1. Project Title

**Disaster Management System**

**TABLE OF CONTENTS**

**1. PROJECT TITLE ..................................................................................................** 3

**2. INTRODUCTION .................................................................................................. 4**

2.1 Objectives................................................................................................... 4

**3. PROJECT CATEGORY ....................................................................................... 5**

**4. TOOLS/PLATFORM, HARDWARE, and SOFTWARE REQUIREMENT SPECIFICATION ..................................................................................................** 6

4.1 Hardware Requirement ............................................................................... 6

4.2 Software Requirement ................................................................................ 6

**5. PROBLEM DEFINITION, REQUIREMENT SPECIFICATIONS .................... 7**

5.1 Problem definition ...................................................................................... 7

5.2 Requirement Specification .......................................................................... 8

5.3 Feasibility Study … ................................................................................... 9

5.4 Functional Requirements … .................................................................... 10

5.4.1 Technical Requirements .................................................................. 10

5.4.2 Nonfunctional Requirements…........................................................ 11

5.5 Project Planning & Scheduling ................................................................. 11

5.5.1 PERT Chart/ Task Network Chart .................................................... 11

5.5.2 GANTT Chart ............................................................................... 12

**6. SCOPE OF THE SOLUTION ............................................................................. 13**

**7. ANALYSIS ........................................................................................................... 14**

7.1 Data flow diagram (DFD) ......................................................................... 15

7.2 Entity Relationship Diagram (ERD).......................................................... 20

7.3 Class Diagram … .................................................................................... 21

**8. A COMPLETE DATABASE & TABLES ........................................................... 22**

**9. A COMPLETE STRUCTURE ............................................................................ 26**

9.1 Number of Modules & Their Description .................................................. 26

9.2 Process Logic of Each Module .................................................................. 28

9.3 Implementation Methodology ................................................................... 31

9.4 List of Reports that are Likely to be Generated ......................................... 31

**10. IMPLEMENTATION OF SECURITY MECHANISMS ................................... 32**

10.1 System Security ...................................................................................... 32

10.2 Database Security ................................................................................... 32

10.3 Hierarchical Security .............................................................................. 33

10.4 System Failures and Recovery ................................................................ 33

10.5 Data Security .......................................................................................... 33

**11. FUTURE SCOPE & FURTHER ENHANCEMENT ......................................... 34**

11.1 Further Enhancements ............................................................................ 34

11.2 Conclusion ............................................................................................. 34

**12. BIBLIOGRAPHY ................................................................................................ 34**

1. Introduction & Objective(s)

Introduction:-

Disaster management essentially deals with management of resources and information towards a disastrous event and is measured by how efficiently, effectively and seamlessly one coordinates these resources. The ability to effectively deal with disasters has become a challenge to modern technology.

Disaster is any occurrence that causes damage, ecological disruption, loss of human life, deterioration of health and health services on a scale, sufficient to warrant an extraordinary response from outside the affected community or area. Disaster management: The body of policy and administrative decisions and operational activities that pertain to various stages of a disaster at all levels. It encompasses all the aspects of planning for and responding to disasters, including both pre and post disaster activities.

Disaster management at the individual and organisational level deals with issues

of planning, coordination, communication and risk assessment. Accordingly, this project

also covers these subjects to enhance their ability for better disaster response. For the most part, the case studies in this project deal with various disasters to promote discussion on issues of disaster and development, national policies on disaster management across the world and problems in disaster management.

This project has been divided into three sections, namely, Disaster ,rescue and Disaster Prevention and Control; and Disaster Risk Analysis and Management.

The project also describe about the numbers of death, injured and loss of property and resources. Therefore, the purpose of this project will be served if in the end it helps to know the economic losses and reduce injuries and death to the affected population.

.

Objective(s):-

Minimize the casualties.

Prevent further casualties.

Rescue the victims.

First aid.

Medical care.

Reconstruction.

Disaster management is the responsibility of all spheres of govt.

Disaster management should use resources that exist for a day-to-day purpose.

Organizations should function as an extension of their core business

The project is aimed to developing a web based disaster management system tools, which is of important to most of the disaster prone countries. Most disaster prone countries in the world has lack of efficient disaster management system that will help in the efficient provision of rescue and relief to the disaster affected areas.

1. Project Category

**This application developed using Python technology, HTML/CSS is used to front end designing but JSP are used as tools and techniques and MYSQL is used as a back end or database. The connection between the front-end and the back-end is established by using Python using Django. Some style sheets and scripting technologies are used to enhance the dynamisms of the project.**

1. Tools/Platforms &

Hardware/Software Requirement Specification

***3.1 HARDWARE SPECIFICATIONS :***

* Processor : 64bit
* RAM : 4GB
* Hard Disk : 80GB

***3.2 SOFTWARE SPECIFICATIONS :***

* Operating System : Windows7 or above
* Technology Used : PYTHON
* Framework : Django
* Back End : MYSQL
* Web technology : Html CSS
* Browser : Google chrome

**Python** : Python is an [interpreted](https://en.wikipedia.org/wiki/Interpreted_language) [high-level](https://en.wikipedia.org/wiki/High-level_programming_language) [general-purpose programming language](https://en.wikipedia.org/wiki/General-purpose_programming_language). Its design philosophy emphasizes [code readability](https://en.wikipedia.org/wiki/Code_readability) with its use of [significant indentation](https://en.wikipedia.org/wiki/Off-side_rule). Its [language constructs](https://en.wikipedia.org/wiki/Language_construct) as well as its [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) approach aim to help [programmers](https://en.wikipedia.org/wiki/Programmers) write clear, logical code for small and large-scale projects. Python is [dynamically-typed](https://en.wikipedia.org/wiki/Type_system#DYNAMIC) and [garbage-collected](https://en.wikipedia.org/wiki/Garbage_collection_(computer_science)) It supports multiple [programming paradigms](https://en.wikipedia.org/wiki/Programming_paradigm), including [structured](https://en.wikipedia.org/wiki/Structured_programming)

(particularly, [procedural](https://en.wikipedia.org/wiki/Procedural_programming)), object-oriented and [functional programming](https://en.wikipedia.org/wiki/Functional_programming).

**MySQL** : MYSQL is an [open-source](https://en.wikipedia.org/wiki/Open-source_software) [relational database management system](https://en.wikipedia.org/wiki/Relational_database_management_system) (RDBMS). Its name is a combination of "My", the name of co-founder [Michael Widenius](https://en.wikipedia.org/wiki/Michael_Widenius)'s daughter, and "[SQL](https://en.wikipedia.org/wiki/SQL)", the abbreviation for [Structured Query Language](https://en.wikipedia.org/wiki/Structured_Query_Language). A [relational database](https://en.wikipedia.org/wiki/Relational_database) organizes data into one or more data tables in which data types may be related to each other; these relations help structure the data

1. **Problem Definition, Requirements Specifications,**

Project Planning & Scheduling

**5. PROBLEM DEFINITION**

Now a days increasing the use of the plastic materials in human life. The main headache is the disposal of plastic waste. In some case people can dropped the plastic waste in landfill or the plastic waste can burn it. It was very harmful and pollute our environment and earth. . So with this project we are going to develop website which will help customers to add waste location to remove the plastic waste from our home and surrounding to keep clean our environment. Plastic recycling is the process of recovering waste or scrap plastic and reprocessing the material into functional and useful products. This activity is known as the plastic recycling process. The recycle unit collect the waste and recycle process to produce new useful products. The customers can purchase recycled new products and make payment through online. Our project will also provide the facilities like security, online payment and the web application will access in anywhere, anytime.

**5.1 Requirements Specification**

The first step of designing a system is to identify the drawbacks in the existing system. The working of the existing system is examined and studied for this purpose. The data is necessary as inputs to the analysis, which is specified based upon the requirements of those directing the analysis or customers (who will use the finished product of the analysis). The general type of entity upon which the data will be collected is referred to as an experimental unit (e.g., a person or population of people). Specific variables regarding a population (e.g., age and income) may be specified and obtained. Data may be numerical or categorical .Data is collected from a variety of sources.

The software requirement specification is produced at the culmination of the analysis task the function and performance allocated to software a part of system engineering are refined by establishing a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints, appropriate validation criteria, and other data pertinent to requirements.

**5.1.1 Functional Requirements:**

This project deals with the different service packages. The web application consists of different modules. The project has two sections namely server side and client side. The main modules are administrator module and customer module and recycle unit module. The administrator module deals with the manage packages, manage user accounts, update all modules, etc. The customer module includes the registration of the customer and add waste location, and purchase products, and enter feedbacks, etc. The recycle module include the registration of the recycle unit and add workers and collect waste and recycle it to produce new products, etc.

Project Planning & Scheduling:-

**PERT Chart/ Task Network Chart:-**

The Program Evaluation Review Technique (PERT) is the cost and time management system. PERT organizes that project is complex that some task must be completed before other can be stated and that the appropriate way to manage a project is to define and control each task. Because projects often fall behind schedules, PERT is designed to facilitate getting a project back on schedule. The PERT chart gives a graphical representation of this information.

Depending on the working priorities, the entire project can be subdivided into the following main modules, those are:-

We can construct our activities plain as follows:-

* Admin
* User
* Organisation
* Hospital
* Volunteer

**Activity Activity Name**

A Admin

B User

C Organisation

D Hospital

E Volunteer

**Chart:-**

|  |  |  |
| --- | --- | --- |
| **Activity** | **Predecessor Activity** | **Time Estimated Weeks (Individual)** |
| A | ……… | 4 |
| B | A | 4 |
| C | B | 2 |
| D | C | 4 |
| E | D,E | 3 |

**GANTT Chart:-**

**A** Gantt chart is a popular type of bar chart that illustrates a project schedule. Gantt charts illustrate the start and finish dates of the terminal elements and summary elements of a project. Terminal elements and summary elements comprise the work breakdown structure of the project.

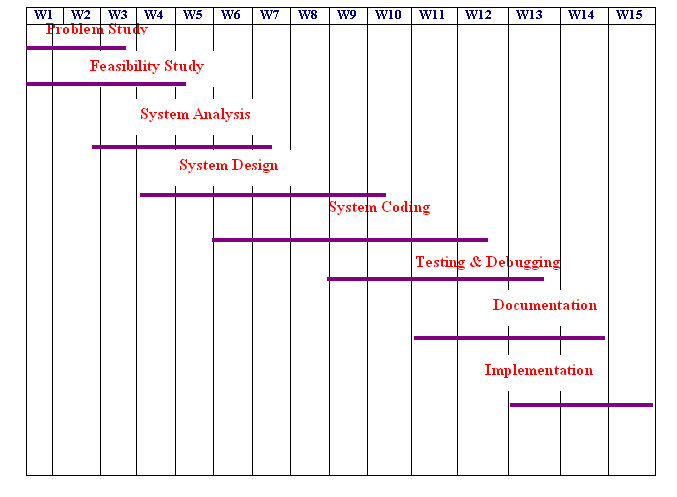


Fig: GANTT Chart

1. **Scope of the Solution**

Disaster management essentially deals with management of resources and information towards a disastrous event and is measured by how efficiently, effectively and seamlessly one coordinates these resources. The ability to effectively deal with disasters has become a challenge to modern technology.

Disaster management at the individual and organisational level deals with issues

of planning, coordination, communication and risk assessment. Accordingly, this project

also covers these subjects to enhance their ability for better disaster response. For the most part, the case studies in this project deal with various disasters to promote discussion on issues of disaster and development, national policies on disaster management across the world and problems in disaster management.

There are registered people in the system. They may be hospitals, organizations, volunteers, normal users. They are approved by the administrator of the system. It is the responsibility of the administrator to register hospitals to the system.

**7. Analysis (DFDs, ER Diagrams/ Class Diagrams etc.)**

The whole approach of analysis of problem should however be based around critical factors like the availability of information for making the decision, the time available for processing the data i.e. the realism. System Requirement Specification or SRS had been prepared after proper discussion with the persons attached with the mentioned “**OSCM**”. Software project management begins with a set of activities collectively called project planning. Software project planning actually encompasses all of the activities. Planning involves estimation- to determine how much money, how much effort, how many resources, and how much time it will take to build a specific software-based system or product.

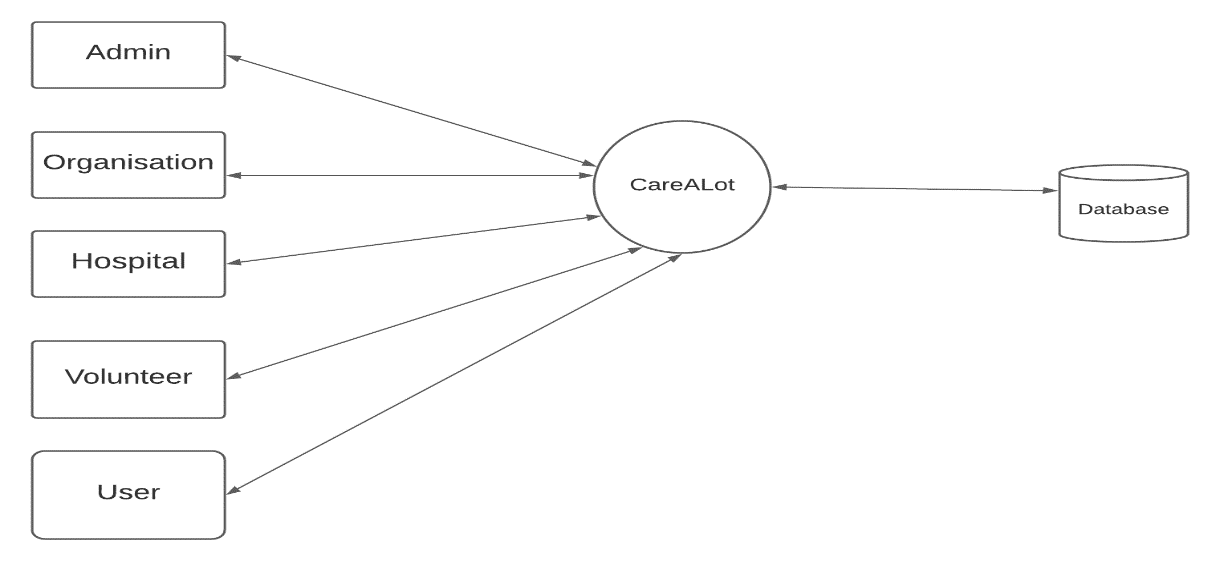
**Data Flow Diagram (DFDs):-**

Data flow diagrams models the passage of data in the system and are represented by lines joining system components. Flows of data in the system can take place between:-

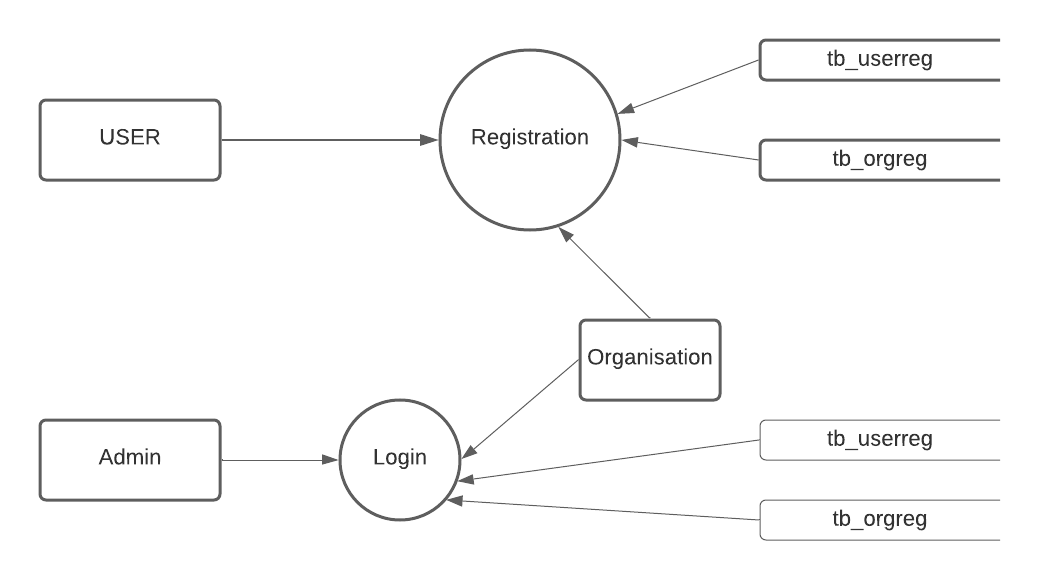
* Between two processes,
* From a data store to a process,
* From a process to data store,
* From a source to process and
* From a process to a sink.

Though the system mainly consists of two parts viz. online admission and online examination and other parts are going to be automated gradually…, so DFD is also, illustrated in two parts, respectively…

**LEVEL 0 :**



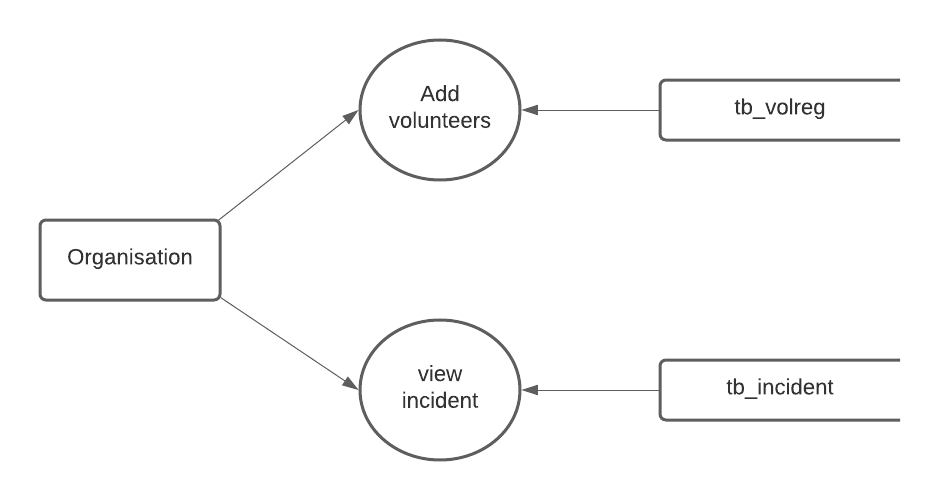
**Level 1:ADMIN**



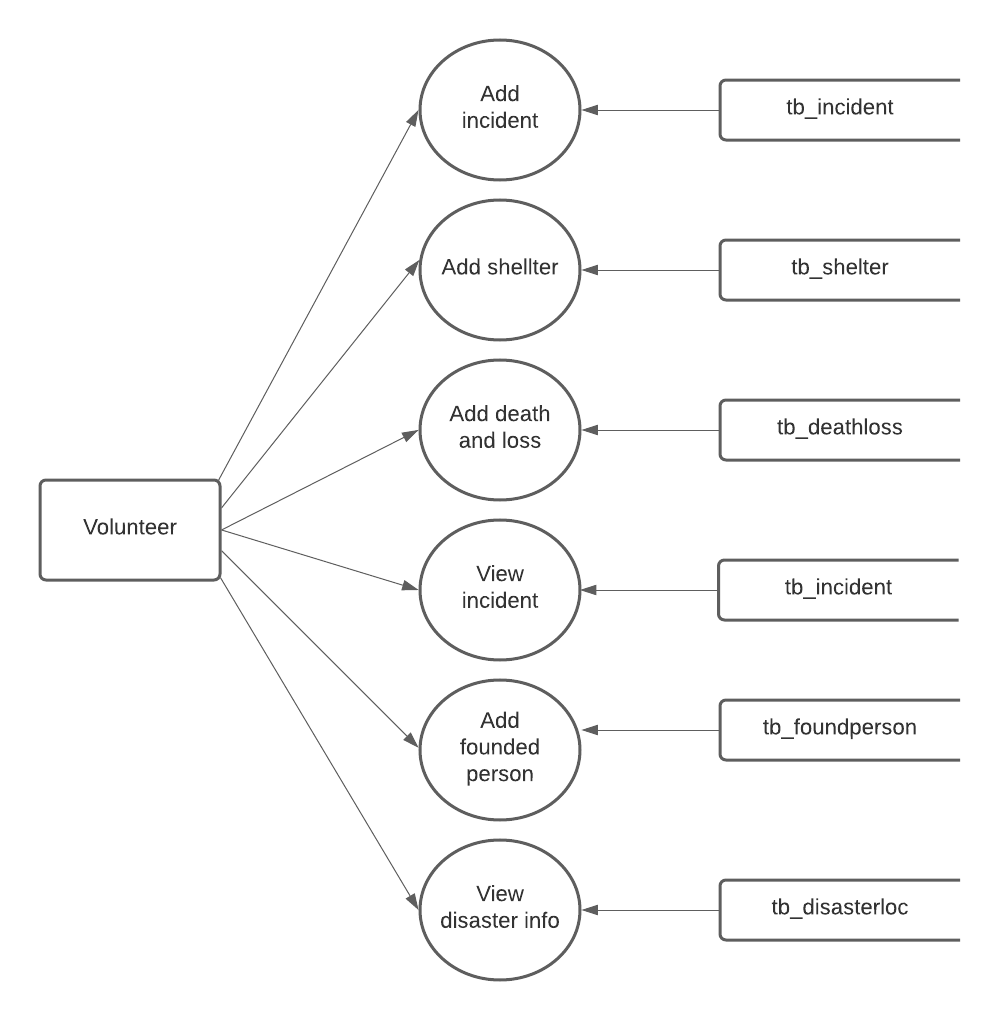
**Level1.1 admin**



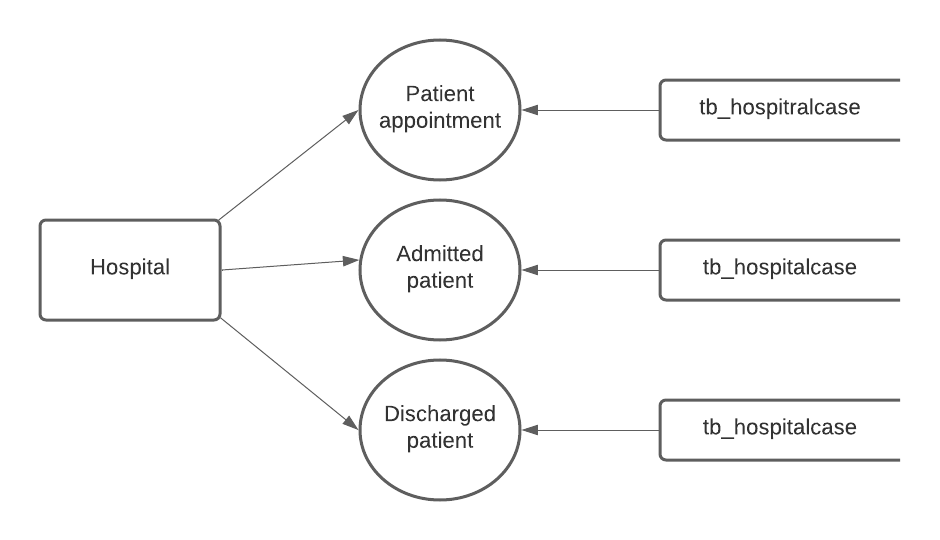
**Level 1.2:Organisation**



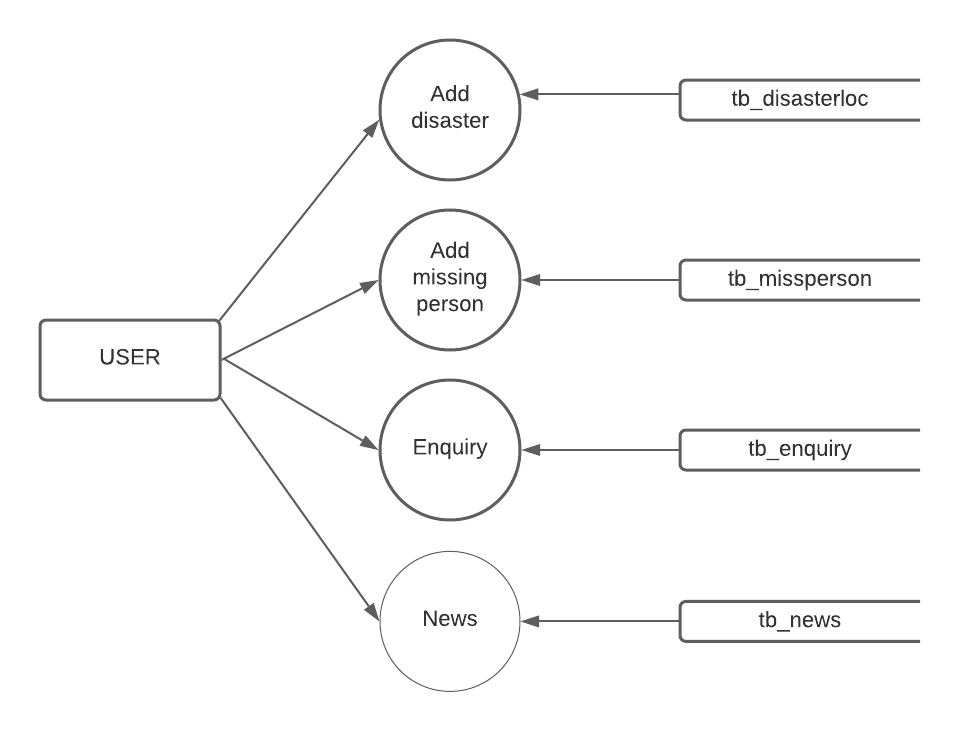
**Level 1.3: volunteer**



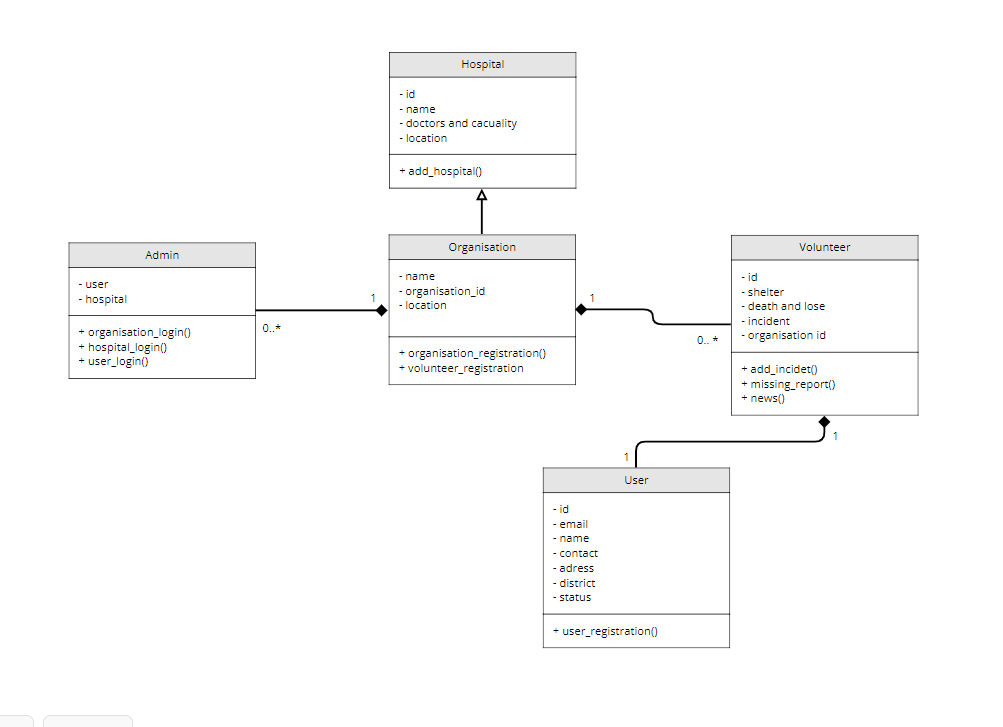
**Level 1.4: Hospital**



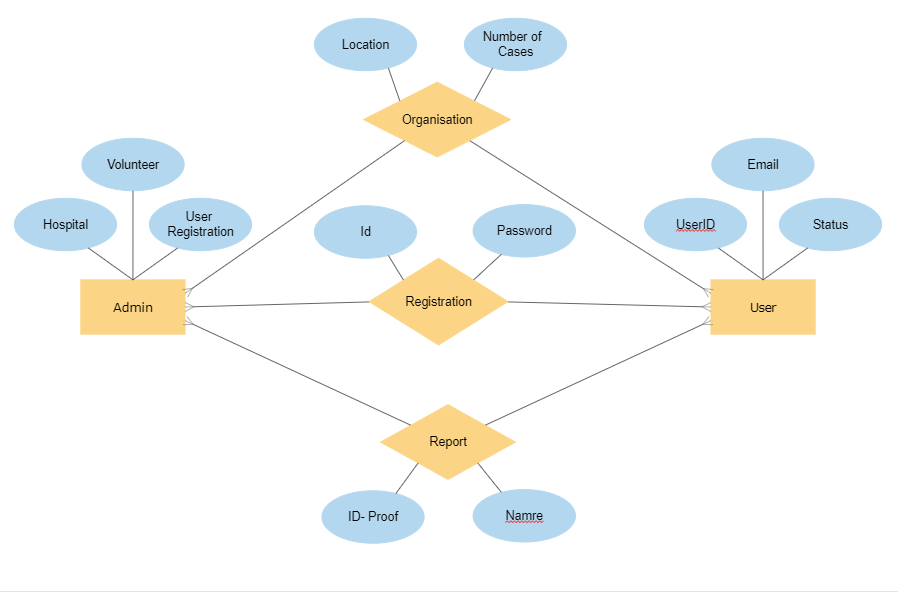
**Level 1.5:USER**



**Class Diagram :**

****

**Entity Relationship Diagram (ERD):-**



**8. A Complete Database & Tables**

Thisis a large enough application, and for building such a type numbers of tables are required. The number of tables is gradually increasing in course of development, as well as with the application of Normalization. Here only a few of those are mentioned, mainly the masters one, just to clear an overview of the application.

**TABLES**

**1.tb\_user-registration**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Userid | Int (11) | Primary Key | Number |
| Email | Varchar (40) | None | Email id |
| Mobile number | Varchar (40) | None | Number |
| District | Varchar (40) | None | District |
| Password | Varchar (40) | None | Password |
| Status | Varchar (40) | None | Status |

**2.tb\_Organisation-registration**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| id | Int (11) | Primary Key | Organisation id |
| Name | Varchar (40) | None | Nameof organisation |
| Address | Varchar (40) | None | Organisation address |
| District | Varchar (40) | None | District |
| City | Varchar (40) | None | City |
| Email | Varchar (40) | None | Email id |
| Password | Varchar (40) | None | Password |
| Status | Varchar (40) | None | Status |

**3.tb\_hospital-registration**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | Primary Key | Hospital id |
| Name | Varchar (40) | None | Hospital name |
| Address | Varchar (40) | None | Hospital address |
| Mobile number | Varchar (40) | None | Number |
| District | Varchar (40) | None | District |
| City | Varchar (40) | None | City |
| Email | Varchar (40) | None | Email id |
| Password | Varchar (40) | None | Password |
| Doctor | Varchar (40) | None | No of doctors |
| Casualty | Varchar (40) | None | No of Casualty |
| Ambulance | Varchar (40) | None | No of ambulance |
| Bed | Varchar (40) | None | No of bed |
| Department | Varchar (40) | None | Departments |
| Authority name | Varchar (40) | None | Name of authority |
| Address | Varchar (40) | None | Authority address |
| Status | Varchar (40) | None | Status |

**4.tb\_Volunteer-registration**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| id | Int (11) | None | Volunteer id |
| name | Varchar (40) | None | Volunteer name |
| Dob | Varchar (40) | None | Date of birth |
| gender | Varchar (40) | None | gender |
| address | Varchar (40) | None | Address |
| district | Varchar (40) | None | District |
| email | Varchar (40) | None | Email id |
| password | Varchar (40) | None | Password |
| Mobile number | Varchar (40) | None | mobilenumber |
| Alternate number | Varchar (40) | None | Alternate number |
| Id proof | Varchar (40) | None | Id proof of volunteer |
| Id number | Varchar (40) | None | Id proof number |
| occupation | Varchar (40) | None | occupation |
| Org\_id\_id | Int (11) | None | Foreign key org\_id |
| status | Varchar (40) | None | status |

**5.tb\_missing-report**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| id | Int (11) | None | Report id |
| name | Varchar (40) | None | Name of missing person |
| Gender | Varchar (40) | None | gender |
| Address | Varchar (40) | None | address |
| Last-loc | Varchar (40) | None | lastlocation |
| District | Varchar (40) | None | district |
| Email | Varchar (40) | None | Email id |
| contactnumber | Varchar (40) | None | contactnumber |
| Id proof | Varchar (40) | None | Id proof |
| Id\_number | Varchar (40) | None | Id number |
| height | Varchar (40) | None | height |
| weight | Varchar (40) | None | weight |
| Eye color | Varchar (40) | None | Eye color |
| image | Varchar (40) | None | image |
| status | Varchar (40) | None | status |
| User\_id id | Varchar (40) | None | Foreign key |
| Missing date | Varchar (40) | None | date |

**6.tb\_news**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int(11) | None | News id |
| Information | Varchar (40) | None | Information for news |
| Precautions | Varchar (40) | None | precautions |
| Date | Varchar (40) | None | date |

**7.tb\_shelter**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | Shelter id |
| Name | Varchar (40) | None | Shelter name |
| Address | Varchar (40) | None | address |
| City | Varchar (40) | None | city |
| Email | Varchar (40) | None | Email id |
| Mobile number | Varchar (40) | None | Mobile number |
| Voli\_id\_id | Int (11) | None | Foreign key |

**8.tb\_death\_loss**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | Death loss id |
| death\_male | Varchar (40) | None | No of death male |
| death\_female | Varchar (40) | None | No of death female |
| missing\_male | Varchar (40) | None | No of missing male |
| missing\_female | Varchar (40) | None | No of missing female |
| Loss | Varchar (40) | None | loss |
| incident\_id\_id | Int (11) | None | Foreign key |
| vol\_id\_id | Int (11) | None | Foreign key |

**9.tb\_allocated\_org**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | Allocated organisation id |
| Incident\_id\_id | Int (11) | None | Foreign key |
| Org\_id\_id | Int (11) | None | Foreign key |
| Status | Varchar (40) | None | status |

**10.tb\_allocated\_volunteer**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | Allocated volunteer id |
| Incident\_id\_id | Int (11) | None | Foreign key |
| vol\_id\_id | Int (11) | None | Foreign key |
| Status | Varchar (40) | None | status |

**11.tb\_founded person\_report**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | Person id |
| Name | Varchar (40) | None | Person name |
| Gender | Varchar (40) | None | gender |
| find\_loc | Varchar (40) | None | Find location |
| Height | Varchar (40) | None | height |
| Weight | Varchar (40) | None | weight |
| eye\_color | Varchar (40) | None | eye\_color |
| Status | Varchar (40) | None | status |
| find\_date | Varchar (40) | None | find\_date |
| incident\_id\_id | Int (11) | None | Foreign key |
| vol\_id\_id | Int (11) | None | Foreign key |
| Image | Varchar (40) | None | image |

**12.tb\_hospital case**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | Case id |
| Condition | Varchar (40) | None | condition |
| Room no | Int (11) | None | Room no |
| Admitted\_date | Varchar (40) | None | Admitted date |
| Discharged\_date | Varchar (40) | None | Discharged\_date |
| Status | Varchar (40) | None | status |
| fid\_id | Int (11) | None | Foreign key |
| hid\_id | Int (11) | None | Foreign key |
| vol\_id\_id | Int (11) | None | Foreign key |

**13.tb\_relief\_camp**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | Camp id |
| status | Varchar (40) | None | status |
| fid\_id | Int (11) | None | Foreign key |
| sid\_id | Int (11) | None | Foreign key |
| vol\_id\_id | Int (11) | None | Foreign key |
| shifted\_date | Varchar (40) | None | Shifted date |

**14.tb\_disaster location**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Data Type** | **Constraints** | **Description** |
| Id | Int (11) | None | location id |
| Latitude | Varchar (40) | None | Latitude |
| Longitude | Varchar (40) | None | Longitude |
| description | Varchar (40) | None | Description |
| place | Varchar (40) | None | place |
| uid\_id | Int (11) | None | Foreign key |
| incident | Varchar (40) | None | Incident name |
| status | Varchar (40) | None | status |

**9. A Complete Structure**

**Number of Modules & Their Description:-**

The application consists of number of modules and sub modules, of which, the most important are discussed briefly…

1. Admin
2. User
3. Organisation
4. Hospital
5. Volunteer

**MODULE DESCRIPTION**

1. **ADMIN**

Administrator or admin has total control over the application. Admin login by using his username and password .

Admin have the right to manage

● view users

● Organization

● Hospital

● Volunteer

● allocate task

● Add information

● View reports

The admin module has the overall control of the project.the admin is responsible for overall processing.the admin module can view the all registered users and it also view the disaster information added by the users and then it can allotte all disasters to all volunteers. the admin module handle the approval of organization ,hospital and volunteers to be registered.the admin can login with mail and its password.

The admin can add the news or informations about the climate,disasters,weather and latest news on theire home page. The admin has the responsibility to allocate the task added by the volunteer ,to specific organizations. The admin check the genuinty of the registered modules. The admin contains the overall details of the organization,hospital, and volunteer. The admin has the approve and remove operations. Only the approved organizations ,hospital and volunteers can successfully registered and do the work on our system. The admin provide the different responsibilitys to each modules.

The admin can control the disaster management system.the admin can view the allocated tasks details.and volunteer gives the latest situations in that locations.the admin provide the record of missing persons and founded persons. The founded persons are shifted to hospital or reliefcamp.

**2.USER**

The user login by using his user email id and password .The user can registered to the system with free payment.

The user has include:

* Patient appointment
* Admitted patient
* Discharged patient

The admin can view the registeres used. The user can add the disaster and missing reports to the management system. The disaster added by the user based on the location.The usercan enquire disaster information and missing persons. The disaster enquire with the incident name and location.the missing person can enquire with missing date and location.the user can view the latest news ana climate variations and weather informations and precautions.

**3.ORGANISATION**

Organisation login by using his username and password . Organisation is approved by the admin. It consists of

***●*** add volunteers

*●*view incident

● Allocate volunteer

Many organisations can register in disaster management system. It include name, location details,email id and password.The organisation can login with own email id and password.it can add volunteers details and allocate them.The organization management can view incident added by the user. And organization can allocate volunteers to disaster location . the organisation has the responsibility to control the volunteers and disaster location .they should be aware at all the times.it can view the incident .the admin is approve the organizations and after that the organisation is registered. The admin has the right to approve and reject.

**4.VOLUNTEER**

Volunteer login by using his username and password . Volunteer is added by appropriate organisations.and the volunteer is approved by the admin. It consists of

* Add incident
* Add shelter
* Add death and Loss
* View incident
* Founded person
* View disaster information

volunteer can view the all incident added by the users. They can view their incident status. They can add genuine situation of that disaster. The volunteer can inform the admin. They can also add shelters for surviving peoples who overcome the disasters. They can also add death and missing peoples. A volunteer can view the incident allotted by the admin. Volunteer management responsible for add the details of founded peoples. They can also view disaster information that added by the user. The volunteer can shift the founded peoples to hospital or relief camp.

**5.HOSPITAL**

Hospital is login by using username and password. It consists of

● Patient appointment

● Admitted patient

● Discharged patient

The hospital can register on this system. They can view the details of patients. The volunteer module can shift the founded peoples to hospital, if they have any injurys or health issues.the hospital module can take patient appointments if they have any health issues.the hospital module contains the no of available docters and casualties.also add the available department,no of ambulance,contact details. The hospital module approved by the admin when hospitals registered.they can admit patient fir further treatments based on docters opinion.the it contain the record of patients health details.the hospital also has the record of discharged patient with the date.the hospital registered by the authorized management .so the hospital module contains the details of who are registered to system

**9.2 Implementation of security mechanisms**

Security involves both policies and mechanisms to protect the data and ensures that it is not accessed, altered or deleted without proper authorization. There are two dimensions for the protection of data in the data base. The logged user can only perform the operations. This ensures that the confidentiality of the data is maintained, second, the data must be protected from accidental or intentional corruption or destruction.

The security features are considered while developing the system, so as to avoid the errors and omissions that may lead to serious problems. The system may have to face the unwanted events called threats. A threat to a computer system is any events that adversely affect the one or more assets or resources, which make up the system. An event can be any of the following:

1. Interruption of communication.
2. Destruction of hardware.
3. Modification of software.
4. Removal of programs.
5. Disclosure of information.

There are many methods for handling a threat.

1. Avoid it by altering the design.
2. Threat retention.
3. Threat reduction that is the frequency of occurrence of a threat is

reduced

The security measures of a computer system should be specified at an early stage in the design of the system. During the system operation each user should understand the procedures required to keep the system secure.

There are many possible threats to the security and integrity of any system where more than one user is associated with the system. Software integrity has become increasingly important. The attribute measures a system’s ability to withstand attacks, both accidental and intentional on its security. Attacks can be made on all the three components of software: programs date and documents.

In this project the data security, data validation checking methods are applied using a password authentication. All the data, which is entered by the administrator, will be validated.

**12. Future Scope & Further Enhancement**

The system is developed in android which makes the system more reliable and compatible with the other environments. The application proves better extensibility and flexibility for future enhancements. Any further application is possible with the same features guaranteed. The design of this System is in such a way that the addition of any modules if necessary is possible without affecting the integrity of the system.

So in future enhancement, since each and very applications should be expand and it should provide a way for updating of the system. All modules in the system is being developed carefully such that future enhancement does not affect the basic performance of the system.

This project has been completed in partial fulfillment and it came as a success in satisfying the requirements to great extent.

13.Bibliography

The End