



**DEPARTMENT OF
MASTER OF COMPUTER APPLICATIONS**



SOFTWARE ENGINEERING Assignment

18MCA21

*Submitted in partial fulfillment of the requirements for the award
of degree of*

MASTER OF COMPUTER APPLICATIONS

By

Ambika Badiger

USN: 1RV19MCA06

Under the Guidance of

Dr. S.S. Nagamuthu Krishnan

Assistant Professor

2019-2020

RV COLLEGE OF ENGINEERING[®],

(Autonomous Institution Affiliated to Visvesvaraya Technological University, Belagavi)

**DEPARTMENT OF
MASTER OF COMPUTER APPLICATIONS**

Bengaluru– 560059



CERTIFICATE

Certified that the Assignment titled "**POLLUTION ANALYSIS AND CONTROL**" carried out by **SNEHAL MAHADEV HUKKERI**, USN : **1RD19MCA09**, a bonafide student of RV College of Engineering, Bengaluru submitted in partial fulfilment for the award of Master of Computer Applications of RV College of Engineering, Bengaluru affiliated to Visvesvaraya Technological University, Belagavi during the year 2019-20. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The report has been approved as it satisfies the partial academic requirement in respect of the course Software Engineering 18MCA21.

Dr.S.S.Nagamuthu Krishnan
Assistant Professor
Department of MCA
RVCE, Bengaluru –59

POLLUTION ANALYSIS AND CONTROL

Prepared by

Ambika Badiger [1RV19MCA06]

Anusha J [1RV19MCA10]

Snehal Hukkeri [1RD19MCA09]

Abstract

Pollution is a word that we all aware of these days. Pollution is an undesirable change in the physical, chemical or biological characteristics of air, water and soil that may harmfully affect the life or create potential health hazard of any living organism. Pollution is thus direct or indirect change in any component of the biosphere that is harmful to the living components and in particular undesirable for man, affecting adversely the industrial progress, cultural and natural assets or general environment of living society. The scope of existing Environmental Health Monitoring Apps is limited to provide data of extent of pollution and corresponding reasons for pollution. The scope of existing apps can be extended to analyzing and resolving pollution related problems.

Methodology of project includes requirement specification, designing implementation and testing. The main objective of the application is to analyze and resolve pollution related problems. The tools used are NETBEANS 8.1, Java programming language toolkit JDK1.8, Java Runtime Environment, MySQL connector java 5.0, MySQL Tools for Server 5.1 and used Apache Tomcat 7.0.server.

This application enables the user to be aware about environmental condition in his locality and provides required measures for improvement of polluted environment. Helps the user to take precautionary actions to prevent the pollution.

Table of Contents

<i>Contents</i>	<i>Page No</i>
Chapter 1: Introduction	
Project Description	6
Existing and Proposed System	7
Tools and Technologies used	10
Hardware and Software Requirements	10
Chapter 2: Software Requirement Specifications	
Introduction	11
General Description	12
Functional Requirement	13
External Interfaces Requirements	16
Non Functional Requirements	17
Chapter 3: System Design	
System Perspective /Architectural Design	19
Context Diagram	27
Chapter 4: Detailed Design	
4.1 System Design	28
Chapter 5 Implementation	
Code Snippets	38
Implementation	38
Chapter 6: Software Testing	
Test cases	46
Testing and Validations	47
Chapter 7: Conclusion & Future Enhancements	
	92
Bibliography	
	93

1.

Introduction

Project Description

Basic Introduction of the Project

This application is to analyze and resolve pollution related problem. This is two tier application. Here user will register to the application where user can be particular person or an organization, user report the problem such as air, water, noise and soil pollutions. The user can also provide innovative ideas to solve the problem. The ideas can be related to solving the problem report or to provide social awareness to the people.

The application helps to analyze the reported problem and by analyzing the reported problem appropriate event will be planned and organized by considering the ideas provided by other users to solve that particular problem. Along with this few events are conducted in order to create awareness among people regarding various types of pollution, their effect on biodiversity and ecosystem and their control measures.

The event details are provided by application which consist description of event such as problem for which the event is organized, place where the event is organized, number of volunteers, budget required to conduct that particular event.

While proceeding the event user can participate as volunteer or sponsor or both, sponsor can sponsor any material or money for the event and volunteer works for event to make an event successful.

By following these methods the application is trying to solve the pollution related problem and trying to provide social awesomeness among people.

Concept relevant to the project

Abstraction-

Data abstraction is the process of hiding certain details and showing only essential information to the user.

Encapsulation-

Encapsulation in Java is a mechanism of wrapping the data (variables) and code acting on the data (methods) together as a single unit.

Polymorphism-

Polymorphism in Java is a concept by which we can perform a single action in different waysSo polymorphism means many forms. There are two types of polymorphism in Java: compile-time polymorphism and runtime polymorphism. We can perform polymorphism in java by method overloading and method overriding.

Inheritance-

It is the mechanism in java by which one class is allow to inherit the features (fields and methods) of another class.

Existing and Proposed System

Problem statement and Scope of the project

The application helps to analyze the reported problem and by analyzing the reported problem appropriate event will be planned and organized by considering the ideas provided by other users to solve that particular problem. Along with this few events are conducted in order to create awareness among people regarding various types of pollution, their effect on biodiversity and ecosystem and their control measures

The application is trying to solve the pollution related problem and trying to provide social awesomeness among people. As it collects the problems regarding various pollution types such as air pollution, soil pollution, water pollution etc.in various areas, reported by user. Based on these problems Admins plan a appropriate event and organize in the area it was reported sometimes based on ideas provided by other users. These events are managed by one of the admins and by conducting events they try to solve reported problem. And conduct some other events to provide social awareness regarding prevention and controls of pollutions to people.

Methodology adopted in the proposed system

Methodology of project includes requirement specification, designing implementation and testing.

The spiral model is similar to the incremental development for a system, with more emphasis placed on risk analysis. The spiral model has four phases: Planning, Design, Construct and Evaluation. A software project repeatedly passes through these phases in iterations called Spirals.

Planning: According to spiral model at first, we began with requirements collection. Requirements were gathered by visiting various websites to know the current need for user, which provides offline guidance for shopping. Later requirements were gathered by consulting various shop owners to get information about emerging trends in various places which can help user in shopping

Design:Later based on these requirements designs were developed. Design phase starts with the design in the baseline spiral and involves architectural, logical design of modules, physical product design and final design in the successive

spirals. Designs such as Activity diagrams, Usecase diagrams, ER diagrams, DFD's, Database designs.

Construct: According to designs developed coding was done. Construct phase refers to development of the final software at every spiral. A Proof of Concept (POC) is developed in this phase to get the users' feedback.

After coding Testing was conducted to test each phase in order to check the correct working of each module.

Likewise, each module is developed separately. Finally developed modules are integrated with other modules

Evaluation and Risk Analysis: Risk analysis includes identifying, estimating, and observing technical feasibility such as schedule slippage and cost overrun.

After testing the build, at the end of first iteration, user evaluates the software and provides the feedback. Based on the customer assessment, development process enters into the next iteration and afterwards follows the linear approach to implement the feedback provided by the user. The process of iterations along the spiral carries on with throughout the life of the software.

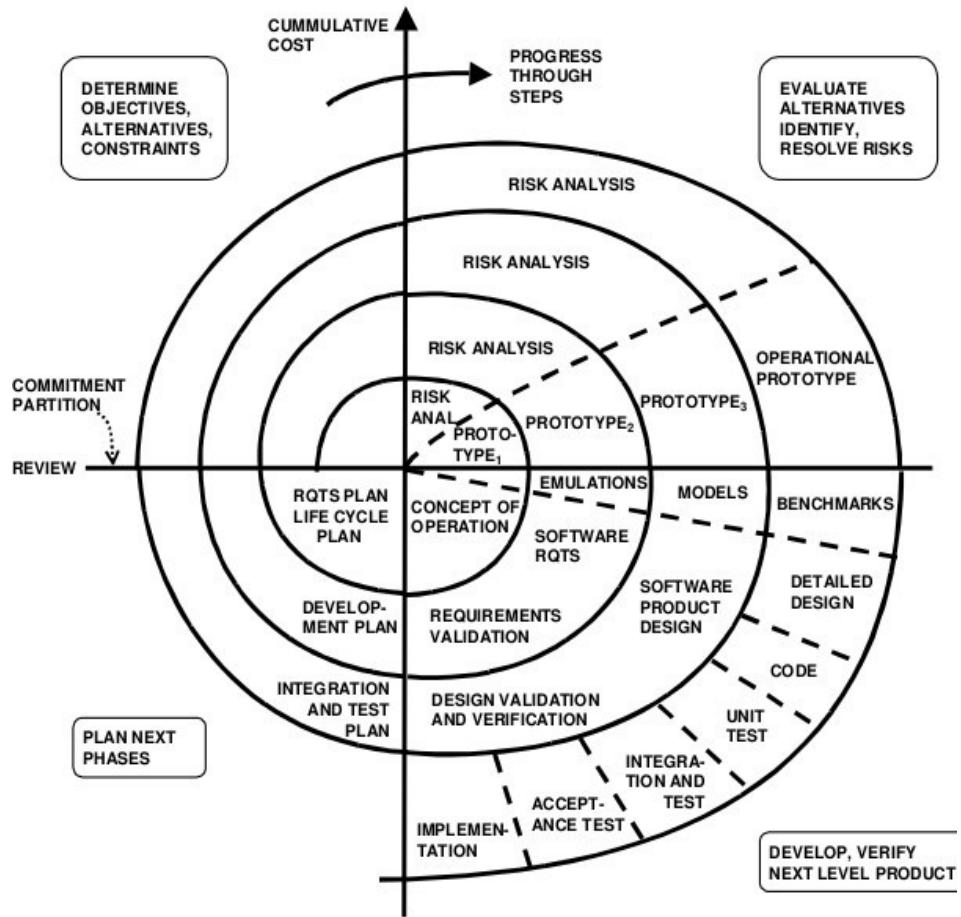


Figure 1: Spiral Model

Technical Features of the proposed system

The tools used are NETBEANS 8.1, Java programming language toolkit JDK1.8, Java Runtime Environment, MySQL connector java 5.0, MySQL Tools for Server 5.1 and used Apache Tomcat 7.0.server.

As have used NETBEANS, our application is platform independent you can run it on Windows, macOS, Linus and Solaris.

Tools and Technologies used

Platform / Tools used in implementing the project

The tools used are NETBEANS 8.1, Java programming language toolkit JDK1.8, Java Runtime Environment, MySQL connector java 5.0, MySQL Tools for Server 5.1 and used Apache Tomcat 7.0.server.We have used Windows platform.

1.4. Hardware and Software Requirements

The application requires computer system with Intel / AMD 64 bit processor, minimum 2.45 MB RAM and 2.93 MB disk space and it uses other hardware devices like mouse and keyboard as positioning devices and monitor for showing the important information and this whole system interact with the internet server for getting and changing information online from anywhere if you are accessing from personal computer.

The application needs NETBEANS 8.1, Java programming language toolkit JDK1.8, Java Runtime Environment, MySQL connector java 5.0, MySQL Tools for Server 5.1.Apache Tomcat 7.0.server.

2. Software Requirement Specifications

Introduction

The application helps to analyze the reported problem and by analyzing the reported problem appropriate event will be planned and organized by considering the ideas provided by other users to solve that particular problem. Along with this few events are conducted in order to create awareness among people regarding various types of pollution, their effect on biodiversity and ecosystem and their control measures.

Purpose

This application is to analyze and resolve pollution related problem. This is two tier application. Here user will register to the application where user can be particular person or an organization , user report the problem such as air, water, noise and soil pollutions. The user can also provide innovative ideas to solve the problem. The ideas can be related to solving the problem report or to provide social awareness to the people.

Document Conventions

The event details are provided by application which consist description of event such as problem for which the event is organized, place where the event is organized , number of volunteers ,budget required to conduct that particular event. While proceeding the event user can participate as volunteer or sponsor or both, sponsorer can sponsor any material or money for the event and volunteer works for event to make an event successful.

Intended Audience and Reading Suggestions

The Intended Audience to the application are users who play the various roles such as volunteer, sponsorer along with reporting problems regarding particular pollution in particular area and also provide idea to solve the problem based on particular pollution types. Speaking of the admins, they plan and organize event based on problems reported by user regarded to particular pollution type one among the admins manage the event and look after the needs of the event.

Project Scope

The application is trying to solve the pollution related problem and trying to provide social awesomeness among people. As it collects the problems regarding various pollution types such as air pollution, soil pollution, water pollution etc.in various areas ,reported by user. Based on these problems Admins plan a appropriate event and organize in the area it was reported sometimes based on ideas provided by other users. These events are managed by one of the admins and by conducting events they try to solve reported problem. And conduct some other events to provide social awareness regarding prevention and controls of pollutions to people.

General Description

Product Perspective

The main perspective of the application is to solve the pollution related problem and trying to provide social awareness among people. By collects the problems regarding various pollution types such as air pollution, soil pollution, water pollution etc.. in various areas. These problems are reported by user registered to the system. By analyzing these problems Admins plan a appropriate event and organize in the area it was reported sometimes based on ideas provided by other users to solve the particular problem. These events are managed by on of the admins and by conducting events they try to solve reported problem. And conduct some other events to provide social awareness regarding prevention and controls of pollutions to people. Here users also act as volunteer and sponsors, where volunteers participate in event to help solve the problems and sponsor sponsors money or material for event that is organized.

Product Functions

The important feature of the application is that it helps resolve pollution related problems, which are reported by users. Solution of the problem will in form an event which will be planned and organized by admins by analyzing the problem reported .

User Characteristics

There are mainly two users of this application one is users and admin . Where users is th one who report problem on various pollution types and play other roles such as volunteer, sponsor and also provide idea to solve the problem based on particular pollution types. The admins plan and organize event based on problems reported by user regarded to particular pollution type one among the admins manage the event and look after the needs of the event.

Operating Environment

The application requires computer system with Intel / AMD 64 bit processor, minimum 2.45 MB RAM and 2.93 MB disk space and it uses other hardware devices like mouse and keyboard as hardware. And NETBEANS 8.1, Java programming language toolkit JDK1.8, Java Runtime Environment, MySQL connector java 5.0, MySQL Tools for Server 5.1 as software specification. We have used Apache Tomcat 7.0.server.

User Documentation

Users are imported modules of the the application , as they play various roles such that report the problems , provide specific idea regarding specific pollution type, volunteer for the event and help in conduction of event and sponsor money or material that would be required for event. This information of various roles of user are known to him once he register to the application.

Assumptions and Dependencies

The application requires internet connection to collect information from users who are at different places. The application also have information dependency for collecting information of status of all the events that are conducted at different places.

Functional Requirement

Introduction

Once the user has registered, the application allows user to login using username and password. Application allows user to report pollution related problems and share ideas to solve them. Based on the pollution related problems, application allows admin to organize event to solve particular problem. It also allows the admin to give details about the outcome of the event. User might give feedback about the event.

Input,Processing and Output

The functional requirements defines functions of a system or its components. Function is described as a set of inputs, the behavior and outputs.

- i. **Registration:** The user register himself to the application. First the user will enter his/her information like name, address, contact number, email id, username and password in the registration form, the form is then validated for correctness of the data at client side and then sent to the server, once it sent to the server and stored in the database user will get confirmation as registration successful.
- ii. **Login:** After user registration , the user who is registered clicks on the login , fills the data like username and password in the form and click on login button. The submitted form is then validated with the database data. If the username and password is matching, then the user is allowed to go to the next page. If the user credentials are not matching, then error message is displayed.
- iii. **Reporting pollution related problems:** The registered user, after login can reports pollution related problems by filling the details of the pollution like pollution type, pollution area, pollution description in the

report problem form, the details are then stored in the database and user will confirmation as Problem reported successfully.

- iv. **.Sharing Idea :** Registered user after login can share ideas to solve pollution related problem by filling details like pollution type,description in the IDEA form and then these details are stored in the database and the user will get confirmation as idea submitted successfully.
- v. **.Event Details:** To solve problems reported by users, admin will login ,views user reported problems and organize event by filling details like event name,pollution type,event place, event date in the EVENT FORM and then event details are stored in the database.
- vi. **Requirements :** Once the event is planned by the admin to solve particular pollution related problem ,he/she will specify needs/requirements of the event to conduct it successfully ,by filling details like required money,required materials and required volunteers in the REQUIREMT FORM and then details are stored in the database.
- vii. **Outcome:** Once the event is conducted successfully the outcome of the event will be updated by admin by filling details like Event status and description in the OUTCOME FORM then details are stored in the database.
- viii. **Feedback:** Registered user after login can give feedback about the event by filling the details like event id,pollution type,description in FEEDBACK FORM then details are stored in the database.

External Interfaces Requirements

User Interfaces:

First interface includes the login option for the user and it asks the user about username and password, If not a valid user the system prompts by displaying error message that "Not a valid user". And if he/she is a valid user then the main page of the application appears where various options are provided. Where user can report problems, share ideas, see event details and give feedback.

Hardware Interfaces:

The application requires computer system with Intel / AMD 64 bit processor, minimum 2.45 MB RAM and 2.93 MB disk space and it uses other hardware devices like mouse and keyboard as positioning devices and monitor for showing the important information and this whole system interact with the internet server for getting and changing information online from anywhere if you are accessing from personal computer.

Software Interfaces:

NETBEANS 8.1, Java programming language toolkit JDK1.8, Java Runtime Environment, MySQL connector java 5.0, MySQL Tools for Server 5.1..We have used Apache Tomcat 7.0.server.

Communications Interfaces:

As we have used JAVA programming language, java object is a combination of data and procedures working on the available data. An object has a state and behavior. The state of an object is stored in fields (variables) while methods (functions) displays the object's behavior.

Communication between objects and classes: Passing object as parameter to a method of an object that belongs to a different class type, this way you pass attribute of an object to another object of a different class just call an methods of object of another class.so you can create an object of this class to get information of other object of different class.

This is how different components communicates in our application.

Non Functional Requirements

- i. **Speed:** Our application responds quickly it will take maximum 50 sec to load each page.

2 aspects of speed are bandwidth and access latency an average 45 seconds.

Sustained bandwidth: Average data rate during a large transfer.

Bytes/transfer time .i.e. bytes per second.

-Data rate when the data stream is actually flowing.

Effective Bandwidth: Average over the entire I/O time.

Access latency: Amount of time needed to locate data. Latency is the delay from the input into a system to a desired outcome.

- ii. **Security:** Every user is provided with a particular fixed set of user privileges.In our application user credentials are, user can view his details,he can edit his details,he can report pollution related problems,he can sponsor,he can volunteer event and he can give feedback but he can't see admin information. Admin credentials are, Admin can view user information but he can't edit it. Admin can view all the information about problems,events,feedbacks.
- iii. **Reliability:** Our application can perform a failure free operation in a netbeans environment using apache server and MYSQL database. Our application assures that it is fault free because we have handled all possible exceptions using exception handling mechanism of JAVA and is reliable for its intended purpose, i.e.solving pollution related problems effectivly.
- iv. **Software Quality:** system is adaptable any file containing information of site details can be inserted to the database. It's availability is high, it

depends only on the maximum number of internet users allowed, it's correctness is high. It also has some more quality attributes like reliability, reusability, testability, flexibility and usability.

- v. **Ease of use:** Our application is user friendly, as its GUI (Graphical user interface) is not so complicated for user to use our application . All required instructions are given in the application. User with basic knowledge of accessing Internet can use our application easily. Using user feedback we are improving user experience.
- vi. **Portability:** Our application is portable and can be run on Windows, MacOS and Linux platforms easily . As we have developed this application using NetBeans , open source IDE and java programming language , it can run across different platforms and is easy to use.

3. System Design (High level or Architectural Design)

System Perspective /Architectural Design

Problem Specification

This application is to analyze and resolve pollution related problem. This is two tier application. Here user will register to the application where user can be particular person or an organization, user report the problem such as air, water, noise and soil pollutions. The user can also provide innovative ideas to solve the problem. The ideas can be related to solving the problem report or to provide social awareness to the people.

The application helps to analyze the reported problem and by analyzing the reported problem appropriate event will be planned and organized by considering the ideas provided by other users to solve that particular problem. Along with this few events are conducted in order to create awareness among people regarding various types of pollution, their effect on biodiversity and ecosystem and their control measures.

Block Diagram

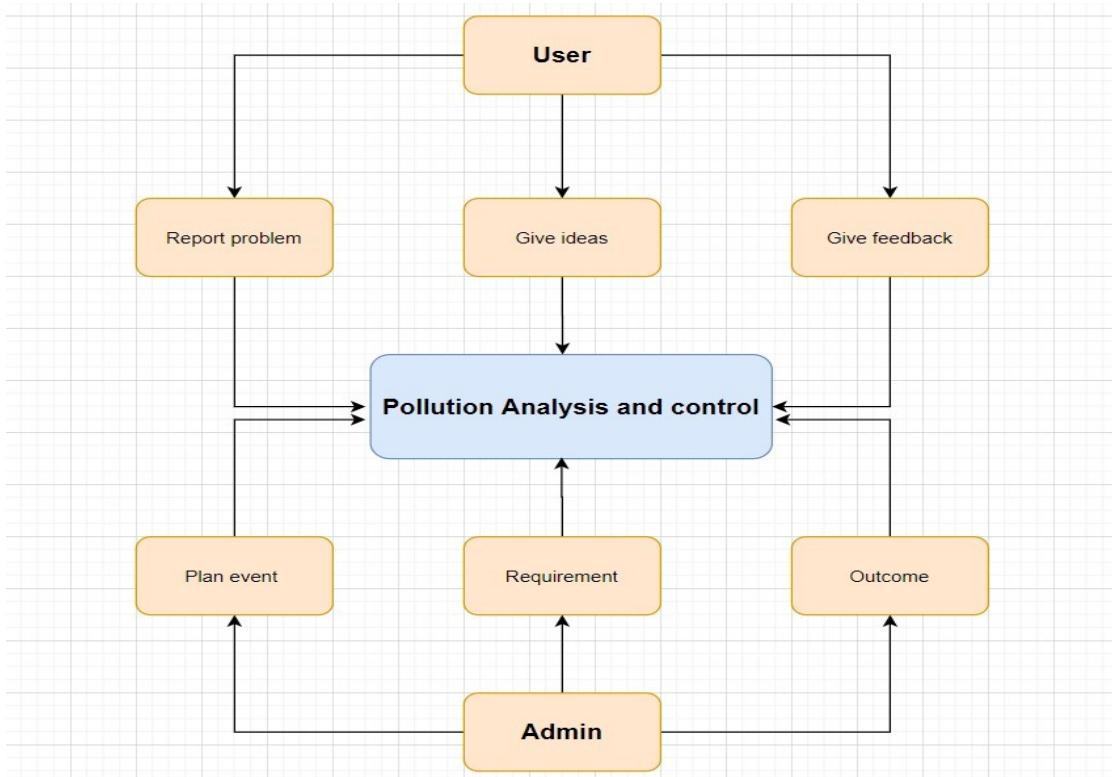


Figure 2:Block Diagram

Data definition/Dictionary (Data base details)

Admin table : Stored information about admin description given below.

The screenshot shows the NetBeans IDE interface with the following details:

- Toolbar:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Connection:** jdbc:mysql://localhost:3306/pollution?zeroDateTimeBehavior=convertToNull [root on Default schema]
- Code Editor:** SQL statement: `SELECT * FROM `admin`;`
- Database Navigator:** Shows the database structure with tables like admin, event, feedback, idea, implement, outcome, problem, require, sponsor, users, and volunteer.
- Table Definition:** A modal window titled "Desc admin" displays the table structure:

#	Field	Type	Null	Key	Default	Extra
1	a_id	int(10)	NO	PRI	<NULL>	auto_increment
2	a_name	char(50)	YES		<NULL>	
3	a_username	varchar(10)	YES		<NULL>	
4	a_password	varchar(10)	YES		<NULL>	
- Output:** SQL 42 execution output: "Activate Windows Go to Settings to activate Windows."
- System Bar:** Shows the date and time as 7/10/2020 12:26 AM.

User table : Stored user information description given below

The screenshot shows the NetBeans IDE interface with the following details:

- Toolbar:** File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help.
- Connection:** jdbc:mysql://localhost:3306/pollution?zeroDateTimeBehavior=convertToNull [root on Default schema]
- Code Editor:** SQL statement: `SELECT * FROM users;`
- Database Navigator:** Shows the database structure with tables like admin, event, feedback, idea, implement, outcome, problem, require, sponsor, users, and volunteer.
- Table Definition:** A modal window titled "desc users" displays the table structure:

#	Field	Type	Null	Key	Default	Extra
1	u_id	int(10)	NO	PRI	<NULL>	auto_increment
2	u_name	char(50)	YES		<NULL>	
3	u_phone_no	bigint(20) unsigned	YES		<NULL>	
4	u_email	varchar(50)	YES		<NULL>	
5	u_address	varchar(50)	YES		<NULL>	
6	u_username	varchar(10)	YES		<NULL>	
7	u_password	varchar(10)	YES		<NULL>	
- Output:** SQL 42 execution output: "Activate Windows Go to Settings to activate Windows."
- System Bar:** Shows the date and time as 7/10/2020 12:33 AM.

Problems table: Stores problem information reported by each user.

The screenshot shows the NetBeans IDE interface with the following details:

- File Menu:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Toolbar:** Standard Java development toolbar.
- Central Area:**
 - Services X:** Shows available databases.
 - Databases:** MySQL Server at loc...
 - Projects:** Shows Java DB, Drivers, jdbcderby://localhost, and jdbcmysql://localhost.
 - Files:** Shows pollution database structure:
 - Tables:** admin, event, feedbac, idea, impleme, outcome, problem (selected).
 - Views:**
 - Procedures:**
 - Other database:**
 - Output:** Shows multiple execution tabs: SQL 42 execution, SQL 43 execution, SQL 44 execution, SQL 45 execution, SQL 46 execution, SQL 47 execution, and SQL 48 execution.
- Bottom:** Taskbar with search bar, system icons, and system status (Windows 10, ENG, IN, 12:15 AM, 7/10/2020).

In the central area, the "problem" table is selected in the Navigator. The "desc problems" command is run, and the resulting table structure is displayed:

#	Field	Type	Null	Key	Default	Extra
1	p_id	int(11)	NO	PRI	<NULL>	auto_increment
2	u_id	int(11)	YES	MUL	<NULL>	
3	p_pollution	varchar(50)	YES		<NULL>	
4	p_place	varchar(100)	YES		<NULL>	
5	p_description	varchar(500)	YES		<NULL>	

Idea table : Stores idea information given by each user.

The screenshot shows the NetBeans IDE interface with the following details:

- File Menu:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Toolbar:** Standard Java development toolbar.
- Central Area:**
 - Services X:** Shows available databases.
 - Databases:** MySQL Server at loc...
 - Projects:** Shows Java DB, Drivers, jdbcderby://localhost, and jdbcmysql://localhost.
 - Files:** Shows pollution database structure:
 - Tables:** admin, event, feedbac, idea (selected), impleme, outcome, problem, requiren, sponsor, users, volunt.
 - Views:**
 - Procedures:**
 - Other database:**
 - Output:** Shows multiple execution tabs: SQL 42 execution, SQL 43 execution, SQL 44 execution, and SQL 45 execution.
- Bottom:** Taskbar with search bar, system icons, and system status (Windows 10, ENG, IN, 12:29 AM, 7/10/2020).

In the central area, the "idea" table is selected in the Navigator. The "desc idea" command is run, and the resulting table structure is displayed:

#	Field	Type	Null	Key	Default	Extra
1	i_id	int(11)	NO	PRI	<NULL>	auto_increment
2	u_id	int(11)	YES	MUL	<NULL>	
3	i_pollution	varchar(50)	YES		<NULL>	
4	i_description	varchar(500)	YES		<NULL>	

Implements : Stores information about which idea is used for which event.

The screenshot shows the NetBeans IDE interface with the following details:

- Navigator:** Shows the project structure and database connections.
- Services:** Shows the 'pollution' database structure, including tables like admin, event, feedbac, idea, implements, outcome, problem, requiren, sponsor, users, and volunteer.
- Central Editor:** Displays the SQL query: `SELECT * FROM implements;` and its description: `desc implements;`
- Output:** Shows the results of the `desc implements;` query, which includes columns: #, Field, Type, Null, Key, Default, and Extra. The results are:

#	Field	Type	Null	Key	Default	Extra
1	e_id	int(11)	YES	MUL	<NULL>	
2	i_id	int(11)	YES	MUL	<NULL>	
- Bottom:** Status bar showing 'SQL statement(s) executed successfully.'

Event table: Stores event information ,description is given below.

The screenshot shows the NetBeans IDE interface with the following details:

- Navigator:** Shows the project structure and database connections.
- Services:** Shows the 'pollution' database structure, including tables like admin, event, feedbac, idea, implements, outcome, problem, requiren, sponsor, users, and volunteer.
- Central Editor:** Displays the SQL query: `SELECT * FROM event;` and its description: `desc event;`
- Output:** Shows the results of the `desc event;` query, which includes columns: #, Field, Type, Null, Key, Default, and Extra. The results are:

#	Field	Type	Null	Key	Default	Extra
1	p_id	int(11)	YES	MUL	<NULL>	
2	e_id	int(11)	NO	PRI	<NULL>	auto_increment
3	a_id	int(11)	YES	MUL	<NULL>	
4	e_name	char(50)	YES		<NULL>	
5	e_pollution	varchar(50)	YES		<NULL>	
6	e_place	varchar(50)	YES		<NULL>	
7	e_date	date	YES		<NULL>	
- Bottom:** Status bar showing 'SQL statement(s) executed successfully.'

Requirements table : Stores Requirement information of each event ,description is given below.

The screenshot shows the NetBeans IDE interface with the following details:

- Toolbar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help.
- Services X:** Databases (MySQL Server at loc, Java DB, Drivers, jdbc:derby://localhost, jdbc:mysql://localhost).
- Projects:** pollution.
- Navigator:** Tables (admin, event, feedbac, idea, implemen, outcome, problem, require, sponsor, users, volunteer, Views, Procedures, Other database: Web Services).
- Code Editor:** Connection: jdbc:mysql://localhost:3306/pollution?zeroDateTimeBehavior=convertToNull [root on Default schema]


```
1 SELECT * FROM requirement;
2 desc requirement;
```
- Table Definition:** desc requirement x

#	Field	Type	Null	Key	Default	Extra
1	e_id	int(11)	YES	MUL	<NULL>	
2	r_id	int(11)	NO	PRI	<NULL>	auto_increment
3	r_money	int(11)	YES		<NULL>	
4	r_materials	varchar(100)	YES		<NULL>	
5	r_volunteers	varchar(100)	YES		<NULL>	
- Output X:** SQL 42 execution, SQL 43 execution, SQL 44 execution, SQL 45 execution, SQL 46 execution, SQL 47 execution, SQL 48 execution, SQL 49 execution.
- Status Bar:** SQL statement(s) executed successfully.

Sponsor table : Stores Sponsor's information

The screenshot shows the NetBeans IDE interface with the following details:

- Toolbar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help.
- Services X:** Databases (MySQL Server at loc, Java DB, Drivers, jdbc:derby://localhost, jdbc:mysql://localhost).
- Projects:** pollution.
- Navigator:** Tables (Admin, event, feedbac, idea, implemen, outcome, problem, require, sponsor, users, volunteer, Views, Procedures, Other database: Web Services, Servers, Maven Repositories).
- Code Editor:** Connection: jdbc:mysql://localhost:3306/pollution?zeroDateTimeBehavior=convertToNull [root on Default schema]


```
1 SELECT * FROM sponsor;
2 desc sponsor;
```
- Table Definition:** desc sponsor x

#	Field	Type	Null	Key	Default	Extra
1	e_id	int(11)	YES	MUL	<NULL>	
2	u_id	int(11)	YES	MUL	<NULL>	
3	amo	int(11)	YES		<NULL>	
- Output X:** SQL 42 execution, SQL 43 execution, SQL 44 execution, SQL 45 execution, SQL 46 execution, SQL 47 execution, SQL 48 execution, SQL 49 execution, SQL 50 execution.
- Status Bar:** SQL statement(s) executed successfully.

Volunteers table: Stores volunteers information.

The screenshot shows the NetBeans IDE interface with the following details:

- Title Bar:** NetBeans IDE 8.1
- Toolbar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Connection:** jdbc:mysql://localhost:3306/pollution?zeroDateTimeBehavior=convertToNull [root on Default schema]
- Code Editor:** SQL 49 [jdbc:mysql://localhost:33...]


```
1 SELECT * FROM volunteer;
2 desc volunteer;
```
- Database Navigator:**
 - Databases: MySQL Server at loc, Java DB, Drivers, jdbc derby://localhost, jdbc derby://localhost, jdbc mySQL://localhost, jdbc mySQL://localhost
 - Tables: pollution (admin, event, feedbac, idea, impleme, outcome, problem, requiren, sponsor, users), volunteer (e_id, u_id, Inde, Fore), Views, Procedures
- Output Window:** desc volunteer x

#	Field	Type	Null	Key	Default	Extra
1	e_id	int(11)	YES	MUL	<NULL>	
2	u_id	int(11)	YES	MUL	<NULL>	
- System Bar:** SQL statement(s) executed successfully.

Outcome table : Stores information about outcome of each event.

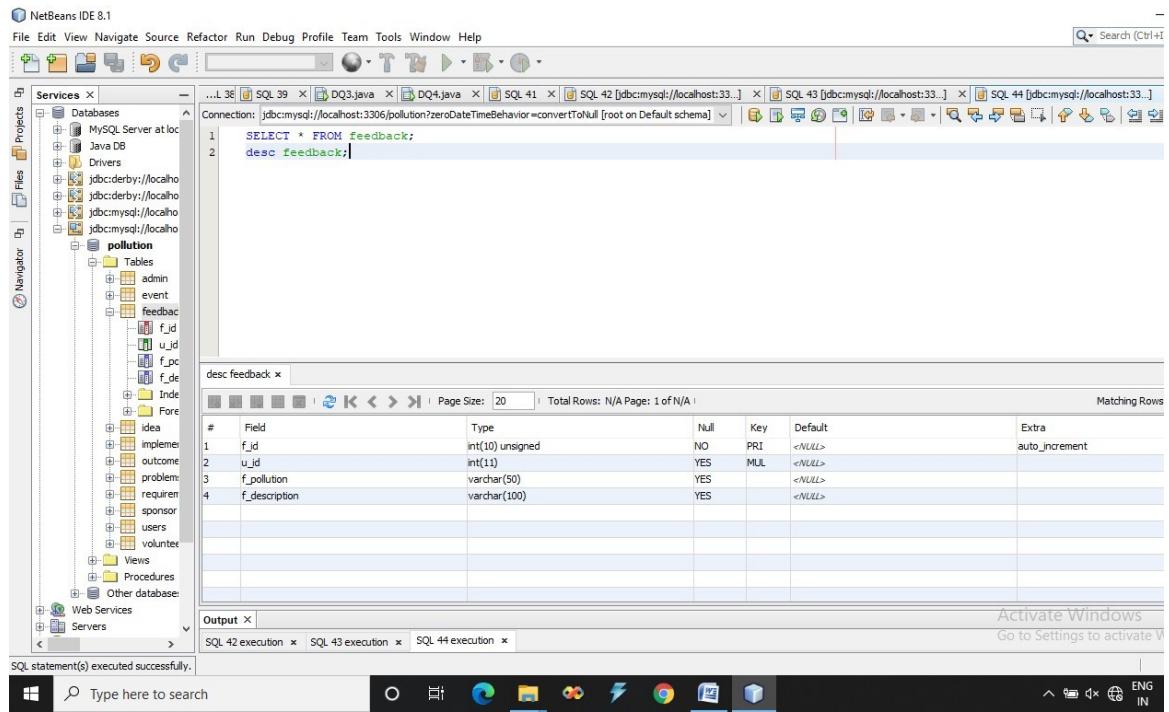
The screenshot shows the NetBeans IDE interface with the following details:

- Title Bar:** NetBeans IDE 8.1
- Toolbar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Connection:** jdbc:mysql://localhost:3306/pollution?zeroDateTimeBehavior=convertToNull [root on Default schema]
- Code Editor:** SQL 44 [jdbc:mysql://localhost:33...]


```
1 SELECT * FROM outcome;
2 desc outcome;
```
- Database Navigator:**
 - Databases: MySQL Server at loc, Java DB, Drivers, jdbc derby://localhost, jdbc derby://localhost, jdbc mySQL://localhost, jdbc mySQL://localhost
 - Tables: pollution (admin, event, feedbac, idea, impleme, outcome, problem, requiren, sponsor, users), volunteer (e_id, u_id, Inde, Fore), Views, Procedures
- Output Window:** desc outcome x

#	Field	Type	Null	Key	Default	Extra
1	e_id	int(11)	YES	MUL	<NULL>	
2	o_status	varchar(20)	YES		<NULL>	
3	o_description	varchar(500)	YES		<NULL>	
- System Bar:** SQL statement(s) executed successfully.

Feedback table : Stores information about feedback of each event.



Module specification

1. User Module:

- i. **Registration:** The user register himself to the application. First the user will enter his/her information like name, address, contact number, email id, username and password in the registration form, the form is then validated for correctness of the data at client side and then sent to the server, once it sent to the server and stored in the database user will get confirmation as registration successful.
- ii. **Login:** After user registration, the user who is registered clicks on the login, fills the data like username and password in the form and click on login button. The submitted form is then validated with the database data. If the username and password is matching, then the user is allowed to go to the next page. If the user credentials are not matching, then error

- message is displayed.
- iii. **Reporting pollution related problems:** The registered user,after login can reports pollution related problems by filling the details of the pollution like pollution type,pollution area,pollution description in the report problem form, the details are then stored in the database and user will confirmation as Problem reported successfully.
 - iv. **.Sharing Idea :** Registered user after login can share ideas to solve pollution related problem by filling details like pollution type,description in the IDEA form and then these details are stored in the database and the user will get confirmation as idea submitted successfully.
 - v. **Feedback:** Registered user after login can give feedback about the event by filling the details like event id,pollution type,description in FEEDBACK FORM then details are stored in the database.

2.Admin module:

- i. **.Event Details:** To solve problems reported by users, admin will login ,views user reported problems and organize event by filling details like event name,pollution type,event place, event date in the EVENT FORM and then event details are stored in the database.
- ii. **Requirements :** Once the event is planned by the admin to solve particular pollution related problem ,he/she will specify needs/requirements of the event to conduct it successfully ,by filling details like required money,required materials and required volunteers in the REQUIREMT FORM and then details are stored in the database.
- iii. **Outcome:** Once the event is conducted successfully the outcome of the event will be updated by admin by filling details like Event status and description in the OUTCOME FORM then details are stored in the database.

Context Diagram –

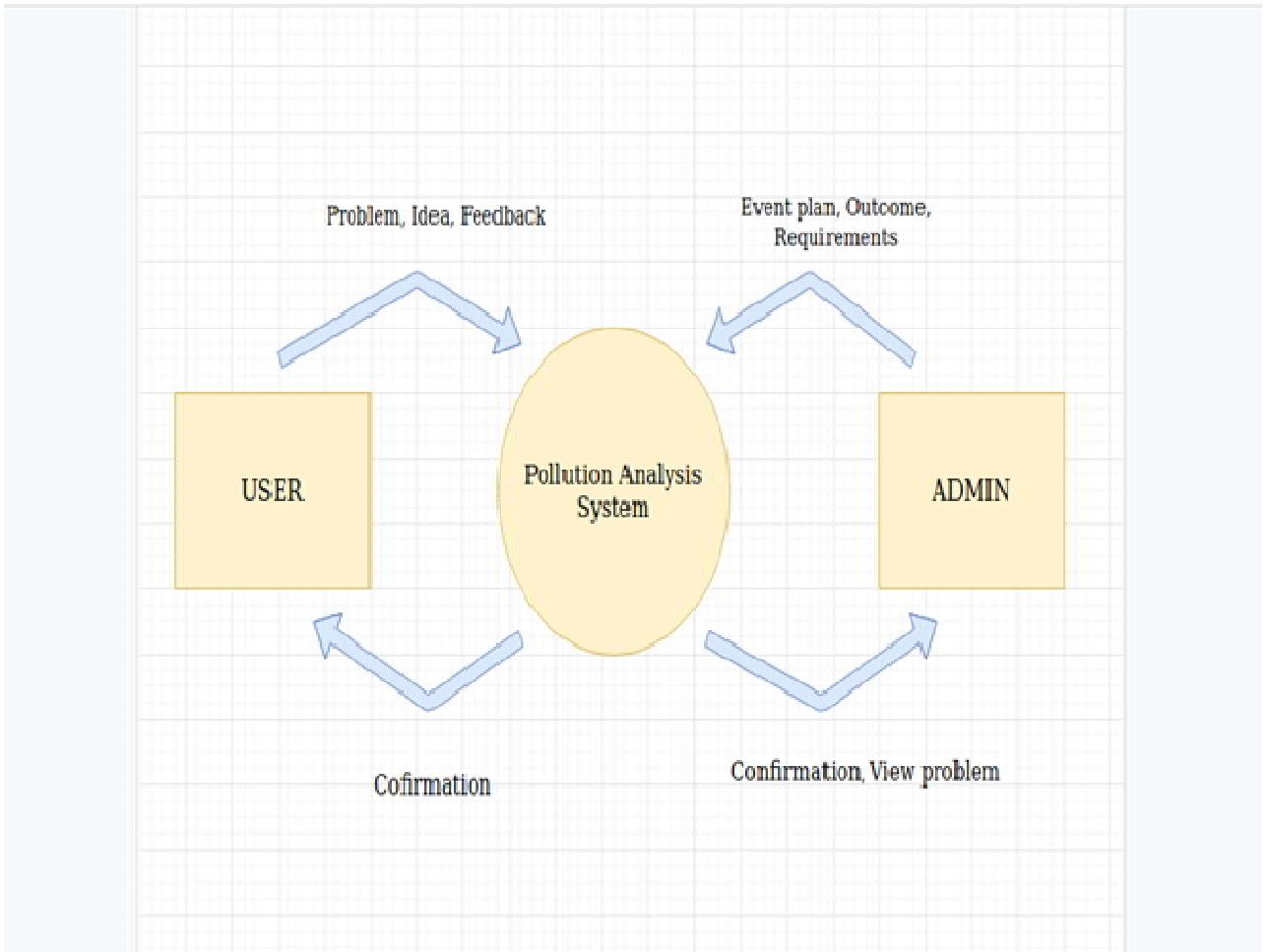


Figure 3: Context Diagram

Context diagram is also called as 0 level data flow diagram. It identifies the flows of information between the system and external entities. The entire system is shown as a single process. Here the flow of information takes place between the pollution analysis system and user, admin.

4. Detailed Design

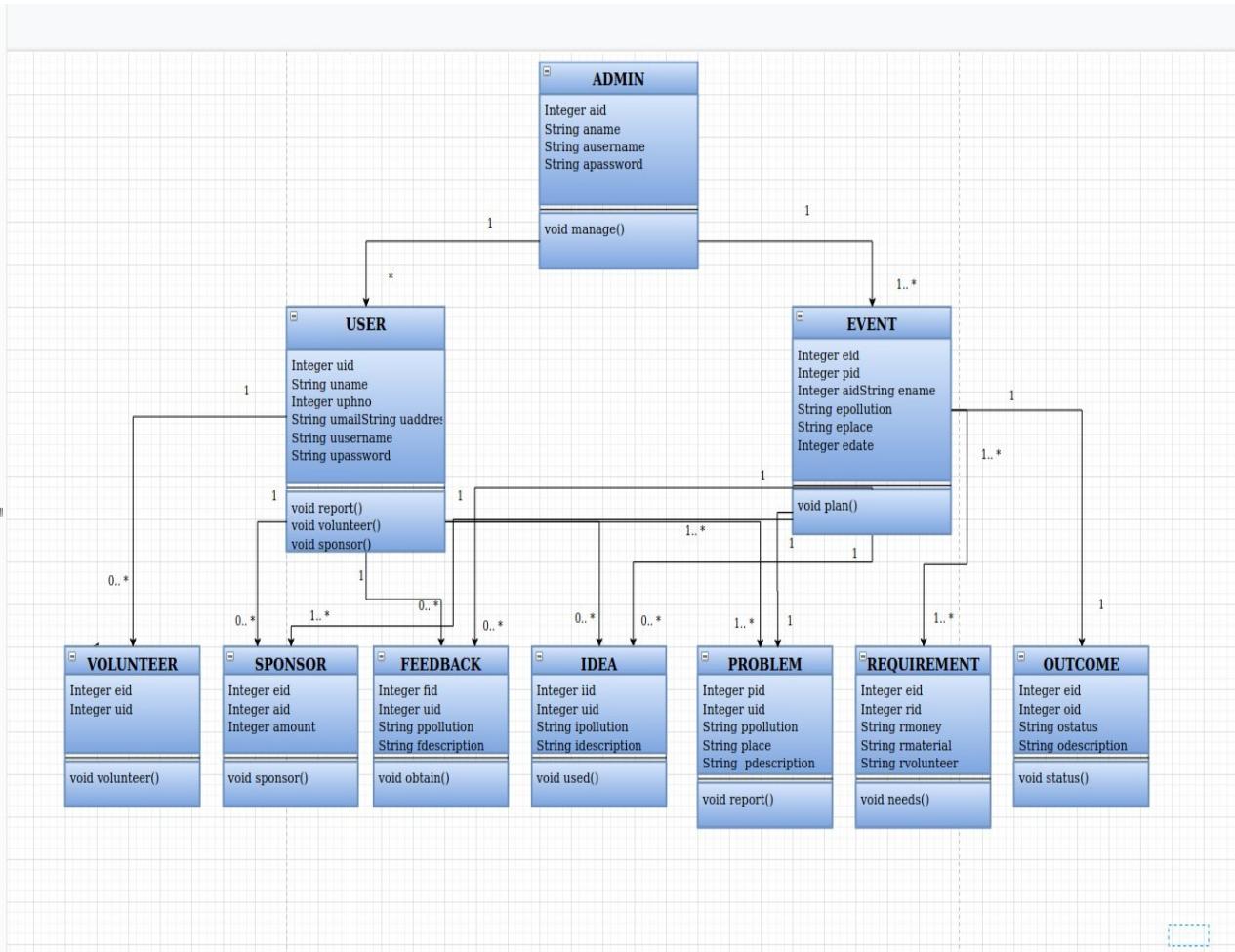
System Design

Brief description (Give justification why you have chosen the approach .

Three stages of object oriented design such as object modeling, dynamic modeling and functional modeling in the context of the project.)

Object Modeling

i. Class Diagram



. Figure 4: Class Diagram

Class diagrams are the main building block in object-oriented modeling. They are used to show the different objects in a system, their attributes, their operations and the relationships among them. In this class diagram we have different classes like user, admin, event, volunteer, sponsor, feedback, requirement, problem, idea and outcome. Admin class have different attributes like admin id, name, user id and password. And the relationship between admin and event class is admin will organize the event according to the problem reported by the user.

Dynamic modeling

ii. Use case Diagram

A use case diagram is a simplest representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. In our application use case is diagram shows how user is interacting with pollution analysis system with different use cases like reporting problem, giving ideas about pollution related problems etc.

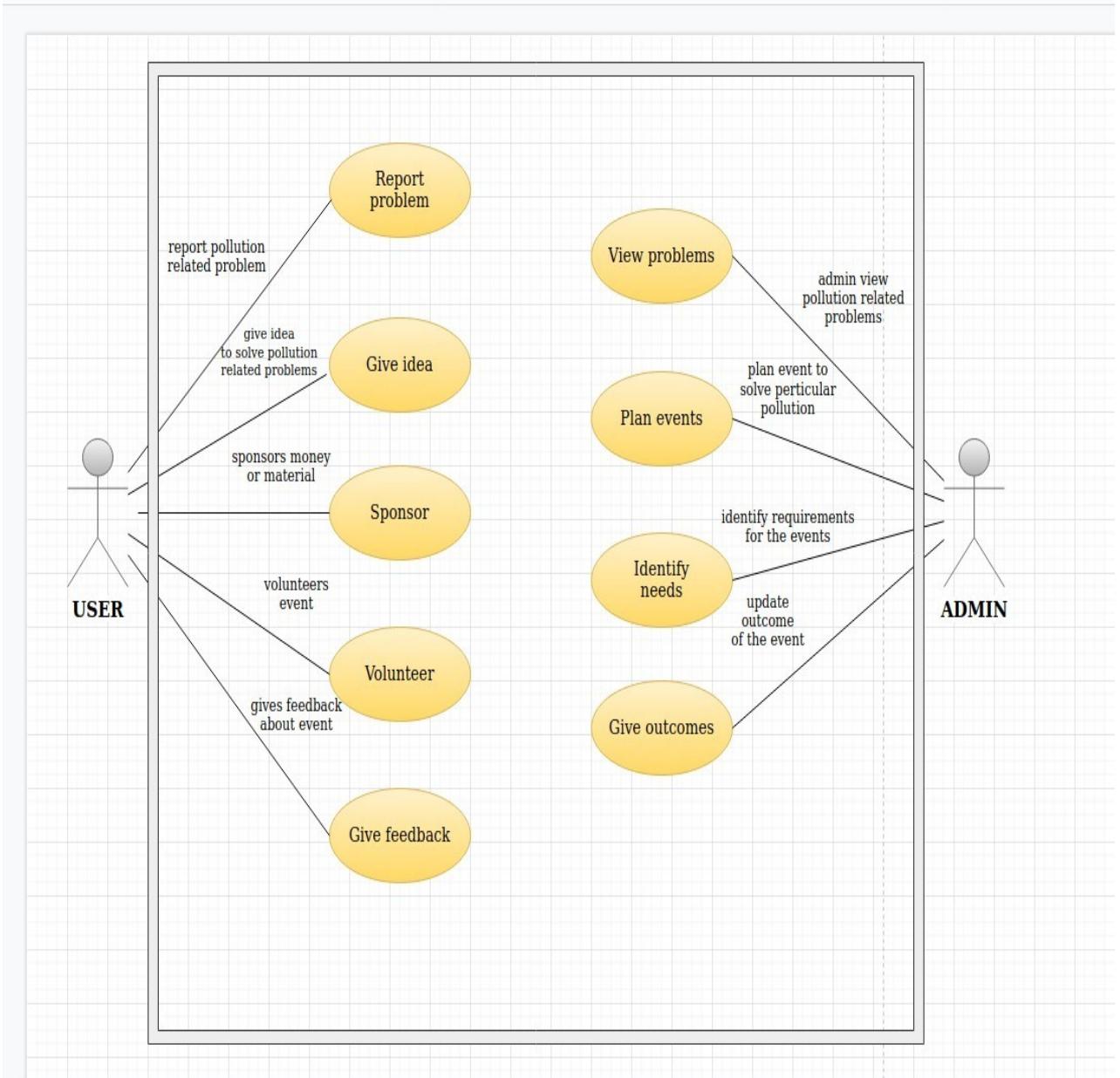


Figure 5: Use case Diagram

iii. Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. In our application we have two activity diagram they are User and admin activity diagram.

1. USER ACTIVITY

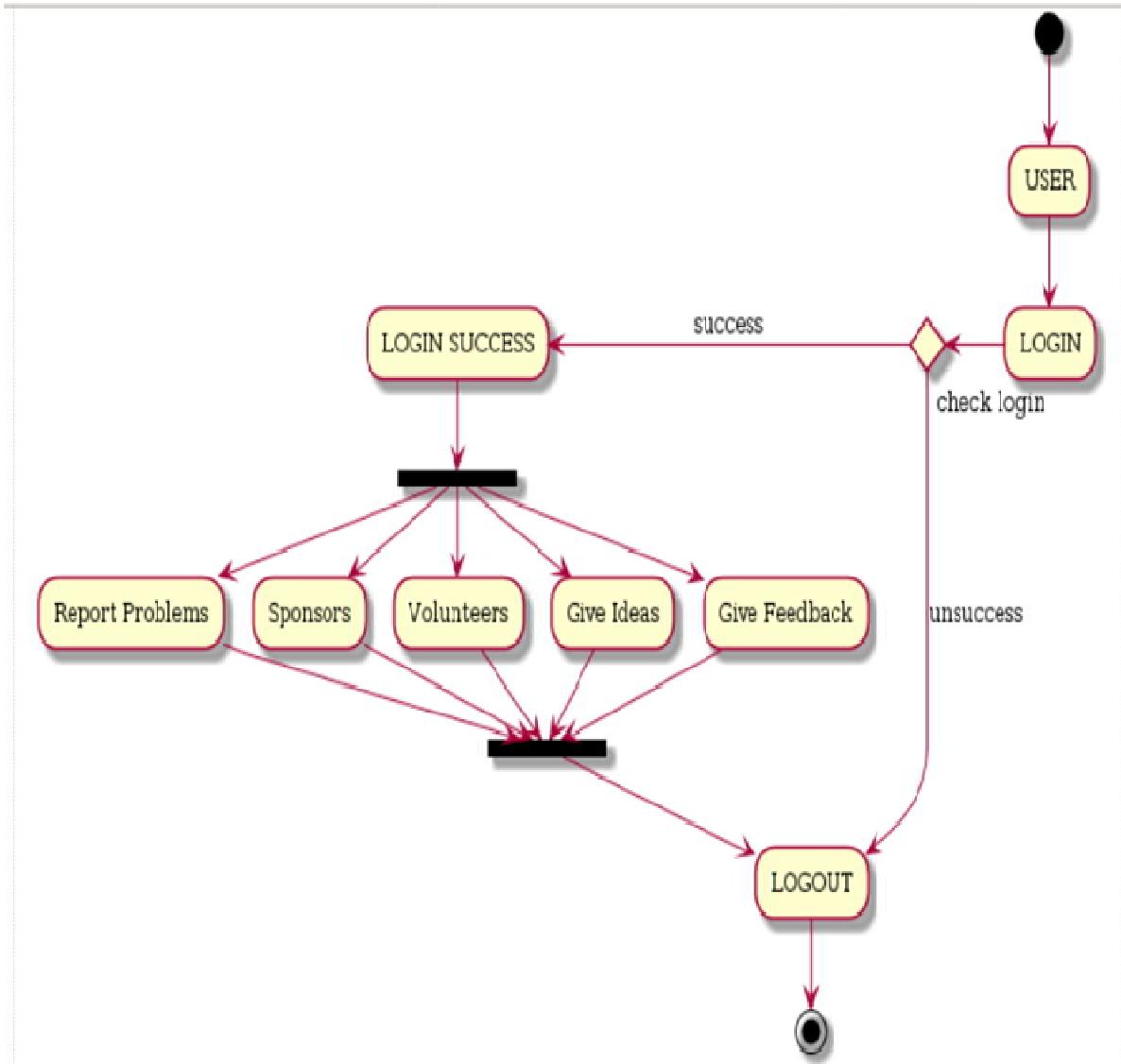


Figure 6(a): User Activity Diagram

First user will login to the application, if login is successful he moves to other activities like he can report problem, give idea, give feedback and he can also be sponsor or volunteer. If he fails to login then he moves to the logout activity. Same process takes place in admin activity diagram.

2. ADMIN ACTIVITY

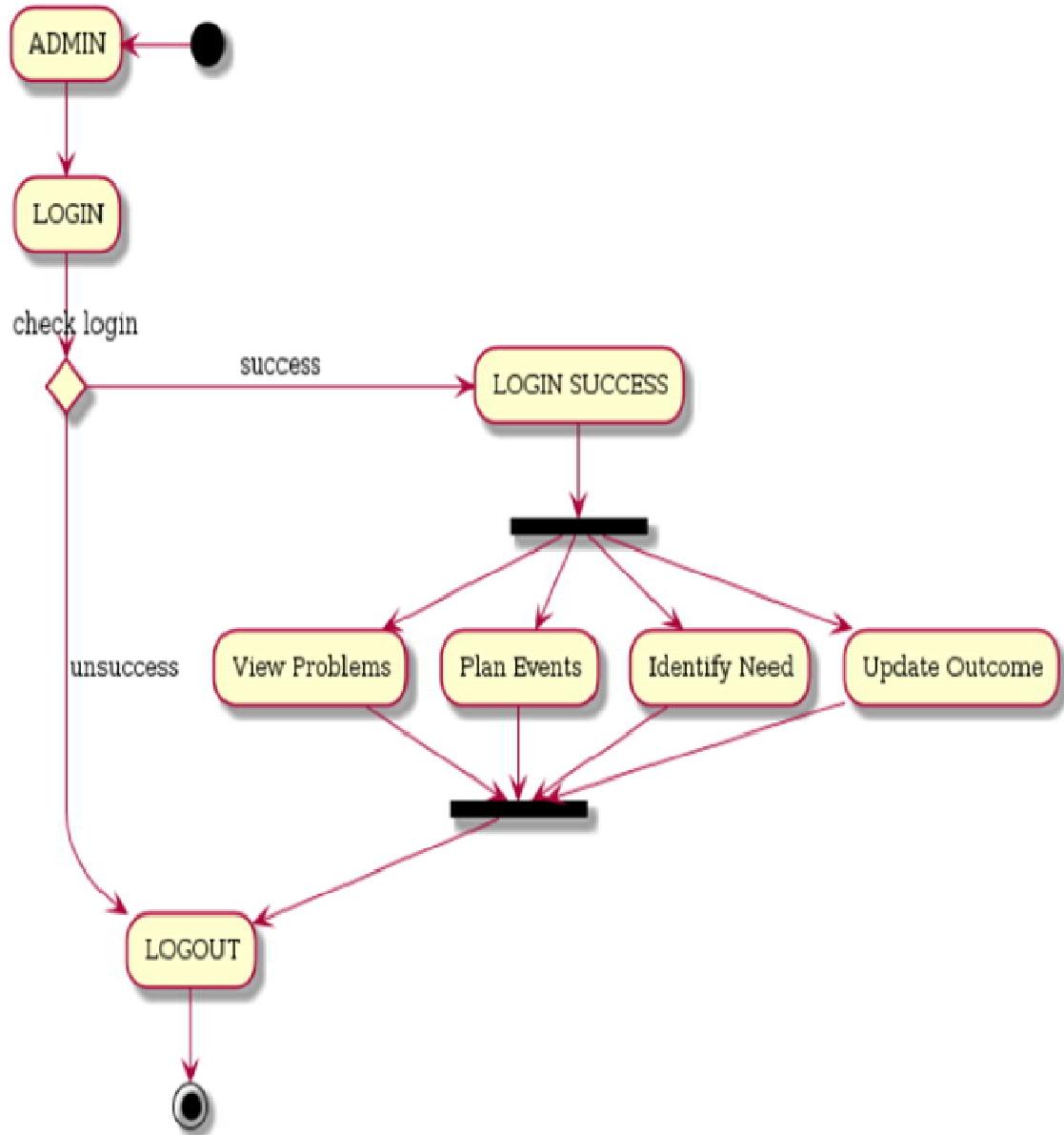


Figure 6(b): Admin Activity Diagram

First admin will login to the system if login successful he can view problems, plan events, Identify needs for the events and also he can update outcome of the event and then he can logout If he fails to login then he moves to the logout activity

iv. Sequence Diagram

The above diagram depicts the sequence diagram of pollution analysis and control system. It performs interaction between objects in a single use case. It illustrates how different parts of the system interact with each other.

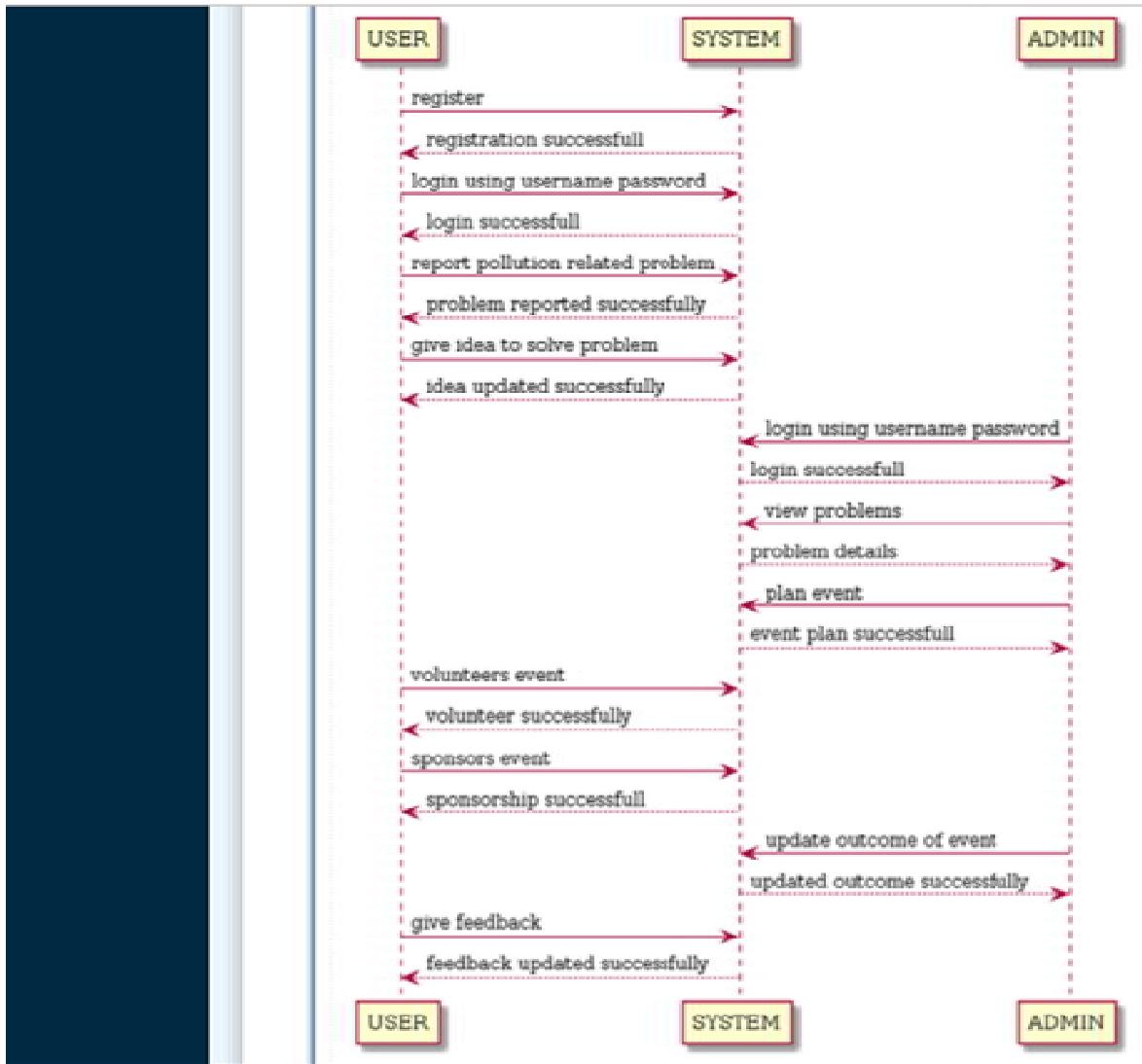


Figure 7: Sequence diagram

In the above diagram it shows the interaction between user and admin model. Firstly the application is opened by the user then he could register or login to the application. Once he is successfully login he will get confirmation message from the system.

User will report problem, give idea and give feedback once he is done he will get confirmation message from the system. Admin can also view the reported problem and plan events.

v. State Machine Diagram

State diagram are used to capture the behavior of the software system.

It provides us an efficient way to model the communication that occur within external entities.

and a system. In this application we have two state diagrams one is user and other is admin.

a. USER STATE DIAGRAM

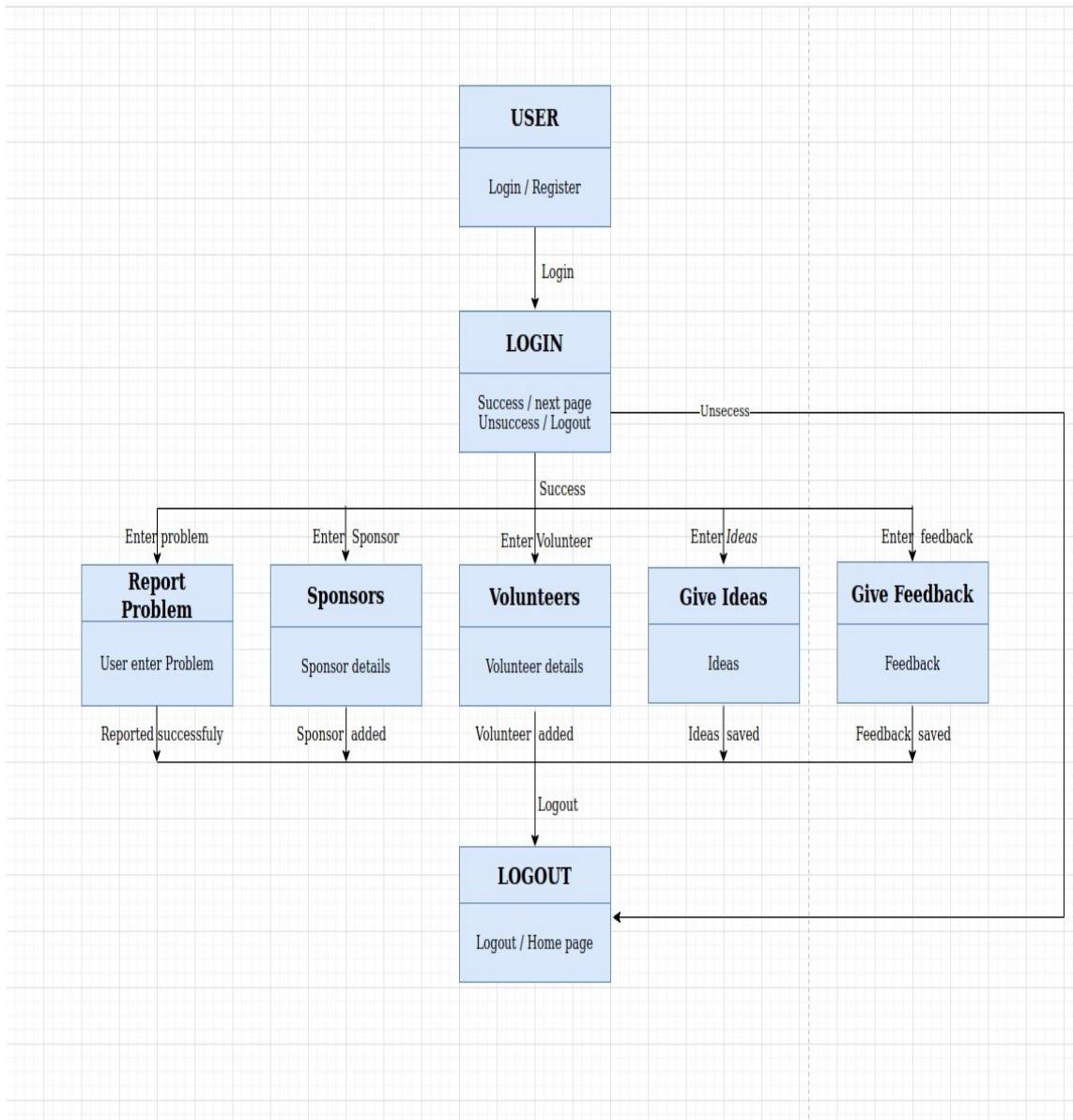


Figure 8(a): User State Diagram

It describes various state of an entity like if user got success in login then it goes to next state where he can report problem , give idea etc. If he is failed to login then he goes to the logout state. Same process is repeated in the admin state diagram.

b. ADMIN STATE DIAGRAM

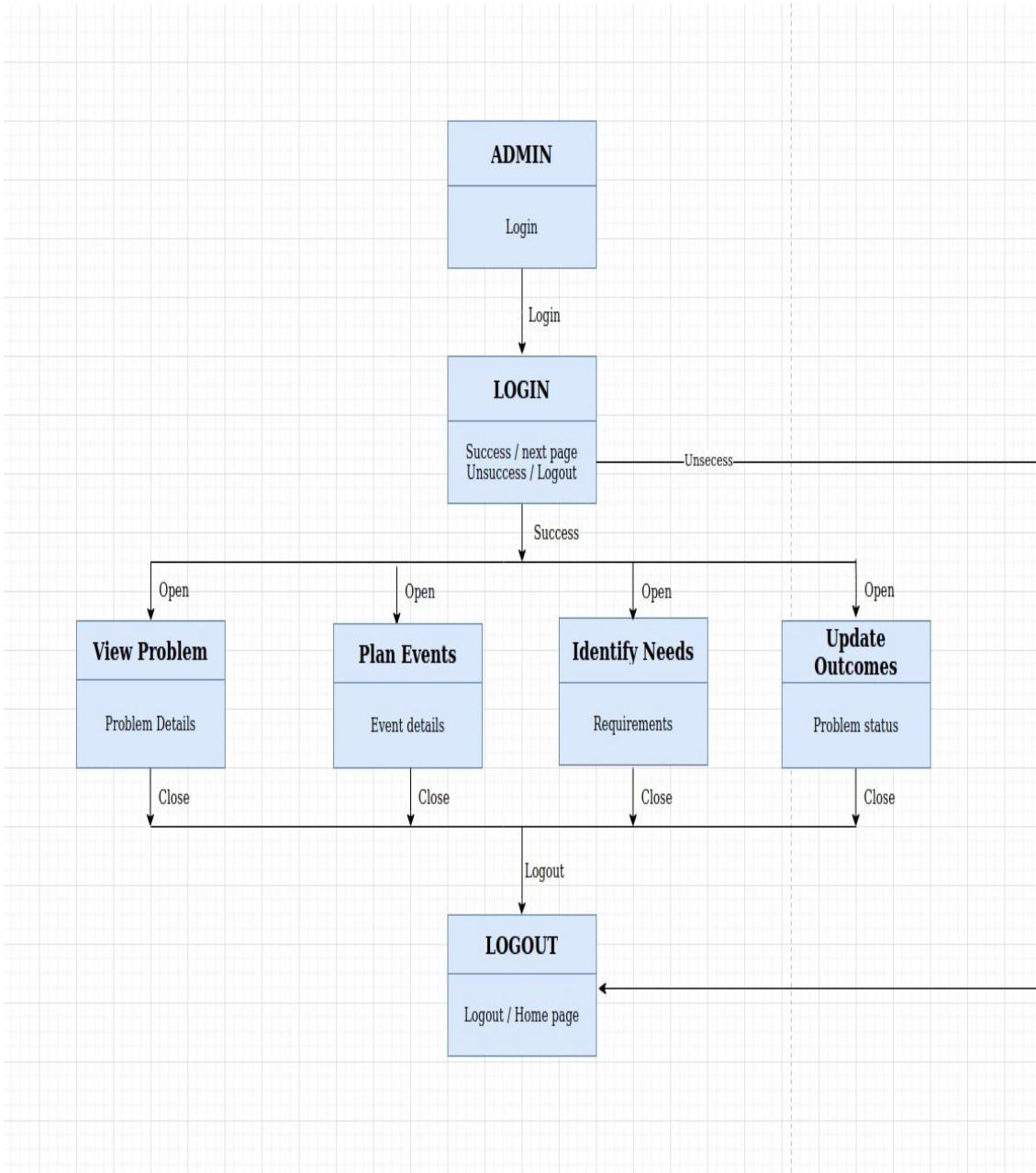


Figure 8(b): Admin state diagram

It describes various state of an entity like if Admin got success in login then it goes to next state where he can view problem, plan events, identify needs and update outcomes once it's done his state further changed to next level. If he is failed to login then he goes to the logout state. Same process is repeated in the admin state diagram.

Functional Modeling

i. Data Flow Diagrams

Data flow diagram is a way of representing flow of data through a system. It also provides the information about the outputs and inputs of each entity and process itself. . We have two dfd diagrams that is user and admin.

a. USER DATAFLOW DIAGRAM

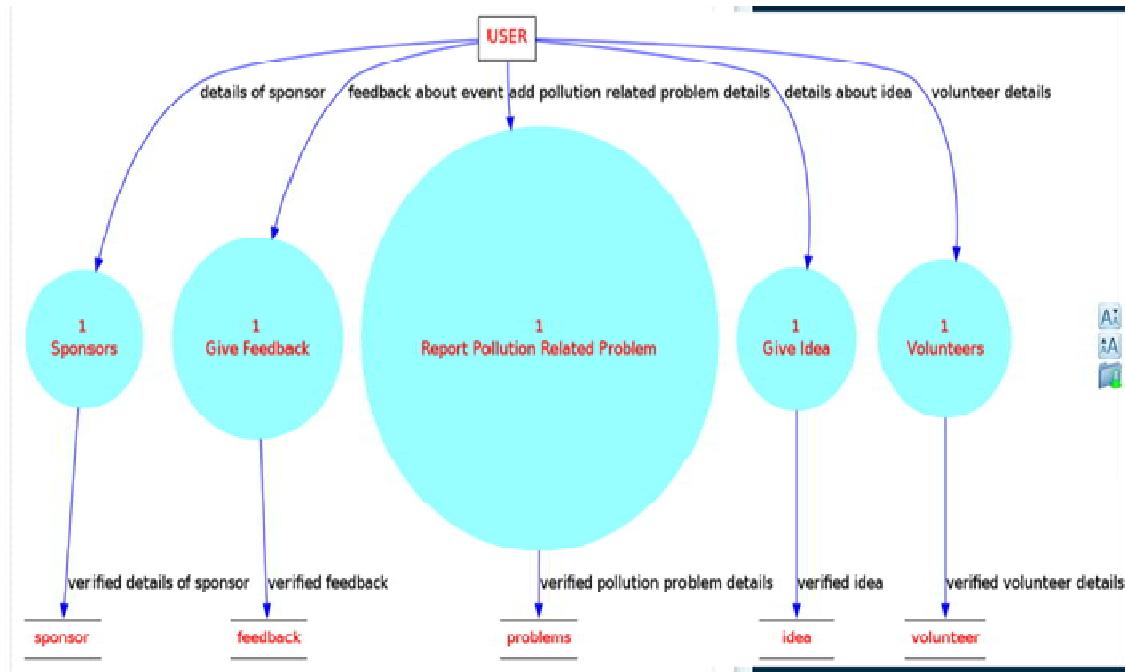


Figure 9(a): USER DATAFLOW DIAGRAM

When user will report the problem it will be verified by the system stored in the database and then viewed by the admin. In the same way every process takes place.

b. ADMIN DATAFLOW DIAGRAM

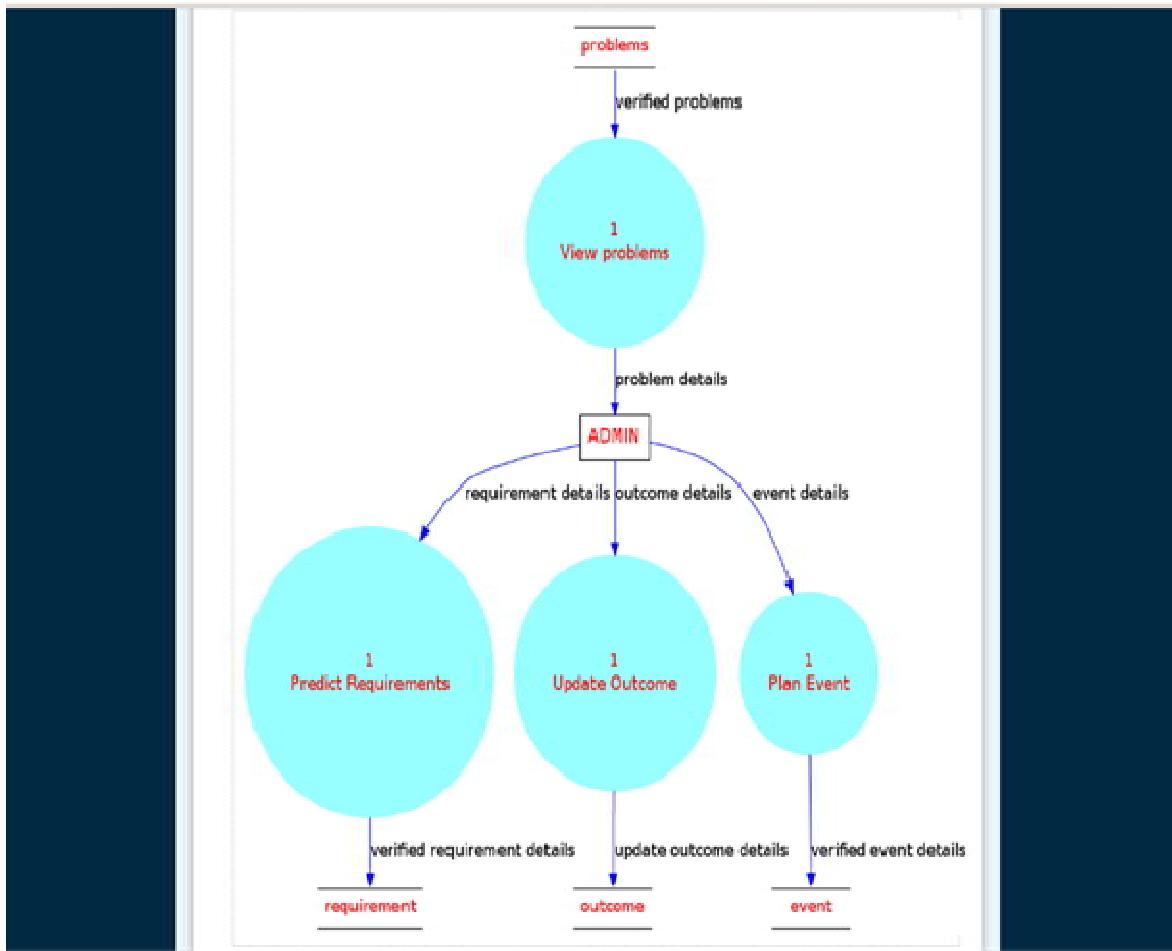


Figure 9(b): ADMIN DATAFLOW DIAGRAM

Verified pollution problem of user is stored in database these are viewed by the admin and then admin will be planning the event to solve those problems, he will be specifying the requirements for the event and finally updates the outcome of the event.

ii. Database Design (ER Diagram / Conceptual Scheme)

ER diagram is a representation of data that describes how data is related to each other. We disintegrate data into entities, attributes and set up relationship between entities.

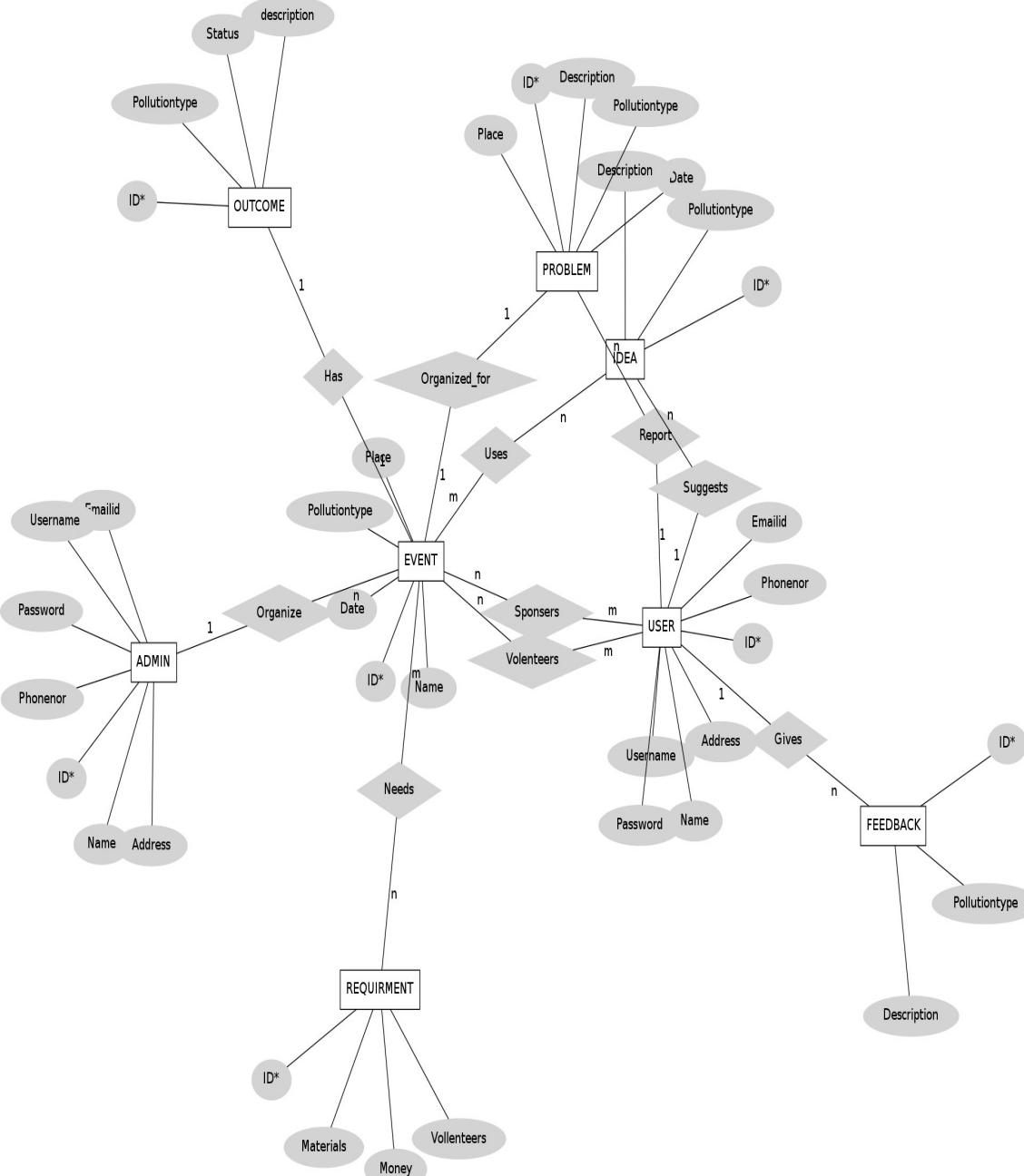


Figure 10: ER Diagram

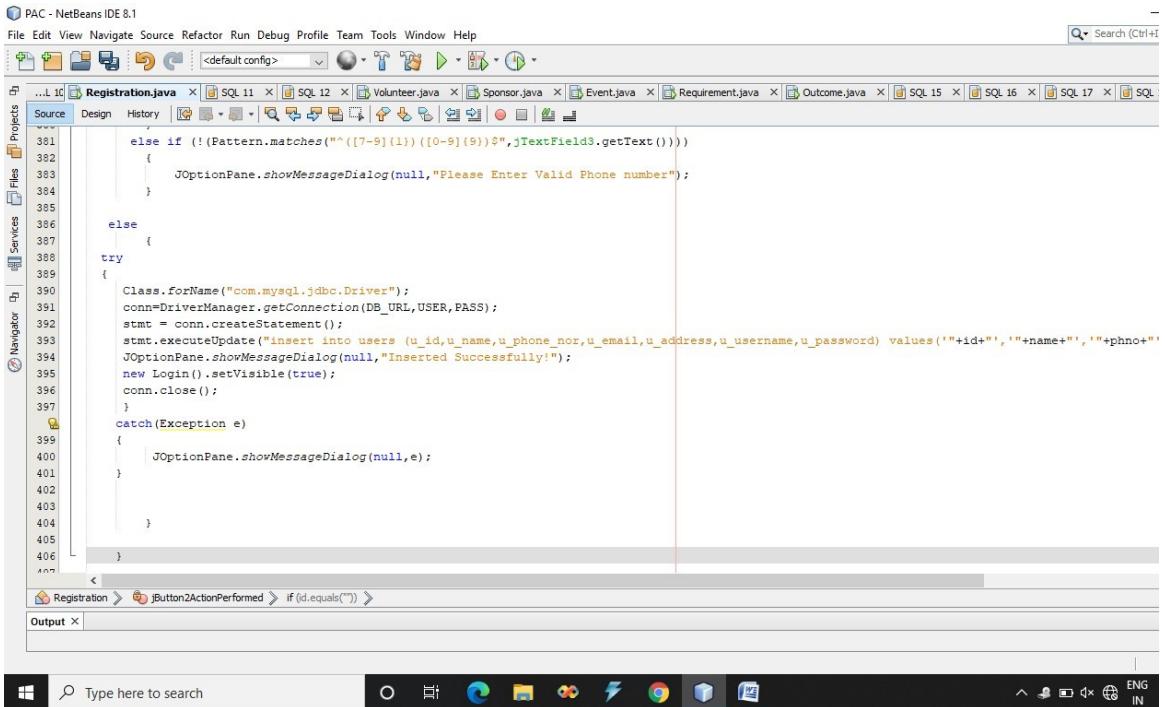
In our application we have entities like user, admin, event, requirement, problem, idea, feedback, and outcome. Attributes of admin are id, name, address, phone number, password, user name and email id. Organize is a relationship between admin and event so that admin will organize the event to solve the problem reported by the user.

5.

Implementation

Code Snippets and Implementation

- 1) **.Registration:** The user register himself to the application. First the user will enter his/her information like name,address,contact number,email id,username and password in the registration form, the form is then validated for correctness of the data at client side and then sent to the server ,once it sent to the server and stored in the database user will get confirmation as registration successful.



The screenshot shows the NetBeans IDE interface with the following details:

- Title Bar:** PAC - NetBeans IDE 8.1
- Menu Bar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Toolbar:** Standard NetBeans toolbar with icons for file operations.
- Search Bar:** Search (Ctrl+F)
- Project Explorer:** Shows a single project named "PAC".
- Source Editor:** Displays the `Registration.java` file with the following code:

```

381     else if (!Pattern.matches("^(\\d{10})|([0-9]{9})$", jTextField3.getText()))
382     {
383         JOptionPane.showMessageDialog(null,"Please Enter Valid Phone number");
384     }
385
386     else
387     {
388         try
389         {
390             Class.forName("com.mysql.jdbc.Driver");
391             conn=DriverManager.getConnection(DB_URL,USER,PASS);
392             stmt = conn.createStatement();
393             stmt.executeUpdate("insert into users (u_id,u_name,u_phone_nor,u_email,u_address,u_username,u_password) values('"+id+"','"+name+"','"+phno+"'";
394             JOptionPane.showMessageDialog(null,"Inserted Successfully!");
395             new Login().setVisible(true);
396             conn.close();
397         }
398         catch(Exception e)
399         {
400             JOptionPane.showMessageDialog(null,e);
401         }
402     }
403 }
404 }
405 }
406 }

```

- Output Panel:** Shows the output of the code execution.
- System Tray:** Shows standard Windows system tray icons.

The insertion code is used in try – catch block along with connectivity code. Insertion code is used to insert user data into database. The code/ query is given bellow

`stmt.executeUpdate("insert into users (u_id,u_name,u_phone_nor,u_email,u_address,u_username,u_password) values('"+id+"','"+name+"','"+phno+"','"+email+"','"+add+"','"+uname+"','"+pas+s+"')");`

- 2) **Login:** After user registration ,the user who is registered clicks on the login ,fills the data like username and password in the form and click on login button. The submitted form is then validated with the database data. If the username and password is matching, then the user is allowed to go to the next page. If the user credentials are not matching, then error message is displayed.

```

Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/pollution","root","root");
Statement st=con.createStatement();
String q="Select a_username,a_password from admin";
ResultSet rs=st.executeQuery(q);
while(rs.next())
{
    String username=rs.getString("a_username");
    String pass=rs.getString("a_password");

    if(a.equals(username) && pass.equals(b))
    {
        new MainPage().setVisible(true);
    }
    else
    {
        JOptionPane.showMessageDialog(null,"wrong Password");
    }
}
con.close();
}
catch(Exception e)
{
    JOptionPane.showMessageDialog(null,"error while establishing connection");
    JOptionPane.showMessageDialog(null,e);
}

```

In this code user input is stored in variable and connectivity is given. Once connectivity is successful then the user inputted data is matched from data in stored in database if the username and password matches data of an registered user then user reaches the user page otherwise an message box is displayed saying wrong password.

- 3) **Reporting pollution related problems:** The registered user,after login can reports pollution related problems by filling the details of the pollution like pollution type,pollution area,pollution description in the report problem form, the details are then stored in the database and user will confirmation as Problem reported successfully.

PAC - NetBeans IDE 8.1

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

<default config>

Services X

Source Design History

...Java Sponsor.java Event.java Requirement.java Outcome.java SQL 15 SQL 16 SQL 17 SQL 18 Problem.java Feedback.java

Databases Web Services Servers Maven Repositories Cloud Hudson Builders Task Repositories JS Test Driver Selenium Server

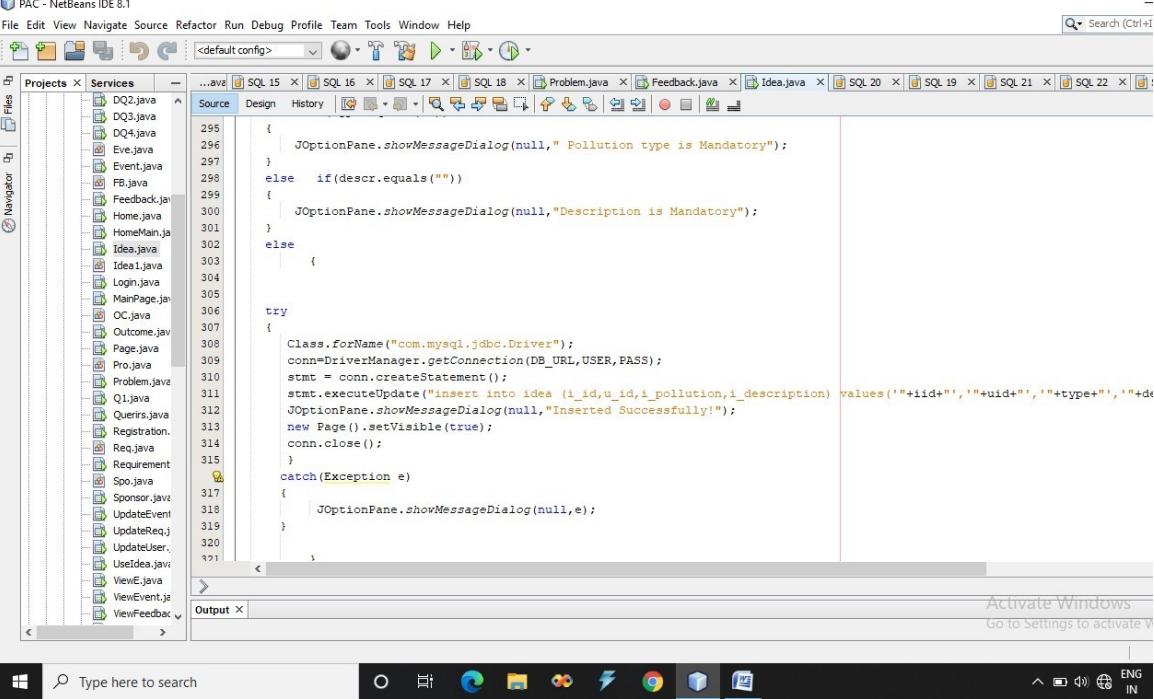
315 JOptionPane.showMessageDialog(null,"Description is Mandatory");
316 }
317 }
318 }
319 }
320 try
321 {
322 Class.forName("com.mysql.jdbc.Driver");
323 conn=DriverManager.getConnection(DB_URL,USER,PASS);
324 stmt = conn.createStatement();
325 stmt.executeUpdate("insert into problems (p_id,u_id,p_pollution,p_place,p_description) values('"+pid+"','"+aid+"','"+
326 JOptionPane.showMessageDialog(null,"Inserted Successfully!");
327 new Page().setVisible(true);
328 conn.close();
329 }
330 catch(Exception e)
331 {
332 JOptionPane.showMessageDialog(null,e);
333 }
334 }
335 }
336 }
337 }
338 }
339 }
340 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
341 // TODO add your handling code here:
>
Output X

Activate Windows
Go to Settings to activate V

The insertion code is used in try – catch block along with connectivity code. Insertion code is used to insert problem data into database. The function used is given below.

stmt.executeUpdate();

- 4) **Sharing Idea** : Registered user after login can share ideas to solve pollution related problem by filling details like pollution type, description in the IDEA form and then these details are stored in the database and the user will get confirmation as idea submitted successfully.



The screenshot shows the NetBeans IDE interface with the following details:

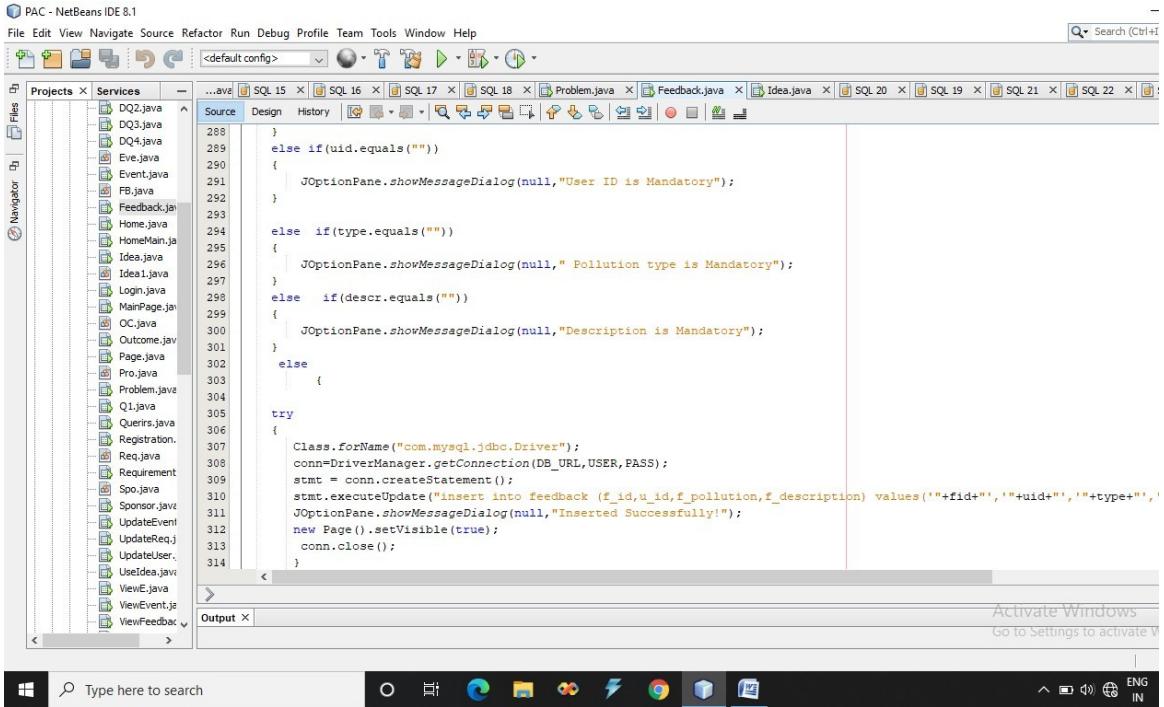
- Title Bar:** PAC - NetBeans IDE 8.1
- Menu Bar:** File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
- Toolbar:** Standard NetBeans toolbar with icons for file operations.
- Project Explorer:** Shows a list of Java files in the project, including DQ2.java, DQ3.java, DQ4.java, Eve.java, Event.java, FB.java, Feedback.java, Home.java, HomeMain.java, Idea.java, Idea1.java, Login.java, MainPage.java, OC.java, Outcome.java, Page.java, Problem.java, Q1.java, Queries.java, Registration.java, Req.java, Requirement.java, Spo.java, Sponsor.java, UpdateEvent.java, UpdateReq.java, UpdateUser.java, UseIdea.java, ViewEvent.java, ViewFeedback.java, and ViewPage.java.
- Code Editor:** Displays Java code for inserting data into a database. The code uses a try-catch block to handle exceptions. It connects to a MySQL database using the JDBC driver, creates a statement, and executes an insert query. It also handles null values for mandatory fields and displays success messages.
- Output Panel:** Shows the message "Activate Windows" and "Go to Settings to activate V".
- System Tray:** Shows standard Windows system tray icons.

The insertion code is used in try – catch block along with connectivity code. Insertion code is used to insert idea data into database. The function used is given below.

stmt.executeUpdate();

5) **Feedback:** Registered user after login can give feedback about the event

by filling the details like event id,pollution type,description in
FEEDBACK FORM then details are stored in the database.

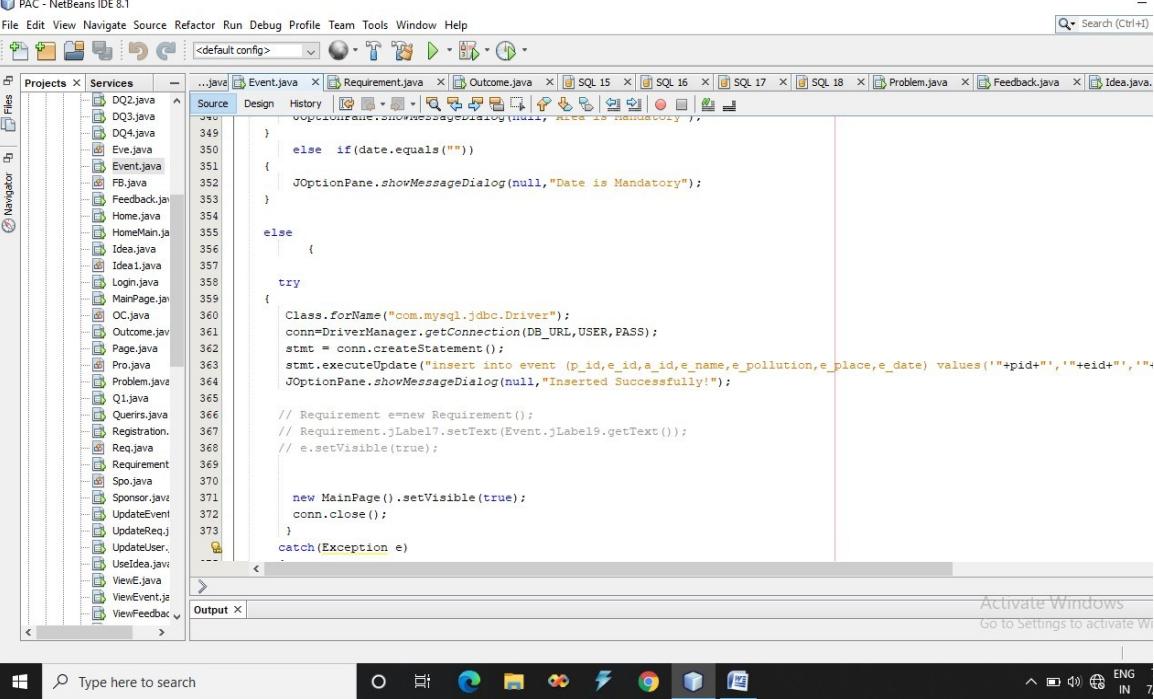


The insertion code is used in try – catch block along with connectivity code. Insertion code is used to insert feedback data into database. The function used is given below.

`stmt.executeUpdate();`

2.ADMIN MODULE

- 1) **Event Details:** To solve problems reported by users, admin will login ,views user reported problems and organize event by filling details like event name,pollution type,event place, event date in the EVENT FORM and then event details are stored in the database.



The screenshot shows the NetBeans IDE interface with the title bar "PAC - NetBeans IDE 8.1". The menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help. The toolbar has icons for file operations like Open, Save, Find, and Run. The main window shows a project tree on the left with files like DQ2.java, DQ3.java, DQ4.java, Eve.java, Event.java, FB.java, Feedback.java, Home.java, HomeMain.java, Idea.java, Idea1.java, Login.java, MainPage.java, OC.java, Outcome.java, Page.java, Pro.java, Problem.java, Q1.java, Querirs.java, Registration.java, Req.java, Requirement.java, Spo.java, Sponsor.java, UpdateEvent.java, UpdateReq.java, UpdateUser.java, UseIdea.java, ViewE.java, ViewEvent.java, and ViewFeedback.java. The central editor pane displays Java code for inserting data into a database:

```

...java Event.java x Requirement.java x Outcome.java x SQL 15 x SQL 16 x SQL 17 x SQL 18 x Problem.java x Feedback.java x Idea.java.

349     }
350     else if(date.equals(""))
351     {
352         JOptionPane.showMessageDialog(null,"Date is Mandatory");
353     }
354     else
355     {
356         try
357         {
358             Class.forName("com.mysql.jdbc.Driver");
359             conn=DriverManager.getConnection(DB_URL,USER,PASS);
360             stmt = conn.createStatement();
361             stmt.executeUpdate("insert into event (p_id,e_id,a_id,e_name,e_pollution,e_place,e_date) values('"+pid+"','"+eid+"','"+e
362             JOptionPane.showMessageDialog(null,"Inserted Successfully!");
363
364             // Requirement e=new Requirement();
365             // Requirement.jLabel7.setText(Event.jLabel9.getText());
366             // e.setVisible(true);
367
368             new MainPage().setVisible(true);
369             conn.close();
370         }
371         catch(Exception e)
372         {
373             e.printStackTrace();
374         }
375     }
376 }

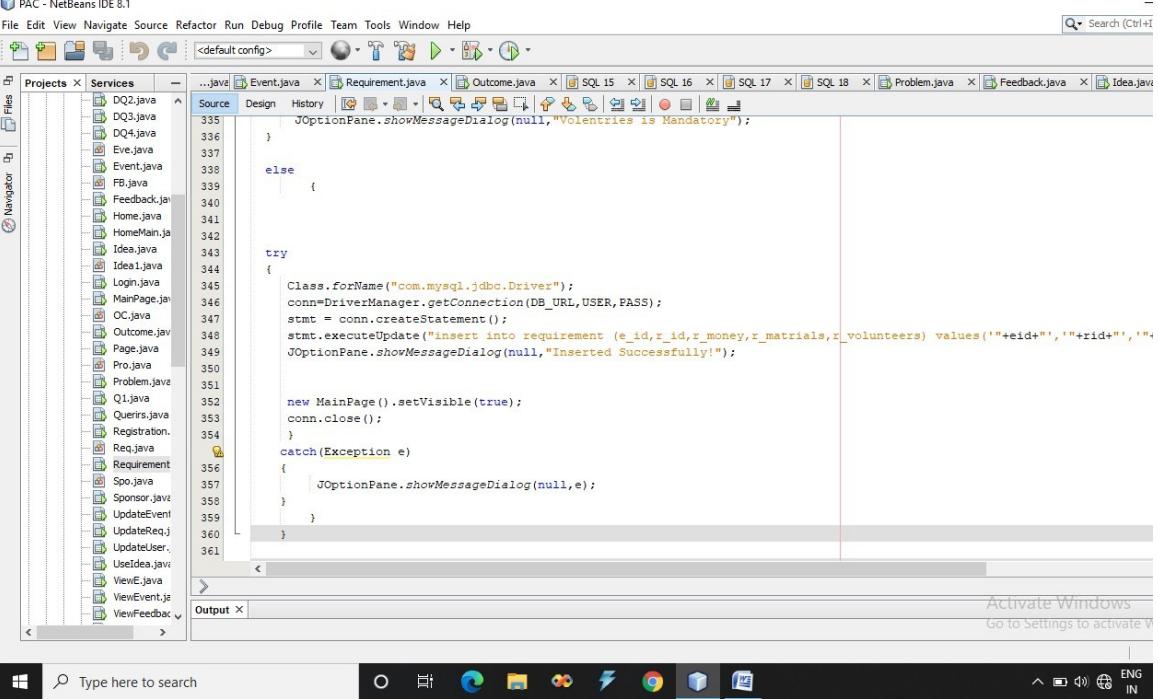
```

The status bar at the bottom right shows "Activate Windows" and "Go to Settings to activate Wi". The taskbar at the bottom has icons for File, Edit, View, Insert, Run, Stop, Refresh, and others.

The insertion code is used in try – catch block along with connectivity code. Insertion code is used to insert event data into database. The function used is given below.

`stmt.executeUpdate();`

- 2) **Requirements :** Once the event is planned by the admin to solve particular pollution related problem ,he/she will specify needs/requirements of the event to conduct it successfully ,by filling details like required money,required materials and required volunteers in the REQUIREMT FORM and then details are stored in the database.



The screenshot shows the NetBeans IDE interface with the title bar "PAC - NetBeans IDE 8.1". The menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help. The toolbar has icons for file operations like Open, Save, Find, and Run. The main window shows a "Projects" tab with a list of Java files: DQ2.java, DQ3.java, DQ4.java, Eve.java, Event.java, FB.java, Feedback.java, Home.java, HomeMain.java, Idea.java, Idea1.java, Login.java, MainPage.java, OC.java, Outcome.java, Page.java, Pro.java, Problem.java, Q1.java, Queris.java, Registration.java, Req.java, Requirement.java, Spo.java, Sponsor.java, UpdateEvent.java, UpdateReq.java, UpdateUser.java, UseIdea.java, ViewEvent.java, ViewFeedback.java, and Idea.java. The "Event.java" file is open in the editor, showing Java code for inserting requirements into a MySQL database. The code uses JDBC to connect to the database and execute an SQL update statement. It also includes JOptionPane.showInputDialog() for user input and JOptionPane.showMessageDialog() for displaying success messages.

```

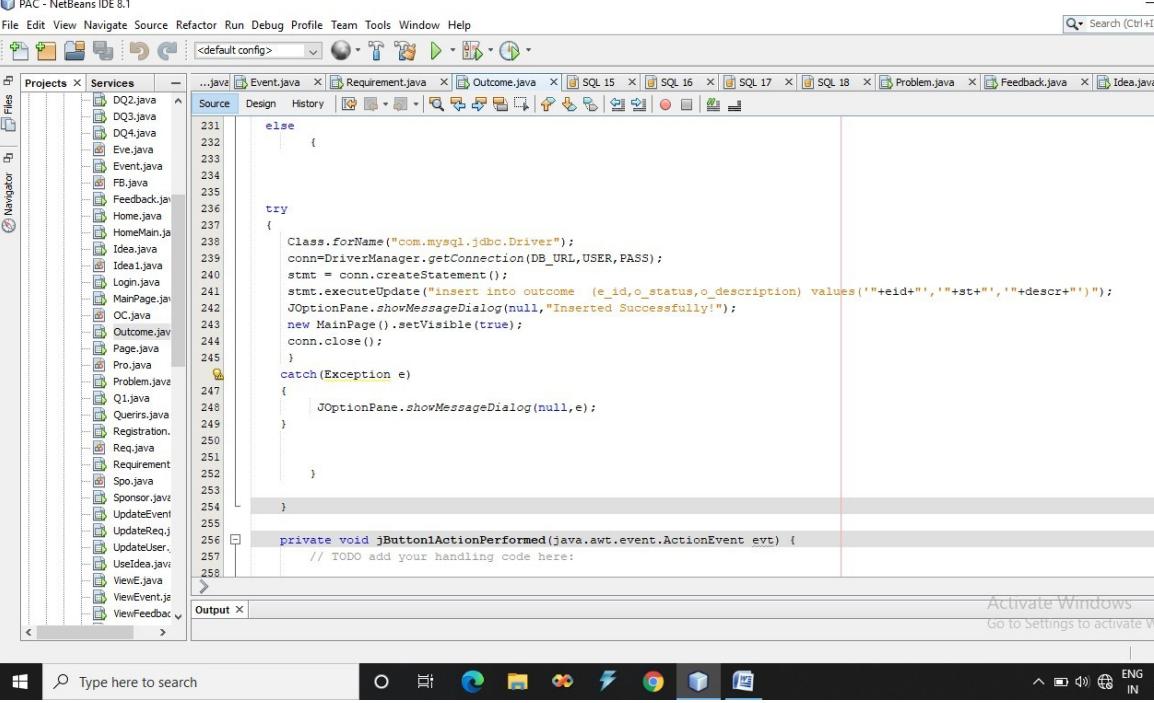
335     }
336     }
337     else
338     {
339         try
340         {
341             Class.forName("com.mysql.jdbc.Driver");
342             conn=DriverManager.getConnection(DB_URL,USER,PASS);
343             stmt = conn.createStatement();
344             stmt.executeUpdate("insert into requirement (e_id,r_id,r_money,r_materials,r_volunteers) values ('"+eid+"','"+rid+"','"+rMoney+"','"+rMaterials+"','"+rVolunteers+"')");
345             JOptionPane.showMessageDialog(null,"Inserted Successfully!");
346
347             new MainPage().setVisible(true);
348             conn.close();
349         }
350         catch(Exception e)
351         {
352             JOptionPane.showMessageDialog(null,e);
353         }
354     }
355 }
356
357 }
358 }
359 }
360 }
361

```

The insertion code is used in try – catch block along with connectivity code. Insertion code is used to insert requirement data into database. The function used is given below.

`stmt.executeUpdate();`

- 3) **Outcome:** Once the event is conducted successfully the outcome of the event will be updated by admin by filling details like Event status and description.



The screenshot shows the NetBeans IDE interface with the title bar "PAC - NetBeans IDE 8.1". The menu bar includes File, Edit, View, Navigate, Source, Refactor, Run, Debug, Profile, Team, Tools, Window, Help. The toolbar has icons for file operations like Open, Save, and Build. The top status bar says "Search (Ctrl+F)". The left sidebar has "Projects" and "Services" tabs, with "Files" selected, showing a list of Java files: DQ2.java, DQ3.java, DQ4.java, Eve.java, Event.java, FB.java, Feedback.java, Home.java, HomeMain.java, Idea.java, Idea1.java, Login.java, MainPage.java, OC.java, Outcome.java, Page.java, Pro.java, Problem.java, Q1.java, Queris.java, Registration.java, Req.java, Requirement.java, Spo.java, Sponsor.java, UpdateEvent.java, UpdateReq.java, UpdateUser.java, UseIdea.java, ViewE.java, ViewEvent.java, ViewFeedback.java. The main editor window displays Java code for an "Event.java" file. The code includes a try-catch block for database connectivity and insertion:

```

else
{
    try
    {
        Class.forName("com.mysql.jdbc.Driver");
        conn=DriverManager.getConnection(DB_URL,USER,PASS);
        stmt = conn.createStatement();
        stmt.executeUpdate("insert into outcome (e_id,o_status,o_description) values ('"+eid+"','"+st+"','"+desc+"')");
        JOptionPane.showMessageDialog(null,"Inserted Successfully!");
        new MainPage().setVisible(true);
        conn.close();
    }
    catch(Exception e)
    {
        JOptionPane.showMessageDialog(null,e);
    }
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}

```

The bottom status bar shows "Activate Windows" and "Go to Settings to activate V". The taskbar at the bottom has icons for Start, Search, Task View, Edge, File Explorer, OneDrive, Task Manager, File History, and File Explorer.

The insertion code is used in try – catch block along with connectivity code. Insertion code is used to insert outcome data into database. The function used is given below.

`stmt.executeUpdate();`

6. Software Testing

Test cases

Unit Testing

The unit testing focuses verification effort on the smallest unit of the software design module. In unit testing different modules are tested against the specifications produced during the design of module. Mainly there are 2 Modules in our application, they are USER and ADMIN all the units of these two modules are tested against the requirement specification.

Integration Testing

After the unit testing integration testing is carried out, the individual units are combined to form component.

In the integration testing the testing modules are combined in to subsystems, which are then tested. Integration testing refers to the retesting components/functionality of the system to ensure that they function properly even after a change has been made to parts of the system. As defects are discovered in a component, modification are made to correct them.

System Testing

Here all the components are integrated and tested as whole system.

In system testing the entire software system is tested. The reference document for this process is requirement document. It also tests to find the discrepancies between the system and the original objective, current specification and system documentation. The entire system is checked to correct deviation to achieve correctness.

Validation Testing

Software validation is achieved through the black box tests that demonstrate the conformity with the requirement. Both plan and procedures are designed to ensure that all the functional requirements are satisfied.

Validation with respect to each form is done to stop the user filling the invalid data which leads to SQL injection.

BLACK -BOX TESTING

In black-box testing, the tester only knows the inputs that can be given to the system and what output the system should give. In other words, the basis for deciding test cases is the requirements or specifications of the system or module. This form of testing is also called functional or behavioral testing.

Testing and Validations

1. REGISTRATION PAGE

Registration id is auto generated.

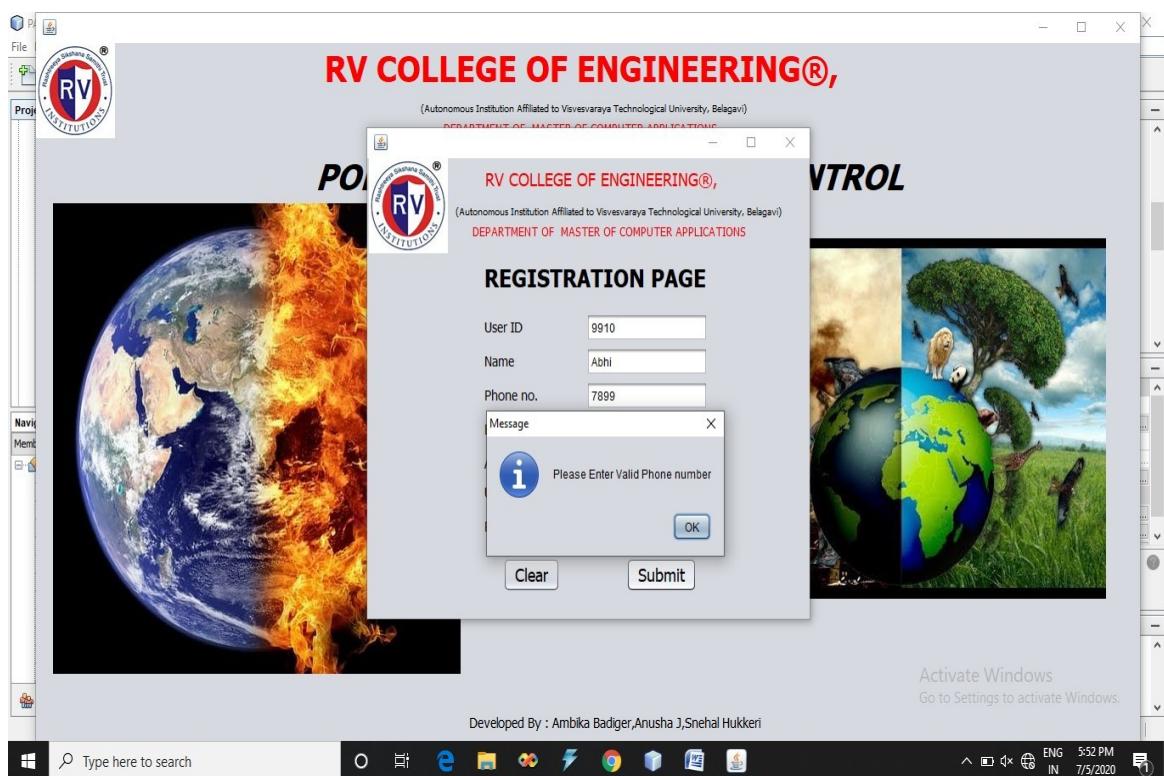
Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass/Fail)
1	Name	Test the name field with correct validation	1.Enter a valid name. 2..Click in submit button.	After entering valid name it must be accepted.	Accepted name	Pass

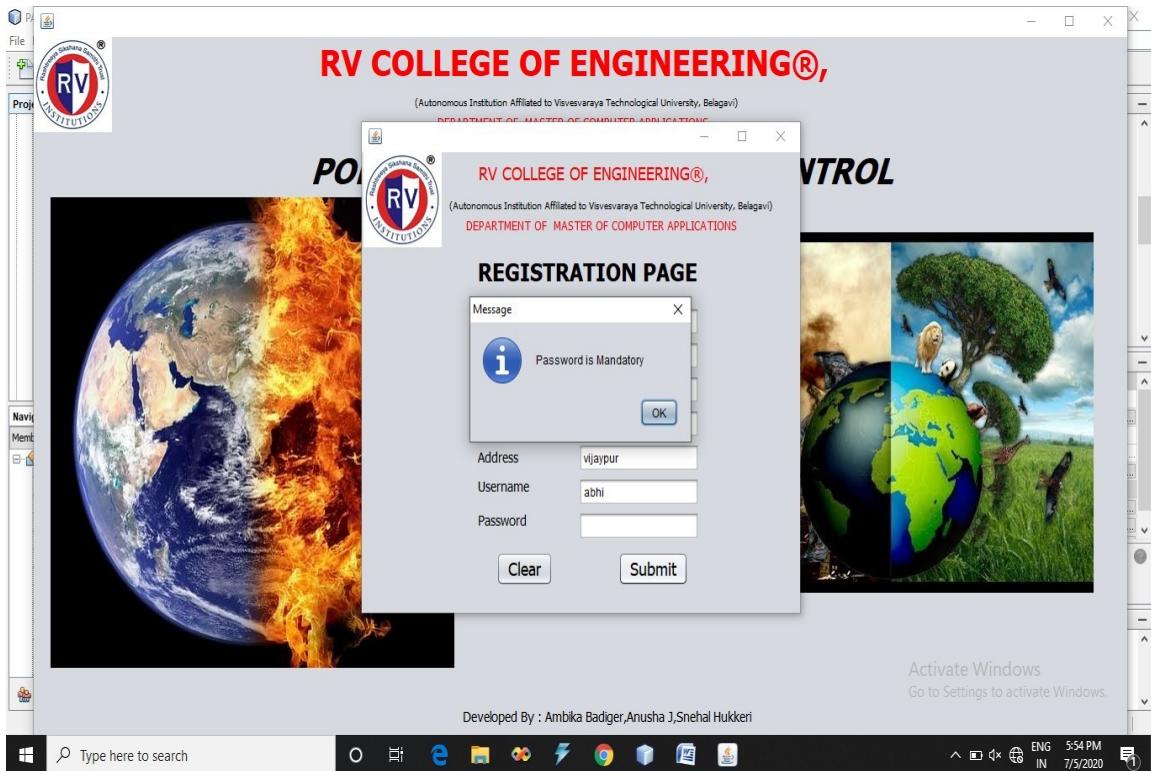
		Test the name with incorrect validation.	1.Enter a invalid name like numbers. 2.Click on submit button.	After entering invalid name it must not be accepted.	Name is not accepted	Pass
		Test the name field without entering the field.	Direct click on submit button.	It should display a proper error message	Form is not submitted.	Pass
2	Email	Test the email field with correct validation.	1.Enter a valid email. 2.Click on submit button.	After entering the correct email it must be accepted.	Accepted email.	Pass
		Test the email field with incorrect	1.Enter a invalid email(with out @ or	After entering the incorrect email it must	Email is not accepted	Pass

		validation. 2.click on submit button.	.com) not be accepted.		
		Test the email field without entering the field.	Direct click on submit button.	It should display error message.	Form is not submitted. Pass
3	Mobile number	Test the mobile number field with correct validation	1.Enter a valid mobile no. 2.Click in submit button.	After entering valid mobile no. it must be accepted.	Accepted mobile no. Pass
		Test the mobile no with incorrect validation.	1.Enter a invalid mobile no.(entering numbers which has	After entering invalid mobile no. it must not be accepted.	Mobile no is not accepted. Pass

			greater than 10 digits and having charaters) 2.Click on submit button.			
		Test the mobile no field without entering the field.	Direct click on submit button.	It should display error message.	Form is not submitted	Pass
4	Username	3.Test username without entering the field.	Detect click on Submit button.	it should display a proper Error message(Plea se fill out this field).	Form is not submitted	Pass
5	Password	3.Test Password	Direct click on submit	it should display a	Form is not submitted.	Pass

		without entering the field.	button.	proper Error message(Please fill out this field).		
	Test if user is able to Register successfully.	1 .Test all fields.	1. Enter all valid Data Fields. 2. Click on Submit button	User must successfully Register.	User has Registered successfully	Pass



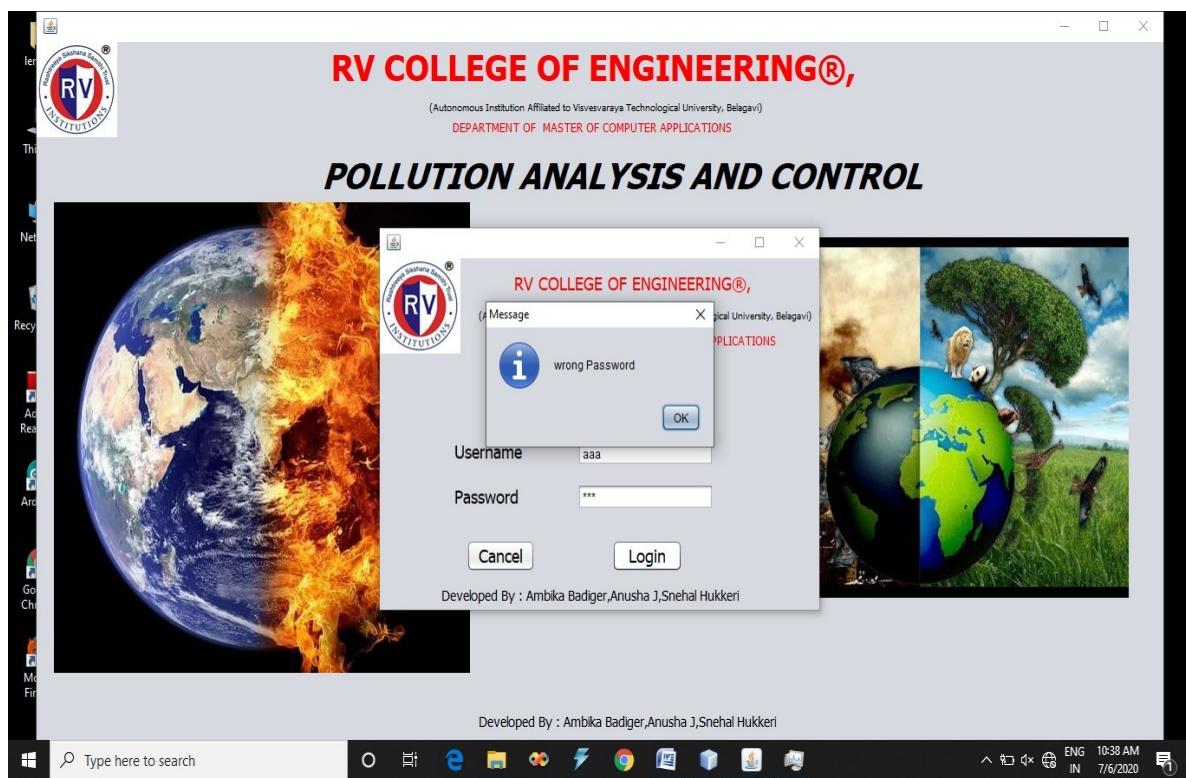
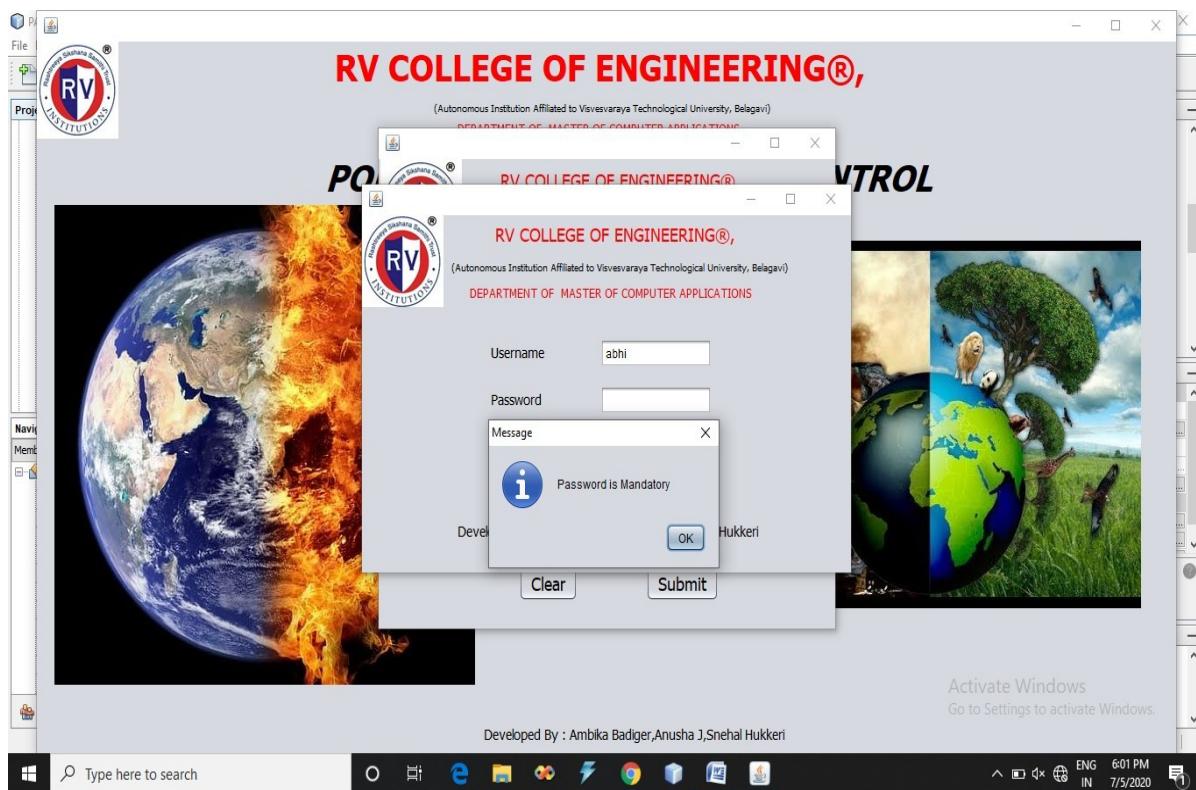


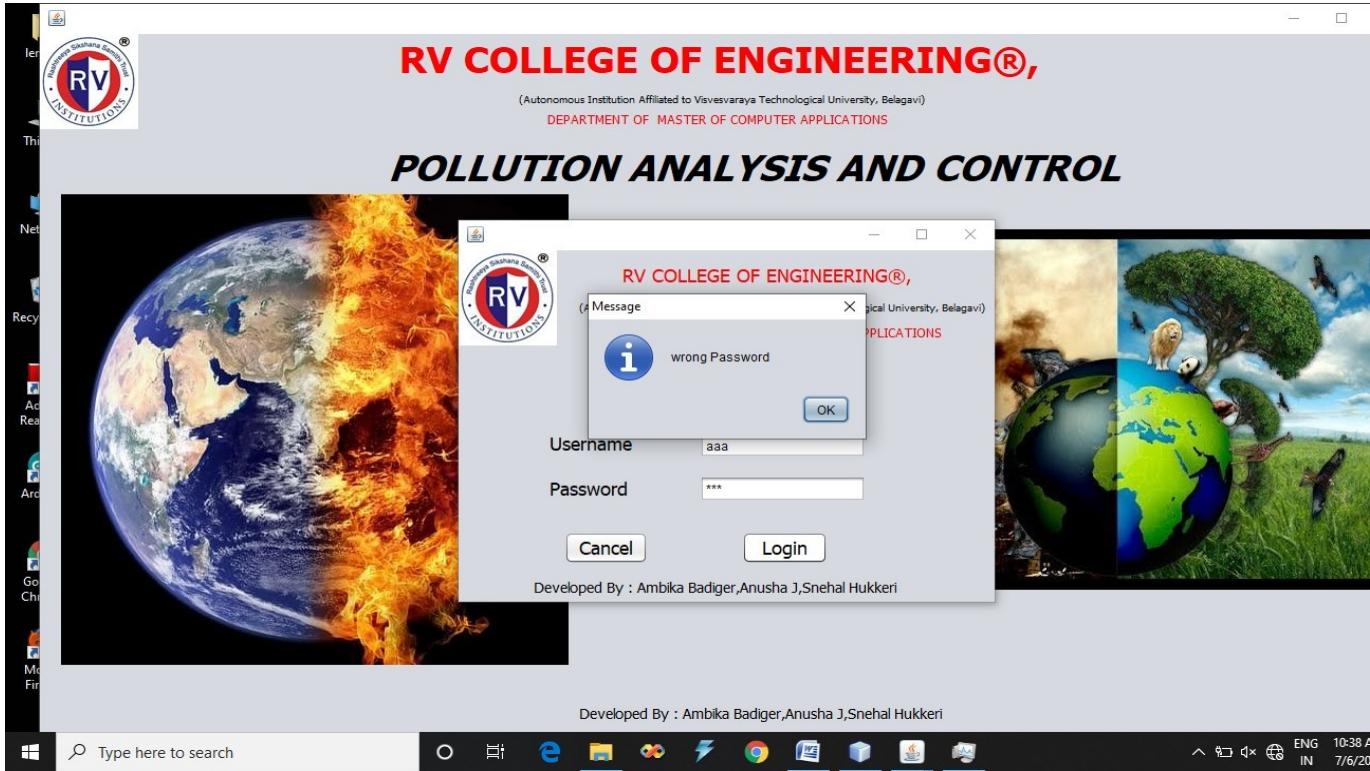
2.LOGIN PAGE

Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass /Fail)
1	username	Test the username with incorrect validation.	1.Enter a invalid username. 2.Click on submit button.	After entering invalidusername. it must not be accepted.	username is not accepted.	Pass
		Test username without entering the field.	Direct click on Submit button.	It should display a proper Error message (username field is mandatory	Form is not submitted	Pass

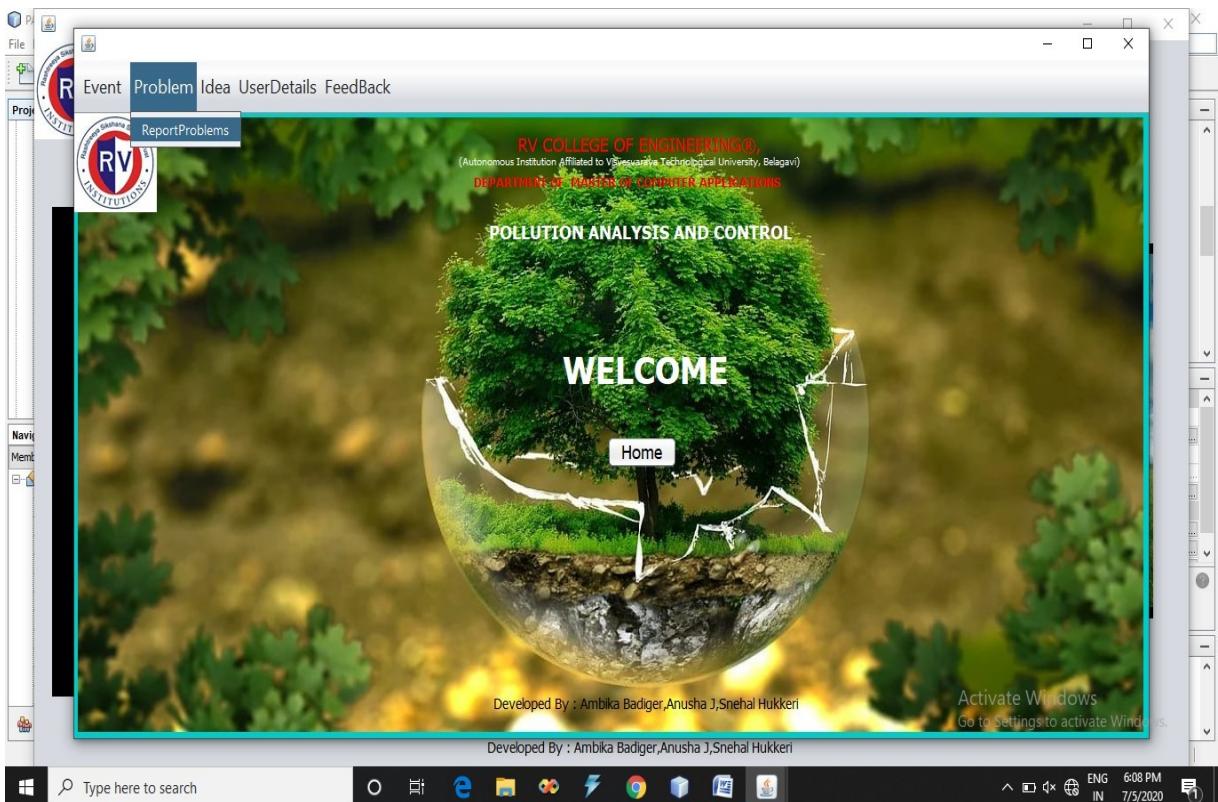
).).		
		Test the username field with correct validation	1.Enter a valid username. 2.Click in submit button.	After entering valid username. it must be accepted.	Accepted username .	Pass
2	Password	Test Password without entering the field.	Direct click on submit button.	it should display a proper Error message(password field is mandatory).	Form is not submitted .	Pass
		Test the password with incorrect validation.	1.Enter a invalid password. 2.Click on submit	After entering invalid password. it must not	password is not accepted.	Pass

			button.	be accepted.		
		Test the password field with correct validation	1.Enter a valid password. 2.Click in submit button.	After entering valid password. it must be accepted.	Accepted password .	Pass
3.	Test if user/Admin is able to login successfully.	1 .Test all fields.	1. Enter all valid Data Fields. 2. Click on Submit button	User/Admin must successfuly Login.	User/Admin has login successfu lly	Pass

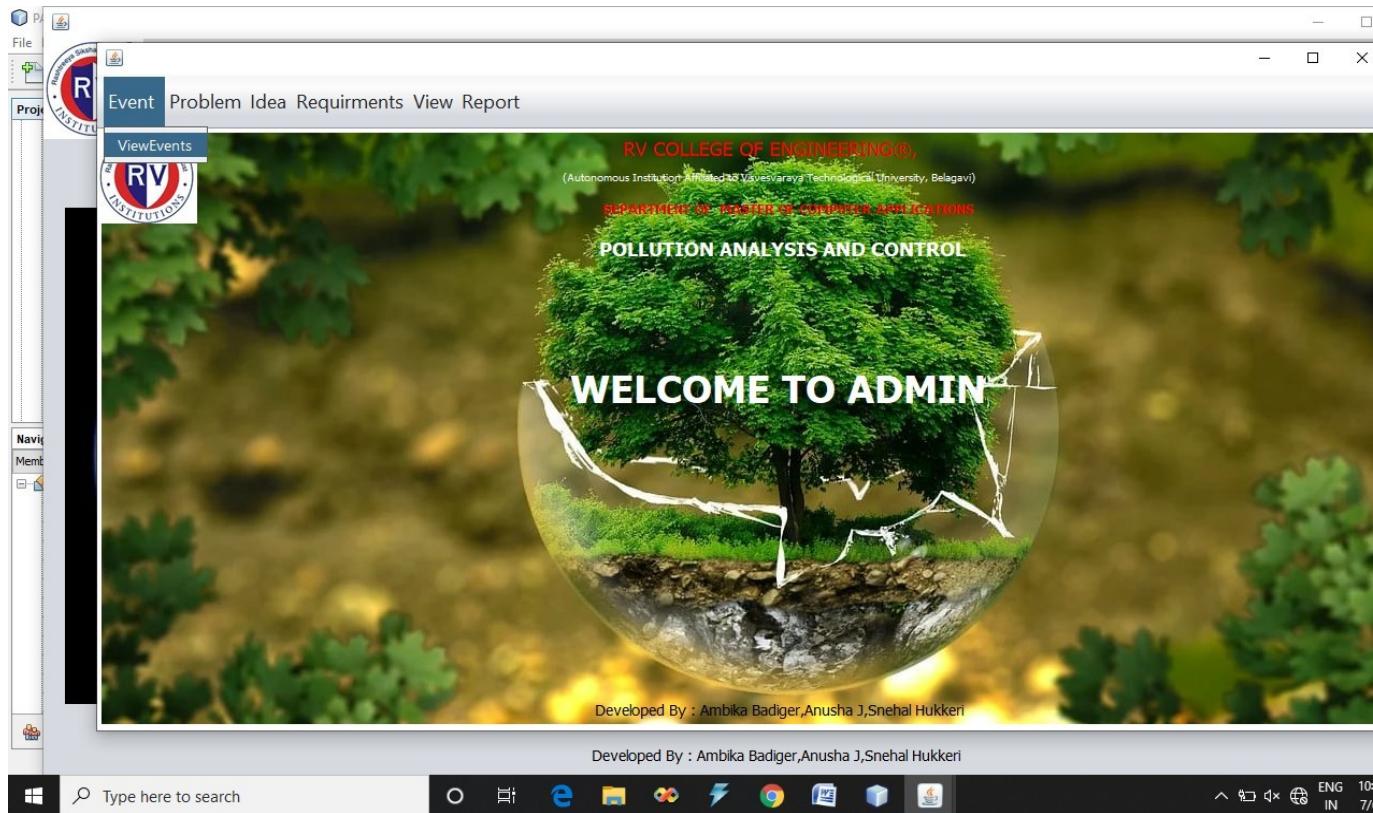




USER HOME



ADMIN HOME



3. Problem Report

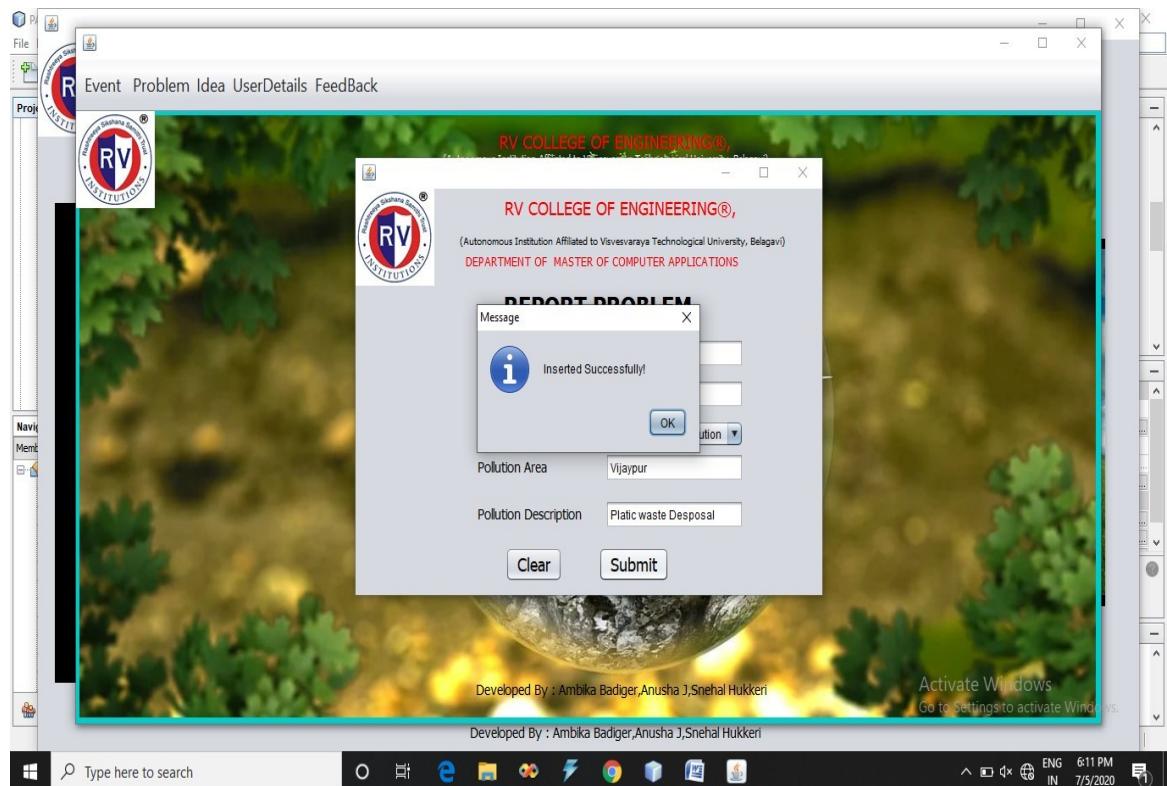
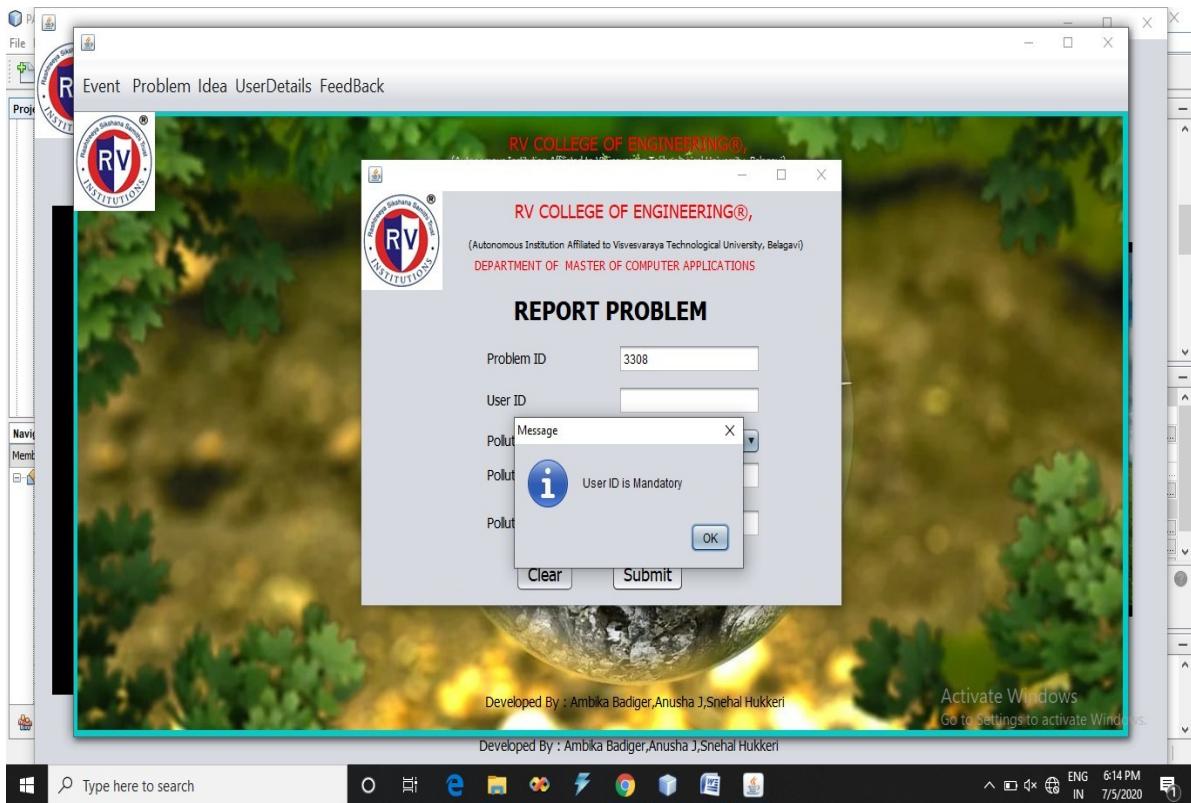
Problem id is auto generated

Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass/Fail)

1	User Id	Test the userid with incorrect validation.	1. Enter a invalid userid. 2. Click on submit button.	After entering invalid userid. it must not be accepted.	userid is not accepted.	Pass
		Test userid without entering the field.	Direct click on Submit button.	It should display a proper Error message (userid field is mandatory).	Form is not submitted	Pass
		Test the userid field with correct validation	1. Enter a valid userid. 2. Click in submit button.	After entering valid userid. it must be accepted.	Accepted userid.	Pass

2	Pollution Type	Test Pollution type field without selecting field from dropdown list .	Direct click on submit button.	it should display a proper Error message(pollution type field is mandatory).	Form is not submitted.	Pass
		Test the pollution type with incorrect validation.	1. Enter a invalid pollution type. 2. Click on submit button.	After entering invalid pollution type. it must not be accepted.	password is not accepted.	Pass
		Test the pollution type field with correct	1. Enter a valid password. 2. Click in	After entering valid pollution	Accepted pollution type.	Pass

		validation	submit button.	type. it must be accepted.		
3.	Test if user is able to report problem successfully.	1 .Test all fields.	1. Enter all valid Data Fields. 2. Click on Submit button	User must successfully report problem.	User has reported problem successfully	Pass



4. Event

Event id is auto generated

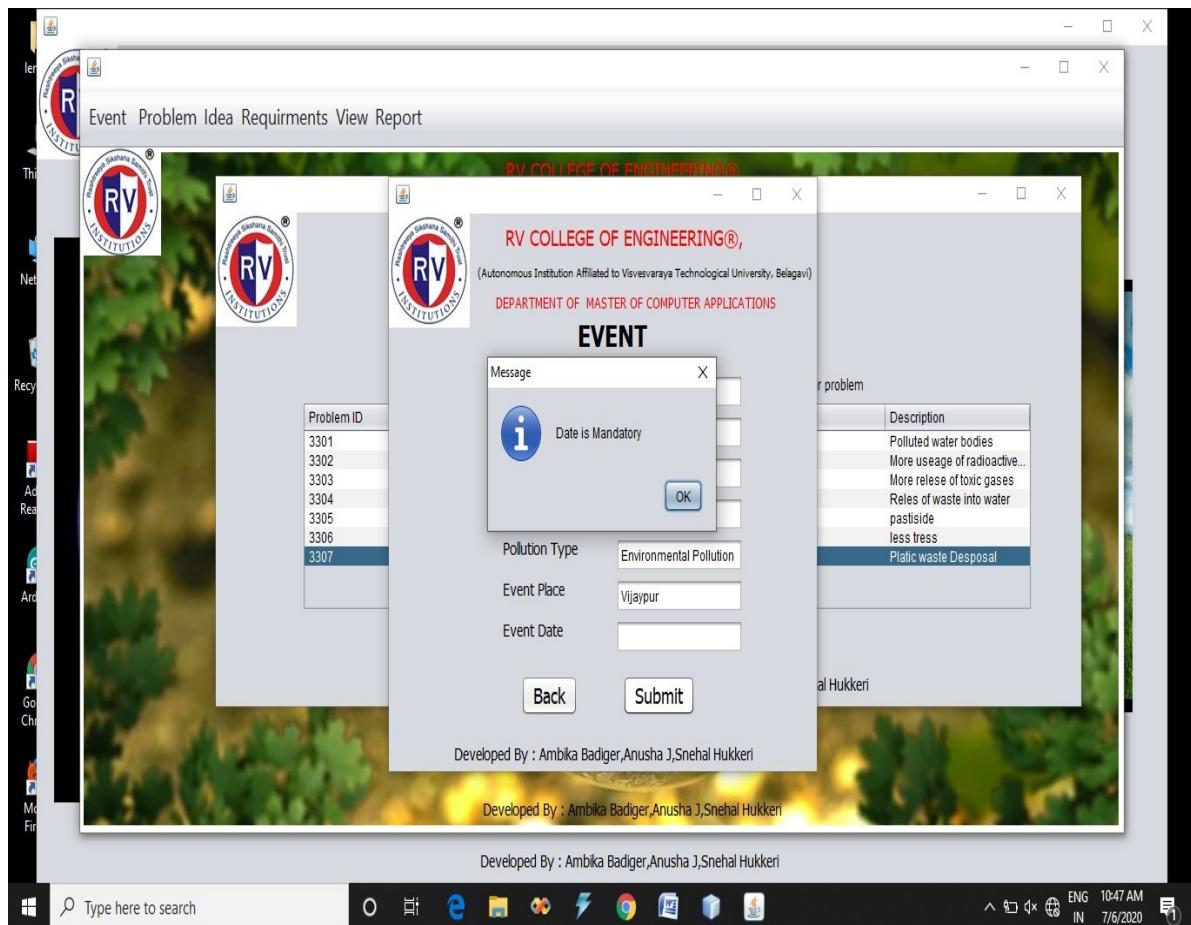
Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass/Fail)
1	Admin id	Test the admin id with incorrect validation.	1. Enter a invalid admin id. 2. Click on submit button.	After entering invalid admin id. it must not be accepted.	Admin id is not accepted.	Pass
		Test admin id without entering the field.	Direct click on Submit button.	It should display a proper Error message (Admin id	Form is not submitted	Pass

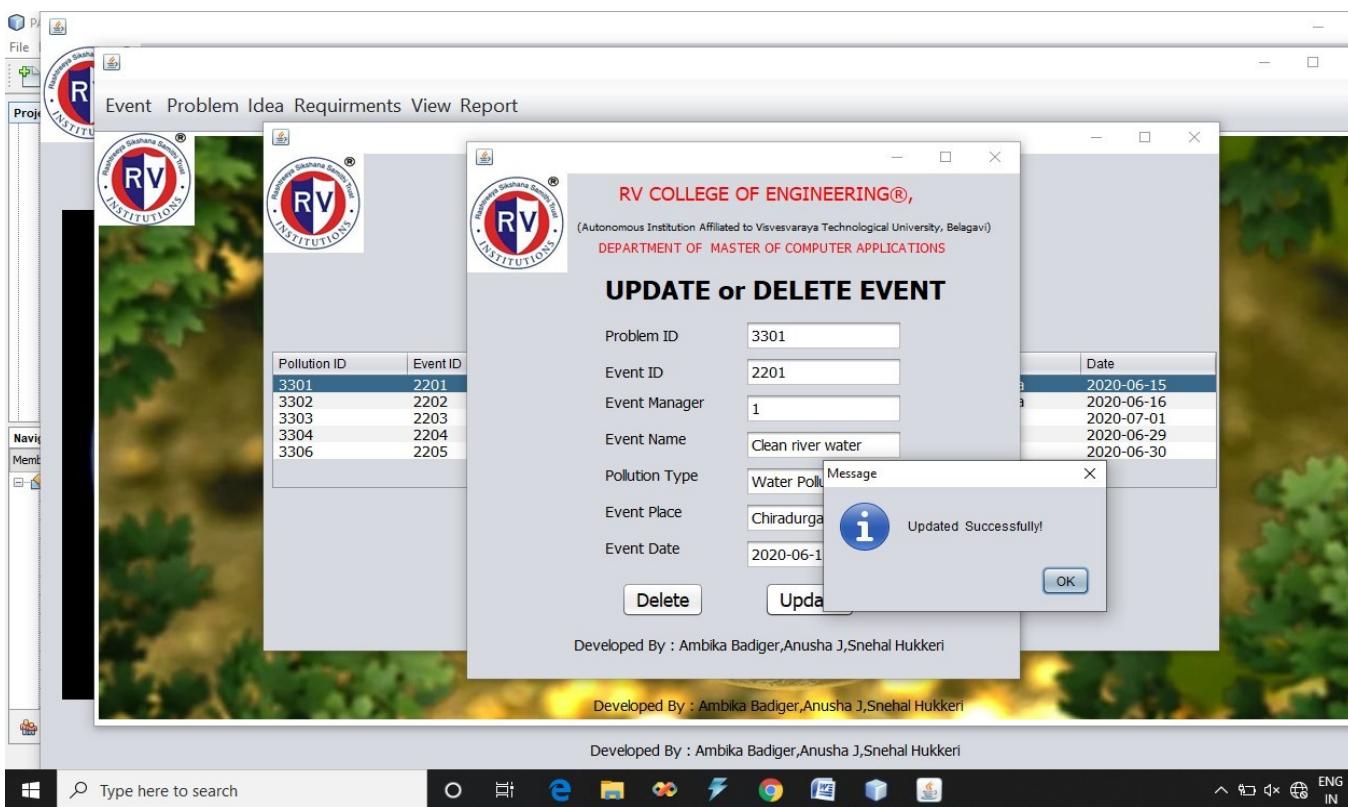
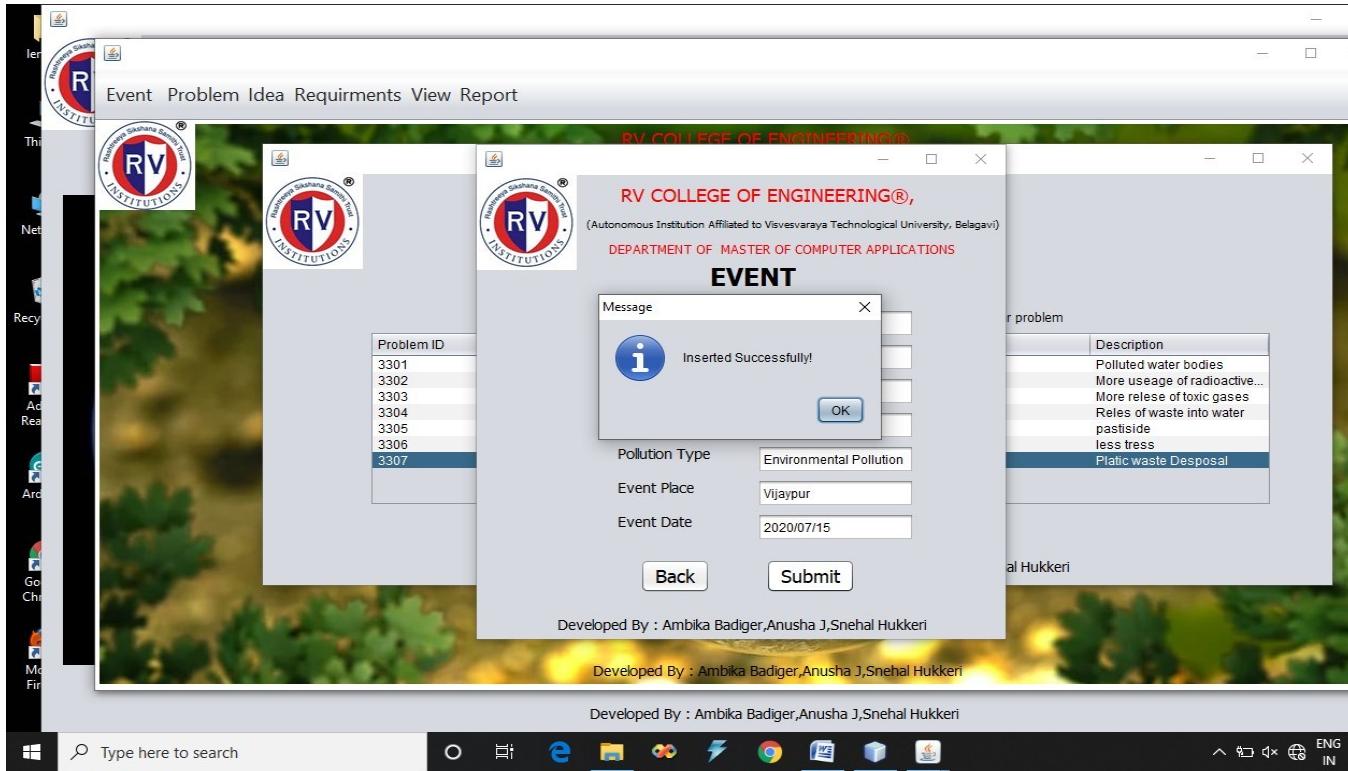
				field is mandatory) .		
		Test the admin id field with correct validation	1. Enter a valid admin id. 2. Click in submit button.	After entering valid admin id. it must be accepted.	Accepted admin id.	Pass
2	Event name	Test Event name without entering the field.	Direct click on submit button.	it should display a proper Error message(Event name field is mandatory) .	Form is not submitted.	Pass

		Test the Event name field with correct validation.	1. Enter a event name. 2. Click in submit button.	After entering event name. it must be accepted.	Accepted event name.	Pass
3	Event Type	Test Event type without selecting the field from dropdown list.	Direct click on submit button.	It should display a proper Error message(Event type field is mandatory).	Form is not submitted.	Pass
		Test the Event name field with correct validation.	1. Select a event type. 2. Click in submit button.	After selecting event type. it must be accepted.	Accepted event type.	Pass

4	Event place	Test Event place without entering the field.	Direct click on submit button.	it should display a proper Error message(Event place field is mandatory).	Form is not submitted.	Pass
		Test the Event place field with correct validation	1. Enter a event place. 2. Click in submit button.	After entering event place. it must be accepted.	Accepted event place.	Pass
5	Event date	Test Event date without entering the field.	Direct click on submit button.	it should display a proper Error message(Event date	Form is not submitted.	Pass

				field is mandatory).		
		Test the Event date field with correct validation	1. Enter a event date. 2. Click in submit button.	After entering event date. it must be accepted.	Accepted event date.	Pass
6.	Test if admin is able to organiz e event successfully.	1 .Test all fields.	1. Enter all valid Data Fields. 2. Click on Submit button	Admin must successfull y organize event.	Admin has organized event successfully	Pass





5. REQUIREMENT

Requirement id is auto generated.

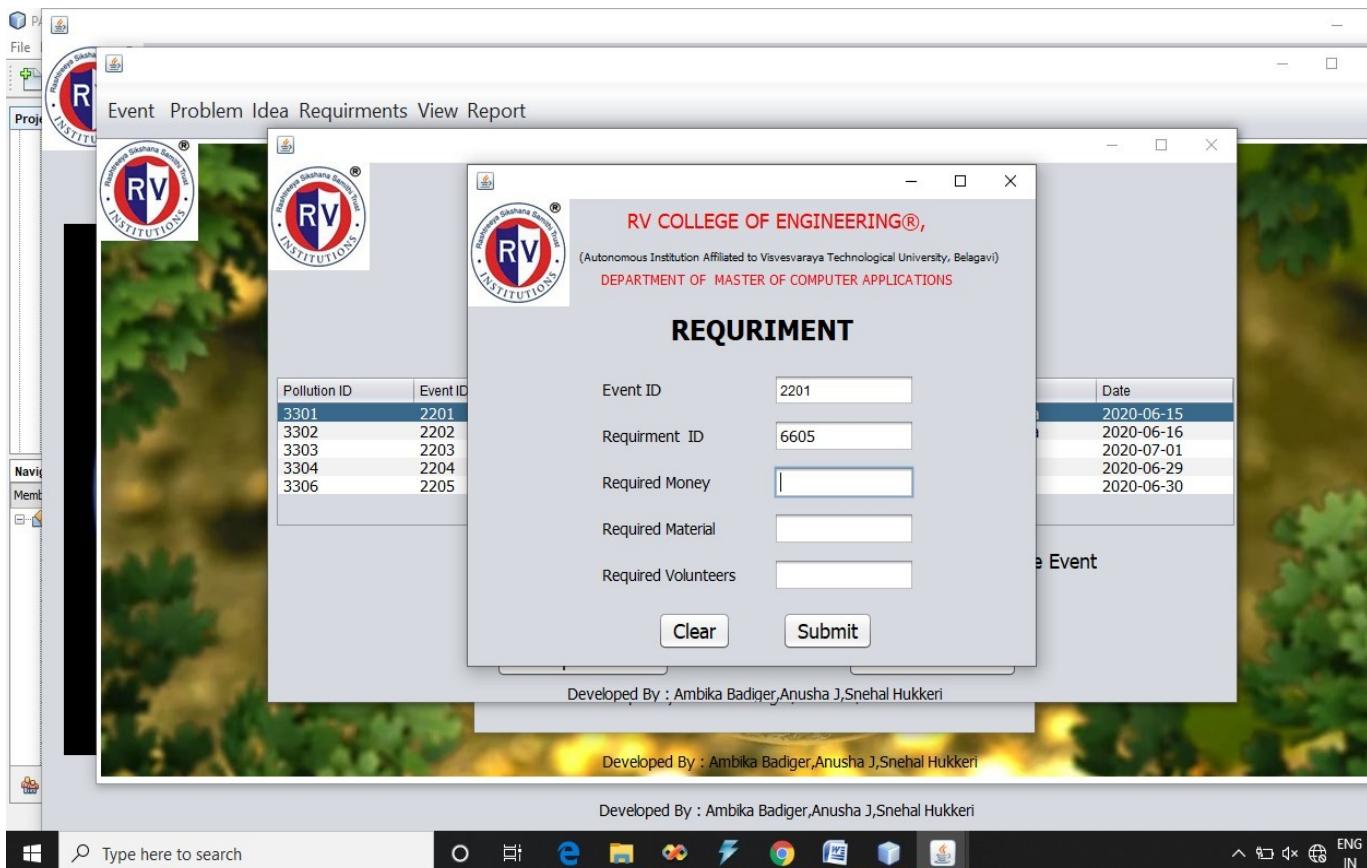
Event id fetched from database.

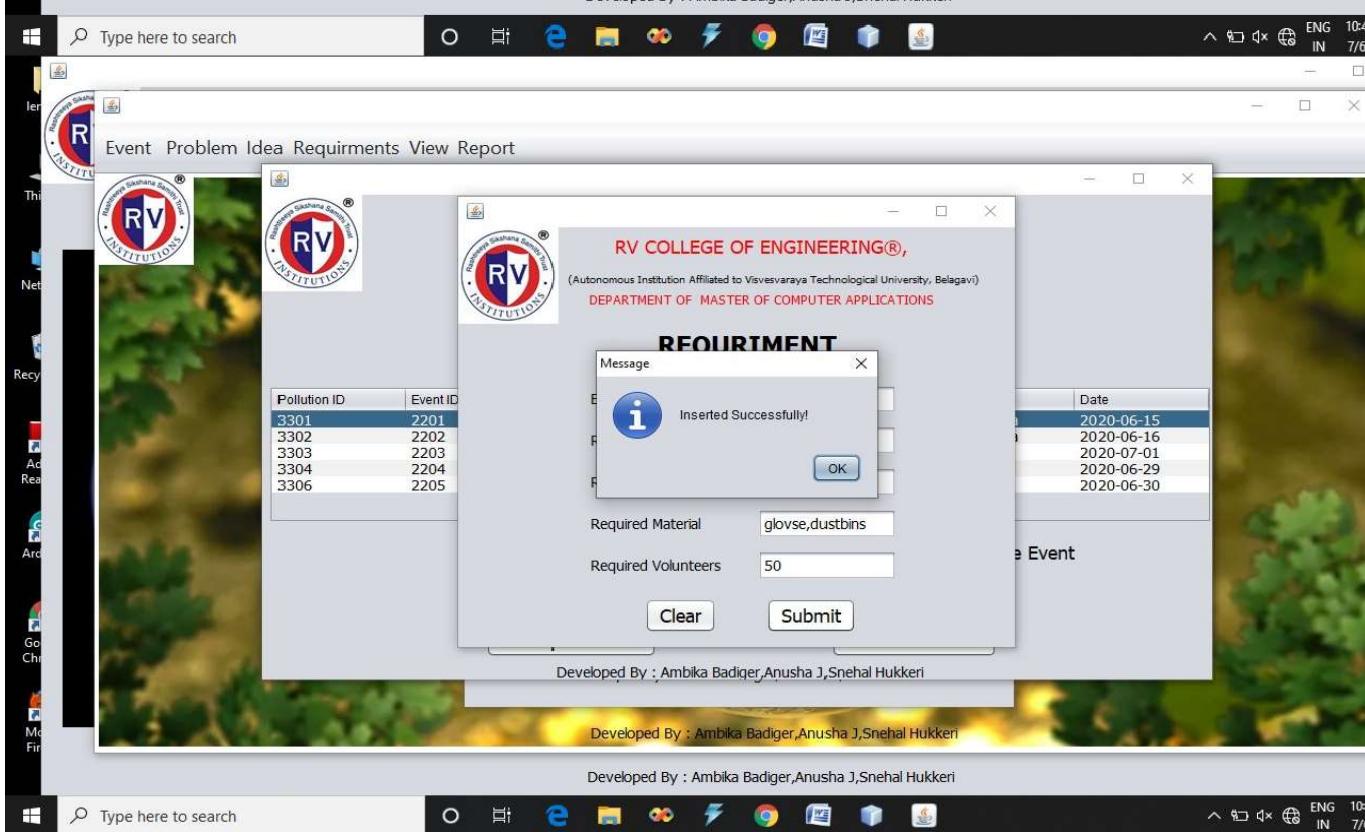
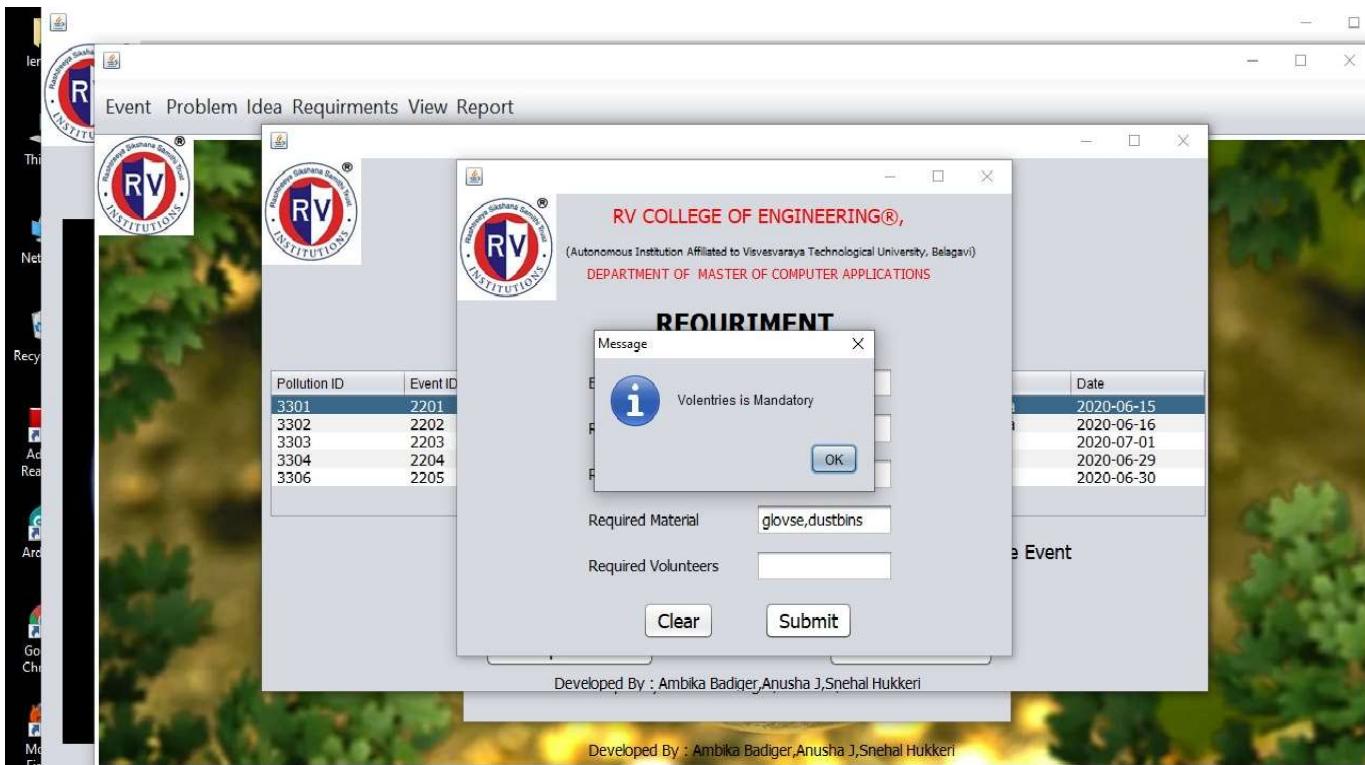
Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass/Fail)
1	Required money	Test required money without entering the field.	Direct click on Submit button.	It should display a proper Error message (Required money field is mandatory).	Form is not submitted	Pass
		Test the Required money field with correct validation	1. Enter a valid required money. 2. Click in	After entering valid required money it	Accepted Required money.	Pass

			submit button.	must be accepted.		
2	Required material	Test required material without entering the field.	Direct click on submit button.	it should display a proper Error message(Required material field is mandatory).	Form is not submitted.	Pass
		Test the Required material field with correct validation	1. Enter a required material. 2. Click in submit button.	After entering required material. it must be accepted.	Accepted required material.	Pass
3	Required volunteers	Test required volunteers without entering the	Direct click on submit button.	it should display a proper Error message(Required	Form is not submitted.	Pass

		field.		volunteer field is mandatory).		
		Test the Required volunteers field with correct validation	1.Enter a required volunteer. 2.Click in submit button.	After entering required volunteer. it must be accepted.	Accepted required volunteers .	Pass
4	Test if admin is able to specify requirement	1 .Test all fields.	1. Enter all valid Data Fields. 2. Click on Submit button	Admin must successfully specify requirements.	Admin has specified requirements successfully	Pass

	of event succes sfully.					
--	----------------------------------	--	--	--	--	--





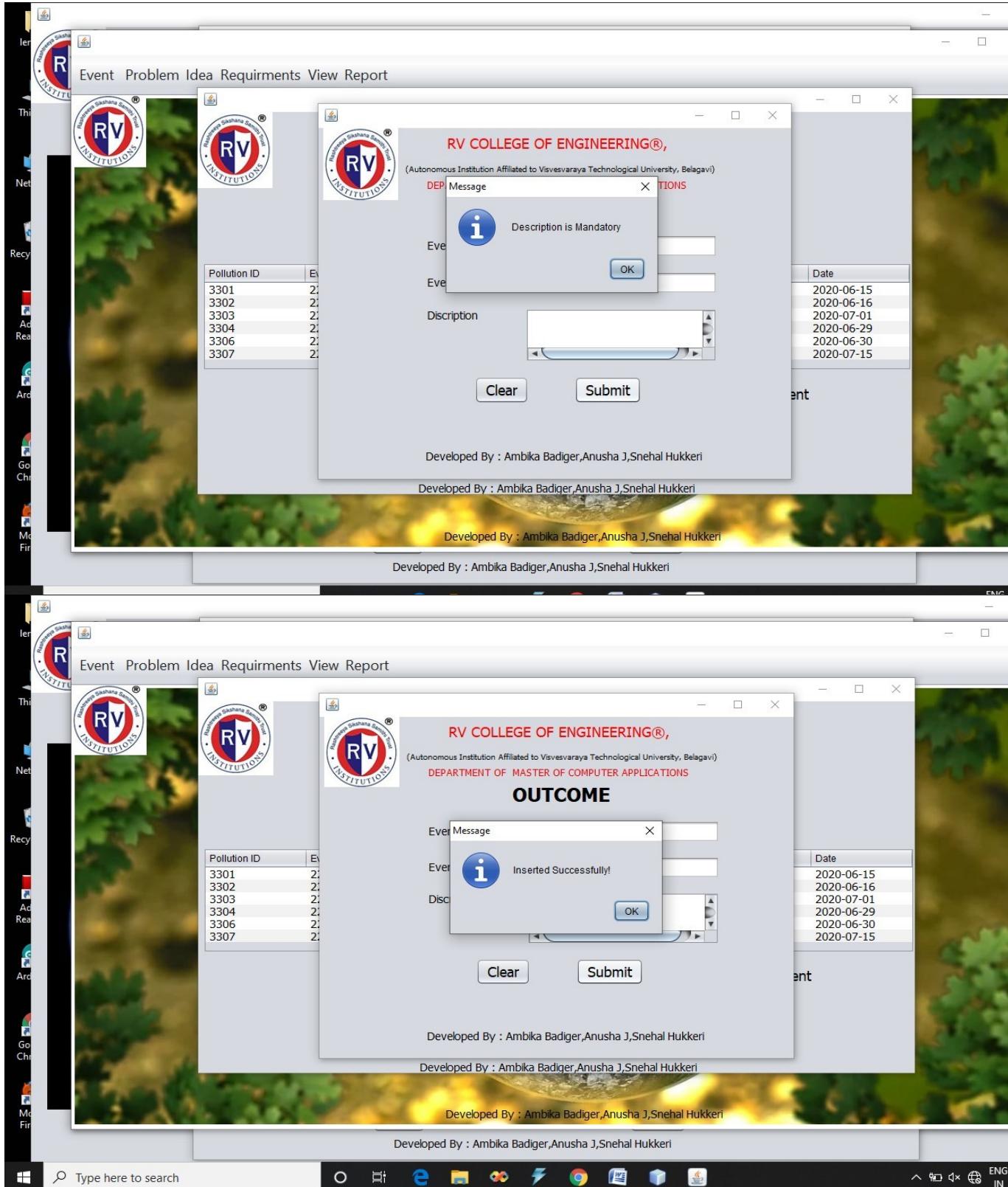
6.OUTCOME

Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass/Fail)
1	Event id	Test the Event id with incorrect validation.	1. Enter a invalid Event id. 2. Click on submit button.	After entering invalid Event id. it must not be accepted.	Event id is not accepted.	Pass
		Test event id without entering the field.	Direct click on Submit button.	It should display a proper Error message (Event id field is mandatory).	Form is not submitted	Pass

		Test the Event id field with correct validation	1. Enter a valid event id. 2. Click in submit button.	After entering valid event id. it must be accepted.	Accepted event id.	Pass
2	Status	Test Status without entering the field.	Direct click on Submit button.	It should display a proper Error message (Status field is mandatory).	Form is not submitted	Pass
		Test the Status field with correct validation	1. Enter a valid Status. 2. Click in submit button.	After entering Status it must be accepted.	Accepted Status.	Pass
3	Description	Test description	Direct click on	It should display a	Form is not	Pass

		without entering the field.	Submit button.	proper Error message (Description field is mandatory).	submitted	
		Test the Description field with correct validation	1.Enter a valid Description. 2.Click in submit button.	After entering Description it must be accepted.	Accepted	Pass
	Test if admin is able to update outcome of	1 .Test all fields.	1. Enter all valid Data Fields. 2. Click on Submit button	Admin must successfully update outcome.	Admin has updated status successfully	Pass

	the event succes sfully.					
--	-----------------------------------	--	--	--	--	--



7. IDEA

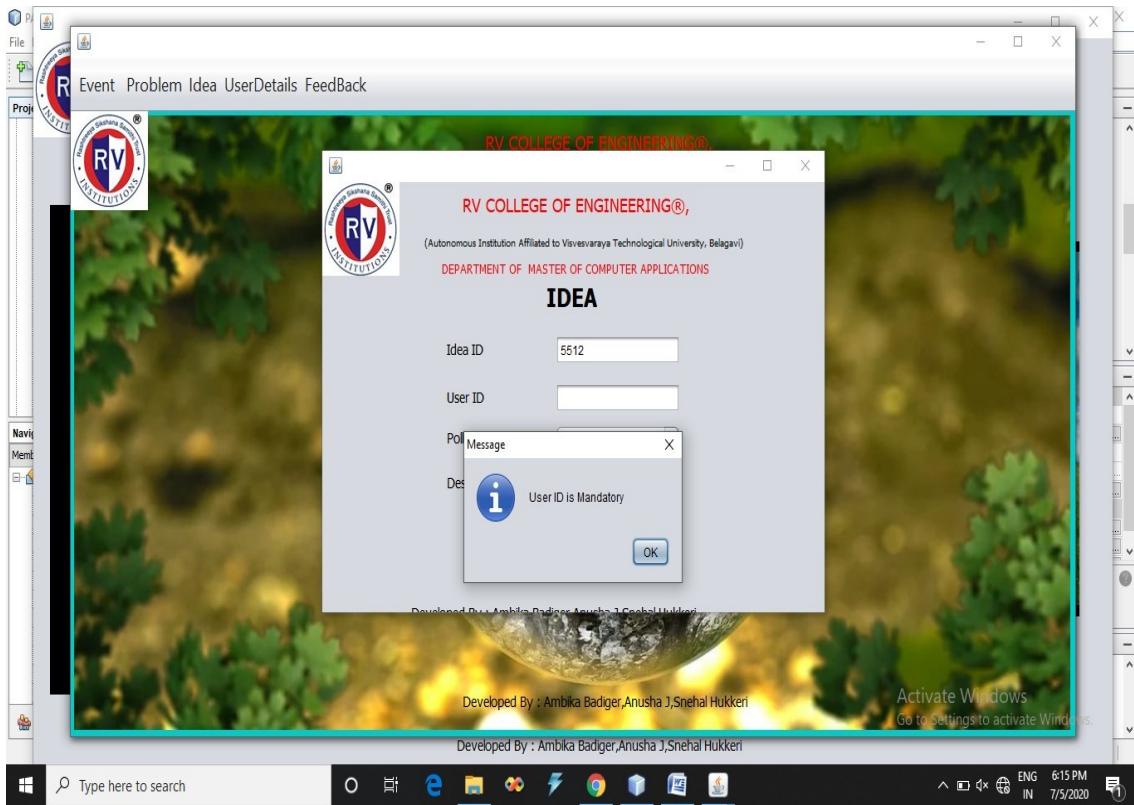
Idea id is auto generated.

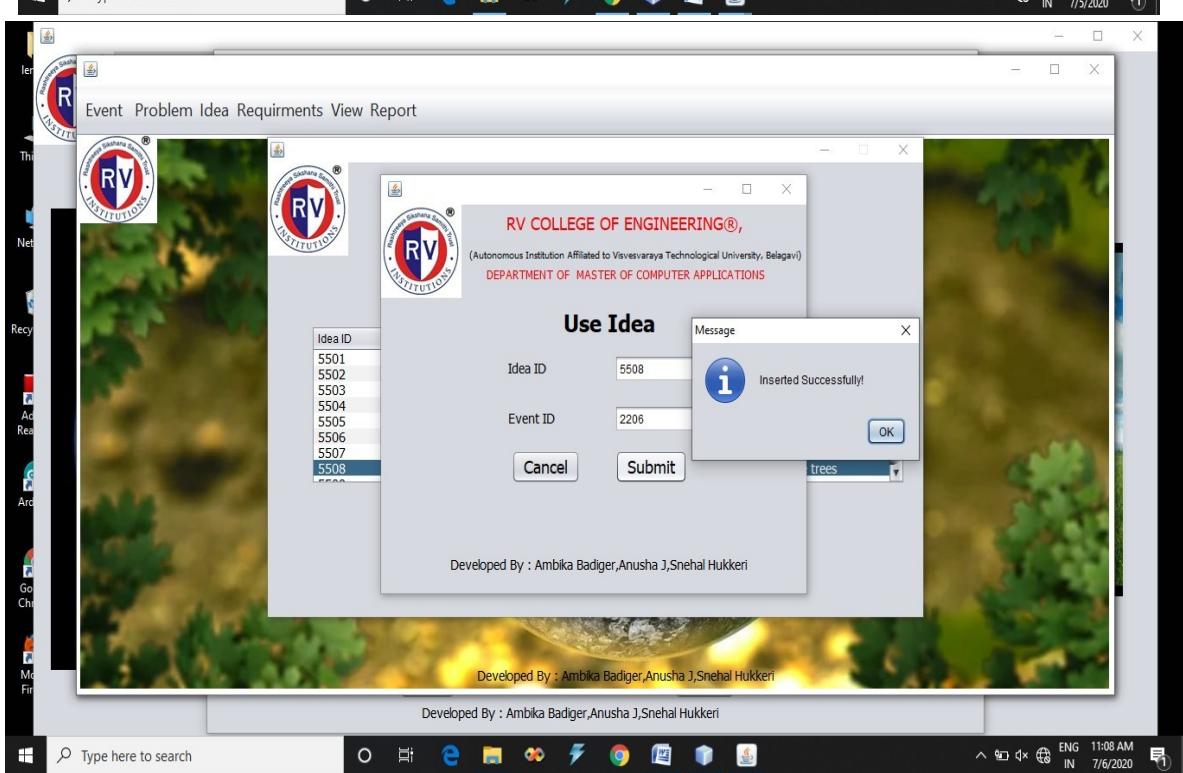
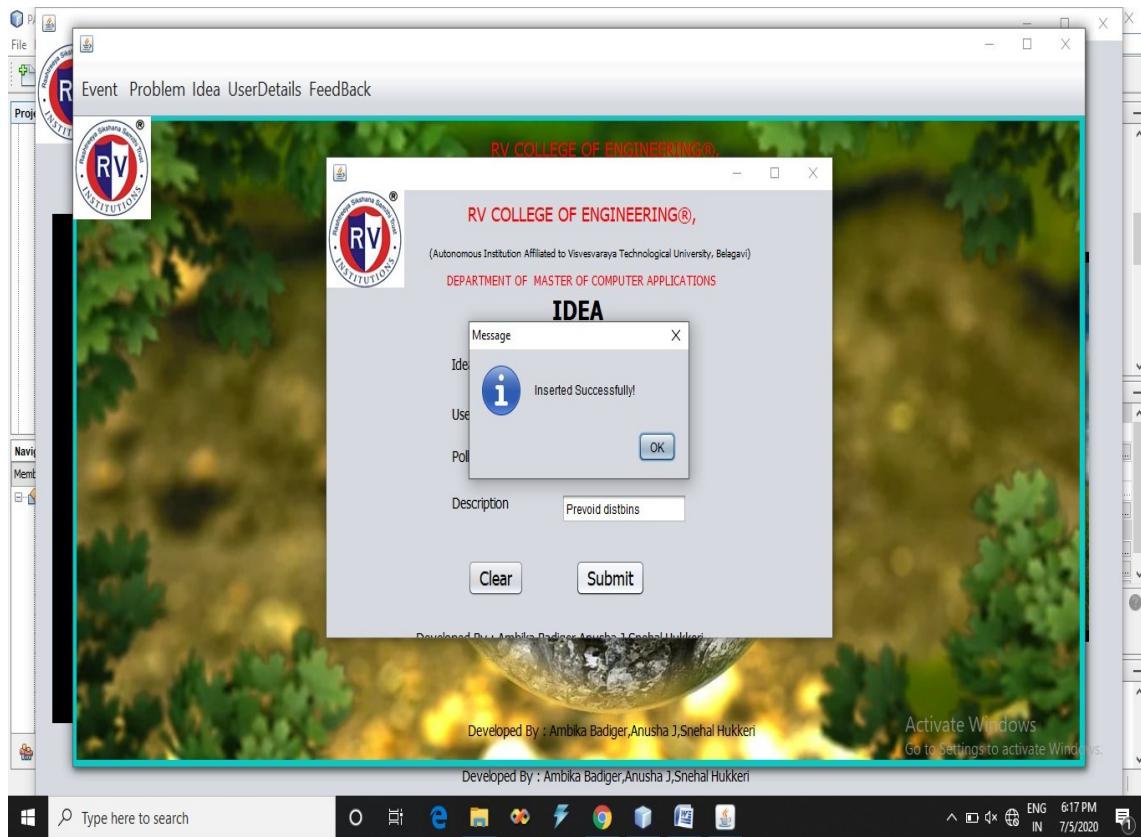
Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass/Fail)
1	User Id	Test the User id with incorrect validation.	1. Enter invalid User id. 2. Click on submit button.	After entering invalid user id. it must not be accepted.	User id is not accepted.	Pass
		Test user id without entering the field.	Direct click on Submit button.	It should display a proper Error message (user id field is mandatory).	Form is not submitted	Pass
		Test the user id field	1.Enter a valid user	After entering	Accepted user	Pass

		with correct validation	id. 2.Click in submit button.	valid user id. it must be accepted.	id.	
2	Pollution type	Test the pollution type field without selecting field from dropdown list .	Direct click on submit button.	it should display a proper Error message(pollution type field is mandatory).	Form is not submitted.	Pass
		Test the pollution type with incorrect validation.	1.Enter a invalid pollution type. 2.Click on submit button.	After entering invalid pollution type. it must not be accepted.	password is not accepted.	Pass
		Test the pollution	1.Enter a valid	After entering	Accepted	Pass

		type field with correct validation	password. 2.Click in submit button.	valid pollution type. it must be accepted.	pollution type.	
3.	Description	Test description without entering the field.	Direct click on Submit button.	It should display a proper Error message (Description field is mandatory).	Form is not submitted	Pass
		Test the Description field with correct validation	1.Enter a valid Description. 2.Click in submit button.	After entering Description it must be accepted.	Accepted Description.	Pass
4	Test if user is	1 .Test all fields.	1. Enter all valid Data	User must successfully	User has	Pass

	able to upload idea successfully.		Fields. 2. Click on Submit button	give idea.	given idea successfully	
--	-----------------------------------	--	--------------------------------------	------------	-------------------------	--





8.FEEDBACK

Feedback id is auto generated.

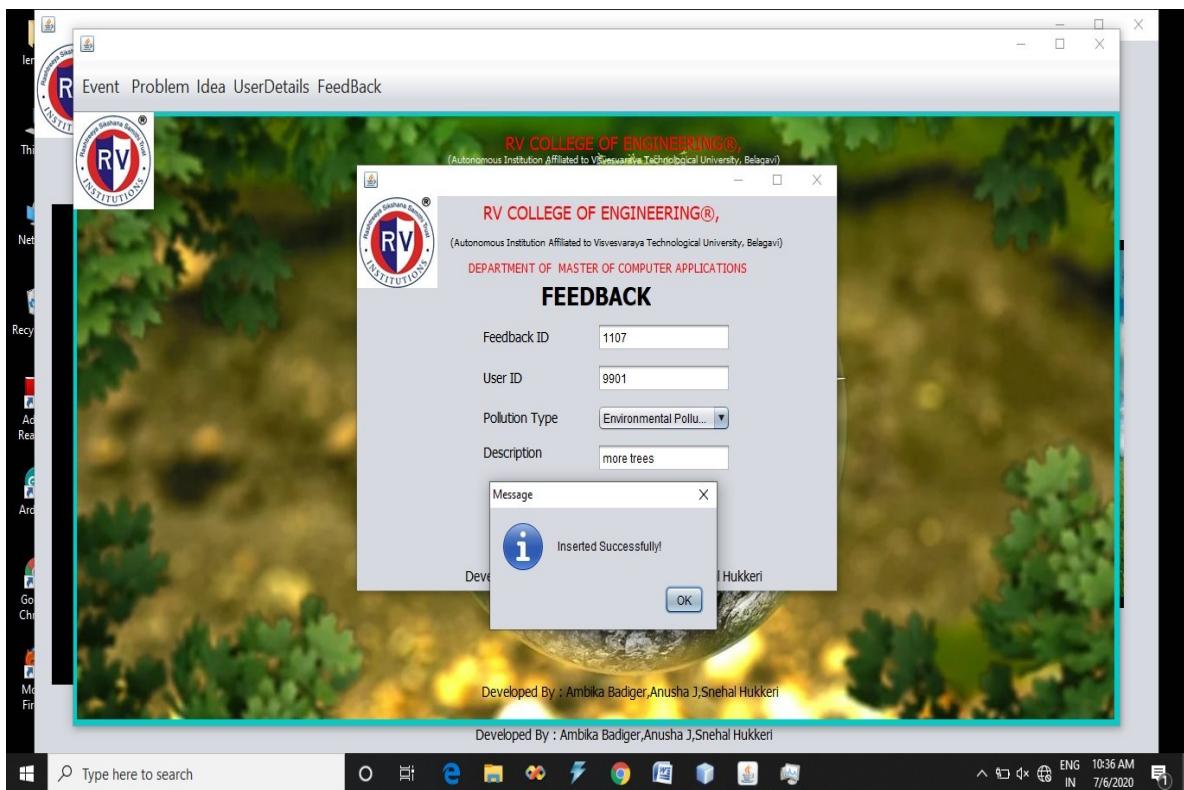
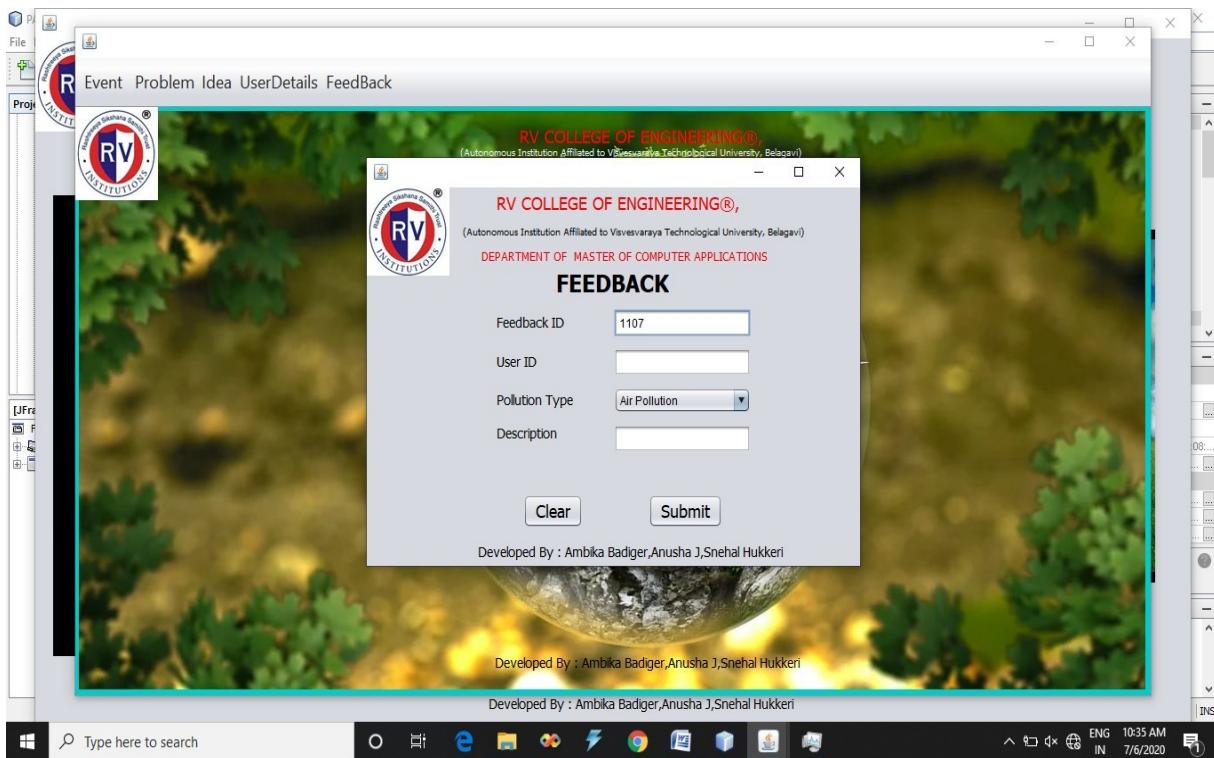
Test Case Id	Test Cases	Test Case Description	Steps To be Executed	Expected Results	Actual results	Status (Pass/Fail)
1	User Id	Test the User id with incorrect validation.	1. Enter invalid User id. 2. Click on submit button.	After entering invalid user id. it must not be accepted.	User id is not accepted.	Pass
		Test user id without entering the field.	Direct click on Submit button.	It should display a proper Error message	Form is not submitted	Pass

				(user id field is mandatory).		
		Test the user id field with correct validation	1. Enter a valid user id. 2. Click in submit button.	After entering valid user id. it must be accepted.	Accepted user id.	Pass
		Test the Required material field with correct validation	1. Enter a required material. 2. Click in submit button.	After entering required material. it must be accepted.	Accepted required material.	Pass
2	Pollution type	Test Pollution type field without selecting	Direct click on submit button.	it should display a proper Error message(p	Form is not submitted.	Pass

		field from dropdown list .		pollution type field is mandatory).		
		Test the pollution type with incorrect validation.	1. Enter a invalid pollution type. 2. Click on submit button.	After entering invalid pollution type. it must not be accepted.	password is not accepted .	Pass
		Test the pollution type field with correct validation	1. Enter a valid password. 2. Click in submit button.	After entering valid pollution type. it must be accepted.	Accepted pollution type.	Pass
3.	Descri	Test description	Direct click on Submit	It should display a	Form is not	Pass

	ption	without entering the field.	button.	proper Error message (Description field is mandatory).	submitted	
		Test the Description field with correct validation	1.Enter a valid Description . 2.Click in submit button.	After entering Description it must be accepted.	Accepted	Pass
4	Test if user is able to give feedba	1 .Test all fields.	1. Enter all valid Data Fields. 2. Click on Submit	User must successfully give feedback.	User has given feedback	Pass

	ck succes sfully.		button		ully	
--	-------------------------	--	--------	--	------	--



Evaluations:

In the attempt to evaluate the designed system, it is imperative that the predefined functionalities, goals and objectives are in relation to the expectations met by the system.

The main objective of our application is to solve the pollution related problem and trying to provide social awesomeness among people. As it collects the problems regarding various pollution types such as air pollution, soil pollution, water pollution, in various areas, reported by user. Based on these problems. Admins plan an appropriate event and organize in the area it was reported sometimes based on ideas provided by other users. These events are managed by one of the admins and by conducting events they try to solve reported problem. And conduct some other events to provide social awareness regarding prevention and controls of pollutions to people. As far as I concern the system met these expectations.

7. Conclusion & Future Enhancements

The System “Pollution Analysis and Control” is developed using the Software requirements and Software methodology, which helps to develop a system that is a user friendly. The software requirements specify the user requirements which are considered during development of the system. The system methodology helps in maintenance of the developed system. These results in a development of a user friendly system which helps to solve the problems. The Pollution Analysis and Control system helps users by organizing the specific event for solving the problems regarding pollutions happening in their area or locality when they post a problems regarding the issue.

In future we can enhance this application with more capabilities like analysis of data and generating reports. Based on analysis we can take correct decisions. We are going to keep track of every event we have organized to control pollution in that area. Thus this application would be helping us to control pollution.

Bibliography

1) Creator

Pritham thing

Title of the video "Registration form validation in java Netbeans with regular expression"

uploaded by " Pritham thing"

link "<https://youtu.be/sHqB8dtoVDs>"

["https://youtu.be/WeON7Ku1bTQ"](https://youtu.be/WeON7Ku1bTQ)

Published on march 15 2018

2) Creator

Gsoft knowledge

Title of the video "how to fetch or populate data from database to jframe or jTable - javatutorial#14"

link "<https://youtu.be/sHqB8dtoVDs>"

Published on april 07 2016

3) Creator

Mouna Naravani

Title of the video

a) "how to display the database records in jTable using Netbeans "

b)"registration form validation in java Netbeans"

uploaded by " Mouna Naravani"

link a)<https://youtu.be/4nvTwvfWt7I>

b)<https://youtu.be/54yGDNLjzro>

Published on December 07 2017

4) Creator

knowledge to share

Title of the video "JAVA SWING | jTextField phone number validation | accept only numbers(with 10 digits)"

link "<https://youtu.be/NXi-atvW73o>"

Published on December 30 2018

More References

- 1) <https://stackoverflow.com/questions/21898053/display-records-from-mysql-database-using-jtable-in-java>
- 2)
<https://stackoverflow.com/questions/13172898/how-to-autoincrement-string-in-java-swing>
- 3)
<https://stackoverflow.com/questions/9612096/i-want-to-make-a-text-field-editable-only-when-a-check-box-is-selected-in-netbea>

