHP, Python, or PERL represents the dynamic scripting languages.

3.2 BACK END

3.2.1 MYSQL

MySQL is the world's most popular open-source database. With its proven performance, reliability and ease-of-use, MySQL has become the leading database choice for web-based applications, used by high profile web properties including Facebook, Twitter, YouTube, Yahoo! and many more. Oracle drives MySQL innovation, delivering new capabilities to power next generation web, cloud, mobile and embedded applications.

MySQL AB was a software company that was founded in 1995. It was acquired by Sun Microsystems in 2008; Sun was in turn acquired by Oracle Corporation in 2010. MySQL AB is the creator of MySQL, a relational database management

system, as well as related products such as MySQL Cluster. The company was dually headquartered in Uppsala, Sweden and Cupertino, California with offices in other countries (Paris, Munich, Dublin, Milan, and Tokyo).

With around 400 employees in 25 countries, MySQL AB was one of the largest open source companies worldwide. Around 70% of the employees worked for MySQL from their home offices.

MySQL is an open source relational database management system (RDBMS) based on Structured Query Language (SQL). MySQL runs on virtually all platforms, including Linux, UNIX, and Windows. Although it can be used in a wide range of applications, MySQL is most often associated with web-based applications and online publishing and is an important component of an open source enterprise stack called LAMP. LAMP is a Web development platform that usesLinux as the operating system, Apache as the Web server, and MySQL as the relational database management system and PHP as the object-oriented scripting language. (Sometimes Perl or Python is used instead of PHP). MySQL, which was originally conceived by the Swedish company MySQL AB, was acquired byOracle in 2008. Developers can still use MySQL under the GNU General Public License (GPL), but enterprises must obtain a commercial license from Oracle.

3.3 OPERATING SYSTEM

Windows 10

Windows 10is an operating system produced by Microsoft for use on personal computers, including home and business desktops, laptops etc. Windows 10 was intended to be a more focused, incremental upgrade to the Windows line, with the goal of being compatible with applications and hardware with which Windows Vista was already compatible. Presentations given by Microsoft in 2008 focused on multi-touch support, a redesigned Windows shell with a new taskbar, referred to as the Super bar, a home networking system called Home Group, and performance improvement

1. Speed

Even aside from incompatibilities and other issues that many people had with Vista, one of the most straightforward was speed – it just felt too sluggish

compared to windows 8.1, even on pumped up hardware. Windows 10 brings a more responsive and sprightlier feel and Microsoft has spent a lot of time and effort getting the Start Menu response just right. Microsoft has also recognized the need for improved desktop responsiveness, which gives the impression that the computer is responding to the user and that they are in control – something that was often lacking with Vista. You can also expect faster boot times. And the boot sequence is now not only prettier than it was with Vista, but it's speedier too.

2.Compatibility

In simple terms, compatibility on Windows 10 will be far better than it was with Vista. Many programs that individuals and companies used on Windows 7 did not work immediately and required updates, but with Windows 7 almost all applications that work on Vista should still run. In essence, Windows Vista has done most of the hard work for Windows 7 in this respect.

3 Lower hardware requirements

Vista gained a reputation for making even the beefiest hardware look rather ordinary. Windows 10, however, will run well on lower end hardware, making the transition from Window 8.1 less painful. Microsoft is even pushing Windows 10 for net books. This could provide a modern replacement for Windows 8.1, which has found a new lease of life as the OS of choice on net books, supplanting Linux. The downside is that Windows 10 Starter Edition, as it will be called, will be limited to only three applications running at the same time.

SYSTEM ANALYSIS

INTRODUCTION

Information is a vital ingredient for the operations and management of any organization. Computers have become an essential part of organizational information processed. This project entitled "Relief Fund Management" is developed for D'Soft Computer education Angamaly. This project contains modules.

- Registration and login
- Admin module
- User module (Donor, applicant)
- Request application
- Verification & approval process
- Application status
- Add emergency assistance scheme
- Payment
- Generate certificate/RECEIPT

We have developed this project by using PHP as front end and MYSQL as back end. We hope that this system will make the task of user easier.

4.1 EXISTING SYSTEM

- Existing System is sometimes complex. The study of existing system helps to gather as much information as possible about the system. The existing system runs in an efficient way but less interactive. The proposed system is much more user interactive and user friendly. The field such as donation is made one time and is cost effective. In the existing system difficult to find donor.

4.2 LIMITATION OF EXISTING SYSTEM

- In the existing system difficult to find donor
- No certificate or receipt for donation
- Fake applicants

The existing manual system is facing this much problems. While using the existing system we are losing our valuable time, effort, cost and human resources. Proposed system can eliminate the drawbacks of the existing system to a great extent. Making the payment in online can attract users and make them easy to use at all time.

4.3 FEASIBILITY STUDIES

All projects are feasible when given unlimited resources and infinite time .It is both necessary and prudent to evaluate the feasibility of a project at the earliest possible time. A feasible study is not warranted for system in which economic justification is observed, technical risk is low, few legal problems are expected and no reasonable alternative exists. An estimate is made of whether the identified user needs may be satisfied using our recent software and hardware technologies. The study will decide if the proposed system will be cost effective, from the business point of view and it can be developed in the existing budgetary. The feasibility study should be relatively sharp and quick .The gesture should inform the decision of whether to go ahead with a more detailed analysis.

Feasibility study may be documented as a separated report to higher officials of the top level management and can be included as appendices to the system specification. Feasibility and risk analysis is detailed in many worries. If there is project risk then the feasibility of producing the quality software is reduced .The study is done in three phases.

4.3.1 Operational Feasibility

4.3.2 Technical Feasibility

4.3.3 Economical Feasibility

4.3.1 OPERATIONAL FEASIBILITY

The purpose of the operational feasibility study is to determine the whether the new system will be used if it is developed and installed. And whether there will be resistance from users that will undermine the possible application benefit. The first challenge was whether the system meets the organizational requirements. This is checked by the system requirement collected from the users and the management and the operational feasibility proved that the system is capable to meet its functional requirements.

The developed system is completely driven and user friendly. In tour booking operational feasibility is dependent on human resources available for the project

and involves projecting whether the system will be used if it is developed and implemented. Operational feasibility is a measure of how proposed system solves the problem and how it satisfies requirements identified in the requirement analysis phase of system development.

4.3.2 TECHNICAL FEASIBILITY

The technical feasibility study is a study of function, performances and constraints and improve the ability to create an acceptable system. Technical feasibility is frequently the most difficult are to achieve at the stage of product engineering process. The system must be evaluated from technical viewpoint first. The assessment of this feasibility must be based on the outline design of the system requirements in the terms of inputs, outputs program procedure and staffs. This project is said to be technically feasible. Technical feasibility centers on the existing computer systems and extend to which it can support the proposed system. This involves financial consideration to technical enhancements.

4.3.3 ECONOMICAL FEASIBILITY

Economic analysis is the most frequently used method for evaluating the effectiveness of the proposed system. It evaluates whether the system benefits greater than cost. The proposed tour booking system is an effective one since the benefits of the software outweigh the cost incurred in installing it. It can be developed under optimal expenses with the available hardware and software.

4.4 PROPOSED SYSTEM

Proposed Beauty Parlor system is very cost effective. Proposed system can eliminate the drawbacks of the existing system to a great extent. The registration procedure is very simple though all users have to make the registration in online. The profile updating procedure is very easy. The proposed system is developed to clear out all the disadvantages faced in the existing system, it is fully computerized. The main advantage are high performance, faster processing, proper security and reduces the manual work

4.5 THE ADVANTAGES OF PROPOSED SYSTEM ARE:

- Reduce the paper work
- Easy to add an application
- Verified applications
- Track the application status
- Donation exempted from income tax
- Various modes of payment For contribution

SYSTEM DESIGN

INTRODUCTION

System design is the solution to the creation of a new system. This phase is composed of several systems. This phase focuses on the detailed implementation of the feasible system. System design has two phases of development logical and physical design. During logical design phase the analyst describes inputs (sources), out puts (destinations), databases (data sores) and procedures (data flows) all in a format that meats the uses requirements. Design goes through the logical and physical stages of development. At an early stage in designing a new system, the system analyst must have a clear understanding of the objectives, which the design is aiming to fulfill. Second input data and master files (database) have to be designed to meet the requirements of the proposed output. The operational (processing) phases are handled through program construction and testing. The system design includes:

- Modular design
- Input design
- Output design
- Database design
- Form design

5.1 MODULAR DESIGN

A software system is always divided into several sub systems that makes it easier for the development. A software system that is structured into several subsystems makes it easy for the development and testing. The different subsystems are known as the modules and the process of dividing an entire system into subsystems is known as modularization or decomposition. The system under consideration has been divided into several modules taking in consideration the above-mentioned criteria. Modules are:

- Registration and login
- Admin module
- User module (Donor, applicant)
- Request application
- Verification & approval process
- Application status

- Add emergency assistance scheme
- Payment
- Generate certificate/RECEIPT

Registration module

The Applicant and Donor can register to this website by providing his or her basic details such as name, contact number, email id, pancard number aadhar number etc… the user can also provide the username and password.

Login module

Common for all users. All users can login using email id and password. After login process each user will move to their corresponding pages.

Manage Application

The admin can add and modify the all information's regarding entire Application submitted by applicant and manage donation. Also adding emergency funds.

Set Offers

The Customers can see available offers while they booking for an appointment. Also providing the seasonal ,monthly and weekly offers.

Booking module

The registered users can search the services details and price. Also they can book the services which they need.

Donation module

The booked customers can take payment through online. The details payment is stored in cart and the status is shown.An advance payment is necessary for booking a service, the payment options are available in the site.

Report Module

In this session, the admin can view applicant details and donation in particular periods. Also donor can be view their transaction. This website is under income tax session, So donor can be get a certificate after a successful donation

5.2 INPUT DESIGN

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data into a usable form for processing

data entry. The activity of putting data into the computer for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system.

- What data should be given as input?
- How the data should be arranged or coded.
- The dialogue to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error Occur.

Login

This feature used by the user and admin to login into system. Users required username and password.

Emergency Category Details

Admin add the emergency category details. It includes the name of category only.

Emergency Sub Category

Admin can be adding sub category. It includes amount need for charities and set a time limit.

5.3 OUTPUT DESIGN

Computer output is the most important and direct information source to the user. Output design is a process that involves designing necessary outputs in the form of reports that should be given to the users according to the requirements. Efficient, intelligible output design should improve the system's relationship with the user and help in decision making. So, while designing output the following things are to be considered.

- Determine what information to present
- Arrange the presentation of information in an acceptable format
- Decide how to distribute the output to intended receipts

View Registration details

Admin can view the registered applicant and donor along with their necessary details.

View Application details

Admin can view the application details that submitted by applicant and admin can also manage their application. He can right to approve or reject the application submitted by applicant

5.4 DATABASE DESIGN

Data Base design is the logical form of design of data storage in the form of records in a particular structure in the form of tables with fields which is not transparent to the normal user but it actually acts as the backbone of the system. As we know database is a collection of which helps the system to manage and store data is called database management system. Data base management system builds some form of constraints like integrity constraints, i.e., the primary key or unique key and referential integrity which help to keep data structure storage and access of data from tables efficiently and accurately and take necessary steps to concurrent access of data and avoid redundancy of data in tables by normalization criterions.

Normalization is the method of breaking down complex table structures into simple table structures by using certain rules thus reduce redundancy and inconsistency and disk space usage and thus increase the performance of the system or application which is directly linked to the database design and also solve the problems of anomalies. There are different forms of normalization, some are:

- First normal form (1NF)
- Second normal form (2NF)
- Third normal form (3NF)
- Boyce code normal form
- Forth normal form (4NF)
- Fifth normal form (5NF)

The data base design of the new system is in Second normal form and every non-key attribute is functionally depend only on the primary key. The master and transaction tables and their structure are shown below.

5.5.1 Table Description

Table Name: Admin

Table Description: Admin login details

Field Name	Datatype	Constraints	Description
Id	Int	Primary key	Unique id
Username	Varchar (25)	Not null	Username unique
password	Varchar(50)	Not null	Encrypted password
Date	Timestamp	Not null	Created date

Table Name: Registration

Table Description: Registration Details for customer

Field Name	Datatype	Constraints	Description
Id	Int(10)	Primary key	registration id
Email	Varchar(20)	Not Null	Email address
Password	Varchar(50)	Not Null	Encrypted password
Phone	Varchar(10)	Not Null	Phone number
Fname	Varchar(20)	Not Null	First name
Lname	Varchar(20)	Not Null	Last name

Aadhar	Varchar(15)	Not Null	Aadhar number
pancard	Varchar(10)	Not Null	Pancard
Date	Timestamp	Not Null	Created date

Table Name: Applicant

Table Description: Employee details

Field Name	**Datatype**	**Constraints**	**Description**
Id	Int(10)	Primary key	Unique id for applicant
Reg_id	Int(10)	Foreign key	Registration id
Dob	Date	Not Null	Date of birth
Gender	Varchar(5)	Not Null	Gender
Address	Varchar(50)	Not Null	Address of Applicant
Pincode	int(8)	Not Null	Pin code
Village	Varchar(15)	Not Null	Village office
District	Varchar(25)	Not Null	District
Occupation	Varchar(25)	Not Null	Occupation
Income	int(11)	Not Null	Income
Status	Int	Not null	Status of Applicant
Date	Timestamp	Not noyll	Created date

Table Name: Bank

Table Description: Category details

Table Name: Payment

Field Name	Datatype	Constraints	Description
Id	Int(10)	Primary key	Unique id
Reg_id	int(11)	Not Null	Registration id
Bank	Varchar(20)	Not Null	Name of Bank
accnum	Int(20)	Not null	Account Number
Ifsc	Varchar(20)	Not null	IFSC code
Passbook	Varchar(50)	Not null	Passbook scanned copy
Date	Timestamp	Not null	Created date

Table Description: Sub Category details

Field Name	Datatype	Constraints	Description
Id	Int(10)	Primary key	Unique id payment
Biil_no	Int (10)	Not Null	Bill Number
Application_id	Int (10)	Foreign key	Application id
Donor_id	Int (10)	Foreign key	Donor id
Name	Varchar(25)	Not Null	Name on the card
Amount	Int(10)	Not Null	Amount
Date	Timestamp	Not Null	Created Date

Table Name: emgcategory

Table Description: booking details

Field Name	Datatype	Constraints	Description
Id	Int(10)	Primary key	Emergency Category id
Category	Varchar(20)	Not Null	Category Name
Date	Timestamp	Not Null	Created Date

Table Name: emgsubcategory

Table Description: Payment details

Field Name	Datatype	Constraints	Description
Id	Int(10)	Primary key	Subcategory id
Category_id	Varchar(25)	Foreign key	Category id
Subcategory	Varchar(25)	Not Null	Subcategory Name
Amount	Int(10)	Foreign key	Amount
Image	Varchar(50)	Not Null	Image
Video	Varchar(50)	Foreign Key	Video
status	Int(1)	Not null	Status of Application
date	Timestamp	Not null	Created date

5.5FORM DESIGN

Login form

Users can login to the system by using the username and password.

Registration Form

New users can interact with the system only after registration. The user will get a username and password through this process.

Add emgcategory

Admin will add the details about emgcategory.

Add Sub category

Admin will add the details about categoryunder subcategory.category,subcategory names such as, description, and price.

Book items

User can book the items which they need.

DESIGN TOOL

6.1 UML Diagrams

UML Approach

UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed. Software design is a process that gradually changes as various new, better and more complete methods with a broader understanding of the of the whole problem in general come into existence. There are various kinds of methods in software design. They are as follows:

- **Use case diagram**
- **Activity diagram**
- **Sequence diagram**
- **Class diagram**

Use case Diagrams:

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what's called an actor. An actor represents an outside entity- either human or technological. Use case diagrams can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do. Use case Diagram consists of use cases and actors and shows the interaction between the use case and actors. The purpose is to show the interactions between use cases and actor. To represent the system requirements from user's perspective. It must be remembered that the use-cases are the functions that are to be performed in the module. An actor could be the end-user of the system or an external system.

Activity Diagram:

The purpose is to show the activities which the users performed. Actives are shown parallel and sequentially in which order they are performed. Some activities are joined and split according to its flow. Flow of data is represented using arrows.

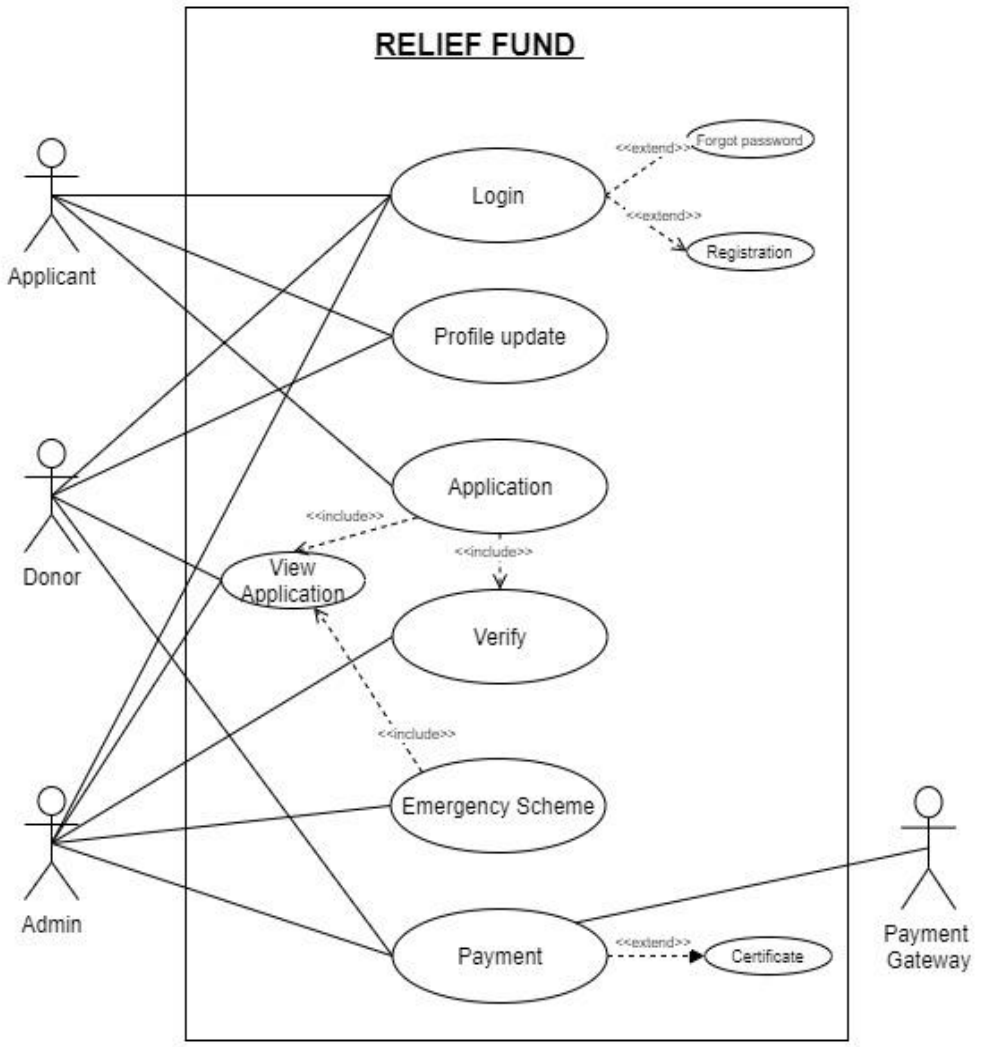
Sequence Diagram:

The purpose is to show the sequential flow through of activities. In other Words, we call it mapping processes in terms of data transfers from the actor through corresponding objects. To represent the logical flow of data with respect to a process. It must be remembered that the sequence diagram display objects and not the classes.

Class Diagram*:*

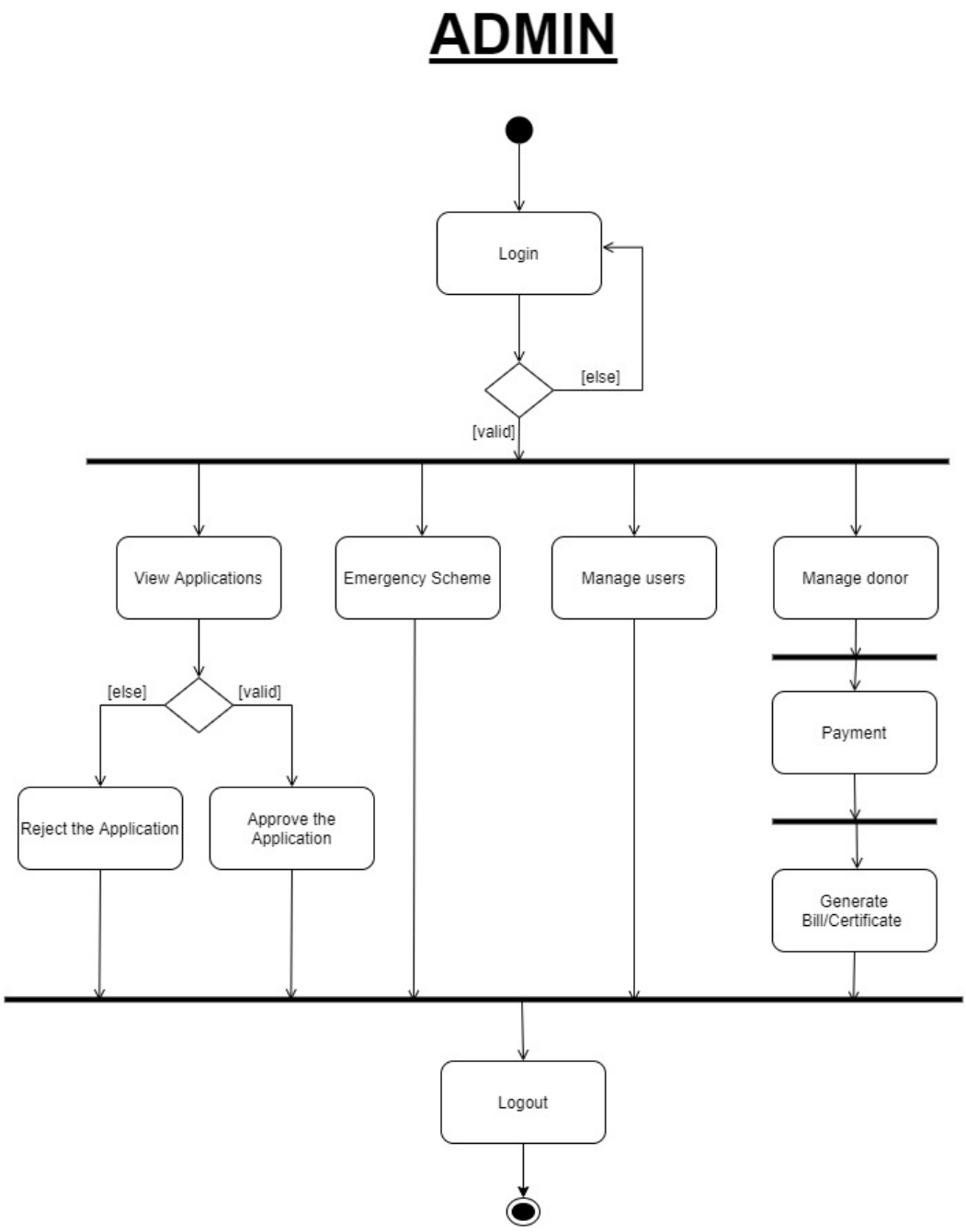
This is one of the most important of the diagrams in development. The diagram breaks the class into three layers. One has the name, the second describes its attributes and the third its methods. The private attributes are represented by a padlock to left of the name. The relationships are drawn between the classes. Developers use the Class Diagram to develop the classes. Analyses use it to show the details of the system. Architects look at class diagrams to see if any class has too many functions and see if they are required to be split.

USECASE DIAGRAM:

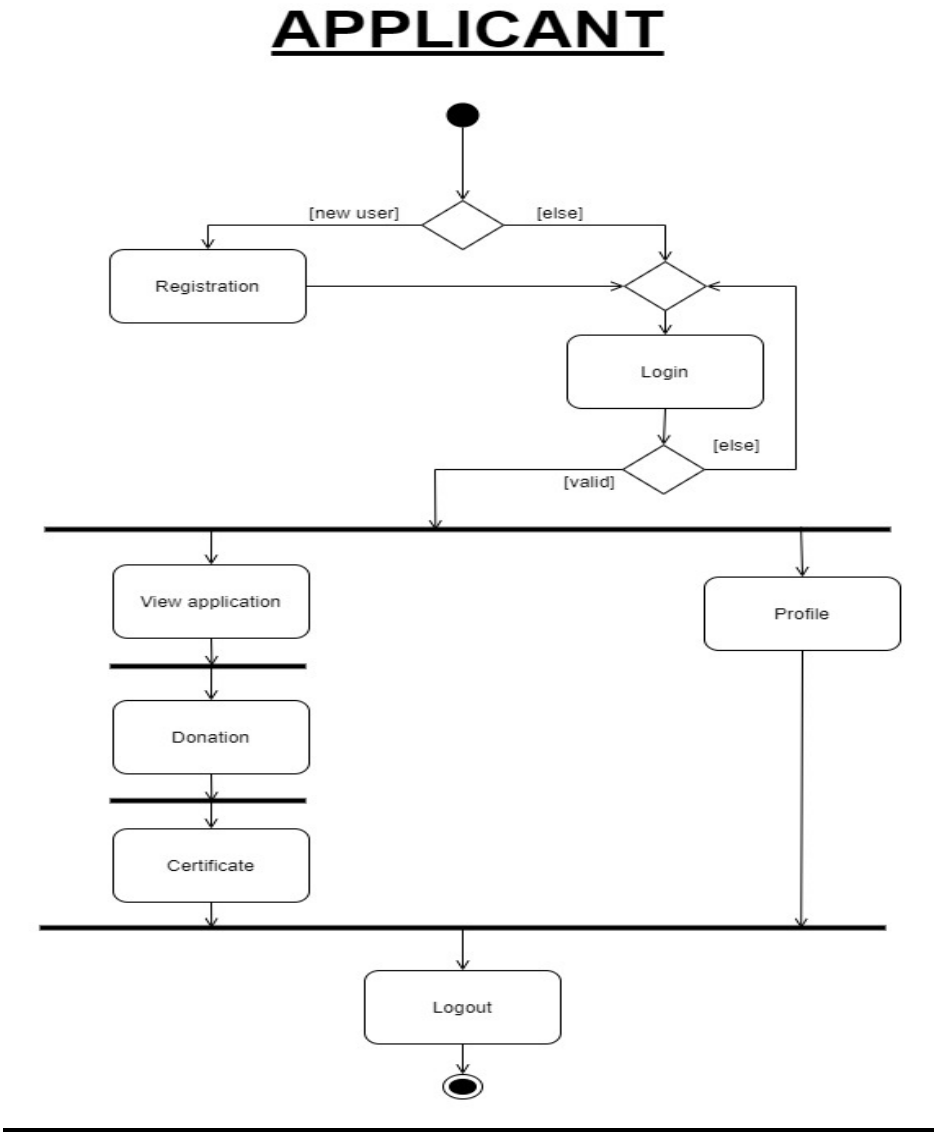


ACTIVITY DIAGRAM

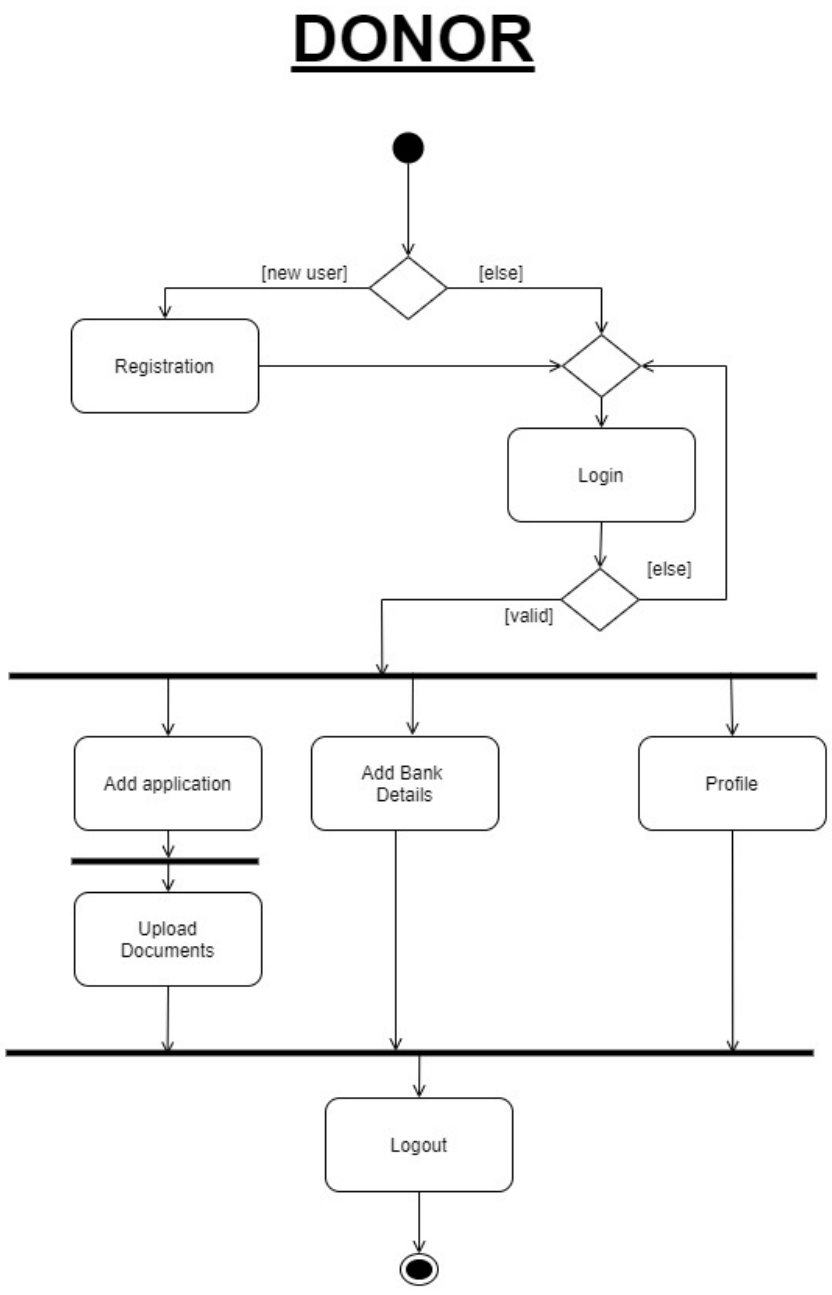
a) ADMIN



b) **APPLICANT**

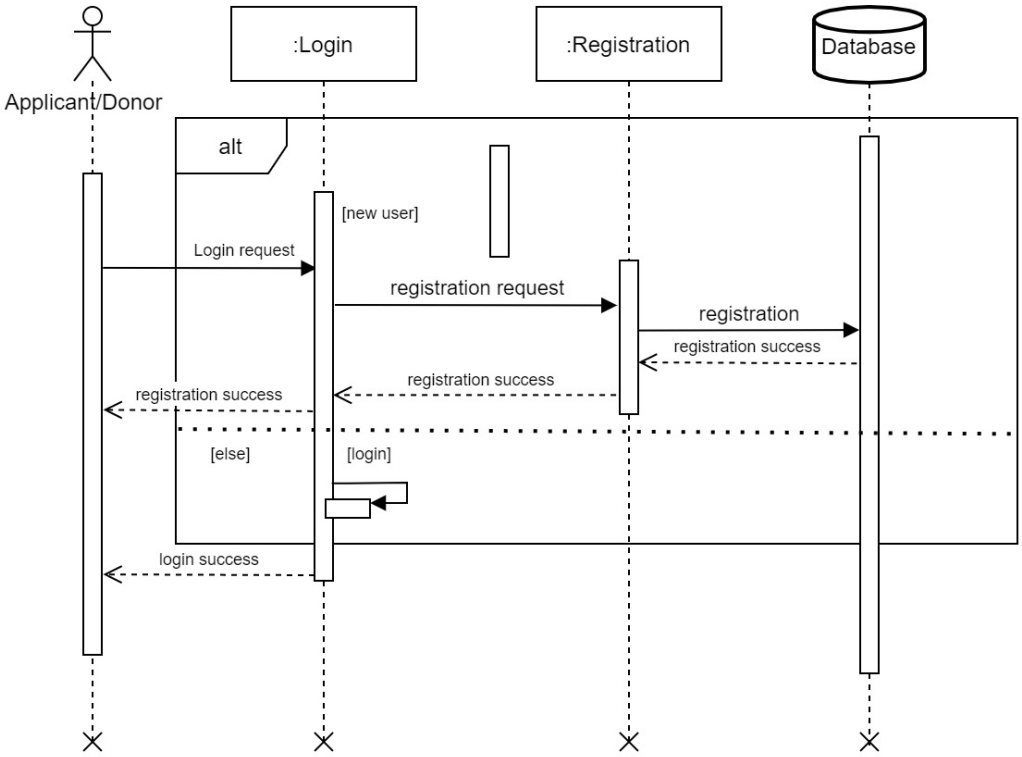

APPLICANT
[new user]
[else]
Registration
Login
[else]
[valid]
View application
Profile
Donation
Certificate
Logout

c) DONOR

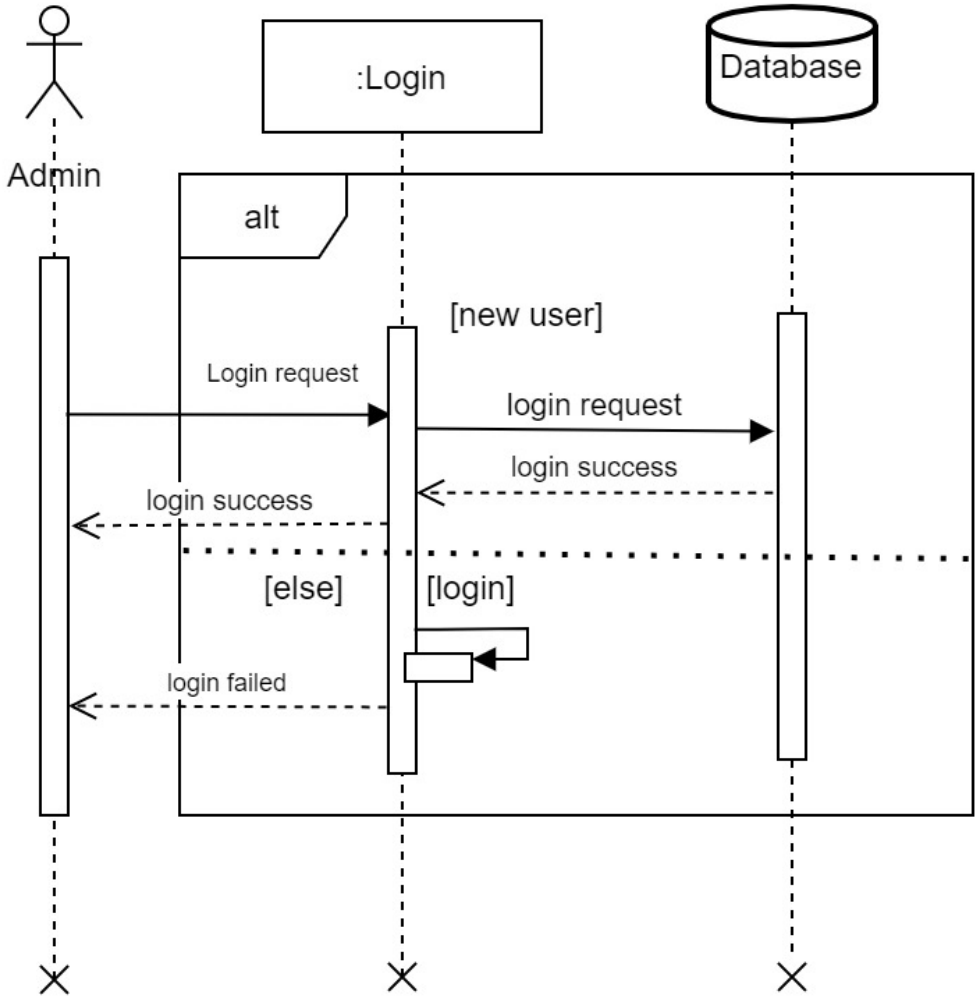

DONOR
[new user]
[else]
Registration
Login
[else]
[valid]
Add application
Add Bank Details
Profile
Upload Documents
Logout

SEQUENCE DIAGRAM

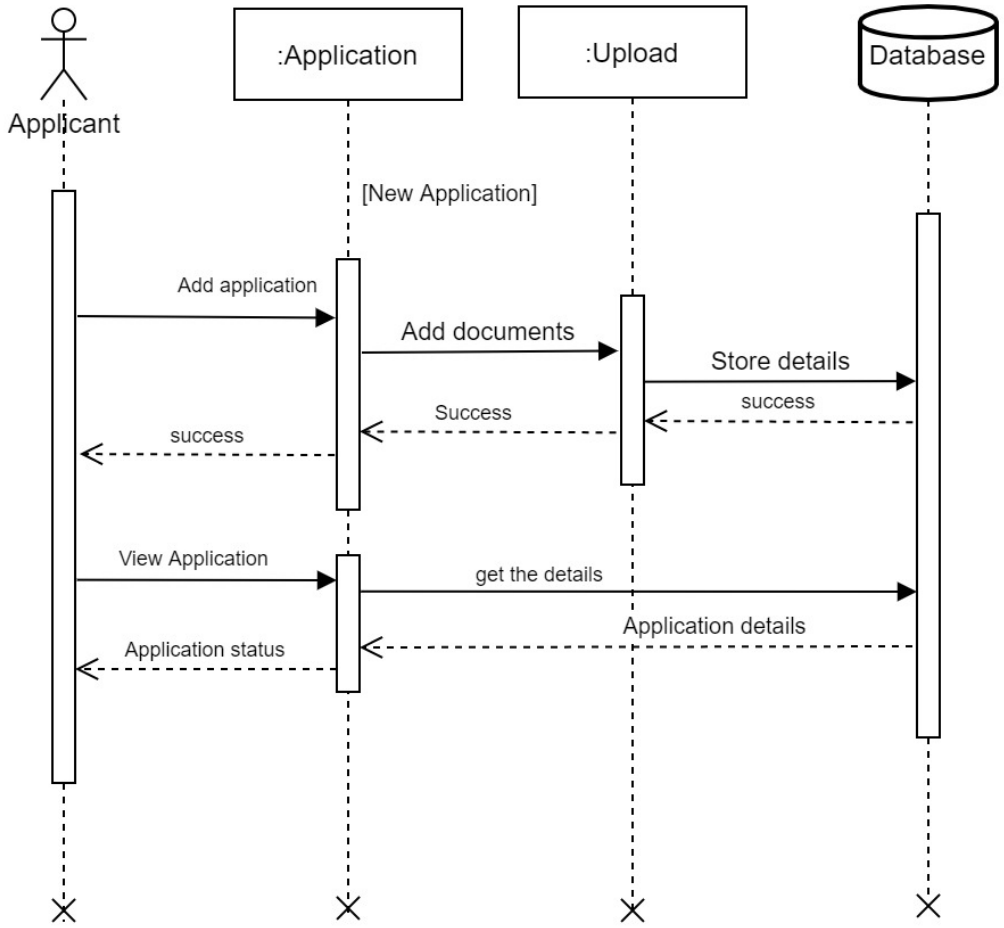
a) SEQUENCE DIAGRAM FOR DONOR AND APPLICANT LOGIN



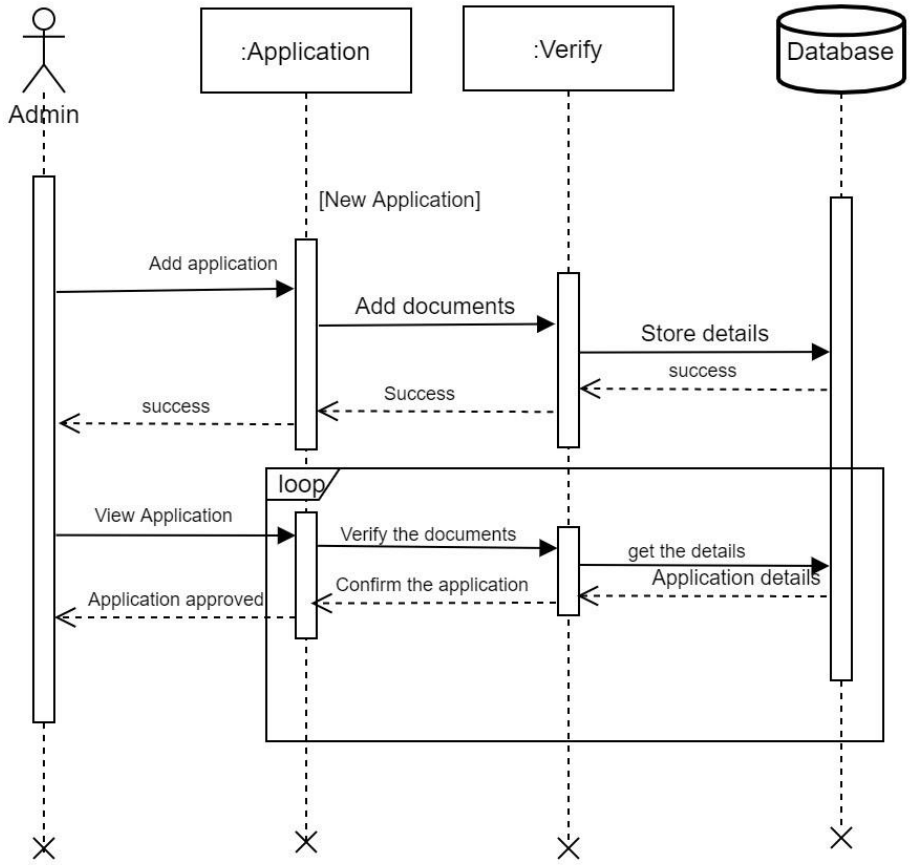
b) SEQUENCE DIAGRAM FOR ADMIN LOGIN



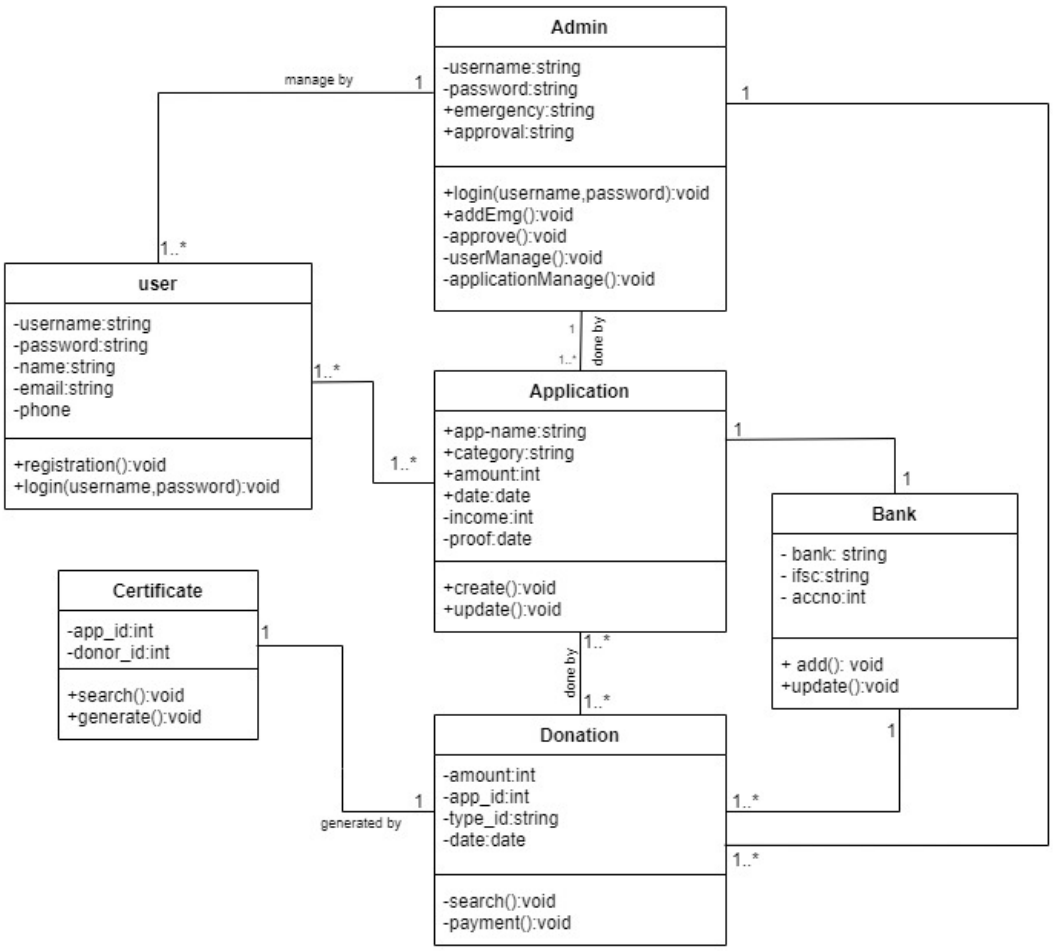
c) SEQUENCE DIAGRAM ADD APPLICATION



d) SEQUENCE DIAGRAM VERIFY APPLICATION BY ADMIN



CLASS DIAGRAM



SYSTEM TESTING

INTRODUCTION

Software testing is a critical element of software quality assurance and represents the ultimate previous of specification, design and coding. The testing phase involves the testing of the development system using test data. While testing the system using these test data, errors were found and corrected. Thus, a series of test were performed to the system before the system was ready for implementation. Testing enhances the integrity of a system by identifying the deviations in the design and development of the expected end product.

It should focus more on the error-prone areas of the application. This helps in the prevention of errors in the system and builds confidence that the system will work without error testing. It is the process of executing a program with the intent of finding an error. Testing also adds value to the product by confirming to the user requirements. Testing verifies that software deliverable conforms precisely to the functional and design specifications that have been compiled during analysis and design phases. A good test case is the one that has the high probability of finding an as yet undiscovered error.

There are different types of system testing methods and some of them are:

7.1 UNIT TESTING

The unit test phase entails converting the design language in to program code and, most important, designing and carrying out tests of the individual units. Once individual modules or units have been tested and accepted, the integration and test phase begin. This initial part of structural testing corresponds to some quick checks that a developer performs before subjecting the code to more extensive code coverage testing or code complexity testing. The developer can perform certain obvious tests knowing the input variables and the corresponding expected output variables. This can be a quick test that checks out any obvious mistakes. This can even be done prior to formal reviews of static testing so that the review mechanism does not waste time.

Unit testing is undertaken when a module has been created and successfully reviewed. In order to test a single module we need to provide a complete environment i.e. besides the module we would require

- The procedures belonging to other modules that the module under test calls
- Non local data structures that module accesses
- A procedure to call the functions of the module under test with appropriate parameters

In my project each module are separated and tested. That means in the admin side and customer sides are separately tested. Check the duplication of data and the duplication is removed. And ensure that the updating are recorded correctly.

Test for the admin module

• Testing admin login form: -This form is used for log in of administrator of the system.In this we enter the username and password. If both are correct admin home page will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for correct or valid username and password.

Test for Customer login module

• Test for Customer login form-This form is used for log in for Customers. In this we enter the username and password. If all these are correct Customer homepage will open otherwise if any of data is wrong it will get redirected back to the login page and again ask for username and password.

7.2 INTEGRATION TESTING

It tests the integration of each module in the system. It also tests to find discrepancies between the system and its original objective. In this testing analysis we are trying to find areas where modules have been designed with different specification for data length, type etc. In my project integration testing is performed after unit testing. In unit

testingit ensure that each module is working properly.In integration testing ensure that complete system working properly. In this project we ensure that the user side and admin side together work properly.

First of all, the admin log in to the system and enters all necessary details such as category,corresponding category name, category image, herbs name, fruit name, description, priceetc. These details are entered so that it is helpful for the user i.e. the customer's registration.

Then the customerlog in to the system and enters their personal details, email & password. They have a provision to edit their profiles and view their own account.

7.3 VALIDATION TESTING

Validation refers to the process of using software in a live environment in order to find errors. During the course of validating the system, failure may occur and sometimes the coding has to be changed according to the requirement.

- Username entry check
  - Ensure that the user must enter a valid username.
- Password entry check
  - Ensure that the user must enter a valid password.
- Numeric and character check to corresponding fields
  - Ensures that the user must enter the numeric and character value in the specified field
- Required Field Validator
  - Ensures that the user does not skip an entry
- Regular Expression Validator
  - Entry matches a pattern defined by a regular expression. This type helps you to check for predictable sequences of characters, such as those in social security numbers, email addresses, telephone numbers, and postal codes and so on.

7.6 TEST CASES

A test case is a set of sequential steps to execute a test operating on a set of predefined inputs to produce certain expected outputs. There are two types of test cases: - manual and automated. Manual test case is executed manually while an automated test case is executed using automation.

TEST CASE FOR LOGIN:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to login	When no id is entered	When the valid username entered	An error message the id field is required	Success
2	Try to login	When no password is entered	When the valid password entered	An error message the password field is required	Success
3	Try to login	When no id and password entered	When the valid username and password entered	An error message please enter the details	Success

TEST CASE FOR ADDINGCATEGORY DETAILS:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to add data to application	When no details entered	When the valid Details entered	An error message the field is required	Success
2	Try to add data to application	When already existing data entered	This data is already existed	An error message please enter a valid data	Success

TEST CASE FOR ADDINGSUB CATEGORY:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to add data to application	When no details entered	When the valid Details entered	An error message the field is required	Success
2	Try to add data to application	When no details entered	This data is already existed	An error message please enter a valid data	Success

TEST CASE FOR ADDING OFFERS:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to Add offers details	When no details entered	When the valid details entered	An error message the field is required	Success
2	Try to add offer details	On clicking check button	Entered information are not valid	An error message you have to enter a valid image	Success
3	Try to add offer details	On clicking submit button	Displays a message details are added successfully	Displays a message details are added successfully	Success

TEST CASE FOR BOOKING AN ITEM:

ID	TEST CASE	INPUT DATA	EXPECTED RESULT	ACTUAL RESULT	REMARK
1	Try to book an item	When no details entered	When the valid Details entered	An error message the field is required	Success
2	Try to book an item	On clicking check button	Nomore services remaining in the package	An error message you have to try again with	Success

				another package	
3	Try to book an item	On clicking book button	Displays a message details are save successfully	Displays a message details are save successfully	Success

SYSTEM IMPLEMENTATION

8.1 SYSTEM IMPLEMENTATION

A crucial phase in the system life cycle is the successful implementation of the new system design. Implementation simply means converting a new system design into operation. This involves creating computer compatible files, training, and telecommunication network before the system is up and running. A crucial factor in conversion is not disrupting the functioning of organization. Actual data were input into the program and the working of the system was closely monitored. It is a process of converting a new or revised system into an operational one. It is the essential stage in achieving a successful new system because usually it involves a lot of upheaval in the user. It must therefore be carefully planned and controlled to avoid problems.

The implementation phase involves the following tasks:

- Careful planning.
- Investigation
- Design of methods
- Training of the staff in the changeover phase.
- Evaluation of changeover.

We implemented this new system in parallel run plan without making any disruptions to the ongoing system, but only computerizing the whole system to make the work, evaluation and retrieval of data easier, faster and reliable.

8.2 TRAINING

System implementation is the process of making the newly designed system fully operational and consistent in performance. The logical miss-working the system can be identified if any. Various combinations of test data were feed. Each process accuracy/reliability checking was made. After the approval, the system was implemented in the user department.

The preparation of implementation of documentation process is often viewed as total sum of the software documentation process. In a well-defined software development environment, however the presentation of implementation documents is essentially an interactive process that synthesis and recognizes document items

that were produced during the analysis and design phase for the presentation to user. The following are the three types of implementation documents. Conversion Guide

- User Guide
- Operation Guide

Conversion Guide

The Conversion Guide phase of the implementation, process the tasks that are required to place the system into an operation mode. They amplify the conversion lane that was defined during the internal design phase and defines file conversion, file creation and data entry requirements.

User Guide

The system application and operation functions describe the overall performance capabilities of the system and define procedures the user must follow to operate the system. In the realm of information system, the content of a user guide must be developed to coincide with a criterion that defines the characteristics of one of the following methods of data processing

- Off-line processing
- Direct access processing

Operation Guide

The function of an operation is to define the control requirements of an online health shopping website and provide instruction for initializing, running and terminating the system. The items contained in an operation guide may be grouped as follows.

- General information
- System overviews
- Run description

8.3 POST IMPLEMENTATION

This project is very useful for the customers for ordering herbs and fruits which will be a solution for their diseases. Currently the system has the only option for book the items. In future we can add more facilities for the users. As a future enhancement we can add doctor

consultancy and book hospitals to this project.Sothat the user can consult doctor about their diseases. Also, the users will have the option to take appointment in particular hospital.

Also, this project can have a large variety of updations like chat between users so that negotiation can take place. Apart from keeping a personnel log we are planning to extend the system to incorporate a shared expense group. We are planning to include a service so as to make the direct cash payment within the application itself.

MAINTENANCE AND REVIEW

9.1 SYSTEM MAINTENANCE

Maintenance is making adaptation of the software for external changes (requirements changes or enhancements) and internal changes (fixing bugs). When changes are made during the maintenance phase all preceding steps of the model must be revisited.

9.2 TYPES OF MAINTENANCE

In my project online tour booking website there are three types of maintenance:

- Corrective (Fixing Bugs/errors)
- Adaptive (Updates due to environment changes)
- Perfective (Enhancements, requirements change

Corrective maintenance: removes software faults.

Perfective maintenance: improves the system without changing its functionality. The objective of perfective maintenance should be to prevent failures and optimize the software.

Adaptive maintenance: modifies the software to keep it up to date with its operative environment. It may be needed because of changes in the user requirements, changes in target platform, or changes in external interfaces. Minor adaptive changes should be handled by normal maintenance process. Major adaptive changes should be carried out as a separate development project.

SYSTEM EVALUVATION

10.1 SYSTEM EVALUVATION

System evaluation is the process of assessing the performance of a complete online health shopping website to discover how it is likely to perform in live scenarios. Systems developers spend a proportion of their trading time trying to improve their existing systems and looking for new and different ideas. It is essential that any proposed system or change is fully evaluated before being used in live situation. System evaluation brings your complete system into play.

CONCLUSION

11.1 CONCLUSION

The project entitled "Relief Fund Management" has been successfully engineered and the ability of the system to work has verified. The aim of the project which is to eliminate the drawback of the existing system has been achieved. This system will speed up every manual operation. The high amount of manual labor and time has been reduced. The system also provides descriptive and precise report through the system.

Hence an attempt of automating an office application had added to our learning experience. it has also helped in adopting an analytical approach to solving and made us realize that system development is a step-by-step process, the system helps to manage both customers and staffs in satisfied manner during scheduling both appointment and worktime. This project 'online beauty parlor management system' is developed using front end PHP and MySQL database is back end.

APPENDIX

12.1 SAMPLE CODE

- **)Registration (registration.php)**

```
<?php
include 'database.php';
$data = new Databases;
if(isset($_POST["register"]))
{
$insert_data = array(
'fname' => mysqli_real_escape_string($data->con, $_POST['fname']),
'lname' => mysqli_real_escape_string($data->con, $_POST['lname']),
'email' => mysqli_real_escape_string($data->con, $_POST['email']),
'phone' => mysqli_real_escape_string($data->con, $_POST['phone']),
'password' => md5(mysqli_real_escape_string($data->con,
$_POST['password'])),
'pancard' => mysqli_real_escape_string($data->con, $_POST['pancard']),
'status' =>mysqli_real_escape_string($data->con, $_POST['status']),
'aadhaar' => mysqli_real_escape_string($data->con, $_POST['aadhaar']),
);
if($data->insert('registration', $insert_data))
{
echo '<script type="text/javascript">';
echo 'alert("Registration Done Successfully");';
echo 'window.location.href = "index.php";';
echo '</script>';
}
}
?>
<!DOCTYPE html>
<html class="no-js">
<head>
<title>Registration</title>
</head>
```

```
<body>
<form method="post" name="vform" id="vform">
<div class="row">
<div class="form-group col-md-12">
<label>Register As</label>
<br>
<select class="form-control" name="status" id="status" required>
<option value="">Choose Type Of User</option>
<option value="0">Applicant</option>
<option value="1">Donor</option>
</select>
</div>
</div>
<div class="row">
<div class="form-group col-md-6">
<label>First Name</label>
<input type="text" name="fname" id="fname" class="form-control" placeholder="First Name*" autofocus data-error="First Name Is Required">
</div>
<div class="form-group col-md-6">
<label>Last Name</label>
<input type="text" name="lname" id="lname" class="form-control" placeholder="Last Name*" >
</div>
</div>
<div class="row">
<div class="form-group col-md-6">
<label>Email Address</label>
<input type="text" name="email" id="email" class="form-control" placeholder="E-mail*">
<div id="email_error"></div>
```

```
</div>
<div class="form-group col-md-6">
<label>Phone Number</label>
<input type="text" name="phone" id="phone" class="form-control" placeholder="Phone Number*">
</div>
</div>
<div class="row">
<div class="form-group col-md-6">
<label>Password</label>
<input type="password" name="password" id="password" class="form-control" placeholder="Password*" >
</div>
<div class="form-group col-md-6">
<label>Confirm password</label>
<input type="password" name="cpassword" id="cpassword" class="form-control" placeholder="Confirm Password*">
</div>
</div>
<div class="row">
<div class="form-group col-md-6">
<label>Aadhaar Number</label>
<input type="text" name="aadhaar" id="aadhaar" class="form-control" value="" placeholder="Aadhaar Number*">
</div>
<div class="form-group col-md-6">
<label>Pancard Number</label>
<input type="text" name="pancard" id="pancard" class="form-control" value="" placeholder="Pan Card Number*">
</div>
</div>
<div class="form-group">
```

```
            <button name="register" id="register" class="btn btn-
            primary">Register</button>
            </div>
            <div class="clearfix"></div>
            </form>
            </body>
            </html>
```

.) Login(Login.php)

```
<?php
include 'database.php';
$data = new Databases;
session_start();
if(isset($_SESSION["email"]))
 {
    $email = $_SESSION['email'];
    $condition_arr = array(
                  'email'=>$email,
                );
    $result = $data->getData('registration','*',$condition_arr);
    if ($result > 0)
    {
     $_SESSION['email'] = $email;
     if($result[0]['status'] == 0)
     {
      header("location:home.php");
     }
     else if ($result[0]['status'] == 1)
     {
      header("location:donor.php");
     }
      else if ($result[0]['status'] == 2)
     {
      header("location:Admin");
     }
    }
    else
    {
     header("location:index.php");
```

```
    }
 }
 if(isset($_POST["login"]))
  {
    $email = mysqli_real_escape_string($data->con, $_POST['email']);
    $password = md5(mysqli_real_escape_string($data->con,
$_POST['password']));
    $condition_arr = array(
                    'email'=>$email,
                    'password'=>$password,
                   );
    $result = $data->getData('registration','*',$condition_arr);
    if ($result > 0)
    {
     $_SESSION['email'] = $email;
     if($result[0]['status'] == 0)
     {
      header("location:home.php");
     }
     else if ($result[0]['status'] == 1)
     {
      header("location:donor.php");
     }
      else if ($result[0]['status'] == 2)
     {
      header("location:Admin");
     }
    }
    else
    {
     echo "<script>alert('Invalid Username or Password')</script>";
    }
  }
 ?>
<!DOCTYPE html>
<html class="no-js">
   <head>
      <title>SADAKA | Charity / Non-profit responsive Bootstrap HTML5
template</title>
   </head>
   <body>
     <div class="modal fade" id="loginModal" tabindex="-1" role="dialog" aria-
labelledby="loginModalLabel" aria-hidden="true">
```

```
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <button type="button" class="close" data-dismiss="modal" aria-
label="Close"><span aria-hidden="true">&times;</span></button>
          <h4 class="modal-title" id="loginModal">LOGIN FORM</h4>
        </div>
        <div class="modal-body">
            <form method="post">
                  <div class="row">
                     <div class="form-group col-md-12">
                        <label class="contact-icon">USER NAME</label>
                        <input type="text" name="email" class="form-control"
placeholder="E-mail*">
                     </div>
                     <div class="form-group col-md-12">
                        <label class="contact-icon">PASSWORD</label>
                       <input type="password" name="password" id="myInput"
class="form-control" placeholder="Password">
                     </div>
                     <div class="form-group col-md-12">
                       <input type="checkbox" onclick="myFunction()">
                       <label>Show Password</label>
                     </div>
                  </div>
                  <div class="row">
                     <div class="form-group col-md-6">
                        <a href="" pull-right>Forgot Passwod? </a><i class="fa fa-lock"
style="font-size:24px"></i>
                     </div>
                     <div class="form-group col-md-6">
                        <a href="registration.php" pull-right>New Registration? </a><i
class="fa fa-sign-in" style="font-size:24px"></i>
                     </div>
                  </div>
                   <div class="form-group">
                     <input type="submit" name="login" class="btn btn-primary pull-
right" value="LOGIN NOW">
                  </div>
                  <div class="clearfix"></div>
               </form>
        </div>
      </div>
```

```
    </div>
   </div>
   </body>
</html>
```

.) Application (application.hp)

```
<?php
 include 'database.php';
 $data = new Databases;
 session_start();
 $email = $_SESSION["email"];
 $condition_arr = array('email'=>$email);
 $user = $data->getData('registration','*',$condition_arr);
 $reg_id = $user[0]['id'];
 $bank_arr = array('reg_id'=>$reg_id);
 $bank = $data->getData('bank','*',$bank_arr);
 if($bank == 0 )
  {
    echo '<script type="text/javascript">';
    echo 'alert("Update Your Account Details");';
    echo 'window.location.href = "bank.php";';
    echo '</script>';
  }
 if(isset($_POST["save"]))
 {
    $incomecertificate = $_FILES['incomecertificate']['name'];
    $ration = $_FILES['ration']['name'];
    $image = $_FILES['doc1']['name'];
    $doc2 = $_FILES['doc2']['name'];
    $doc3 = $_FILES['doc3']['name'];
    $selfdeclaration = $_FILES['selfdeclaration']['name'];
    $target1 = "document/".basename($incomecertificate);
    $target2 = "document/".basename($ration);
    $target3 = "images/".basename($image);
    $target4 = "document/".basename($doc2);
    $target5 = "document/".basename($doc3);
    $target6 = "document/".basename($selfdeclaration);
    $insert_data = array(
        'reg_id' => $reg_id,
        'cat_id' => mysqli_real_escape_string($data->con, $_POST['category']),
        'subcategory' => mysqli_real_escape_string($data->con,
$_POST['subcategory']),
```

```
        'amount' => mysqli_real_escape_string($data->con, $_POST['amount']),

        'bal_amt' => 0,
        'description' => mysqli_real_escape_string($data->con,
$_POST['description']),
        'datehappened' => mysqli_real_escape_string($data->con,
$_POST['datehappened']),

        'incomecertificate' => mysqli_real_escape_string($data->con,
$_FILES['incomecertificate']['name']),
        'ration' => mysqli_real_escape_string($data->con, $_FILES['ration']['name']),
        'image' => mysqli_real_escape_string($data->con, $_FILES['doc1']['name']),
        'doc2' => mysqli_real_escape_string($data->con, $_FILES['doc2']['name']),
        'doc3' => mysqli_real_escape_string($data->con, $_FILES['doc3']['name']),
        'selfdeclaration' => mysqli_real_escape_string($data->con,
$_FILES['selfdeclaration']['name']),
        'status' => 0
    );
    if($data->insert('application', $insert_data))
    {
        move_uploaded_file($_FILES['incomecertificate']['tmp_name'], $target1);
        move_uploaded_file($_FILES['ration']['tmp_name'], $target2);
        move_uploaded_file($_FILES['doc1']['tmp_name'], $target3);
        move_uploaded_file($_FILES['doc2']['tmp_name'], $target4);
        move_uploaded_file($_FILES['doc3']['tmp_name'], $target5);
        move_uploaded_file($_FILES['selfdeclaration']['tmp_name'], $target6);
        echo '<script>alert("Successfully Add Your Application")</script>';
    }
 }
?>
<!DOCTYPE html>
<html class="no-js">
   <head>
      <title>Registration</title>
   </head>
   <body>
        <form method="post" action="" enctype="multipart/form-data">
          <div class="row">
                    <div class="form-group col-md-6">
                    <label>First Name</label>
                    <input type="text" name="fname" class="form-control"
value="<?php echo($user[0]['fname']);?>" placeholder="First Name*"
maxlength="25"  autofocus disabled>
```

```
                    <span style="color:red;"></span>
                    </div>

                     <div class="form-group col-md-6">
                       <label>Last Name</label>
                       <input type="text" name="lname" class="form-control"
value="<?php echo($user[0]['lname']);?>" placeholder="Last Name*"
maxlength="25" disabled>
                       <span style="color:red;"></span>
                    </div>
          </div>
          <div class="row">
                    <div class="form-group col-md-6">
                       <label>Email Address</label>
                       <input type="text" name="email" class="form-control"
value="<?php echo($user[0]['email']);?>" required placeholder="E-mail*"
maxlength="30" disabled>
                       <span style="color:red;"></span>
                    </div>

                    <div class="form-group col-md-6">
                       <label>Phone Number</label>
                       <input type="text" name="phone" id="phone" value="<?php
echo($user[0]['phone']);?>" class="form-control" placeholder="Phone Number*"
maxlength="10" disabled>
                       <span style="color:red;"></span>
                    </div>
          </div>
          <div class="row">
                    <div class="form-group col-md-6">
                      <label>Aadhaar Number</label>
                       <input type="text" name="aadhaar" id="aadhaar" class="form-
control" value="<?php echo($user[0]['aadhaar']);?>" placeholder="Aadhaar
Number*" maxlength="14" disabled>
                       <span style="color:red;"></span>
                    </div>
                     <div class="form-group col-md-6">
                       <label>Pan card Number</label>
                       <input type="text" name="pancard" class="form-control"
value="<?php echo($user[0]['pancard']);?>" placeholder="Pan Card Number*"
maxlength="10" disabled>
                       <span style="color:red;"></span>
                    </div>
```

```
        </div>
        <div class="row">
                <div class="form-group col-md-6">
                 <?php
                 $category = $data->getData('emgcategory','*');
                 ?>
                  <label>Category</label>
                  <select class="form-control" name="category">
                   <option>---Select Category---</option>
                   <?php foreach ($category as $value)
                   {
                   ?>
                     <option value="<?php echo $value['id'];?>"><?php echo
$value['category'];?></option>
                   <?php
                   }
                   ?>
                  </select>
                  <span style="color:red;"></span>
                </div>
                <div class="form-group col-md-6">
                   <label>Sub Category</label>
                   <input type="text" class="form-control" name="subcategory">
                   <span style="color:red;"></span>
                </div>
        </div>
        <div class="row">
                <div class="form-group col-md-6">
                   <label>Expected Amount</label>
                   <input type="text" class="form-control" name="amount">
                   <span style="color:red;"></span>
                </div>
                <div class="form-group col-md-6">
                   <label>Which Day Its Happened</label>
                   <input type="date" name="datehappened" class="form-control"
max="<?php echo(date('Y-m-d'))?>">
                   <span style="color:red;"></span>
                </div>
        </div>
        <div class="row">
                <div class="form-group col-md-12">
                 <label>Description</label>
```

```
                    <textarea rows="5" class="form-control" name="description"
maxlength="500"></textarea>
                    <span style="color:red;"></span>
                  </div>
          </div>
          <div class="row">
                   <div class="form-group col-md-6">
                    <label>Image</label>
                    <input type="file" name="doc1" class="form-control">
                    <span style="color:red;"></span>
                  </div>
                  <div class="form-group col-md-6">
                    <label>Income Certificate</label>
                    <input type="file" name="incomecertificate" class="form-
control">
                    <span style="color:red;"></span>
                  </div>
                  <div class="form-group col-md-6">
                     <label>Ration Card</label>
                     <input type="file" name="ration" class="form-control">
                     <span style="color:red;"></span>
                  </div>
                   <div class="form-group col-md-6">
                     <label>Supporting Document/Certificate 1</label>
                    <input type="file" name="doc2" class="form-control">
                     <span style="color:red;"></span>
                  </div>
                  <div class="form-group col-md-6">
                    <label>Supporting Document/Certificate 2</label>
                    <input type="file" name="doc3" class="form-control">
                    <span style="color:red;"></span>
                  </div>

                   <div class="form-group col-md-6">
                     <label>Self Declaration</label>
                    <input type="file" name="selfdeclaration" class="form-control">
                     <span style="color:red;"></span>
                  </div>
          </div>
          <div class="row">
            <br>
                  <div class="form-group col-md-8">
                  </div>
```

```
                    <div class="form-group col-md-2">
                      <input type="submit" name="preview" value="Preview"
class="btn btn-primary pull-right" />
                    </div>
                     <div class="form-group col-md-2">
                       <input type="submit" name="save" id="save" value="Final
Submit" class="btn btn-primary pull-right" />
                    </div>
          </div>
        </form>
   </body>
</html>
```

.) Donor (Donor.php)

```
<?php
  include 'database.php';
  $data = new Databases;
  session_start();
  include 'donorlogin.php';
  $email = $_SESSION["email"];
  $condition_arr = array('email'=>$email);
  $user = $data->getData('registration','*',$condition_arr);
  $reg_id = $user[0]['id'];
  $status_arr = array('status'=>1);
  $result = $data->getData('application','*',$status_arr);
?>
<!DOCTYPE html>
<html class="no-js">
   <head>
   </head>
   <body>
   <?php
     include('include/donor-header.php');
   ?>
         <div class="main-container">
                  <div>
               <div class="container">
                  <h2 class="title-style-1">NEED YOUR HELP <span class="title-
under"></span></h2>
                  <div class="row">
                  <?php
            if(isset($result['0']))
```

```
            {
              foreach ($result as $value)
              {
              ?>
                            <div class="col-md-3 col-sm-6">
                               <div class="cause text-center">
                                  <img src="images/<?php echo($value['image']);?>"
alt="" class="cause-img">
                        <?php
                          $percentage = ($value['bal_amt'] / $value['amount']) * 100 ;
                        ?>
                        &#8377; <?php echo($value['bal_amt']);?> / &#8377; <?php
echo($value['amount']);?>
                                  <div class="progress cause-progress">
                                    <div class="progress-bar" role="progressbar" aria-
valuenow="30" aria-valuemin="0" aria-valuemax="100" style="width:<?php
echo($percentage);?>%">
                                    </div>
                                  </div>

                                  <h4 class="cause-title"><a href="#"><?php
echo($value['subcategory']);?></a></h4>
                                  <div class="cause-details" style="text-align:center;">
                                     <?php echo($value['description']);?>
                                  </div>
                                  <div class="btn-holder text-center">
                                    <a href="payment.php?id=<?php echo $value['id']?>"
class="btn btn-primary"> DONATE NOW</a>
                                  </div>
                                </div>
                              </div>
               <?php
               }
              }
              else
              {?>
                 <div class="btn-holder text-center">
                   <label>No Application Found</label>
                 </div>
              <?php
              }
              ?>
                  </div>
```

```
                </div>
            </div>
        </div>

    <?php
      include('include/donor-footer.php');
    ?>
    </body>
</html>
```

.) Payment (payment.php)

```
<?php
  include 'database.php';
  $data = new Databases;
  session_start();
  $email = $_SESSION["email"];
  $condition_arr = array('email'=>$email);
  $user = $data->getData('registration','*',$condition_arr);
  $donor_id = $user[0]['id'];
  $result = $data->getData('application','*');
  $id = $_GET['id'];
  $app_array = array('id'=>$id);
  echo $id;
  $app_id = $data->getData('application','*',$app_array);
  $application_id = $app_id[0]['id'];
  $amount = $app_id[0]['bal_amt'];
if(isset($_POST["donate"]))
{
    $bill = $data->getData('payment','*','','id','desc','','1');
    $last = $bill[0]['bill_no'];
    if($last == '')
    {
        $bill_no = "DRF101";
    }
    else
    {
        $bill_no = substr($last, 5);
        $bill_no = intval($bill_no);
        $bill_no = "DRF10" . ($bill_no + 1);
    }
     $insert_data = array(
            'bill_no' => $bill_no,
```

```
        'application_id' => $application_id,
        'donor_id' => $donor_id,
        'name' => mysqli_real_escape_string($data->con, $_POST['card_name']),
        'amount' => mysqli_real_escape_string($data->con, $_POST['amount']),
    );
     $bal_amt = $amount + $_POST['amount'];
     if($bal_amt > $app_id[0]['amount'])
     {
        $update_arr = array(
                    'bal_amt' => $bal_amt,
                    'status' => 3);
        $data->updateData('application',$update_arr,$application_id);

     }
     else
     {
        $update_arr = array('bal_amt'=>$bal_amt);
        $data->updateData('application',$update_arr,$application_id);
     }
    if($data->insert('payment', $insert_data))
    {
       echo '<script type="text/javascript">';
       echo 'alert("Payment Successfully");';
       echo 'window.location.href = "certificate.php";';
       echo '</script>';
    }
}
?>
<!DOCTYPE html>
<html class="no-js">
   <head>
      <meta charset="utf-8">
      <title>Contact | Charity / Non-profit responsive Bootstrap HTML5 template</title>
   </head>
   <body>

   <?php
      include('header.php');
   ?>
   <div class="main-container fadeIn animated">

      <div class="container">
```

```
        <div class="row">

            <div class="col-md-7 col-sm-12 col-form">

                <h2 class="title-style-2">DONATE ONLINE<span class="title-
under"></span></h2>

                <form method="POST">

                    <div class="row">

                        <div class="form-group col-md-12">
                            <label class="contact-icon">CARD HOLDER NAME</label>
                            <input type="text" name="card_name" class="form-control"
placeholder="Name*" required>
                        </div>

                    </div>
                    <div class="row">

                        <div class="form-group col-md-6">
                            <label class="contact-icon">CARD NUMBER</label>
                            <input type="text" name="card_number" id="cr_no"
class="form-control" placeholder="0000 0000 0000 0000*" maxlength="19"
minlength="19" required>
                        </div>

                        <div class="form-group col-md-6">
                            <label class="contact-icon">AMOUNT</label>
                            <input type="text" name="amount" class="form-control"
placeholder="Amount*" required>
                        </div>
                    </div>
                    <label>EXP MONTH & YEAR</label>
                    <div class="row">
                        <div class="form-group col-md-4">
                            <select class="form-control" id="year" name="expmonth">
                                <option>Jan</option>
                                <option>Feb</option>
                                <option>Mar</option>
                                <option>Apr</option>
                                <option>May</option>
```

```
                        <option>Jun</option>
                        <option>Jul</option>
                        <option>Aug</option>
                        <option>Sep</option>
                        <option>Oct</option>
                        <option>Nov</option>
                        <option>Dec</option>
                      </select>
                    </div>
                     <div class="form-group col-md-4">
                      <!-- <input type="email" name="email" class="form-control" placeholder="E-mail*" required> -->
                      <select class="form-control" id="year" name="expyear">
                        <option selected>Year</option>
                        <option>2021</option>
                        <option>2022</option>
                        <option>2023</option>
                        <option>2024</option>
                        <option>2025</option>
                        <option>2026</option>
                        <option>2027</option>
                      </select>
                    </div>
                    <div class="form-group col-md-4">
                      <input type="password" name="ccv" class="form-control" placeholder="CVV*" maxlength="3" minlength="3" required>
                    </div>
                  </div>
                   <div class="form-group">
                    <!-- <button type="submit" class="btn btn-primary pull-right">DOANTE NOW</button> -->
                    <input type="submit" name="donate" class="btn btn-primary pull-right" value="DOANTE NOW">
                  </div>

                  <div class="clearfix"></div>
                </form>
              </div>
            <div class="col-md-4 col-md-offset-1 col-contact">
                <h2 class="title-style-2"> Accepted Cards <span class="title-under"></span></h2>
                <div class="contact-items">
                  <i class="fa fa-cc-visa" style="color:navy;font-size:250%;"></i>
```

```
                    <i class="fa fa-cc-amex" style="color:blue;font-size:250%;"></i>
                    <i class="fa fa-cc-mastercard" style="color:red;font-size:250%;"></i>
                    <i class="fa fa-cc-discover" style="color:orange;font-size:250%;"></i>
                </div>
            </div>
        </div> <!-- /.row -->
      </div>
   </div>
   </div>
 <?php
      include('footer.php');
   ?>
   </body>
</html>
```

.) Certificate(certificate.php)

```
<?php
 include 'database.php';
 $data = new Databases;
 session_start();
 $email = $_SESSION["email"];
 $condition_arr = array('email'=>$email);
 $user = $data->getData('registration','*',$condition_arr);
 $id = $user[0]['id'];
 $pay_arr = array('donor_id'=>$id);
 $result = $data->getData('payment','*',$pay_arr,'bill_no ');
?>
<!DOCTYPE html>
<html class="no-js">
   <head>
      <meta charset="utf-8">
      <title>Single cause | Charity / Non-profit responsive Bootstrap HTML5 template</title>
   </head>
   <body>
   <?php
     include('include/donor-header.php');
   ?>
         <div class="page-heading text-center">

                  <div class="container zoomIn animated">
```

```
                    <h1 class="page-title">CAUSE TITLE <span class="title-
under"></span></h1>
                    <p class="page-description">
                        Lorem ipsum dolor sit amet, consectetur adipisicing elit
Necessitatibus.
                    </p>
                </div>
            </div>
            <div class="main-container">
                <div class="container">
                    <div class="row ">
                        <div class="col-md-12 fadeIn animate-onscroll">

                            <h2 class="title-style-2">APPLICATION
STATUS <span class="title-under"></span></h2>
                                <table class="table table-style-1">
                                <thead>
                                  <tr>
                    <th>SNO</th>
                    <th>Bill Number</th>
                    <th>Name</th>
                    <th>Date</th>
                    <th>Amount</th>
                <th>Download</th>
                                  </tr>
                                </thead>
                                <tbody>
                    <?php
                    if(isset($result['0']))
                    {
                    $sn = 0;
                       foreach ($result as $value)
                       {
                         $sn +=1;
                       ?>
                                  <tr>
                                  <td><?php echo $sn ?></td>
                    <td><?php echo($value['bill_no']);?></td>
                    <td><?php echo($value['name']);?></td>
                                  <td><?php echo($value['date']);?></td>
                                  <td><?php echo($value['amount']);?></td>
```

```
                                        <td><a
href="download_certificate.php?id=<?php echo($value['id']);?>" class="btn btn-
success">Print</a></td>
                                        </tr>
                                        <?php
                    }
                }
                else
                {
                    echo "<td colspan='6' style='text-align:center'>No Result
Found</td>";
                }
                    ?>
                                    </tbody>
                                </table>
                            </div>
                        </div>
                </div>
            </div> <!-- /.main-container  -->
    <?php
      include('footer.php');
    ?>
    </body>
</html>
```

.) Add and Emergency Funds (emgcategory.php)

```
<?php
  include '../../../database.php';
  $data = new Databases;
   if(isset($_POST["submit"]))
   {
       $insert_data = array(
          'category' => mysqli_real_escape_string($data->con, $_POST['category'])
       );
       if($data->insert('emgcategory', $insert_data))
       {
           echo '<script>alert("Successful Add a New Category ")</script>';
       }
   }
?>
<!DOCTYPE html>
<html lang="en">
<head>
```

```
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>AdminLTE 3 | General Form Elements</title>
</head>
<body class="hold-transition sidebar-mini">
<div class="wrapper">
  <!-- Navbar -->
  <?php
    include('include/navbar.php');
  ?>
<?php
  include('include/sidebar.php');
?>
  <div class="content-wrapper">
    <section class="content-header">
      <div class="container-fluid">
        <div class="row mb-2">
          <div class="col-sm-6">
            <h1>Emergency Category Form</h1>
          </div>
          <div class="col-sm-6">
            <ol class="breadcrumb float-sm-right">
              <li class="breadcrumb-item"><a href="#">Home</a></li>
              <li class="breadcrumb-item active">Category Form</li>
            </ol>
          </div>
        </div>
      </div>
    </section>
    <section class="content">
      <div class="container-fluid">
        <div class="row">
          <div class="col-md-12">
            <div class="card card-primary">
              <div class="card-header">
                <h3 class="card-title">Category Form</h3>
              </div>
              <form method="post" action="emgcategory.php">
                <div class="card-body">
                  <div class="col-6">
                    <label for="exampleInputEmail1">Category Name</label>
                    <input type="text" class="form-control" id="exampleInputEmail1"
placeholder="Category Name" name="category" required>
```

```
            </div>

          </div>
          <div class="card-footer">
            <input type="submit" name="submit" class="btn btn-primary"
value="Submit">
          </div>
        </form>
      </div>

    </div>
  </div>
</section>
<section class="content">
  <div class="container-fluid">
    <div class="row">
      <div class="col-12">
        <div class="card">
          <div class="card-header">
            <h3 class="card-title">DataTable with minimal features & hover
style</h3>
          </div>
        </div>
        <div class="card">
          <div class="card-header">
            <h3 class="card-title">DataTable with default features</h3>
          </div>
          <div class="card-body">
            <table id="example1" class="table table-bordered table-striped">
              <thead>
              <tr>
                <th>SNO</th>
                <th>Category Name</th>
                <th>Edit</th>
                <th>Delete</th>
              </tr>
              </thead>
              <tbody>
                <?php
                  $result = $data->getData('emgcategory','*');
                  $sn=1;
                  if(isset($result['0']))
                  {
```

```
                    foreach ($result as $value)
                    {
                      echo "<tr><td>" . $sn . "</td>";
                      echo "<td>" . $value['category'] . "</td>";
                      ?>
                      <td><a href="editemgcategory.php?id=<?php echo
$value['id'];?>"><i style="font-size:24px" class="fa">&#xf044;</i></a></td>
                      <td><a href="delete.php?id=<?php echo
$value['id']?>&table=emgcategory" id="a_id"><i style='font-size:24px'
class='fa'>&#xf2ed;</i></a></td></tr>
                    <?php
                    $sn++;
                    }
                  }
                  else
                  {
                    ?>
                    <td colspan="5" style="text-align: center;"><b>No Result
Found</b></td>
                    <?php
                  }
                ?>
                </tbody>
              </table>
</body>
</html>
```

.) Application Approval (application.php)

```
<?php
  include '../../../database.php';
  $data = new Databases;
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>AdminLTE 3 | General Form Elements</title>
</head>
<body class="hold-transition sidebar-mini">
    <section class="content">
      <div class="container-fluid">
        <div class="row">
```

```
        <div class="col-12">
          <div class="card">
            <div class="card-header">
              <h3 class="card-title">DataTable with minimal features & hover
style</h3>
            </div>
          </div>
          <div class="card">
            <div class="card-header">
              <h3 class="card-title">DataTable with default features</h3>
            </div>
            <!-- /.card-header -->
            <div class="card-body">
              <table id="example1" class="table table-bordered table-striped">
                <thead>
                <tr>
                  <th>SNO</th>
                  <th>Applicant Name</th>
                  <th>Category Name</th>
                  <th>Amount</th>
                  <th>Date</th>
                  <th>Image</th>
                  <th>View</th>
                  <th>Action</th>
                  <th>Status</th>
                </tr>
                </thead>
                <tbody>
                  <?php
                    $result = $data->getData('application','*');
                    $sn=1;
                    if(isset($result['0']))
                    {
                      foreach ($result as $value)
                      {
                        $app_id = $value['reg_id'];
                        $app_arr = array('id'=>$app_id);
                        $applicant =  $data->getData('registration','*',$app_arr);

                        echo "<tr><td>" . $sn . "</td>";
                        echo "<td>" . $applicant[0]['fname'] . " " . $applicant[0]['lname'] .
"</td>";
                        echo "<td>" . $value['subcategory'] . "</td>";
```

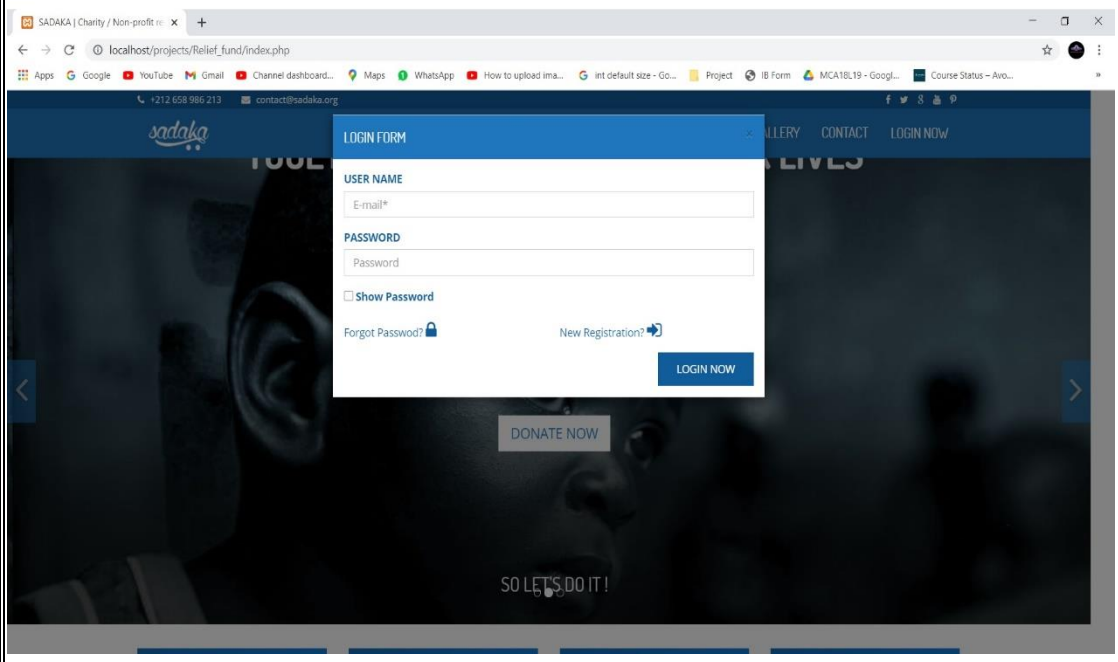
```
                    echo "<td>" . $value['amount'] . "</td>";
                    echo "<td>" . $value['datehappened'] . "</td>";
                    echo "<td><img width='100' height='100'
src='../../../images/".$value['image']."' ></td>";
                    ?>
                    <td><a href="viewapplication.php?id=<?php echo
$value['id'];?>&table=application"><i class="fa fa-eye" style="font-
size:150%"></i></a></td>
                     <td>
                      <?php
                      if($value['status']!=3)
                      {
                      ?>
                        <a href="confirm.php?id=<?php echo
$value['id']?>&table=application" id="a_id1"><span class="badge badge-success"
style="width:40%;">Confirm</span></a>
                        <a href="reject.php?id=<?php echo
$value['id']?>&table=application" id="a_id"><span class="badge badge-danger"
style="width:40%;">Reject</span></a>
                      <?php
                      }
                      else
                      {?>
                        <a href="" style="width:100%;" class="badge badge-
info">Completed</span></a>
                        <?php
                      }
                     ?>
                     </td>
                    <?php
                    if($value['status']==0)
                    {
                    ?>
                      <td><span class="badge badge-warning"
style="width:100%;">Pending</span></td></tr>
                    <?php
                    }
                    else if($value['status']==1)
                    {
                    ?>
                      <td><span class="badge badge-success"
style="width:100%;">Confirmed</span></td>
                    <?php
```

```
                    }
                    else if($value['status']==2)
                    {
                    ?>
                      <td><span class="badge badge-danger"
style="width:100%;">Rejected</span></td>
                    <?php
                    }
                    else
                    {
                    ?>
                      <td><span class="badge badge-info"
style="width:100%;">Completed</span></td>
                    <?php
                    }
                    ?>
                    </tr>
                  <?php
                  $sn++;
                  }
                }
                else
                {
                  ?>
                  <td colspan="5" style="text-align: center;"><b>No Result
Found</b></td>
                  <?php
                }
              ?>
              </tbody>
            </table>
          </div>
        </div>
      </div>
    </div>
  </section>
 </div>
</body>
```

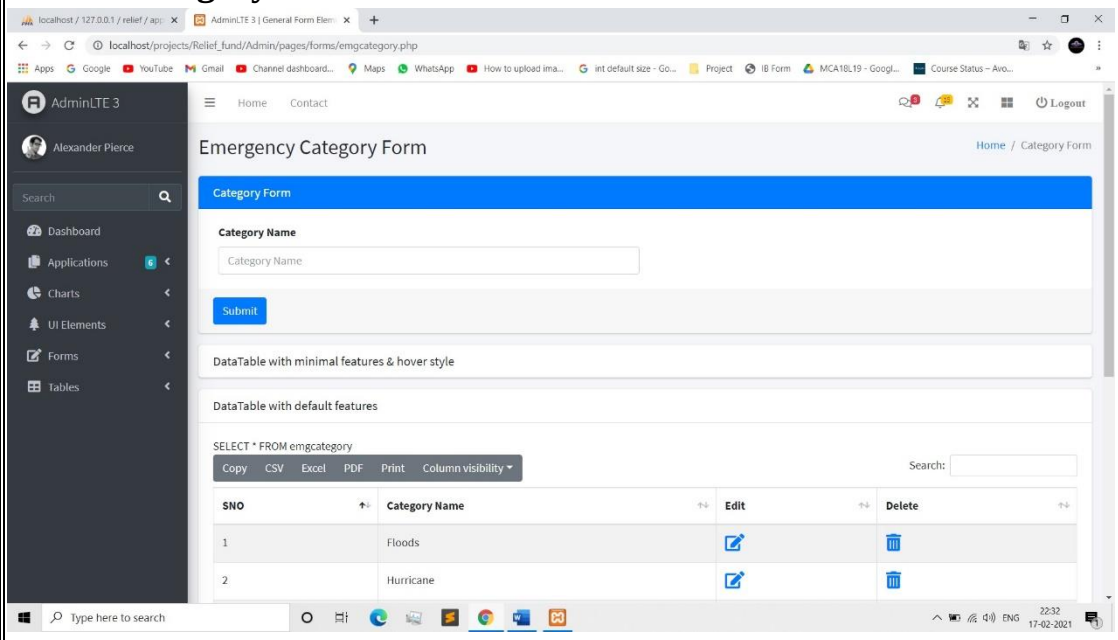
12.2 SCREEN SHOTS

Admin

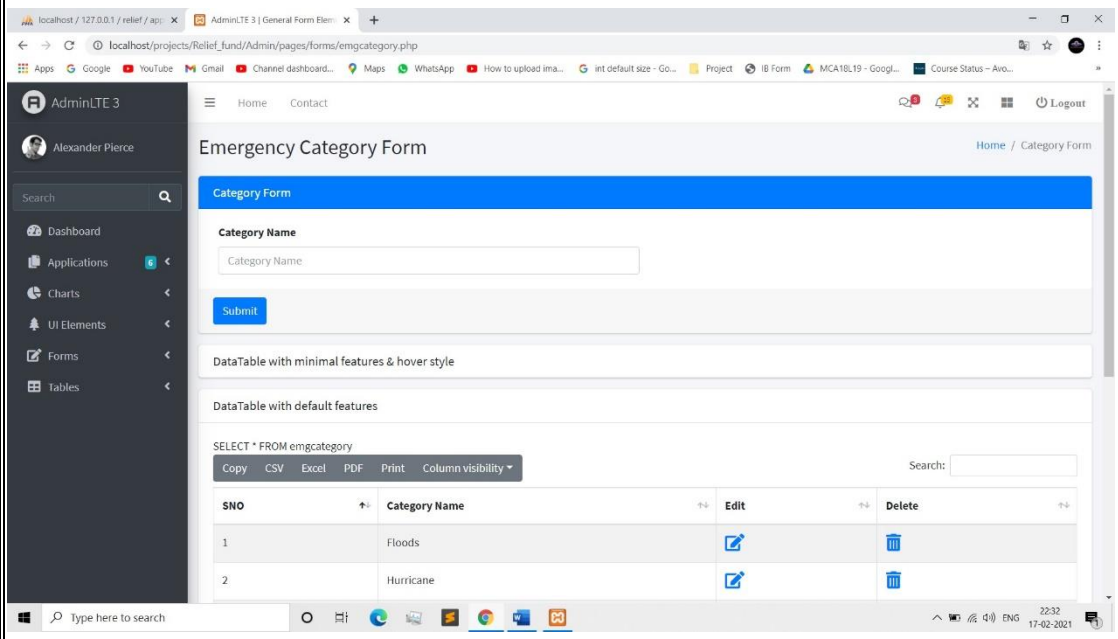
1.Login



2. Add Category

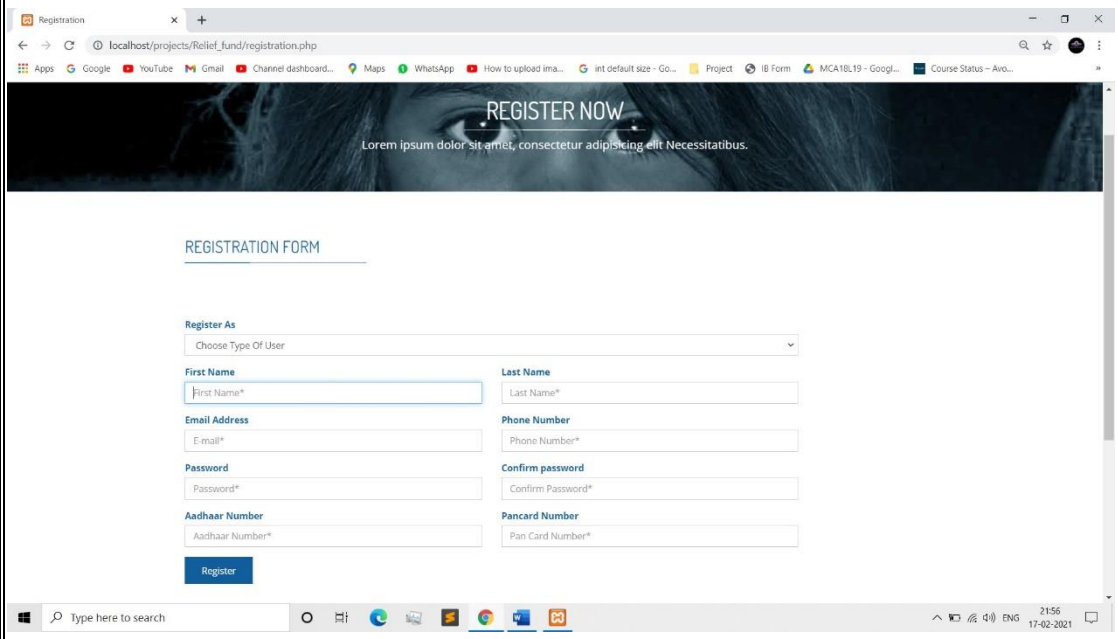


3.Verify Application

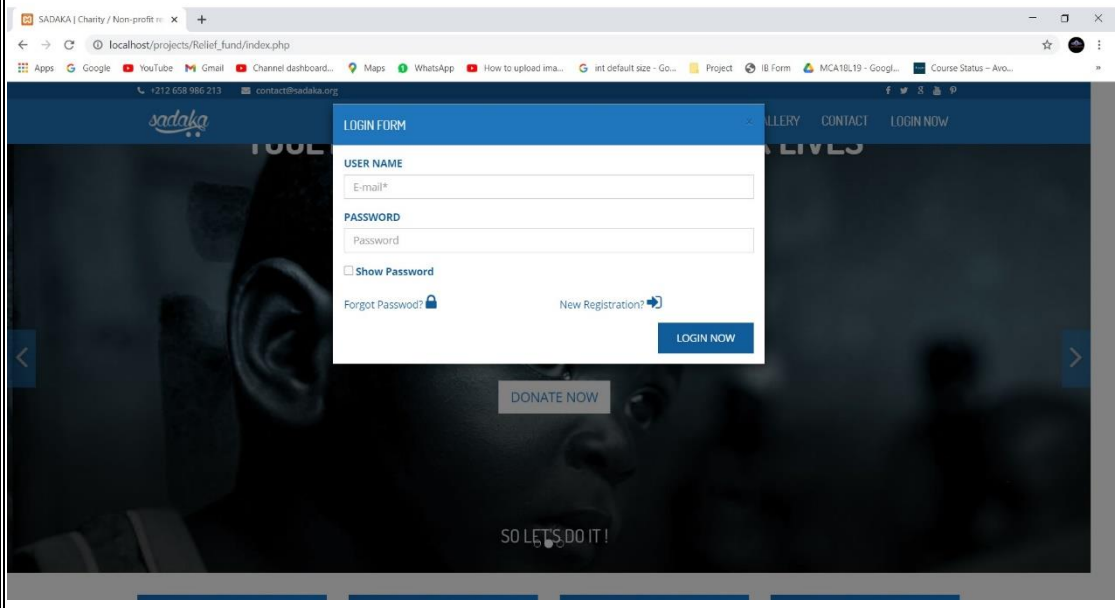


Applicant

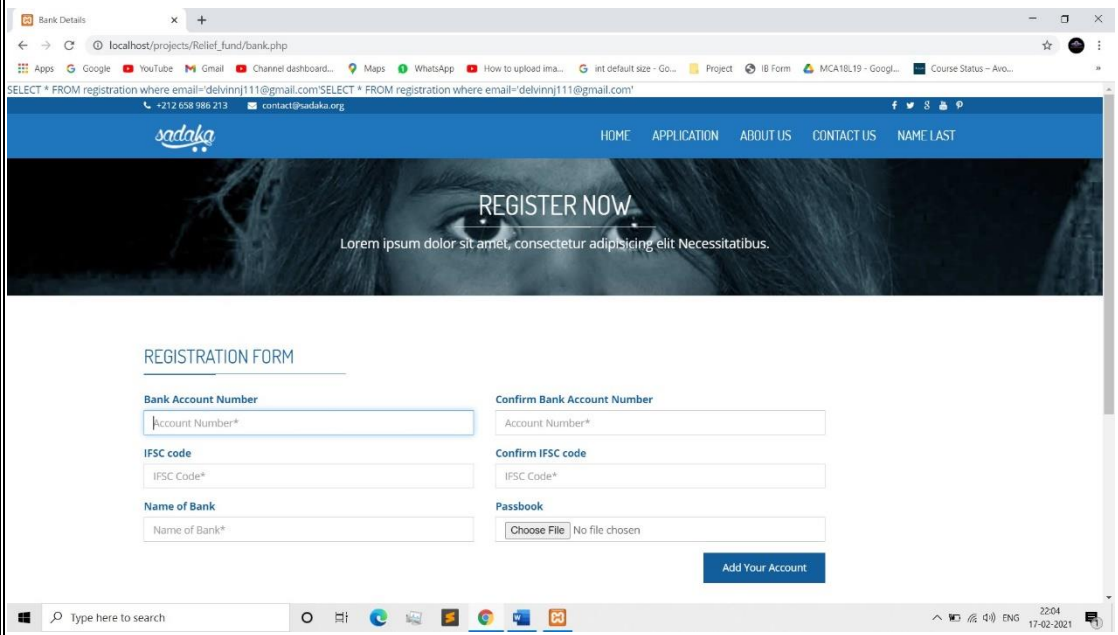
1.Registration



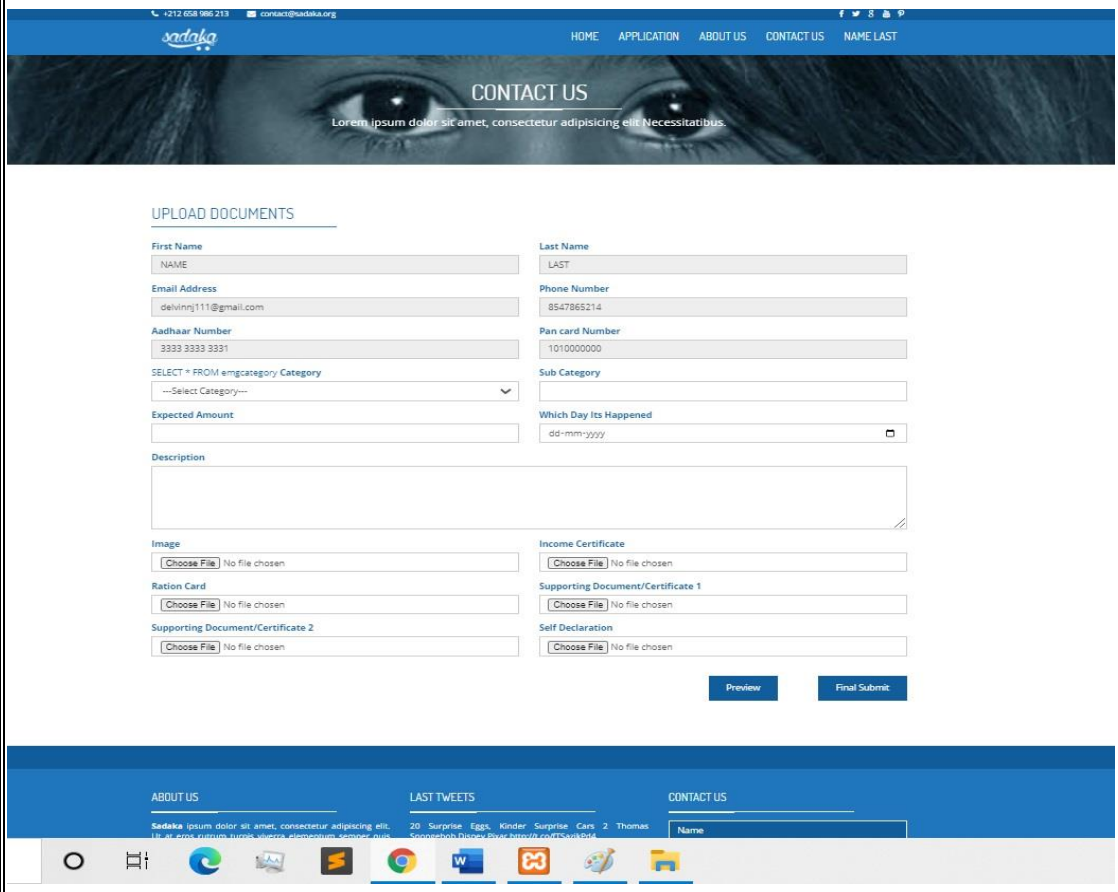
2.Login



4. Bank Details

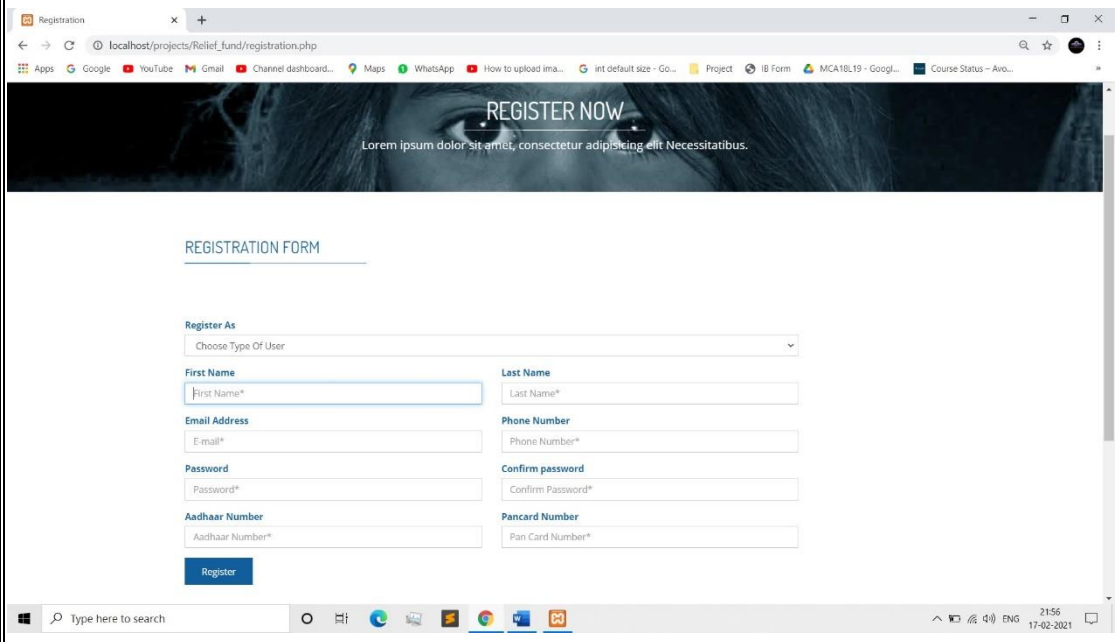


5. Add Application

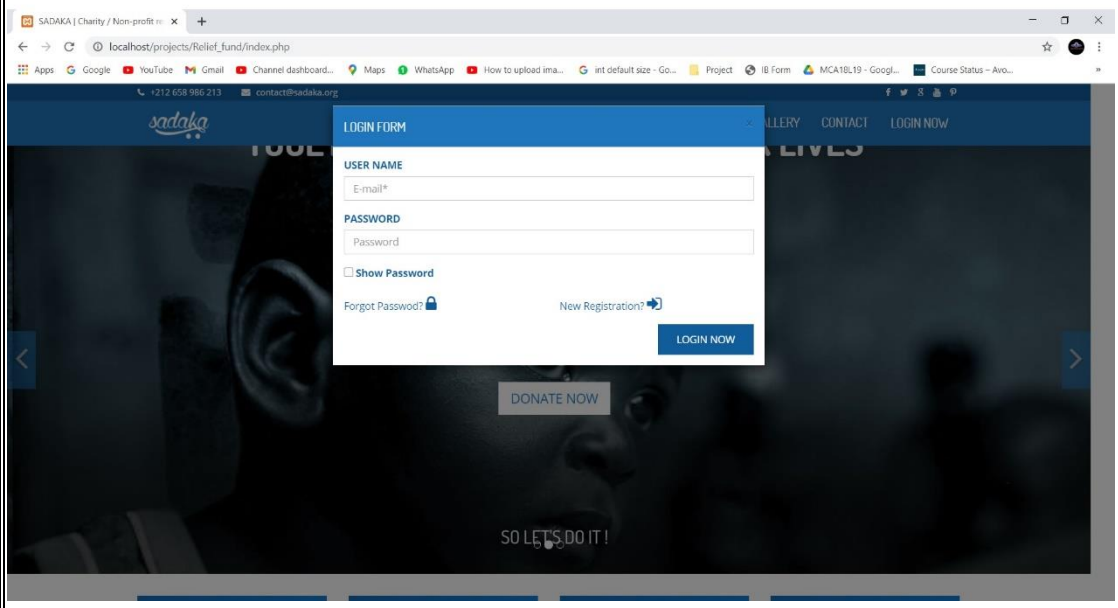


Donor

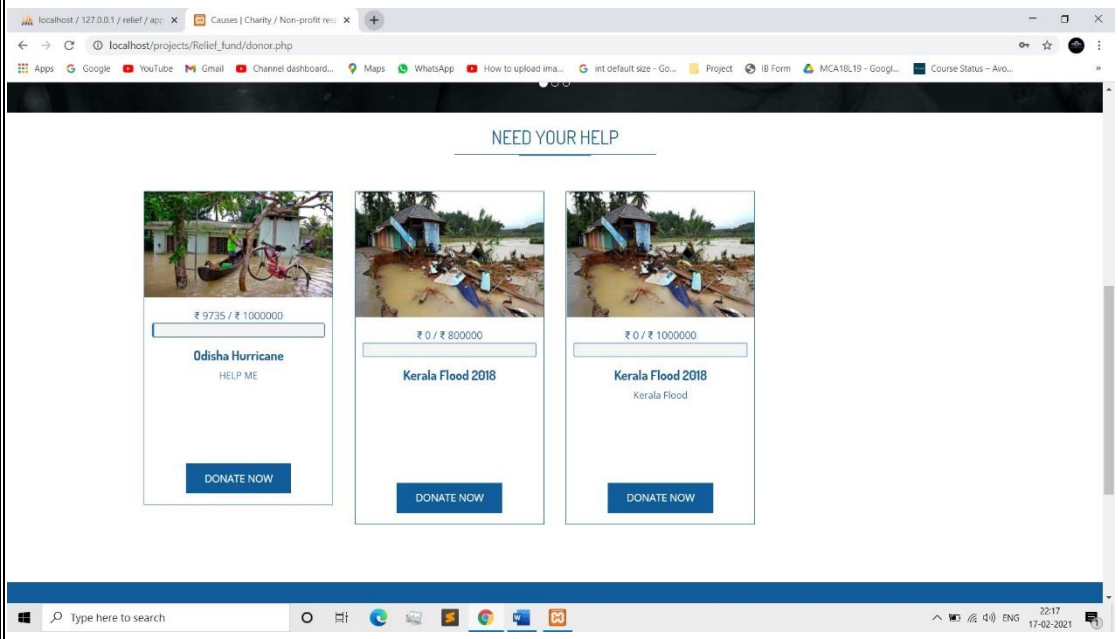
1. Donor Registration



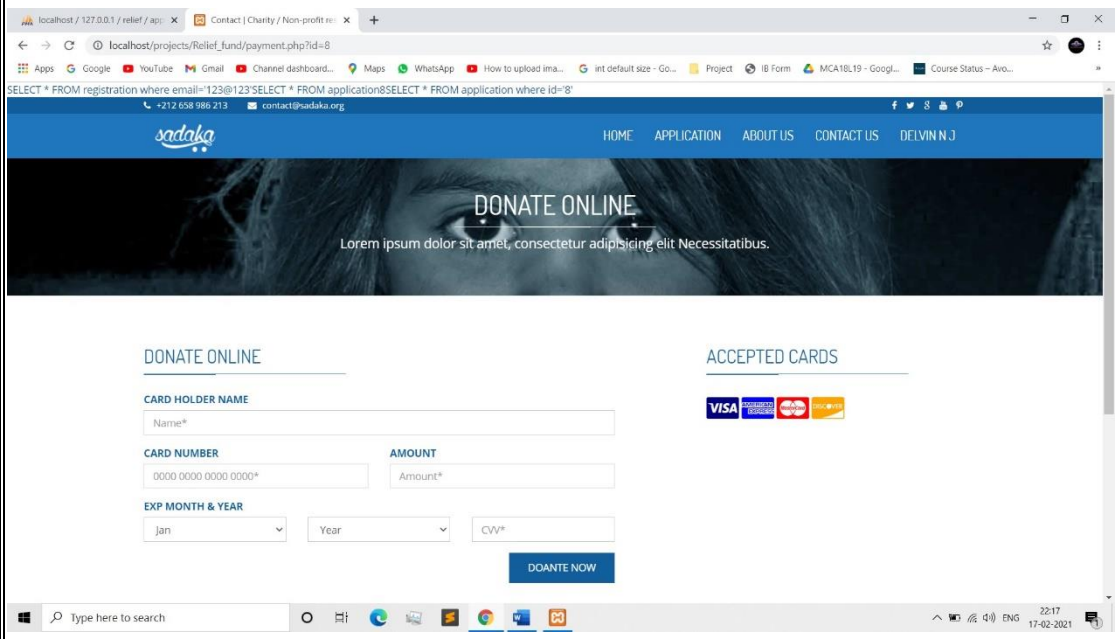
2. Login



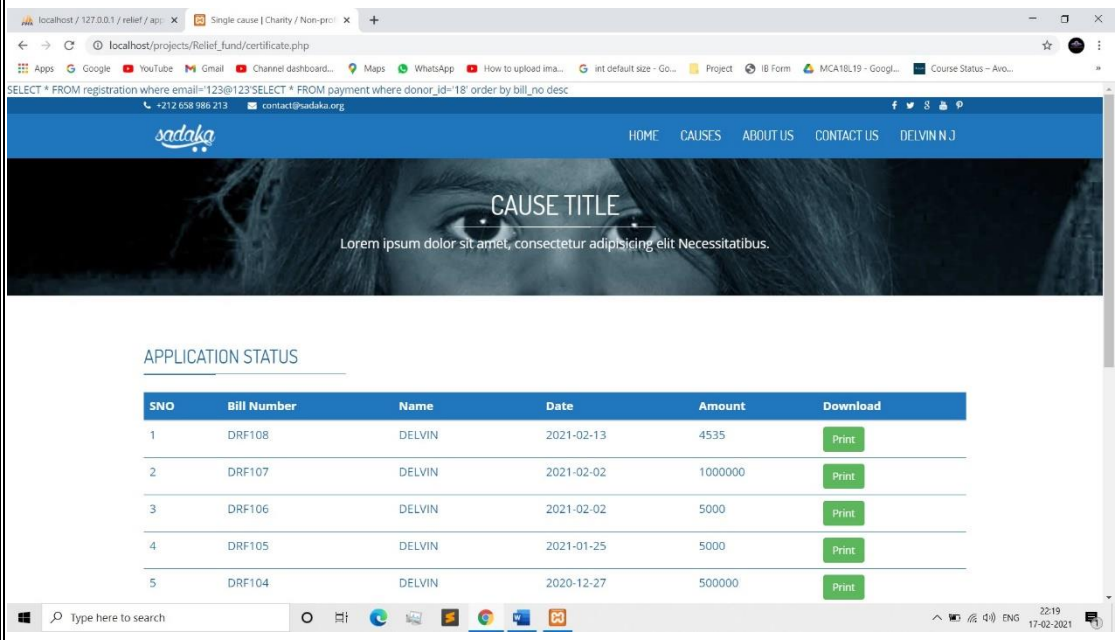
3. View



4. Payment



5.Report



ACRONYMS

12.4 ACRONYMS

- PHP:Hypertext Preprocessor
- HTML: Hypertext Markup Language
- SQL: Structured Query Language
- Wamp64, Apache, MySQL and PHP
- CGI: Common Gateway Interface
- CLI: Command Line Interface
- RDBMS: Relational Database Management System

BIBLIOGRAPHY

13.BIBLIOGRAPHY

WEB SITES

- https://www.w3schools.com/
- https://stackoverflow.com/
- https://plugins.jquery.com/

BOOKS

- Php Reference: Beginner to intermediate PHP5 –Mario Larig
- Php and MySQL development- Luke Welling
- System analysis and design,Award .E.M , First edition
- Introduction to system analysis and design, Lee, Second edition
- Software Engineering – A Practitioner's Approach by: Roger S. Pressman
- Software Engineering by: Ian Somerville