

CSYE6200 Project

Motor Service Management

Group 3 Members:

Lohita Gundapaneni

Pragnya Kondrakunta

Vamsi Mokari

Surya Bandaru

Vikyath Kumar Reddy

Rohith Chevvakula

Problem Statement

- Add and Manage Employees
 - Employee name, age, salary, credentials
 - Separate dashboard to login and access application
- Add and Manage incoming Clients
 - Client name, age, credentials
 - Report the required motor service jobs to the client
 - Separate dashboard to login and access application
- Add Mechanics
 - Mechanic name, age, credentials
 - Separate dashboard to login and access application
- Book appointment with mechanics to motor service jobs
- Maintain job service status
- Record the completed jobs

Build Instructions

Setting up the project, the dependencies and running it.

1. Open Apache NetBeans IDE 12.6
2. Go to File>Import Project>From ZIP. Select the project .zip and click Open.
3. If any Project Problems pops up resolve it using the following .jar files (Found in Jars folder of the project): **"mongo-java-driver-3.12.8.jar"**
4. Once the project is open, click on **Run Project** to start the application.
5. Start the mongo db server using:

mongod

Ensure that it is running on the assigned port 27017.

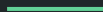
Packages & Libraries

Packages

- java.util
- javax.swing

Libraries

- "mongo-java-driver-3.12.8.jar"



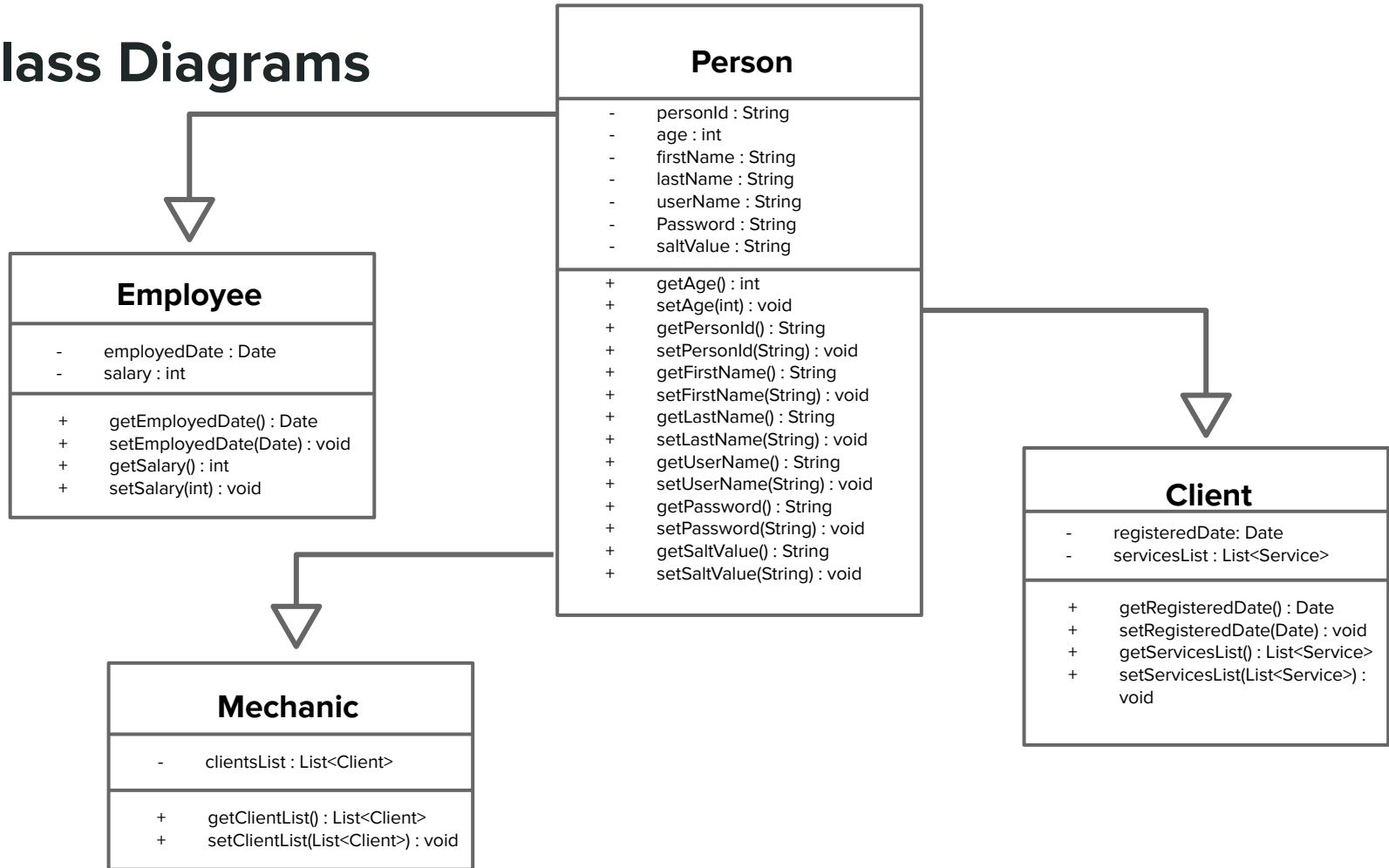
Design Features

- Collections
 - Exception Handling
 - Singleton Design Pattern
 - Streams API
 - Java Swing
 - OOP Concepts
 - Inheritance
 - Polymorphism
 - Abstraction (private attributes and public methods)
-

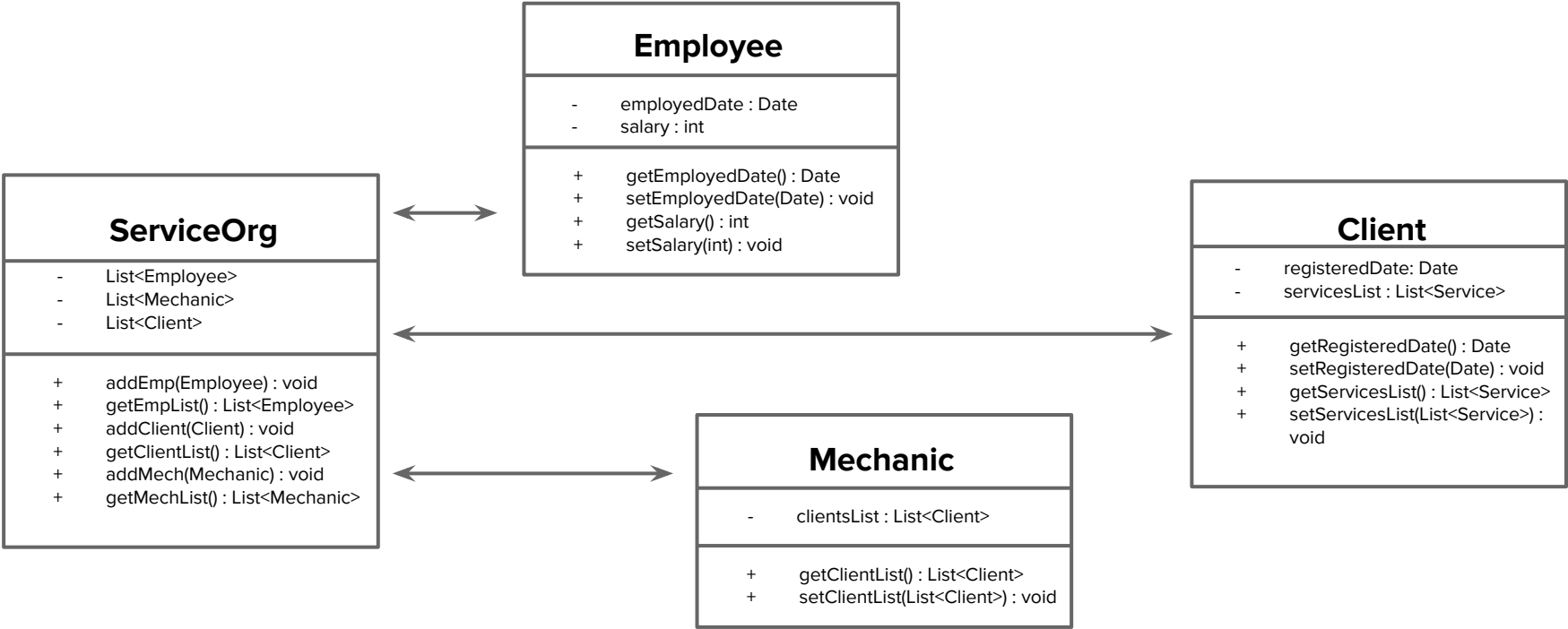
Class Hierarchy

- java.lang.**Object**
 - Business.Client.**ClientDirectory**
 - java.awt.**Component** (implements java.awt.image.ImageObserver , java.awt.MenuContainer , java.io.Serializable)
 - java.awt.**Container**
 - javax.swing.**JComponent** (implements java.io.Serializable)
 - javax.swing.**JPanel** (implements javax.accessibility.Accessible)
 - userInterface.Employee.**AddClientJPanel**
 - userInterface.Admin.**AddEmployee**
 - userInterface.Admin.**AddMechanic**
 - userInterface.Admin.**AdminWorkSpace**
 - userInterface.Employee.**ClientServiceInfo**
 - userInterface.Employee.**EmployeeWorkSpace**
 - userInterface.Employee.**ManageClients**
 - userInterface.Client.**ManageClientWorkSpace**
 - userInterface.Admin.**ManageEmployee**
 - userInterface.Mechanic.**MechanicWorkSpace**
 - userInterface.Admin.**UpdateEmployee**
 - java.awt.**Window** (implements javax.accessibility.Accessible)
 - java.awt.**Frame** (implements java.awt.MenuContainer)
 - javax.swing.**JFrame** (implements javax.accessibility.Accessible , javax.swing.RootPaneContainer , javax.swing.WindowConstants)
 - userInterface.**MainJFrame**
 - Business.ServiceCenter.**ConfigureSystem**
 - Business.Employee.**EmployeeDirectory**
 - Business.Organization.**Organization**
 - Business.ServiceCenter.**ServiceOrg**
 - Encrypt.Passcode.**PasscodeUtils**
 - Business.Person.**Person**
 - Business.Client.**Client**
 - Business.Employee.**Employee**
 - Business.Mechanic.**Mechanic**
 - Business.Service.**Service**
 - Business.Factory.**UsersFactory**

Class Diagrams



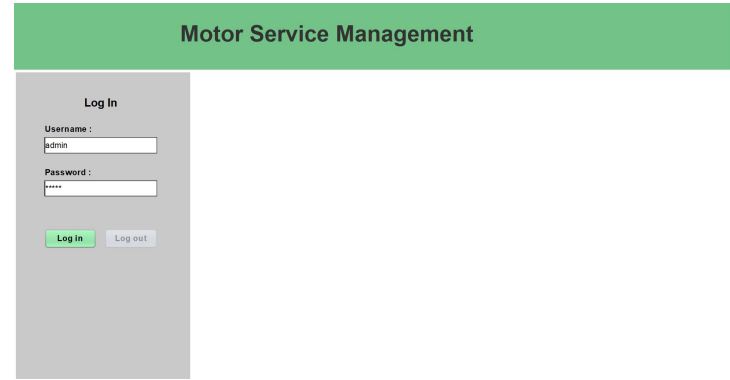
Class Diagrams



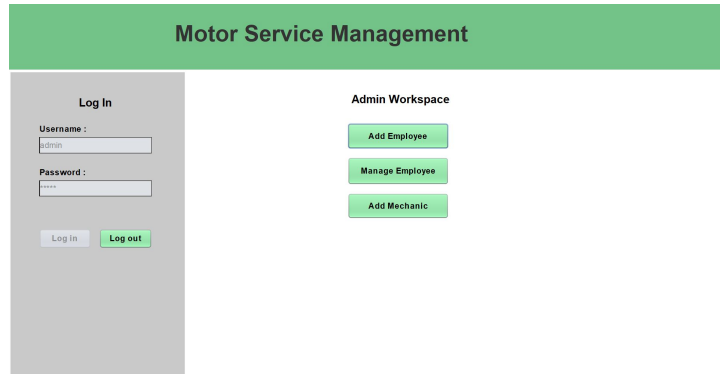
Operation Instructions

Operation Instructions

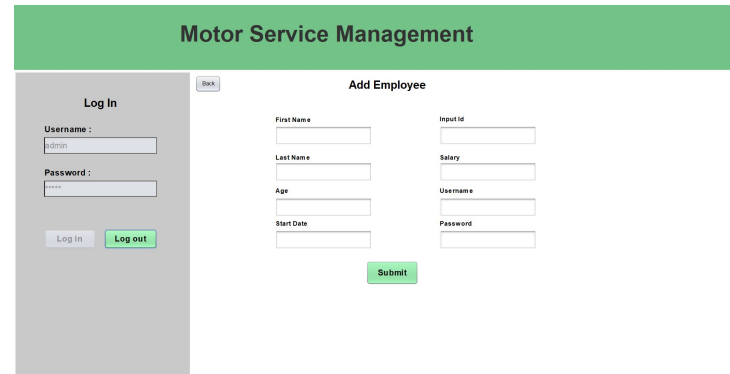
1. Log in as an admin:
 - a. Username: admin
 - b. Password: admin
2. Add an employee:
 - a. Fill in the details and click on Submit to Add an Employee.



The screenshot shows the 'Motor Service Management' application with a green header. On the left, there is a 'Log In' panel with a grey background. It contains a 'Username' field with 'admin' entered, a 'Password' field with 'admin' entered, and two buttons: 'Log In' (green) and 'Log out' (grey). The rest of the page is empty.



The screenshot shows the 'Motor Service Management' application with a green header. On the left, there is a 'Log In' panel with a grey background, identical to the previous screen. To the right of the 'Log In' panel is the 'Admin Workspace' section, which contains three green buttons: 'Add Employee', 'Manage Employee', and 'Add Mechanic'.



The screenshot shows the 'Motor Service Management' application with a green header. On the left, there is a 'Log In' panel with a grey background, identical to the previous screens. To the right of the 'Log In' panel is the 'Add Employee' section. It features a 'Back' button (grey) and a form with the following fields: 'First Name', 'Last Name', 'Age', 'Start Date', 'Input Id', 'Salary', 'Username', and 'Password'. A green 'Submit' button is located at the bottom right of the form.

Operation Instructions

3. Manage Employees:

- Employee details can be updated and deleted by clicking on the Manage Employees button.

The screenshot displays the 'Motor Service Management' application. On the left is a 'Log In' sidebar with fields for 'Username' (containing 'admin') and 'Password' (containing '*****'), and 'Log In' and 'Log out' buttons. The main area is titled 'Manage Employee' and includes a 'Back' button. Below the title is a table with columns: Employee ID, Name, Age, Emp. Date, Username, and Object. The table contains one row with the following data: Employee ID: 123, Name: pignya, Age: 23, Emp. Date: Sat Dec 11 00:00:00, Username: pignya, and Object: ["*", "123", "pignya"]. Below the table are 'Delete Employee' and 'Update Employee' buttons.

Employee ID	Name	Age	Emp. Date	Username	Object
123	pignya	23	Sat Dec 11 00:00:00	pignya	["*", "123", "pignya"]

4. Add a mechanic:

- Fill in the details and click on **Submit** to add the mechanic.

The screenshot displays the 'Motor Service Management' application. On the left is a 'Log In' sidebar with fields for 'Username' (containing 'admin') and 'Password' (containing '*****'), and 'Log In' and 'Log out' buttons. The main area is titled 'Add Mechanic' and includes a 'Back' button. Below the title are input fields for 'First Name', 'Last Name', 'Age', 'Years of Experience', 'Username', and 'Password'. A 'Submit' button is located at the bottom right.

Operation Instructions

5. Log in as employee:

- Use your employee credentials to log in as an employee

The screenshot shows the 'Motor Service Management' application interface. On the left, there is a 'Log In' panel with fields for 'Username' (containing 'pragnya') and 'Password' (containing 'pragnya'), and 'Log In' and 'Log out' buttons. On the right, the 'Employee Workspace' is visible, showing 'Login User: pragnya' and two buttons: 'Add Client' and 'Manage Client'.

6. Add a client:

- Fill in the details of the client.
- Identify and select the requisite service jobs (like Engine Failure, Flat Tyre etc.) for the client.

The screenshot shows the 'Motor Service Management' application interface for adding a new client. On the left, the 'Log In' panel is visible. The main area is titled 'Add client' and includes a 'Back' button. It contains two columns of form fields: 'First Name', 'Last Name', 'Age', 'Registration Date', 'Username', and 'Password' on the left; and 'Flat Tyre', 'Excessive Emissions', 'Mileage Mischief', 'Electric Problems', 'Worn Break Pads', and 'Engine Failure' on the right. Each field on the right has a 'Required' label and a dropdown arrow. A 'Submit' button is located at the bottom right.

Operation Instructions

5. Manage clients:

- In the **Manage Client**, you can see all the clients. By clicking on a particular client and clicking on Service details, you can view the status of each service.

Motor Service Management

Log In

Back

Username :

pragnya

Password :

Log In

Log out

Manage Clients

Client ID	Name	Age	Registration Date	Username
C001	Abhiram Kumar	29		cust1
C002	John Doe	45		cust2

ID: C002

Service Details

Motor Service Management

Log In

Back

Username :

pragnya

Password :

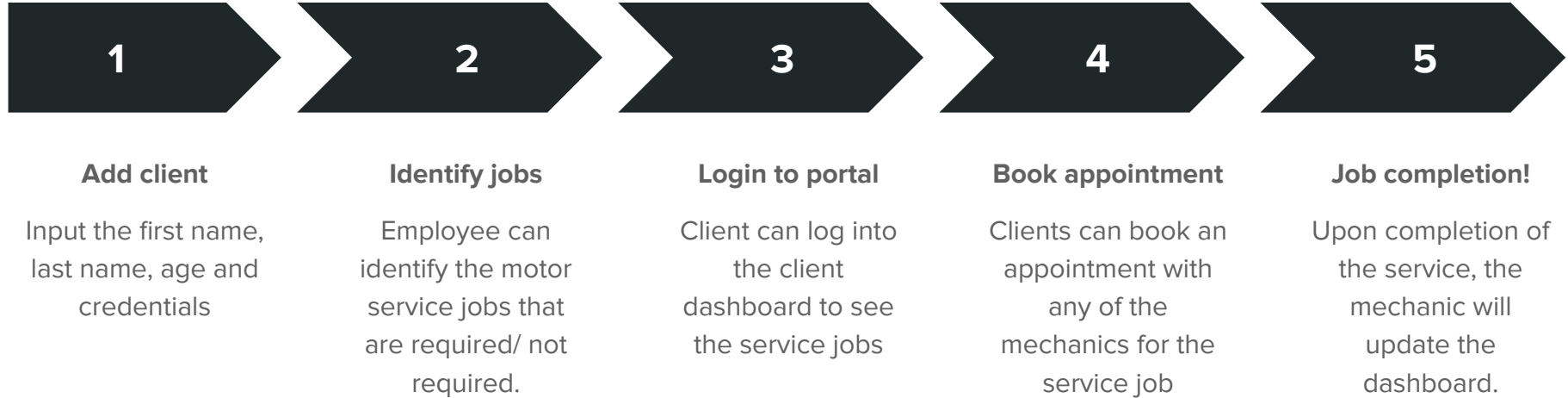
Log In

Log out

client Service Details

Not Required	Required	Awaiting	Completed
Flat Tyre Excessive Emissions	Manage Mischief Electric Problems Worn Break pads Engine Failure		

Journey of Client



Operation Instructions

5. Log in as a client:

- Use credentials to log in as an client and the required service jobs can be seen here.
- You can select any available mechanic and book an appointment with them
- Once you book the appointment the status of the service status changes to **awaiting**.

6. Log in as a mechanic

- Use credentials to log in as a mechanic and all bookings can be seen here.
- Once the service is completed, you can click on the button and the service status changes to **completed**.

Motor Service Management

Log In

Username :

Password :

Hello cust1!

client ID	Service	Status
C001	Mileage Mischief	Required
C001	Electric Problems	Required
C001	Worn Break pads	Required

Available Mechanic:

Motor Service Management

Log In

Username :

Password :

Hello mech1!

Your Clients

Client Id	Client name	Service Name
C001	cust1	Worn Break pads

Operation Instructions

All this can be monitored on the employee's dashboard.

Motor Service Management

Log In

Username :

pragnya

Password :

Log in

Log out

Back

Client Service Details

Not Required

Flat Tyre
Excessive Emissions
Engine Failure

Required

Electric Problems

Awaiting

Worn Break pads

Completed

Mileage Mischief

Future Scope

- Can be extended to other service level organisations (like pest control, car wash, salons).
- Analytics of the services & its status with graphical representations.
- Email/Phone integration for notifications to the client

Other libraries

- "jcommon-1.0.23.jar"
- "bson-4.2.3-javadoc.jar"
- "jfreechart-1.0.19.jar"
- "jfreechart-1.0.19-swt.jar"

Application Demo

Team

Lohita & Vamsi

- Backend & logic implementation

Pragnya & Surya

- UI design & Swing implementation

Vikyath & Rohith

- Logic Design & implementation

Everyone worked on the overall class structure and design of the project.

Thank you. Any Questions?
