

RELIFE JOB PORTAL

RELIFE - A JOB PORTAL

A MINI-PROJECT REPORT

Submitted by:

Roshan A 240701438

Sharweshvar P 240701491

in partial fulfillment of the award of the degree

Of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING



RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

An Autonomous Institute

CHENNAI

NOVEMBER 2025

BONAFIDE CERTIFICATE

BONAFIDE CERTIFICATE

Certified that this project “**Relife Job Portal**” is the Bonafide work of “**ROSHAN A ,SHARWESVAR P**” who carried out the project work under my supervision.

SIGNATURE

MS. S. SATHIYAVATHI

ASSISTANT PROFESSOR

Dept. of Computer Science and Eng,

Rajalakshmi Engineering College

Chennai

This mini project report is submitted for the viva voce examination to be held on _____

INTERNAL EXAMINER

EXTERNAL EXAMINER

ABSTRACT

ABSTRACT

This project presents the design and development of the ReLife Job Portal System, an interactive platform built using Java Swing for the graphical user interface and MySQL for persistent data storage via Java Database Connectivity (JDBC). The system facilitates seamless communication between job seekers and employers by providing essential functionalities such as user registration, job search, application management, and profile updates.

A key architectural feature is the use of a Data Access Layer (DAO), centralized in the **DatabaseHelper** class, which handles all database operations—ensuring data consistency, modularity, and secure access. The platform also incorporates an Admin Dashboard, allowing administrators to manage users, view qualifications, and monitor job activities, including saved and applied jobs.

Through this implementation, the ReLife Job Portal demonstrates efficient user-data management, responsive UI design, and real-time database integration, emphasizing scalability, reliability, and ease of maintenance. The system exemplifies a practical and scalable employment management solution that bridges the gap between job seekers and employers within a digital ecosystem.

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

We express our sincere thanks to our beloved and honorable chairman **MR. S. MEGANATHAN** and the chairperson **DR. M. THANGAM MEGANATHAN** for their timely support and encouragement.

We are greatly indebted to our respected and honorable principal **Dr. S.N. MURUGESAN** for his able support and guidance.

No words of gratitude will suffice for the unquestioning support extended to us by our Head of The Department **Dr. E.M. MALATHY** and our Deputy Head of The Department **Dr. J. MANORANJINI** for being an ever supporting force during our project work.

We also extend our sincere and hearty thanks to our internal guide **Ms. S. SATHIYAVATHI**, for her valuable guidance and motivation during the completion of this project.

Our sincere thanks to our family members, friends and other staff members of computer science engineering.

1. ROSHAN A

2. SHARWESVAR P

TABLE OF CONTENTS

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	III
1 INTRODUCTION		7
1.1 INTRODUCTION		
1.2 SCOPE OF THE WORK		
1.3 PROBLEM STATEMENT		
1.4 AIM AND OBJECTIVES OF THE PROJECT		
2 SYSTEM SPECIFICATIONS		9
2.1 HARDWARE SPECIFICATIONS		
2.2 SOFTWARE SPECIFICATIONS		
3 MODULE DESCRIPTION		10
4 CODING		12
5 SCREENSHOTS		14
6 CONCLUSION AND FUTURE ENHANCEMENT		18
7 REFERENCES		19

LIST OF FIGURES

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
5.1	LOGIN PAGE	14
5.2	REGISTRATION PAGE	14
5.3	DASHBOARD PANEL	15
5.4	JOB SEARCH PANEL	15
5.5	ADMIN LOGIN	16
5.6	ADMIN FRAME	17

INTRODUCTION

INTRODUCTION

1.1 INTRODUCTION

In the modern era of online career building, having a centralized platform to manage and explore employment opportunities is essential. The **ReLife Job Portal** is a desktop-based application developed using **Java Swing** and **MySQL** to simplify job search. The system allows **users** to register, log in, browse available jobs, apply for positions, and save favorite listings. It also includes a separate **admin login module**, enabling administrators to monitor user activity and update records. Through its clean and intuitive interface, ReLife ensures efficient interaction between job seekers and system administrators while maintaining reliable database connectivity through **JDBC**.

1.2 SCOPE OF THE WORK

The ReLife system is designed to provide both job seekers and administrators with essential features that streamline career management and application processing. **User Interface:** A modern, GUI-driven experience designed in Java Swing for desktop systems.

The scope includes:

- **User Features:** Registration, login, job search, jobs saved tracking, and profile updates.
- **Admin Features:** Job posting, saved jobs monitoring, user activity tracking, and admin login security.
- **Data Management:** Secure data storage using **MySQL** and automatic table creation.

PROBLEM STATEMENT

1.3 PROBLEM STATEMENT

Job seekers often face fragmented systems with poor data handling and limited interaction. Many existing job portals rely on web servers, which can be overkill for small-scale educational or enterprise use. Hence, the problem addressed by this project is: "To design and implement a desktop-based job portal system that efficiently manages users, job postings, and application data using a single local database, ensuring data consistency, simplicity, and real-time interaction."

Challenges include:

- Managing user profiles and saved jobs persistently.
- Enabling secure login and role-based access (user/admin).
- Maintaining an intuitive interface without complex dependencies.

1.4 AIM AND OBJECTIVES OF THE PROJECT

Aim: To design and develop a reliable, database-driven desktop **Career Management and Job Portal** application using Java Swing and MySQL for user and admin interaction.

Objectives:

- To implement a **User** and **Admin** authentication system with MySQL-backed verification. To develop modules for job search, saved jobs, and job applications.
- To integrate **profile management**, enabling users to view and edit their details.
- To provide **admin tools** for monitoring and managing user and job data.

CHAPTER 2 SYSTEM SPECIFICATIONS 2

CHAPTER 2

SYSTEM SPECIFICATIONS

2.1 HARDWARE SPECIFICATIONS

Component	Detailed Specification
Processor	Intel(R) Core(TM) i3-1005G1 CPU @ 1.20GHz(1.20 GHz)
RAM	8.00 GB (7.69 GB usable)
Storage	100 MB free disk space
Network	Stable Internet or LAN connection

2.2 SOFTWARE SPECIFICATIONS (Updated for MySQL)

Component	Detailed Specification
Operating System	Windows 11
Programming Language	Java SE Development Kit (JDK) 17+
GUI	java. swing MySQL Server (v8.0 or later)
Connector	MySQL Connector/J (JDBC Driver)
Database Utility	MySQL Workbench

MODULE DESCRIPTIPN

CHAPTER 3

MODULE DESCRIPTION

The application is structured into four primary modules: **Data Model**, **Data Access Layer (DAO)**, **Database Utility**, and **User Interface (UI)**.

3.1 DATA MODEL MODULE

User.java: Acts as the blueprint for every user registered in the ReLife Job Portal. It encapsulates data fields (`userId`, `name`, `email`, `phone`, `gender`, `role`, `createdAt`) and includes constructors, getters, and setters for data manipulation.

Job.java: Represents job listings available in the system. It includes fields such as (`jobId`, `title`, `company`, `location`, `salary`, `type`, `description`) and provides accessor and mutator methods for dynamic job display.

3.2 DATA ACCESS LAYER (DAO) MODULE

This module contains the core business logic for database interaction.

DatabaseHelper.java: Manages all interactions between the application and the MySQL database. It handles CRUD operations for all modules such as users, qualifications, and job applications.

Methods Implemented:

- `loginUser(String email, String password)` – Verifies login credentials of users.

- `registerUser(String name, String email, String phone, String password)` – Registers a new user into the system.
- `getAllUsers()` – Retrieves all registered users for the admin dashboard.
- `getAllUserQualifications()` – Fetches qualification details of all users.
- `getAllSavedJobs()` – Returns all jobs saved by users.
- `getAllAppliedJobs()` – Displays job application records of users.
- `updateUserProfile(User user)` – Updates user profile information.
- `deleteUser(int userId)` – Deletes user accounts when needed.

The DAO layer ensures secure and optimized SQL operations using `PreparedStatement` and follows clean separation of concerns.

MODULE

3.3 DATABASE UTILITY MODULE

This module manages the database connection and table setup.

Functions Implemented:

- Establishes a connection to the **relife_db** MySQL database.
- Automatically creates required tables (**users**, **user_qualifications**, **saved_jobs**, **job_applications**, **admins**) if not already present.
- Handles connection pooling and exception management.
- Ensures smooth and reliable data flow between the front-end and the database.

3.4 USER INTERFACE (UI) MODULE

This module is responsible for all graphical user interfaces in the system, built using **Java Swing**.

Login and Registration Screens: Handle user and admin authentication with input validation and database verification.

Dashboard Panel: Displays all available job listings, user activity, applied jobs, and saved jobs with filtering options like job type, experience, salary, and location.

Admin Dashboard: Provides administrative control to view all users, qualifications, saved jobs, and applied jobs. Additional features include search functionality and account management tools.

DBCONNECTION CODE

CHAPTER 4

SAMPLE CODING

DB CONNECTION CODE:

```
1 package com.relife.db;
2 RelifeApp/src/com/relife/ui/RelifeApp.java
3 import java.sql.*;
4 import java.util.*;
5
6 import com.relife.model.User;
7
8
9 public class DatabaseHelper {
10     private static final String URL = "jdbc:mysql://localhost:3306/relife_db";
11     private static final String USER = "root";
12     private static final String PASSWORD = "root17";
13
14     static {
15         try {
16             Class.forName("com.mysql.cj.jdbc.Driver");
17             System.out.println("MySQL Driver loaded successfully!");
18         } catch (ClassNotFoundException e) {
19             System.out.println("MySQL Driver not found: " + e.getMessage());
20         }
21     }
22
23     public static Connection getConnection() throws SQLException {
24         return DriverManager.getConnection(URL, USER, PASSWORD);
25     }
26
27     public static void initializeDatabase() {
28         try (Connection conn = getConnection();
29              Statement stmt = conn.createStatement()) {
30
31             // Create users table with all needed columns
32             stmt.executeUpdate("CREATE TABLE IF NOT EXISTS users (");
33             stmt.executeUpdate("    id INT AUTO_INCREMENT PRIMARY KEY," +
34             "    name VARCHAR(255) NOT NULL," +
35             "    email VARCHAR(255) NOT NULL," +
36             "    password VARCHAR(255) NOT NULL," +
37             "    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP," +
38             "    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP);");
39         }
40     }
41 }
```

DASHBOARD PANEL CODE:

DASHBOARD PANEL CODE:

```
-- 93
94  public DashboardPanel(ReLifeApp app) {
95      this.app = app;
96      System.out.println("DEBUG DASHBOARD: Current user = " + app.getCurrentUser());
97      setBackground(new Color(38, 0, 77));
98      setLayout(new BorderLayout(0, 0));
99      add(createHeaderPanel(), BorderLayout.NORTH);
100
101     JPanel mainContent = new JPanel(new BorderLayout(15, 0));
102     mainContent.setBackground(new Color(38, 0, 77));
103     mainContent.setBorder(new EmptyBorder(20, 20, 20, 20));
104     mainContent.add(createSidebar(), BorderLayout.WEST);
105
106     contentLayout = new CardLayout();
107     contentPanel = new JPanel(contentLayout);
108     contentPanel.setBackground(new Color(245, 245, 245));
109
110     contentPanel.add(createHomePanel(), "HOME");
111     contentPanel.add(createAddDonationPanel(), "RENEWBOX");
112     contentPanel.add(createFindResourcesPanel(), "RESOURCES");
113     contentPanel.add(createSearchJobsPanel(), "JOBS");
114     contentPanel.add(createSavedJobsPanel(), "SAVED_JOBS");
115     contentPanel.add(createMentorshipPanel(), "MENTORSHIP");
116     contentPanel.add(createProfilePanel(), "PROFILE");
117
118     JScrollPane scrollPane = new JScrollPane(contentPanel);
119     scrollPane.setBackground(new Color(245, 245, 245));
120     scrollPane.getVerticalScrollBar().setUnitIncrement(16);
121     scrollPane.setHorizontalScrollBarPolicy(JScrollPane.HORIZONTAL_SCROLLBAR_NEVER);
122     mainContent.add(scrollPane, BorderLayout.CENTER);
123     add(mainContent, BorderLayout.CENTER);
```

LOGIN PAGE

CHAPTER 5

SCREEN SHOTS

Fig 5.1 LOGIN PAGE:

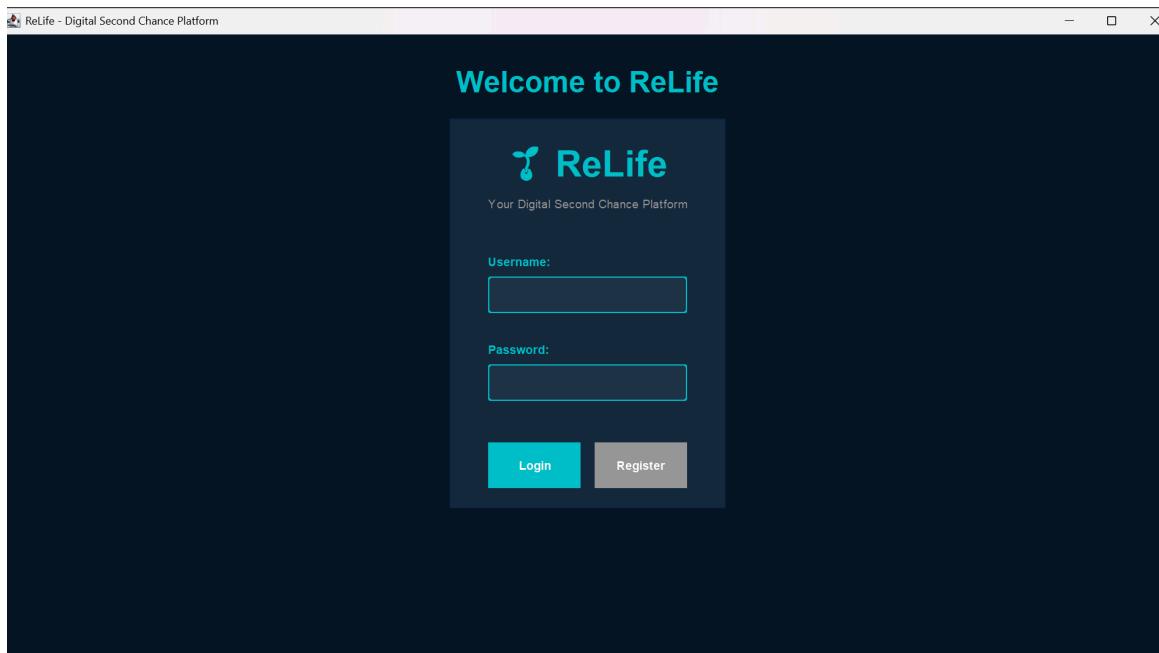
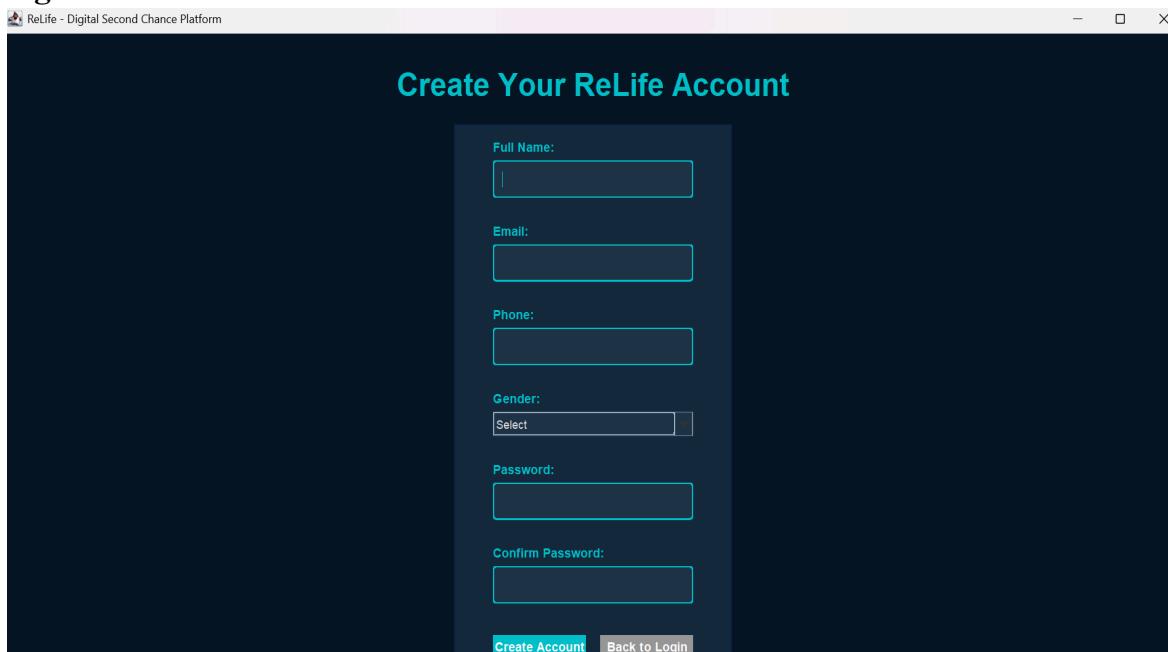


Fig 5.2 REGISTRATION PAGE:



DASHBOARD PANEL

Fig 5.3 DASHBOARD PAGE:

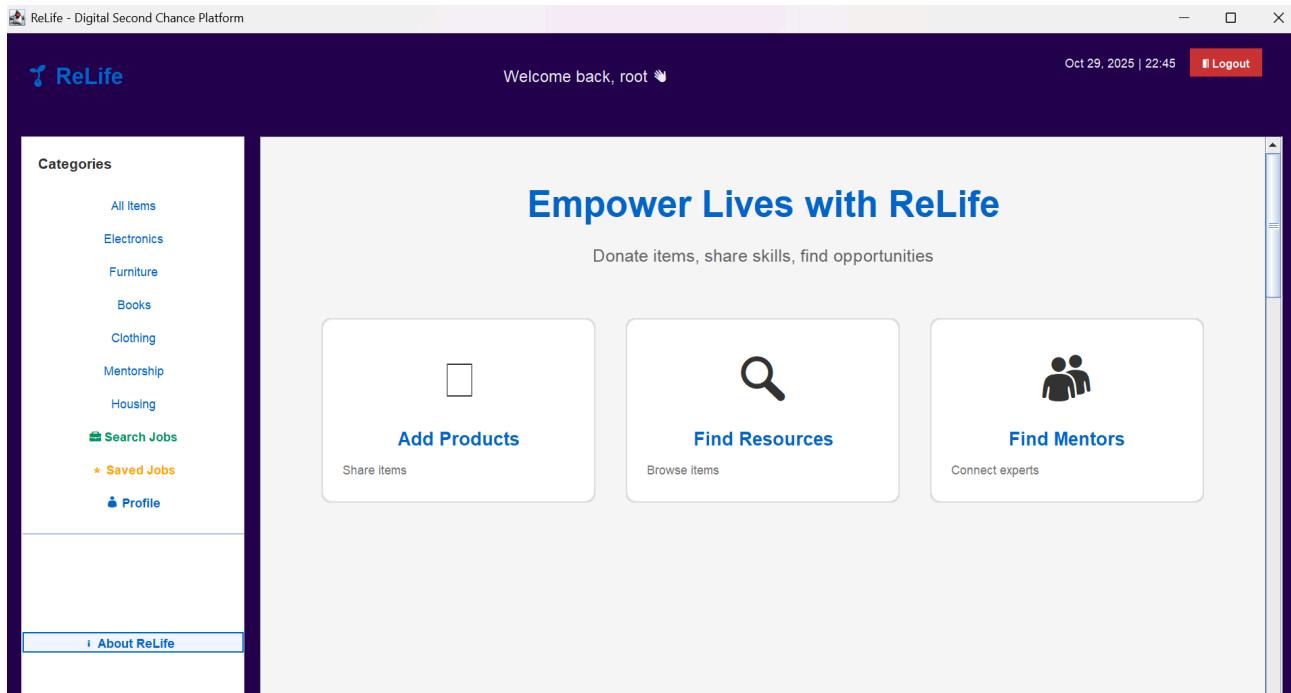
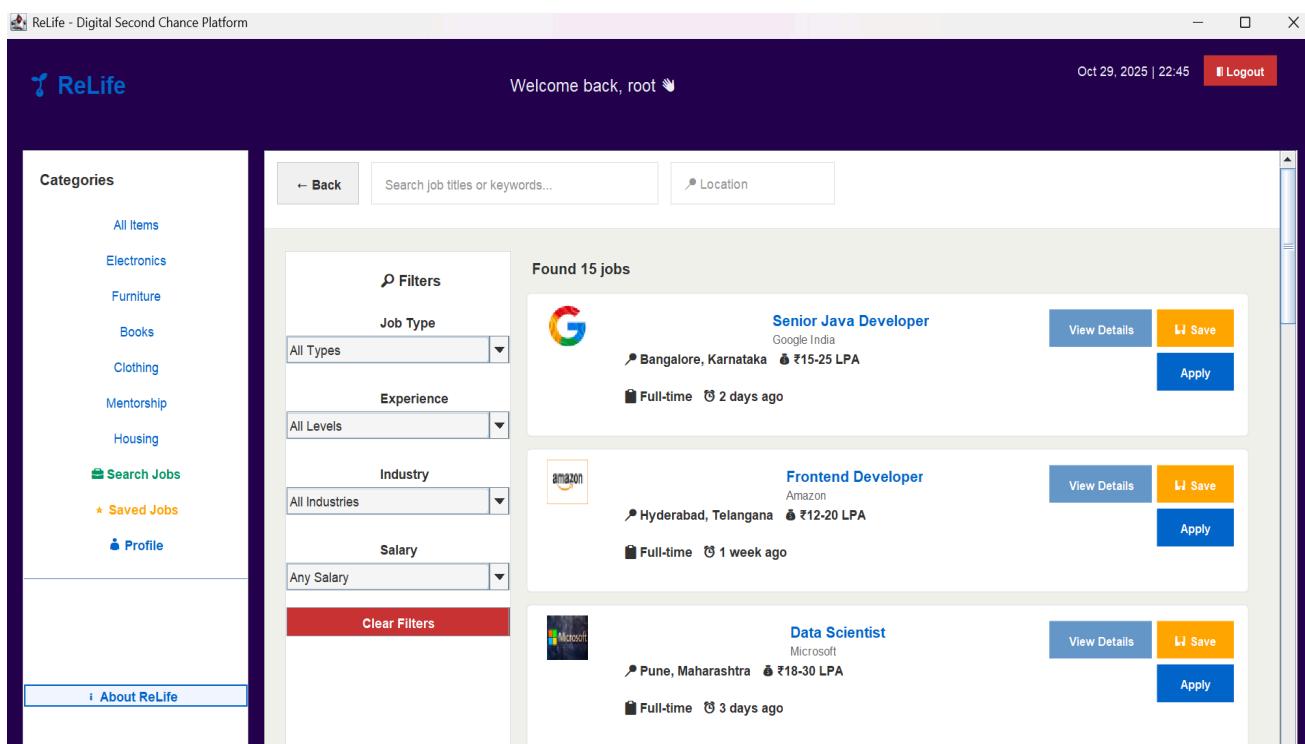
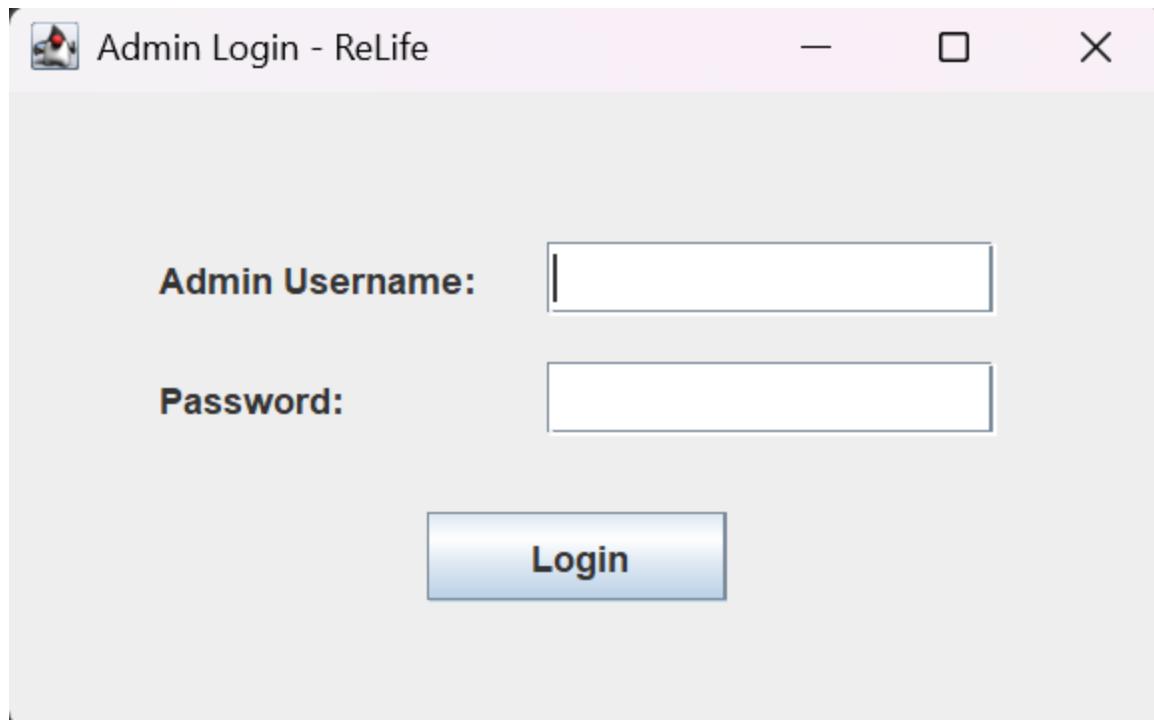


Fig 5.4 JOB SEARCH PANEL:



ADMIN LOGIN

Fig 5.5 ADMIN LOGIN:



ADMIN FRAME

Fig 5.6 ADMIN FRAME(USER LIST):

The screenshot shows a Windows application window titled "ReLife Admin Dashboard". The main title bar is "ReLife Admin Panel". Below the title bar is a table with the following columns: User ID, Username, Email, Phone, Gender, and Created At. The data in the table is as follows:

User ID	Username	Email	Phone	Gender	Created At
7	pant	pant@gmail.com	9874653728	Male	2025-10-30 00:10:00
6	neymar	neymar@gmail.com	6123662270	Male	2025-10-29 22:59:07
5	cook	cook@gmail.com	8755643924	Male	2025-10-29 22:09:29
4	ross	ross@gmail.com	9874143780	Male	2025-10-28 18:20:08
2	groot	groot@gmail.com	0987654321	Male	2025-10-26 15:18:14
1	root	root@gmail.com	1234567890	Male	2025-10-26 15:11:23

At the bottom of the window are four buttons: "View All Users", "Load Data", "Delete Selected User", and "Logout".

Fig5.7 ADMIN FRAME(QUALIFICATION):

The screenshot shows a Windows application window titled "ReLife Admin Dashboard". The main title bar is "ReLife Admin Panel". Below the title bar is a table with the following columns: Username, Degree, Field of Study, Institution, Graduation Year, and Created At. The data in the table is as follows:

Username	Degree	Field of Study	Institution	Graduation Year	Created At
ross	B.Tech	Computer Science an...	IIT Madras	2022	2025-10-30 00:19:55
cook	MBA	Human Resource Man...	IIM Bangalore	2021	2025-10-30 00:19:55
neymar	B.Com	Finance and Accounting	Delhi University	2019	2025-10-30 00:19:55
pant	M.Sc	Data Science	Anna University	2023	2025-10-30 00:19:55
root	Bachelor of Technology	Computer Science an...	National Institute of Te...	2023	2025-10-29 19:44:45
groot	Master of Business A...	Marketing and Analytics	Indian Institute of Man...	2022	2025-10-29 19:44:45

At the bottom of the window are four buttons: "View All Qualifications", "Load Data", "Delete Selected User", and "Logout".

CONCLUSION

CHAPTER 6

CONCLUSION AND FUTURE ENHANCEMENT

CONCLUSION : The ReLife Job Portal System successfully bridges the gap between job seekers and employers through a simplified and efficient digital platform. This application provides a secure user authentication system, profile management, job application, and saving functionalities — all supported by a robust MySQL database. The addition of the **Admin Dashboard** further enhances the system's control and monitoring capabilities, allowing administrators to manage user data, view saved and applied jobs, and oversee system activity efficiently. The project demonstrates the effective integration of Java Swing for front-end design and MySQL for backend data handling, ensuring both usability and performance. Overall, the system achieves its goal of providing a structured, interactive, and user-friendly platform for employment-related interactions.

FUTURE ENHANCEMENT :

Resume Upload & Parsing: Allow users to upload resumes and automatically extract skills, education, and experience to populate their profiles.

AI-Based Job Recommendations: Implement machine learning algorithms to suggest jobs based on user activity, skills, and preferences.

Email Notification System: Send automated emails to users when their job applications are approved, rejected, or when new jobs matching their profile are posted.

Mobile Application Integration: Extend the portal to a mobile app (Android/iOS) for wider accessibility.

REFERENCES

REFERENCES

- 1. <https://docs.oracle.com/javase/tutorial/jdbc/index.html>**
- 2. <https://dev.mysql.com/doc/refman/8.0/en/>**
- 3. <https://dev.mysql.com/doc/connector-j/8.0/en/>**
- 4. <https://www.oracle.com/java/technologies/data-access-object.html>**