1. INTRODUCTION

1.1 ABOUT THE PROJECT

"**ENVIRONMENTAL THREAT ANALYTICS**" is an online site to upload the image of polluted area by the user and corporation will make the arrangement to clean the area and return a rectified image to the user. Inefficient municipal solid waste management system may create serious negative environmental impacts like infectious diseases, land and water pollution, obstruction of drains and loss of bio diversity. Some waste will eventually rot, but not all, and in the process it may smell, or generate methane gas, which is explosive and contributes to the greenhouse effect. Leach ate produced as waste decomposes may cause pollution. Badly-managed landfill sites may attract vermin or cause litter.

Here the user can upload the image of the polluted location with the help of Google map and a text box select area. People can also request corporation to remove garbage from their residence. In return corporation (Admin) will send a reply and confirmation status with the amount user has to pay for the garbage removal. User can make payment through this site.

The system is mainly focused on waste management in municipality where there is no systematic garbage management. This website is developed in PHP as front end and MYSQL as back end. High speed of PHP gives it an advantage over other scripting languages and gives it an application in important administrations such as the server administration and mail functionalities. One of the important advantages of PHP is that it is Open Source. Therefore, PHP is readily available and is entirely free MySQL is a powerful open-source database server built based on a relational database management system (RDBMS) and is capable of handling a large concurrent database connection.

SYSTEM ANALYSIS

2. SYSTEM ANALYSIS

2.1 REQUIREMENT ANALYSIS

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the fact to improve the system. The analysis is the first step in the system development life cycle. Identifying the need for a new information system and launching an investigation for the required system which satisfies the exact requirement of the user is the first part of the development activities of a system. The Software Requirement Analysis process converts the complex task of eliciting and documenting the requirements of all these users. Modelling and analysing these requirements and documenting them as a basis for system design.

2.1.1 EXISTING SYSTEM

Studying the current system is a method that is used to gather the requirements in the research. The purpose of studying the current system is to identify the existing entities and also gather requirements and identify problem in the current system. Few researches had been made on the manual system. The existing system has many drawbacks. In the existing systems employers clear the garbage periodically. No systematic approach towards clearing the waste.

Limitations of Existing System

- In the existing systems employers clear the garbage periodically.
- No systematic approach towards clearing the waste.
- Surroundings are not clean and bad odor of wastes emphasizes an unhealthy environment.
- To avoid all these limitations and make the working more accurately the system needs to be computerized in a better way In order to tackle all these situations computerization of the existing system is necessary.

2.1.2 PROPOSED SYSTEM

In this system, the user can upload the image of the polluted location with the help of Google map and a text box select area. People can also request corporation to remove garbage from their residence. In return corporation (Admin) will send a reply and confirmation status with the amount user has to pay for the garbage removal. User can make payment through this site. The proposed system overcomes many of the disadvantages that were found in the existing system.

This Project is coupled with material on how to use the various tool, subsets available in PHP and MYSQL. The need for today's software development is competence in a GUI based front-end tool, which can connect to Relational Database engines. This gives the programmer the opportunity to develop client server-based commercial applications. These applications give users the power and ease of a GUI with the multi-user capabilities of Novell, Windows-based RDBMS engines such as MYSQL.

2.1.3 ADVANTAGES OF PROPOSED SYSTEM

- User can upload the image of the polluted area and the rectified image will be send by the corporation.
- This site will provide an option for the users to request the corporation for removing the garbage from their residence and they can pay the bill online.

2.2 FEASIBILITY STUDY

The main objective of the feasibility study is to test the technical, social and economic feasibility of developing a system. This should be done before developing a system. This is done by investigating the existing system in the area under investigation and generating ideas about the new system. The feasibility study involves the analysis of the problem and the collection of relevant information such as the different data items which would be the input to the system, the processing

required to be carried out on these data, the output data required to be produced by the system as well as various constraints on the behaviour of the system.

Three key considerations involved in the feasibility analysis involve the size of the project, cost, and benefits. Thus, when a new project is proposed, it normally goes through a feasibility study before it is approved for development.

Type's are-

- Economic Feasibility
- Operational Feasibility
- Technical Feasibility

2.2.1 ECONOMIC FEASIBILITY

The economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system. A cost evaluation is weighed against the ultimate benefit derived from the proposed system. The economic analysis is the most frequently used method for evaluating the effectiveness of a candidate system. If benefits outweigh costs then the decision is taken to design and implement the system. Further justifications or alterations in the proposed system will have to be made only if there is a chance of being approved. There is no need for any additional cost for both developing and implementation. So "**Environmental Threat Analysis**" is fully economically feasible for the current organization.

2.2.2 OPERATIONAL FEASIBILITY

In the beginning farmers are not ready to implement new system'. After discussing the pros and cons of the proposed system they agreed to implement. The performance of the system is more accurate, user-friendly and effective. The proposed "**Environmental Threat Analysis**" supports the operational feasibility to a great extent.

2.2.3 TECHNICAL FEASIBILITY

In Technical feasibility, the management determines whether the current

level of technology can support the proposed system. This system can be operated with minimum technical support. It uses PHP as front end, MYSQL as a database at windows platform and Mozilla, Firefox and Google Chrome as the browser. It also provides accurate and reliable data security. The proposed "**Environmental Threat Analysis**" supports the technical feasibility to a great extent.

2.3 SOFTWARE REQUIREMENT SPECIFICATION

A software Requirement Specification (SRS) is a requirement specification for a software system is a complete description of the behaviour of a system to be developed. It includes a set of use cases that describe all the interactions the users will have with the software. Use cases are also known as Functional Requirements. In addition to the use cases , the SRS also contain non-functional requirements. In addition to the use cases; the SRS also contains non-functional(or supplementary) requirements. Non-functional requirements are requirements which impose constraints on the design or implementations(such as performance engineering requirements , quality standards, or design constraints).

Characteristics of good SRS

A good SRS document has certain characteristics that must be present. The characteristics are:

- **Correctness:** An SRS is correct if every requirement included in the SRS represents something required in the final system.
- **Completeness:** An SRS is complete when it is documented after.
- **Unambiguous:** An SRS is unambiguous if and only if every requirement started has one and only interpretation. Requirements are often written in a natural language.
- **Verifiable:** An SRS is verifiable if and only if there exists some cost-effective process that can check whether the final products meets the requirements.
- **Modifiable:** An SRS is modifiable if its structure and styles are such that any necessary change can be made easily while preserving completeness and inconsistency.
- **Traceable:** An SRS is traceable if the original of each of the requirements is

clear and if it facilitates and referencing of each requirements in future development or enhancement documentation.

- **Consistency:** Consistency in the SRS is essential to achieve correct results across the documentation.
- **Testability:** An SRS should be written in such a way that is possible to create plan to confirm whether specification can be met and requirements can be met and the requirements can be delivered.
- **Clarity:** An SRS is clear when it has a single interpretation for the author(analysis), the user, the end user, the stakeholder, the developer, the tester and the customer.

2.3.1 FUNCTIONAL REQUIREMENTS

A modular design reduces the complexity, facilitates change and results in easier implementation by encouraging parallel development of different parts of a system. Software with effective modularity is easier to develop because function may be compartmentalized and interfaces re simplified. Software architecture embodies modularity, that is , software is divided in to separately named and addressable components called modules, which is integrated to satisfy problem requirements. Modularity is the single attribute of the software that allows a program to be intellectually manageable.

The Proposed System is divided into two modules:

- Admin
- Users

Admin module:

In the Admin module, admin has the right to log into the website. The main functions carried out by the admin are manage users, manage images of the

locations, Upload cleaned area, view feedback, view request from the user, Send request reply, view confirmation, Update confirmation status with payment, view payments, ,manage notification.

- Login
- Manage users
- Manage locations,
- Upload cleaned area
- view feedback
- view waste disposal request
- Send request reply
- View confirmation
- Update confirmation status with payment
- View payments,
- Manage notification.

User module:

In this module, User has the power to login to the website. The main functions carried out by the user includes Photo upload, Search location wise details, View uploaded photos, Edit details, Delete locations, View cleaned area, Send feedback, Send request, View request reply, Send confirmation, View status, Make payment, View notification, View reports, Edit profile.

- Login
- Photo upload
- Search location wise details
- View uploaded photos
- Edit details
- Delete locations
- View cleaned area

- Send feedback
- Send request
- View request reply
- Send confirmation
- View status
- Make payment
- View notification
- View reports
- Edit profile

2.3.2 HARDWARE SPECIFICATION

Processor	:	I3 or above
Processor speed	:	3.99 GHz
Cache Memory	:	512 KB Cache or above
Random Access Memory	:	4 GB

2.3.3 SOFTWARE SPECIFICATION

Operating System	:	Windows 7
Web Server	:	Apache
Database	:	MySQL
Front End	:	PHP

2.3.4 SOFTWARE REQUIREMENTS

Application Requirements

Operating System	:	Windows
Front End	:	HTML, Javascript, CSS, jQuery

Back End : SQL

HYPER TEXT TRANSFER PROTOCOL (HTTP)

HTTP is the protocol “spoken” by Web servers. Client programs that can speak HTTP, are known as browsers, are used by the people on the Internet to connect to HTTP servers. These servers provide access to distributed hyperlinked documents, applications and databases. HTTP is a stateless, object-oriented application-level protocol that has been in the existence since the early days of the WWW. NSCA HTTP is an HTTP/1.0 compliant Web Server and is credited with being one of the first HTTP servers available. It supports multiple schemes of authentication.

ABOUT PHP

PHP is mainly focused on server-side scripting, so you can do anything any other CGI program can do, such as collect form data, generate dynamic page content, or send and receive cookies. But PHP can do much more.

There are three main areas where PHP scripts are used.

- Server-side scripting. This is the most traditional and main target field for PHP. You need three things to make this work. The PHP parser (CGI or server module), a web server and a web browser. You need to run the web server, with a connected PHP installation. You can access the PHP program output with a web browser, viewing the PHP page through the server. All these can run on your home machine if you are just experimenting with PHP programming. See the installation instructions section for more information.
- Command-line scripting. You can make a PHP script to run it without any server or browser. You only need the PHP parser to use it this way. This type of usage is ideal for scripts regularly executed using Cron (on Unix or

Linux) or Task Scheduler (on Windows). These scripts can also be used for simple text processing tasks. See the section about Command line usage of PHP for more information.

- Writing desktop applications. PHP is probably not the very best language to create a desktop application with a graphical user interface, but if you know PHP very well, and would like to use some advanced PHP features in your client-side applications you can also use PHP-GTK to write such programs. You also have the ability to write cross-platform applications this way. PHP-GTK is an extension to PHP, not available in the main distribution. If you are interested in PHP-GTK, visit » its own website.

PHP can be used on all major operating systems, including Linux, many Unix variants (including HP-UX, Solaris, and OpenBSD), Microsoft Windows, Mac OS X, RISC OS, and probably others. PHP has also support for most of the web servers today. This includes Apache, Microsoft Internet Information Server, Personal Web Server, Netscape and iPlanet servers, Oreilly Website Pro server, Caudium, Xitami, OmniHTTPd, and many others. For the majority of the servers, PHP has a module, for the others supporting the CGI standard, PHP can work as a CGI processor.

So with PHP, you have the freedom of choosing an operating system and a web server. Furthermore, you also have the choice of using procedural programming or object-oriented programming or a mixture of them. Although not every standard OOP feature is implemented in PHP 4, many code libraries and large applications (including the PEAR library) are written only using OOP code. PHP 5 fixes the OOP related weaknesses of PHP 4 and introduces a complete object model.

With PHP you are not limited to output HTML. PHP's abilities include outputting images, PDF files and even Flash movies (using Libswf and Ming) generated on the fly. You can also output easily any text, such as XHTML and any other XML file. PHP can autogenerate these files, and save them in the file system, instead of printing it out, forming a server-side cache for your dynamic content.

One of the strongest and most significant features in PHP is its support for a wide range of databases. Writing a database-enabled web page is incredibly simple.

The following databases are currently supported:

We also have a database abstraction extension (named PDO) allowing you to transparently use any database supported by that extension. Additionally PHP supports ODBC, the Open Database Connection standard, so you can connect to any other database supporting this world standard.

PHP also has support for talking to other services using protocols such as LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (on Windows) and countless others. You can also open raw network sockets and interact using any other protocol. PHP has support for the WDDX complex data exchange between virtually all Web programming languages. Talking about interconnection, PHP has support for instantiation of Java objects and using them transparently as PHP objects. You can also use our CORBA extension to access remote objects.

PHP has extremely useful text processing features, from the POSIX Extended or Perl regular expressions to parsing XML documents. For parsing and accessing XML documents, PHP 4 supports the SAX and DOM standards, and you can also use the XSLT extension to transform XML documents. PHP 5 standardizes all the XML extensions on the solid base of libxml2 and extends the feature set adding SimpleXML and XMLReader support.

At last but not least, we have many other interesting extensions, theMnogoSearch, search engine functions, the IRC Gateway functions, many compression utilities (gzip, bz2, zip), calendar conversion, and translation. PHP is a powerful tool for making dynamic and interactive Web pages. PHP is the widely-used, free, and efficient alternative to competitors such as Microsoft's ASP. In our PHP tutorial, you will learn about PHP, and how to execute scripts on your server.

Taken directly from PHP's home, PHP.net, "PHP is an HTML-embedded scripting language. Much of its syntax is borrowed from C, Java, and Perl with a couple of unique PHP-specific features thrown in. The goal of the language is to allow web developers to write dynamically generated pages quickly." This is generally a good definition of PHP. However, it does contain a lot of terms you may not be used to. Another way to think of PHP is powerful, behind the scenes scripting

language that your visitors won't see!

When someone visits your PHP webpage, your web server processes the PHP code. It then sees which parts of it need to show to visitors(content and pictures) and hides the other stuff(file operations, math calculations, etc.) then translates your PHP into HTML. After the translation into HTML, it sends the webpage to your visitor's web browser.

ABOUT MySQL

MySQL is the world's most popular open-source database software, with over 100 million copies of its software downloaded or distributed throughout its history. With its superior speed, reliability, and ease of use, MySQL has become the preferred choice for Web, Web 2.0, SaaS, ISV, Telecom companies and forward-thinking corporate IT Managers because it eliminates the major problems associated with downtime, maintenance, and administration for modern, online applications.

Many of the world's largest and fastest-growing organizations use MySQL to save time and money powering their high-volume Web sites, critical business systems, and packaged software — including industry leaders such as Yahoo!, Alcatel-Lucent, Google, Nokia, YouTube, Wikipedia, and Booking.com. The flagship MySQL offering is MySQL Enterprise, a comprehensive set of production-tested software, proactive monitoring tools, and premium support services available in an affordable annual subscription.

MySQL is a key part of LAMP (Linux, Apache, MySQL, PHP / Perl / Python), the fast-growing open-source enterprise software stack. More and more companies are using LAMP as an alternative to expensive proprietary software stacks because of its lower cost and freedom from platform lock-in.

MySQL is a relational database management system (RDBMS) that runs as a server providing multi-user access to a number of databases. MySQL is officially pronounced /maɪˌɛskjuːˈɛl/ ("My S-Q-L"), but is often also pronounced /maɪˈsiːkwəl/ ("My Sequel"). It is named for original developer Michael Widenius's daughter.

The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation.

Members of the MySQL community have created several forks such as Drizzle, OurDelta, Percona Server, and MariaDB. All of these forks were in progress before the Oracle acquisition (Drizzle was announced 8 months before the Sun acquisition).

MySQL is primarily an RDBMS and therefore ships with no GUI tools to administer MySQL databases or manage data contained within. Users may use the included command-line tools,[16] or download MySQL frontends from various parties that have developed desktop software and web applications to manage MySQL databases, build database structure, and work with data records. One of the popular front ends is PhpMyAdmin

MySQL can be built and installed manually from source code, but this can be tedious so it is more commonly installed from a binary package unless special customizations are required. On most Linux distributions the package management system can download and install MySQL with minimal effort, though the further configuration is often required to adjust security and optimization settings.

Though MySQL began as a low-end alternative to more powerful proprietary databases, it has gradually evolved to support higher-scale needs as well.

It is still most commonly used in small to medium scale single-server deployments, either as a component in a LAMP-based web application or as a standalone database server. Much of MySQL's appeal originates in its relative simplicity and ease of use, which is enabled by an ecosystem of open source tools such as phpMyAdmin. In the medium range, MySQL can be scaled by deploying it on more powerful hardware, such as a multi-processor server with gigabytes of memory.

There are however limits to how far performance can scale on a single server, so on larger scales, multi-server MySQL deployments are required to provide

improved performance and reliability. A typical high-end configuration can include a powerful master database which handles data write operations and is replicated to multiple slaves that handle all read operations. The master server synchronizes continually with its slaves so in the event of failure a slave can be promoted to become the new master, minimizing downtime. Further improvements in performance can be achieved by caching the results from database queries in memory using memcached or breaking down a database into smaller chunks called shards which can be spread across a number of distributed server clusters.

ABOUT OPERATING SYSTEM

An Operating System may be viewed as an organized collection of software extensions of hardware, consisting of control routines for operating a computer and for providing an environment for execution of programs. Programs rely on facilities provided by the operating system to gain access to computer system resources such as files and input/output devices. I.e. the operating system acts as an interface between users and the hardware of a computer system.

By definition, a great PC experience means seamless operation across all of your hardware, software, and devices. We want to make sure you have a broad choice for the programs and devices you use with your PCs. Today, the Windows Vista Compatibility Center lists more than 10,500 applications and 9,500 devices with a compatible download or driver version. In most cases, the same software and hardware that works on Windows Vista will also work on Windows 7. Furthermore, we were careful to design Windows 7 so that the investments that our partners and our enterprise customers have made in Windows Vista will continue to deliver great value as they upgrade to Windows 7. We've worked closely with our partners from the earliest planning phase. Before writing code, we talked to original equipment manufacturers, the companies that make PCs. Before defining application programming interfaces, we talked to developers. And before adding support for new devices, we worked with device manufacturers to understand how next-generation hardware could create new scenarios.

Windows 7 Starter makes small notebook PCs and other PCs with limited hardware easier to use because it puts less between you and what you want to do—less waiting, less clicking, and less hassle connecting to networks. Windows 7

Starter is designed to meet basic needs and combines the latest in reliability and responsiveness with the familiarity of Windows. Key features that you will find in the Starter edition include the following:

- Broad application and device compatibility
- Safety, reliability, and responsiveness
- Ability to join a homegroup

Windows 7 Home Basic is an entry-level edition that will be offered in a number of emerging markets to customers with value-priced PCs. Windows 7 Home Basic makes it faster and easier to get to the programs and documents you use most often, so you can spend less time looking and more time doing the things you want to do. In addition to the benefits found in Windows 7 Starter, Home Basic includes the following:

- Live thumbnail previews
- Advanced networking support (ad hoc wireless networks and Internet connection sharing)
- Ability to extend your screen across multiple monitors

The best edition for consumers, Windows 7 Home Premium provides the best entertainment experience on your PC. You'll find easy ways to connect to other PCs and devices, all in a visually rich environment that makes everyday use simpler and more engaging. With Home Premium you can create a homegroup to share all of your favorite photos, videos, and music with other networked PCs running Windows 7. And you'll be able to watch shows for free when and where you want with Internet TV on Windows Media® Center. In addition to the features in Windows 7 Starter and Home Basic.

2.3.5 DATA FLOW DIAGRAM

The Data Flow Diagram was first developed by Larry Constantine as a way of expressing system requirements in a graphical form; this led to a modular design. A Data Flow Diagram, also known as "bubble chart" has the purpose of clarifying system requirements and identifying major transformations that will become

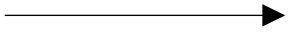
programs in system design. A DFD consists of a series of bubbles joined by lines. The bubbles represent data transformations and the lines represent data flows in the system.

A few guidelines to aid the derivation of a DFD

- The level 0 DFD should depict the software/ system as a single bubble
- Primary input and output should be noted
- Refinement should begin by isolating candidate processes, data objects, and the stores to be represented in the next level.
- All arrows and bubbles should be labelled with a meaningful name

To construct the DFD use the following components:

Data flows

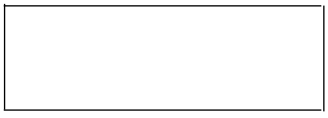


An arrow identifies the data flow in motion. It is the pipeline through which information is flown

Data Stores

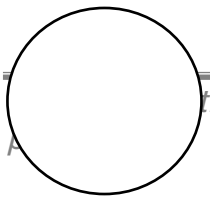
It represents the stores of data within the system. They are drawn as open ended rectangles with the unique box at enclosed end and the name of the data store at the open section.

External Entities



It defines the source of destination of the system.

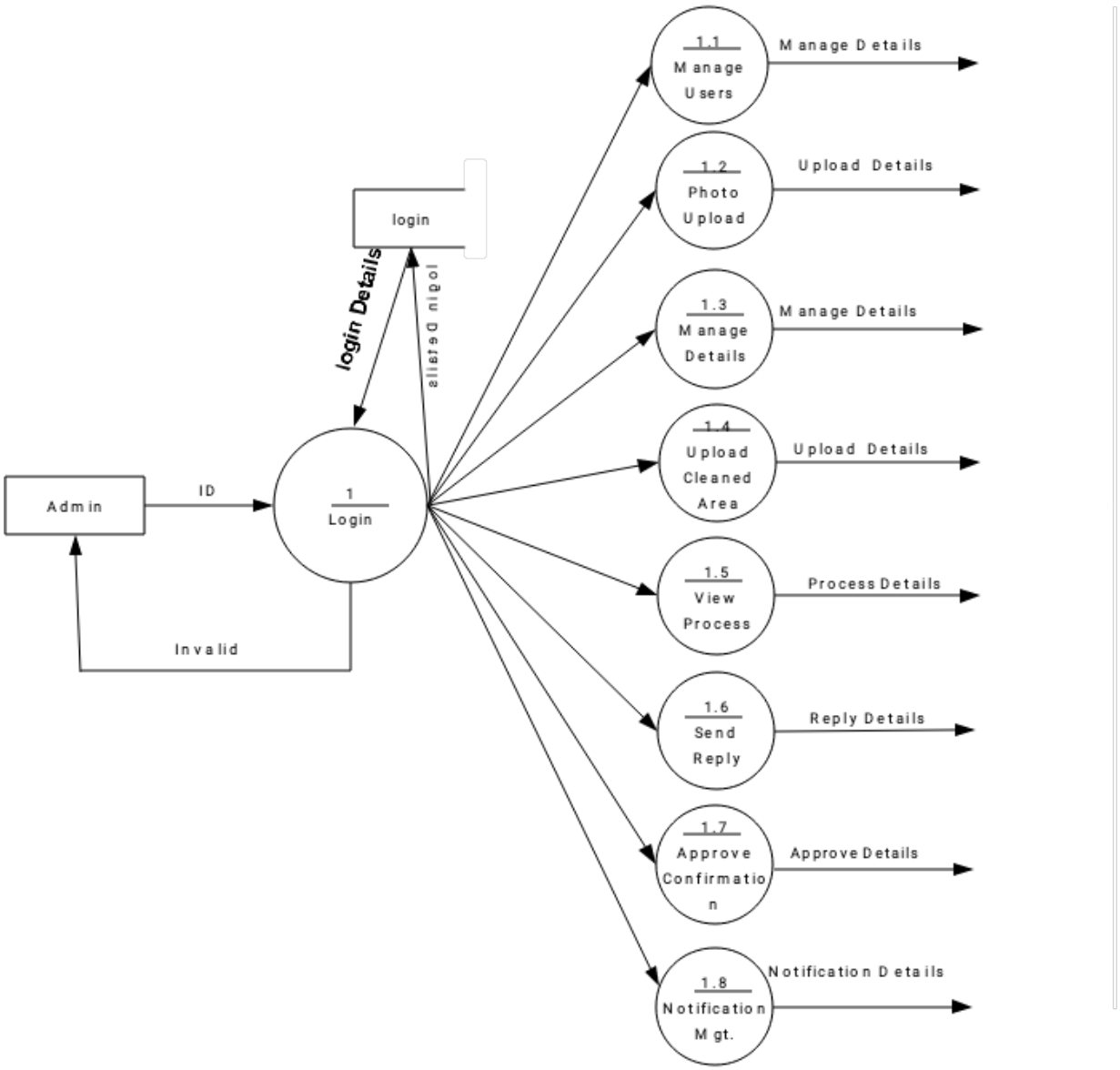
Processes



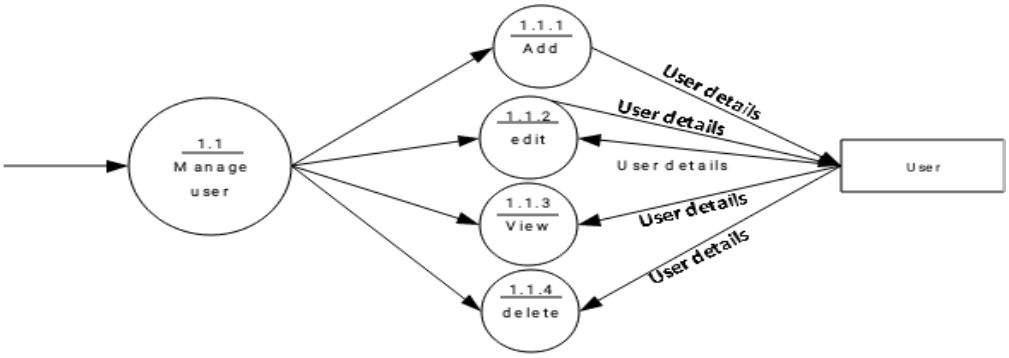
The process specifies the transformation that is applied to the input to produce output. All operations are performed here. The process converts data into information.

Level 0: Context Level

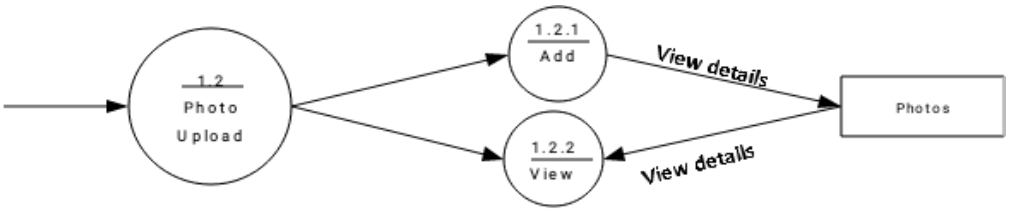
Level 1: Admin



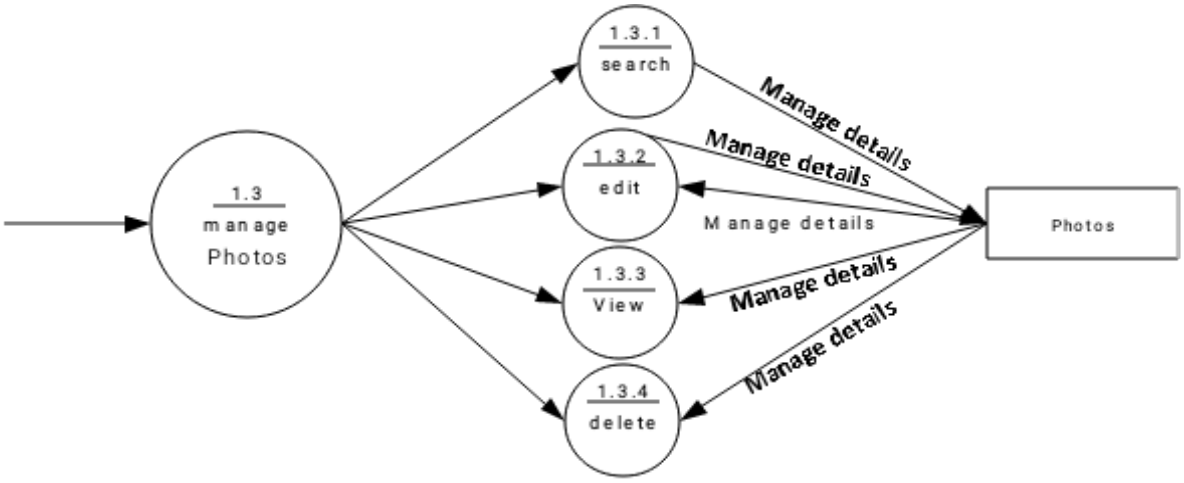
Level 1.1: Admin



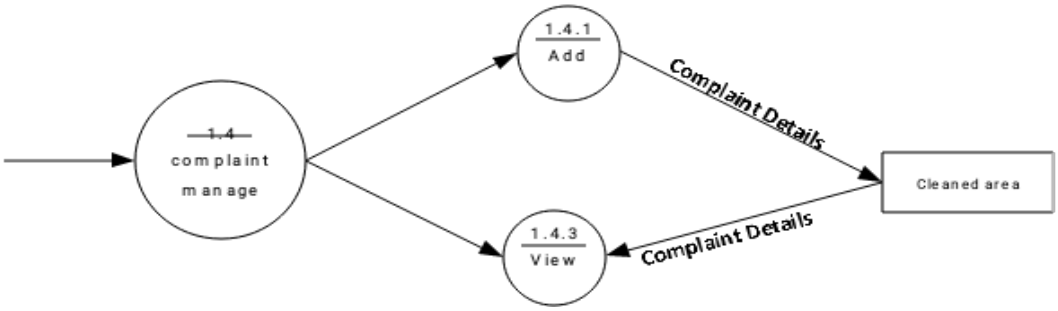
Level 1.2: Admin



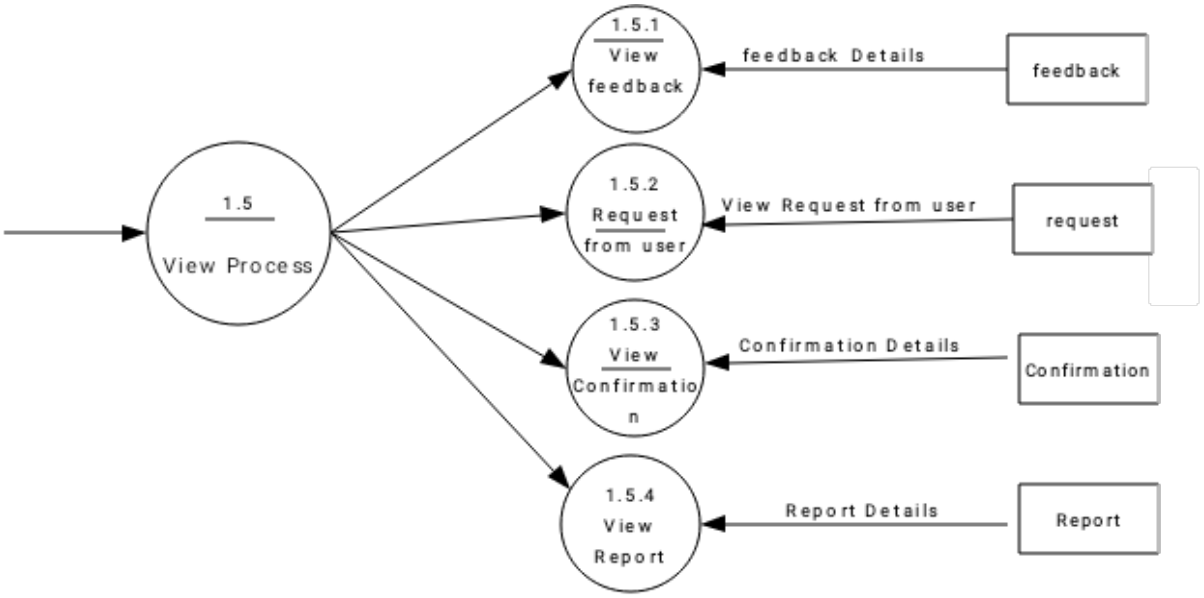
Level 1.3 : Admin



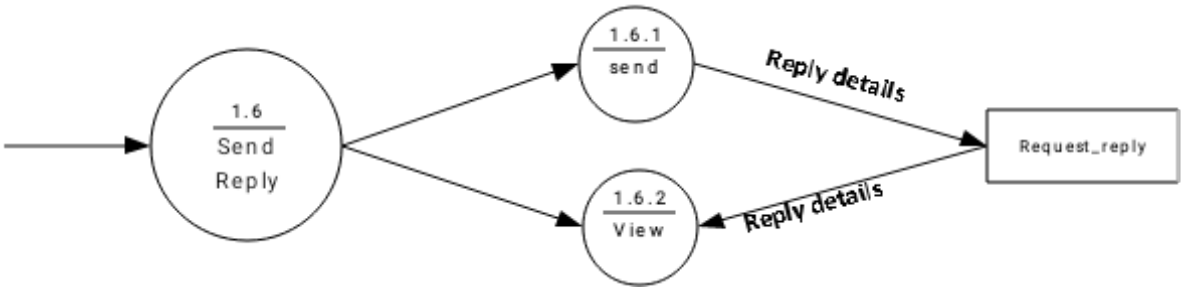
Level 1.4 : Admin



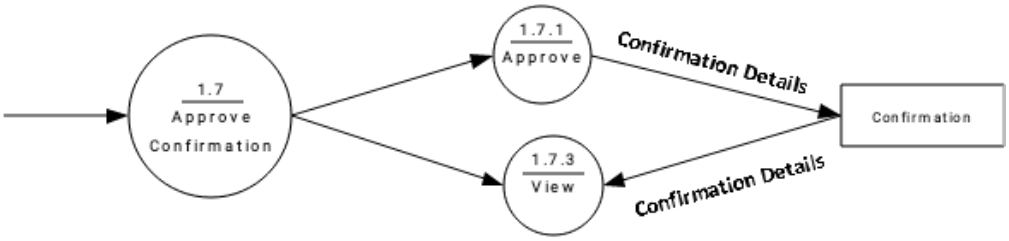
Level 1.5: Admin



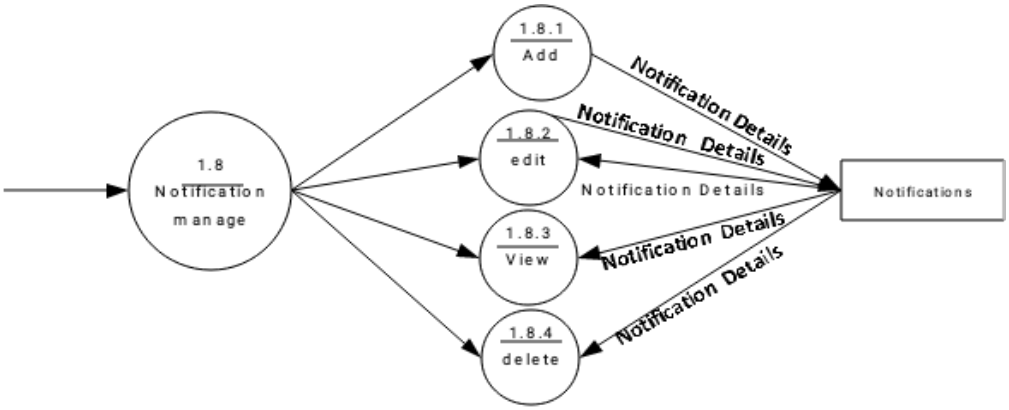
Level 1.6: Admin



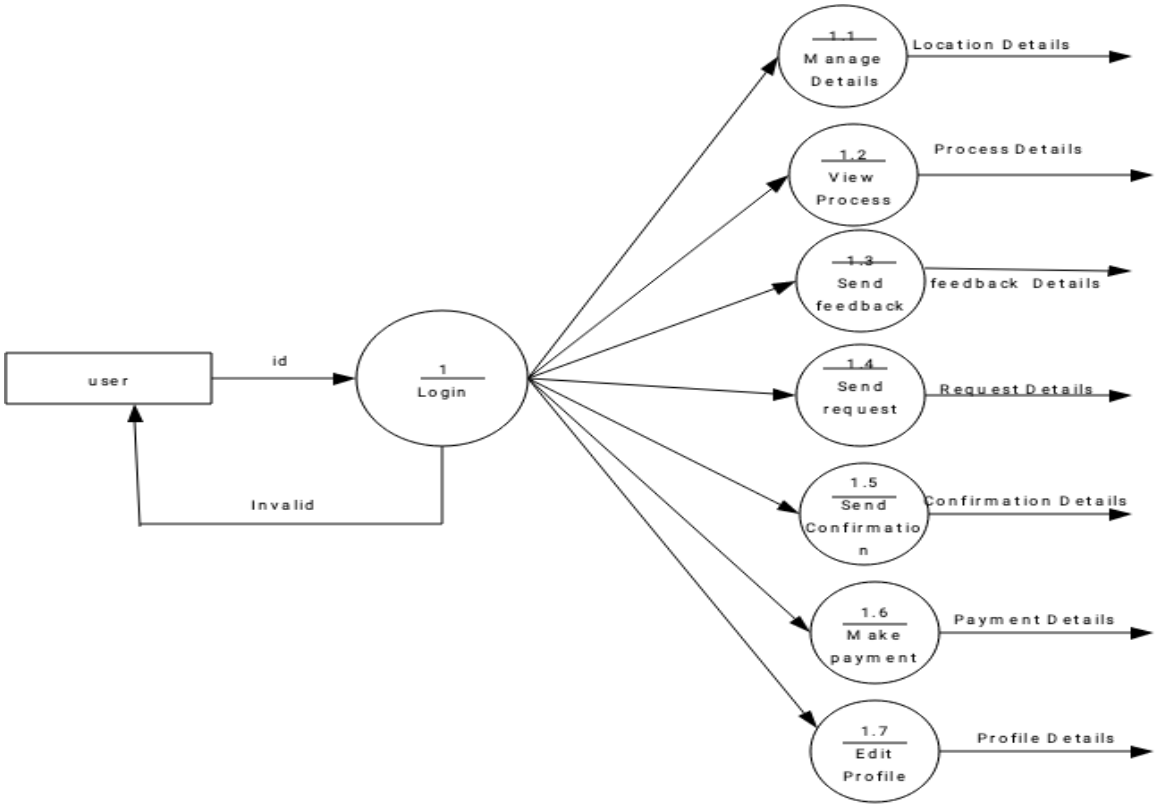
Level 1.7: Admin



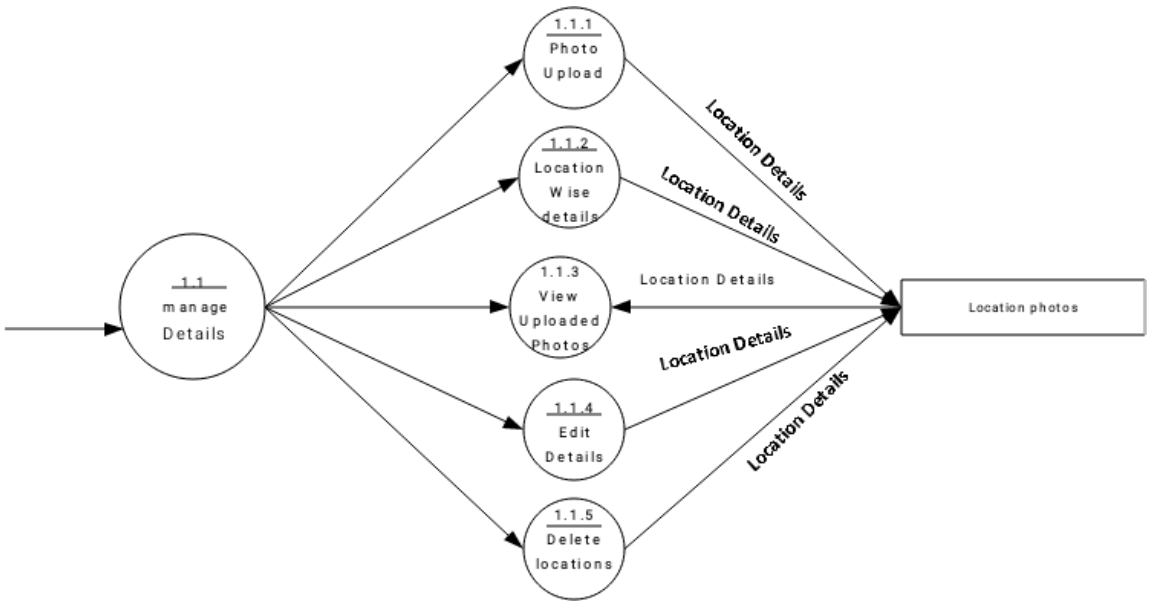
Level 1.8: Admin



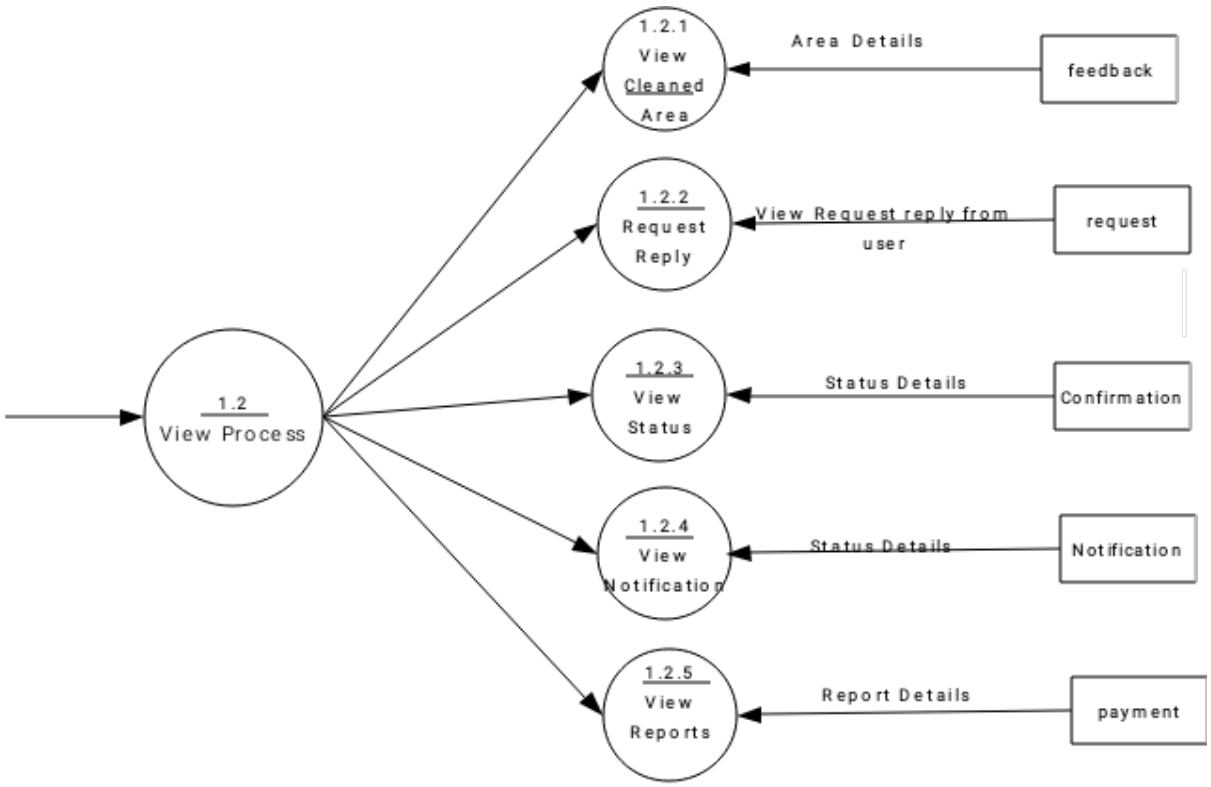
Level 1: User



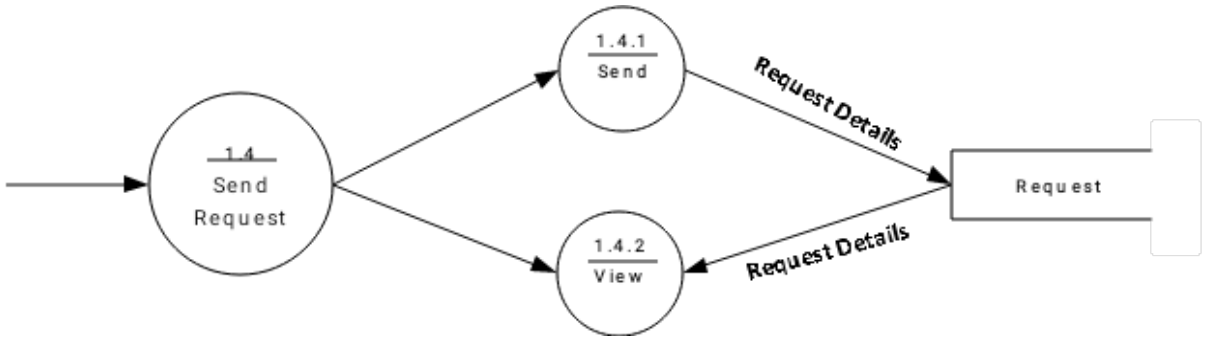
Level 1.1: User



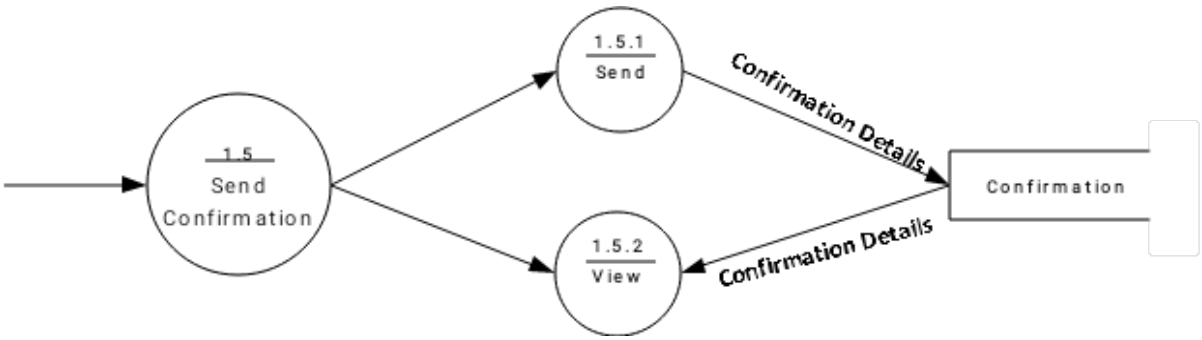
Level 1.2: User



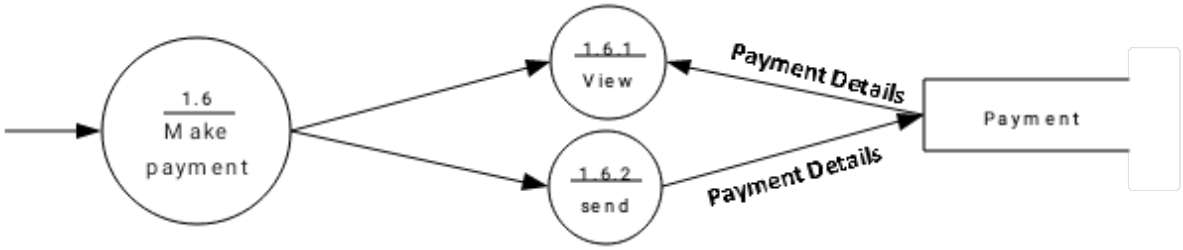
Level 1.4 : User



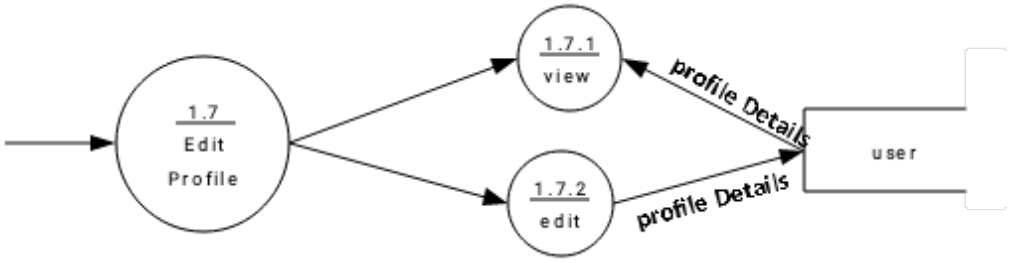
Level 1.5: User



Level 1.6: User



Level 1.7: User



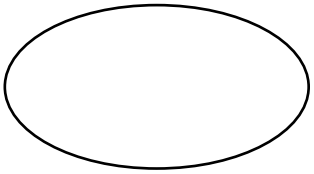
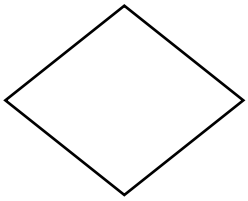
2.3.6 ENTITY RELATIONSHIP DIAGRAM (E-R DIAGRAM)

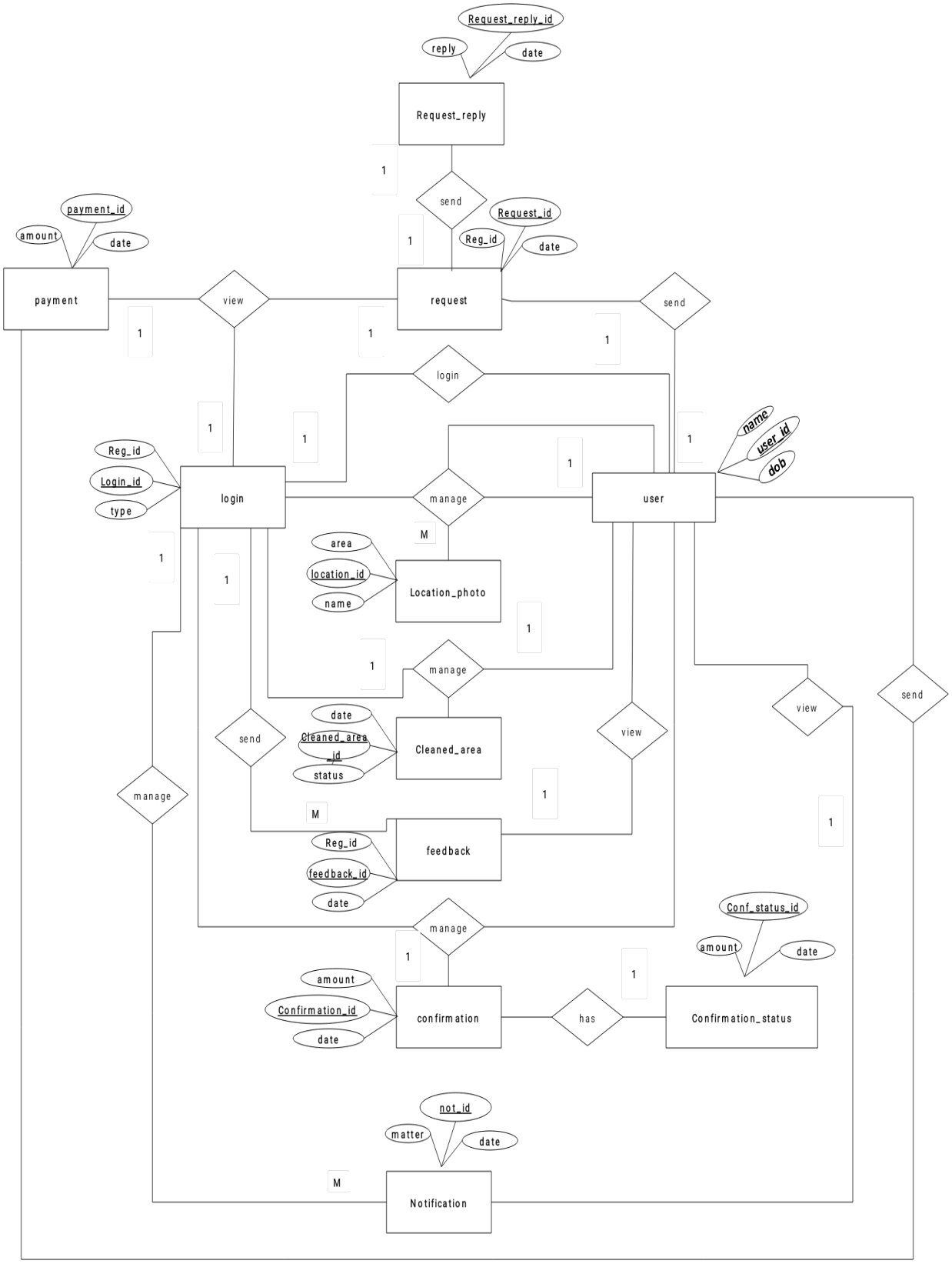
An Entity Relationship diagram (ER) is a data modelling technique that graphically illustrate the information system's entities and the relationships between hose entities. An ER diagram is a conceptual and representational model of data used to represent the entity framework infrastructure.

The Elements of ER Diagram are:

- Entities
- Relationships
- Attributes

Basic symbols used in ER-Diagram are:

NAME	SYMBOLS	USE IN ER
RECTANGLE		Represents Entity sets
OVAL		Represents attributes
DIAMONDS		Represents relationships
LINES		Link attributes to entity sets and vise versa



2.3.7 STRUCTURE CHART

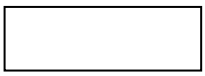
A structure chart is a chart which shows the breakdown of a system to its lowest manageable levels. It is a design tool that pictorially shows the relationship between processing modules in computer software. Describes the hierarchy of components modules and the data are transmitted between them. This chart is used in structured programming to arrange the program modules in a tree structure. Each module is represented by a box, which contains the module name. The tree structure visualizes the relationships between the modules.

A structure chart is a top-down modular design tool, constructed of square representing the different modules of the system, and lines that connect them. The lines represent the connection or ownership between activities and sub activities as they are used in organizational charts.

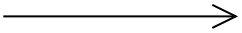
A structured chart depicts

- The size and complexity of the system.
- The number of readily identifiable function and modules within each function.
- Whether each identifiable function is a manageable entity or should be broken down into smaller components.

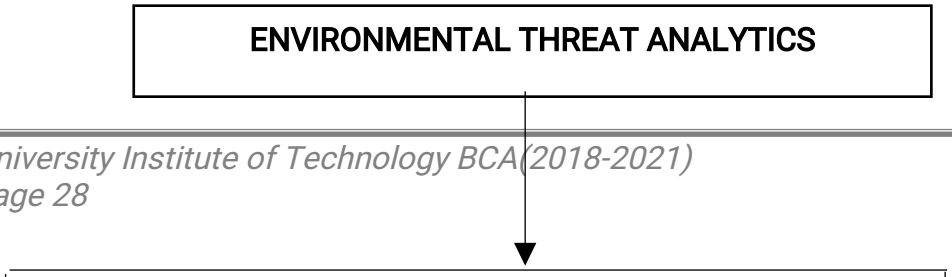
Symbols used in structure chart are:

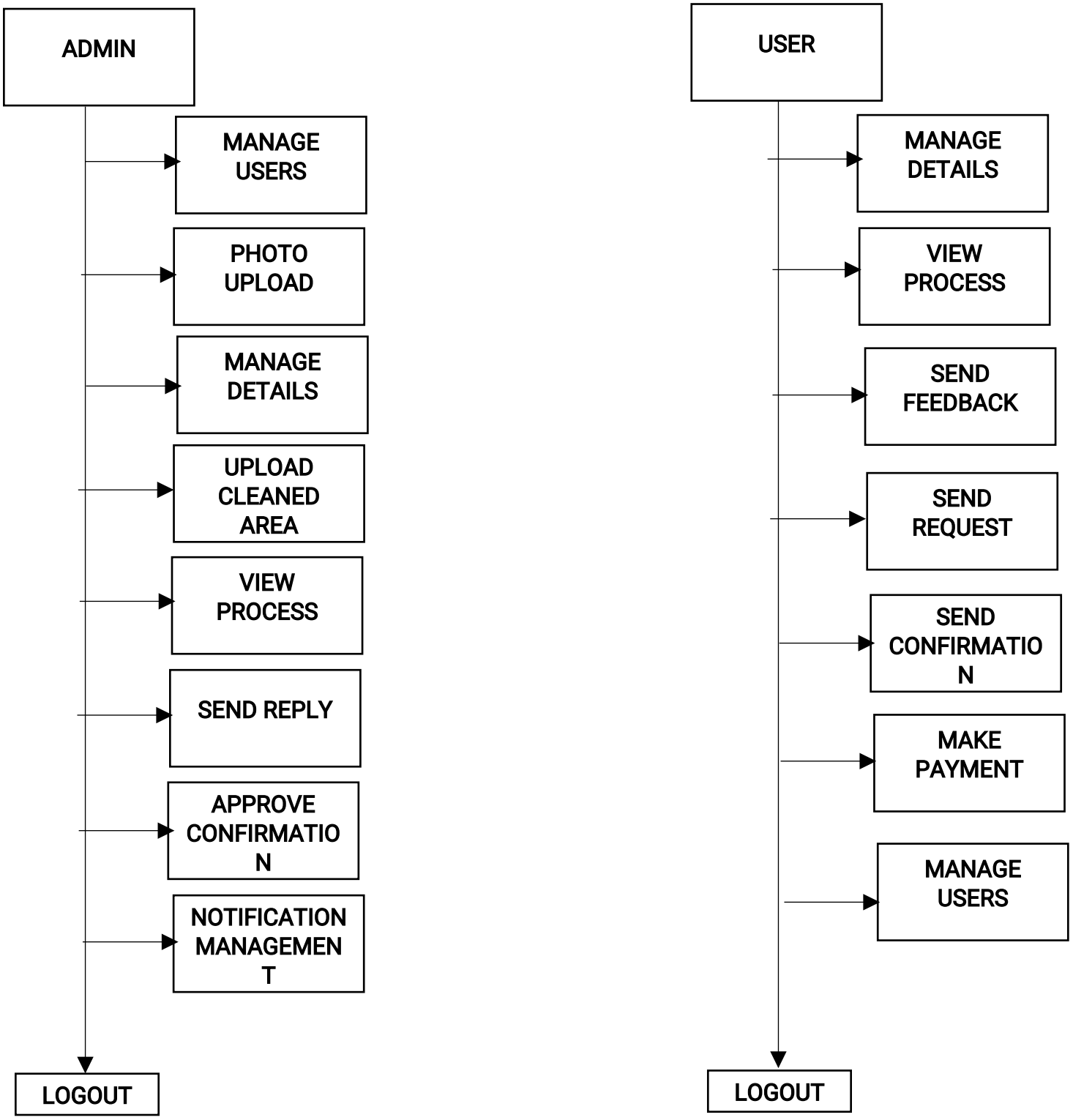


A rectangle represents the module



Arrow represents the connection

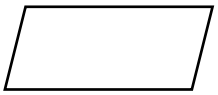
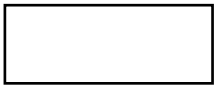
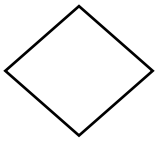
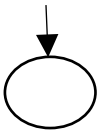




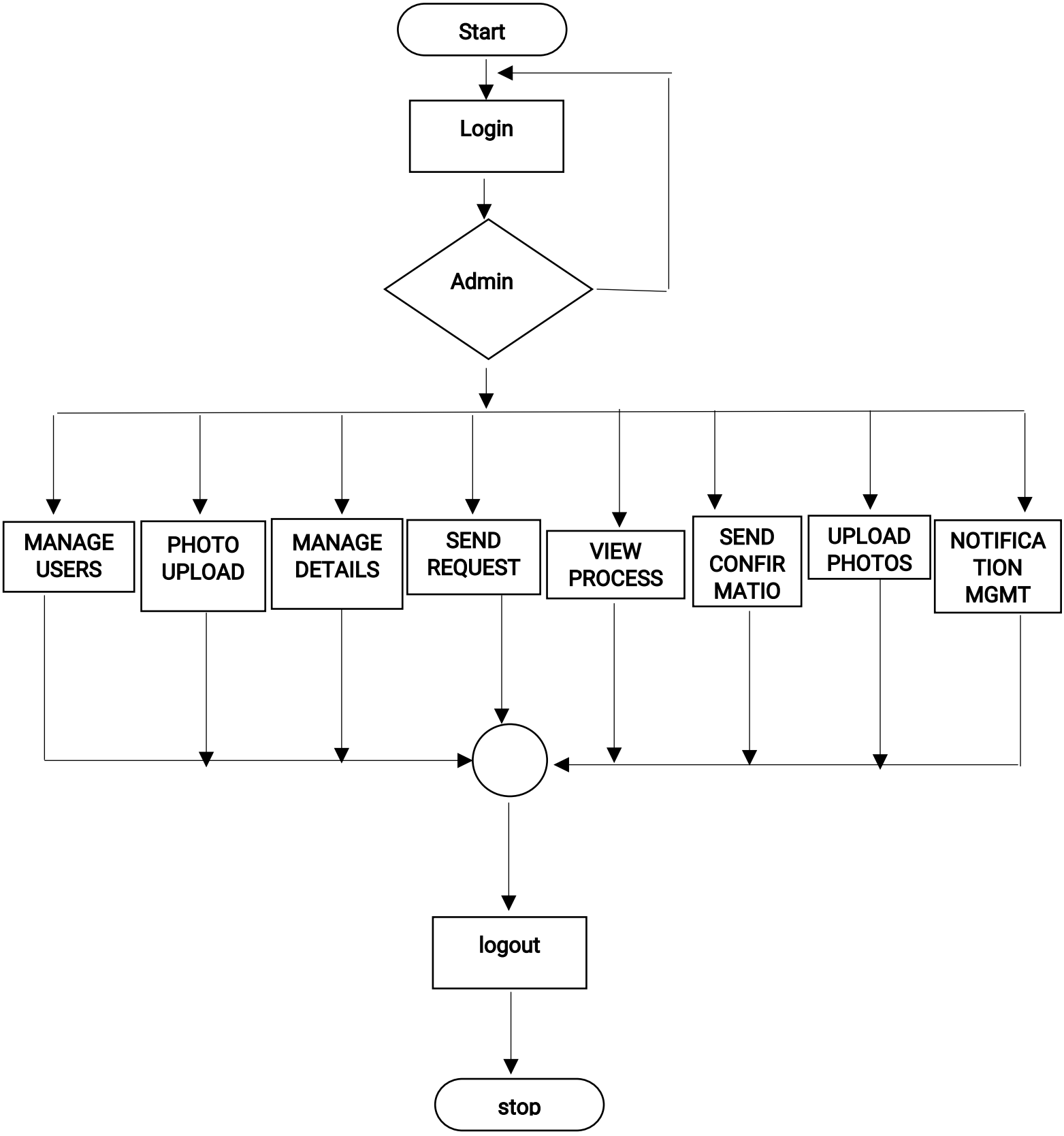
2.3.8 FLOW CHART

A flow chart is a graphical or symbolic representation of a process. Each step in the process is represented by a different symbol and contain a short description of the process step. The flow chart symbols are linked together with arrows showing the process flow direction. The flow chart is a means of visually presenting the flow of data through an information processing systems, the operations performed within the system and the sequence in which they are performed.

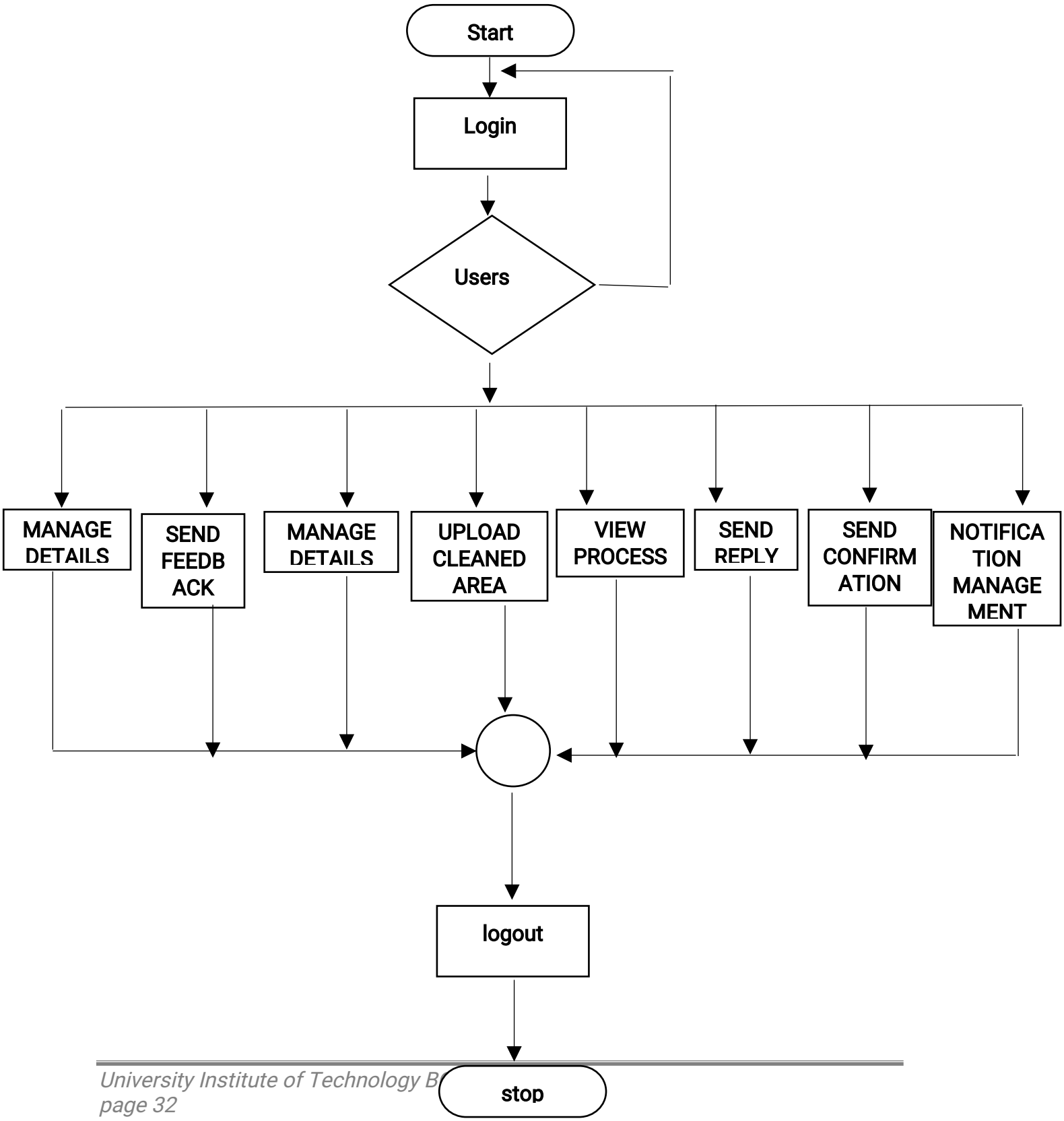
The symbols uses in flow charts are:

Name	Symbols	Use in flow chart
Rounded Rectangle		Denotes the start and end of a program.
Flow Line		Denotes the direction of logic flow in a program.
Parallelogram		Denotes either an input operation or an output operation.
Rectangle		Denotes a process to be carried out.
Diamond		Denotes a decision to be made
Circle		Denotes the connector

Flow Chart : Admin



Flow Chart: Users



SYSTEM DESIGN

3 SYSTEM DESIGN

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. Description: A systemic approach is required for a coherent and well-running system.

3.1 Database Design

A relational database is a collection of data items organized as a set of formally – described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database tables. The relational database was invented by E. F. Codd at IBM in 1970.

A relational database management system (RDBMS) is a program that lets you create, update, and administer a relational database. Most commercial RDBMS's use the Structured Query Language (SQL) to access the database, although SQL was invented after the development of the relational model and is not necessary for its use.

A database is an organized mechanism that has the capability of storing information through which a user can retrieve stored information in an effective and efficient manner. The data is the purpose of any database and must be protected.

The database design is a two-level process. In the first step user requirements are gathered together and a database is designed which will meet these requirements as cleanly as possible. This step is called Information Level Design and it is taken independent of any individual DBMS.

In the second step, this information level design is transferred into a design for the specific DBMS that will be used to implement the system in question. This step is called Physical Level Design, concerned with the characteristics of the specific DBMS that will be used. A database design runs parallel with the system design. The organization of the data in the database is aimed to achieve the following two major objectives.

- Data Integrity

- Data Independence

TABLES

Table : login

Description : used to store login details

Primary Key : login_id

Field Name	Data type	Constraint	Description
login_id	int(10)	Primary key	loginid for identify each records in logged in users
reg_id	int(10)	Not null	reg_id for identify each records in registered users
Username	varchar(20)	Not null	Username credential for login
Password	varchar(15)	Not null	Password credential for login
Type	varchar(10)	Not null	Provide common login interface for two types of users

Table : user

Description : used to store user details

Primary Key : user_id

Field Name	Data type	Constraint	Description
user_id	int(10)	Primary key	Uniquely identify Each users in users table
Name	Varchar(20)	Not null	Name of the user registered
Address	varchar(50)	Not null	Address of the user registered
gender	Varchar(6)	Not null	Gender of the user
Dob	date	date	DOB of the user
Adhaar_no	Varchar(20)	Not null	UID Aadhaar no ensure the

			nationality
mobile	varchar(12)	Not null	Contact number

Table : Location photos

Description : Used to details photos location

Primary Key : location_id

Field Name	Data type	Constraint	Description
loacation_id	int(10)	Primary key	Unique loction_id For every location in which the user is requesting
Name	Varchar(20)	Not null	Name attribute to identify the location in which the request has been initiated
Area	Varchar(25)	Not null	Specifies in which area under the given location specified

Latitude	Varchar(20)	Notnull	Stores latitude(x) geographical coordinate for GPS Using Gmap API
Longitude	Varchar(20)	Notnull	Stores longitude(x) geographical coordinate for GPS Using Gmap API
Photo	Varchar(20)	Notnull	It holds the path location in which the photo is saved
Date	Date	Date	Date at which the request has been initiated
reg_id	int(10)	Not null	referential attribute has the reference of all registered users
Type	varchar(10)	Not null	Type of the user

Table : Feedback

Description : used to store feedback details

Primary Key :feedback_id

Field Name	Data type	Constraint	Description
feedback_id	int(10)	Primary Key	Uniquely identify each feedbacks from users

reg_id	int (10)	Not null	Identify the user by details
Feedback	Varchar(25)	Not Null	Feedback content of user

Table : Cleaned Area

Description : used to cleaned area details

Primary Key : cleanedarea_id

Field Name	Data type	Constraint	Description
Cleanedarea_id	int(10)	Primary Key	Identifies each cleaned area
loacation_id	int(10)	Foreign key	To find all the locations uploaded
Image	Varchar(25)	Not null	Image path of cleaned area
Date	Date	Date	Date at which cleaned area was uploaded
Status	Varchar(25)	Not Null	Current status
Description	Varchar(50)	Not Null	Description by the concerned authority after they've done that

Table : Request

Description : used to store notification details

Primary Key : request_id

Field Name	Data type	Constraint	Description
request_id	int(10)	Primary Key	Identifies each request by an authenticated user
reg_id	int (10)	Not null	Identify user details on every request
request	Varchar(25)	Not null	It accepts a request showing the demand of collection
Date	Date	Date	Date of that request sent

Table : Request reply

Description : used to store notification details

Primary Key : request_reply_id

Field Name	Data type	Constraint	Description
request_reply_id	int(10)	Primary Key	Identifies each request replies based on the user
request_id	int (10)	Not null	Identifies user by the users request
Reply	Varchar(25)	Not null	Reply to the user who has been requested
Date	Date	Date	Date of the request reply

Table : Confirmation

Description : used to store notification details

Primary Key : confirmation_id

Field Name	Data type	Constraint	Description
confirmation_id	int(10)	Primary Key	Identifies each confirmation for each users registered
Request_reply_id	int(10)	Foreign Key	Identifies user request reply for every confirmation
Conf-date	Date	Date	The date on which the authority would only be available
Available date	Date	Date	The date on which the user will be available

Table : Confirmation-status

Description : used to store notification details

Primary key : confirmation_status_id

Field Name	Data type	Constraint	Description
Confirmation_status_id	int(10)	Primary Key	Identifies each confir mation status

Confirmation_id	int(10)	Foreign Key	Identifies each confirm Mation details
amount	Decimal(18,2)	Not null	Payable amount if the user hasn't been available on given date
Status	Varchar(25)	Not null	Shows the status of the final confirmation
Date	Date	Date	Date of the confirmation status

Table : Payment

Description : used to store payment details

Primary Key : payment_id

Field name	Data type	Constraint	Description
payment_id	int (10)	Primary key	It uniquely identifies each users payment transaction
Confirmation_status_id	int(10)	Foreign Key	Proceed only when the confirmation has been made successfully

amount	Decimal(18,2)	Not null	Payable amount for their service
date	date	date	Date of the payment transaction

Table : card

Description : used to store card details

Primary Key : card_id

Field name	Data type	Constraint	Description
card_id	int(10)	Primary key	Uniquely identifies each credit card transactions
Card_no	int(10)	Not null	Card number
Card_holder_name	int(20)	Not null	Card holder name
cvv_code	int(10)	Not null	Dummy values to compare with.
Exp_date	date	date	Expiration date

Table : Notification

Description : used to details photos location

Primary Key : not_id

Field Name	Data type	Constraint	Description
not_id	int(10)	Primary key	Unique id of each notifications to the users
Matter	Varchar(25)	Not null	Content of the notification
date	date	date	Date of the notification

3.2 ARCHITECTURAL DESIGN

Architectural design is concerned with understanding how a system should be organized and designing the overall structure of that system. The output of the architectural design process is an architectural model that describes how the system is organized as a set of communicating components.

MENU TREE

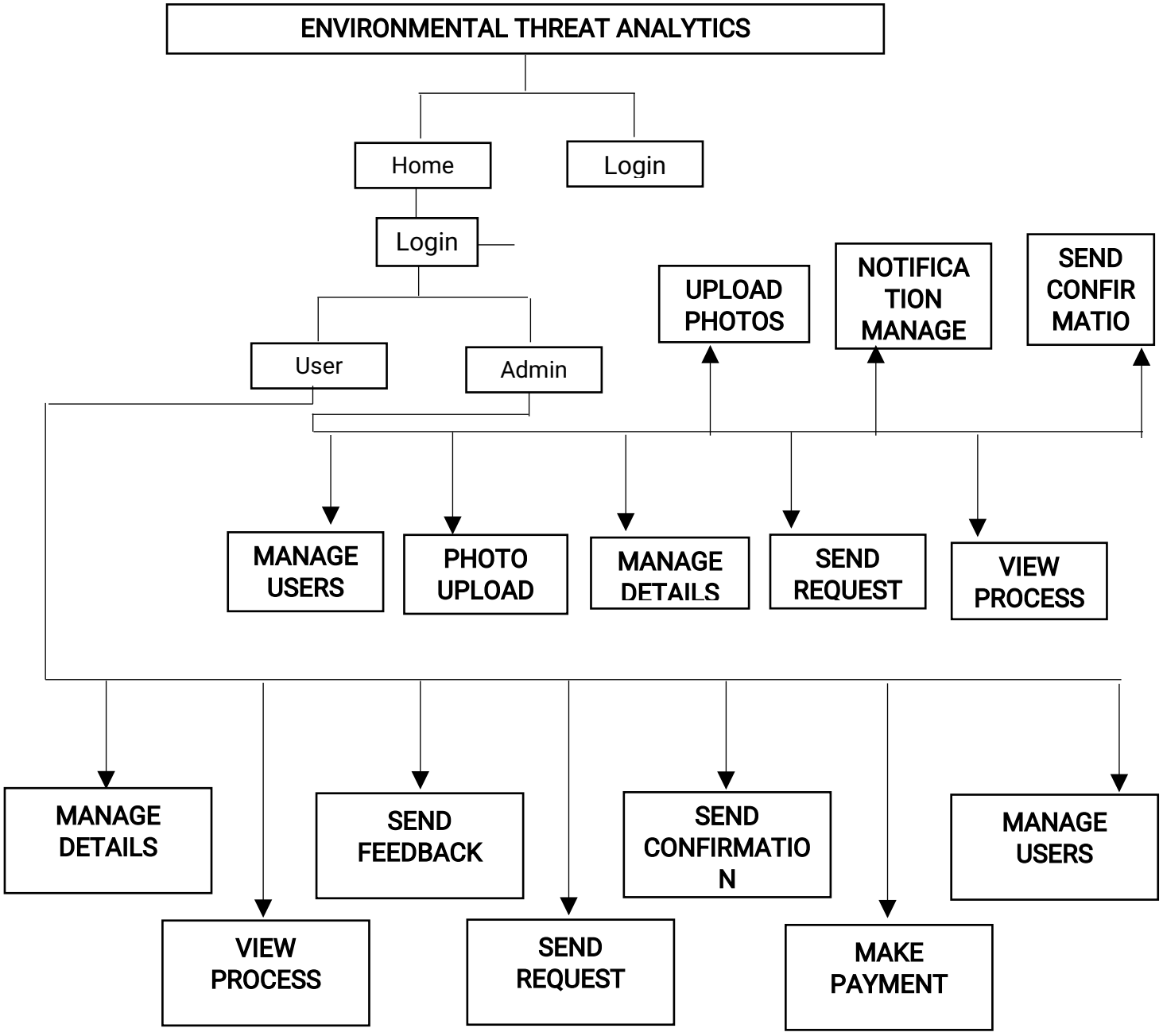
Menu Tree is also helpful for representing the simplified version of the system, which is in the form of a tree structure. The entire system taken as ate main part, and then each modules coming in the system can be represented as branches and their functions of each modules can be represented as same as the leaf in the tree. So menu tree is the hierarchical representation of the entire system, so it is very helpful for identifying the system easily.

Symbols used in Menu Tree:

A rectangle represents the menus

Logout

A line represents the connection



3.3 PROCEDURAL DESIGN

Admin

In the Admin module, admin has the right to log into the website. The main functions carried out by the admin are manage users, manage images of the locations, Upload cleaned area, view feedback, view request from the user, Send request reply, view confirmation, Update confirmation status with payment, view payments, ,manage notification.

- Login
- Manage users
- Manage locations,
- Upload cleaned area
- view feedback
- view waste disposal request
- Send request reply
- View confirmation
- Update confirmation status with payment
- View payments,
- Manage notification.

User

In this module, User has the power to login to the website. The main functions carried out by the user includes Photo upload, Search location wise details, View uploaded photos, Edit details, Delete locations, View cleaned area, Send feedback, Send request, View request reply, Send confirmation, View status, Make payment, View notification, View reports, Edit profile.

- Login
- Photo upload
- Search location wise details
- View uploaded photos
- Edit details
- Delete locations
- View cleaned area
- Send feedback
- Send request
- View request reply
- Send confirmation
- View status
- Make payment
- View notification
- View reports
- Edit profile

3.4 INTERFACE DESIGN

In this phase, the goals and intentions of each user were established. The object and actions were isolated to create a list of objects and actions. The source objects, the target object and the impact of each action were identified. Then the screen layout was performed. The designs of web pages were carried out in the screen layout process.

The interface prototype is made from initial interface design. Once the interface prototype is established this is evaluated by the user and designer. Further modifications are made based on the suggestions.

3.5 INPUT DESIGN

Controlling Amount of Input: Whenever user input is required, giving possible input values as default in the area reduces the number of user keystrokes. Thus the user can pass on the next data without much typing. This makes the data entry much fast and error-free. When the user to give the input in the same format.

- Avoiding Delay:

  A processing delay resulting from data operations or data entry operations is called a bottleneck. Such bottlenecks are made obsolete in this project by breaking up the amount of data to be entered in each form into different smaller and simpler forms.

- Avoiding errors in data:

  The rate at which errors occur depends on the quantity of data. As told in the above objective these errors are reduced by making the number of data to be entered in each form is reduced.

- Avoid extra steps:

  To fulfill any operation the user hence no needs to do complex steps, instead, the operation can be done simple easy to do steps.

- Keeping the process simple:

  This implies that that system has all measure to keep the errors out even if the user is giving wrong data. It handles the situation with grace and does not create many types about the situation to the user.

3.6 OUTPUT DESIGN

Output design is the most important to the user. Efficient, intelligible output design improves the system relationship with the user and helps the decision making. Computer outputs are the most important and direct source to the user. An

efficient output system improves the interaction of the system with the user and it provides his/her required information. The output can be displayed on the screen or copied. In our system, hard copies are preferred because of a document for further references. Careful considerations have been given while developing the output reports as if helps in decision making. Other than the remote system desktops visuals, received messages from other systems, etc. are also outputted.

Output Objectives

The output form information system should accomplish the following objectives:

- Convey information about past activities, current status or projection of the future.
- Signal important events, opportunities, problems or warning.
- Trigger an action
- Confirm an action
- Output design of the project is made with objectives in mind.
- External outputs, whose destination is within the organizations and is the main image of the organization.
- Internal outputs, whose destination is within the organization and which require careful design because it is the user's main interface with the computer.
- Operational outputs, whose use is purely within the computer departments.
- Interactive outputs, which the user in communicating directly with the computer.

SYSTEM CODING

4 SYSTEM CODING

```
<!DOCTYPE html>

<html lang="en">

<head>

<title>City clean |</title>

<link rel="icon"

type="image/png"

href="images/log.jpg">
```

```
<meta charset="utf-8">

<link rel="stylesheet" href="css/reset.css" type="text/css" media="screen">

<link rel="stylesheet" href="css/style.css" type="text/css" media="screen">

<link rel="stylesheet" href="css/grid.css" type="text/css" media="screen">

<script src="js/jquery-1.6.3.min.js" type="text/javascript"></script>

<script src="js/cufon-yui.js" type="text/javascript"></script>

<script src="js/cufon-replace.js" type="text/javascript"></script>

<script src="js/Kozuka_Gothic_Pro_OpenType_300.font.js"
type="text/javascript"></script>

<script src="js/Kozuka_Gothic_Pro_OpenType_500.font.js"
type="text/javascript"></script>

<script src="js/FF-cash.js" type="text/javascript"></script>

<script type="text/javascript" src="js/jquery.easing.1.3.js"></script>

<script type="text/javascript" src="js/tms-0.3.js"></script>

<script type="text/javascript" src="js/tms_presets.js"></script>

<script src="js/jcarousellite_1.0.1.js" type="text/javascript"></script>

<script type="text/javascript">

$(document).ready(function () {

$('.carousel .jCarouselLite').jCarouselLite({

btnNext: '.carousel .next',

btnPrev: '.carousel .prev',

speed: 600,

easing: 'easeOutQuart',

vertical: false,

circular: false,
```

```
visible: 4,

start: 0,

scroll: 1

});

});

</script>

<!--[if lt IE 9]>

<script type="text/javascript" src="js/html5.js"></script>

<link rel="stylesheet" href="css/ie.css" type="text/css" media="screen">

<![endif]-->

</head>

<body id="page1">

<!--
==============================header=================================
=-->

<header>

<div class="menu-row">


<div class="main">

<h1 class="main_head"><a href="index.html">Exterior Design</a></h1>

<nav class="menu_wrapper">

<ul class="menu">

<li><a class="active" href="index.php">Home</a></li>

<li><a href="about.html">About us</a></li>

<li><a href="services.html">Services</a></li>
```

```
<li><a href="portfolio.html">Portfolio</a></li>

<li><a href="user_registration.php">Registration</a></li>

<li><a href="login.php">Login        </a></li>

</ul>

</nav>

</div>

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

<div class="row-bot-bg">

<div class="main">

<div class="slider-wrapper">

<div class="slider">

<ul class="items">

<li><imgsrc="images/slider-img1.jpg" alt=""></li>

<li><imgsrc="images/slider-img2.jpg" alt=""></li>

<li><imgsrc="images/slider-img3.jpg" alt=""></li>

</ul>

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

</header>


<!--
==============================content================================
=-->

<section id="content">

<div class="main">

<div class="container_12">
```

```
<div class="wrapper">

<article class="grid_4">

<h3 class="prev-indent-bot">Shortly About Us</h3>

<p class="img-indent-bot">“ENVIRONMENTAL THREAT ANALYSIS”is an online site to upload the image of polluted area by the user and corporation will make the arrangement  to clean the area and return a rectified image to the user. Inefficient municipal solid waste management system may create serious negative environmental impacts like infectious diseases, land and water pollution, obstruction of drains and loss of biodiversity. .Some waste will eventually rot, but not all, and in the process it may smell, or generate methane gas, which is explosive and contributes to the greenhouse effect. Leach ate produced as waste decomposes may cause pollution. Badly-managed landfill sites may attract vermin or cause litter.</p>

<a class="button" href="#"><span>read more</span></a></article>

<!--
==============================footer================================
=-->

<footer>

<div class="main">

<div class="aligncenter">

<p class="p0">Copyright &copy; <a href="#">environmentalthreatanalytics.edu</a> All Rights Reserved</p>

Design by <a target="_blank" href="#">Rijo Mathew Tharakan and Team</a></div>

</div>
```

```
</footer>
<script type="text/javascript">Cufon.now();</script>
<script type="text/javascript">
$(window).load(function () {
$('.slider')._TMS({
duration: 800,
easing: 'easeOutQuad',
preset: 'diagonalExpand',
slideshow: 5000
})
})
</script>
</body>
</html>
```

```
<?php
include_once("user_header.php");
include_once("lib/init.php");
include_once("function.php");
?>
<!--
==========content==========
-->
```

```
<section id="content">
<div class="main">
<div class="container_12">
<div class="wrapper">
<?phpinclude_once("user_sidebar.php"); ?>
<article class="grid_9">
<h3>Request for waste disposal</h3>
<div class="wrapper indent-bot ">
<div>
<form class="frm_container" action="#" method="POST">
<table><tr><td>
Request
</td>
<td><textarea name="req" cols="8" rows="9" ></textarea></td>
</tr><tr><td></td>
<td><button type="reset" name="reset"  class="btn_frm">Clear</button>
<button type="submit" name="btn_save"  class="btn_frm">Send</button></td>
</tr>
</table>
</form></div></div>
</article></div></div></div></section>
<!--
==============================footer================================
=-->
<?phpinclude_once("footer.php"); ?>
```

```
<?php
include_once("user_header.php");
include_once("lib/init.php");
include_once("function.php");
$reg_id = $_SESSION['reg_id'];
?>
<!--
==============================content================================
=-->
<section id="content">
<div class="main">
<div class="container_12">
<div class="wrapper">
<?phpinclude_once("user_sidebar.php"); ?>
<article class="grid_9">
<h3>Request Replies</h3>
<div>
<div class="wrapper indent-bot ">
<table id="dis_table">
<tr>
<th>Request</th>
<th>Req_date</th>
<th>Reply</th>
<th>Available Date</th>
<th>Amount</th>
```

```
<th></th>

</tr>

<?php $result = getCfmStatus($reg_id);

while($row = mysql_fetch_array($result)){

?>

<tr>

<td><?php echo $row['request']; ?></td>

<td><?php echo $row['date']; ?></td>

<td><?php echo $row['reply']; ?></td>

<td><?php echo $row['conf_date']; ?></td>

<td><?php echo $row['amount']; ?></td>

<td><a href="user_payment.php?id=<?php echo $row['conf_status_id'];
 ?>&&amt=<?php echo $row['amount']; ?>">Send Payment</a></td>

</tr>

<?php } ?></table></form></div>

</div>

<?php

Function getUsers(){

$qry=”select * from tbl_user”;

$result=mysql_query($qry);

Return $result;

}

function getFeedbacks(){

$qry=”select  fed.feedback_id,fed.reg_id,fed.feedback,fed.date,usr.name,usr.address
from tbl_feedback fed inner join tbl_usr ur on reg_id=usr.usr_id”;
```

```
$result=mysql_query($qry);

Return $result;

}

function getNotification(){

$qry= “select * from tbl_notification”;

$result=mysql_query($qry);

Return $result;

}

function getReqReplies($reg_id){

$sql=”select rply.reply,rply.date as rply_date

,req.request,req.date,req.req_id.rply.req_reply_id from tbl req_request_reply rply

Inner join tbl_request req on rply.req_id=req.req_id where req.reg_id=$reg_id”;

$result=mysql_query($sql);

return $result;

}


function getUserById($user_id){

$qry=”select * from tbl_user where user_id=$user_id”;

$result=mysql_query($qry);

return $result;

}

Function getRequests(){

$qry=”select req.requests,

req.date,

usr.name,
```

```
usr.address,
req.req_id
rply.req_rply_id,
st.amount,
st.date,
st.conf_status_id,
from table confirmation_status st inner join tbl_confirmation cfm on
st.confirmation_id=cfm.confirmation_id
inner join tbl_request_rply rply on
cfm.request_reply_id=rply.req_id
inner join tbl_request req on rply.req_id=req.req_id
where req.req_id = $reg_id";
$result=mysql_query($sql);
return $result;
}
```

```
div.nav-controls a.prev {
display:block;
width:39px;
height:39px;
background:url(../images/carousel-prev.png) 0 0 no-repeat;
text-indent:-9999em;
cursor:pointer;
position:absolute;
top:-583px;
```

```
left:0;
z-index:99;
}
div.nav-controls a.next {
display:block;
width:39px;
height:39px;
background:url(../images/carousel-next.png) 0 0 no-repeat;
text-indent:-9999em;
cursor:pointer;
position:absolute;
top:-583px;
right:0;
z-index:99;
}
div.nav-controls a.prev:hover, div.nav-controls a.next:hover {
cursor:pointer;
}
div.slideshow-container {
position: relative;
height: 370px; /* This should be set to be at least the height of the largest image in the slideshow */
z-index:1;
width:940px;
margin:0 auto;
```

```
}
div.slideshow {
}
div.caption {
width:100%;
height:100%;
}
div.slideshow span.image-wrapper {
display: block;
width: 920px;
height: 347px;
background:url(../images/img-border2.jpg) 0 0 no-repeat;
position:absolute;
left:0;
top:0;
margin:0;
padding:11px;
}
div.slideshow a.advance-link {
display: block;
margin: 0;
font-size:0;
line-height:0;
text-decoration:none;
}
```

```
div.slideshow a.advance-link:hover, div.slideshow a.advance-link:active,
div.slideshow a.advance-link:visited {
text-decoration: none;
}
div.download {
float: right;
}
div.caption-container {
float:right;
width: 270px;
height: 620px;
position:relative;
}
span.image-caption {
display: block;
position: absolute;
width: 100%;
height: 100%;
top: 0px;
left: 0;
z-index:10;
background:#fff;
}
div#thumbs {
padding: 0;
```

```
margin:0 auto 111px;
width:940px;
}
ul.thumbs {
padding: 0 0 0 0;
width:100%;
overflow:hidden;
position:relative;
}
ul.thumbs li {
float:left;
margin-right:17px;
margin-bottom:18px;
width:200px;
height:137px;
padding:11px;
background:url(../images/img-border.png) 0 0 no-repeat;
}
ul.thumbs li.last {
margin-right:0;
}
ul.thumbs li.last2 {
margin-bottom:0;
}
ul.thumbs li.last3 {
```

```
margin:0;
}
ul.thumbs li span {
display:block;
font-size:15px;
line-height:1.2em;
color:#f9f9f9;
text-transform:uppercase;
font-weight:bold;
}
a.thumb {
display:block;
}
a.thumb:focus {
outline: none;
}
#controls {
width:100%;
}
div.pagination {
clear: both;
text-align:center;
position:relative;
z-index:10;
}
```

```
div.top.pagination {
display:none;
}
div.navigation div.bottom {
display:none;
}
div.pagination a, div.pagination span.current, div.pagination span.ellipsis {
padding:0 4px;
font-weight:bold;
color:#fff;
}
div.pagination a:hover {
text-decoration: none;
color:#ffeaa8;
}
div.pagination span.current {
color:#ffeaa8;
}
div.pagination span.ellipsis {
border: none;
padding: 5px 0 3px 2px;
}
/* -- gallery end --*/

/***** contact form *****/
```

```
#contact-form {
display:block;
width:100%;
}
#contact-form label {
display:block;
height:40px;
overflow:hidden;
margin-top: 23px;
}
#contact-form input {
float:left;
width:265px;
font-size:12px;
line-height:1.25em;
color:#030303;
padding:8px 10px;
margin:0;
font-family:Arial, Helvetica, sans-serif;
border:1px solid #ebebeb;
background:#fff;
outline:none;
margin-top: 23px;
}
#contact-form textarea {
```

```
float:left;
height:196px;
width:505px;
font-size:12px;
line-height:1.25em;
color:#030303;
padding:5px 10px;
margin:0;
font-family:Arial, Helvetica, sans-serif;
border:1px solid #ebebeb;
background:#fff;
overflow:auto;
outline:none;
}
.text-form {
float:left;
display:block;
font-size:13px;
line-height:32px;
width:93px;
color:#030303;
font-family:Arial, Helvetica, sans-serif;


}
.buttons {
```

```
padding:8px 0 0 0;
text-align:right;
}
.buttons a {
margin-left:6px;
}
/*****************************footer************************/
footer {
width:100%;
padding:10px 0;
background:#9ab059;
}
.list-services {
padding:0;
}
.list-services li {
float:left;
padding:0 0 0 10px;
}
.frm_wrper{
margin-top: 69px;
min-height: 450px;
}
```

SYSTEM TESTING

5. SYSTEM TESTING

5.1 UNIT TESTING

Unit testing focuses verification effort on the smallest unit of software design – the software component or module. Using the component-level design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and uncovered scope established for unit testing. The unit testing is White box oriented, and step can be conducted in parallel for multiple components. The modular interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that all statements in a module have been executed at least once. Finally, all error handling paths are tested. Tests of data flow across a module interface are required before any other test is initiated. If data do not enter and exit properly, all other tests are moot. Selective testing of execution paths is an essential task during the unit test. Good design dictates that error conditions be anticipated and error handling paths set up to reroute or cleanly terminate processing when an error does occur.

Boundary testing is the last task of unit testing step. Software often fails at its boundaries. Unit testing was done in Sell-Soft System by treating each module as a separate entity and testing. Each one of them with a wide spectrum of test inputs. Some flaws in the internal logic of the modules were found and were

rectified. After coding each module is tested and run individually. All unnecessary code was removed and ensured that all modules are working, and gives the expected result.

5.2 INTEGRATION TESTING

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been dictated by design. The entire program is tested as a whole. Correction is difficult because isolation of causes is complicated by a vast expanse of the entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loop.

After performing unit testing in the System all the modules were integrated to test for any inconsistencies in the interfaces. Moreover, differences in program structures were removed and a unique program structure was evolved.

5.3 SYSTEM TESTING

This is the final step in testing. In this, the entire system was tested as a whole with all forms, code, modules and class modules. This form of testing is popularly known as Black Box testing or System tests. Black Box testing method focuses on the functional requirements of the software. That is, Black Box testing enables the software engineer to derive sets of input conditions that will fully exercise all functional requirements for a program. Black Box testing attempts to find errors in the following categories; incorrect or missing functions, interface errors, errors in data structures or external data access, performance errors and initialization errors and termination errors.

BLACK BOX TESTING

Black box testing focuses on the functional requirements of the software and on the input and output of the module. That is, black box testing enables to drive sets of input condition that will fully exercise all functional requirements for a program. Black box testing is not an alternative to white box techniques. Rather, it is a complementary approach that is likely to uncover a different classes of errors that white box testing method. It attempts to find the error in the following categories.

- Error in database structure and database access.
- Incorrect and missing function.
- Performance errors.
- Initialization and termination error.

The following are the two main approaches to design black box test case.

- Equivalence class partitioning.
- Boundary value analysis.

WHITE BOX TESTING

White box testing is also called Glass box testing is test case design method that uses the control structure of the procedural design to drive test case. Using white box testing method, the test can derive following cases:

- Exercise all logical decision on their true and false sides.
- Guarantee that all independent paths within a module have been exercised at least once

SYSTEM IMPLEMENTATION

6. SYSTEM IMPLEMENTATION

6.1 IMPLEMENTATION AND MAINTANENCE

Implementation is the process of converting a new or revised system design into operation. It is the key stage in achieving a successful new system because usually, it reveals a lot of up heal the using department. It must, therefore, be carefully planned and controlled. Apart from planning the two major tasks of preparing for implementation are education and training of users and testing of the system. Training has to be given to the staff regarding the new system. Once the staff has been trained, the system can be tested. Implementation is the stage of the project where the theoretical design is turned into the working system or it is the key stage in achieving a successful new system because usually, it involves a lot of up heal in the user department.

Implementation is the final and important phase. It is the phase where theoretical design is turned into a working system, which works for the user in the

most effective manner. It involves careful planning, investigation of the present system and the constraints involved, user training, system testing and successful running of the developed proposed system. The implementation process begins with preparing a plan for the implementation of the system. According to this plan, the activities are to be carried out, discussions made regarding the equipment and resources and the additional equipment has to be acquired to implement the new system. The user tests the developed system and changes are made according to their needs. The testing phase involves the testing of a system using various kinds of data. This method also offers the greatest security since the old system can take over if the errors are found or inability to handle a certain type of transactions while using the new system.

The tasks involved in the normal implementation process are:

IMPLEMENTATION PLANNING

The implementation of a system involves people from different departments and system analysts are confronted with the practical problems of controlling the activities of people outside their own data processing department prior to this point in the project system, system analyst has interviewed department staff with the permission of their respective managers.

The implementation coordination committee should be responsible for a successful implementation. The composition of the committee is important. There should be at least one representative of each department affected by the changes and other members should be co-opted for discussion of specific topics.

TRAINING

Training sessions must aim to give user staff the specific skills required in their new jobs. The training will be most successful if conducted by the supervisor with the system analyst in attendance to sort out any queries, new methods gain acceptable more quickly in this way.

EDUCATION

Education is complementary to training. Education involves the creation of the right atmosphere and motivating user staff. Education sessions should encourage participation from all staff, with protection for individuals from group

criticism. Educational information can also make training more interesting and understandable.

Maintenance is the enigma of system development. It holds the software industry captive, trying up programming resources. Analysts and programmers spend far more time maintaining resources. As important as it is, many programmers and analysts are reluctant to perform or identify themselves with the maintenance effort. There is psychological, personality, and professional reasons for this.

Whereas the cost of hardware has steadily declined, the cost of producing programs has skyrocketed. That is software maintenance is expensive. One way to reduce maintenance costs is through maintenance management and software modification audits. Software modification consists of program rewrites system-level updates, and re-audits of low-ranking programs to verify and correct the soft spots. The outcome should be more reliable software, reduced maintenance backlog, and higher satisfaction and morale among the maintenance staff.

After the installation, a phase is completed and the user staff is adjusted to the changes created by the candidate system evaluation and maintenance. Like any system, there is an aging process that requires periodic maintenance of hardware & software for customer evaluation, tasks required to obtain customer feedback based on evaluation of the software representations created during the engineering stage and implemented during the installation stage.

6.2 FUTURE ENHANCEMENT

The future enhancement of our project is to reduce the limitations to some extent. The enhancement may be required if there will be any change in the requirements, user environment or priorities. Enhancement means adding, modifying or developing the code to support the changes in specification. Every module in the system is being developed carefully such that the future enhancement does not affect the basic performance of the system. In the future, we can add any links or services to the system very easily. And we would like to deploy high level browser compatible machine learning model for classification of uploaded images into categorical class having their location names and also

another model for recognising the fake uploaded images for misdirecting the flow of our system.

CONCLUSION

7. CONCLUSION

One of the main concerns with our environment has been solid waste management which impacts the health and environment of our society. The detection, monitoring and management of wastes is one of the primary problems of the present era. The traditional way of manually monitoring the wastes in waste bins is a cumbersome process and utilizes more human effort, time and cost which can easily be avoided with our present technologies. This project is very effective in managing waste in municipality or corporation area. This system is used to clean

polluted area by uploading images of that location and they will rectify the complaint and return a rectified image to the person who uploads the image.

This project was successfully completed within the time span allotted. Every effort has been made to present the system in more user-friendly manner. All the activities provides a feeling like an easy walkover the user who is interfacing with the system. A trial run of the system has been made and it showed very good result.

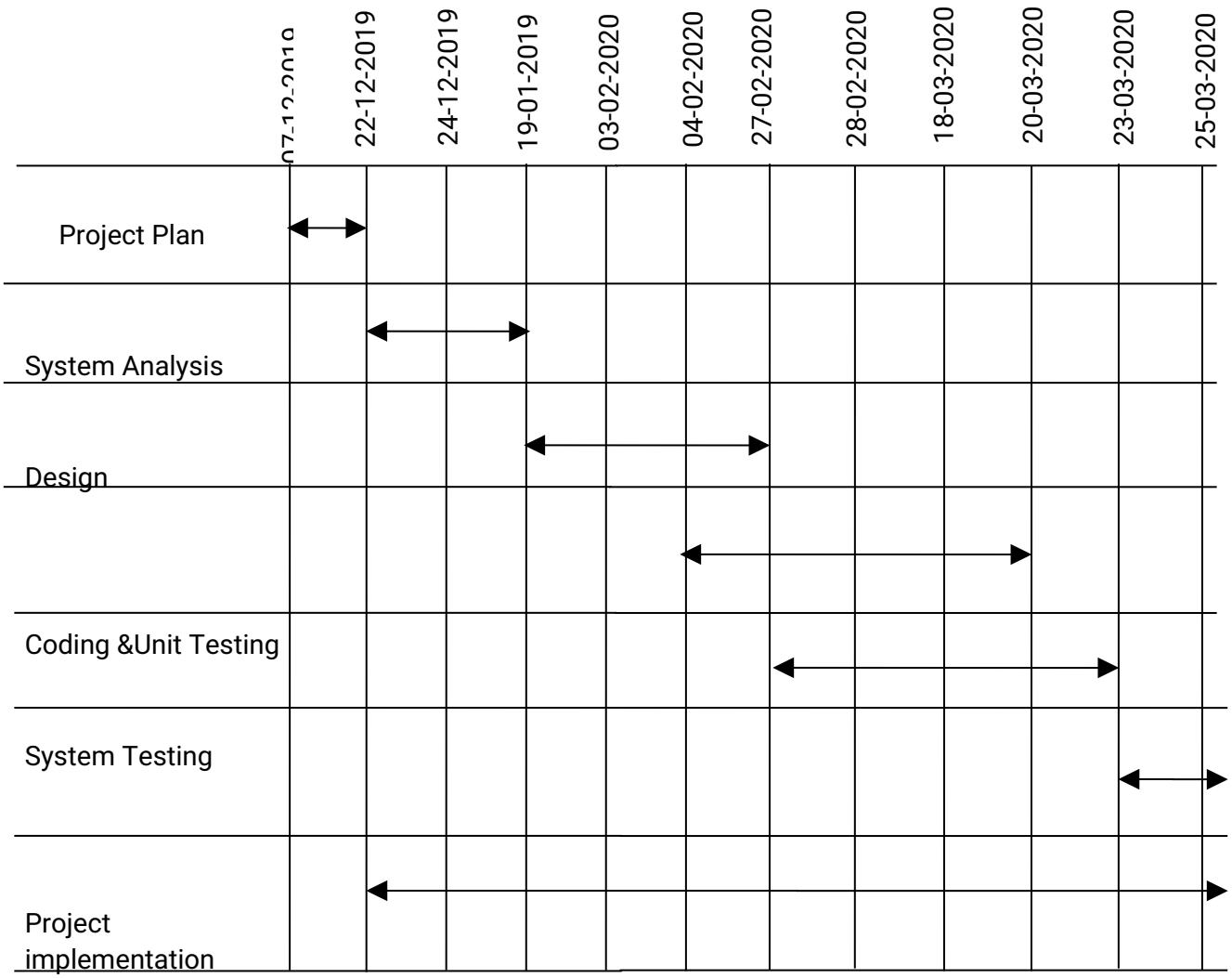
APPENDIX

8. APPENDIX

8.1 GANTT CHART

Gantt chart shows a time relationship between 'events' of the production program was regarded as revolutionary in management. Gantt chart recognizes the total program goals and it should be regarded as a series of the inter-related supporting plan (or events), that people can comprehend and follow.

The following figure is the Gantt chart of **"Environmental Threat Analysis"** .The plan explains the task versus the time they will take to complete.



8.2 MEETING MINUTES

Meeting Minute – 1

Date : 20-Nov-2019

Time : 10.00 am – 3.30 pm

Location : Kollam Corporation authority

Present :

1. Abhijith A
2. Abhishek Chandra

3. Rijo Mathew Tharakan
4. Vishakh S

On this day we collect all the required data for the development of this system. On the base of those data we select modules and assign them to each of the team and also started system analysis on this day. On collecting information on our proposed system we actually don't realize the existing manual system is as crucial as it would be when it suddenly comes to an automated virtual environment and then the challenges is as usual as in everywhere had been here in our scenario too.

Meeting Minute – 2

Date : 03-Dec-2019

Time : 10.00 am – 4.00 pm

Location : UIT KOTTARAKARA

Present :

1. Abhijith A
2. Abhishek Chandra
3. Rijo Mathew Tharakan
4. Vishakh S

On this day, we were in our college, On that day we were discussing about the designing of our system, we started design our system. Our system has 12 database tables for store and retrieve data, which are designed with the help of our project guide Mrs. Sabitha Beevi A. We went to Mrs with our idea of developing a system for the purpose. We took a while with her discussing database designs and its various entities, and we've talked about various database issues related to the data that we're supposed to store then we analysed related sites then we got an idea about the forms and other user interfaces that would make it better.

Meeting Minute – 3

Date : 22-Dec-2019

Time : 10.00 am – 3.30 pm

Location : NICT KOLLAM

Present :

1. Abhijith A
2. Abhishek Chandra
3. Rijo Mathew Tharakan
4. Vishakh S

On this day we started system coding with the guidance and assistance of our respected project guide Miss. Sabitha Beevi, She has been with us throughout the development of this project . After the analysis and designing we divided our project into two functionally independent modules admin and user module. This phase is a very important and more complex phase in Software Development Life Cycle. After then we divided our project module wise coding and given each module to individual members in our group.

Meeting Minute – 4

Date : 19-Jan-2020

Time : 10.00 am – 3.00 pm

Location : NICT KOLLAM

Present :

1. Abhijith A
2. Abhishek Chandra
3. Rijo Mathew Tharakan
4. Vishakh S

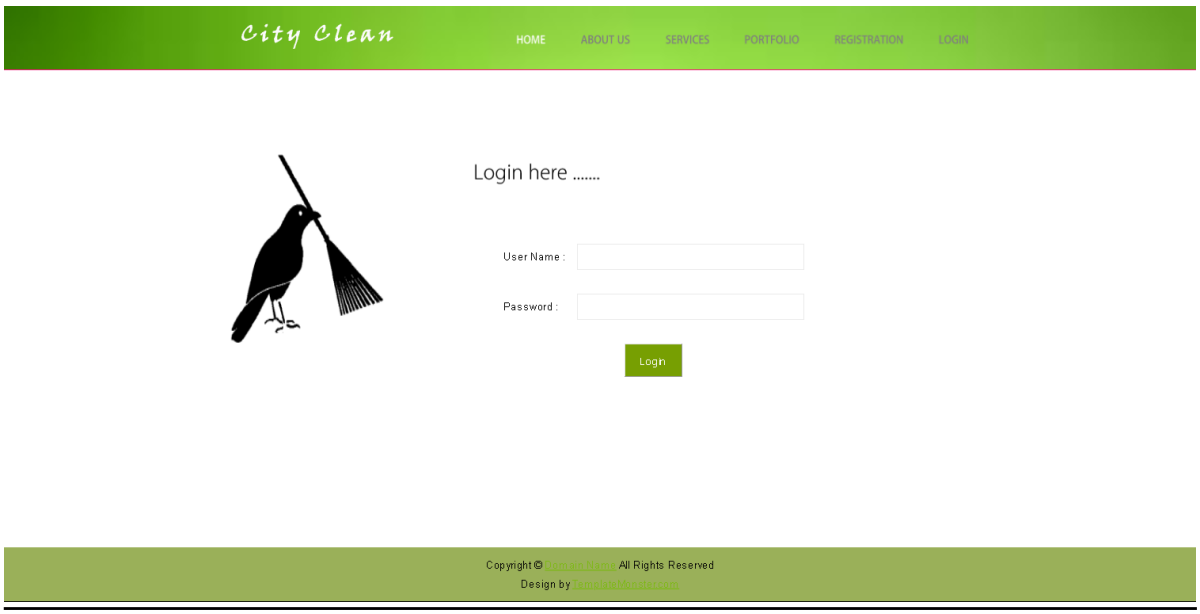
On this day we start testing each module of our project. After the testing, each module has implemented. Also, maintenance of our project is done. Finally, we have checked the full working of our project.

8.3 SCREEN LAYOUTS AND REPORTS

MAIN HOME

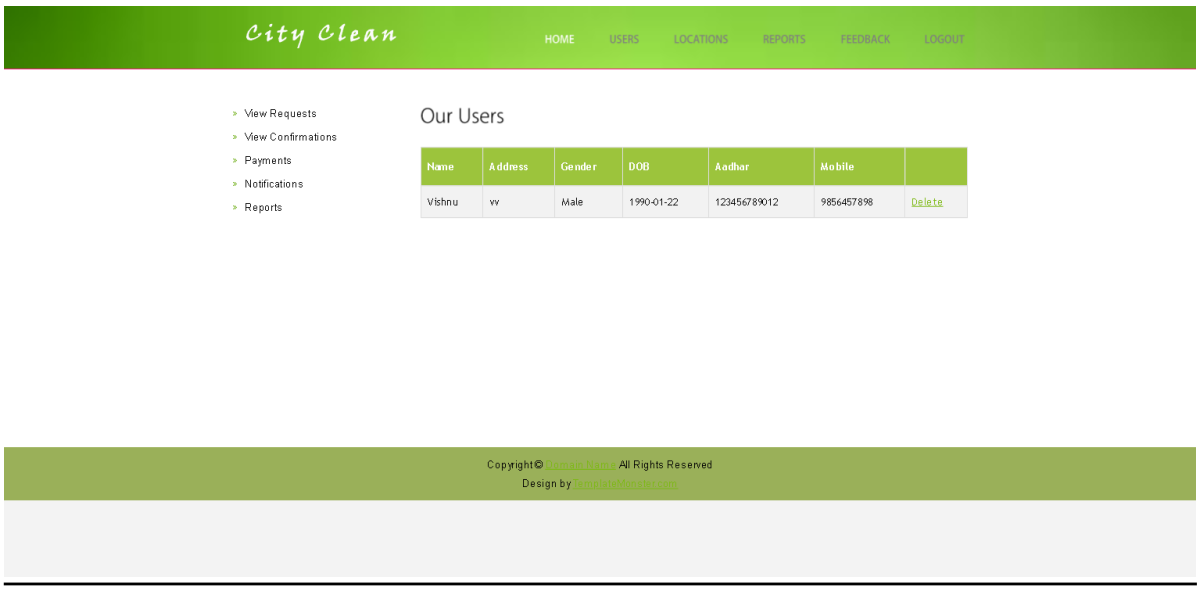

City Clean
HOME
ABOUT US
SERVICES
PORTFOLIO
REGISTRATION
LOGIN
RECYCLE ALL OBSOLETE PRODUCTS IN AN ENVIRONMENT FRIENDLY MANNER.
Shortly About Us
"ENVIRONMENTAL THREAT ANALYSIS"is an online site to upload the image of polluted area by the user and corporation will make the arrangement to clean the area and return a rectified image to the user. Inefficient municipal solid waste management system may create serious negative environmental impacts like infectious diseases, land and water pollution, obstruction of drains and loss of biodiversity .Some waste will eventually rot, but not all, and in the process it may smell, or generate methane gas, which is explosive and contributes to the greenhouse effect. Leach ate produced as waste decomposes may cause pollution. Badly-managed landfill sites may attract vermin or cause litter.
» read more
Useful links
സിവിൽ രജിസ്ട്രേഷൻ
ക്ഷേമ പെൻഷൻ
വസ്തുനികുതി
ബിൽഡിംഗ് പെർമിറ്റ് ഐ.ബി.പി.എം.എസ്
ഫയൽ അന്വേഷണം
ജനപ്രതിനിധികൾ
Our Locations
Copyright © Domain Name. All Rights Reserved
Design by TemplateMonster.com

LOGIN

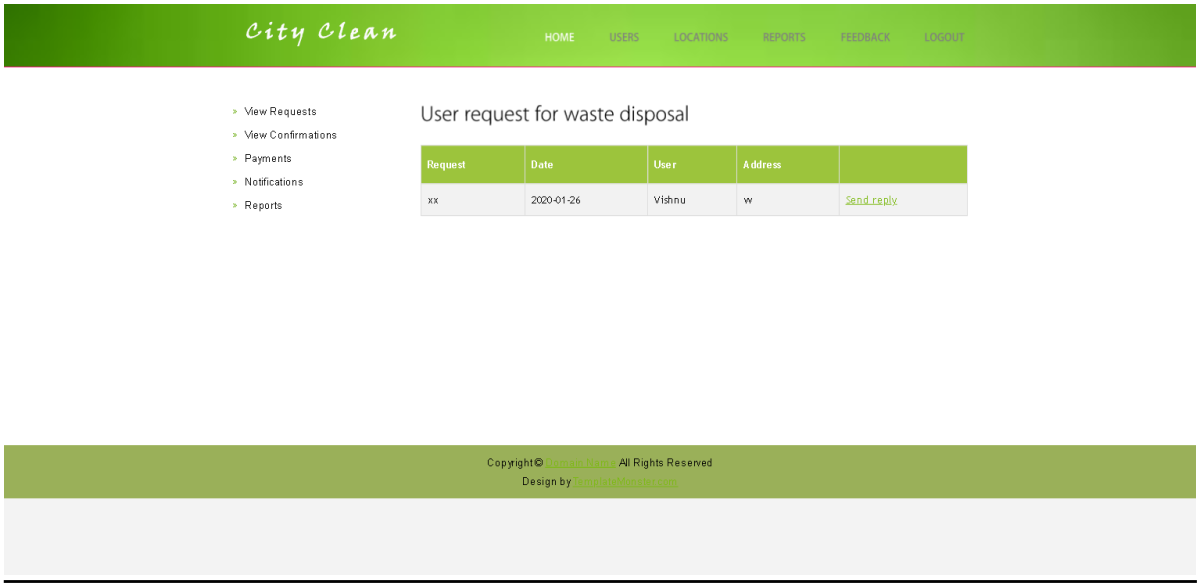


USER REGISTRATION

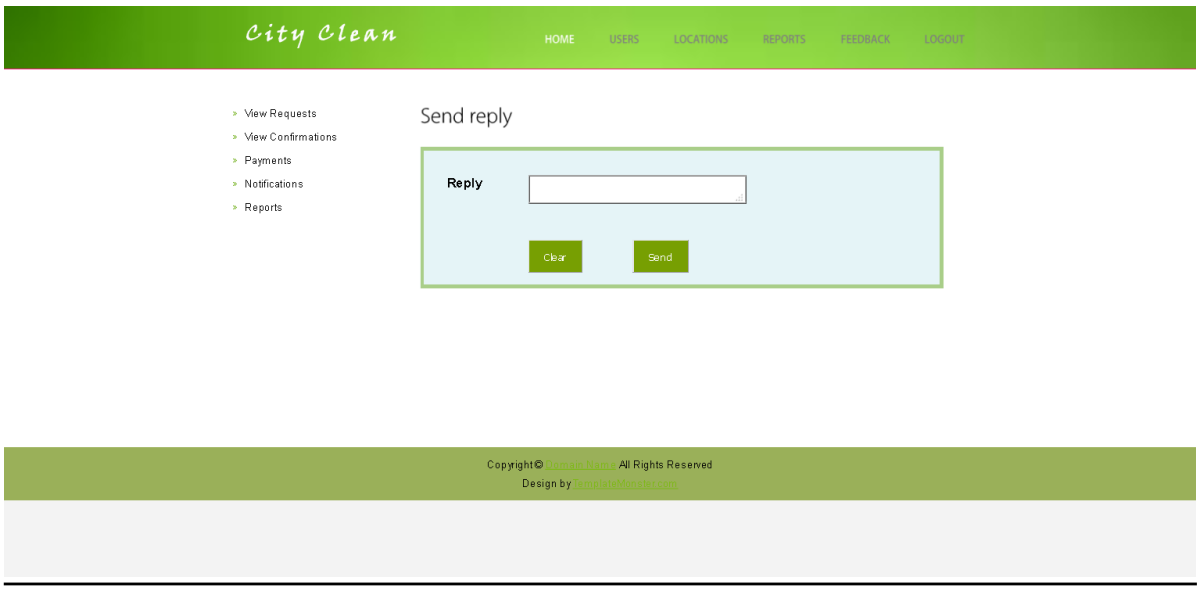
ADMIN MANAGE USERS



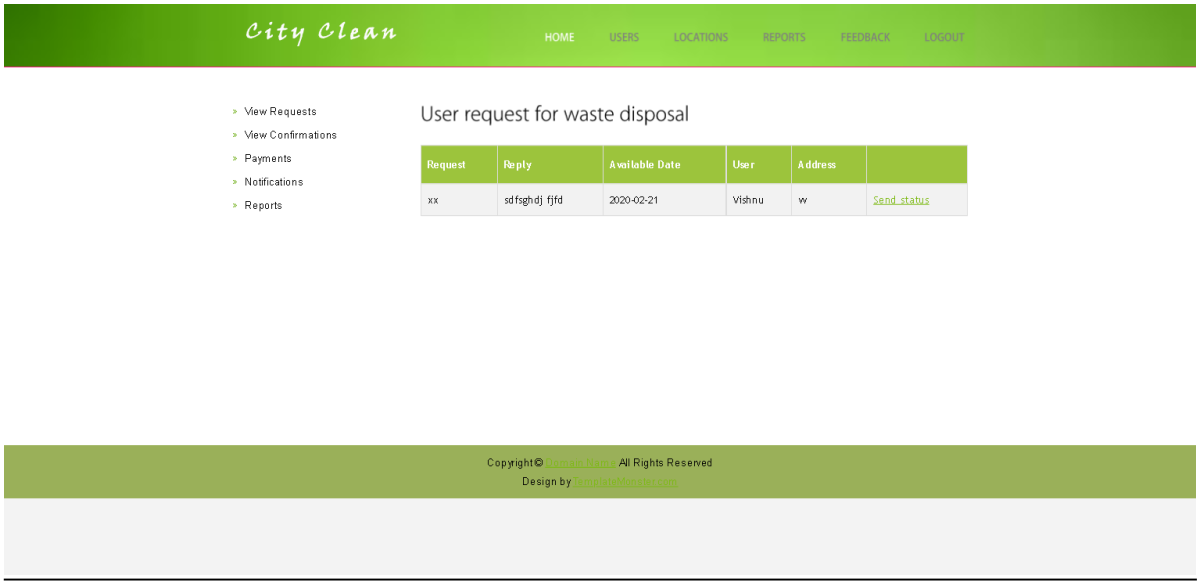
ADMIN VIEW USER REQUESTS



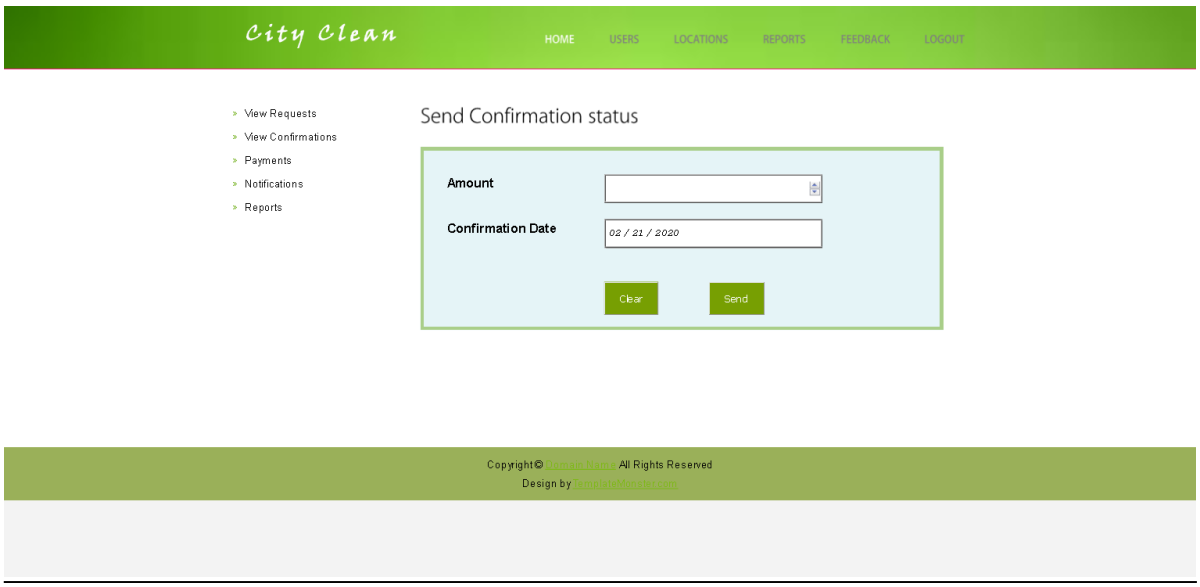
ADMIN SEND REPLY



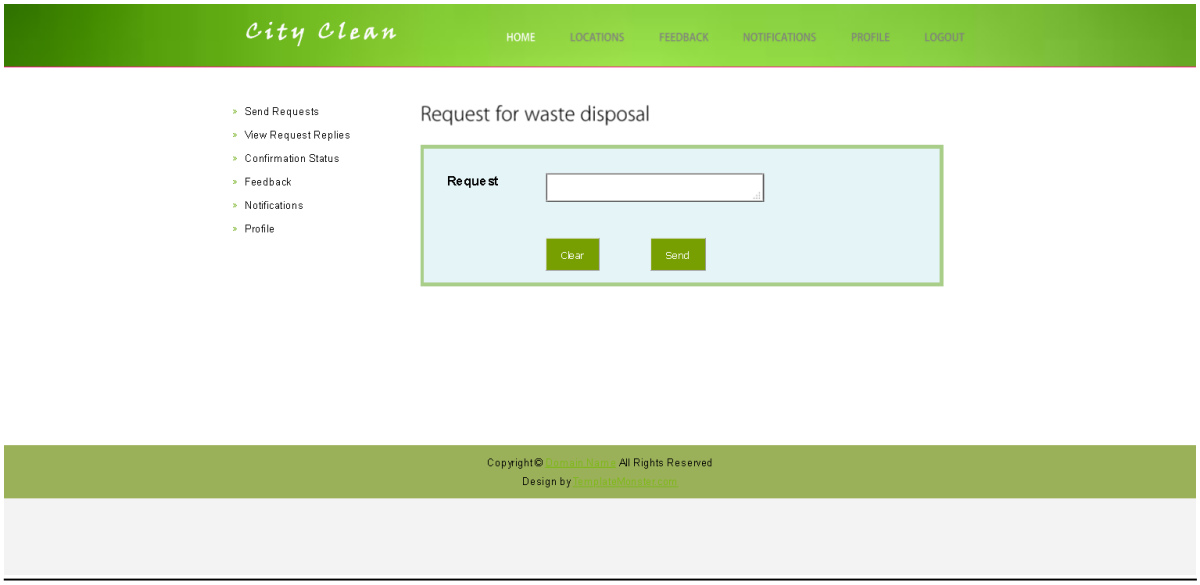
ADMIN VIEW CONFIRMATION



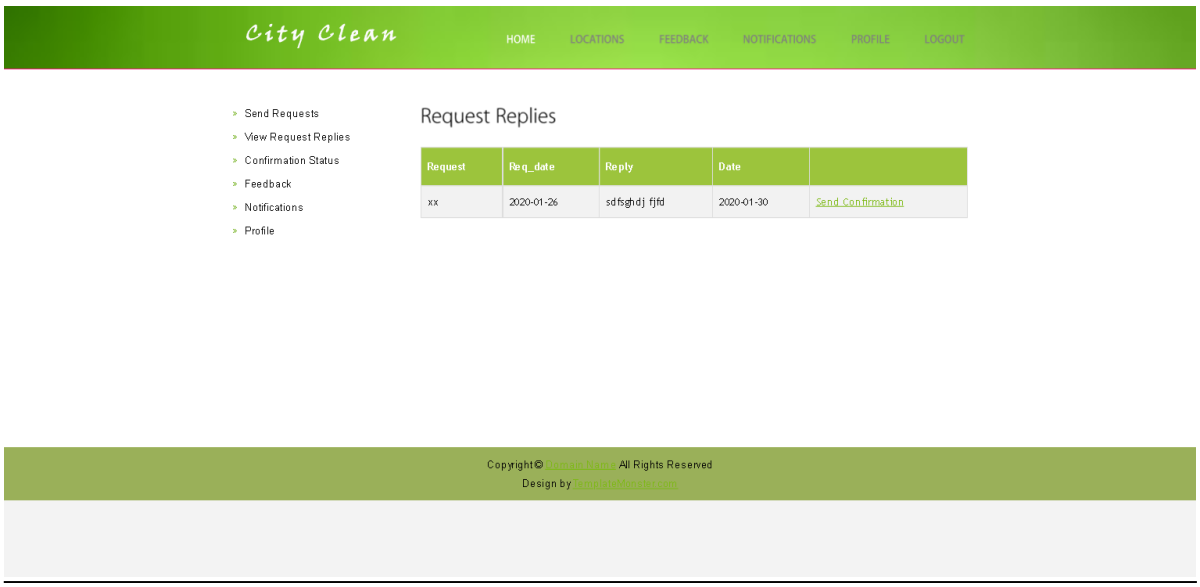
ADMIN SEND CONFIRMATION STATUS



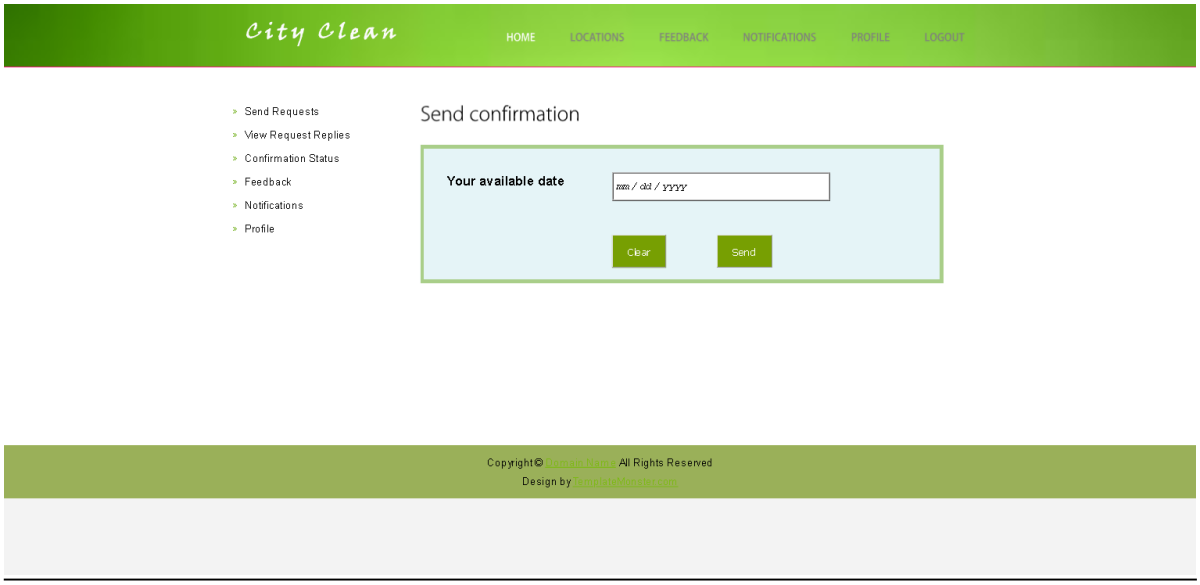
USER REQUEST FOR WASTE DISPOSAL



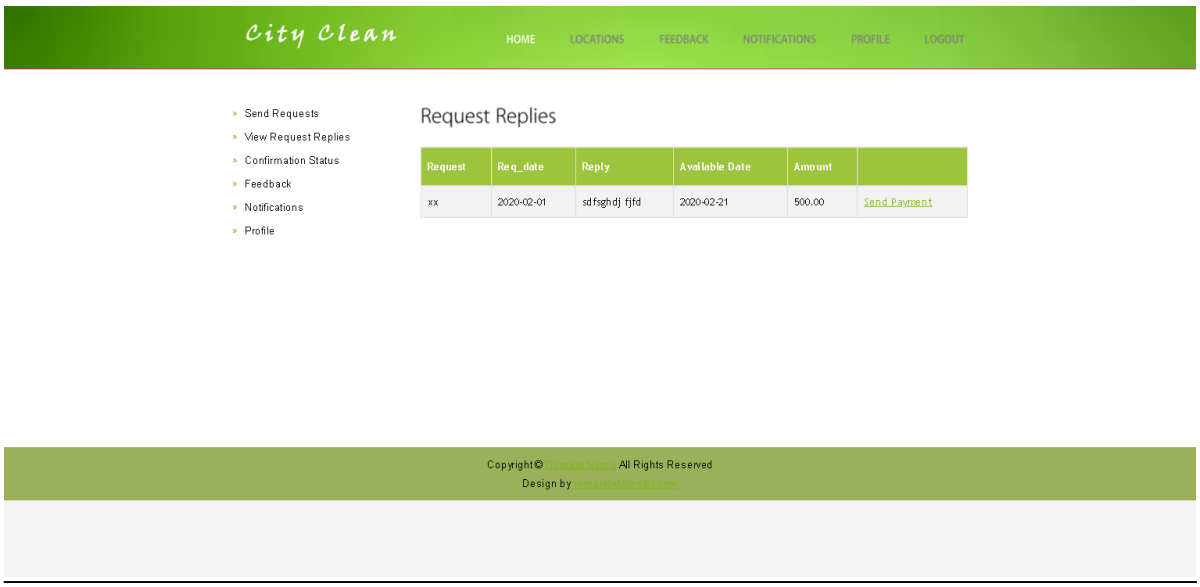
USER VIEW REQUEST REPLIES



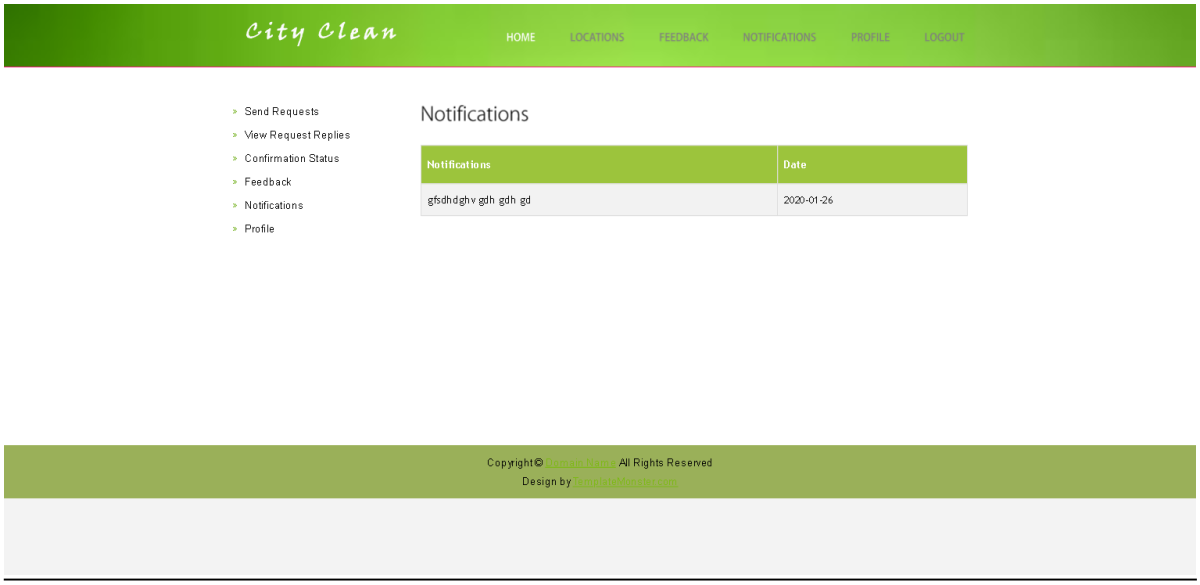
USER SEND CONFIRMATION FORM



USER VIEW CONFIRMATION STATUS



USER VIEW NOTIFICATIONS



BIBLIOGRAPHY

9. BIBLIOGRAPHY

1. Rajib Mall, Fundamentals of Software Engineering-PHI Learning Private Limited(2014)

2. Julie C Meloni, PHP, MySQL and Apache, Pearson Education

3. V.K. Jain, Advanced Programming in Web Design, Cyber Tech Publications

4. ELIAS M.AWAD, "*System Analysis and Design"*, TataMcgrawHill**,** Second Edition, 2000

Related Websites

- www.developer.google.com/web
- www.developers.blogspot.in
- www.w3schools.com
- www.stackoverflow.com
- www.developer.mozilla.org
- www.tuitorialspoint.com