

PROJECT REPORT
ON
Automatic Resume Builder JAVA

Submitted to
University of Mumbai

In partial fulfilment of requirement of
Master of Computer Applications

Submitted by
Boyar Raghupathy
Roll No - 10
&
Rahul Danveer Gupta
Roll No - 27

Under the Guidance of
Dr. Vinita Gaikwad, Director, TIMSCDR
and
Mr. Pankaj Mudholkar, Asst. Professor, TIMSCDR

Thakur Institute of Management Studies, Career Development & Research
Mumbai

2020-2021

ACKNOWLEDGEMENT

In writing this project report, it has been our endeavor to present the subject in simple and lucid manner to those who wish to refer it for enhancement or improving the existing features. The knowledge that we have gathered, study of relevant literature for understanding our project, planning of details and suitable software platform by itself and acknowledgement to the industry zeal and technical competence of those many individuals who have contributed to it with profound gratitude, we wish to acknowledge them.

We sincerely thank Thakur Institute of Management Studies, Career Development & Research for training us in the MCA course. We express our deep gratitude to our honorable director Dr. Vinita Gaikwad and we would also express our deep gratitude to our internal guides Mr. Pankaj Mudholkar. Their expertise and insight were invaluable to us during our internship.

CERTIFICATE

This is to certify that the project entitled Automatic Resume Builder has been carried out by Boyar Raghupathy Rajendiran (10) and Rahul Danveer Gupta (27) under my guidance during the academic year (Aug –Dec) 2020.

Internal Guide

External Guide

CONTENTS

1. INTRODUCTION

- 1.1 Problem definition
- 1.2 Objective of Project
- 1.3 Scope of Project

2. ANALYSIS

- 2.1 Introduction
- 2.2 Software Requirement Specification
 - 2.2.1 User requirement
 - 2.2.2 Software requirement
 - 2.2.3 Hardware requirement

3. DESIGN

- 3.1 UML diagram (All UML diagrams from use case to deployment diagram)

4. IMPLEMENTATION & RESULTS

- 4.1 Implementation Code
- 4.2 Input/ Output Screens

5. TESTING & VALIDATION

- 5.1 Introduction
- 5.2 Design of test cases and scenarios
- 5.3 Validation

6. CONCLUSION:

- 6.1 Project Conclusion
- 6.2 Future enhancement

1. INTRODUCTION:

An online resume builder is a software developed to simplify the task of creating a resume for individuals. The application provides an effective means of designing desired resume in fact a professional looking resume. The system is flexible to be used and reduces the need of thinking and designing an appropriate resume according to qualifications. Usually individuals get confused while creating a resume especially for a novice person such as graduate students. They don't get a clear idea of what things and information must be included in a resume. Hence the system is developed to provide them an easy way for creating a professional looking resume.

This project is user-friendly and requires minimum human intervention. Individuals just have to fill up a form that specifies questions from all required fields such as personal questions, educational, qualities, interest, skills and so on. The answers provided by the users are stored and the system automatically generates a well-structured resume. Users have option to create resume in any format and file.

1.1 PROBLEM DEFINATION

If you take a deep breath and a closer look, some of these problems may be easy to sort out with a few simple word changes or formatting revisions. Here are a few common resume challenges that aren't as intimidating as they sometimes seem.

Problem #1: Too Much Information

How can you fit every one of your proudest accomplishments and previous positions into a one-page document?

The Solution

You can't. And you don't have to. Choose the most important and most relevant details of your career history, and employers will usually be able to fill in the blanks on their own. For example, if you won an international grant for your protein synthesis research, you don't have to list every entry-level position you held as you worked your way up in the protein synthesis field. Some accomplishments can be used to speak for the others.

Problem #2: Not Enough Information

How can you frame a long career history that includes only a short list of positions?

The Solution

If you've only held two positions during your 20-year career, and both of them were with the same company, don't despair over the fact that your resume covers only half a page. Instead, break these two positions down into components and list the most important accomplishments and responsibilities you held for each of them. Meanwhile, find a way to frame your loyalty and reliability as an asset.

Problem #3: Boring Job Titles

Explaining the challenges of a difficult position with a bland-sounding title.

The Solution

"Retail Expert," "Project Manager," "Production Specialist," and "Sales Associate" are all titles that could suggest low-responsibility jobs OR extremely challenging positions with high stress, high stakes, and years of complex training. We're measured by the obstacles we successfully overcome, but also by the height of these obstacles in the first place. If your experience, awards, and training have been more rigorous than they sound, use numbers and timelines to make this clear.

Problem #4: Burned Bridges

You've been fired, and you have no idea how to hide this fact from employers.

The Solution

Don't hide it! If you've been fired from a position, if you left an employer on bad terms, or if you left a job after a project failure or performance problem, there's no reason to leave this experience off your resume. List this position like any other—just don't use this employer as a reference. And if asked, be ready to frame the experience in a positive way during your interview.

Problem #5: Timeline Gaps

The work experience section of your resume has some pretty lengthy gaps in-between each position.

The Solution

Don't try to hide these. Everyone has them and most of the time, they reflect a complex life full of risks and real experiences (both of which are valued by intelligent, responsible employers.) If you were sidelined for a few years due to a layoff, a return to school, or a parenting gap, don't worry about this and don't fudge your employment dates in order to spin the truth. Just be ready to explain what you were doing during this chapter, what you learned, and how you stayed in touch with changes in your industry.

1.2 OBJECTIVE OF PROJECT

This Online resume builder is designed to help the job seekers to create a professional resume for them. The candidates are not required to spend more time designing and creating professional resume. They can enter their details directly into the pop-up box and their resume will be created automatically. A well-structured resume will be generated, once the user submits his/her details and user can download it in any file format, as per his/her requirement.

Facilities provided by this project are as follows: -

- Details of customers are recorded.
- Update of data is easy.
- Flow of information is fast and easy.
- Customers can login to their accounts and view & update their data
- Notifications about resume views & downloads.

EXISTING SYSTEM

If someone wants to make his/her resume, then he/she has to look for every single detail such as formatting, alignment, designs, patterns. So, it is a very complex process and thus, the productivity of the resume also decreases. The existing system is very complex and making resume via word files and excels is a tough task. Sometimes, while making a resume, a person ends up forgetting some very crucial information while going through all these designs and formats.

PROPOSED SYSTEM

In the proposed Online resume builder, a person has to only fill the questionnaire asked by the online platform. All the other alignments and formats of the resume are being directly taken care of by the system. Even, the final resume which came out is very professional as per the industry requirement and the candidate can download the resume as per the file format, he/she requires. This is avoiding need of putting manual effort for creating resume

1.3 SCOPE OF PROJECT

- Online Resume Builder can be used in accordance with the requirements of the users. Users can customize their resumes with their choice of themes & details. The services are hard to be defeated by the competitors as the system is providing the customers exactly what they want.
- Automatic resume builder application is going to be a resume builder software, it will be able to generate and download the resume.
- Develop a Java Application for automating the process of Resume Writing. This would be facilitating the employees to make and print their resumes in a proper format. In addition, it will be facilitating the higher management to search the employees depending upon their skills sets and other attributes.
- The Basic Requirement is to have a centralized repository of all the skill-holders in the organization so that an employee with a particular skill set can be immediately found in case of urgent requirement.
- The fact that the projects is save the time to design your resume.

2. ANALYSIS

2.1 INTRODUCTION

Analysis of resume builder platform taken into consideration in order to build new system with improved functions.

System analysis will be performed to determine if it is feasible to design an information based on policies and plans of the organization and on user requirements and to eliminate the weaknesses of the present system.

2.2 SOFTWARE REQUIREMENT SPECIFICATION

A Software requirements specification document describes the intended purpose, requirements and nature of a software to be developed. It also includes the yield and cost of the software.

In this document, SRS for Automatic Resume Builder using java is explained in detail. Below is the work breakdown structure for better understating of SRS.

2.2.1 USER REQUIREMENT

Every user should be:

- Comfortable in working with computer.
- He must have knowledge of resume writing.
- He must also have basic knowledge of English language.

2.2.2 SOFTWARE REQUIREMENT

- Windows XP
- JAVA
- Eclipse
- Documentation Tools – Microsoft Word

2.2.3 HARDWARE REQUIREMENT

- Pentium-IV (Processor).
- Hard disk 10 GB
- Microsoft Compatible 101 or more Key Board
- 256 MB Ram
- 512 KB Cache Memory
- Internet connection.

JAVA (JDK/ JRE):

JDK:

JDK is an acronym for Java Development Kit. The Java Development Kit (JDK) is a software development environment which is used to develop Java applications and applets. It physically exists. It contains JRE + development tools.

JDK is an implementation of any one of the below given Java Platforms released by Oracle Corporation:

- Standard Edition Java Platform
- Enterprise Edition Java Platform
- Micro Edition Java Platform

The JDK contains a private Java Virtual Machine (JVM) and a few other resources such as an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (Javadoc), etc. to complete the development of a Java Application.

JRE:

JRE is an acronym for Java Runtime Environment. It is also written as Java RTE. The Java Runtime Environment is a set of software tools which are used for developing Java applications. It is used to provide the runtime environment. It is the implementation of JVM. It physically exists. It contains a set of libraries + other files that JVM uses at runtime.

The implementation of JVM is also actively released by other companies besides Sun Microsystems

Java AWT (Abstract Window Toolkit):

AWT stands for Abstract Window Toolkit. It is a platform dependent API for creating Graphical User Interface (GUI) for java programs.

Why AWT is platform dependent?

Java AWT calls native platform (Operating systems) subroutine for creating components such as textbox, checkbox, button etc. For example, an AWT GUI having a button would have a different look and feel across platforms like windows, Mac OS & Unix, this is because these platforms have different look and feel for their native buttons and AWT directly calls their native subroutine that creates the button. In simple, an application build on AWT would look like a windows application when it runs on Windows, but the same application would look like a Mac application when runs on Mac OS.

Eclipse IDE:

Eclipse is an integrated development environment used in computer programming. It contains a base workspace and an extensible plug-in system for customizing the environment.

3. DESIGN

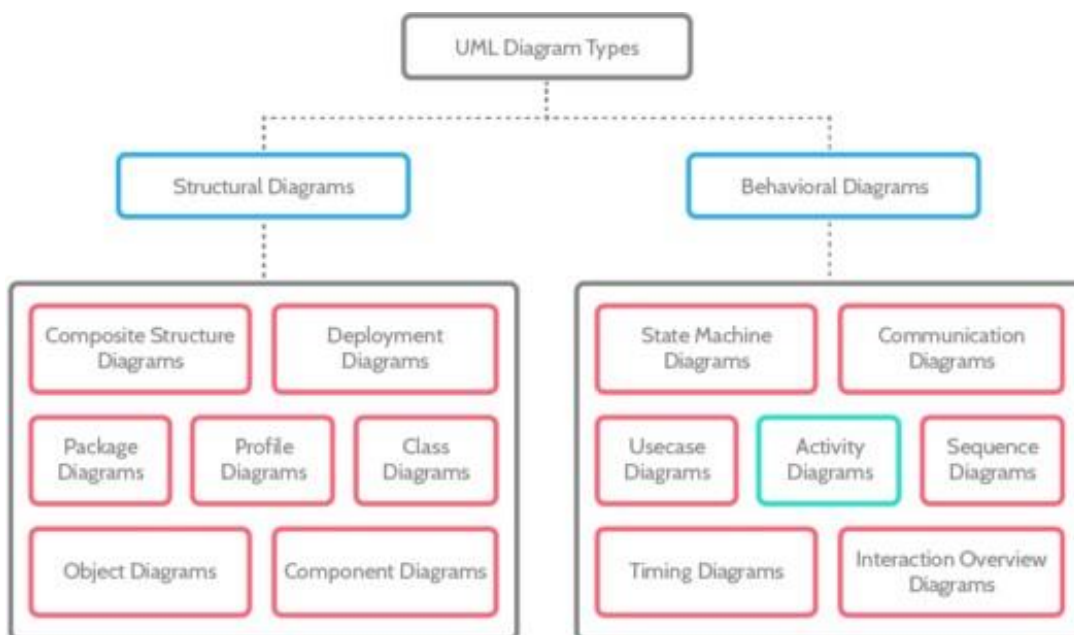
Better understanding of the working and functions of the applications UML diagrams representation is used for designing and explaining various aspects through different diagrams.

3.1 UML DIAGRAM

We prepare UML diagrams to understand the system in a better and simple way. A single diagram is not enough to cover all the aspects of the system. UML defines various kinds of diagrams to cover most of the aspects of a system.

There are two broad categories of diagrams and they are again divided into subcategories –

- **Structural Diagrams:** The structural diagrams represent the static aspect of the system. These static aspects represent those parts of a diagram, which forms the main structure and are therefore stable.
- **Behavioral Diagrams:** Basically, capture the dynamic aspect of a system. Dynamic aspect can be further described as the changing/moving parts of a system.



STRUCTURAL DIAGRAMS:

Deployment Diagram –

Deployment diagrams are used to visualize the topology of the physical components of a system, where the software components are deployed.

Deployment diagrams are used to describe the static deployment view of a system. Deployment diagrams consist of nodes and their relationships.

How to Draw a Deployment Diagram?

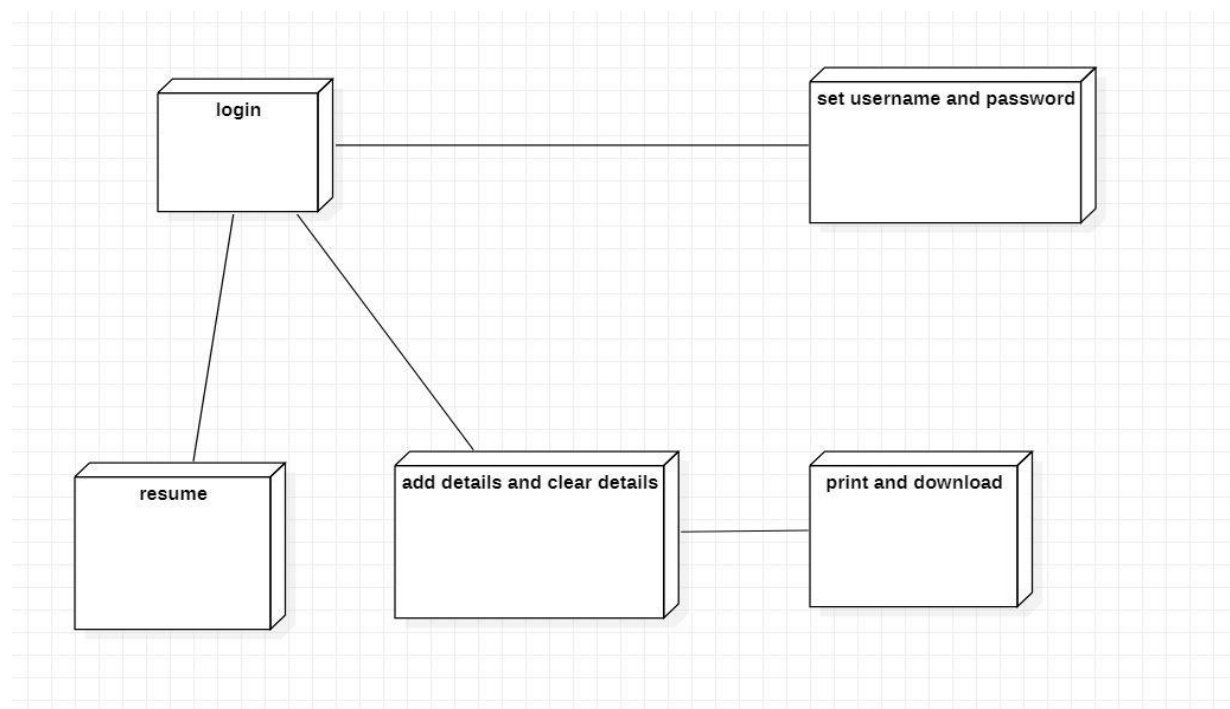
Deployment diagram represents the deployment view of a system. It is related to the component diagram because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware used to deploy the application.

Deployment diagrams are useful for system engineers. An efficient deployment diagram is very important as it controls the following parameters –

- Performance
- Scalability
- Maintainability
- Portability

Before drawing a deployment diagram, the following artifacts should be identified –

- Nodes
- Relationships among nodes



Package Diagram -

Package diagram, a kind of structural diagram, shows the arrangement and organization of model elements in middle to large scale project. Package diagram can show both structure and dependencies between sub-systems or modules, showing different views of a system, for example, as multi-layered (aka multi-tiered) application - multi-layered application model.

Purpose of Package Diagrams

Package diagrams are used to structure high level system elements. Packages are used for organizing large system which contains diagrams, documents and other key deliverables. Package Diagram can be used to simplify complex class diagrams, it can group classes into packages.

A package is a collection of logically related UML elements.

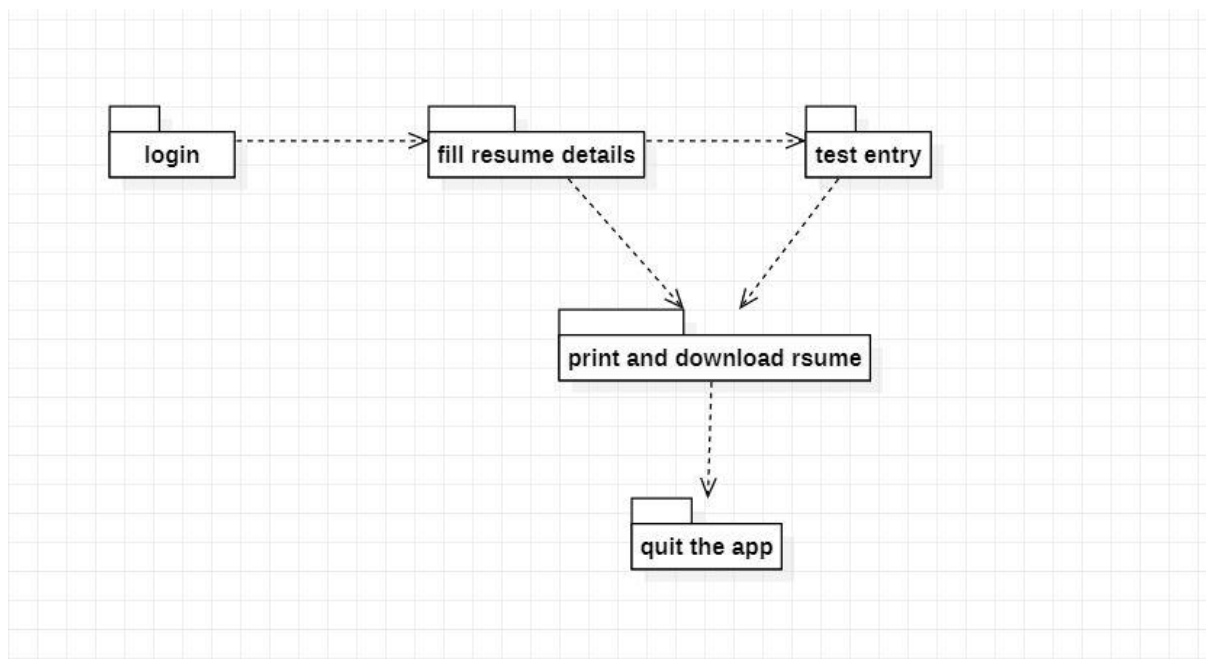
Packages are depicted as file folders and can be used on any of the UML diagrams.

Package Diagram at a Glance

Package diagram is used to simplify complex class diagrams, you can group classes into packages. A package is a collection of logically related UML elements.

The diagram below is a business model in which the classes are grouped into packages:

- Packages appear as rectangles with small tabs at the top.
- The package name is on the tab or inside the rectangle.
- The dotted arrows are dependencies.
- One package depends on another if changes in the other could possibly force changes in the first.



Class Diagram –

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application.

Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object-oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages.

Purpose of Class Diagrams

The purpose of class diagram is to model the static view of an application. Class diagrams are the only diagrams which can be directly mapped with object-oriented languages and thus widely used at the time of construction.

UML diagrams like activity diagram, sequence diagram can only give the sequence flow of the application, however class diagram is a bit different. It is the most popular UML diagram in the coder community.

The purpose of the class diagram can be summarized as –

- Analysis and design of the static view of an application.
- Describe responsibilities of a system.
- Base for component and deployment diagrams.
- Forward and reverse engineering.

How to Draw a Class Diagram?

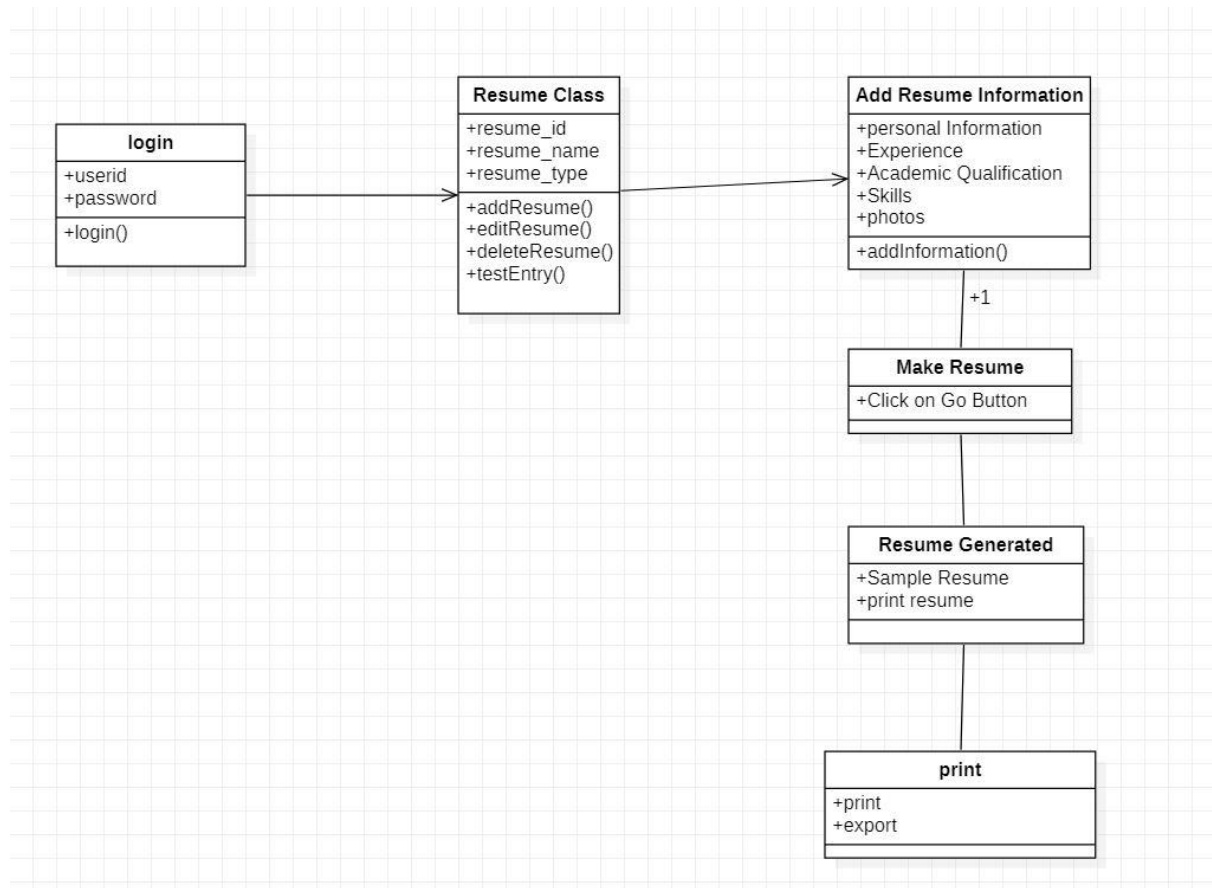
Class diagrams are the most popular UML diagrams used for construction of software applications. It is very important to learn the drawing procedure of class diagram.

Class diagrams have a lot of properties to consider while drawing but here the diagram will be considered from a top-level view.

Class diagram is basically a graphical representation of the static view of the system and represents different aspects of the application. A collection of class diagrams represents the whole system.

The following points should be remembered while drawing a class diagram –

- The name of the class diagram should be meaningful to describe the aspect of the system.
- Each element and their relationships should be identified in advance.
- Responsibility (attributes and methods) of each class should be clearly identified
- For each class, minimum number of properties should be specified, as unnecessary properties will make the diagram complicated.
- Use notes whenever required to describe some aspect of the diagram. At the end of the drawing it should be understandable to the developer/coder.
- Finally, before making the final version, the diagram should be drawn on plain paper and reworked as many times as possible to make it correct.



Object Diagram –

Object diagrams are derived from class diagrams so object diagrams are dependent upon class diagrams.

Object diagrams represent an instance of a class diagram. The basic concepts are similar for class diagrams and object diagrams. Object diagrams also represent the static view of a system but this static view is a snapshot of the system at a particular moment.

Object diagrams are used to render a set of objects and their relationships as an instance.

Purpose of Object Diagrams

The purpose of a diagram should be understood clearly to implement it practically. The purposes of object diagrams are similar to class diagrams.

The difference is that a class diagram represents an abstract model consisting of classes and their relationships. However, an object diagram represents an instance at a particular moment, which is concrete in nature.

The purpose of the object diagram can be summarized as –

- Forward and reverse engineering.
- Object relationships of a system
- Static view of an interaction.
- Understand object behavior and their relationship from practical perspective

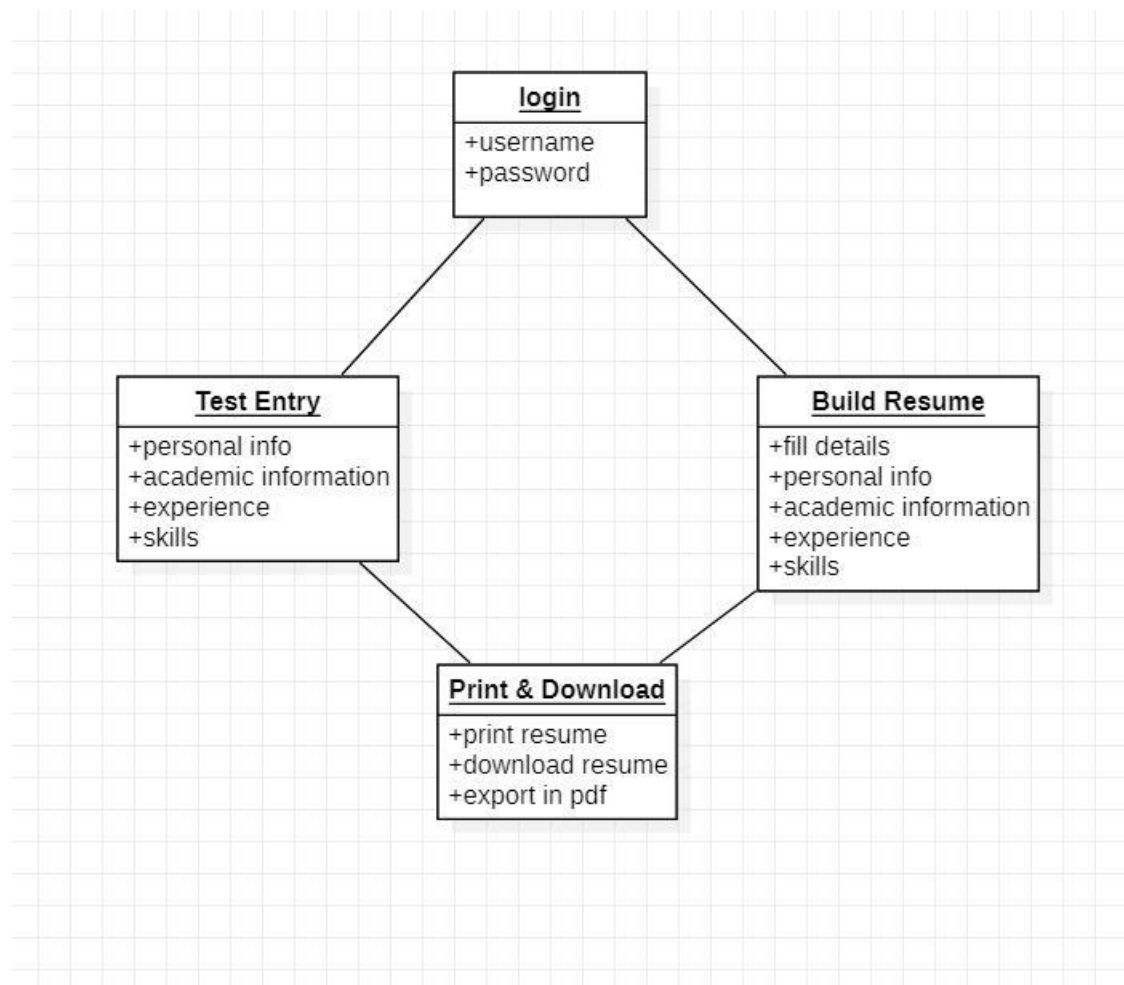
How to Draw an Object Diagram?

We have already discussed that an object diagram is an instance of a class diagram. It implies that an object diagram consists of instances of things used in a class diagram.

So, both diagrams are made of same basic elements but in different form. In class diagram elements are in abstract form to represent the blue print and in object diagram the elements are in concrete form to represent the real-world object.

Hence, the solution is –

- First, analyze the system and decide which instances have important data and association.
- Second, consider only those instances, which will cover the functionality.
- Object diagrams consist of objects.
- The link in object diagram is used to connect objects.
- Objects and links are the two elements used to construct an object diagram.
- After this, the following things are to be decided before starting the construction of the diagram –
- The object diagram should have a meaningful name to indicate its purpose.



Component Diagram –

Component diagrams are different in terms of nature and behavior. Component diagrams are used to model the physical aspects of a system. Now the question is, what are these physical aspects? Physical aspects are the elements such as executables, libraries, files, documents, etc. which reside in a node.

Component diagrams are used to visualize the organization and relationships among components in a system. These diagrams are also used to make executable systems.

How to Draw a Component Diagram?

Component diagrams are used to describe the physical artifacts of a system. This artifact includes files, executables, libraries, etc.

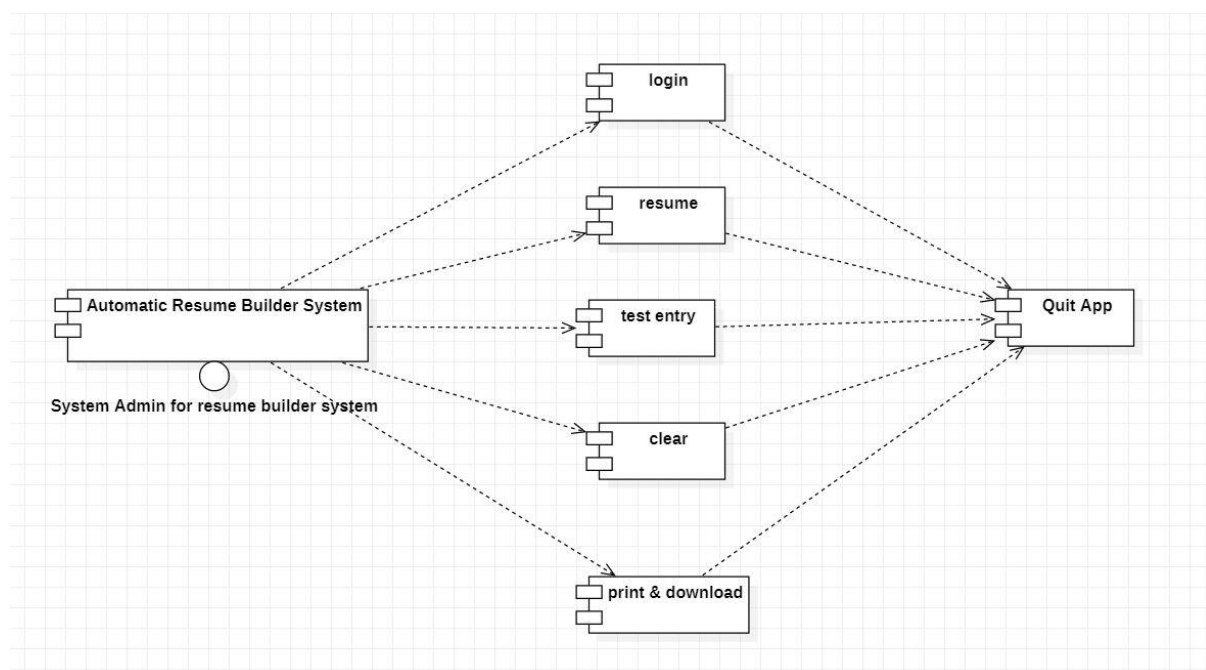
The purpose of this diagram is different. Component diagrams are used during the implementation phase of an application. However, it is prepared well in advance to visualize the implementation details.

Initially, the system is designed using different UML diagrams and then when the artifacts are ready, component diagrams are used to get an idea of the implementation.

This diagram is very important as without it the application cannot be implemented efficiently. A well-prepared component diagram is also important for other aspects such as application performance, maintenance, etc.

Before drawing a component diagram, the following artifacts are to be identified clearly –

- Files used in the system.
- Libraries and other artifacts relevant to the application.
- Relationships among the artifacts.



BEHAVIORAL DIAGRAM:

State Machine Diagram –

State Diagram are used to capture the behaviour of a software system. UML State machine diagrams can be used to model the behaviour of a class, a subsystem, a package, or even an entire system. It is also called a State chart or State Transition diagram.

What is a State chart Diagram?

State chart diagrams provide us an efficient way to model the interactions or communication that occur within the external entities and a system. These diagrams are used to model the event-based system.

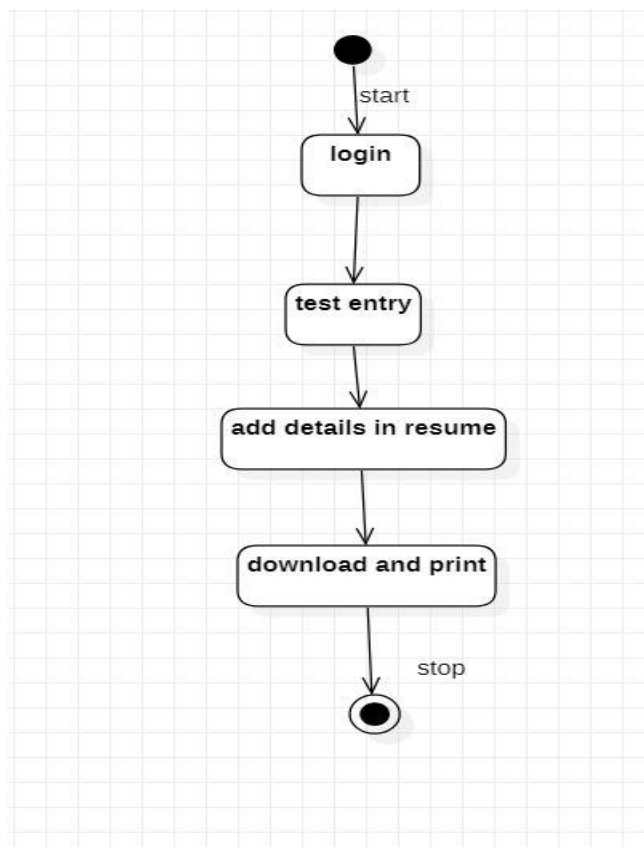
There is a total of two types of state machine diagram in UML:

1) Behavioral state machine

- It captures the behavior of an entity present in the system.
- It is used to represent the specific implementation of an element.

2) Protocol state machine

- These diagrams are used to capture the behavior of a protocol.
- It represents how the state of protocol changes concerning the event. It also represents corresponding changes in the system.



Use Case Diagram –

To model a system, the most important aspect is to capture the dynamic behavior. Dynamic behavior means the behavior of the system when it is running/operating.

Only static behavior is not sufficient to model a system rather dynamic behavior is more important than static behavior. In UML, there are five diagrams available to model the dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction.

These internal and external agents are known as actors. Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system.

Hence to model the entire system, a number of use case diagrams are used.

Purpose of Use Case Diagrams

The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams.

Use case diagrams are used to gather the requirements of a system including internal and external influences. These requirements are mostly design requirements. Hence, when a system is analyzed to gather its functionalities, use cases are prepared and actors are identified.

When the initial task is complete, use case diagrams are modelled to present the outside view.

In brief, the purposes of use case diagrams can be said to be as follows –

- Used to gather the requirements of a system.
- Used to get an outside view of a system.
- Identify the external and internal factors influencing the system.
- Show the interaction among the requirements are actors.

How to Draw a Use Case Diagram?

Use case diagrams are considered for high level requirement analysis of a system. When the requirements of a system are analyzed, the functionalities are captured in use cases.

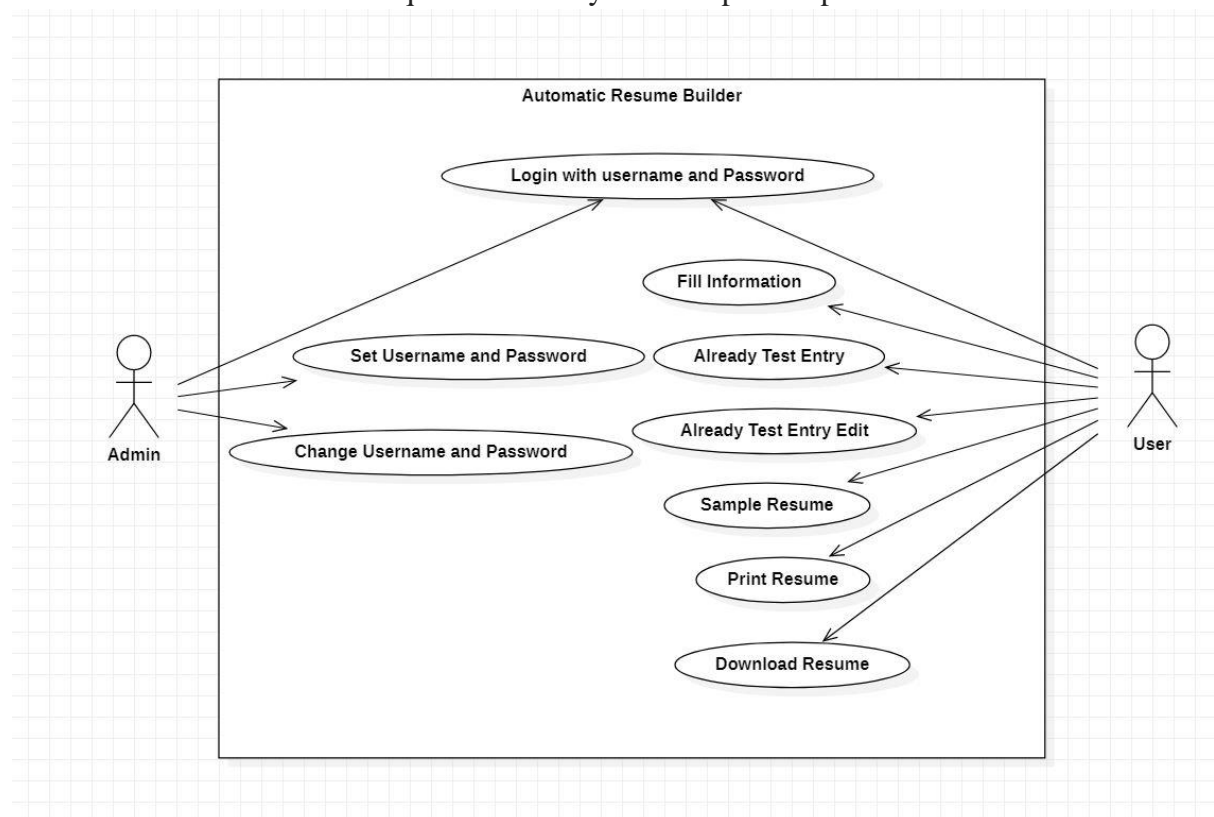
We can say that use cases are nothing but the system functionalities written in an organized manner. The second thing which is relevant to use cases are the actors. Actors can be defined as something that interacts with the system.

Actors can be a human user, some internal applications, or may be some external applications. When we are planning to draw a use case diagram, we should have the following items identified.

- Functionalities to be represented as use case
- Actors
- Relationships among the use cases and actors.

Use case diagrams are drawn to capture the functional requirements of a system. After identifying the above items, we have to use the following guidelines to draw an efficient use case diagram

- The name of a use case is very important. The name should be chosen in such a way so that it can identify the functionalities performed.
- Give a suitable name for actors.
- Show relationships and dependencies clearly in the diagram.
- Do not try to include all types of relationships, as the main purpose of the diagram is to identify the requirements.
- Use notes whenever required to clarify some important points.



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.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc.

Purpose of Activity Diagrams

The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

The purpose of an activity diagram can be described as –

- Draw the activity flow of a system.
- Describe the sequence from one activity to another.
- Describe the parallel, branched and concurrent flow of the system.

How to Draw an Activity Diagram?

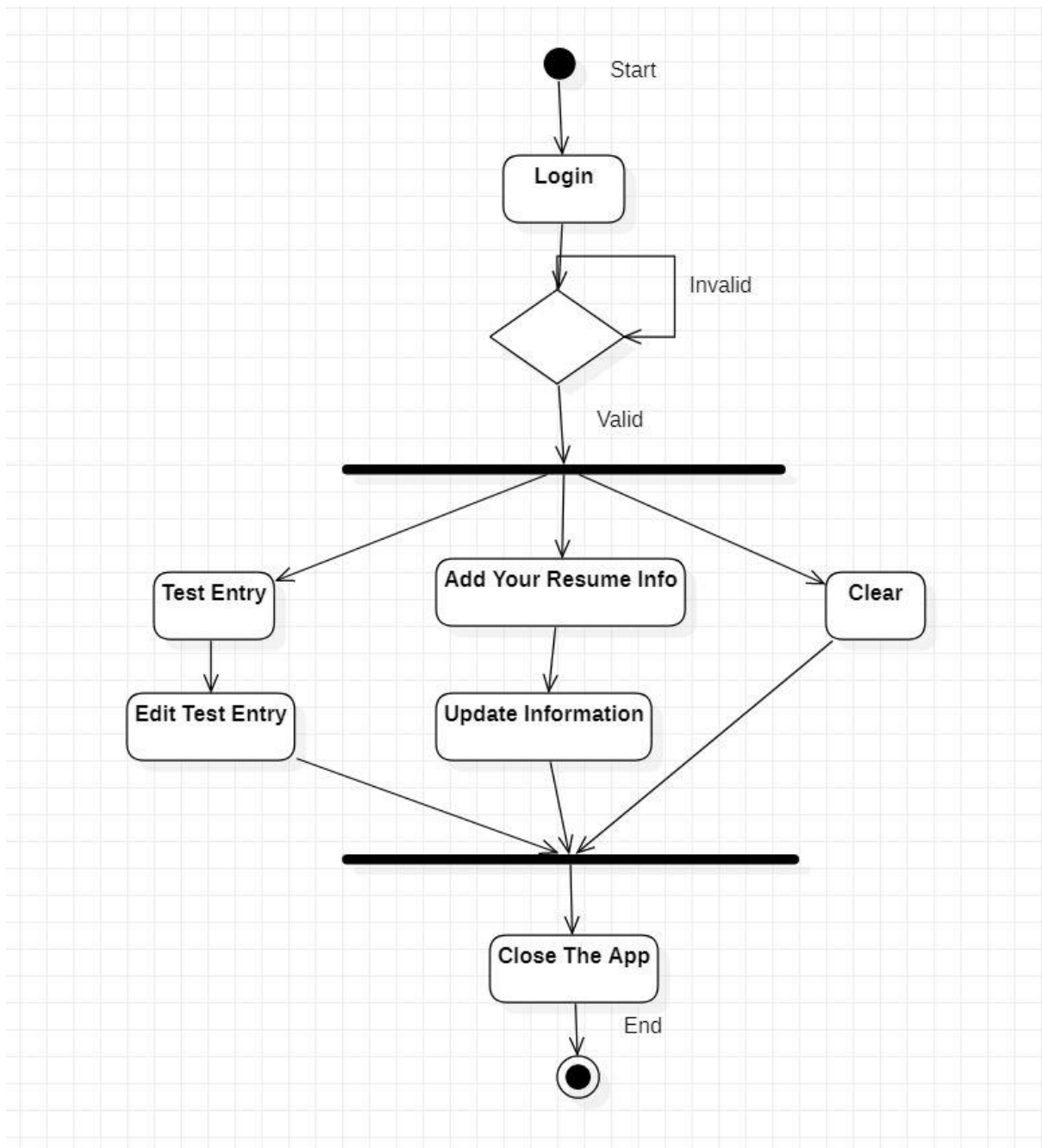
Activity diagrams are mainly used as a flowchart that consists of activities performed by the system. Activity diagrams are not exactly flowcharts as they have some additional capabilities. These additional capabilities include branching, parallel flow, swim lane, etc.

Before drawing an activity diagram, we must have a clear understanding about the elements used in activity diagram. The main element of an activity diagram is the activity itself. An activity is a function performed by the system. After identifying the activities, we need to understand how they are associated with constraints and conditions.

Before drawing an activity diagram, we should identify the following elements –

- Activities
- Association
- Conditions
- Constraints

Once the above-mentioned parameters are identified, we need to make a mental layout of the entire flow. This mental layout is then transformed into an activity diagram.



Sequence Diagram –

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together. These diagrams are used by software developers and business professionals to understand requirements for a new system or to document an existing process. Sequence diagrams are sometimes known as event diagrams or event scenarios.

Benefits of sequence diagrams

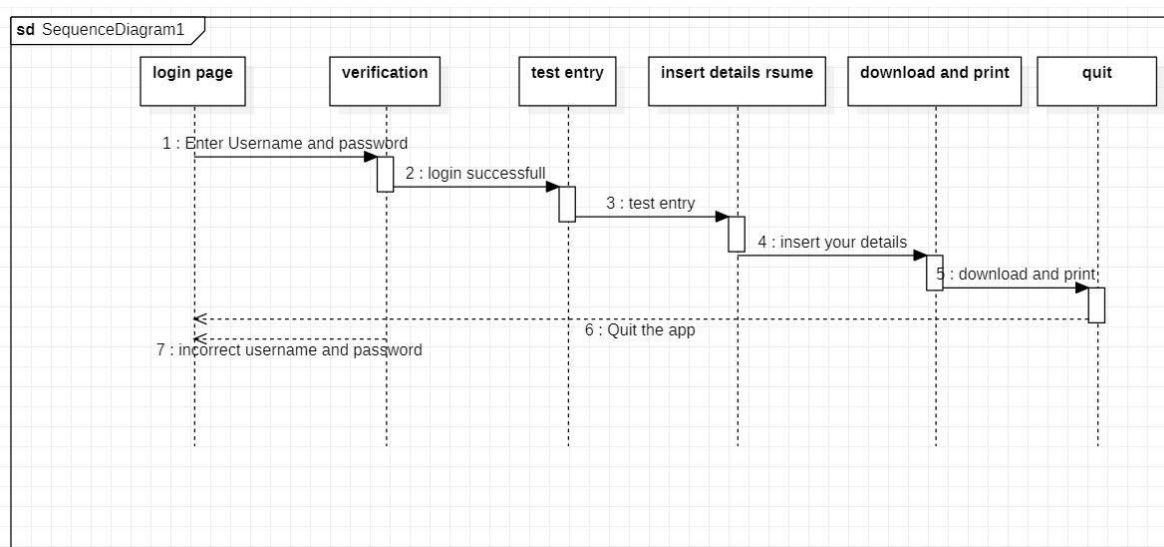
Sequence diagrams can be useful references for businesses and other organizations. Try drawing a sequence diagram to:

- Represent the details of a UML use case.
- Model the logic of a sophisticated procedure, function, or operation.
- See how objects and components interact with each other to complete a process.
- Plan and understand the detailed functionality of an existing or future scenario.

Use cases for sequence diagrams

The following scenarios are ideal for using a sequence diagram:

- **Usage scenario:** A usage scenario is a diagram of how your system could potentially be used. It's a great way to make sure that you have worked through the logic of every usage scenario for the system.
- **Method logic:** Just as you might use a UML sequence diagram to explore the logic of a use case, you can use it to explore the logic of any function, procedure, or complex process.
- **Service logic:** If you consider a service to be a high-level method used by different clients, a sequence diagram is an ideal way to map that out.



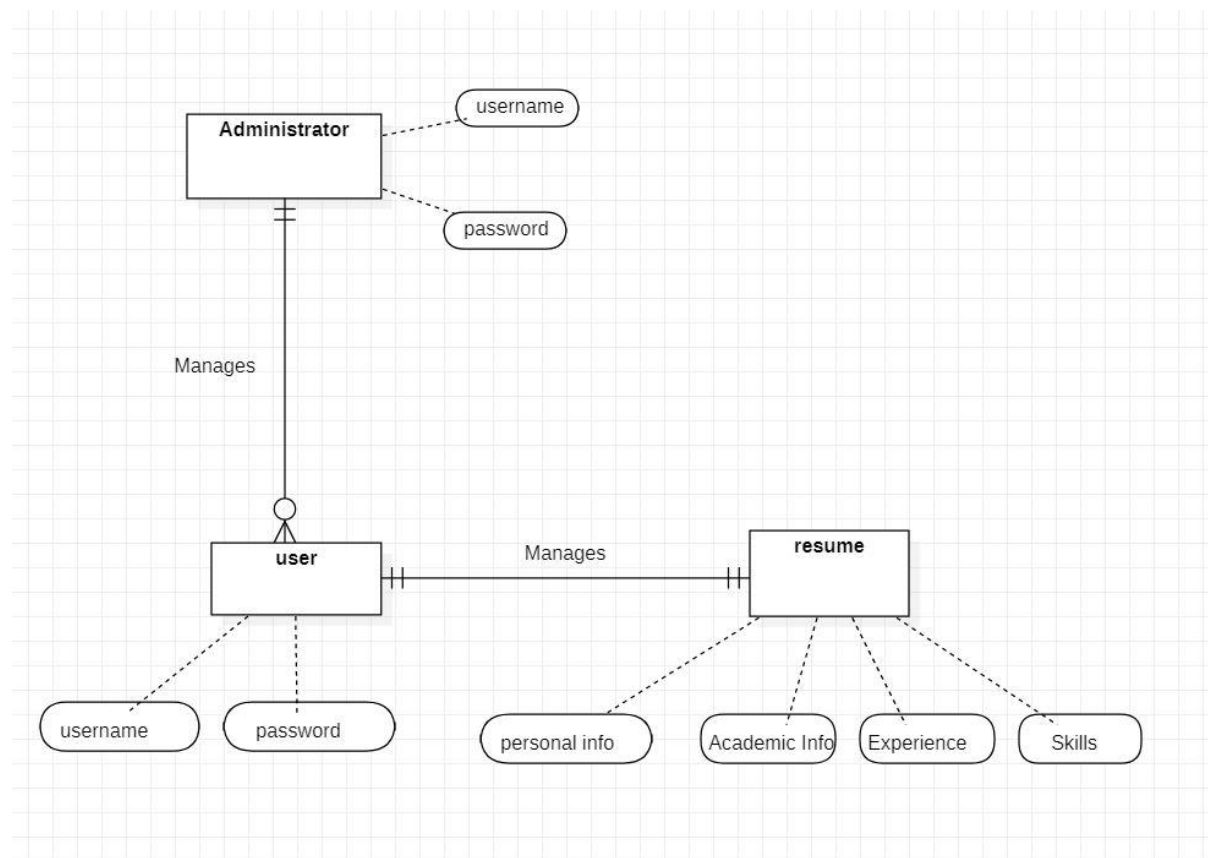
ER Diagram:

ER Model is used to model the logical view of the system from data perspective which consists of these components:

Entity, Entity Type, Entity Set –

An Entity may be an object with a physical existence – a particular person, car, house, or employee – or it may be an object with a conceptual existence – a company, a job, or a university course.

An Entity is an object of Entity Type and set of all entities is called as entity set. e.g.; E1 is an entity having Entity Type Student and set of all students is called Entity Set.



4. IMPLEMENTATION & RESULTS

Below are the implementation and results of the project.

4.1 IMPLEMENTATION CODE

Login.java:

```
package CV;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JTextField;
import java.awt.Font;
import java.awt.Image;
import javax.swing.JPasswordField;
import javax.swing.JCheckBox;
import javax.swing.JButton;
import java.awt.Window.Type;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JMenuBar;
import javax.swing.JPanel;
import javax.swing.ImageIcon;
import java.awt.Cursor;

public class Login {
    JFrame frmLogin;
    private JPasswordField pwdPassword;
    private JTextField user;
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        int x=0;
        preview frame = new preview();
        frame.setVisible(true);

        try {

            for(x= 0 ; x<=100 ; x++) {
                Thread.sleep(50);
                frame.progressBar.setValue(x);
                frame.label.setText(Integer.toString(x)+" %");
            }
        }
    }
}
```

```

        } catch (Exception e) {
            e.printStackTrace();
        }
        if(x>=100) {
            frame.dispose();
        }
    }
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                Login window = new Login();
                window.frmLogin.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}
/**
 * Create the application.
 */
public Login() {
    initialize();
}
/**
 * Initialize the contents of the frame.
 */
private void initialize() {
    frmLogin = new JFrame();
    frmLogin.getContentPane().setBackground(new Color(128, 0, 128));
    frmLogin.setType(Type.UTILITY);
    frmLogin.setTitle("Login");
    frmLogin.setBounds(400, 100, 562, 429);
    ImageIcon icon1 = new ImageIcon(this.getClass().getResource("icon.png"));
    frmLogin.setIconImage(icon1.getImage());
    frmLogin.getContentPane().setLayout(null);

    JPanel panel = new JPanel();
    panel.setBackground(Color.BLACK);
    panel.setBounds(222, 0, 324, 365);
    frmLogin.getContentPane().add(panel);
    panel.setLayout(null);

    JLabel label = new JLabel("User Name");
    label.setForeground(new Color(255, 255, 255));
    label.setBackground(new Color(220, 220, 220));
    label.setFont(new Font("Times New Roman", Font.BOLD, 14));
    label.setBounds(28, 117, 83, 17);
    panel.add(label);

```

```

JButton btnLogin = new JButton("Login>>");

btnLogin.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String un=user.getText();
        @SuppressWarnings("deprecation")
        String ps=pwdPassword.getText();
        if(un.equalsIgnoreCase("rahul") && ps.equalsIgnoreCase("rahul")) {
            Resume_Builder window = new Resume_Builder();
            window.frmResumebuilder.setVisible(true);
        } else {
            JOptionPane.showMessageDialog(null,"User Name Or
Password Incorrect");
        }
    }
});

btnLogin.setFont(new Font("Tahoma", Font.BOLD, 16));
btnLogin.setForeground(Color.WHITE);
btnLogin.setBackground(Color.RED);
btnLogin.setBounds(148, 248, 116, 35);
panel.add(btnLogin);

JCheckBox checkBox = new JCheckBox("Show");

checkBox.setCursor(Cursor.getPredefinedCursor(Cursor.DEFAULT_CURSOR));
checkBox.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        if(checkBox.isSelected()) {
            pwdPassword.setEchoChar((char)0);
        }
        else {
            pwdPassword.setEchoChar((char)2);
        }
    }
});
checkBox.setForeground(new Color(255, 255, 255));
checkBox.setBackground(new Color(0, 0, 0));
checkBox.setBounds(134, 193, 61, 23);
panel.add(checkBox);

JLabel label_1 = new JLabel("Password");
label_1.setForeground(new Color(255, 255, 255));
label_1.setBackground(new Color(220, 220, 220));
label_1.setFont(new Font("Times New Roman", Font.BOLD, 14));
label_1.setBounds(39, 163, 83, 17);
panel.add(label_1);

pwdPassword = new JPasswordField();
pwdPassword.setEchoChar('*');

```

```

pwdPassword.setForeground(Color.BLACK);
pwdPassword.setBackground(Color.WHITE);
pwdPassword.setFont(new Font("Tahoma", Font.BOLD, 15));
pwdPassword.setBounds(101, 157, 188, 29);
panel.add(pwdPassword);

user = new JTextField();
user.setCursor(Cursor.getPredefinedCursor(Cursor.TEXT_CURSOR));
user.setForeground(Color.BLACK);
user.setBackground(Color.WHITE);
user.setFont(new Font("Tahoma", Font.BOLD, 15));
user.setColumns(10);
user.setBounds(101, 111, 188, 28);
panel.add(user);
JLabel logo = new JLabel("");
Image img = new
ImageIcon(this.getClass().getResource("login.png")).getImage();
logo.setIcon(new ImageIcon(img));
logo.setBounds(-11, 56, 224, 264);
frmLogin.getContentPane().add(logo);

JMenuBar menuBar = new JMenuBar();
menuBar.setBackground(new Color(0, 0, 0));
frmLogin.setJMenuBar(menuBar);

JButton btnAbout = new JButton("About");
btnAbout.setForeground(new Color(255, 255, 255));
btnAbout.setBackground(new Color(0, 0, 0));
btnAbout.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        About window = new About();
        window.frmAbout.setVisible(true);
    }
});
menuBar.add(btnAbout);
}
}

```

About.java:

```

package CV;
import java.awt.EventQueue;
import javax.swing.JFrame;
import java.awt.Color;
import javax.swing.JLabel;
import javax.swing.JTextArea;
import java.awt.Font;
import java.awt.Window.Type;

public class About {

```

```

JFrame frmAbout;

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                About window = new About();
                window.frmAbout.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the application.
 */
public About() {
    initialize();
}

/**
 * Initialize the contents of the frame.
 */
private void initialize() {
    frmAbout = new JFrame();
    frmAbout.setType(Type.UTILITY);
    frmAbout.setTitle("About");
    frmAbout.getContentPane().setBackground(new Color(250, 250, 210));
    frmAbout.setBounds(400, 100, 450, 247);
    frmAbout.getContentPane().setLayout(null);

    JLabel lblAboutApp = new JLabel("About App");
    lblAboutApp.setFont(new Font("Tahoma", Font.BOLD, 18));
    lblAboutApp.setBounds(167, 11, 137, 43);
    frmAbout.getContentPane().add(lblAboutApp);

    JTextArea txtrThisAppIs = new JTextArea();
    txtrThisAppIs.setEditable(false);
    txtrThisAppIs.setBackground(new Color(250, 250, 210));
    txtrThisAppIs.setText("This app is Created By Raghu and Rahul ");
    txtrThisAppIs.setBounds(21, 47, 403, 22);
    frmAbout.getContentPane().add(txtrThisAppIs);

    JTextArea txtrH = new JTextArea();

```

```

        txtrH.setEditable(false);
        txtrH.setBackground(new Color(250, 250, 210));
        txtrH.setText("Roll Number :- 10 & 27");
        txtrH.setBounds(21, 75, 403, 33);
        frmAbout.getContentPane().add(txtrH);

        JTextArea txtrOopLabProject = new JTextArea();
        txtrOopLabProject.setEditable(false);
        txtrOopLabProject.setBackground(new Color(250, 250, 210));
        txtrOopLabProject.setText("Java lab Project: Automatic Resume Builder");
        txtrOopLabProject.setBounds(21, 108, 403, 22);
        frmAbout.getContentPane().add(txtrOopLabProject);

        JTextArea txtrUserNameIqra = new JTextArea();
        txtrUserNameIqra.setEditable(false);
        txtrUserNameIqra.setBackground(new Color(250, 250, 210));
        txtrUserNameIqra.setText("User Name: rahul");
        txtrUserNameIqra.setBounds(21, 138, 380, 33);
        frmAbout.getContentPane().add(txtrUserNameIqra);

        JTextArea txtrPasswordIqra = new JTextArea();
        txtrPasswordIqra.setEditable(false);
        txtrPasswordIqra.setBackground(new Color(250, 250, 210));
        txtrPasswordIqra.setText("Password: rahul");
        txtrPasswordIqra.setBounds(21, 169, 391, 33);
        frmAbout.getContentPane().add(txtrPasswordIqra);
    }
}

```

Preview.java:

```

package CV;
import java.awt.BorderLayout;
import java.awt.EventQueue;
import javax.swing.Icon;
import javax.swing.ImageIcon;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import java.awt.Color;
import java.awt.Window.Type;
import javax.swing.JProgressBar;
import javax.swing.JLabel;
import javax.swing.SwingConstants;
import java.awt.Font;
import javax.swing.border.LineBorder;

public class preview extends JFrame {
    private JPanel contentPane;

```

```

public JProgressBar progressBar;
public JLabel label;
private JLabel label_1;
public JLabel lblWelcome;
public JPanel panel;
private JLabel lblLoading;

public preview() {
    setUndecorated(true);
    setType(Type.POPUP);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(400, 100, 562, 500);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    setContentPane(contentPane);
    contentPane.setLayout(null);

    panel = new JPanel();
    panel.setBackground(new Color(32, 32, 32));
    panel.setBounds(0, -11, 603, 515);
    contentPane.add(panel);
    panel.setLayout(null);

    progressBar = new JProgressBar();
    progressBar.setBorder(new LineBorder(Color.DARK_GRAY));
    progressBar.setForeground(new Color(0, 255, 127));
    progressBar.setBackground(new Color(32, 32, 32));
    progressBar.setBounds(23, 427, 519, 23);
    panel.add(progressBar);

    label = new JLabel("0%");
    label.setForeground(new Color(0, 206, 209));
    label.setFont(new Font("Times New Roman", Font.BOLD, 16));
    label.setHorizontalAlignment(SwingConstants.CENTER);
    label.setBounds(270, 373, 48, 32);
    panel.add(label);

    Icon icon2 = new ImageIcon(this.getClass().getResource("pro.gif"));
    label_1 = new JLabel(icon2);
    label_1.setBounds(10, 110, 529, 223);
    panel.add(label_1);

    lblWelcome = new JLabel("Welcome to Resume Builder");
    lblWelcome.setForeground(new Color(255, 255, 255));
    lblWelcome.setFont(new Font("Times New Roman", Font.BOLD, 36));
    lblWelcome.setBounds(47, 44, 492, 42);
    panel.add(lblWelcome);

    lblLoading = new JLabel("Loading.....");

```



```

        lblLoading.setForeground(Color.CYAN);
        lblLoading.setFont(new Font("Times New Roman", Font.BOLD, 16));
        lblLoading.setBounds(269, 344, 72, 23);
        panel.add(lblLoading);
    }
}

```

Sample1.java:

```

package CV;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JPanel;
import java.awt.Color;
import javax.swing.JLabel;
import java.awt.Font;
import java.awt.Image;
import java.awt.Window.Type;
import javax.swing.JTextField;
import javax.swing.border.LineBorder;
import javax.swing.ImageIcon;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class Sample1 {
    JFrame frmSample;
    JTextField name;
    JTextField father;
    JTextField phone;
    JTextField email1;
    JTextField homeaddress;
    JTextField gpro;
    JTextField ggrade;
    JTextField gy;
    JTextField ipro;
    JTextField igrade;
    JTextField iy;
    JTextField mpro;
    JTextField mgrade;
    JTextField my;
    JTextField C1;
    JTextField C2;
    JTextField C3;
    JTextField D1;
    JTextField D2;
    JTextField D3;
    JTextField Y1;
    JTextField Y2;
}

```

```

JTextField Y3;
JTextField C4;
JTextField D4;
JTextField Y4;
JTextField s1;
JTextField s2;
JTextField s3;
JTextField s4;
JTextField l1;
JTextField l3;
JTextField l2;
JTextField l4;
JTextField S1;
JTextField S2;
JLabel finalphoto;
private JLabel lblDob;
JTextField Dob;
private JLabel label;

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                Sample1 window = new Sample1();
                window.frmSample.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the application.
 */
public Sample1() {
    initialize();
}

/**
 * Initialize the contents of the frame.
 */
private void initialize() {
    @SuppressWarnings("unused")
    Resume_Builder rb=new Resume_Builder();

    frmSample = new JFrame();

```

```

frmSample.setBackground(Color.DARK_GRAY);
frmSample.getContentPane().setBackground(new Color(255, 255, 255));
frmSample.setType(Type.UTILITY);
frmSample.setTitle("Sample 1");
ImageIcon icon1 = new ImageIcon(this.getClass().getResource("icon.png"));
frmSample.setIconImage(icon1.getImage());
frmSample.setBounds(200,0, 713, 798);
frmSample.getContentPane().setLayout(null);

JPanel panel = new JPanel();
panel.setBackground(new Color(255, 255, 255));
panel.setBounds(0, 23, 697, 736);
frmSample.getContentPane().add(panel);
panel.setLayout(null);

name = new JTextField();
name.setForeground(new Color(50, 105, 225));
name.setText("    ABC");
name.setFont(new Font("Arial", Font.BOLD, 34));
JButton btnView = new JButton("Print");
btnView.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        @SuppressWarnings("rawtypes")
        Print p = new Print();
        p.print(panel);
    }
});
btnView.setBounds(40, 0, 89, 23);
frmSample.getContentPane().add(btnView);
name.setBorder(new LineBorder(new Color(255, 255, 255)));
name.setBounds(40, 21, 406, 48);
panel.add(name);
name.setColumns(10);

JLabel lblSo = new JLabel("S.O:");
lblSo.setFont(new Font("Tahoma", Font.BOLD, 13));
lblSo.setBounds(10, 80, 46, 23);
panel.add(lblSo);

father = new JTextField();
father.setBorder(new LineBorder(Color.WHITE));
father.setBounds(50, 83, 199, 20);
panel.add(father);
father.setColumns(10);

JLabel phonelogo = new JLabel("");

phonelogo.setBounds(24, 134, 25, 31);
Image img1 = new
ImageIcon(this.getClass().getResource("phone.png")).getImage();

```

```

        phonelogo.setIcon(new ImageIcon(img1));
        panel.add(phonelogo);

        phone = new JTextField();
        phone.setBorder(new LineBorder(Color.WHITE));
        phone.setBounds(48, 142, 201, 20);
        panel.add(phone);
        phone.setColumns(10);

        JLabel mail = new JLabel("");

        mail.setBounds(20, 165, 32, 31);
        Image img2 = new
ImageIcon(this.getClass().getResource("mail.png")).getImage();
        mail.setIcon(new ImageIcon(img2));
        panel.add(mail);

        JLabel address = new JLabel("");

        address.setBounds(24, 195, 25, 31);
        Image img3 = new
ImageIcon(this.getClass().getResource("address.png")).getImage();
        address.setIcon(new ImageIcon(img3));
        panel.add(address);

        email1 = new JTextField();
        email1.setBorder(new LineBorder(Color.WHITE));
        email1.setForeground(new Color(30, 144, 255));
        email1.setBounds(50, 168, 258, 20);
        panel.add(email1);
        email1.setColumns(10);

        homeaddress = new JTextField();
        homeaddress.setBorder(new LineBorder(Color.WHITE));
        homeaddress.setBounds(50, 195, 327, 20);
        panel.add(homeaddress);
        homeaddress.setColumns(10);

        JLabel label_2 = new JLabel("");
        label_2.setBorder(new LineBorder(new Color(255, 0, 0), 3));
        label_2.setBounds(10, 250, 676, 4);
        panel.add(label_2);

        JLabel lblObjectives = new JLabel("OBJECTIVES:");
        lblObjectives.setFont(new Font("Arial", Font.BOLD, 30));
        lblObjectives.setForeground(Color.BLUE);
        lblObjectives.setBackground(Color.WHITE);
        lblObjectives.setBounds(50, 215, 327, 31);
        panel.add(lblObjectives);

```

```
JLabel lblOrganizedAndMotivated = new JLabel("Organized and motivated  
employee able to apply time management and organizational skills in");  
lblOrganizedAndMotivated.setFont(new Font("Arial", Font.PLAIN, 14));  
lblOrganizedAndMotivated.setBounds(50, 265, 636, 23);  
panel.add(lblOrganizedAndMotivated);
```

```
JLabel lblAndOrganizationalSkills = new JLabel("various environments.  
Seeking to join Acme Corp as an administrative assistant to help ensure");  
lblAndOrganizationalSkills.setFont(new Font("Tahoma", Font.PLAIN, 14));  
lblAndOrganizationalSkills.setBounds(50, 288, 615, 23);  
panel.add(lblAndOrganizationalSkills);
```

```
JLabel lblSeekingToJoin = new JLabel("good internal communications and  
budget management.");  
lblSeekingToJoin.setFont(new Font("Arial", Font.PLAIN, 14));  
lblSeekingToJoin.setBounds(50, 314, 615, 23);  
panel.add(lblSeekingToJoin);
```

```
JLabel label_3 = new JLabel("");  
label_3.setBorder(new LineBorder(new Color(255, 0, 0), 3));  
label_3.setBounds(10, 379, 676, 4);  
panel.add(label_3);
```

```
JLabel lblAcademicQualification = new JLabel("ACADEMIC  
QUALIFICATION:");  
lblAcademicQualification.setForeground(Color.BLUE);  
lblAcademicQualification.setFont(new Font("Arial", Font.BOLD, 30));  
lblAcademicQualification.setBackground(Color.WHITE);  
lblAcademicQualification.setBounds(50, 337, 421, 31);  
panel.add(lblAcademicQualification);
```

```
JLabel lblProgram = new JLabel("Program:");  
lblProgram.setFont(new Font("Arial", Font.BOLD, 13));  
lblProgram.setBounds(20, 414, 72, 14);  
panel.add(lblProgram);
```

```
JLabel lblGrade = new JLabel("Grade:");  
lblGrade.setFont(new Font("Arial", Font.BOLD, 13));  
lblGrade.setBounds(20, 439, 72, 14);  
panel.add(lblGrade);
```

```
JLabel lblPassingYear = new JLabel("Passing Year:");  
lblPassingYear.setFont(new Font("Arial", Font.BOLD, 13));  
lblPassingYear.setBounds(20, 466, 94, 14);  
panel.add(lblPassingYear);
```

```
JLabel lblGraduation = new JLabel("Graduation:");  
lblGraduation.setForeground(Color.RED);  
lblGraduation.setFont(new Font("Arial", Font.BOLD, 14));  
lblGraduation.setBounds(116, 387, 94, 23);
```

```

panel.add(lblGraduation);

gpro = new JTextField();
gpro.setBorder(new LineBorder(Color.WHITE));
gpro.setBounds(119, 412, 127, 20);
panel.add(gpro);
gpro.setColumns(10);

ggrade = new JTextField();
ggrade.setBorder(new LineBorder(Color.WHITE));
ggrade.setColumns(10);
ggrade.setBounds(119, 437, 127, 20);
panel.add(ggrade);

gy = new JTextField();
gy.setBorder(new LineBorder(Color.WHITE));
gy.setColumns(10);
gy.setBounds(119, 464, 127, 20);
panel.add(gy);

JLabel lblIntermediate = new JLabel("Intermediate:");
lblIntermediate.setForeground(Color.RED);
lblIntermediate.setFont(new Font("Arial", Font.BOLD, 14));
lblIntermediate.setBounds(306, 387, 94, 23);
panel.add(lblIntermediate);

ipro = new JTextField();
ipro.setBorder(new LineBorder(Color.WHITE));
ipro.setColumns(10);
ipro.setBounds(309, 412, 127, 20);
panel.add(ipro);

igrade = new JTextField();
igrade.setBorder(new LineBorder(Color.WHITE));
igrade.setColumns(10);
igrade.setBounds(309, 437, 127, 20);
panel.add(igrade);

iy = new JTextField();
iy.setBorder(new LineBorder(Color.WHITE));
iy.setColumns(10);
iy.setBounds(309, 464, 127, 20);
panel.add(iy);

JLabel lblMatriculation = new JLabel("Matriculation:");
lblMatriculation.setForeground(Color.RED);
lblMatriculation.setFont(new Font("Arial", Font.BOLD, 14));
lblMatriculation.setBounds(505, 387, 111, 23);
panel.add(lblMatriculation);

```

```

mpro = new JTextField();
mpro.setBorder(new LineBorder(Color.WHITE));
mpro.setColumns(10);
mpro.setBounds(508, 412, 127, 20);
panel.add(mpro);

mgrade = new JTextField();
mgrade.setBorder(new LineBorder(Color.WHITE));
mgrade.setColumns(10);
mgrade.setBounds(508, 437, 127, 20);
panel.add(mgrade);

my = new JTextField();
my.setBorder(new LineBorder(Color.WHITE));
my.setColumns(10);
my.setBounds(508, 464, 127, 20);
panel.add(my);

JLabel lblExperience = new JLabel("EXPERIENCE:");
lblExperience.setForeground(Color.BLUE);
lblExperience.setFont(new Font("Arial", Font.BOLD, 30));
lblExperience.setBackground(Color.WHITE);
lblExperience.setBounds(50, 495, 210, 31);
panel.add(lblExperience);

JLabel lblSkills = new JLabel("SKILLS:");
lblSkills.setForeground(Color.BLUE);
lblSkills.setFont(new Font("Arial", Font.BOLD, 30));
lblSkills.setBackground(Color.WHITE);
lblSkills.setBounds(476, 495, 210, 31);
panel.add(lblSkills);

C1 = new JTextField();
C1.setBorder(new LineBorder(Color.WHITE));
C1.setColumns(10);
C1.setBounds(24, 524, 151, 20);
panel.add(C1);

C2 = new JTextField();
C2.setBorder(new LineBorder(Color.WHITE));
C2.setColumns(10);
C2.setBounds(24, 549, 151, 20);
panel.add(C2);

C3 = new JTextField();
C3.setBorder(new LineBorder(Color.WHITE));
C3.setColumns(10);
C3.setBounds(24, 576, 151, 20);
panel.add(C3);

```

```
D1 = new JTextField();
D1.setBorder(new LineBorder(Color.WHITE));
D1.setColumns(10);
D1.setBounds(185, 524, 127, 20);
panel.add(D1);
```

```
D2 = new JTextField();
D2.setBorder(new LineBorder(Color.WHITE));
D2.setColumns(10);
D2.setBounds(185, 549, 127, 20);
panel.add(D2);
```

```
D3 = new JTextField();
D3.setBorder(new LineBorder(Color.WHITE));
D3.setColumns(10);
D3.setBounds(185, 576, 127, 20);
panel.add(D3);
```

```
Y1 = new JTextField();
Y1.setBorder(new LineBorder(Color.WHITE));
Y1.setColumns(10);
Y1.setBounds(319, 524, 127, 20);
panel.add(Y1);
```

```
Y2 = new JTextField();
Y2.setBorder(new LineBorder(Color.WHITE));
Y2.setColumns(10);
Y2.setBounds(319, 549, 127, 20);
panel.add(Y2);
```

```
Y3 = new JTextField();
Y3.setBorder(new LineBorder(Color.WHITE));
Y3.setColumns(10);
Y3.setBounds(319, 576, 127, 20);
panel.add(Y3);
```

```
C4 = new JTextField();
C4.setBorder(new LineBorder(Color.WHITE));
C4.setColumns(10);
C4.setBounds(24, 600, 151, 20);
panel.add(C4);
```

```
D4 = new JTextField();
D4.setBorder(new LineBorder(Color.WHITE));
D4.setColumns(10);
D4.setBounds(185, 600, 127, 20);
panel.add(D4);
```

```
Y4 = new JTextField();
Y4.setBorder(new LineBorder(Color.WHITE));
```



```

Y4.setColumns(10);
Y4.setBounds(319, 600, 127, 20);
panel.add(Y4);

s1 = new JTextField();
s1.setBorder(new LineBorder(Color.WHITE));
s1.setColumns(10);
s1.setBounds(486, 524, 179, 20);
panel.add(s1);

s2 = new JTextField();
s2.setBorder(new LineBorder(Color.WHITE));
s2.setColumns(10);
s2.setBounds(486, 549, 179, 20);
panel.add(s2);

s3 = new JTextField();
s3.setBorder(new LineBorder(Color.WHITE));
s3.setColumns(10);
s3.setBounds(486, 576, 179, 20);
panel.add(s3);

s4 = new JTextField();
s4.setBorder(new LineBorder(Color.WHITE));
s4.setColumns(10);
s4.setBounds(486, 600, 179, 20);
panel.add(s4);

JLabel lblLanguages = new JLabel("LANGUAGES:");
lblLanguages.setForeground(Color.BLUE);
lblLanguages.setFont(new Font("Arial", Font.BOLD, 30));
lblLanguages.setBackground(Color.WHITE);
lblLanguages.setBounds(24, 631, 210, 31);
panel.add(lblLanguages);

JLabel lblSports = new JLabel("SPORTS:");
lblSports.setForeground(Color.BLUE);
lblSports.setFont(new Font("Arial", Font.BOLD, 30));
lblSports.setBackground(Color.WHITE);
lblSports.setBounds(476, 631, 210, 31);
panel.add(lblSports);

l1 = new JTextField();
l1.setBorder(new LineBorder(Color.WHITE));
l1.setColumns(10);
l1.setBounds(24, 668, 127, 20);
panel.add(l1);

l3 = new JTextField();
l3.setBorder(new LineBorder(Color.WHITE));

```

```

l3.setColumns(10);
l3.setBounds(24, 693, 127, 20);
panel.add(l3);

l2 = new JTextField();
l2.setBorder(new LineBorder(Color.WHITE));
l2.setColumns(10);
l2.setBounds(158, 668, 127, 20);
panel.add(l2);

l4 = new JTextField();
l4.setBorder(new LineBorder(Color.WHITE));
l4.setColumns(10);
l4.setBounds(158, 693, 127, 20);
panel.add(l4);

S1 = new JTextField();
S1.setBorder(new LineBorder(Color.WHITE));
S1.setColumns(10);
S1.setBounds(489, 668, 176, 20);
panel.add(S1);

S2 = new JTextField();
S2.setBorder(new LineBorder(Color.WHITE));
S2.setColumns(10);
S2.setBounds(489, 693, 176, 20);
panel.add(S2);

finalphoto = new JLabel("");
finalphoto.setBounds(476, 21, 189, 187);
panel.add(finalphoto);

lblDob = new JLabel("D.O.B");
lblDob.setFont(new Font("Tahoma", Font.BOLD, 11));
lblDob.setBounds(10, 114, 37, 14);
panel.add(lblDob);

Dob = new JTextField();
Dob.setColumns(10);
Dob.setBorder(new LineBorder(Color.WHITE));
Dob.setBounds(50, 111, 199, 20);
panel.add(Dob);
Resume_Builder r = new Resume_Builder();
label = new JLabel("");
label.setBounds(476, 28, 189, 187);
if(r.temp1==1) {
    Image img21 = new
ImageIcon(this.getClass().getResource("/Sample.jpg")).getImage();
    label.setIcon(new ImageIcon(img21));
}

```

```
        panel.add(label);  
    }  
}
```

Resume_Builder.java:

```
package CV;  
import java.awt.Color;  
import java.awt.EventQueue;  
import java.awt.Font;  
import java.awt.Image;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
import java.awt.event.MouseWheelEvent;  
import java.awt.event.MouseWheelListener;  
import java.awt.event.WindowAdapter;  
import java.awt.event.WindowEvent;  
import java.io.File;  
  
import javax.swing.DefaultComboBoxModel;  
import javax.swing.ImageIcon;  
import javax.swing.JButton;  
import javax.swing.JComboBox;  
import javax.swing.JFileChooser;  
import javax.swing.JFrame;  
import javax.swing.JLabel;  
import javax.swing.JLayeredPane;  
import javax.swing.JPanel;  
import javax.swing.JProgressBar;  
import javax.swing.JRadioButton;  
import javax.swing.JTextArea;  
import javax.swing.JTextField;  
import javax.swing.border.LineBorder;  
import javax.swing.SwingConstants;  
  
public class Resume_Builder extends Login{  
    JFrame frmResumebuilder;  
    private JTextField fullname;  
    private JTextField fathename;  
    private JTextField email;  
    private JTextField phonenum;  
    private JTextField desig1;  
    private JTextField comp2;  
    private JTextField desig2;  
    private JTextField comp3;  
    private JTextField desig3;  
    private JTextField comp4;  
    private JTextField desig4;
```

```

private JTextField comskill1;
private JTextField comskill2;
private JTextField comskill3;
private JTextField comskill4;
private JTextField lang3;
private JTextField lang4;
private JTextField lang2;
private JTextField lang1;
private JTextField sports1;
private JTextField sports2;
private int a=20,b=40,c=60,d=80;
private JProgressBar progressBar = new JProgressBar();
public String date[]=
{"01","02","03","04","05","06","07","08","09","10","11","12","13","14","15","16","17","18",
"19","20","21","22","23","24","25","26","27","28","29","30","31"};
public String month[]=
{"January","February","March","April","May","june","July","August","September","october",
"November","December"};
public String Year[]=
{"1970","1971","1972","1973","1974","1975","1976","1977","1978","1979","1980","1981",
"1982","1983","1984","1985","1986","1987","1988","1989","1990","1991","1992","1993","
1994","1995","1996","1997","1998","1999","2000","2001","2002","2003","2004","2005","2
006","2007","2008","2009","2010","2011","2012","2013","2014","2015","2016","2017","20
18","2019","2020"};
public String Year1[]=
{"1970","1971","1972","1973","1974","1975","1976","1977","1978","1979","1980","1981",
"1982","1983","1984","1985","1986","1987","1988","1989","1990","1991","1992","1993","
1994","1995","1996","1997","1998","1999","2000","2001","2002","2003","2004","2005","2
006","2007","2008","2009","2010","2011","2012","2013","2014","2015","2016","2017","20
18","2019","2020"};
public String Year2[]=
{"1970","1971","1972","1973","1974","1975","1976","1977","1978","1979","1980","1981",
"1982","1983","1984","1985","1986","1987","1988","1989","1990","1991","1992","1993","
1994","1995","1996","1997","1998","1999","2000","2001","2002","2003","2004","2005","2
006","2007","2008","2009","2010","2011","2012","2013","2014","2015","2016","2017","20
18","2019","2020"};
public String Year3[]=
{"1970","1971","1972","1973","1974","1975","1976","1977","1978","1979","1980","1981",
"1982","1983","1984","1985","1986","1987","1988","1989","1990","1991","1992","1993","
1994","1995","1996","1997","1998","1999","2000","2001","2002","2003","2004","2005","2
006","2007","2008","2009","2010","2011","2012","2013","2014","2015","2016","2017","20
18","2019","2020"};
public String Year4[]=
{"1970","1971","1972","1973","1974","1975","1976","1977","1978","1979","1980","1981",
"1982","1983","1984","1985","1986","1987","1988","1989","1990","1991","1992","1993","
1994","1995","1996","1997","1998","1999","2000","2001","2002","2003","2004","2005","2
006","2007","2008","2009","2010","2011","2012","2013","2014","2015","2016","2017","20
18","2019","2020"};
public String Year5[]=
{"1970","1971","1972","1973","1974","1975","1976","1977","1978","1979","1980","1981",

```



```

public String grade[]= {"A+","A","B+","B","C","D"};
private JTextField url;
public String s;
public String n;
public String filename;
        int temp1;

/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                Resume_Builder window = new
Resume_Builder();
                window.frmResumebuilder.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the application.
 */
public Resume_Builder() {
    initialize();
}

/**
 * Initialize the contents of the frame.getExtendedState(
 */
@SuppressWarnings("unchecked")
private void initialize() {
    frmResumebuilder = new JFrame();
    frmResumebuilder.getContentPane().setBackground(new Color(135,
206, 250));
    frmResumebuilder.getContentPane().addMouseListener(new
MouseListener() {
        public void mouseWheelMoved(MouseWheelEvent e) {

        }

    });
    frmResumebuilder.addMouseListener(new
MouseListener() {
        public void mouseWheelMoved(MouseWheelEvent e) {

        }
    }
}

```

```

});
frmResumebuilder.setTitle("Resume_Builder");
frmResumebuilder.addWindowListener(new WindowAdapter() {

    public void windowOpened(WindowEvent e) {

        frmResumebuilder.setExtendedState(frmResumebuilder.getExtendedState()|J
Frame.MAXIMIZED_BOTH);
    }
});
frmResumebuilder.setBounds(100, 100, 1294, 790);
ImageIcon icon1 = new
ImageIcon(this.getClass().getResource("icon.png"));
frmResumebuilder.setIconImage(icon1.getImage());

frmResumebuilder.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frmResumebuilder.getContentPane().setLayout(null);

JLayeredPane layeredPane = new JLayeredPane();
layeredPane.setBorder(new LineBorder(new Color(255, 255, 0), 3));
layeredPane.setForeground(new Color(50, 205, 50));
layeredPane.setBounds(10, 58, 345, 311);
frmResumebuilder.getContentPane().add(layeredPane);

JPanel panel = new JPanel();
panel.setBackground(new Color(135, 206, 250));
panel.setBounds(10, 11, 325, 290);
layeredPane.add(panel);
panel.setLayout(null);

JLabel lblFullName = new JLabel("Full Name:");
lblFullName.setBounds(10, 11, 72, 25);
panel.add(lblFullName);

setFullname(new JTextField());
getFullname().setBounds(92, 11, 223, 25);
panel.add(getFullname());
getFullname().setColumns(10);

JLabel lblFatherName = new JLabel("Father Name:");
lblFatherName.setBounds(10, 47, 81, 25);
panel.add(lblFatherName);

fathername = new JTextField();
fathername.setHorizontalAlignment(SwingConstants.LEFT);
fathername.setBounds(92, 47, 223, 25);
panel.add(fathername);
fathername.setColumns(10);

```

```
JLabel lblPhone = new JLabel("Phone#");
lblPhone.setBounds(10, 83, 72, 25);
panel.add(lblPhone);
```

```
phonenum = new JTextField();
phonenum.setHorizontalAlignment(SwingConstants.LEFT);
phonenum.setBounds(92, 83, 223, 25);
panel.add(phonenum);
phonenum.setColumns(10);
```

```
JLabel lblEmail = new JLabel("E-Mail:");
lblEmail.setBounds(10, 119, 72, 25);
panel.add(lblEmail);
```

```
email = new JTextField();
email.setHorizontalAlignment(SwingConstants.LEFT);
email.setForeground(new Color(30, 144, 255));
email.setBounds(92, 121, 223, 23);
panel.add(email);
email.setColumns(10);
```

```
JLabel lblDob = new JLabel("D.O.B");
lblDob.setBounds(10, 165, 48, 14);
panel.add(lblDob);
```

```
JLabel lblDay = new JLabel("Date:");
lblDay.setBounds(92, 145, 48, 14);
panel.add(lblDay);
```

```
JLabel lblMonth = new JLabel("Month");
lblMonth.setBounds(164, 145, 48, 14);
panel.add(lblMonth);
```

```
JLabel lblYear = new JLabel("Year");
lblYear.setBounds(243, 145, 48, 14);
panel.add(lblYear);
```

```
@SuppressWarnings("rawtypes")
JComboBox dob = new JComboBox();
dob.setEditable(true);
dob.setBounds(92, 161, 48, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel dm = new DefaultComboBoxModel(date);
dob.setModel(dm);
panel.add(dob);
```

```
@SuppressWarnings("rawtypes")
JComboBox mob = new JComboBox();
mob.setEditable(true);
mob.setBounds(143, 161, 95, 22);
```



```
@SuppressWarnings("rawtypes")
DefaultComboBoxModel mo=new DefaultComboBoxModel(month);
mob.setModel(mo);
panel.add(mob);
```

```
@SuppressWarnings("rawtypes")
JComboBox yob = new JComboBox();
yob.setEditable(true);
yob.setBounds(243, 161, 72, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr = new DefaultComboBoxModel(Year);
yob.setModel(yr);
panel.add(yob);
```

```
JLabel lblAddress = new JLabel("Address:");
lblAddress.setBounds(10, 214, 60, 14);
panel.add(lblAddress);
```

```
JTextArea address = new JTextArea();
address.setBounds(92, 209, 211, 69);
panel.add(address);
```

```
JLabel lblPersonalInformation = new JLabel("Personal Information:");
lblPersonalInformation.setFont(new Font("Times New Roman",
Font.BOLD | Font.ITALIC, 14));
lblPersonalInformation.setBounds(21, 33, 142, 22);
frmResumebuilder.getContentPane().add(lblPersonalInformation);
```

```
JLabel lblAcademicQualification = new JLabel("Academic
Qualification:");
lblAcademicQualification.setFont(new Font("Times New Roman",
Font.BOLD | Font.ITALIC, 14));
lblAcademicQualification.setBounds(21, 380, 153, 22);
frmResumebuilder.getContentPane().add(lblAcademicQualification);
```

```
JPanel panel_1 = new JPanel();
panel_1.setBackground(new Color(135, 206, 250));
panel_1.setBorder(new LineBorder(new Color(255, 255, 0), 3));
panel_1.setBounds(10, 407, 345, 296);
frmResumebuilder.getContentPane().add(panel_1);
panel_1.setLayout(null);
```

```
JLayeredPane layeredPane_1 = new JLayeredPane();
layeredPane_1.setBounds(172, 5, 1, 1);
panel_1.add(layeredPane_1);
```

```
JLayeredPane layeredPane_2 = new JLayeredPane();
layeredPane_2.setBackground(new Color(135, 206, 250));
layeredPane_2.setBounds(10, 17, 325, 275);
panel_1.add(layeredPane_2);
```

```

JLabel lblGraduation = new JLabel("Graduation:");
lblGraduation.setFont(new Font("Tahoma", Font.BOLD |
Font.ITALIC, 14));

lblGraduation.setBounds(115, 11, 118, 22);
layeredPane_2.add(lblGraduation);

JLabel lblProgram = new JLabel("Program:");
lblProgram.setBounds(10, 35, 81, 22);
layeredPane_2.add(lblProgram);

@SuppressWarnings("rawtypes")
JComboBox graduation_pro = new JComboBox();
graduation_pro.setEditable(true);
graduation_pro.setBackground(new Color(255, 255, 255));
graduation_pro.setBounds(0, 60, 129, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel gp=new DefaultComboBoxModel(Uni);
graduation_pro.setModel(gp);
layeredPane_2.add(graduation_pro);

JLabel lblCgpa = new JLabel("Grade:");
lblCgpa.setBounds(138, 35, 48, 22);
layeredPane_2.add(lblCgpa);

@SuppressWarnings("rawtypes")
JComboBox cgpa = new JComboBox();
cgpa.setEditable(true);
cgpa.setBounds(139, 60, 60, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel Grade=new
DefaultComboBoxModel(grade);
cgpa.setModel(Grade);
layeredPane_2.add(cgpa);

JLabel lblPassingYear = new JLabel("Passing year:");
lblPassingYear.setBounds(220, 35, 94, 22);
layeredPane_2.add(lblPassingYear);

@SuppressWarnings("rawtypes")
JComboBox graduation_Year = new JComboBox();
graduation_Year.setEditable(true);
graduation_Year.setBounds(220, 60, 81, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr1 = new DefaultComboBoxModel(Year1);
graduation_Year.setModel(yr1);
layeredPane_2.add(graduation_Year);

JLabel lblIntermediate = new JLabel("12th Board:");
lblIntermediate.setFont(new Font("Tahoma", Font.BOLD |

```

```

Font.ITALIC, 14));

lblIntermediate.setBounds(115, 93, 118, 26);
layeredPane_2.add(lblIntermediate);

JLabel lblCourse = new JLabel("Course:");
lblCourse.setBounds(10, 119, 48, 22);
layeredPane_2.add(lblCourse);

JLabel lblGrade = new JLabel("Grade:");
lblGrade.setBounds(138, 119, 48, 22);
layeredPane_2.add(lblGrade);

JLabel lblPassingYear_1 = new JLabel("Passing Year:");
lblPassingYear_1.setBounds(220, 119, 81, 22);
layeredPane_2.add(lblPassingYear_1);

@SuppressWarnings("rawtypes")
JComboBox inter_program = new JComboBox();
inter_program.setEditable(true);
inter_program.setBounds(0, 142, 129, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel ip=new DefaultComboBoxModel(Inter);
inter_program.setModel(ip);
layeredPane_2.add(inter_program);

@SuppressWarnings("rawtypes")
JComboBox Inter_grade = new JComboBox();
Inter_grade.setEditable(true);
Inter_grade.setBounds(139, 142, 60, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel Grade1=new
DefaultComboBoxModel(grade);
Inter_grade.setModel(Grade1);
layeredPane_2.add(Inter_grade);

@SuppressWarnings("rawtypes")
JComboBox inter_year = new JComboBox();
inter_year.setEditable(true);
inter_year.setBounds(220, 142, 81, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr2 = new DefaultComboBoxModel(Year2);
inter_year.setModel(yr2);
layeredPane_2.add(inter_year);

JLabel lblMatriculation = new JLabel("10th Board:");
lblMatriculation.setFont(new Font("Tahoma", Font.BOLD |
Font.ITALIC, 14));

lblMatriculation.setBounds(115, 175, 118, 22);
layeredPane_2.add(lblMatriculation);

```

```

JLabel lblCourse_1 = new JLabel("Course:");
lblCourse_1.setBounds(10, 204, 48, 14);
layeredPane_2.add(lblCourse_1);

JLabel lblGrade_1 = new JLabel("Grade:");
lblGrade_1.setBounds(138, 204, 48, 14);
layeredPane_2.add(lblGrade_1);

JLabel lblPassingYear_2 = new JLabel("Passing Year:");
lblPassingYear_2.setBounds(220, 204, 81, 14);
layeredPane_2.add(lblPassingYear_2);

@SuppressWarnings("rawtypes")
JComboBox matric_course = new JComboBox();
matric_course.setEditable(true);
matric_course.setBounds(0, 229, 129, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel mp=new DefaultComboBoxModel(Matric);
matric_course.setModel(mp);
layeredPane_2.add(matric_course);

@SuppressWarnings("rawtypes")
JComboBox matric_Grade = new JComboBox();
matric_Grade.setEditable(true);
matric_Grade.setBounds(139, 229, 60, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel Grade2=new
DefaultComboBoxModel(grade);
matric_Grade.setModel(Grade2);
layeredPane_2.add(matric_Grade);

@SuppressWarnings("rawtypes")
JComboBox matric_year = new JComboBox();
matric_year.setEditable(true);
matric_year.setBounds(220, 229, 81, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr3 = new DefaultComboBoxModel(Year3);
matric_year.setModel(yr3);
layeredPane_2.add(matric_year);

JLayeredPane layeredPane_3 = new JLayeredPane();
layeredPane_3.setBorder(new LineBorder(new Color(255, 255, 0), 3));
layeredPane_3.setBounds(365, 58, 622, 311);
frmResumebuilder.getContentPane().add(layeredPane_3);

JPanel panel_2 = new JPanel();
panel_2.setBackground(new Color(135, 206, 250));
panel_2.setBounds(10, 11, 602, 289);
layeredPane_3.add(panel_2);
panel_2.setLayout(null);

```

```
JLabel lblCompany = new JLabel("Organization Name:");  
lblCompany.setBounds(35, 11, 124, 32);  
panel_2.add(lblCompany);
```

```
JLabel lblDesignation = new JLabel("Designation:");  
lblDesignation.setBounds(274, 11, 110, 32);  
panel_2.add(lblDesignation);
```

```
JLabel lblWorkExperience = new JLabel("Work Experience");  
lblWorkExperience.setBounds(462, 11, 119, 32);  
panel_2.add(lblWorkExperience);
```

```
JTextField comp1 = new JTextField();  
comp1.setBackground(new Color(255, 255, 255));  
comp1.setBounds(35, 41, 228, 23);  
panel_2.add(comp1);  
comp1.setColumns(10);
```

```
desig1 = new JTextField();  
desig1.setBackground(new Color(255, 255, 255));  
desig1.setBounds(274, 41, 177, 23);  
panel_2.add(desig1);  
desig1.setColumns(10);
```

```
comp2 = new JTextField();  
comp2.setColumns(10);  
comp2.setBounds(35, 93, 228, 24);  
panel_2.add(comp2);
```

```
desig2 = new JTextField();  
desig2.setColumns(10);  
desig2.setBounds(274, 93, 177, 24);  
panel_2.add(desig2);
```

```
comp3 = new JTextField();  
comp3.setColumns(10);  
comp3.setBounds(35, 153, 228, 23);  
panel_2.add(comp3);
```

```
desig3 = new JTextField();  
desig3.setColumns(10);  
desig3.setBounds(274, 153, 177, 23);  
panel_2.add(desig3);
```

```
comp4 = new JTextField();  
comp4.setColumns(10);  
comp4.setBounds(35, 213, 228, 21);  
panel_2.add(comp4);
```

```

desig4 = new JTextField();
desig4.setColumns(10);
desig4.setBounds(274, 213, 177, 21);
panel_2.add(desig4);

@SuppressWarnings("rawtypes")
JComboBox fromyear1 = new JComboBox();
fromyear1.setEditable(true);
fromyear1.setBackground(new Color(255, 255, 255));
fromyear1.setBounds(462, 41, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr4 = new DefaultComboBoxModel(Year4);
fromyear1.setModel(yr4);
panel_2.add(fromyear1);

@SuppressWarnings("rawtypes")
JComboBox toyear1 = new JComboBox();
toyear1.setEditable(true);
toyear1.setBackground(new Color(255, 255, 255));
toyear1.setBounds(525, 41, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr5 = new DefaultComboBoxModel(Year5);
toyear1.setModel(yr5);
panel_2.add(toyear1);

@SuppressWarnings("rawtypes")
JComboBox fromyear2 = new JComboBox();
fromyear2.setEditable(true);
fromyear2.setBounds(462, 95, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr6 = new DefaultComboBoxModel(Year6);
fromyear2.setModel(yr6);
panel_2.add(fromyear2);

@SuppressWarnings("rawtypes")
JComboBox toyear2 = new JComboBox();
toyear2.setEditable(true);
toyear2.setBounds(525, 95, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr7 = new DefaultComboBoxModel(Year7);
toyear2.setModel(yr7);
panel_2.add(toyear2);

@SuppressWarnings("rawtypes")
JComboBox fromyear3 = new JComboBox();
fromyear3.setEditable(true);
fromyear3.setBounds(462, 153, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr8 = new DefaultComboBoxModel(Year8);
fromyear3.setModel(yr8);

```

```

panel_2.add(fromyear3);

@SuppressWarnings("rawtypes")
JComboBox toyear3 = new JComboBox();
toyear3.setEditable(true);
toyear3.setBounds(525, 153, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr9 = new DefaultComboBoxModel(Year9);
toyear3.setModel(yr9);
panel_2.add(toyear3);

@SuppressWarnings("rawtypes")
JComboBox fromyear4 = new JComboBox();
fromyear4.setEditable(true);
fromyear4.setBounds(462, 213, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr10 = new
DefaultComboBoxModel(Year10);
fromyear4.setModel(yr10);
panel_2.add(fromyear4);

@SuppressWarnings("rawtypes")
JComboBox toyear4 = new JComboBox();
toyear4.setEditable(true);
toyear4.setBounds(525, 213, 67, 22);
@SuppressWarnings("rawtypes")
DefaultComboBoxModel yr11 = new
DefaultComboBoxModel(Year11);
toyear4.setModel(yr11);
panel_2.add(toyear4);

JLabel lblExperience = new JLabel("Experience");
lblExperience.setFont(new Font("Times New Roman", Font.BOLD |
Font.ITALIC, 14));
lblExperience.setBounds(382, 33, 78, 22);
frmResumebuilder.getContentPane().add(lblExperience);

JLayeredPane photo = new JLayeredPane();
photo.setBackground(new Color(135, 206, 250));
photo.setBorder(new LineBorder(new Color(135, 206, 250)));
photo.setBounds(1034, 81, 222, 205);
frmResumebuilder.getContentPane().add(photo);

JLabel label = new JLabel("");
label.setHorizontalAlignment(SwingConstants.CENTER);
label.setBounds(-30, -21, 268, 226);
photo.add(label);

JLabel lblPicture = new JLabel("Photo");

```

```

14));

lblPicture.setFont(new Font("Tahoma", Font.BOLD | Font.ITALIC,

lblPicture.setBounds(1120, 37, 87, 29);
frmResumebuilder.getContentPane().add(lblPicture);
progressBar.setForeground(new Color(50, 205, 50));

progressBar.setBackground(new Color(135, 206, 250));
progressBar.setBounds(10, 714, 1246, 14);
frmResumebuilder.getContentPane().add(progressBar);

JButton btnAttachPhoto = new JButton("Attach Photo");
btnAttachPhoto.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        @SuppressWarnings("unused")
        Sample1 window = new Sample1();
        progressBar.setValue(100);
        JFileChooser chooser = new JFileChooser();
        chooser.showOpenDialog(null);
        File f=chooser.getSelectedFile();
        filename=f.getAbsolutePath();
        url.setText(filename);
        @SuppressWarnings("unused")
        Image getabsolutePath=null;
        ImageIcon icon = new ImageIcon(filename);
        @SuppressWarnings("unused")
        Image image
=icon.getImage().getScaledInstance(label.getWidth(),
label.getHeight(),Image.SCALE_SMOOTH);
        label.setIcon(icon);

    }

});
btnAttachPhoto.setFont(new Font("Tahoma", Font.BOLD, 11));
btnAttachPhoto.setBackground(new Color(119, 136, 153));
btnAttachPhoto.setBounds(1102, 320, 117, 29);
frmResumebuilder.getContentPane().add(btnAttachPhoto);

JLayeredPane layeredPane_5 = new JLayeredPane();
layeredPane_5.setBorder(new LineBorder(new Color(255, 255, 0), 3));
layeredPane_5.setBounds(365, 407, 622, 296);
frmResumebuilder.getContentPane().add(layeredPane_5);

JLayeredPane layeredPane_6 = new JLayeredPane();
layeredPane_6.setBackground(new Color(135, 206, 250));
layeredPane_6.setBounds(10, 11, 602, 274);
layeredPane_5.add(layeredPane_6);

JLabel lblComputer = new JLabel("Computer");

```



```

12));

lblComputer.setFont(new Font("Tahoma", Font.BOLD | Font.ITALIC,

lblComputer.setBounds(45, 11, 80, 14);
layeredPane_6.add(lblComputer);

comskill1 = new JTextField();
comskill1.setBounds(45, 36, 244, 20);
layeredPane_6.add(comskill1);
comskill1.setColumns(10);

comskill2 = new JTextField();
comskill2.setColumns(10);
comskill2.setBounds(299, 36, 259, 20);
layeredPane_6.add(comskill2);

comskill3 = new JTextField();
comskill3.setColumns(10);
comskill3.setBounds(45, 67, 244, 20);
layeredPane_6.add(comskill3);

comskill4 = new JTextField();
comskill4.setColumns(10);
comskill4.setBounds(299, 67, 259, 20);
layeredPane_6.add(comskill4);

JLabel lblLanguages = new JLabel("Languages");
lblLanguages.setFont(new Font("Tahoma", Font.BOLD |
Font.ITALIC, 12));
lblLanguages.setBounds(45, 98, 86, 14);
layeredPane_6.add(lblLanguages);

lang3 = new JTextField();
lang3.setColumns(10);
lang3.setBounds(45, 153, 244, 20);
layeredPane_6.add(lang3);

lang4 = new JTextField();
lang4.setColumns(10);
lang4.setBounds(299, 153, 259, 20);
layeredPane_6.add(lang4);

lang2 = new JTextField();
lang2.setColumns(10);
lang2.setBounds(299, 122, 259, 20);
layeredPane_6.add(lang2);

lang1 = new JTextField();
lang1.setColumns(10);
lang1.setBounds(45, 122, 244, 20);
layeredPane_6.add(lang1);

```

```

12));

JLabel lblSports = new JLabel("Sports");
lblSports.setFont(new Font("Tahoma", Font.BOLD | Font.ITALIC,

sports1.setBounds(45, 193, 48, 14);
layeredPane_6.add(lblSports);

sports1 = new JTextField();
sports1.setColumns(10);
sports1.setBounds(45, 218, 244, 20);
layeredPane_6.add(sports1);

sports2 = new JTextField();
sports2.setColumns(10);
sports2.setBounds(299, 218, 259, 20);
layeredPane_6.add(sports2);

JLabel lblSkills = new JLabel("Skills:");
lblSkills.setFont(new Font("Tahoma", Font.BOLD | Font.ITALIC,
14));

lblSkills.setBounds(382, 381, 60, 19);
frmResumebuilder.getContentPane().add(lblSkills);

JButton btnGo = new JButton("GO >>>");
btnGo.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        @SuppressWarnings("rawtypes")
        Print p= new Print();
        Sample1 window = new Sample1();
        window.name.setText(fullname.getText());
        window.father.setText(fathename.getText());
        window.phone.setText(phonenum.getText());
        window.email1.setText(email.getText());
        window.homeaddress.setText(address.getText());
        window.gpro.setText((String)
graduation_pro.getSelectedItemAt());
        window.ipro.setText((String)
inter_program.getSelectedItemAt());
        window.mpro.setText((String)
matric_course.getSelectedItemAt());
        window.ggrade.setText((String)
cgpa.getSelectedItemAt());
        window.mgrade.setText((String)
matric_Grade.getSelectedItemAt());
        window.igrade.setText((String)
Inter_grade.getSelectedItemAt());
        window.gy.setText((String)
graduation_Year.getSelectedItemAt());
        window.iy.setText((String)

```

```

inter_year.getSelectedItemAt());
matric_year.getSelectedItemAt());

window.my.setText((String)
window.C1.setText(comp1.getText());
window.C2.setText(comp2.getText());
window.C3.setText(comp3.getText());
window.C4.setText(comp4.getText());
window.D1.setText(desig1.getText());
window.D2.setText(desig2.getText());
window.D3.setText(desig3.getText());
window.D4.setText(desig4.getText());
@SuppressWarnings("unused")
String a=(String) toyear1.getSelectedItemAt();
window.Y1.setText(p.year((String)
toyear1.getSelectedItemAt(),(String) fromyear1.getSelectedItemAt()+" Years");
window.Y2.setText(p.year((String)
toyear2.getSelectedItemAt(),(String) fromyear2.getSelectedItemAt()+" Years");
window.Y3.setText(p.year((String)
toyear3.getSelectedItemAt(),(String) fromyear3.getSelectedItemAt()+" Years");
window.Y4.setText(p.year((String)
toyear4.getSelectedItemAt(),(String) fromyear4.getSelectedItemAt()+" Years");
window.s1.setText(comskill1.getText());
window.s2.setText(comskill2.getText());
window.s3.setText(comskill3.getText());
window.s4.setText(comskill4.getText());
window.l1.setText(lang1.getText());
window.l2.setText(lang2.getText());
window.l3.setText(lang3.getText());
window.l4.setText(lang4.getText());
window.S1.setText(sports1.getText());
window.S2.setText(sports2.getText());
window.my.setText((String)

matric_year.getSelectedItemAt());

window.Dob.setText((String) dob.getSelectedItemAt()+" -
"+(String) mob.getSelectedItemAt()+" - "+(String) yob.getSelectedItemAt());

ImageIcon icon1 = new ImageIcon(filename);
@SuppressWarnings("unused")
Image image1
=icon1.getImage().getScaledInstance(window.finalphoto.getWidth(),
window.finalphoto.getHeight(),Image.SCALE_SMOOTH);
window.finalphoto.setIcon(icon1);
window.frmSample.setVisible(true);

}
});
btnGo.setBounds(1118, 579, 89, 23);

```

```

frmResumebuilder.getContentPane().add(btnGo);

JLabel lblClickHereFor = new JLabel("Click Here For Samples");
lblClickHereFor.setFont(new Font("Tahoma", Font.BOLD, 14));
lblClickHereFor.setForeground(new Color(47, 79, 79));
lblClickHereFor.setBounds(1086, 539, 170, 29);
frmResumebuilder.getContentPane().add(lblClickHereFor);

JRadioButton rdbtnNewRadioButton = new JRadioButton("");
rdbtnNewRadioButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        if(rdbtnNewRadioButton.isSelected()) {
            a=20;
            progressBar.setValue(a);
        }else {
            progressBar.setValue(0);
        }
    }
});
rdbtnNewRadioButton.setBackground(new Color(135, 206, 250));
rdbtnNewRadioButton.setBounds(169, 37, 22, 18);
frmResumebuilder.getContentPane().add(rdbtnNewRadioButton);

JRadioButton radioButton = new JRadioButton("");
radioButton.setBounds(170, 380, 21, 23);
frmResumebuilder.getContentPane().add(radioButton);
radioButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        if (radioButton.isSelected()){
            b=40;
            progressBar.setValue(b);
        }else {
            progressBar.setValue(a);
        }
    }
});
radioButton.setBackground(new Color(135, 206, 250));

JRadioButton radioButton_1 = new JRadioButton("");
radioButton_1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        if(radioButton.isSelected()) {
            c=60;
            progressBar.setValue(c);
        }else {
            progressBar.setValue(40);
        }
    }
});

```

```

radioButton_1.setBackground(new Color(135, 206, 250));
radioButton_1.setBounds(463, 33, 27, 22);
frmResumebuilder.getContentPane().add(radioButton_1);

```

```

JRadioButton radioButton_3 = new JRadioButton("");
radioButton_3.setBounds(432, 380, 21, 23);
frmResumebuilder.getContentPane().add(radioButton_3);
radioButton_3.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {
        if(radioButton_3.isSelected()) {
            d=80;
            progressBar.setValue(d);
        }
        else {
            progressBar.setValue(c);
        }
    }
});
radioButton_3.setBackground(new Color(135, 206, 250));

```

```

url = new JTextField();
url.setEnabled(false);
url.setEditable(false);
url.setForeground(new Color(135, 206, 250));
url.setBackground(new Color(135, 206, 250));
url.setBounds(1044, 297, 224, 20);
frmResumebuilder.getContentPane().add(url);
url.setColumns(10);

```

```

JButton btnTestEntry = new JButton("Test Entry");
btnTestEntry.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {

```

```

        fullname.setText("Rahul Gupta");
        fathername.setText("Danveer Gupta");
        address.setText("Gala no.13, sahara apt, valaipada road,
santosh bhuvan, nallasopara east");
        phonenum.setText("+91-8796138807");
        comp1.setText("TrackpointGPS PVT LTD");
        desig1.setText("Junior Software Developer");
        comp2.setText("TrackpointGPS PVT LTD");
        desig2.setText("Tech Support Executive");
        comp3.setText("Alpha Solutions");
        desig3.setText("Assistant Manager HR");
        comp4.setText("Iqra University");
        desig4.setText("Data Entry Operator");
        comskill1.setText("Microsoft Office");
        comskill2.setText("SAP MM Module");
        comskill3.setText("Internet Surfing");
    }
}

```

```

        comskill4.setText("Basic Programming Languages");
        email.setText("coolrahulb1995@gmail.com");
        lang1.setText("English");
        lang2.setText("Urdu");
        lang3.setText("Punjabi");
        lang4.setText("Saraiki");
        sports1.setText("Cricket");
        sports2.setText("Volley Ball");
        progressBar.setValue(80);

        progressBar.setValue(100);

    }

});
btnTestEntry.setForeground(new Color(255, 255, 0));
btnTestEntry.setBackground(new Color(0, 0, 0));
btnTestEntry.setBounds(878, 11, 111, 36);
frmResumebuilder.getContentPane().add(btnTestEntry);

JButton clear = new JButton("Clear");
clear.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        fullname.setText("");
        fathername.setText("");
        address.setText("");
        phonenum.setText("");
        comp1.setText("");
        desig1.setText("");
        comp2.setText("");
        desig2.setText("");
        comp3.setText("");
        desig3.setText("");
        comp4.setText("");
        desig4.setText("");
        comskill1.setText("");
        comskill2.setText("");
        comskill3.setText("");
        comskill4.setText("");
        email.setText("");
        lang1.setText("");
        lang2.setText("");
        lang3.setText("");
        lang4.setText("");
        sports1.setText("");
        sports2.setText("");
        progressBar.setValue(0);
        label.setIcon(null);
    }
});
clear.setForeground(Color.YELLOW);

```

```

        clear.setBackground(Color.BLACK);
        clear.setBounds(762, 11, 111, 36);
        frmResumebuilder.getContentPane().add(clear);

    }

    /**
     * @return the fullname
     */
    public JTextField getFullname() {
        return fullname;
    }

    /**
     * @param fullname the fullname to set
     */
    public void setFullname(JTextField fullname) {
        this.fullname = fullname;
        fullname.setHorizontalAlignment(SwingConstants.LEFT);
    }
}

```

Print.java:

```

package CV;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.print.PageFormat;
import java.awt.print.Printable;
import java.awt.print.PrinterException;
import java.awt.print.PrinterJob;

import javax.swing.JPanel;

public class Print<ContentPne>{

    public void print(JPanel a) {
        PrinterJob job = PrinterJob.getPrinterJob();
        job.setPrintable(new Printable() {

            public int print(Graphics pg, PageFormat pf, int
pageNum)throws PrinterException {
                pf.setOrientation(PageFormat.PORTRAIT);
                if(pageNum>0) {
                    return Printable.NO_SUCH_PAGE;

```

```

        }

        Graphics2D g2 =(Graphics2D)pg;
        g2.translate(pf.getImageableX()*2,

pf.getImageableY()*2);

        g2.scale(0.86, 1);;
        a.print(g2);

        return Printable.PAGE_EXISTS;
    }
});
boolean ok = job.printDialog();
if(ok) {
    try {
        job.print();
    }
    catch(PrinterException ex) {}
}

}

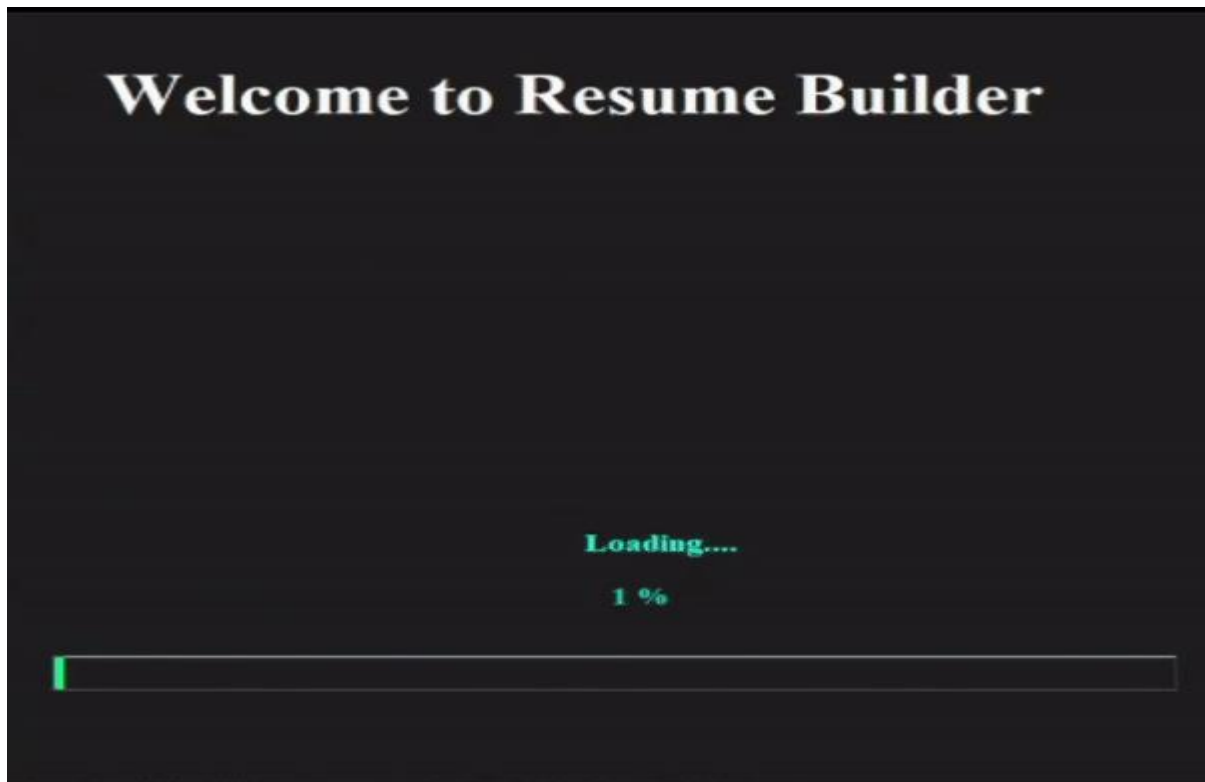
public String year(String a,String b) {
    int x=Integer.parseInt(a);
    int y=Integer.parseInt(b);
    int z= x-y;

    return String.valueOf(z);
}
}

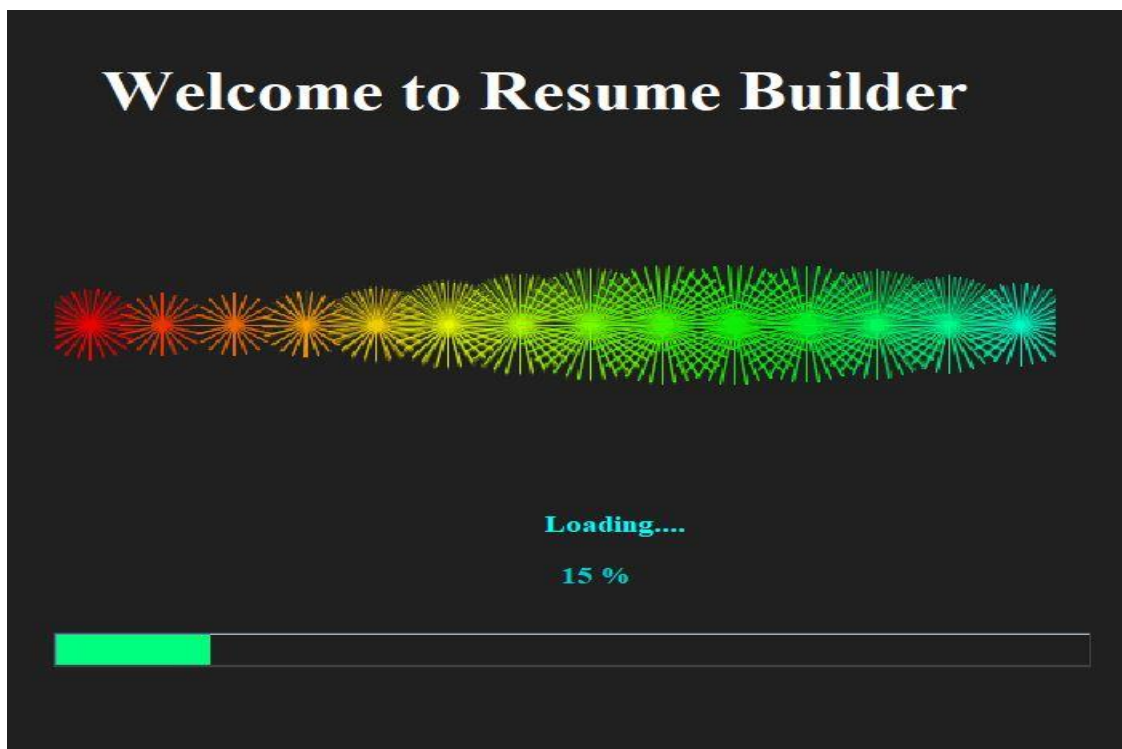
```


4.2 INPUT/OUTPUT SCREENS

- Welcome Page gif



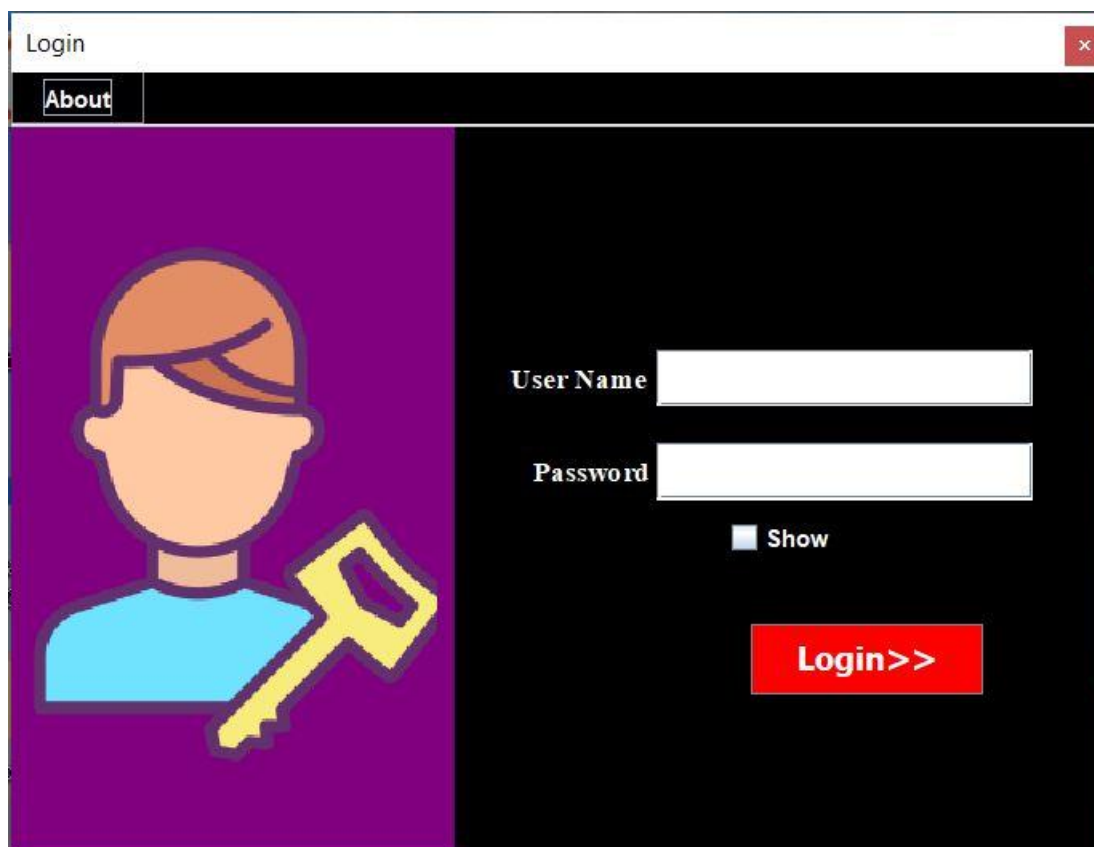
- Welcome page:



- **About Page**

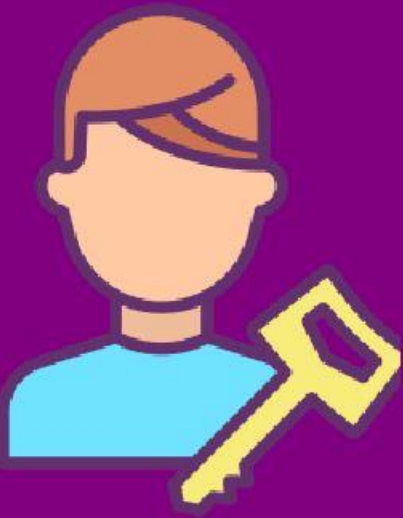


- **Login Page**



Login

About



User Name

rahul

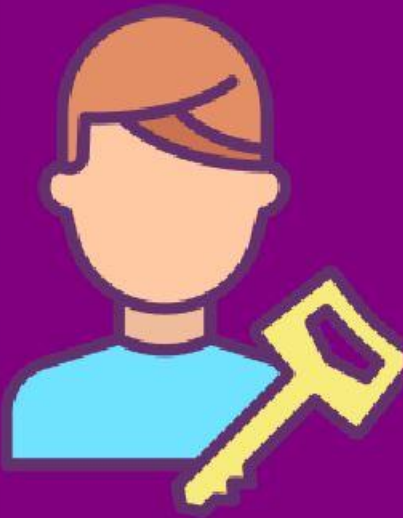
Password

☐ Show

Login>>

Login

About



User Name

rahul

Password

rahul

☒ Show

Login>>

- Resume Information Page

Resume_Builder

Personal Information: Experience Clear Test Entry

Full Name:

Father Name:

Phone#:

E-Mail:

D.O.B: 01 January 1970

Address:

Organization Name: Designation: Work Experience: 1970 1970

1970 1970

1970 1970

1970 1970

Photo

Attach Photo

Academic Qualification: Skills

Graduation:

Program: Grade: Passing year:

MCA A+ 1970

12th Board:

Course: Grade: Passing Year:

HSC A+ 1970

10th Board:

Course: Grade: Passing Year:

SSC A+ 1970

Computer

Languages

Sports

Click Here For Samples

GO >>>

- After Fill the Information

Resume_Builder

Personal Information: Experience Clear Test Entry

Full Name: Rahul Gupta

Father Name: Danveer Gupta

Phone#: +91-8796138807

E-Mail: coolrahul1995@gmail.com

D.O.B: 01 January 1970

Address: Gala no 13, sahara apt, valapada road, t

Organization Name: TrackpointGPS PVT LTD Designation: Junior Software Developer Work Experience: 1970 1970

TrackpointGPS PVT LTD Tech Support Executive 1970 1970

Alpha Solutions Assistant Manager HR 1970 1970

Igra University Data Entry Operator 1970 1970

Photo

Attach Photo

Academic Qualification: Skills

Graduation:

Program: Grade: Passing year:

MCA A+ 1970

12th Board:

Course: Grade: Passing Year:

HSC A+ 1970

10th Board:

Course: Grade: Passing Year:

SSC A+ 1970

Computer

Microsoft Office SAP MM Module

Internet Surfing Basic Programming Languages

Languages

English Urdu

Punjabi Sariki

Sports

Cricket Volley Ball

Click Here For Samples

GO >>>

- **Resume Generated**


Sample 1


Print


Rahul Gupta


S.O: Danveer Gupta

D.O.B 01 - January - 1970

 +91-8796138807

 coolrahulb1995@gmail.com

 , sahara apt, valaipada road, santosh bhuvan, nallasopara east



OBJECTIVES:

Organized and motivated employee able to apply time management and organizational skills in various environments. Seeking to join Acme Corp as an administrative assistant to help ensure good internal communications and budget management.

ACADEMIC QUALIFICATION:

	Graduation:	Intermediate:	Matriculation:
Program:	MCA	HSC	SSC
Grade:	A+	A+	A+
Passing Year:	1970	1970	1970

EXPERIENCE:

TrackpointGPS PVT LTD	Senior Software Developer	0 Years
TrackpointGPS PVT LTD	Tech Support Executive	0 Years
Alpha Solutions	Assistant Manager HR	0 Years
Iqra University	Data Entry Operator	0 Years

LANGUAGES:

English	Urdu
Punjabi	Saraiki

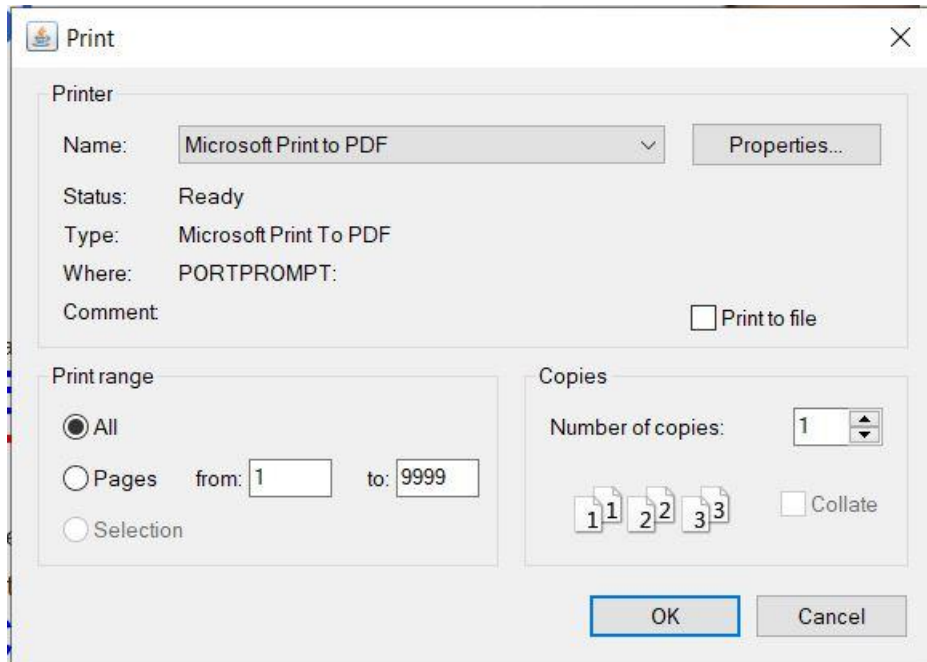
SKILLS:

- Microsoft Office
- SAP MM Module
- Internet Surfing
- Basic Programming Languages

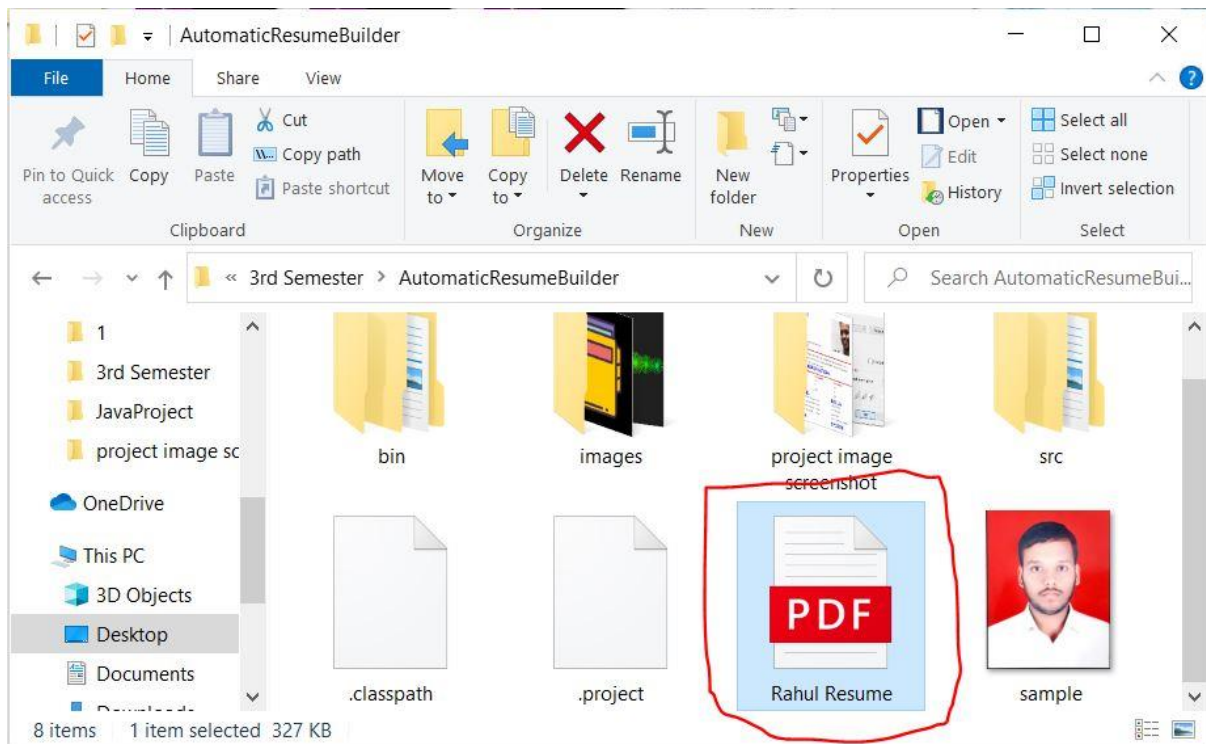
SPORTS:

- Cricket
- Volley Ball

- **Print Resume**



- **Save in the file**



5. TESTING & VALIDATION

5.1 INTRODUCTION

A Test case is a set of actions executed to verify a particular feature or functionality of your software application. A Test Case contains test steps, test data, precondition, post condition developed for specific test scenario to verify any requirement. Validation testing is the process of ensuring if the tested and developed software satisfies the client user needs. ... As a tester, you need to evaluate if the test execution results comply with that mentioned in the requirements document.

In software engineering, a test case is a specification of the inputs, execution conditions, testing procedure, and expected results that define a single test to be executed to achieve a particular software testing objective, such as to exercise a particular program path or to verify compliance with a specific requirement. Test cases underlie testing that is methodical rather than haphazard. A battery of test cases can be built to produce the desired coverage of the software being tested.

5.2 DESIGN OF TEST CASES AND SCENARIO

The sample of data used to provide an unbiased evaluation of a final model fit on the training dataset

Test ID	Test Description	Test Steps	Test Data	Expected Result	Actual Result	Pass
T01	Status verify login	Enter id & password	User data must be present in the code	User must be able to login to this account	User logged into the app	Pass
T02	Add Information to resume	Enter all details	Data must be saved	Data uploaded and save	Resume data save successfully	Pass
T03	Already data filled	Test entry	Data must be saved	Data uploaded and save	Resume data save successfully	Pass
T04	Print & download resume	Details of resume to print and download	User data must be added	Print and download resume	Must be print and download	Pass

5.3 VALIDATION

Validation is the process of evaluating the final product to check whether the software meets the needs. Validation for our Application is done through functional testing.

VALIDATION Whenever we go through web sites, we need to register ourselves to become members of the websites. Forms consist of various fields that need to be filled in with the candidates, to get all to the user's details. Sometimes, by mistake, we provide the information in the wrong format. Also, when we provide incomplete information that should be indicated is incorrect, so that the user should be able to modify, or correct the information filled in by the candidates for proper registration. So, by validating the form, it indicates to the user that the information filled in is incorrect (in case of wrong or incorrect information) and makes it correct.

6. CONCLUSION

6.1 PROJECT CONCLUSION

The online resume builder is one of the most fantastic systems for the people who are either freshers in their domain or if they don't have enough idea about the resume or don't have enough time to create the resume of good designs or patterns, then this platform is a very productive place for them. It saves a lot of time and cost-effective.

The project Online Resume Builder is for computerizing the working of building resumes. The software takes care of all the requirements of the process and is capable to provide easy and effective storage of information related to users and resumes that come up to the system. It generates reports for users & administrators. Provides easy designing tools and other interesting features. This system promises very less or no paper work and also provides help to customers and viewers. In this system everything is stored electronically so very less amount of paper work is required and information can be retrieved very easily without searching here and there into registers.

6.2 FUTURE ENHANCEMENT

In future we would like to extend the Application with:

- Make signup and login page for every user and save their data according to their login
- Make more format of resume
- Working with new fonts and design.
- Adding more design in the future.
- Doing changes in the application of fonts and colors.