HYDROHUB: YOUR SOURCE FOR WATER INFORMATION

Project Report Submitted
To
The Faculty of Science

Osmania University, Hyderabad, Telangana.

Submitted in the Partial fulfilment of the requirements for the award of

Bachelor of Science
(Mathematics, Statistics & Computer Science)

Submitted by

A. ANSHU	**1212-21-467-044**
M. ANUSHA	**1212-21-467-011**
K. AKHILA	**1212-21-467-027**
G. ASINI	**1212-21-467-028**
T. SINDHU	**1212-21-467-049**

ST. PIOUS X DEGREE AND PG COLLEGE FOR WOMEN
(Autonomous - Affiliated to Osmania University)
Nacharam Road, Hyderabad,
Telangana, India-500076
(2021-2024)

CERTIFICATE

This is to certify that the project entitled on **“HYDROHUB:YOUR SOURCE FOR WATER INFORMATION”** is a Bonafide work done by

NAME OF THE STUDENT	HALL TICKET
A. ANSHU	1212-21-467-044
M. ANUSHA	1212-21-467-011
K. AKHILA	1212-21-467-027
G. ASINI	1212-21-467-028
T. SINDHU	1212-21-467-049

In partial fulfilment for the Award of the Bachelor of Science (Mathematics, Statistics & Computer Science) to the Osmania University.

K. Kala Bharathi

Project Guide

Ms. Udaya Lakshmi
Head of the department
Computer Science

Principal

External Examiner

MANAC
Infotech (P) Limited

Ref: M02/PROJ-23-24/D0829

PROJECT COMPLETION CERTIFICATE

This is to certify that

A. ANSHU	**1212-21-467-044**
M. ANUSHA	**1212-21-467-011**
K. AKHILA	**1212-21-467-027**
G. ASINI	**1212-21-467-028**
T. SINDHU	**1212-21-467-049**

Has successfully completed the Project on "**HYDROHUB: YOUR SOURCE FOR WATER INFORMATION**" in Partial fulfilment of the **B.Sc. (Mathematics, Statistics & Computer Science)** course for the duration of 3 Months

SOFTWARE: JAVA

We wish them good luck in all future endeavors.

Project Manager

An IIM Alumnus Enterprise

DECLARATION

NAME OF THE STUDENT	HALL TICKET NO
A. ANSHU	1212-21-467-044
M. ANUSHA	1212-21-467-011
K. AKHILA	1212-21-467-027
G. ASINI	1212-21-467-028
T. SINDHU	1212-21-467-049

hereby declare that the project report entitled **"HYDROHUB: YOUR SOURCE FOR WATER INFORMATION"** submitted in partial fulfilment of the requirement for the award of Bachelors in Science Degree from **OSMANIA UNIVERSITY** under the guidance of **K. Kala Bharathi** is an original piece of work. The result drawn there on are based on the data collected originally by me for the purpose of the research. Secondarily data where ever used has been duly acknowledged and cited appropriately in the report. No part of this report has been submitted for evaluation elsewhere for the award for any degree

A. ANSHU	**1212-21-467-044**
M. ANUSHA	**1212-21-467-011**
K. AKHILA	**1212-21-467-027**
G. ASINI	**1212-21-467-028**
T. SINDHU	**1212-21-467-049**

ACKNOWLEDGEMENT

It is a matter of immense pleasure and privilege to pay special thanks to **K. Kala Bharathi,** Lecturer, **ST. PIOUS X DEGREE AND COLLEGE FOR WOMEN,** who patiently supervised this work with a great critical insight and intellectual acumen. Under her guidance, things always looked within reach. In fact, without her consistent critical suggestions, cheerful encouragement and sincere cooperation, this report should have remained futile. We express my cordial thanks to other members of teaching faculty of this department for their valuable help and cooperation.

We would like to express my gratitude for all the people who extended unending support at all stages of the project.

Our special thanks to **K.V.M. Udaya Lakshmi, Head of Department of Science**, **ST. PIOUS X DEGREE AND COLLEGE FOR WOMEN,** for her support and cooperation during the course of the project.

We wish to express my sincere thanks to **Dr.Sr.B.Velangini kumari, Principal**, **ST. PIOUS X DEGREE AND COLLEGE FOR WOMEN,** for the valuable inputs and guidance provided through Our project.

Last but not least, we would also thanks to all the employees of "**MANAC INFOTECH PVT. LTD.**" Who have directly or indirectly contributed to the successful completion of the project.

.

A. ANSHU	**1212-21-467-044**
M. ANUSHA	**1212-21-467-011**
K. AKHILA	**1212-21-467-027**
G. ASINI	**1212-21-467-028**
T. SINDHU	**1212-21-467-049**

INDEX

TOPICS **Page No's**

HYDROHUB: YOUR SOURCE FOR WATER INFORMATION

ABSTRACT

Access to safe and clean drinking water is a fundamental human right, and its availability and quality are of paramount importance. To ensure the efficient management of drinking water resources, a comprehensive and user-friendly information system is essential. This abstract outline the development of a web portal for a drinking water details system (dwds) aimed at providing real-time access to critical information related to drinking water sources, quality, distribution, and consumption.

The development of the DWDS web portal represents a significant step towards ensuring the sustainable management of drinking water resources. It empowers stakeholders with the tools and information needed to make informed decisions, enhances water quality monitoring, and promotes efficient water resource management practices. This system contributes to the broader goal of ensuring safe and reliable access to clean drinking water for all, thereby improving the well-being and health of communities.

hydro Hub emerges as a pioneering online platform dedicated to providing a comprehensive and accessible source of water-related information. In an era characterized by growing concerns over water scarcity, quality, and sustainable management, Hydro Hub stands as a crucial resource for individuals, communities, policymakers, and businesses seeking reliable data and insights on all aspects of water. HydroHub's mission is to empower individuals and organizations with the knowledge and tools needed to make informed decisions about water resources. By promoting data-driven, sustainable water management practices and fostering a sense of shared responsibility, hydro Hub serves as a catalyst for positive change in how we approach one of our most precious and vital resources. In a world where access to clean and reliable water is increasingly essential, hydro Hub stands as a beacon of knowledge and collaboration in the pursuit of water sustainability.

CHAPTER-1

INTRODUCTION

INTRODUCTION TO PROJECT: In an era where access to clean and safe drinking water is an indispensable human right and a critical determinant of public health, the development of a web portal dedicated to managing, disseminating, and monitoring drinking water details is a monumental stride towards ensuring the well-being of communities worldwide. This web portal, aptly named the Drinking Water Details System (DWDS) portal, emerges as a potent and indispensable tool in the realm of water resource management.

The DWDS portal is meticulously designed to achieve the dual objectives of enhancing transparency and optimizing efficiency in the multifaceted domain of drinking water management. In this introduction, we will explore the pivotal objectives, core functionalities, and the profound significance of this portal in the context of delivering safe and reliable drinking water to communities.

Access to clean and safe drinking water is a fundamental human right and an essential component of public health and well-being. To ensure the availability of this vital resource, effective management, monitoring, and dissemination of information about drinking water sources, quality, distribution, and consumption is imperative. In response to these needs, the development of a web portal for a Drinking Water Details System (DWDS) has emerged as a significant technological advancement in the domain of water resource management.

The DWDS web portal represents a digital platform designed to address the complexities and challenges associated with the management and maintenance of drinking water infrastructure and resources. This introduction provides an overview of the key objectives, features, and significance of such a portal in the context of ensuring safe and reliable access to clean drinking water for communities worldwide.

Objectives of the DWDS Web Portal:

The primary raison d'être of the DWDS web portal is to usher in a new era of water resource management by:

- **Real-time Monitoring:** Facilitating the instantaneous and continuous monitoring of water sources, treatment processes, and distribution networks. This real-time data empowers stakeholders to promptly respond to emerging challenges and ensure the uninterrupted supply of safe drinking water.
- **Data Accessibility:** Serving as a centralized repository for a wealth of critical data, the portal streamlines the exchange of information among diverse stakeholders including government agencies, water utilities, researchers, and the general public.
- **Water Quality Assurance:** Ensuring water quality through the integration of routine testing and immediate identification of deviations from safety standards, thereby safeguarding public health.
- **Resource Planning:** Utilizing Geographic Information System (GIS) integration to provide a visual representation of water infrastructure, facilitating informed decision- making in planning, maintenance, and expansion of water supply systems.
- **Public Engagement:** Empowering the public with easy access to vital information about local water sources, water quality reports, conservation tips, and real-time emergency alerts. Informed and engaged communities are pivotal to promoting sustainable water practices.

Significance of the DWDS Web Portal:

The DWDS web portal transcends mere technological innovation; it embodies a profound significance in addressing the most pressing challenges in the realm of drinking water, including:

1. **Public Health:** Safeguarding public health through the assurance of access to clean and safe drinking water, thereby preventing waterborne diseases and enhancing overall well-being.
2. **Resource Conservation:** Promoting responsible water consumption and conservation practices, which are pivotal in the sustainable management of water resources amidst growing concerns of water scarcity.
3. **Data-Driven Decision-Making:** Enabling data-backed decisions for stakeholders at all levels, be it in infrastructure investments, policy formulation, or emergency response, leading to more effective and efficient outcomes.
4. **Community Empowerment:** Empowering communities to actively participate in the oversight of their drinking water sources, fostering a sense of ownership, and catalysing accountability from water service providers.

In summation, the DWDS web portal is not merely a digital platform; it is a paradigm shift in the way we monitor, manage, and share information about drinking water. It embodies the spirit of collective responsibility and inclusivity, serving as a beacon of hope in our endeavour to secure this invaluable resource for the well-being of current and future generations.

CHAPTER-2

LITERATURE SURVEY

TITLE: A Web portal for safe drinking water

AUTHOR: C. Nivetha, S.P. Sangeetha

ABSTRACT: Water is the basic need for survival of all living beings. It also supports the economic growth of the country in terms of agricultural and industrial developments. Only 1% of water is fit to use out of the 71% that is available on Earth. Remaining 70% of water is being polluted with organic and inorganic matters, chemicals and other contaminants by various natural and human activities. This study is a review of various examinations done on some important Indian water bodies on its quality. This paper deals with the water quality analysis and determining the inception that spoils the quality of water. The analysis of water quality in terms of its physio- chemical properties proves that most of the water bodies are being polluted by industrial activities and appropriate water management schemes or treatment should be done to protect the water quality. It has also been recorded that the agricultural soil is affected due to the presence of excess heavy metals.

TITLE: An Integrated Web Portal for Water Quality Monitoring through Wireless Sensor Networks

AUTHOR: Lule Ahmedi, Figene Ahmedi, Besmir Sejdiu

ABSTRACT: Wireless sensor networks (WSNs) are aiding water quality monitoring with support for real-time and remote quality measurements in terrain. Environmental monitoring portals receiving data from sensors have been a practice since a while among researchers. However, the Web portal introduced here is essentially an integrated portal since it supports modelling and management of both, the observational stream data on water quality coming from wireless sensors – dynamic data, as well as of the data describing the WSN itself, its devices and the corresponding site allocation data – static da-ta. Access is given to a wide range of individuals, from water experts to WSN engineers, to general public. Experts' module infers statistics about water parameters given the experts' data and rules. The portal is further distinguished for its level of scalability: it allows adding with ease new components, like add certain new regulatory documents for water quality, and directly compare data measured by sensors with corresponding quality standards. The aim is to enrich the portal with semantics in future.

TITLE: Online Drinking Water Quality Monitoring: Review on Available and Emerging Technologies.

AUTHOR: Syed Imran, H. Najjaran, muinul H. Banna, Alex Francisque

ABSTRACT: Online drinking water quality monitoring technologies have made significant progress for source water surveillance and water treatment plant operation. The use of these technologies in the distribution system has not been favorable due to the high costs associated with installation, maintenance, and calibration of a large distributed array of monitoring sensors. This has led to a search for newer technologies that can be economically deployed on a large scale. This paper includes a brief description of important parameters for drinking water and current available technologies used in the field. The paper also provides a thorough review of the advances in sensor technology for measurement of common water quality parameters (pH, turbidity, free chlorine, dissolved oxygen, and conductivity) in drinking water distribution systems.

TITLE: IoT based smart water quality monitoring system.

AUTHOR: Varsha Lakshmikantha, Anjitha Hiriyannagowda, Akshay Manjunath, Aruna Patted, Jagadeesh Basavaiah.

ABSTRACT: Pollution of water is one of the main threats in recent times as drinking water is getting contaminated and polluted. The polluted water can cause various diseases to humans and animals, which in turn affects the life cycle of the ecosystem. If water pollution is detected in an early stage, suitable measures can be taken and critical situations can be avoided. To make certain the supply of pure water, the quality of the water should be examined in real-time. Smart solutions for monitoring of water pollution are getting more and more significant these days with innovation in sensors, communication, and Internet of Things (IoT) technology. In this paper, a detailed review of the latest works that were implemented in the arena of smart water pollution monitoring systems is presented. The paper proposes a cost effective and efficient IoT based smart water quality monitoring system which monitors the quality parameters uninterruptedly. The developed model is tested with three water samples and the parameters are transmitted to the cloud server for further action.

TITLE: Perceptions of Drinking Water Quality.

AUTHOR: Sarkar, Mousumi

ABSTRACT: water quality perceptions, discusses existing surveys that ask questions on water quality, and outlines information on actual water quality data that are available for the United States. Past research has shown that though most householders in the U.S. view their water quality favourably, a significant number have negative perceptions of their tap water; those with negative views disproportionately tend to be ethnic/racial minorities and those with low incomes. In turn, these households are more likely to turn to expensive and environmentally damaging alternatives such as bottled water.

As no other federal survey asks about perception of water quality and resulting behaviors, the authors argue that HUD could include a module of questions to the American Housing Survey (AHS) that would provide data to determine and track the link between perception of water quality and local-level communication regarding it, the effectiveness of the frequency of receiving such information, and other factors that link perception and consumption.

CHAPTER-3

SYSTEM ANALYSIS

3.1 Existing System

- The existing system for accessing drinking water details often relies on scattered and outdated information sources, making it challenging for the public to access up-to-date and comprehensive data about their water supply.

- These systems typically lack user-friendly interfaces, making it difficult for users to understand and interpret the information. Moreover, there is limited interaction between water utilities and the public, resulting in a lack.

3.1.1 Disadvantages

- Limited Access to Information
- Lack of Transparency
- Reduced Accountability
- Inefficient Communication

3.2 Proposed System

- The proposed web portal for drinking water details system seeks to revolutionize the way users' access and interact with water-related information. It will provide a centralized and user-friendly platform, offering real-time and accurate data about local water supplies.
- This proposed system will foster transparency by enabling direct communication between water utilities, regulatory bodies, and the public. Additionally, it will empower individuals and communities to actively participate in water conservation efforts and raise awareness about the importance of clean drinking water.

3.2.1 Advantages

- Accessibility
- Transparency
- Accountability
- Efficient communication

CHAPTER-4

SYSTEM REQURIEMENTS

4.2.1 Software Requirements

- **Operating System** : Windows Family or higher version
- **Techniques** : JDK 1.8
- **Data Bases** : MySQL
- **Server : **Apache Tomcat

4.2.2 Hard ware Requirements:

- **Processor :: Pentium-III (or) Higher**
- **Ram :: 64MB (or) Higher**
- **Cache :: 512MB**
- **Hard disk :: 10GB**

CHAPTER-5

SYSTEM STUDY

5.1 FEASIBILITY STUDY:

Preliminary investigation examines project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All systems are feasible if they are given unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
- Operation Feasibility
- Economic Feasibility

TECHNICAL FEASIBILITY

The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipment's have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?

Are there technical guarantees of accuracy, reliability, ease of access and data security?

OPERATIONAL FEASIBILITY

User-friendly

Customer will use the forms for their various transactions i.e., for adding new routes, viewing the routes details. Also, the Customer wants the reports to view the various transactions based on the constraints. These forms and reports are generated as user-friendly to the Client.

Reliability

The package wills pick-up current transactions on line. Regarding the old transactions, User will enter them in to the system.

Security

The web server and database server should be protected from hacking, virus etc

Portability

The application will be developed using standard open-source software (Except Oracle) like Java, tomcat web server, Internet Explorer Browser etc. this software will work both on Windows and Linux o/s. Hence portability problems will not arise.

Availability

This software will be available always.

Maintainability

The system called the wheels uses the 2-tier architecture. The 1st tier is the GUI, which is said to be front-end and the 2nd tier is the database, which uses My-Sql, which is the back-end.

The front-end can be run on different systems (clients). The database will be running at the server. Users access these forms by using the user-ids and the passwords.

ECONOMIC FEASILITY

The computerized system takes care of the present existing system's data flow and procedures completely and should generate all the reports of the manual system besides a host of other management reports.

It should be built as a web-based application with separate web server and database server. This is required as the activities are spread throughout the organization customer wants a centralized database. Further some of the linked transactions take place in different locations. Open-source software like TOMCAT, JAVA, MySQL and Linux is used to minimize the cost for the Customer.

5.2 FEASIBILITY ANALYSIS

Identifies Feasibility analysis is a critical phase in the project planning process, serving as the compass that guides decision-makers. It encompasses a comprehensive evaluation of the proposed project's potential, considering various aspects such as technical, economic, operational, scheduling, legal, and market feasibility.

From a technical perspective, it addresses whether the project can be successfully executed given the available technology, expertise, and infrastructure. It examines compatibility with existing systems, potential technical challenges, and assesses the feasibility of integrating with external systems.

CHAPTER-6

SYSTEM DESIGN

6.1 DATA FLOW DIAGRAM Process (Rectangle):

There is one process labelled "Order Processing." This process represents a generic action or function within the system, such as processing customer orders.

Data Store (Open Rectangle):

There is one data store labelled "Order Database." This data store represents a storage location for order data within the system.

Data Flow (Arrow):

There is one data flow represented by an arrow. It connects the "Order Processing" process to the "Order Database" data store. This arrow indicates the flow of order data from the process to the data store.

External Entity (Rounded Rectangle):

There is one external entity labelled "Customer." This external entity represents the user or entity external to the system that interacts with it, in this case, a customer placing an order.

In this example, the DFD illustrates a simplified process where order data flows from the "Customer" external entity to the "Order Processing" process, which then stores the order data in the "Order Database" data store. This is a basic representation of how data moves within a system. In practice, DFDs can become much more complex as additional processes, data stores, data flows, and external entities are included to model the interactions and data flow within a system.

6.2 UML DIAGRAM

UML stands for Unified Modeling Language. UML is a standardized general-purpose modeling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group.

The goal is for UML to become a common language for creating models of object- oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with UML.

The Unified Modeling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modeling and other non-software systems.

The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems.

The UML is a very important part of developing objects-oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects.

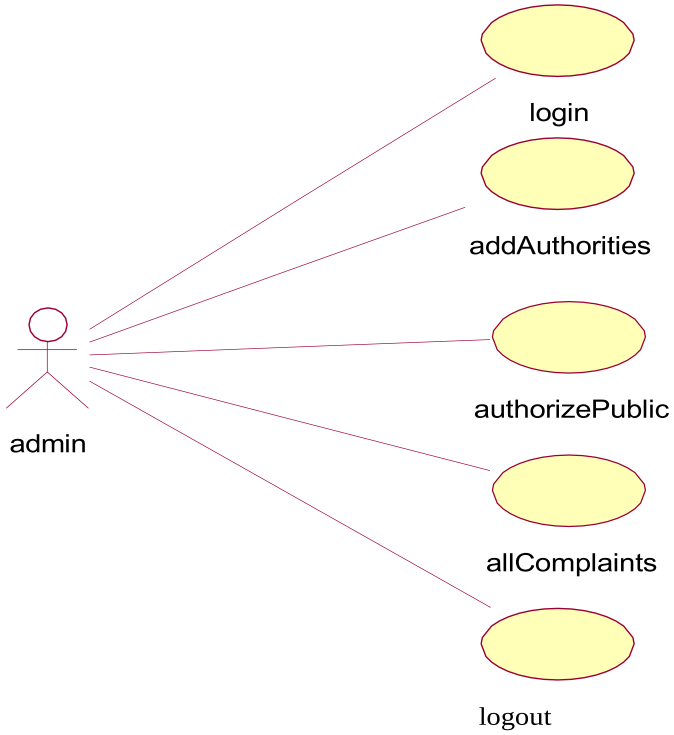
GOALS:

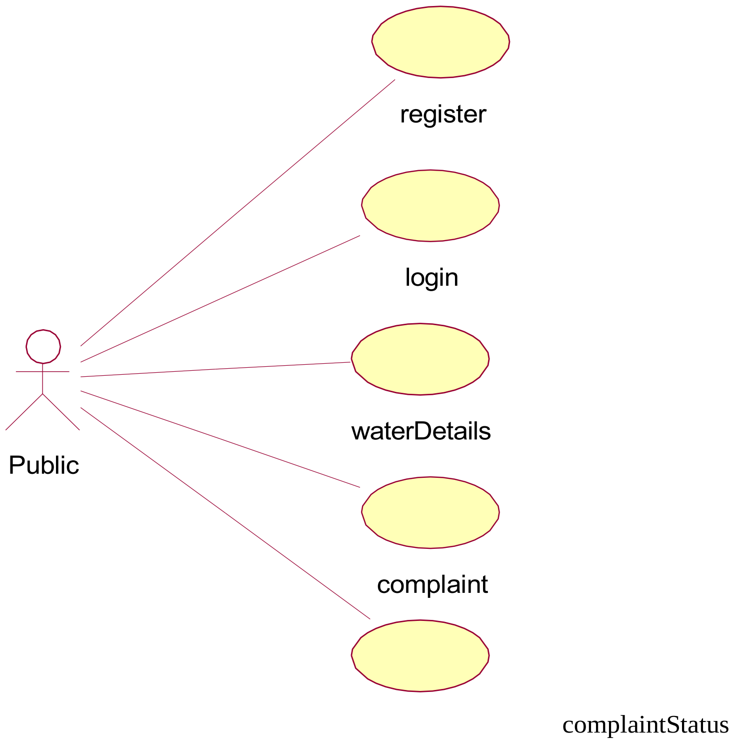
The Primary goals in the design of the UML are as follows:

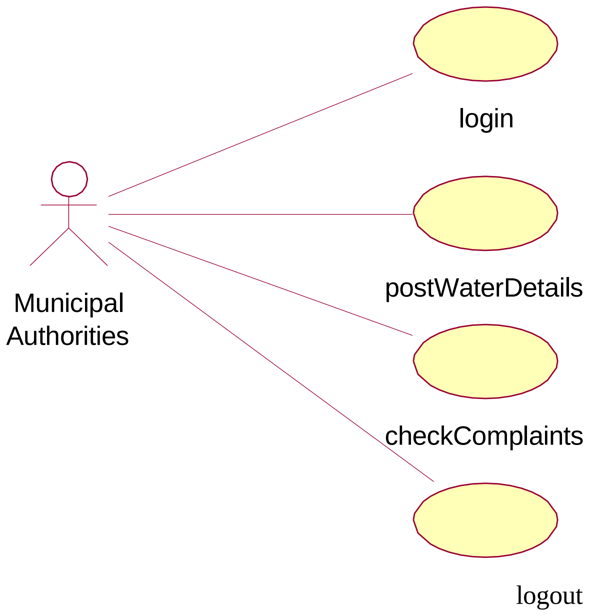
1. Provide users a ready-to-use, expressive visual modeling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modeling language.
5. Encourage the growth of OO tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.
8. Class Diagram

6.2.1 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted

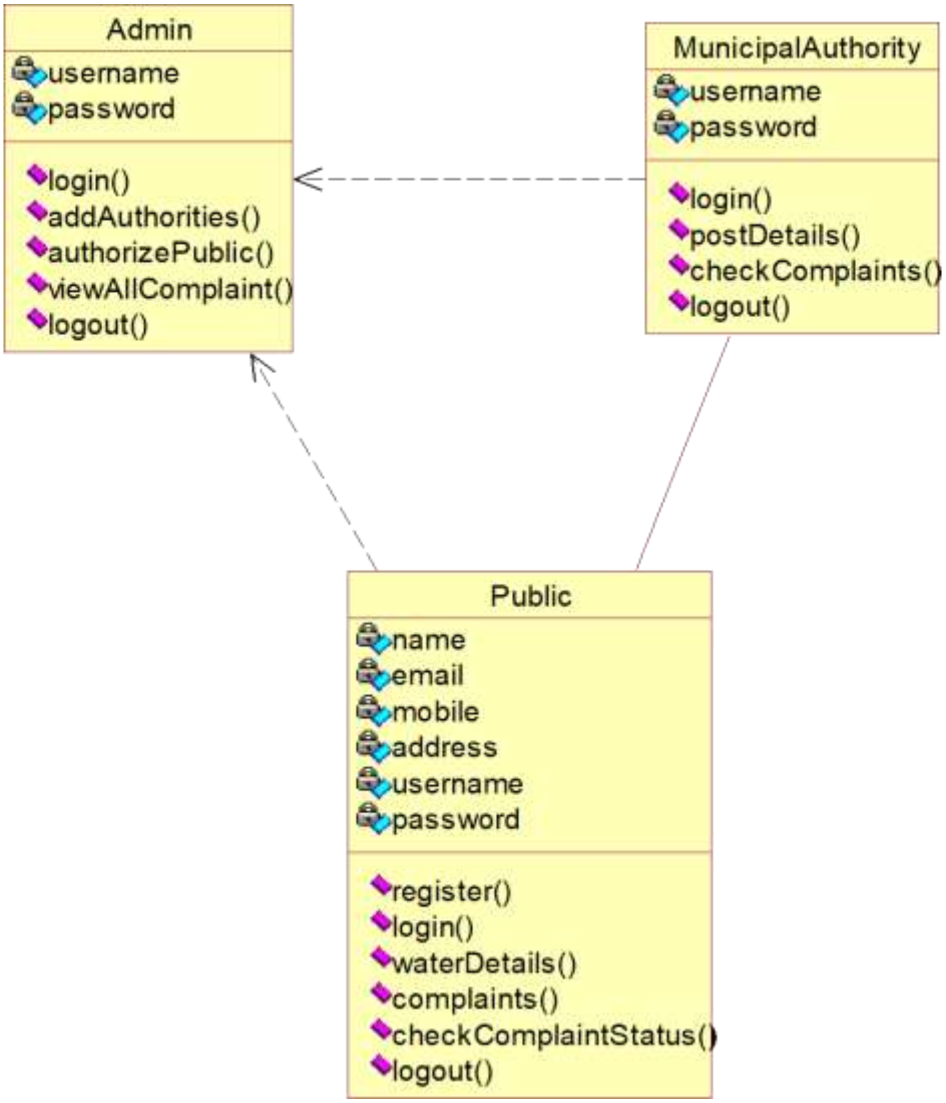



register
login
waterDetails
complaint
complaintStatus
Public


login
postWaterDetails
checkComplaints
logout
Municipal Authorities

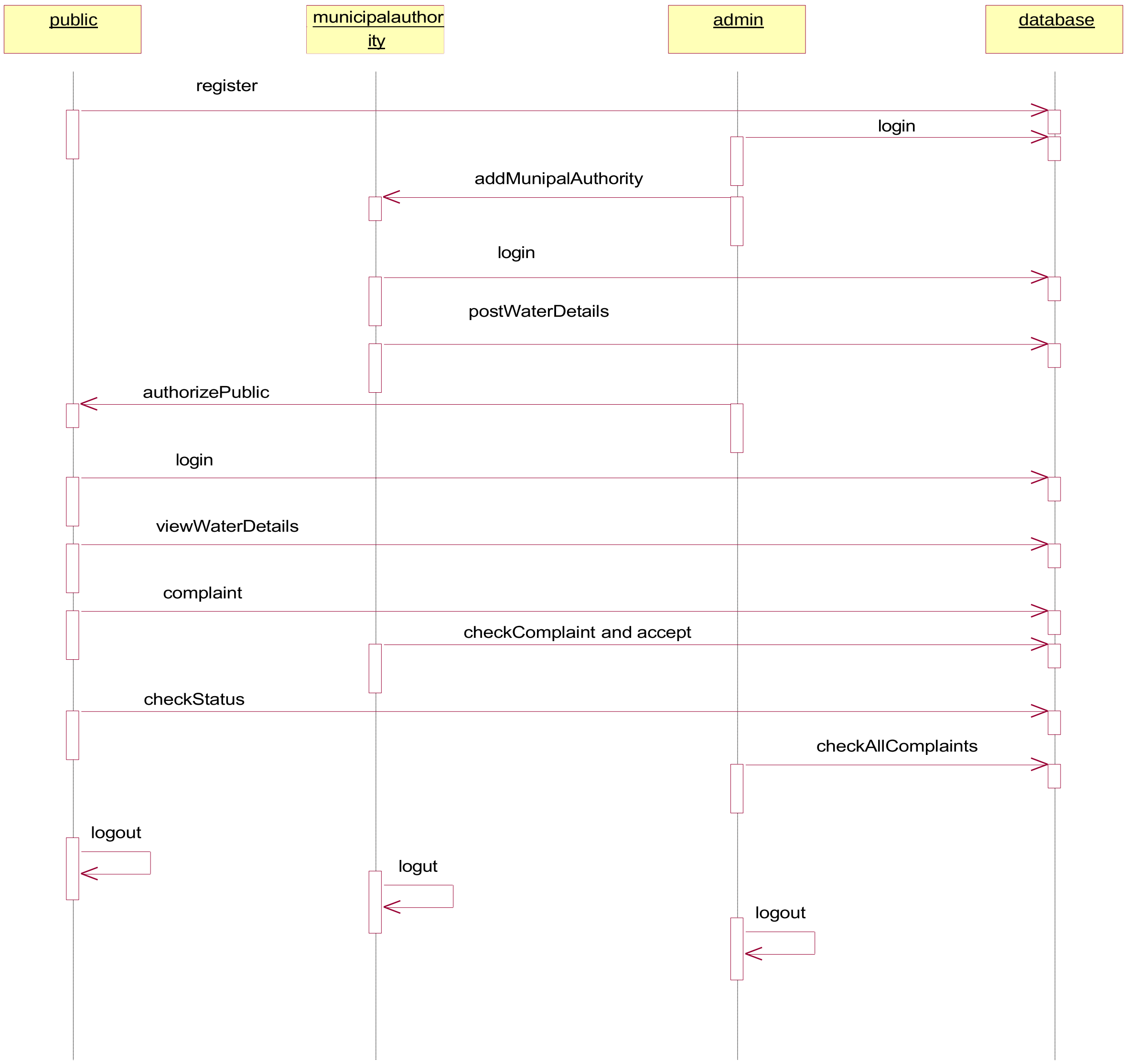
6.2.2 CLASS DIAGRAM

A class diagram is a type of diagram in the Unified Modeling Language (UML) that represents the structure and relationships of classes in a system. It is a fundamental tool used in software engineering and object-oriented modeling to visually depict the various classes, their attributes, methods, and associations within a software application or system. Class diagrams are essential for designing and understanding the architecture of software systems.



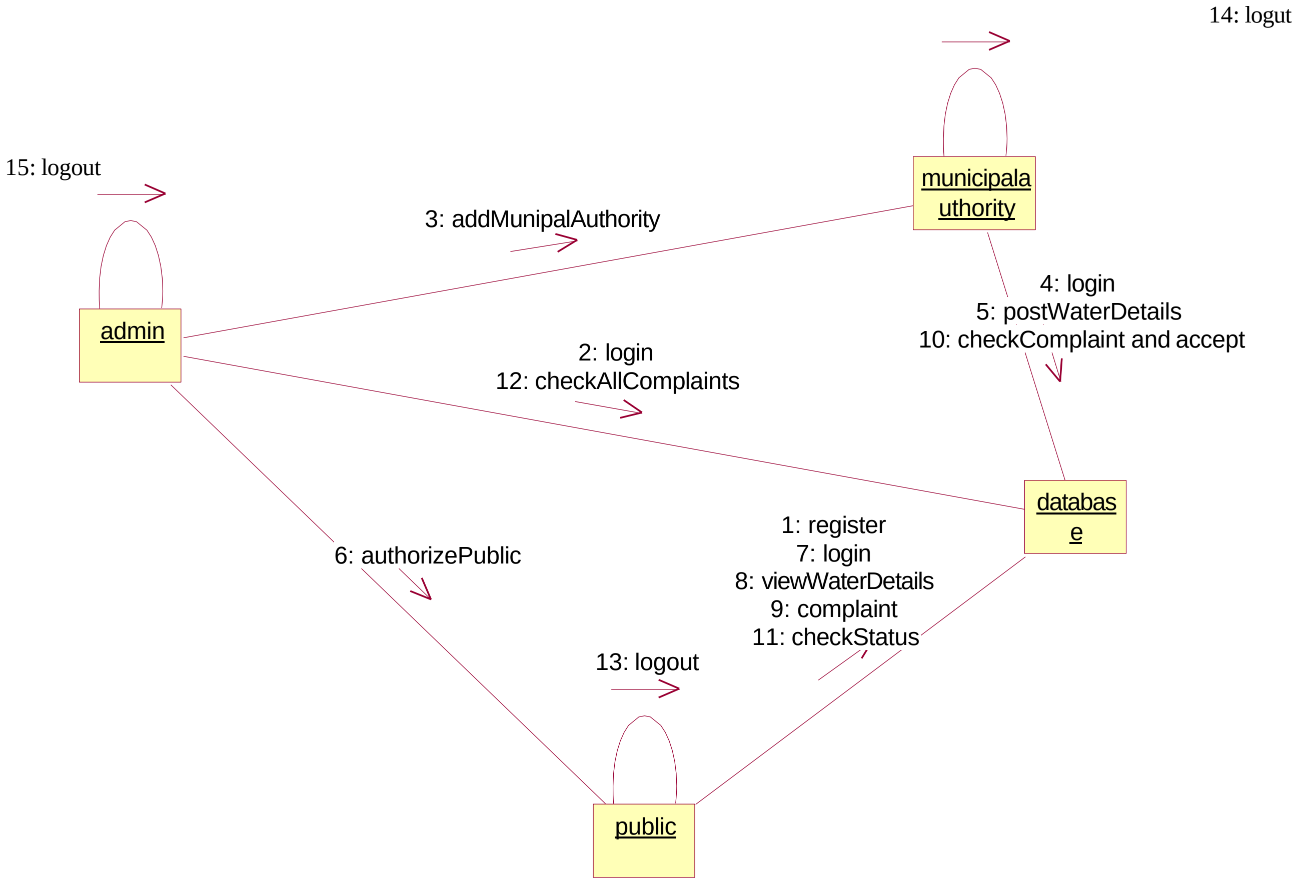
6.2.3 SEQUENCE DIAGRAM

A sequence diagram is a type of Unified Modeling Language (UML) diagram that visualizes the interactions and communication among objects or components in a system over a specific period of time. Sequence diagrams are particularly useful for modeling the dynamic aspects of a system, showing how objects or actors collaborate to achieve a particular functionality or scenario.



6.2.4 COLLABORATION DIAGRAM

Collaboration diagrams, also known as communication diagrams, are a type of Unified Modeling Language (UML) diagram that focus on visualizing the interactions and collaborations among objects or components in a system. Collaboration diagrams provide an alternative view to sequence diagrams for understanding the dynamic behavior of a system, emphasizing the relationships between objects and their interactions.



CHAPTER-7

INPUT AND OUTPUT DESIGN

7.1 INPUT DESIGN

Input design is a part of overall system design. The main objective during the input design is as given below:

- To produce a cost-effective method of input.
- To achieve the highest possible level of accuracy.
- To ensure that the input is acceptable and understood by the user.

INPUT TYPES:

It is necessary to determine the various types of inputs. Inputs can be categorized as follows:

- External inputs, which are prime inputs for the system.
- Internal inputs, which are user communications with the system.
- Operational, which are computer department's communications to the system?
- Interactive, which are inputs entered during a dialogue.
-

7.2 OUTPUT DESIGN

In general, are:

- External Outputs whose destination is outside the organization.
- Internal Outputs whose destination is with in organization and they are the User's main interface with the computer. Outputs from computer systems are required primarily to communicate the results of processing to users. They are also used to provide a permanent copy of the results for later consultation. The various types of outputs
- Operational outputs whose use is purely with in the computer department.
- Interface outputs, which involve the user in communicating directly with the system.

OUTPUT MEDIA:

In the next stage it is to be decided that which medium is the most appropriate for the output. The main considerations when deciding about the output media are:

- The suitability for the device to the particular application.
- The need for a hard copy.
- The response time required.
- The location of the users
- The software and hardware available.

Keeping in view the above description the project is to have outputs mainly coming under the category of internal outputs. The main outputs desired according to the requirement specification are: The outputs were needed to be generated as a hard copy and as well as queries to be viewed on the screen. Keeping in view these outputs, the format for the output is taken from the outputs, which are currently being obtained after manual processing. The standard printer is to be used as output media for hard copies.

CHAPTER-8

IMPLEMENTATION

8.1 MODULES

1. Admin
2. Municipal Authorities
3. Public

8.1.1 MODULES DESCRIPTION

- **Admin**.

In this application the admin is one of the Modules and here the admin can directly login with the application. And the admin can add the authorities,

Authorize public and view all complaints.

- **Municipal Authorities**

Here the municipal authorities can login with the application and the municipal authorities perform some actions like post water details, and check complaints.

- **Public**

Here the public is also one of the modules the public can register with the application and public should authorize by the admin, then only the public can login into the application, and the public can perform the following operations such as check the water details, complaint on water and check the complaint status.

CHAPTER-9

SOFTWARE ENVIRONMENT

9.1 JAVA

Initially the language was called as “oak” but it was renamed as “java” in 1995.The primary motivation of this language was the need for a platform-independent (i.e. architecture neutral) language that could be used to create software to be embedded in various consumer electronic devices.

➢ Java is a programmer’s language ➢ Java is cohesive and consistent

➢ Except for those constraint imposed by the Internet environment. Java gives the programmer, full control

Finally Java is to Internet Programming where c was to System Programming.

Importance of Java to the Internet

Java has had a profound effect on the Internet. This is because; java expands the Universe of objects that can move about freely in Cyberspace. In a network, two categories of objects are transmitted between the server and the personal computer. They are passive information and Dynamic active programs. in the areas of Security and probability. But Java addresses these concerns and by doing so, has opened the door to an exciting new form of program called the Applet.

Applications and applets.

An application is a program that runs on our computer under the operating system of that computer. It is more or less like one creating using C or C++.Java’s ability to create Applets makes it important. An Applet I San application, designed to be transmitted over the Internet and executed by a Java-compatible web browser. An applet I actually a tiny Java program, dynamically downloaded across the network, just like an image. But the difference is, it is an intelligent program, not just a media file. It can be reacted to the user input and dynamically change

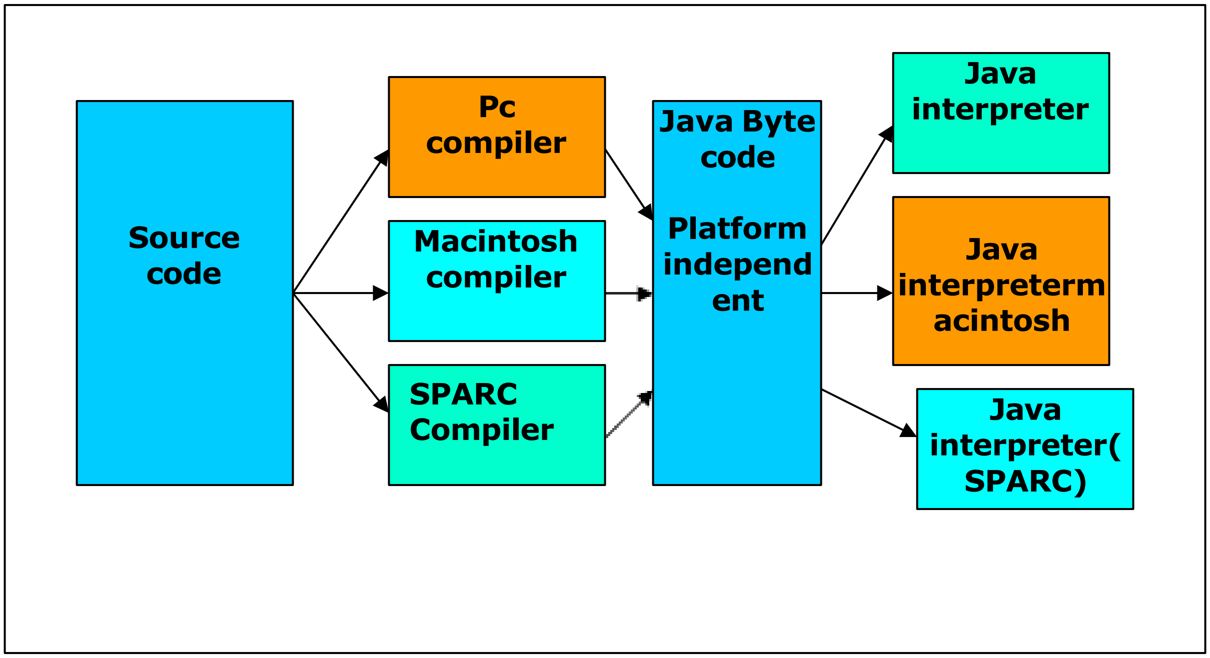
Java Architecture

Java architecture provides a portable, robust, high performing environment for development. Java provides portability by compiling the byte codes for the Java Virtual Machine, which is then interpreted on each platform by the run-time environment. Java is a dynamic system, able to load code when needed from a machine in the same room or across the planet.

Compilation of code

When you compile the code, the Java compiler creates machine code (called byte code) for a hypothetical machine called Java Virtual Machine (JVM). The JVM is supposed t executed the byte code. The JVM is created for the overcoming the issue of probability. The code is written and compiled for one machine and interpreted on all machines. This machine is called Java Virtual Machine.

Compiling and interpreting java source code.



During run-time the Java interpreter tricks the byte code file into thinking that it is running on a Java Virtual Machine. In reality this could be an Intel Pentium windows 95 or sun SPARCstation running Solaris or Apple Macintosh running system and all could receive code from any computer through internet and run the Applets.

Simple:

Java was designed to be easy for the Professional programmer to learn and to use effectively. If you are an experienced C++ Programmer. Learning Java will oriented features of C++. Most of the confusing concepts from C++ are either left out of Java or implemented in a cleaner, more approachable manner. In Java there are a small number of clearly defined ways to accomplish a given task.

Object oriented

Java was not designed to be source-code compatible with any other language. This allowed the Java team the freedom to design with a blank state. One outcome of this was a clean usable, pragmatic approach to objects. The object model in Java is simple and easy to extend, while simple types, such as integers, are kept as high-performance non-objects.

Robust

The multi-platform environment of the web places extraordinary demands on a program, because the program must execute reliably in a variety of systems. The ability to create robust programs. Was given a high priority in the design of Java. Java is strictly typed language; it checks your code at compile time and runtime.

Java virtually eliminates the problems of memory management and deal location, which is completely automatic. In a well-written Java program, all run-time errors can and should be managed by your program.

4.4.2 Servlets/JSP INTRODUCTION

A Servlet Is a generic server extension. a Java class that can be loaded

Dynamically to expand the functionality of a server. Servlets are commonly used with web servers. Where they can take the place CGI scripts.

A servlet is similar to proprietary server extension, except that it runs inside a Java Virtual Machine (JVM) on the server, so it is safe and portable

Servlets operate solely within the domain of the server.

Unlike CGI and Fast CGI, which use multiple processes to handle separate program or separate requests, separate threads within web server process handle all servlets. This means that servlets are all efficient and scalable.

Servlets are portable; both across operating systems and also across web servers. Java Servlets offer the best possible platform for web application development.

Servlets are used as replacement for CGI scripts on a web server, they can extend any sort of server such as a mail server that allows servlet's extend its functionality perhaps by performing a virus scan on all attached documents or handling mail filtering tasks.

Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming including inextensible scripting solutions platform- specific API's and incomplete interface.

Servlets are objects that conform to a specific interface that can be plugged into a Java- based server. Servlets are to the server-side what applets are to the server-side what applets are to the client-side-object byte codes that can be dynamically loaded off the net. They differ from applets in than they are faceless objects (without graphics or a GUI component).They serve as platform independent, dynamically loadable, pluggable helper byte code objects on the server side that can be used to dynamically extend server-side functionality.

For example, an HTTP servlet can be used to generate dynamic HTML content when you use servlets to do dynamic content you get the following advantages:

- They’re faster and cleaner than CGI scripts ➢ They use a standard API (the servlet API)
- They provide all the advantages of Java (run on a variety of servers without needing to be rewritten)

Attractiveness of Servlets:

They are many features of servlets that make them easy and attractive to use these include:

- Easily configure using the GUI-based Admin tool]
- Can be Loaded and Invoked from a local disk or remotely across the network.
- Can be linked together or chained, so that on servlet can call another servlet, or several servlets in sequence.
- Can be called dynamically from within HTML, pages using server-side include- tags.
- Are secure-even when downloading across the network, the servlet security model and servlet and box protect your system from unfriendly behavior.,

Advantages of the servlet API

One of the great advantages of the servlet API is protocol independent. It assumes nothing about:

- The protocol being used to transmit on the net ➢ How it is loaded
- The server environment it will be running in
- These quantities are important, because it allows the Servlet API to be embedded in many different kinds of servers. There are other advantages to the servlets API as well These include:
- It’s extensible-you can inherit all your functionality from the base classes made available to you
- It’s simple small, and easy to use.

- Servlets are persistent. Servlet are loaded only by the web server and can maintain services between requests.
- Servlets are fast. Since servlets only need to be l\loaded once, they offer much better performance over their CGI counterparts.
- Servlets are platform independent.
- Servlets are extensible Java is a robust, object-oriented programming language, which easily can be extended to suit your needs.
- Servlets are secure
- Servlets are used with a variety of client.

Servlets are classes and interfaces from tow packages, javax. servlet and javax. servlet. Http. The java. Servlet package contains classes t support generic, protocol-independent servlets. The classes in the javax. servlet. Http package To and HTTP specific functionality extend these classes

Every servlet must implement the javax. servlets interface. Most servlets implement it by extending oneoftwoclasses. javax. Servlet. Generic Servlet or javax. servlet. Http. HTTP Servlet. A protocol-independent servlet should subclass Generic-Servlet. while an Http servlet should subclass HTTP Servlet, which is itself a subclass of Generic-servlet with added HTTP-specific functionality.

Unlike a java program, a servlet does not have a main () method, Instead the server in the process of handling requests invoke certain methods of a servlet. Each time the server dispatches a request to a servlet, it invokes the servlet's Service () method,

A generic servlet should override its service () method to handle requests as appropriate for the servlet. The service() accepts two parameters a request object and a response object .The request object tells the servlet about the request, while the response object is used to return a response

InContrast.anHttp servlet usually does not override the service () method. Instead it overrides doGet() to handle GET requests and do Post() to handle Post requests. An Http servlet can override either or both of these modules the service () method of HTTP Servlet handles the setup and dispatching to all the doXXX () methods. Which is why it usually should not be overridden

The remainders in the javax. servlet and javax. servlet. Http. Package are largely support classes. The Servlet Request and Servlet Response classes in javax. servlet provide access to generic server requests and responses while HttpServletRequest and HttpServletResponse classes in javax.servelt provide access to generic server requests and responses while HttpServletRequest and HttpServletResponse in javax.servlet.http provide access a HTTP requests and responses . The javax. servlet. Http provide contains an HTTP Session class that provides built-in session tracking functionality and Cookie class that allows quickly setup and processing HTTP Cookies.

Loading Servlets:

Servlets can be loaded from their places. From a directory that is on the CLASSPATH. The CLASSPATH of the Java Webserver includes service root/classes/, which is where the system classes reside

From the <SERVICE_ROOT/servlets/directory. This is not in the server’s class path. A class loader is used to create servlets form this directory. New servlets can be added-existing servlets can be recompiled and the server will notice these changes. From a remote location. For this a code base like http://nine.eng/classes/foo/ is required in addition to the servlet’s class name. Refer to the admin Gui docs on servlet section to see how to set this up.

Loading Remote Servlets

Remote servlets can be loaded by:

➢ Configuring the admin Tool to setup automatic loading of remote servlets. ➢ Selecting up server side include tags in .html files

➢ Defining a filter chain Configuration

Invoking Servlets

A servlet invoker is a servlet that invokes the “server” method on a named servlet. If the servlet is not loaded in the server, then the invoker first loads the servlet (either form local disk or from the network) and the then invokes the “service” method. Also like applets, local servlets in the server can be identified by just the class name. In other words, if a servlet name is not absolute.it is treated as local.

A Client can Invoke Servlets in the Following Ways:

- The client can ask for a document that is served by the servlet.
- The client(browser) can invoke the servlet directly using a URL, once it has been mapped using the SERVLET ALIASES Section of the admin GUI
- The servlet can be invoked through server side include tags.
- The servlet can be invoked by placing it in the servlets/directory ➤ The servlet can be invoked by using it in a filter chain

The Servlet Life Cycle: -

The Servlet life cycle is one of the most exciting features of Servlets. This life cycle is a powerful hybrid of the life cycles used in CGI programming and lower-level NSAPI and ISAPI programming.

The servlet life cycle allows servlet engines to address both the performance and resource problems of CGI and the security concepts of low-level server API programming.

Servlet life cycle is highly flexible Servers have significant leeway in how they choose to support servlets. The only hard and fast rule is that a servlet engine must conform to the following life cycle contact:

destroyed an garbage collected without handling any client request or after handling just one request

The most common and most sensible life cycle implementations for HTTP servlets are: Single java virtual machine and astatine persistence.

Init and Destroy:-

Just like Applets servlets can define init () and destroy () methods, A servlets init(ServiceConfig) method is called by the server immediately after the server constructs the servlet's instance. Depending on the server and its configuration, this can be at any of these times

@ When the server states
@ When the servlet is first requested, just before the service() method is invoked @ At the request of the server administrator

In any case, nit() is guaranteed to be called before the servlet handles its first request

The init() method is typically used to perform servlet initialization creating or loading objects that are used by the servlet in handling of its request. In order to providing a new servlet any information about itself and its environment, a server has to call a servlets in it() method and pass an object that implement the ServletConfig interface.

This ServletConfig object supplies a servlet with information about its initialization parameters. These parameters are given to the servlets and are not associated with any single request. They can specify initial values, such as where a counter should begin counting, or default values, perhaps a template to use when not specified by the request,

The server calls a servlet's destroy() method when the servlet is about to be unloaded. In the destroy() method, a servlet should free any resources it has acquired that will not be garbage collected. The destroy() method also gives a servlet a chance to write out its unsaved. cached information or any persistent information that should be read during the next call to init().

Session Tracking:

HTTP is a stateless protocol, it provides no way for a server to recognize that a sequence of requests is all from the same client. This causes a problem for application such as shopping cart applications. Even in chat application server can’t know exactly who’s making a request of several clients.

The solution for this is for client to introduce itself as it makes each request, Each clients needs to provide a unique identifier that lets the server identify it, or it needs to give some information that the server can use to properly handle the request, There are several ways to send this introductory information with each request Such as:

USER AUTHORIZATION:

One way to perform session tracking is to leverage the information that comes with User authorization. When a web server restricts access to some of its resources to only those clients that log in using a recognized username and password. After the client logs in, the username is available to a servlet through getRemoteUser ()

Wean use the username to track the session. Once a user has logged in, the browser remembers her username and resends the name and password as the user views new pages on the site. A servlet can identify the user through her username and they’re by
Track her session.

The biggest advantage of using user authorization to perform session tracking is that it’s easy to implement. Simply tell the protect a set of pages, and use getRemoteUser () to identify each client. Another advantage is that the technique works even when the user accesses your site form or exists her browser before coming back.

The biggest disadvantage of user authorization is that it requires each user to register for an account and then log in in each time the starts visiting your site. Most users will tolerate registering and lagging in as a necessary evil when they are accessing sensitive information, but its all overkill for simple session tracking. Other problem with user authorization is that a user cannot simultaneously maintain more than one session at the same site.

Hidden Form Fields:

One way to support anonymous session tracking is to use hidden from fields. As the name implies, these are fields added to an HTML, form that are not displayed in the client's browser, they are sent back to the server when the form that contains them is submitted.

In a sense, hidden form fields define constant variables for a form. To a servlet receiving a submitted form, there is no difference between a hidden fields and a visible filed.

As more and more information is associated with a client session. It can become burdensome to pass it all using hidden form fields. In these situations, it's possible to pass on just a unique session ID that identifies as particular client's session.

That session ID can be associated with complete information about its session that is stored on the server.

The advantage of hidden form fields is their ubiquity and support for anonymity. Hidden fields are supported in all the popular browsers, they demand on special server requirements, and they can be used with clients that haven't registered or logged in.

The major disadvantage with this technique, however is that works only for a sequence of dynamically generated forms, The technique breaks down immediately with static documents, emailed documents book marked documents and browser shutdowns.

URL Rewriting:

URL rewriting is another way to support anonymous session tracking, With URL rewriting every local URL the user might click on is dynamically modified. or rewritten, to include extra, information. The extra information can be in the deform of extra path information, added parameters, or some custom, server-specific.URL change. Due to the limited space available in rewriting a URL, the extra information is usually limited to a unique session.

Each rewriting technique has its own advantage and disadvantage

Using extra path information works on all servers, and it works as a target for forms that use both the Get and Post methods. It does not work well if the servlet has to use the extra path information as true path information

The advantages and disadvantages of URL. Rewriting closely match those of hidden form fields, The major difference is that URL rewriting works for all dynamically created documents, such as the Help servlet, not just forms. With the right server support, custom URL rewriting can even work for static documents.

Persistent Cookies:

A fourth technique to perform session tracking involves persistent cookies. A cookie is a bit of information. sent by a web server to a browser that can later be read back form that browser. When a browser receives a cookie, it saves the cookie and there after sends the cookie back to the server each time it accesses a page on that server, subject to certain rules. Because a cookie's value can uniquely identify a client, cookies are often used for session tracking.

Persistent cookies offer an elegant, efficient easy way to implement session tracking. Cookies provide as automatic an introduction for each request as we could hope for. For each request, a cookie can automatically provide a client's session ID or perhaps a list of client's performance. The ability to customize cookies gives them extra power and versatility.

The biggest problem with cookies is that browsers don't always accept cookies sometimes this is because the browser doesn't support cookies. More often it's because

The browser doesn't support cookies. More often it's because the user has specifically configured the browser to refuse cookies.

The power of serves:

The power of servlets is nothing but the advantages of servlets over other approaches, which include portability, power, efficiency, endurance, safety elegance, integration, extensibility and flexibility.

Portability:

As servlets are written in java and conform to a well-defined and widely accepted API. They are highly portable across operating systems and across server implementation

We can develop a servlet on a windows NT machine running the java web server and later deploy it effortlessly on a high-end Unix server running Apache. With servlets we can really “write once, serve everywhere”

Servlet portability is not the stumbling block it so often is with applets, for two reasons

First, Servlet portability is not mandatory i.e. servlets has to work only on server machines that we are using for development and deployment

Second, servlets avoid the most error-prone and inconstancy implemented portion of the java languages.

Power:

Servlets can harness the full power of the core java. API’s: such as Networking and Url access, multithreading, image manipulation, data compression, data base connectivity, internationalization, remote method invocation(RMI) CORBA connectivity, and object serialization, among others.

Efficiency And Endurance:

Servlet invocation is highly efficient, once a servlet is loaded it generally remains in the server’s memory as a single object instance, there after the server invokes the servlet to handle a request using a simple, light weighted method invocation. Unlike the CGI, there’s no process to spawn or interpreter to invoke, so the servlet can begin handling the request almost immediately, Multiple, concurrent requests are handled the request almost immediately. Multiple, concurrent requests are handled by separate threads, so servlets are highly scalable.

Servlets in general are enduring objects. Because a servlet stays in the server's memory as a single object instance. it automatically maintains its state and can hold onto external resources, such as database connections.

Safety:

Servlets support safe programming practices on a number of levels.

As they are written in java, servlets inherit the strong type safety of the java language. In addition, the servlet API is implemented to be type safe. Java's automatic garbage collection and lack of pointers mean that servlets are generally safe from memory management problems like dangling pointers invalid pointer references and memory leaks.

Servlets can handle errors safely, due to java's exception – handling mechanism. If a servlet divides by zero or performs some illegal operations, it throws an exception that can be safely caught and handled by the server.

A server can further protect itself from servlets through the use of java security manager.
A server can execute its servlets under the watch of a strict security manager.

Elegance:

The elegance of the servlet code is striking. Servlet code is clean, object oriented modular and amazingly simple one reason for this simplicity is the served API itself. Which includes methods and classes to handle many of the routine chores of servlet development. Even advanced to operations like cookie handling and session tracking are abstracted int convenient classes.

Integration:

Servlets are tightly integrated with the server. This integration allows a servlet to cooperate with the server in two ways. for e.g.: a servlet can use the server to translate file paths, perform logging, check authorization, perform MIME type mapping and in some cases even add users to the server's user database.

Extensibility and Flexibility:

The servlet API is designed to be easily extensible. As it stands today the API includes classes that are optimized for HTTP servlets. But later it can be extended and optimized for another type of servlets. It is also possible that its support for HTTP servlets could be further enhanced.

Servlets are also quite flexible, Sun also introduced java server pages. which offer a way to write snippets of servlet code directly with in a static HTML page using syntax similar to Microsoft's Active server pages (ASP)

4.4.3 JDBC

What is JDBC?

any relational database. One can write a single program using the JDBC API, and the JDBC is a Java Api for executing SQL, Statements (As a point of interest JDBC is trademarked name and is not an acronym; nevertheless, Jdbc is often thought of as standing for Java Database Connectivity. It consists of a set of classes and interfaces written in the Java Programming language. JDBC provides a standard API for tool/database developers and makes it possible to write database applications using a pure Java API

Using JDBC, it is easy to send SQL statements to virtually program will be able to send SQL. statements to the appropriate database. The Combination of Java and JDBC lets a programmer writes it once and run it anywhere.

What Does JDBC Do?

Simply put, JDBC makes it possible to do three things

- Establish a connection with a database
- Send SQL statements
- Process the results
- JDBC Driver Types
- The JDBC drivers that we are aware of this time fit into one of four categories
- JDBC-ODBC Bridge plus ODBC driver
- Native-API party-java driver

- JDBC-Net pure java driver
- Native-protocol pure Java driver

An individual database system is accessed via a specific JDBC driver that implements the java. SQL. Driver interface. Drivers exist for nearly all-popular RDBMS systems, through few are available for free. Sun bundles a free JDBC-ODBC bridge driver with the JDK to allow access to a standard ODBC, data sources, such as a Microsoft Access database, Sun advises against using the bridge driver for anything other than development and very limited development.

JDBC drivers are available for most database platforms, from a number of vendors and in a number of different flavours. There are four driver categories

Type 01-JDBC-ODBC Bridge Driver

Type 01 drivers use a bridge technology to connect a java client to an ODBC database service. Sun's JDBC-ODBC bridge is the most common type 01 driver. These drivers implemented using native code.

Type 02-Native-API party-java Driver

Type 02 drivers wrap a thin layer of java around database-specific native code libraries for Oracle databases, the native code libraries might be based on the OCI (Oracle call Interface) libraries, which were originally designed for **C/ C++** programmers, because type- 02 drivers are implemented using native code. in some cases, they have better performance than their all-java counter parts. They add an element of risk, however, because a defect in a driver's native code section can crash the entire server

Type 03-Net-Protocol All-Java Driver

Type 03 drivers communicate via a generic network protocol to a piece of custom middleware. The middleware component might use any type of driver to provide the actual database access. These drivers are all java, which makes them useful for applet deployment and safe for servlet deployment

Type-04-native-protocol All-java Driver

Type o4 drivers are the most direct of the lot. Written entirely in java, Type 04 drivers understand database-specific networking. protocols and can access the database directly without any additional software

JDBC-ODBC Bridge

If possible, use a Pure Java JDBC driver instead of the Bridge and an ODBC driver. This completely eliminates the client configuration required by ODBC. It also eliminates the potential that the Java VM could be corrupted by an error in the native code brought in by the Bridge (that is, the Bridge native library, the ODBC driver manager library, library, the ODBC driver library, and the database client library)

9.2 SOURCE CODE

INDEX CODE

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">

  <title> HYDROHUB
</title>
<!--
Pipeline
http://www.templatemo.com/tm-496-pipeline
-->
<!-- load stylesheets -->
  <link rel="stylesheet" href="//fonts.googleapis.com/css?family=Open+Sans:300,400">  <!--
Google web font "Open Sans", https://fonts.google.com/ -->
  <link rel="stylesheet" href="font-awesome-4.6.3/css/font-awesome.min.css">           <!-- Font
Awesome, http://fontawesome.io/ -->
  <link rel="stylesheet" href="css/bootstrap.min.css">                                <!-- Bootstrap style,
http://v4-alpha.getbootstrap.com/ -->
  <link rel="stylesheet" href="css/magnific-popup.css">                               <!-- Magnific pop up
style, http://dimsemenov.com/plugins/magnific-popup/ -->
  <link rel="stylesheet" href="css/templatemo-style.css">                             <!-- Templatemo
style -->
<!-- HTML5 shim and Respond.js for IE8 support of HTML5 elements and media queries -->
<!-- WARNING: Respond.js doesn't work if you view the page via file:// -->
<!--[if lt IE 9]>
<script src="https://oss.maxcdn.com/html5shiv/3.7.2/html5shiv.min.js"></script>
```

```
<script src="https://oss.maxcdn.com/respond/1.4.2/respond.min.js"></script>
<![endif]-->
</head>
<body>
<div class="container-fluid">

            <section id="welcome" class="tm-content-box tm-banner margin-b-10">
                                                             <div class="tm-banner-inner">
                                                               <h1 class="tm-banner-title">
HYDROHUB: YOUR SOURCE FOR WATER INFORMATION
</h1>
}
                 <tr><th>EMAIL<th><td><input type="text" name="email"
value="<%=email%>"></td></tr>
                 <tr><th>LOCATION<th><td><input type="text" name="location"
value="<%=location%>"></td></tr>
                  <tr><th>AREA<th><td><input type="text" name="area"
value="<%=area%>"></td></tr>
                  <tr><th>FROM DATE<th><td><input type="date" name="from"
required=""></td></tr>
                  <tr><th>TO DATE<th><td><input type="date" name="to"
required=""></td></tr>
                  <tr><th>SESSION<th><td><input type="text" name="session"
required=""></td></tr>
                   <tr><th>HOURS<th><td><input type="text" name="hours"
required=""></td></tr>
                  <tr><th><th><td><input type="submit" value="ADD DETAILS"></td></tr>
</table>
</form>
</center>
```

```
</div>


                    <div id="gallery" class="tm-content-box" style="margin- bottom:500px;">
</div>


</body>
</html>
```

POST ACTION CODE:

```
   String email=request.getParameter("email");
   String location=request.getParameter("location"); String area=request.getParameter("area");
String from=request.getParameter("from"); String to=request.getParameter("to");
   String ses=request.getParameter("session"); String hour=request.getParameter("hours");


   try{
   Connection con=DBCon.getCon();


  Statement pst=con.createStatement(); ResultSet r=null;
   PreparedStatement st=null;


   st=con.prepareStatement("insert into water values(null,?,?,?,?,?,?,?)");


   st.setString(1,email); st.setString(2,location); st.setString(3,area); st.setString(4,from);
   st.setString(5,to); st.setString(6,ses); st.setString(7,hour);
   int i=st.executeUpdate(); if(i>0){
   response.sendRedirect("PostDetails.jsp?msg=success");
  }else{ response.sendRedirect("PostDetails.jsp?msg=failed");
   }
 }catch(Exception e){ out.println(e);
   }


   </html>
```

CHAPTER-10

RESULTS/ DISCUSSIONS

10.1 SYSTEM TESTING

Introduction to Testing:

Testing is a process, which reveals errors in the program. It is the major quality measure employed during software development. During software development. During testing, the program is executed with a set of test cases and the output of the program for the test cases is evaluated to determine if the program is performing as it is expected to perform.

TESTING IN STRATEGIES

In order to make sure that the system does not have errors, the different levels of testing strategies that are applied at differing phases of software development are:

Unit Testing:

Unit Testing is done on individual modules as they are completed and become executable. It is confined only to the designer's requirements.

Each module can be tested using the following two Strategies:

Black Box Testing:

In this strategy some test cases are generated as input conditions that fully execute all functional requirements for the program. This testing has been uses to find errors in the following categories:

- Incorrect or missing functions
- Interface errors
- Errors in data structure or external database access
- Performance errors
- Initialization and termination errors.

In this testing only the output is checked for correctness. The logical flow of the data is not checked.

White Box testing:

In this the test cases are generated on the logic of each module by drawing flow graphs of that module and logical decisions are tested on all the cases. It has been uses to generate the test cases in the following cases:

- ✓ Guarantee that all independent paths have been Executed.
- ✓ Execute all logical decisions on their true and false Sides.
- ✓ Execute all loops at their boundaries and within their operational bounds
- ✓ Execute internal data structures to ensure their validity.

Integrating Testing:

Integration testing ensures that software and subsystems work together a whole. It tests the interface of all the modules to make sure that the modules behave properly when integrated together.

System Testing:

Involves in-house testing of the entire system before delivery to the user. It's aim is to satisfy the user the system meets all requirements of the client's specifications.

Acceptance Testing:

It is a pre-delivery testing in which entire system is tested at client's site on real world data to find errors.

Test Approach:

Testing can be done in two ways:

➤ Bottom-up approach ➤ Top-down approach

Bottom-up Approach:

Testing can be performed starting from smallest and lowest level modules and proceeding one at a time. For each module in bottom up testing a short program executes the module and provides the needed data so that the module is asked to perform the way it will when embed with in the larger system. When bottom level modules are tested attention turns to those on the next level that use the lower-level ones they are tested individually and then linked with the previously examined lower-level modules.

Top-down approach:

This type of testing starts from upper-level modules. Since the detailed activities usually performed in the lower-level routines are not provided stubs are written. A stub is a module shell called by upper-level module and that when reached properly will return a message to the calling module indicating that proper interaction occurred. No attempt is made to verify the correctness of the lower-level module.

Validation:

The system has been tested and implemented successfully and thus ensured that all the requirements as listed in the software requirements specification are completely fulfilled. In case of erroneous input corresponding error messages are displayed.

10.2 TEST CASES:

Test case1:

Test case for Login form:

FUNCTION:	LOGIN
EXPECTED RESULTS:	Should Validate the user and check his existence in database
ACTUAL RESULTS:	Validate the user and checking the user against the database
LOW PRIORITY	No
HIGH PRIORITY	Yes

Test case2:

Test case for User Registration form:

FUNCTION:	USER REGISTRATION
EXPECTED RESULTS:	Should check if all the fields are filled by the user and saving the user to database.
ACTUAL RESULTS:	Checking whether all the fields are field by user or not through validations and saving user.
LOW PRIORITY	No
HIGH PRIORITY	Yes

Test case3:

Test case for Change Password:

When the old password does not match with the new password, then this results in displaying anerror message as “OLD PASSWORD DOES NOT MATCH WITH THE NEW PASSWORD”.

Test case 4:

Test case for Forget Password:

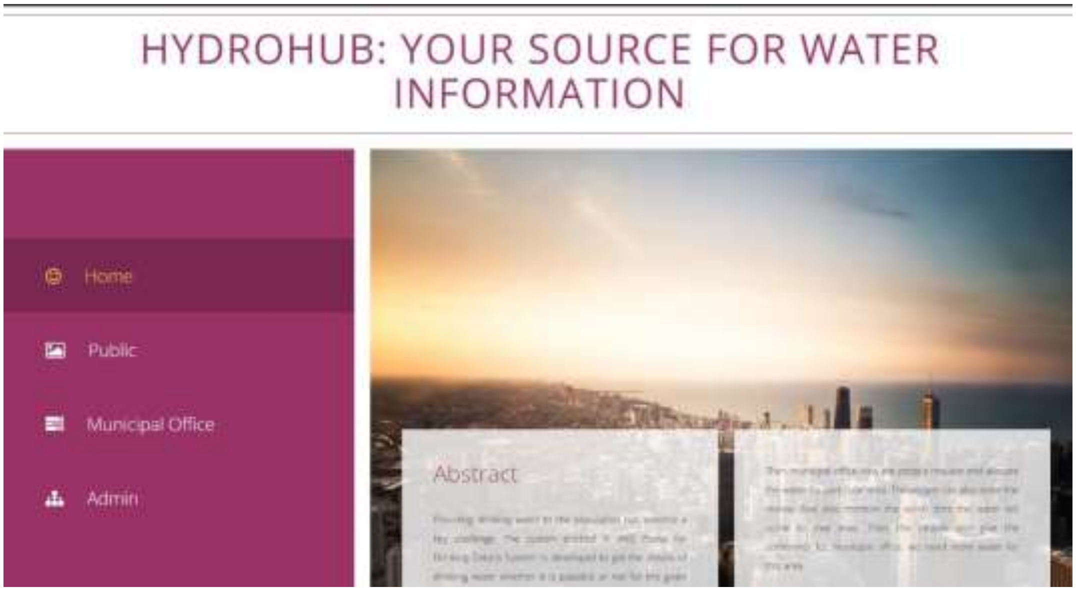
When a user forgets his password, he is asked to enter Login name, ZIP code, Mobile number. If these are matched with the already stored ones then user will get his original password.

Module	Functionality	Test Case	Expected Results	Actual Results	Resut	Prioriy
User	Login Use case	1. Navigate To Www.Sample. Com 2. Click On Submit Button Without Entering Username and Password	A Validation Should Be as Below “Please Enter Valid Username & Password”	A Validation Has Been Populated as Expected	Pass	High

		1. Navigate To Www.Sample. Com 2. Click On Submit Button with Out Filling Password AndWith Valid Username	A Validation Should Be as Below "Please Enter Valid Password or Password Field CanNot Be Empty "	A Validation Is Shown as Expected	Pass	High
		1. Navigate To Www.Sample. Com 2. Enter Both Username and Password Wrong And Hit Enter	A Validation Shown as Below "The Username Entered Is Wrong"	A Validation Is Shown as Expected	Pass	High
		1. Navigate To Www.Sample. Com 2. Enter Validate Username and Password And Click On Submit	Validate Usernameand Password In Database And Once If They Correct Then ShowThe Main Page	Main Page/ Home PageHas Been Displayed	Pass	High

10.3 SCREENSHOTS

HOME SCREEN



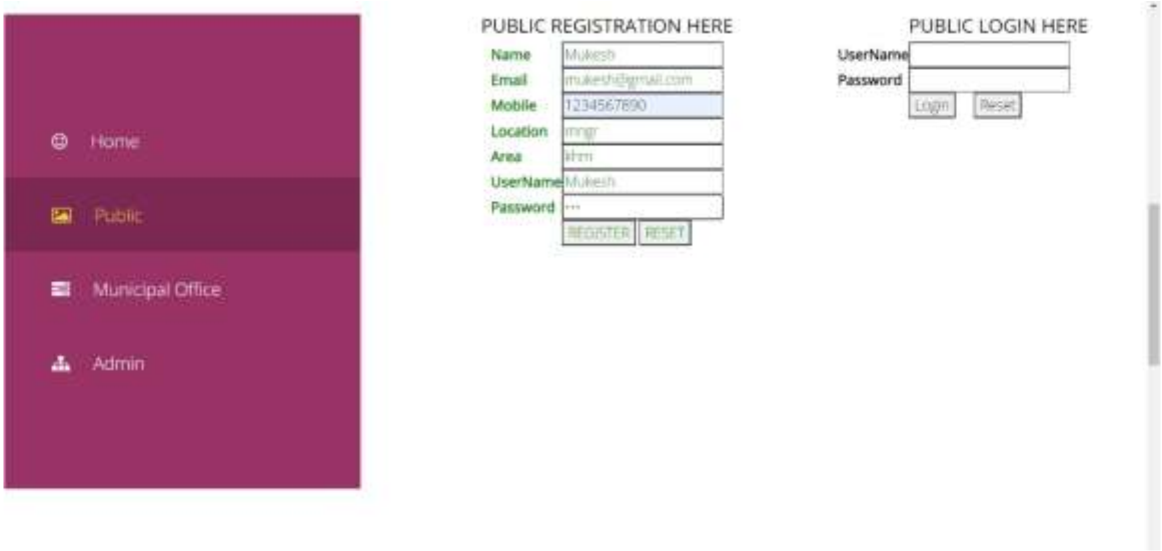
HYDROHUB: YOUR SOURCE FOR WATER INFORMATION

Abstract

Providing drinking water to the population has become a key challenge. The system entitled [illegible] Drinking Details System is developed to get the details of drinking water whether it is possible or not for the given

Then municipal office they are going to measure and allocate the water to [illegible] area. The [illegible] can also enter the [illegible] also mention the [illegible] time the water will come to their area. Then the people can give the complaint to municipal office and need more water for this area.

USER REGISTRATION

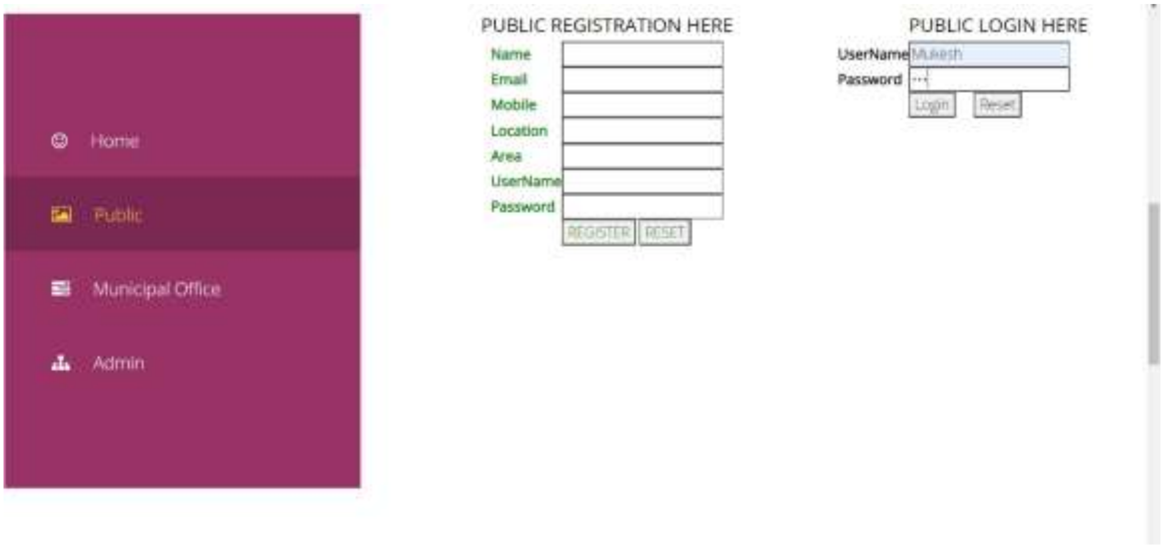


PUBLIC REGISTRATION HERE

Name Mukesh
Email mukesh@gmail.com
Mobile 1234567890
Location mngr
Area khm
UserName Mukesh
Password ···

PUBLIC LOGIN HERE

UserName
Password



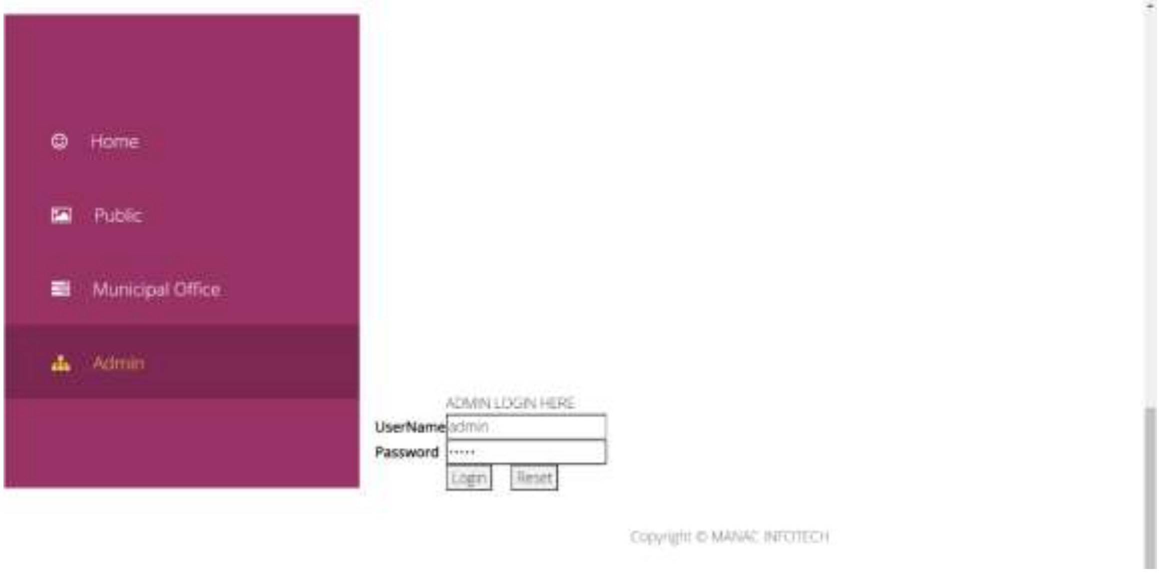
PUBLIC REGISTRATION HERE

Name
Email
Mobile
Location
Area
UserName
Password

PUBLIC LOGIN HERE

UserName Mukesh
Password ···

Admin login



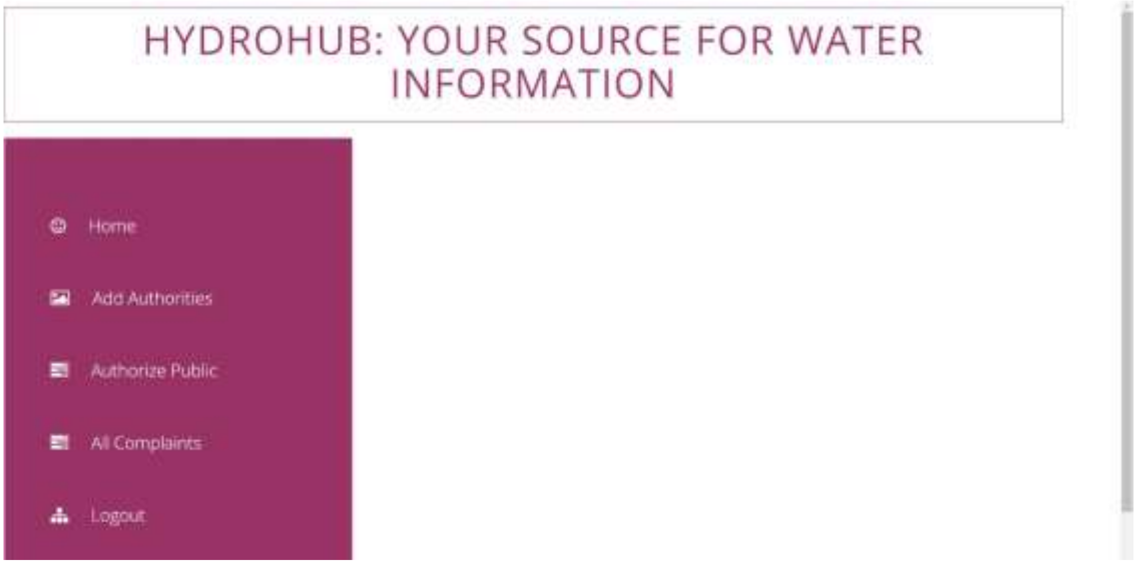
ADMIN LOGIN HERE

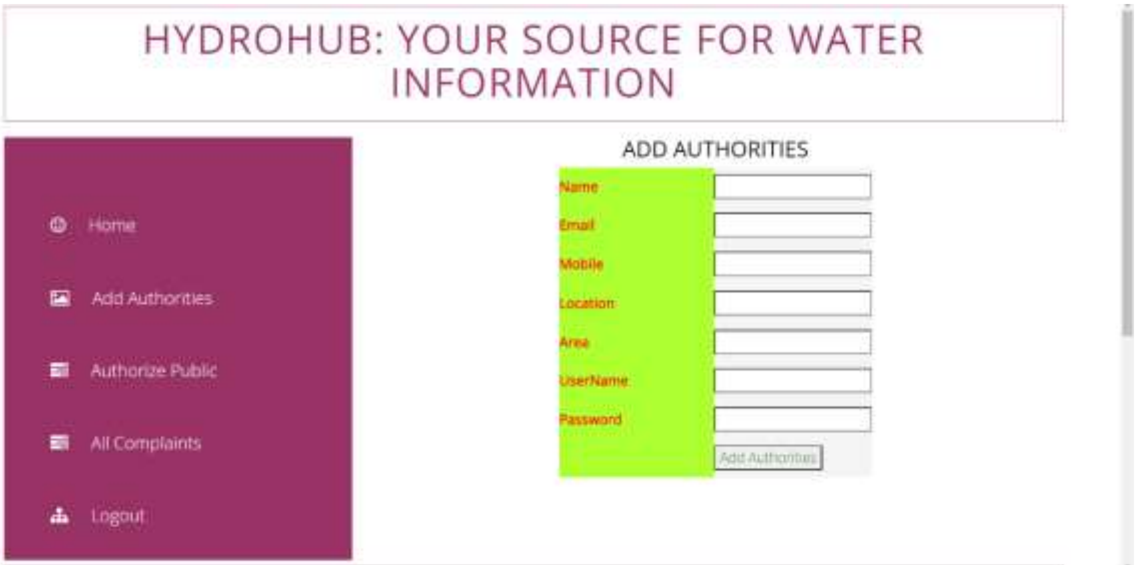
UserName admin

Password ·····

Copyright © MANAC INFOTECH

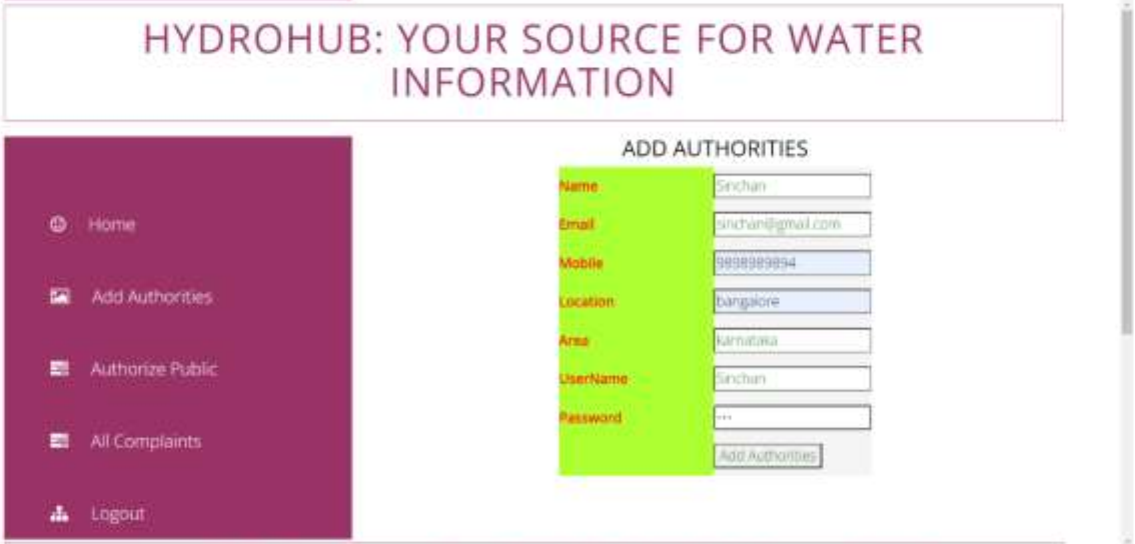
Admin Home Screen





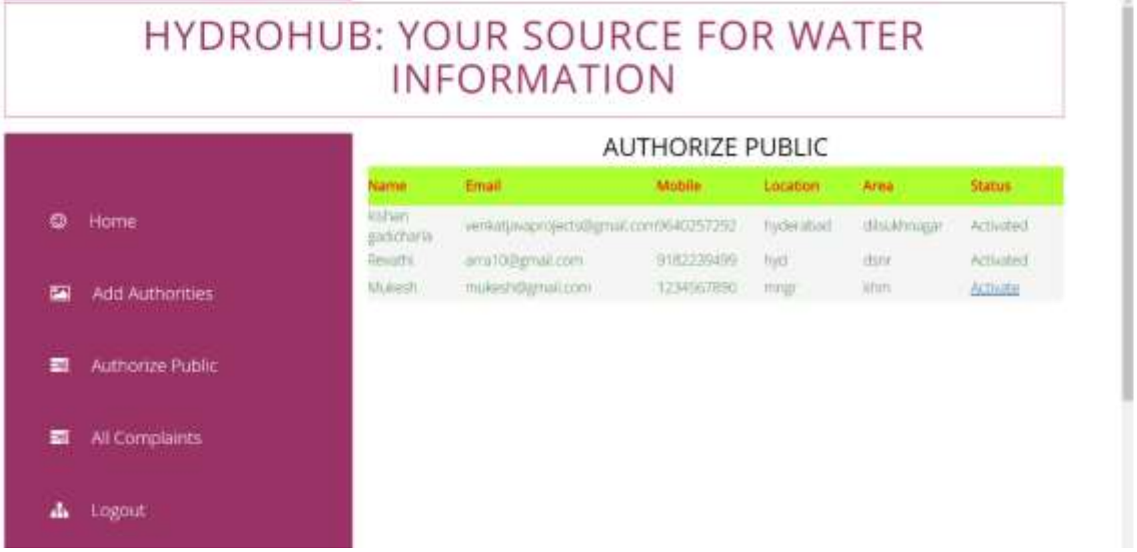
ADD AUTHORITIES

Name	
Email	
Mobile	
Location	
Area	
UserName	
Password	



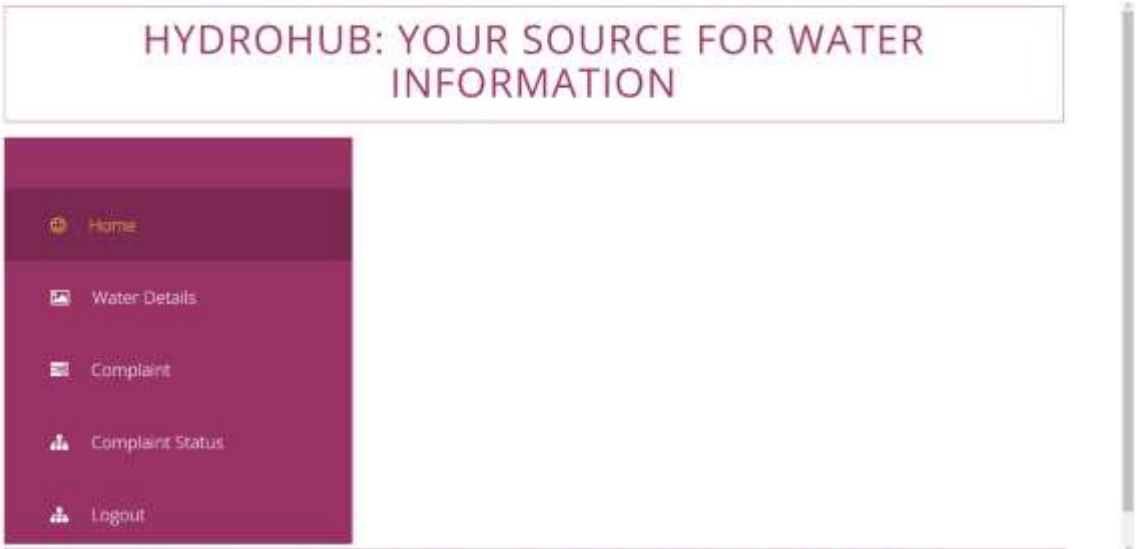
ADD AUTHORITIES

Name	Sinchan
Email	sinchan@gmail.com
Mobile	9898989894
Location	bangalore
Area	karnataka
UserName	Sinchan
Password	•••

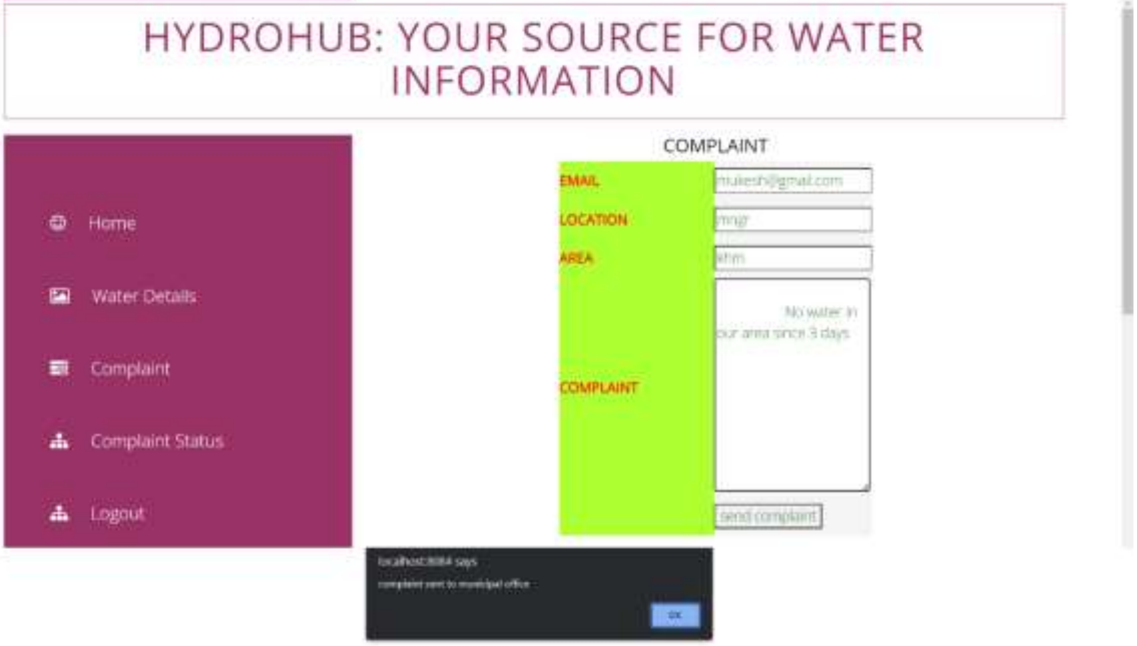


AUTHORIZE PUBLIC

Name	Email	Mobile	Location	Area	Status
kishan gadicharla	venkatjavaprojects@gmail.com	9640257252	hyderabad	dilsukhnagar	Activated
Revathi	arra10@gmail.com	9182239499	hyd	dsnr	Activated
Mukesh	mukesh@gmail.com	1234567890	mngp	khm	Activate



HYDROHUB: YOUR SOURCE FOR WATER INFORMATION



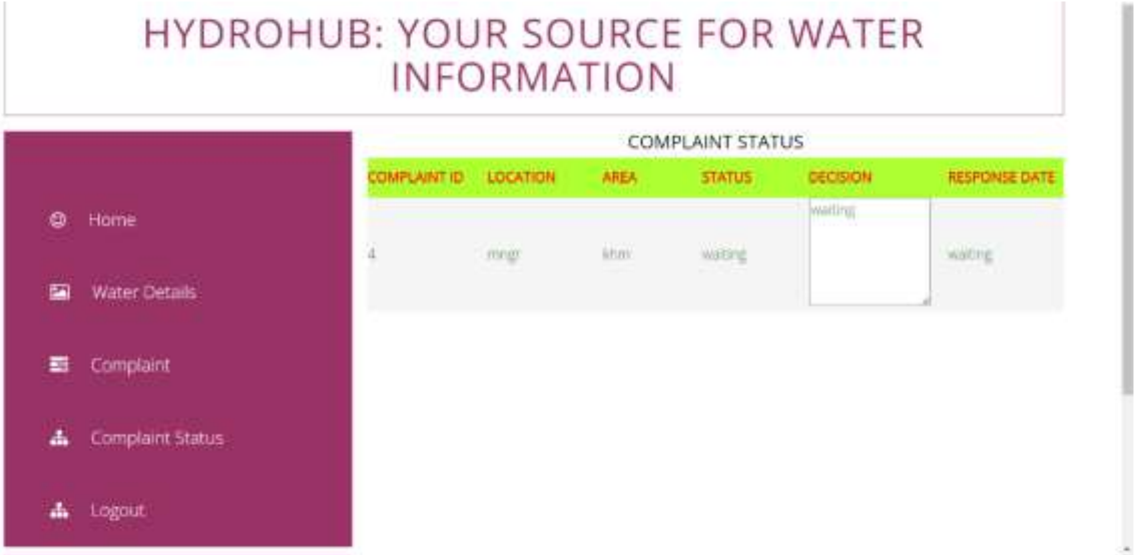
HYDROHUB: YOUR SOURCE FOR WATER INFORMATION

COMPLAINT

EMAIL	mukesh@gmail.com
LOCATION	mngr
AREA	khm
COMPLAINT	No water in our area since 3 days

localhost:8084 says

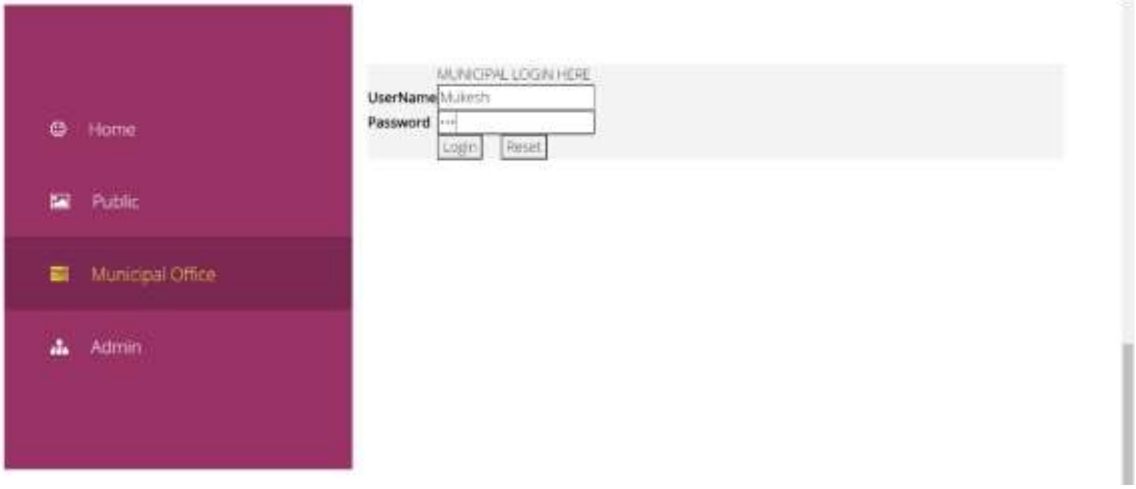
complaint sent to municipal office



HYDROHUB: YOUR SOURCE FOR WATER INFORMATION

COMPLAINT STATUS

COMPLAINT ID	LOCATION	AREA	STATUS	DECISION	RESPONSE DATE
4	mngr	khm	waiting	waiting	waiting



MUNICIPAL LOGIN HERE

UserName Mukesh

Password •••

CHAPTER-11

CONCLUSION

A web portal for safe drinking water is an essential resource for ensuring the well-being and health of communities around the world. In conclusion, such a portal plays a crucial role in addressing various aspects related to safe drinking water, including information dissemination, access to critical resources, and raising awareness. It serves as a centralized platform for:

1. Information Dissemination: Providing comprehensive information on water quality standards, water treatment processes, and water-related regulations to educate the public and empower them with knowledge about safe drinking water.

2. Access to Resources: Offering access to resources such as water quality reports, testing kits, and contact information for relevant authorities, enabling individuals and communities to monitor and address water quality concerns effectively.

3. Community Engagement: Facilitating community engagement through forums, discussion boards, and educational materials to foster a sense of shared responsibility for safe drinking water and encourage collaboration on water-related issues.

4. Emergency Response: Providing real-time updates on water quality in case of emergencies, natural disasters, or contamination incidents, helping communities take immediate action to safeguard their health.

In summary, a web portal for safe drinking water is a vital tool in the effort to ensure that people have access to clean and safe water. It promotes transparency, awareness, and community involvement, all of which are crucial in maintaining and improving the quality of drinking water, thereby contributing to the health and well-being of individuals and communities.

CHAPTER-12

REFERENCES

1. Gleick, Peter H. "Water in Crisis: A Guide to the World's Fresh Water Resources." Oxford University Press, 1993.
2. United Nations. "World Water Development Report." UNESCO, 2020. [Available online at: https://www.unwater.org/publications/world-water-development-report/]
3. National Geographic. "Water: Our Thirsty World." National Geographic Society, 2010.
4. World Health Organization (WHO) and UNICEF. "Progress on Drinking Water, Sanitation and Hygiene: 2017 Update and SDG Baselines." World Health Organization, 2017. [Available online at: https://www.who.int/teams/water-sanitation-hygiene-and-health/publications/jmp-2017-report]
5. United States Environmental Protection Agency (EPA). "Drinking Water Contaminants." EPA, 2022. [Available online at: https://www.epa.gov/ground-water-and-drinking-water/national-primary-drinking-water-regulations]
6. UNESCO. "World Water Assessment Programme." UNESCO, 2022. [Available online at: http://www.unwater.org/world-water-assessment-programme/]
7. The World Bank. "World Development Report 2021: Data for Better Lives." The World Bank, 2021.
8. National Aeronautics and Space Administration (NASA). "Global Water Cycle: NASA's Eyes on the Earth." NASA, 2022. [Available online at: https://earthobservatory.nasa.gov/features/WaterCycle]
9. Food and Agriculture Organization (FAO). "The State of Food and Agriculture 2020." FAO, 2020. [Available online at: http://www.fao.org/state-of-food-agriculture/en/]
10. International Water Association (IWA). "WaterWiki: An Open Access Platform for Water Knowledge." IWA, 2022. [Available online at: https://www.iwapublishing.com/waterwiki]