# 3. Design:

In Software development lifecycle design is the subsequent phase where the system is developed based on the user requirements. It is divided into three parts which include structural, behavioral and database design. here I have made diagrams like activity, flowchart, class diagrams etc. using tools like star uml to design the project according to users requirements by taking references from analysis phase.

## 3.1 Structural:

Structural design is the pattern that easiness association between entities and also displays the constant view of the system.

### 3.1.1 Class Diagram

The constant diagram which defines the system classes, attributes, methods and then relationships among the objects. I have designed class diagram for the Dhurmus Suntali foundation involving methods, classes and attributes.

* The admin class perform CRUD operation over further objects.
* The blog class have operations to display the daily journal on the website, read the comments and then add, edit and delete them.
* The gallery class have another operations to display the images of the foundation and can add and delete it.
* The videos class is also same like gallery which display the videos related to foundations which is responsible for adding and deleting the videos.
* Feedback class shows the feedback sent by the user and through which user will be contacted by the admin.

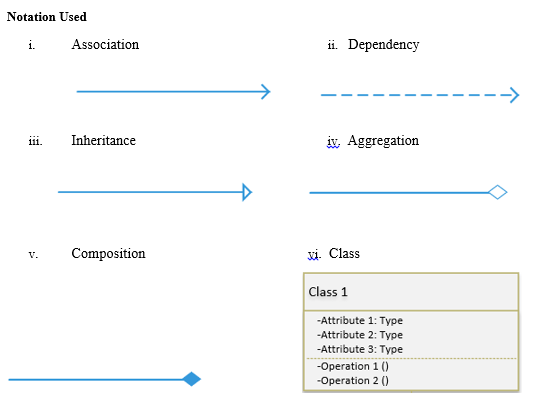


Fig: notation used in while designing class diagram.

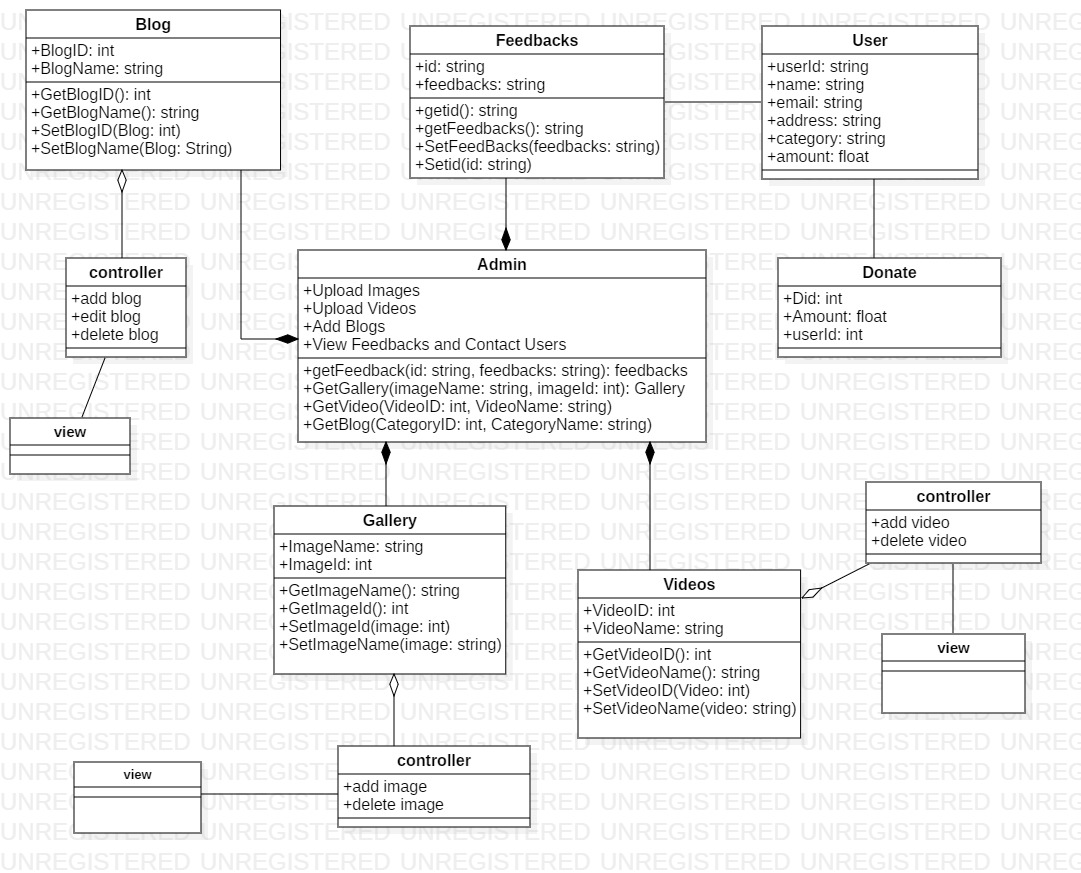


Fig: class diagram of Dhurmus Suntali foundation.

Justification

* The above class diagram is designed in star uml software in MVC (Model, View, and Controller) pattern which is clearly shown in the diagram and also shows that how these pattern are related with each other.
* To show relationship between classes I have used aggregation, composition and association. The composition is used because the composed object like controller cannot exist without another object like gallery. Likewise aggregation is used because without gallery admin can exists to so that composition is used in every main classes.
* And from the above shown class diagram admin has the ability to perform all the operation so all classes are connected with it.

## 3.1.2. Flowchart

Flowchart is the diagram that solve the task by step by step method. It is also used to clarify the logic behind the program before starting the code. I have choose this diagram rather than DFD (Data Flow Diagram) to spell out the logic of my system because it is easy to design and easy to understand make other understand.

## 

Fig: Flowchart for admin.

Justification

* The above flowchart diagram is for admin to show how admin can perform overall all operations after login in the system.
* To show the flow of diagram I have used flow line, decision box for if else statement and then shown process in rectangle box.
* After login in the dashboard opens and then he can add, edit and delete the task like image, videos and so on. if the task is successfully completed then the admin logout and check out in home page otherwise admin repeat that operation again and again.

## 2.2 Flowchart for user.

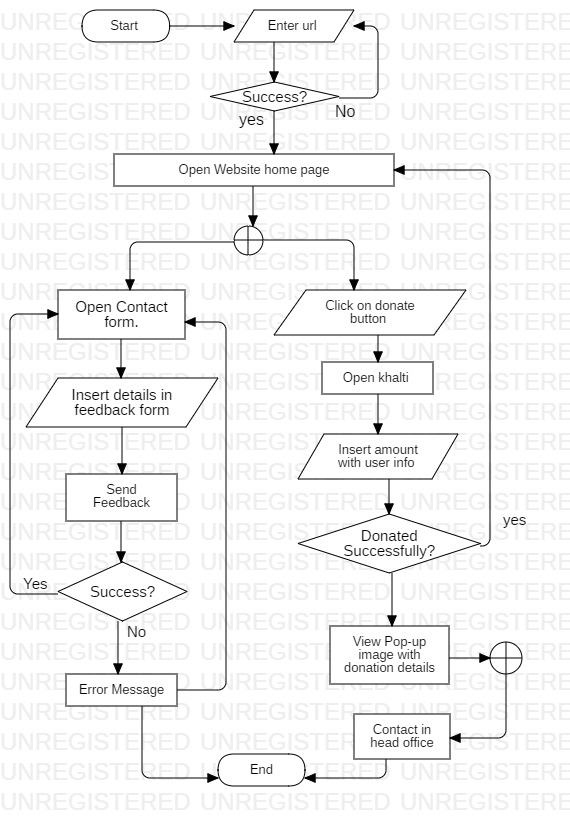


Fig: Flowchart for user.

Justification

* The above flow chart diagram shows that flow of system after accessing the website. First user either open the donation form to donate or send feedbacks to the admin about the website so circle shape with cross sign is used to show as ‘or’ option.
* If user donate the amount and if it is donated successfully then he return homepage after seeing successful message.
* Otherwise he return to the view page to see the pop up image to see donation details and then donate according to that details or the donators contact in the head office. So here again that circle shape with cross sign is used to show as ‘or’ option.

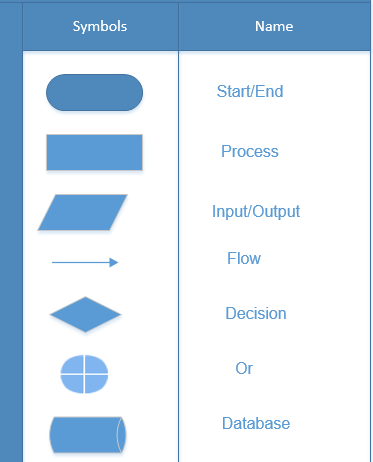


Fig: notation that are used while designing flow chart.

## 3.2 Behavioral:

The behavioral design pattern shows how the system interact with each other with one object to another object which signifies the relations among the structural diagrams and also show the dynamic environment of the system. It consists of various diagrams like activity, use case and sequence diagrams.

## 3.2.1 Activity Diagram

Activity diagram shows the movement from activity to activity inside the system which also address the dynamic view of the system.

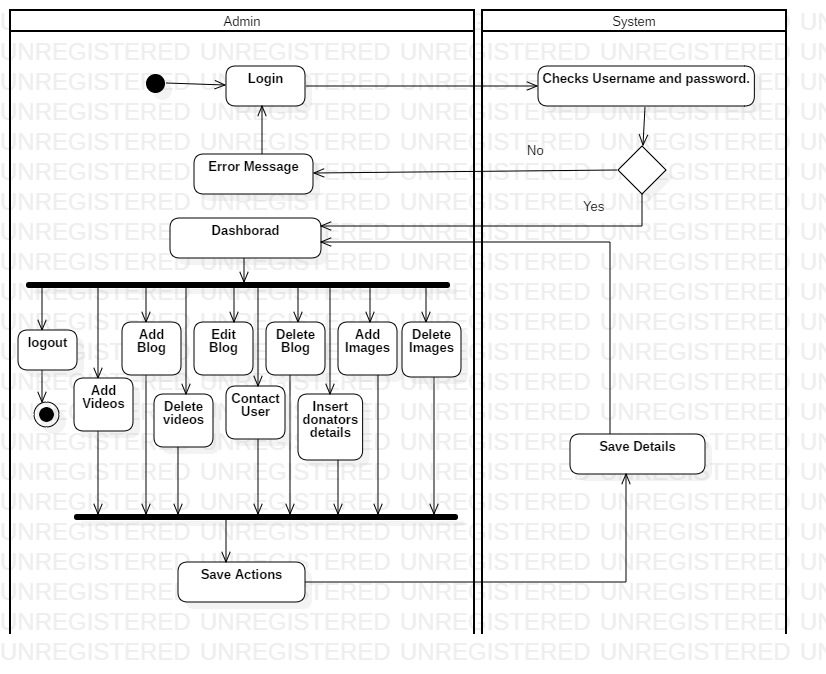


Fig: Activity diagram of admin of Dhurmus Suntali foundation.

Justification

* The above activity diagram is for admin’s activity and also display the whole activities involved during the duration of the system.
* After login in the dashboard opens and then he can add, edit and delete the task like image, videos and so on. If the task is successfully completed then the performed action is saved and also save the saved actions details in the system and then he return back to dashboard and logout. Hence overall task ends according to the above shown activity diagram.

## 3.2.2 Activity Diagram

# 

Fig: User activity diagram of Dhurmus Suntali Foundation.

Justification

* The above activity diagram is for user’s activity and also display the whole activities involved during the duration of the system.
* The above activity diagram illustrates that when user access home page then the user do various operation like send feedbacks, donate etc. but when the user donate and if it is successfully donated then the details is saves and then return back to the page otherwise they see the details to donate from the home page and donate from the bank or contact in the head office.

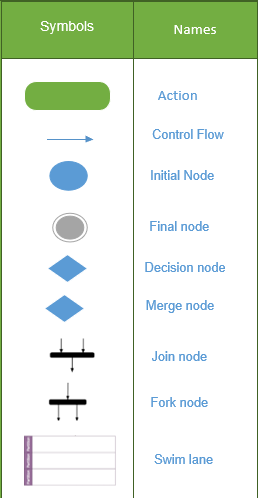


Fig: Notation used in activity diagram

## 3.3.1 Sequence diagram

Sequence diagram are the diagram that display interaction in the time order and show how the operation are carried out.

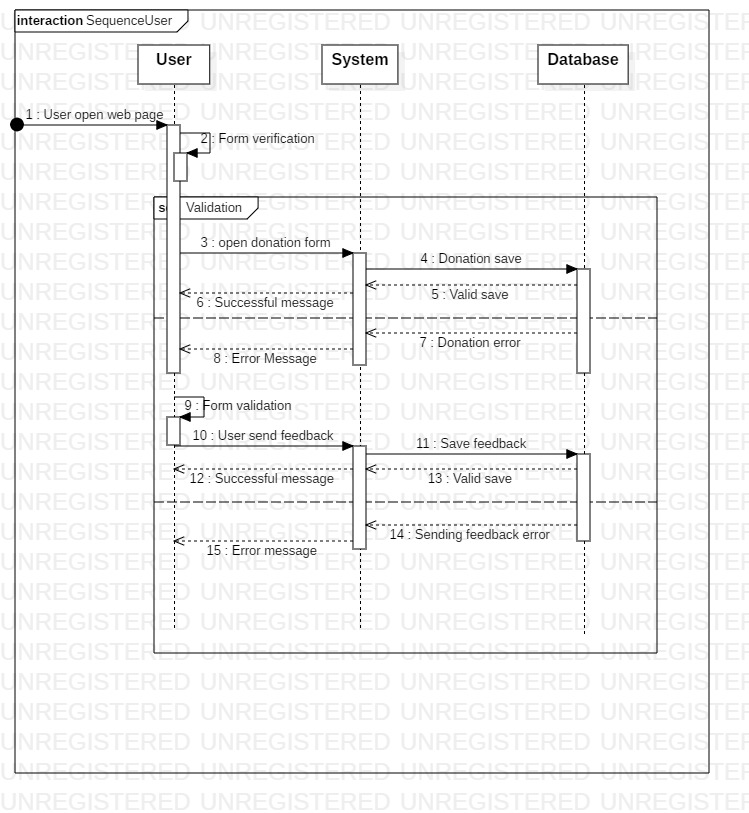


Fig: sequence diagram of user.

Justification

* The above sequence diagram illustrates that when the user access website the they either donate or do other things but if the donate then they donate from the system which flow of line is connected with the activation box from user and then donation is save in the database. If the donation is successful the user return to home page otherwise retry that process again.
* And in the case of sending feedback if the user want to send null data then it won’t let it to send that because before sending form validate whether there is null data or not so there is curved arrow touched in the activation box which is also called as self-message.

## 

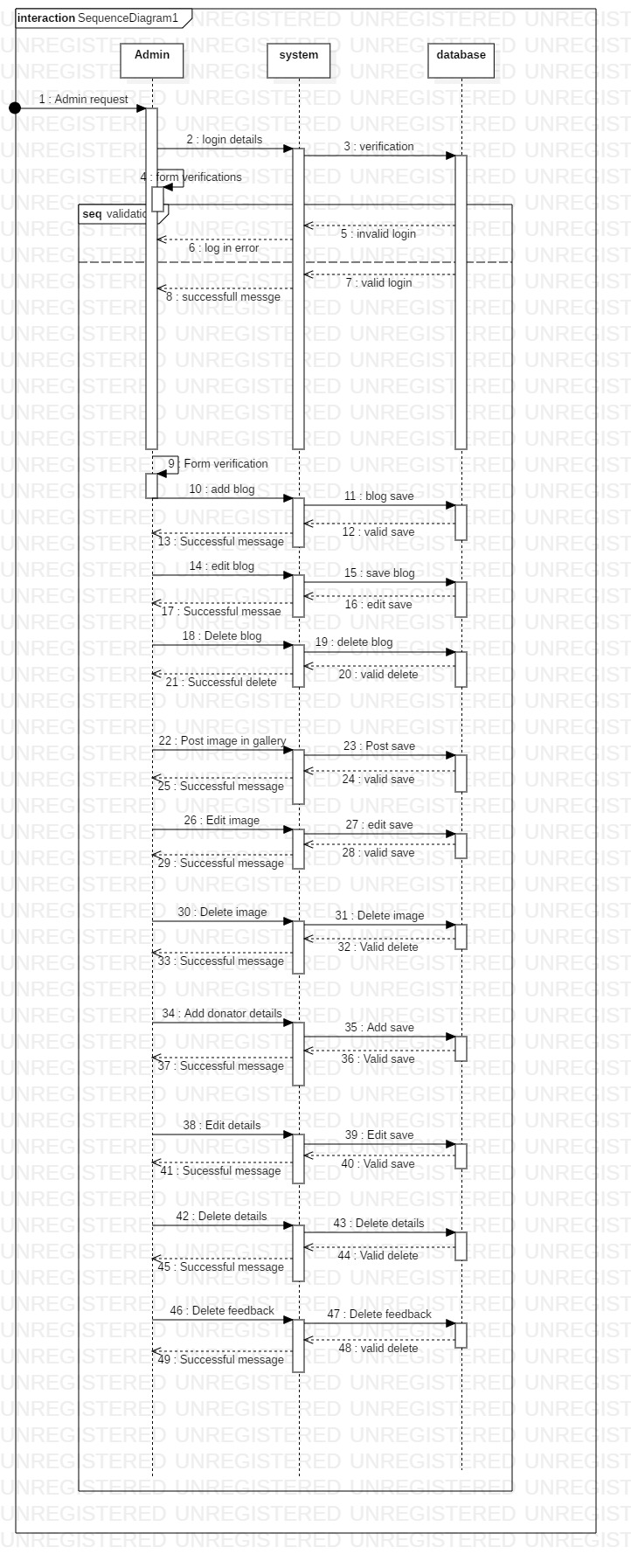


Fig: sequence diagram of admin.

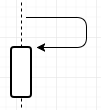
Justification

* The above sequence diagram display the overall operation of the admin. First of all, the admin log in to access the admin page if the login is successful then he can do rest of the work otherwise admin return with error message in login form again. This process repeats until and unless the admin successfully logged in.
* And when the admin is successfully logged in then admin can perform the task like add, update and delete blogs, images etc. here again form validation is performed in every form to prevent admin from sending null data so if admin wont fill any field then it won’t let it to send that data. After successfully completing the task then admin return back to the dashboard and do other work or log out from there.







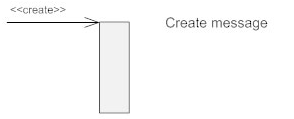


Fig: notation used while making sequence diagram.

# Database

## Data dictionary

Data dictionary is the collection of the information that shows the name, data type, length, key while creating table on the database. Here I have use MS Excel to design data dictionary of each classes.

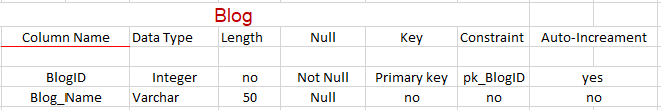


Fig: data dictionary of blog.

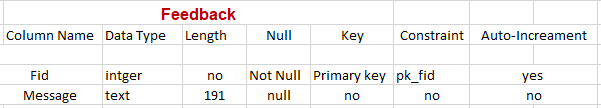


Fig: data dictionary of feedback.

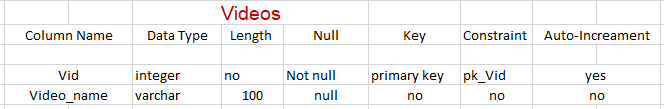


Fig: data dictionary of video.

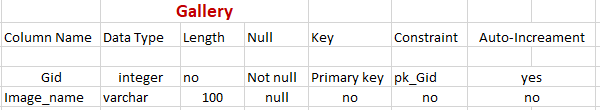


Fig: data dictionary of gallery.

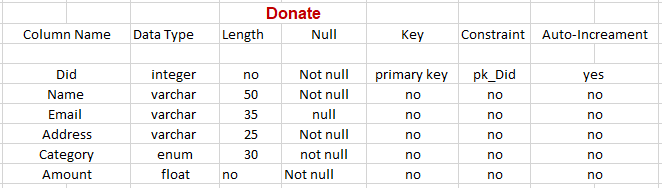


Fig: data dictionary of donate.

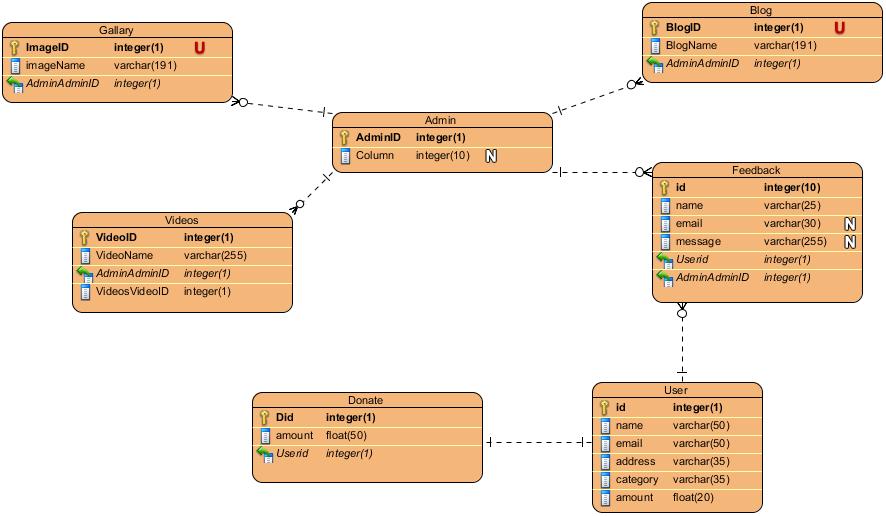


Fig: Entity relationship diagram of Dhurmus Suntali foundation.

Justification.

* Here I have made the Entity relationship diagram of my project in visual paradigm. Because many professionals use this software to design ER diagram and also I have used this software before too so that I feel comfortable to use it.
* And the above diagram shows that different table connected to each other. All the tables are connected with admin with one to many relationships because admin can perform the task multiple times like admin can create many blogs, can post many images into the gallery so they are linked with one to many relationships.
* There is also another table having relationships because one user can donate only one time so it is linked with one to many and also one user can send many feedbacks do it is connected with one to many relationship.

# Architecture

I have choose three tier architecture to do this project because

* It individually helps to sustain the elements on single platforms and then it develop system’s functional procedure, graphical user interface, system storage.
* I have chances to restructure my project or product and it not only look for present requirements but also for futures requirements, which will be easy for me to adapt changes made by the customer. So, I don’t think that I can get this features by using others architecture. That’s why I chose three tier architecture.

Basically, it is divided into three parts:

**Presentation layer** which help to display the web page using HTML/CSS/JS and whatever data is send by the user from the form which is contained in web page those data are delivered to the **application layer** and it runs the query and finally distributed to the database layer and at last the **database layer** perform query in database management system and then return the results to the application layer which setups it into the web page. It is also explained in the figure:

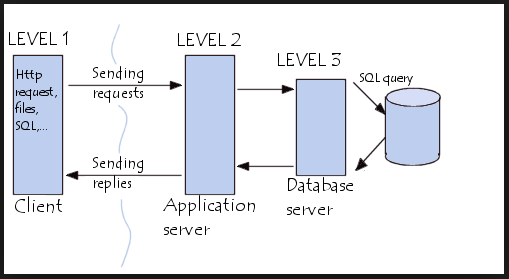


Fig: three tier architecture.

The advantages of three tier architecture are:

* Complex and enormous projects can be easily understand and maintain.
* The one tier can be easily update or maintain without impacting other part of the application layer.
* The modules can be reused.

# Architecture (User Design).

Architecture design is the way of representing the design of the software which also weighs all top level of design. Making user design makes easier to understand the system and also makes the judgement making method well-organized so that it is important to design.

## Prototyping

To design the prototype I have used software name balsamic mock up because

* It can be used in any type of device like laptop, tablet, desktop etc.
* It is fast, trustworthiness software and also can make prototype of any kind of software.
* It has features of drag and drop which saves time and resources and also it is secured than paper prototyping. I can redesign if the prototype is wrong by using balsamic but designing in paper I have to draw in another new paper which lost time and resources.



Fig: Digital prototyping about us page.

## 1.2 Home page

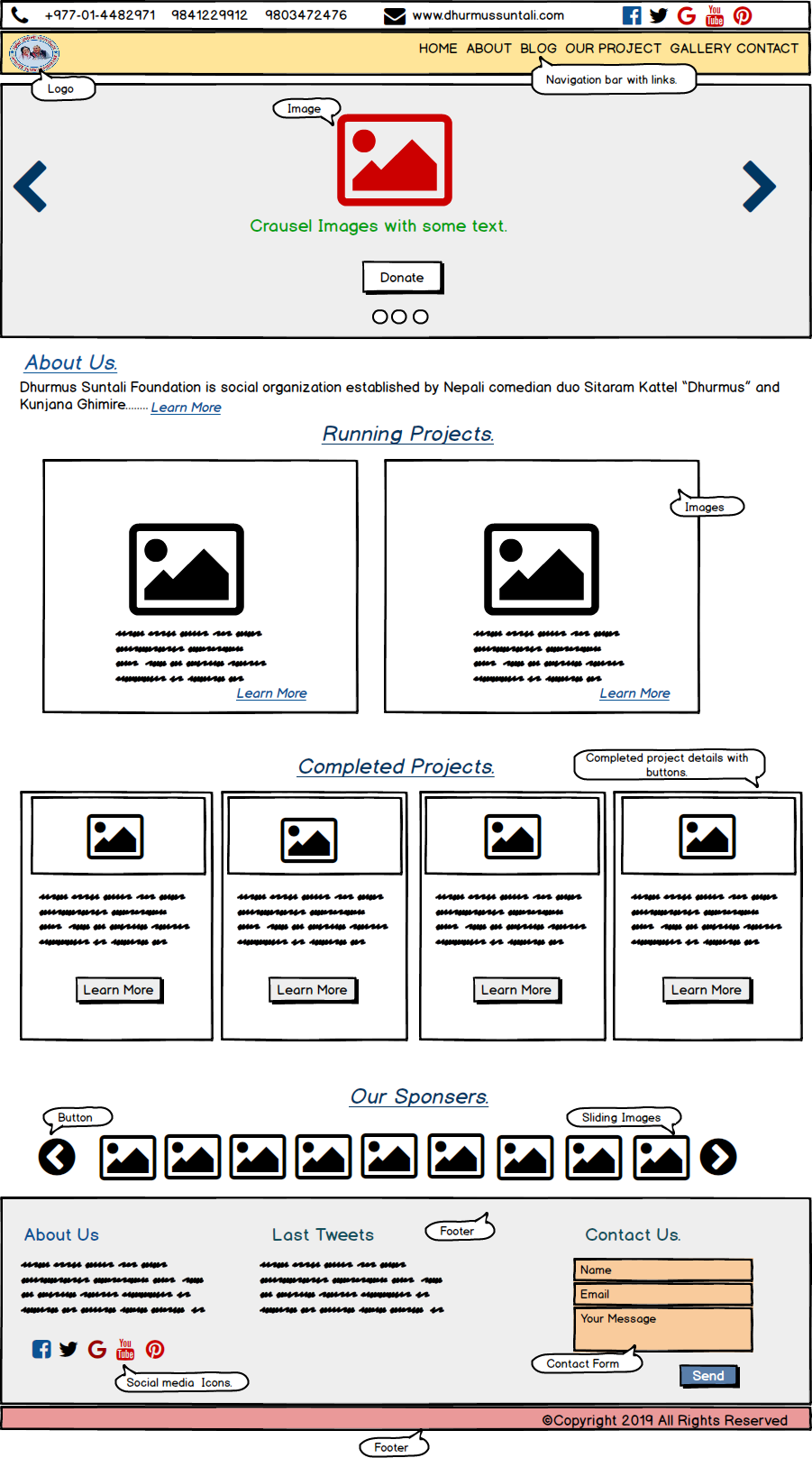


Fig: Digital prototyping of home page.

## – Blog

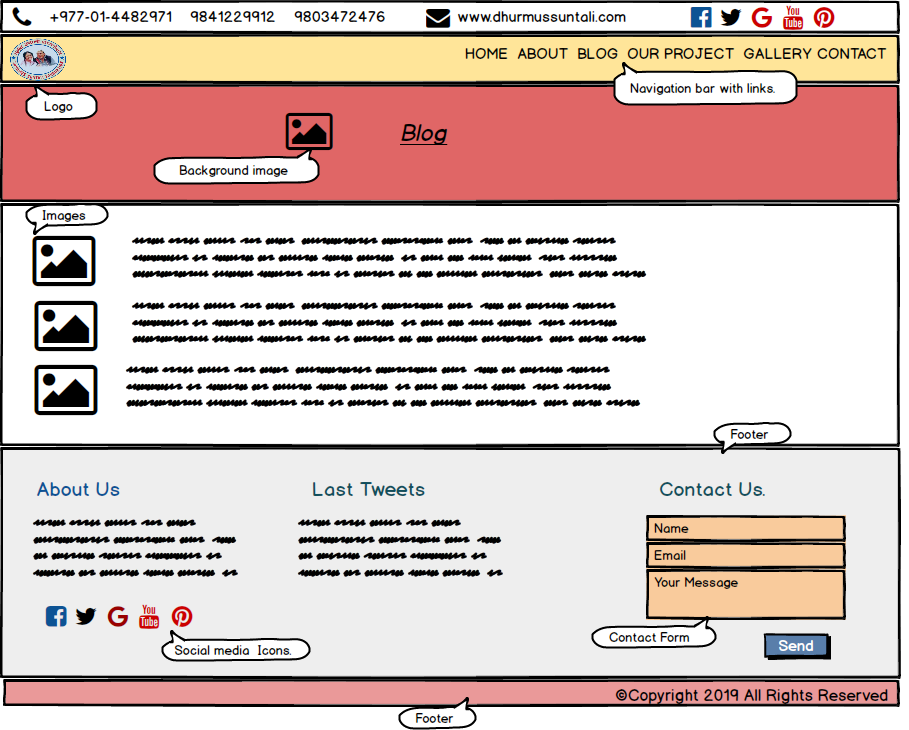


Fig: Digital prototyping blog page.

## – Our project

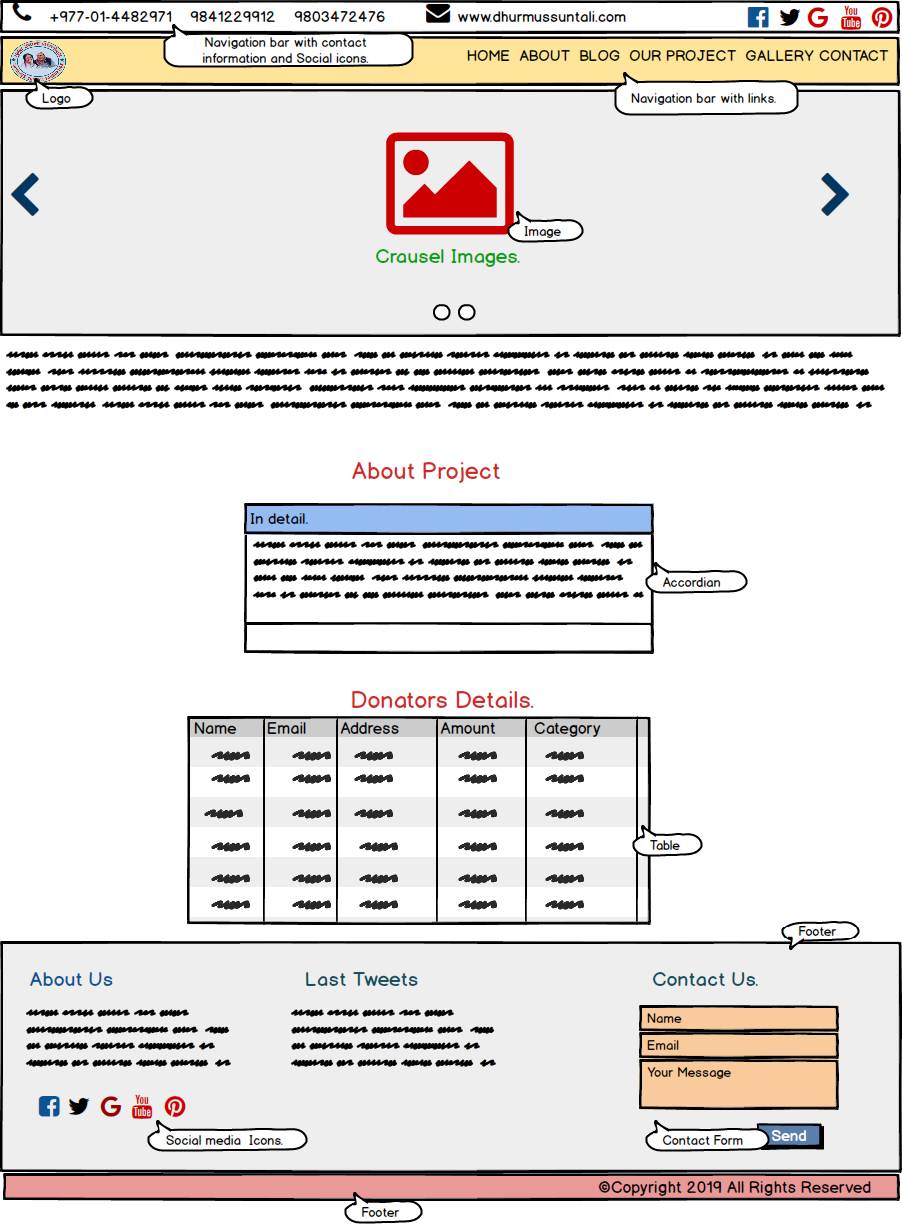


Fig: Digital prototyping of our project page.

## – Gallery

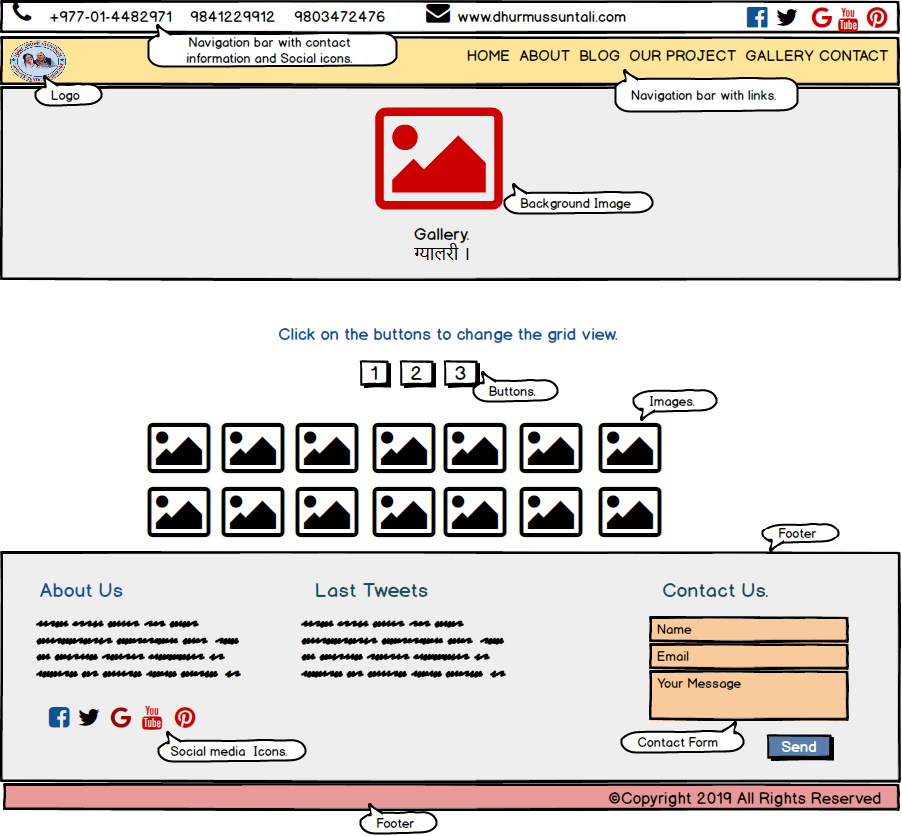


Fig: Digital Prototyping of gallery page.

## – Contact

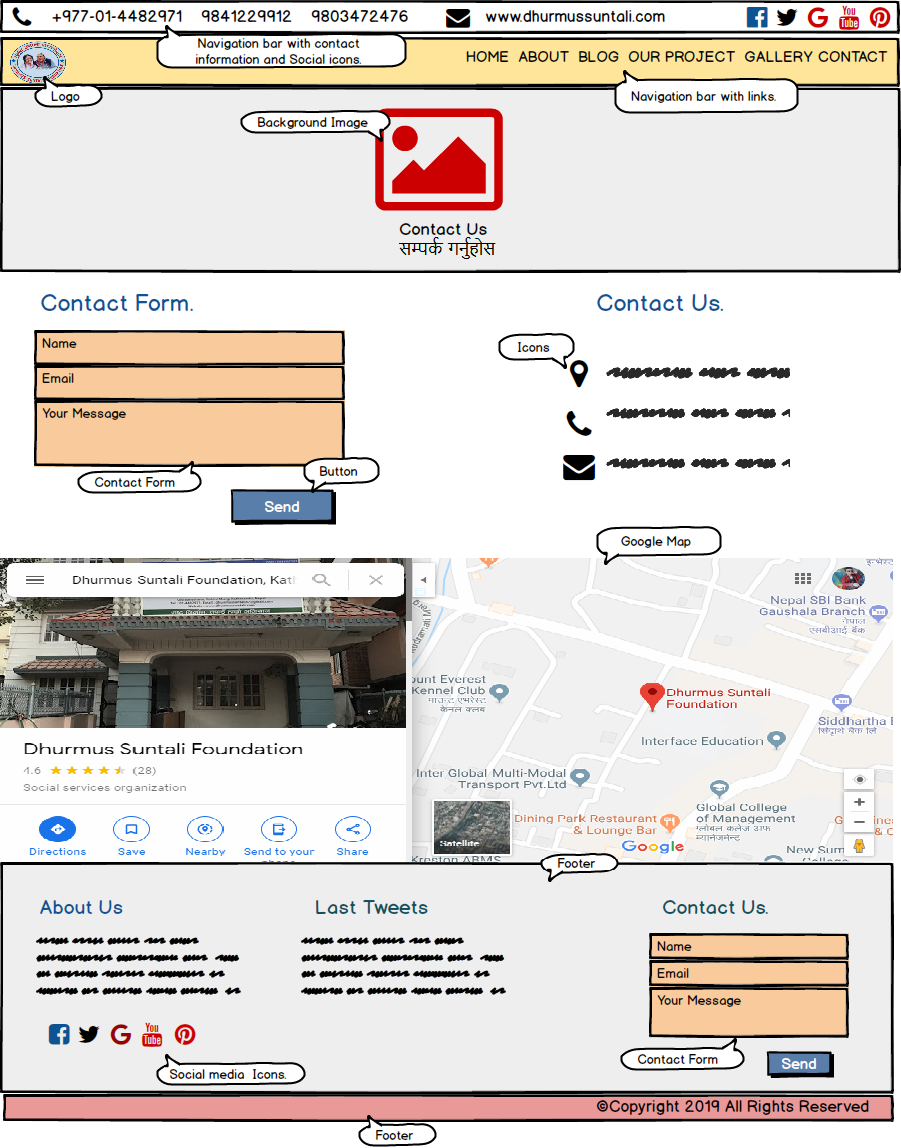


Fig: Digital Prototyping of contact page.

## – Admin login page

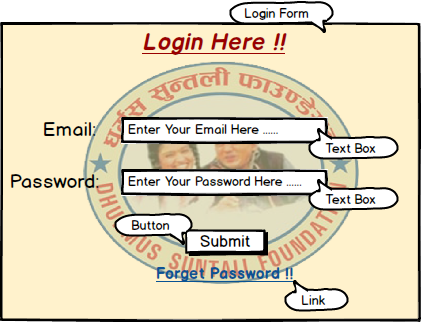


Fig: Digital prototyping of login page.

## - Admin page.

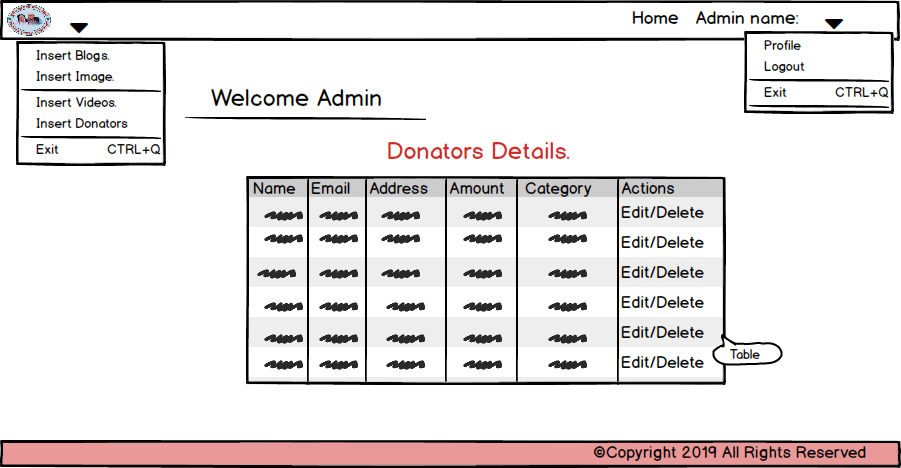


Fig: Digital prototyping of admin page.

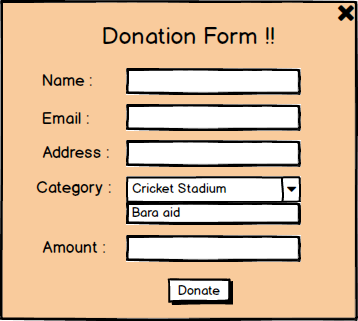


Fig: Donation Form.

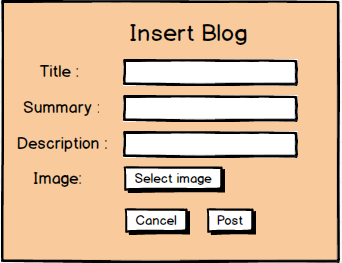


Fig: Insert blog form.

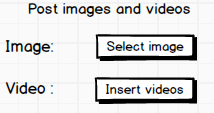


Fig: post image and video form.