

DEVELOPMENT OF REAL TIME SYSTEM IN OBJECT ORIENTED PROGRAMMING WITH JAVA AND UML DESIGNS FOR INSPECT-RICAL ELECTRICAL CONTRACT COMPANY

Syed Md Maruf Hasan

Student, BSc (Hons) in Software Engineering, Asia Pacific University, Malaysia

ABSTRACT

This paper described about the implementation of real time system for Inspect-Rical company by using object oriented programming language in java also UML designs supported to implement the system. Reason for developing the system which provides more data protection due to the concepts of object oriented programming. The UML designs provided here is usecase, activity diagram and physical diagram with object oriented implementation. Multithreading also explained here, also for future enhancement payment method, membership discount and biometric functions will be implemented to make more convenient for the clients.

Keywords: Activity diagram, Class diagram, Multithreading, Object Oriented Programming, Physical diagram

I. INTRODUCTION

According to our given case study, INSPECT-RICALS is a small electrical contracting company which deals with electrical wiring and installation domestically. The main stakeholders of this company is condominium developers and owners. This provides an installation and inspection services for them (developers and owners) to ensure the safety of all electrical wiring within the apartment. They basically do the service for older and newer development both. As the firm need to maintain the inspection process and produce the necessary working reports, they required a system which will help them to facilitate and manage the following. The users those are going to have a direct impact on the system are administrators, engineers, clerks and managers. As a result, depending on individual job roles we developed the system as followings:

System Functions:

Users:

a) Clerks:

- Able to register new client.
- Can take new order (inspection and installation).
- Can check order records.
- Can check and create reports summary.

- Can update clients order.
 - Can update client’s credentials.
- b) Managers:**
- Can check the works percentage done by each engineers.
- c) Engineers:**
- Can check their respective works allocated by clerks for individual clients.
 - Can check the client report and produce invoice for individuals.
- d) Administrators:**
- Can access every functionalities like managers, engineers and clerks.
 - Can register new managers, clerks and engineers using the system.

Here we will discuss how the system works following the UML (Uniform Modelling Language) diagrams, physical design and the concept of OO (Object Oriented) development.

II. UML DESIGN

UML is known as uniform modelling language which is highly used for visualizing the system in the field of software engineering. It is an important aspect for any object oriented software development. Using this we can illustrate how the system interact with users as well as how it flows data into the system. Here we will illustrate the overall functionality and the interaction of the system with users using different UML design.

2.1 Use Case

As we are having four types of users those are directly involve to the system, so we will show “Fig.1”, “Fig.2”, “Fig.3”, “Fig.4” the use case according to individual users interactions separately.

a. Clerk

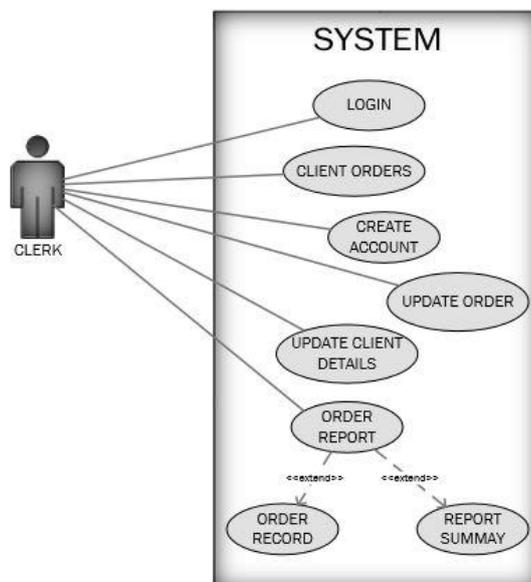


Figure.1 Clerk usecase

b. Engineer

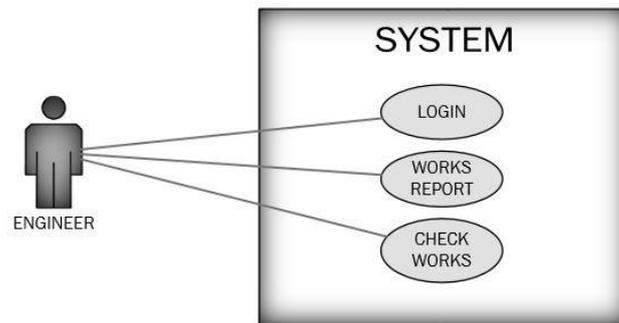


Figure.2 Engineer usecase

c. Manager

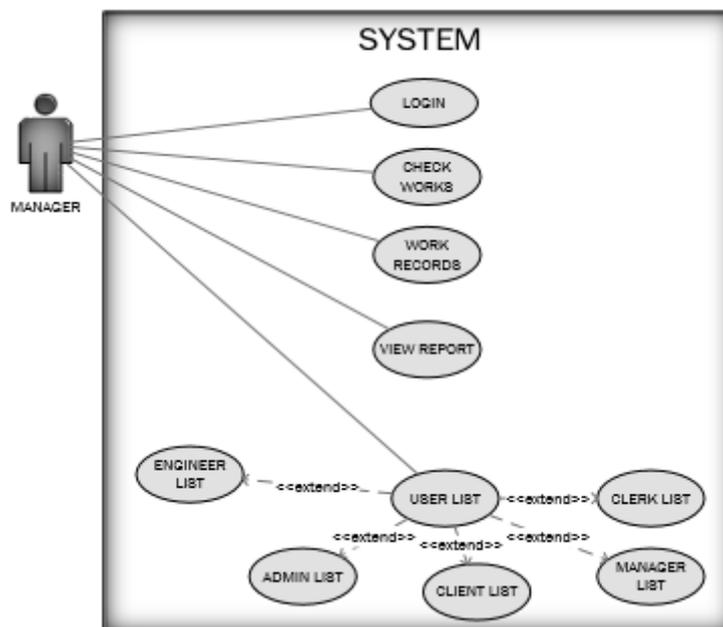


Figure.3 Manager usecase

c. Administrator

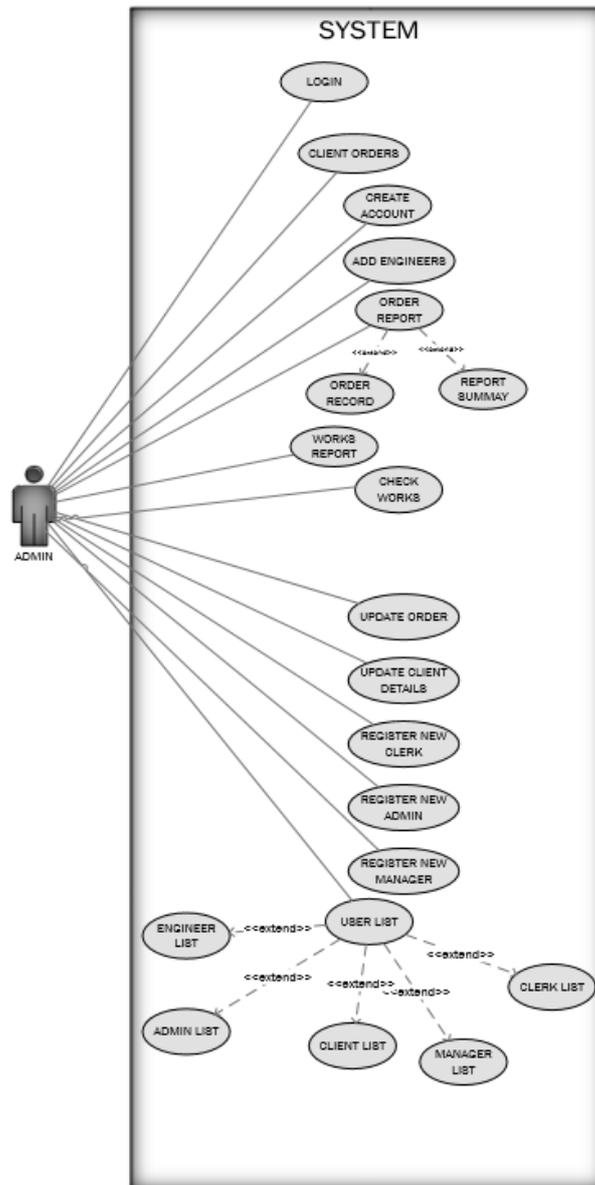


Figure.4 Administrator usecase

2.2 Activity

As we are having four types of users those are directly involve to the system, so we will show the overall activity according to individual users interactions separately. We will describe how the data flows according to users input shown in “Fig.5” and in “Fig.6” shown class diagram.

a. Clerk

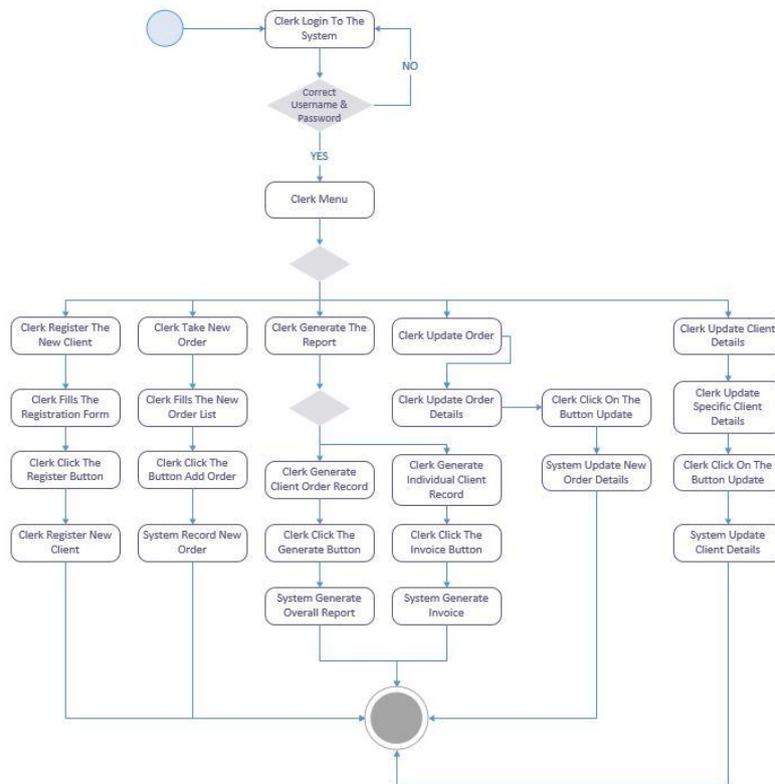


Figure.5 Clerk activity

2.3 Class

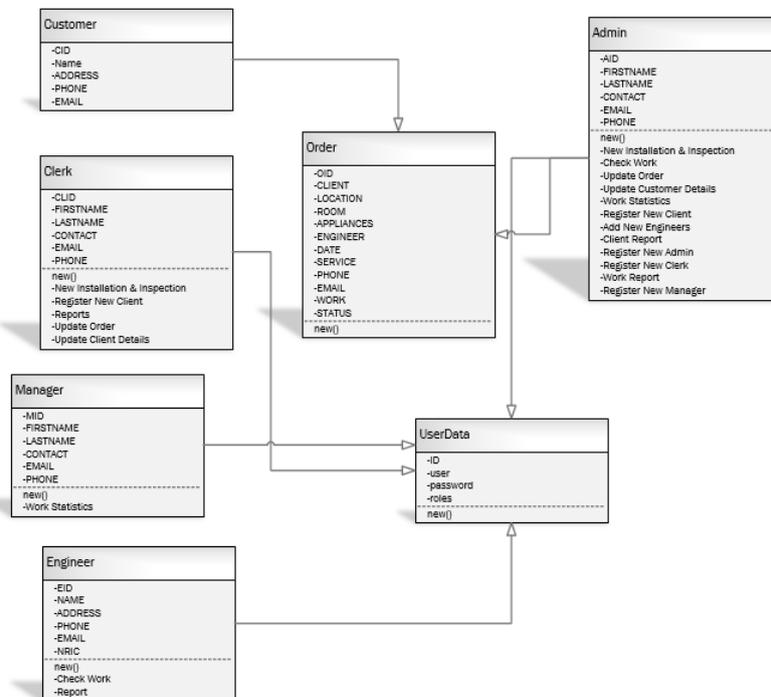


Figure.6 Class diagram

III. PHYSICAL DESIGN

Here we will describe the overall system interfaces and how it works depending on user input. We will follow the below steps of users while describing.

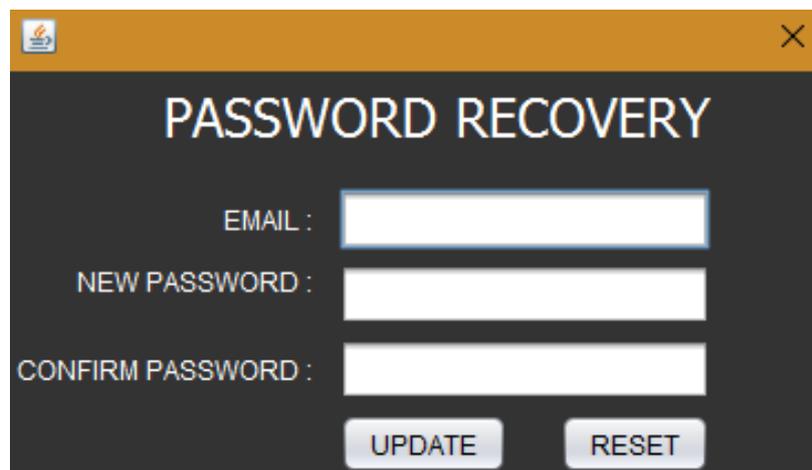
- a) Clerk
- b) Engineer
- c) Manager
- d) & Admin

3.1 Login



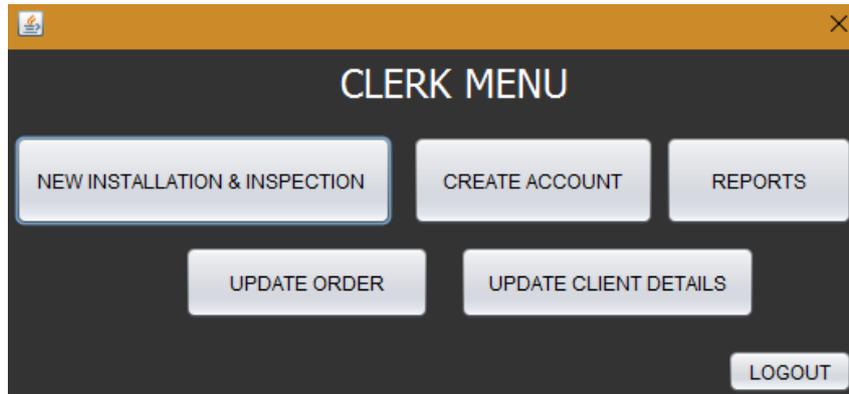
The screenshot shows a web browser window with a dark background and a gold header bar. The header bar contains a small logo on the left and window control buttons (minimize, maximize, close) on the right. The main content area features the large white letters "IR" at the top center. Below this, the text "WELCOME TO INSPECT RICALS SYSTEM" is displayed in white. There are two input fields: "EMAIL :" followed by a white text box, and "PASSWORD :" followed by a white text box. Below the password field is a blue button labeled "LOGIN". At the bottom left, there is a link "FORGOT PASSWORD ?" and at the bottom right, there is a blue button labeled "EXIT".

3.2 Forgot Password

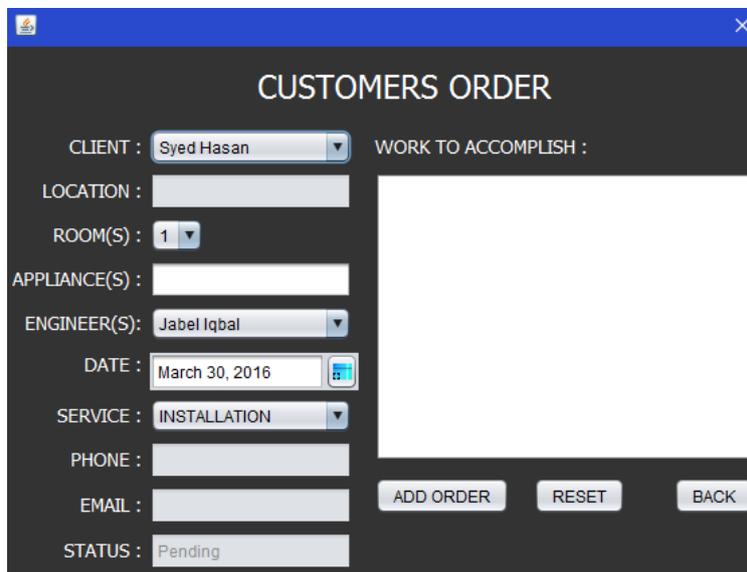


The screenshot shows a web browser window with a dark background and a gold header bar. The header bar contains a small logo on the left and a close button (X) on the right. The main content area features the text "PASSWORD RECOVERY" at the top center in white. Below this, there are three input fields: "EMAIL :" followed by a white text box, "NEW PASSWORD :" followed by a white text box, and "CONFIRM PASSWORD :" followed by a white text box. At the bottom, there are two blue buttons: "UPDATE" on the left and "RESET" on the right.

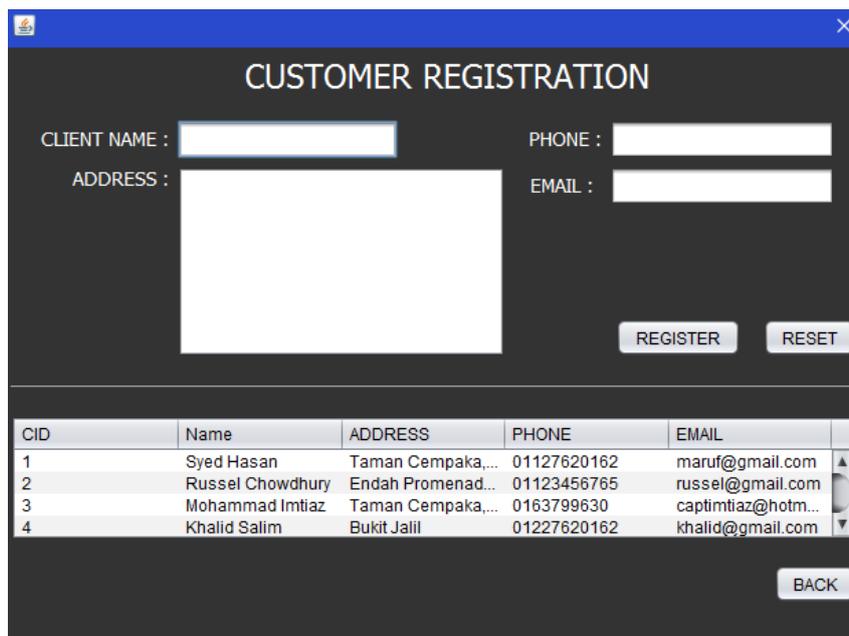
3.3 User Menu (Clerk)



3.4 New Installation & Inspection



3.5 Customer Registration



3.6 Clients Order Report

OID	CLIENT	LOCATION	ROOM	APPLIANC...	ENGINEER	DATE	SERVICE	PHONE	EMAIL	WORK	STATUS
1	Mohamma...	Taman Ce...	1	Electric Cir...	Jabel Iqbal	March 15, 2...	INSPECTI...	0163799630	captimiaz...	Inspect Th...	Completed
2	Russel Ch...	Endah Pro...	2	Electric Cir...	Jabel Iqbal	March 12, 2...	INSTALLAT...	011234567...	russel@g...	Kindly Insta...	Completed
3	Mohamma...	Taman Ce...	4	Water Heat...	Jabel Iqbal	March 25, 2...	INSTALLAT...	0163799630	captimiaz...	Intall The N...	Pending
4	Russel Ch...	Endah Pro...	1	Bulb	Jabel Iqbal	March 30, 2...	INSTALLAT...	011234567...	russel@g...	Install The ...	Pending

Description (Clients Order Report)

Above showing the overall order records of the clients. Clerk can print the details of this records by clicking the button “Generate”. The system will then generate the overall order records of the clients.

3.7 Clients Individual Report

CLIENT: ID:

OID	CLIENT	LOCATION	ROOM	APPLIANC...	ENGINEER	DATE	SERVICE	PHONE	EMAIL	WORK	STATUS
7	Khalid Salim	Bukit Jalil	4	TV Cable	Mani	March 19, 2...	INSPECTI...	012276201...	khalid@g...	Please Ch...	Completed
9	Khalid Salim	Bukit Jalil	1	Satellite C...	Mani	3/26/16	INSTALLAT...	012276201...	khalid@g...	Please Inst...	Pending

3.8 Update Order

CLIENT: STATUS:

OID	CLIENT	LOCATION	ROOM	APPLIAN...	ENGINEER	DATE	SERVICE	PHONE	EMAIL	WORK	STATUS
4	Russel C...	Endah Pr...	1	Bulb	Jabel Iqbal	March 30, ...	INSTALLA...	01123456...	russel@g...	Install Th...	Pending

UPDATE PANEL

OID: DATE:

APPLIANCES: WORK:

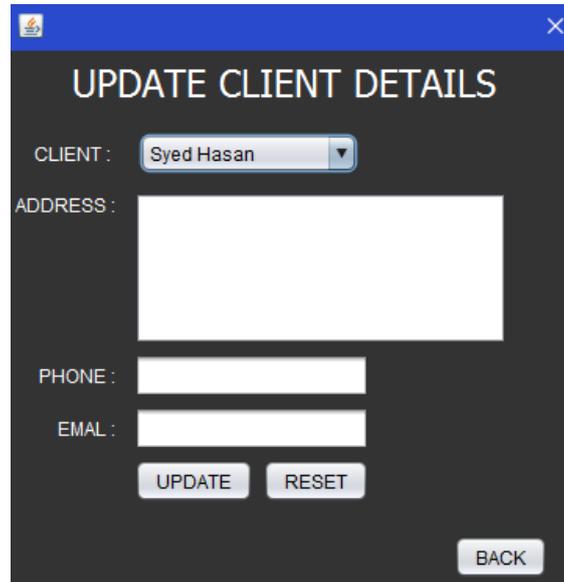
SERVICE:

STATUS:

Description (Update Order)

Above shows the update order panel for clients. Clerks can update the order details on clients demand. After entering the necessary details clerks need to click the button “Update” to update the details of the order. The system will then update the details and save it to database.

3.9 Update Clients Details



UPDATE CLIENT DETAILS

CLIENT : Syed Hasan

ADDRESS :

PHONE :

EMAL :

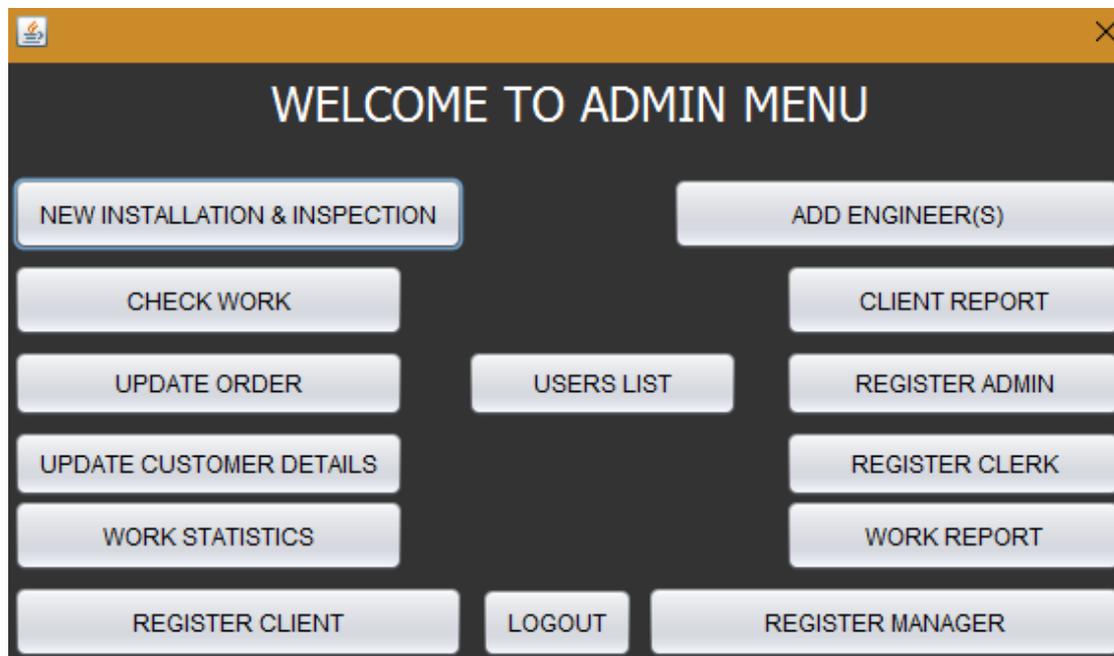
UPDATE RESET

BACK

Description (Update Clients Details)

Above shows the clients details update panel. Clerk can update the details of specific clients from this panel. After entering necessary details they need to click the button “Update” and the system will update the client’s details successfully.

3.10 User Menu (Admin)



WELCOME TO ADMIN MENU

NEW INSTALLATION & INSPECTION

ADD ENGINEER(S)

CHECK WORK

CLIENT REPORT

UPDATE ORDER

USERS LIST

REGISTER ADMIN

UPDATE CUSTOMER DETAILS

REGISTER CLERK

WORK STATISTICS

WORK REPORT

REGISTER CLIENT

LOGOUT

REGISTER MANAGER

IV. OBJECT ORIENTED IMPLEMENTATION

We developed the system using NetBeans IDE 8.1 drag and drop graphical user interface so the description of our object oriented implementation will slightly be different then hard coded system. Here we will describe how we implemented the OO (Object Oriented) concept in details [1].

4.1 Inheritance

Inheritance can be defined as the process where one class acquires the properties of another class. The class which inherits the properties of another class is also known as subclass. Usually to show the inheritance we use the word “Extend” [2].

Implementation (Inheritance)

```
public class INSTALINSPEC extends javax.swing.JDialog {  
  
    /**  
     * Creates new form INSTALINSPEC  
     */  
    public INSTALINSPEC(java.awt.Frame parent, boolean modal) {  
        super(parent, modal);  
        initComponents();  
  
        fillComboBox();  
    }  
}
```

4.2 Report

Here we will provide the detailed report of our research done for object oriented topics. We will follow some steps to proceed until the conclusion of this report.

Details

Object oriented is a type of computer programming language where programmers used to define the data types of a data structure and types of operation which we known as function. This function can be applied to the data structure as well. The basic concept of object oriented include the followings:

- a) Object
- b) Class
- c) Inheritance
- d) Interface
- e) & Package

4.3 Interface

Interface is known as the core part of object oriented programming languages. In java we used to call this as BULEPRINT of class. It includes static and abstract method only. For multiple inheritance we need to use interface. By implementing interface classes are allowed to be more formal about its behavior. Generally interface doesn't have any constructor. As java doesn't support multiple inheritance we need to use interface. Although it doesn't extend class but it extend to another interface. If a class want to implement the interface, it must provide all methods or else it will be consider as abstract class. Here we will show some examples of interface implementation of java object-oriented programming language [3]:

```
interface printable{
void print();
}

class A6 implements printable{
public void print(){System.out.println("Hello");}

public static void main(String args[]){
A6 obj = new A6();
obj.print();
}
}
```

4.4 Multithreading

One of the extra feature we should know about java object-oriented programming language is “Multithreading”. It is known as process which can execute multiple threading simultaneously. It is basically used to achieve multitasking in java program. Below are the advantages of using multithreading [4]:

- As threads are independent, it doesn't clock the users. As a result users are able to perform multiple operation in friendly environment.
- It saves time as users perform multiple operations.
- Being independent it doesn't affect other threads. Although it depends on single exceptions.

4.5 Polymorphism

It is refers to a programming languages which has the ability to process the objects differently depending on individual data types or classes. To be more specific it redefined the methods for derived classes. It allows developers to program in general rather than more specific. We also knows it as one method with multiple implementations for a certain class of action. Depending on the data types of the object implementation done at runtime. There are two types of polymorphism [5]. They are:

- Static
- & Dynamic

Advantages

Object-Oriented programming have some advantages. The advantages are given below:

- It provides developers a clear modular structure while doing programming.
- For defining the abstract data types it is good.
- Easy maintenance and can modify the existing code.
- It provides some important properties like inheritance, object, instance, methods etc.
- We can implement the real time scenarios.
- Encapsulation, Polymorphism, Abstraction are also a kind of fundamentals in object-oriented programming.
- It improve the software development quality
- As the code can be reused, so it lower the cost of development.

- Developers can focus on more to analysis and design work rather than coding.

Disadvantages

Although object oriented has more advantages, it also has some drawbacks too. Some of the drawbacks are given below:

- It is usually slower compare to procedural as it require many instructions to follow.
- It requires a lot of work to create. For a single line of code a great deal of planning take place.
- It is much larger than any other programming concept.

V. EXTRA FEATURES

In this system we have added some extra features to make more user friendly for the end users. We added the registration for manager, admin, clerk and engineer on the system admin panel. It will help admin to add new users who are directly involved in Inspect-Racal's System.

Besides, we also added a JTable on client registration form which shows the details of registered clients. After the new client registration the clerk can see the new client on real time from the JTable. Admin can also see the overall list of clients, managers, engineers and admins.

Along with that from manager panel, manager can see the users (manager, admin, engineer and clerk) and clients list and can keep the track of new registered users.

We also added the panel for forgot password. If users forget their password for login they can update the password by providing their username which is their email address.

VI. LIMITATIONS

Although we have added some of the extra functions in the system, this system has some limitations too. One of the limitations is client's payment. We didn't add any functions by which clerk can access the payment for clients while taking order. When selecting the date for engineers to allocate the works, the system will not show any free slots for engineers and engineers need to update the date of service if necessary.

Besides, the details of the engineers, managers, clerks and admin won't be able to update by the system rather than email and password as we are having forgot password function. The system will not show the percentage of works as required. It will only show how many completed works is under each engineers. Along with that engineers cannot provide the report from the system. They need to do it manually by writing in text file.

VII. FUTURE ENHANCEMENT

In future we will update the system for make it more user friendly. We will add the payment for clients while ordering new service. We will add membership discount for the registered members. Currently we are hosting database from local server. We will introduce real time so that other branches can also use the same data which will make more convenient to take order from more geographically dispersed clients.

In case of security we will add biometric function where the users need to use their finger prints to access the system.

VIII. CONCLUSION

In conclusion it can be said that this system is much user friendly for INSPECT-RICALS. Users can easily interact with the system without any hesitations. The UI (User Interface) is very much common for the users so that it will help them to operate successfully. Compare to previous time the data will be saved into the database safely and users can easily retrieve data directly from the database.

IX. ACKNOWLEDGMENT

The author would like to share gratitude to Mr Umapathy Eaganathan, Lecturer in Computing, Asia Pacific University, Malaysia also Miss Angel Aron for the constant support and motivation helped me to participate in this International Conference and also for journal publication.

REFERENCES

- [1] BeginnersBook, 2016. BeginnersBook. [Online] Available at: <http://beginnersbook.com/2013/03/inheritance-in-java/> [Accessed 30 March 2016].
- [2] Chaitanya, 2014. BeginnersBook. [Online] Available at: <http://beginnersbook.com/2013/03/oops-in-java-encapsulation-inheritance-polymorphism-abstraction/>
- [3] JavaTPoint, 2014. JavaTPoint. [Online] Available at: <http://www.javatpoint.com/interface-in-java> [Accessed 30 March 2016].
- [4] JavaTPoint, 2014. JavaTPoint. [Online] Available at: <http://www.javatpoint.com/multithreading-in-java>
- [5] Chaitanya, 2015. BeginnersBook. [Online] Available at: <http://beginnersbook.com/2013/03/polymorphism-in-java/> [Accessed 31 March 2016].