DLithe™
Be Agile..Be in Demand

SKILL DEVELOPMENT

Project Report

DLithe Consultancy Services Pvt. Ltd.

DLithe™
Be Agile..Be in Demand

Project Report Assessment

Student Name: Akshata Gurav

Reg. no:2JR23CS008

Assignment: Java
Organization: DLithe Consultancy Services Pvt. Ltd.
Supervisor's Name: Archana SM

Submitted to

Signature of Training Supervisor

Date:

Signature of Students

Date:

DLithe™
Be Agile..Be in Demand

TABLE OF CONTENTS

DLithe™
Be Agile..Be in Demand

1.INTRODUCTION

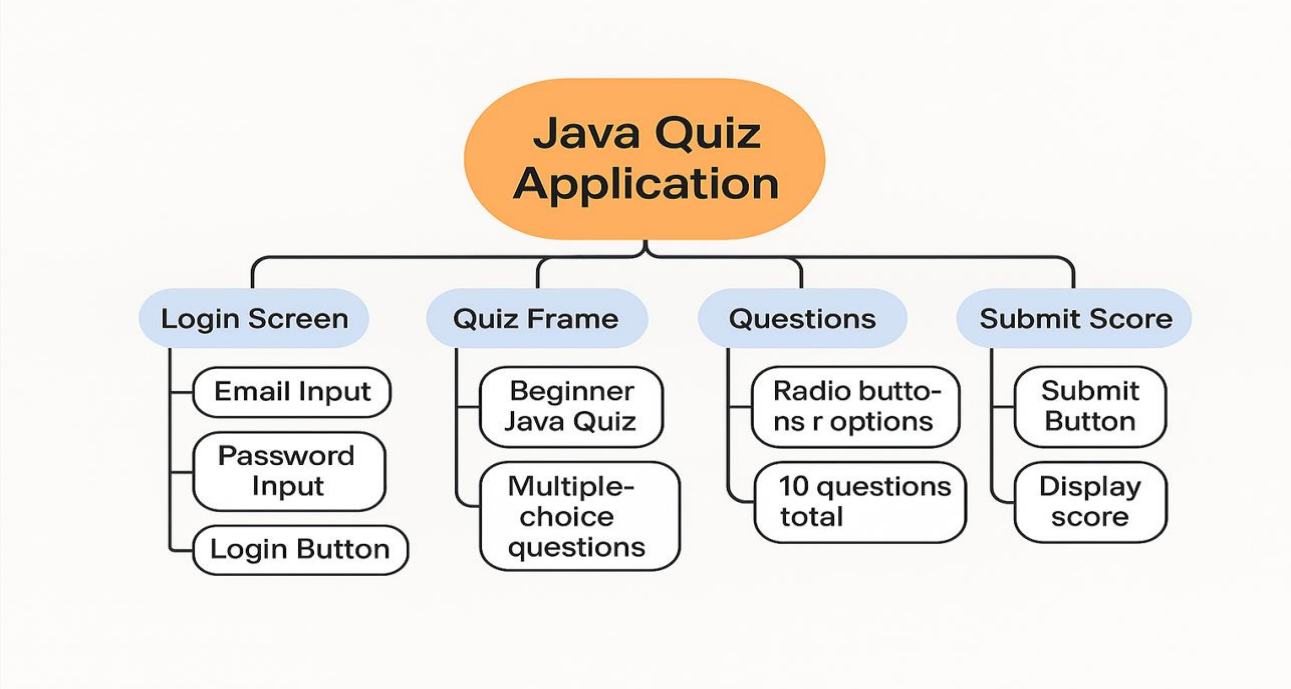
The Java Quiz Application is a GUI-based desktop application designed using Java Swing. The application allows users to log in and then take a 10-question technical multiple-choice quiz targeted at beginners. It provides an interactive environment for testing programming knowledge and understanding basic Java concepts.

The project covers core Java concepts such as:

- Object-Oriented Programming (OOP)
- Swing for GUI development
- Event Handling
- Arrays and ButtonGroups
- Basic form validation and navigation

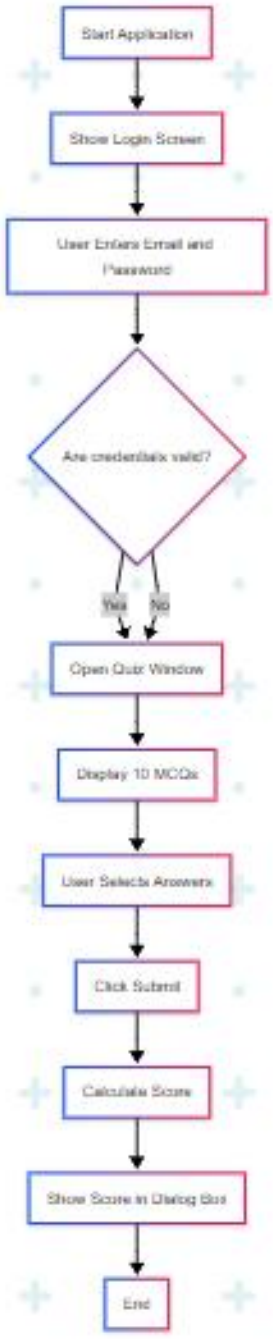
It serves as a real-world beginner-friendly Java project demonstrating how GUIs work in Java, with logic structured for easy expansion.

MIND-MAP



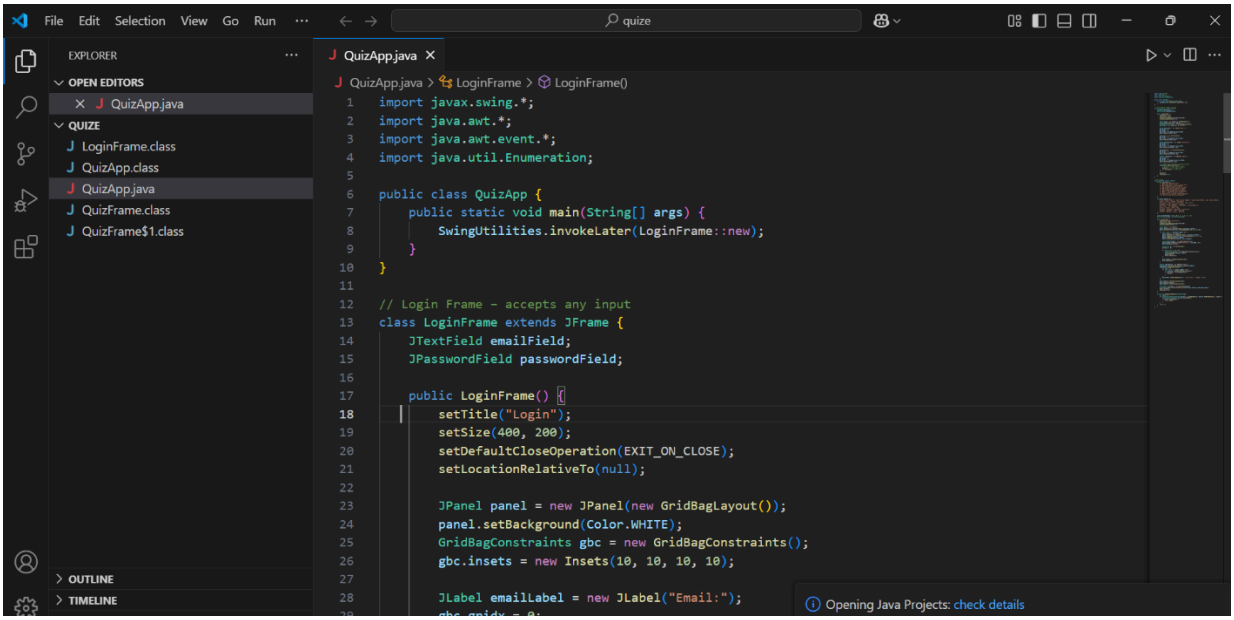
DLithe™
Be Agile..Be in Demand

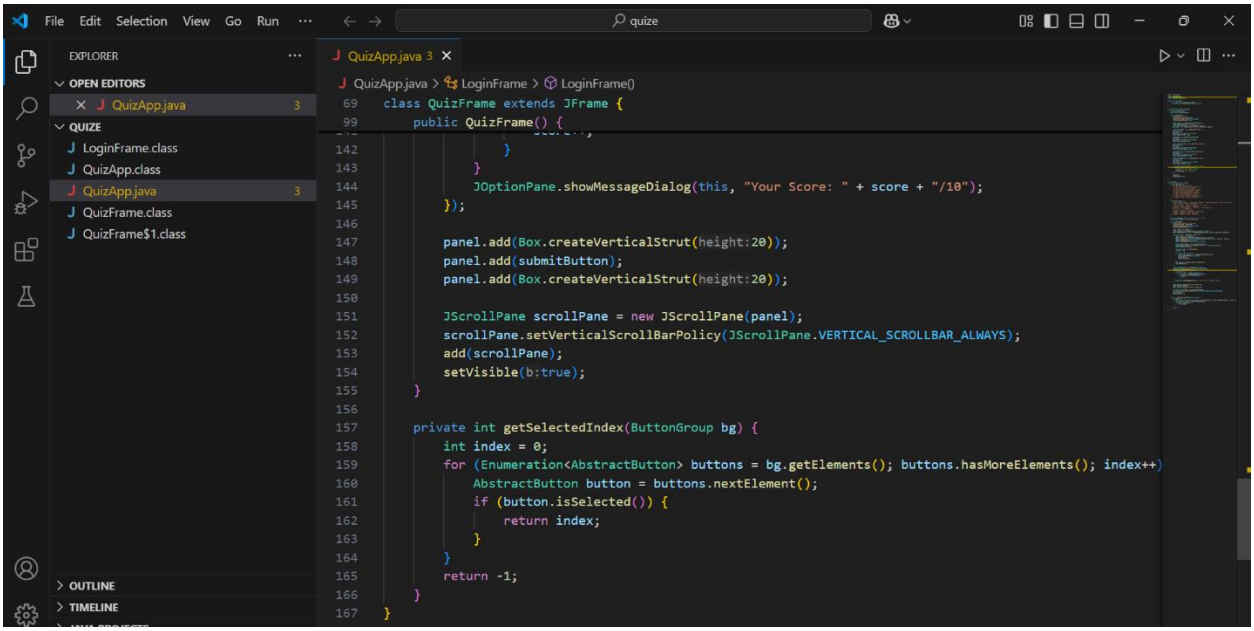
FLOWCHART



DLithe™
Be Agile..Be in Demand

Project Development Images(CODE)





DLithe™
Be Agile..Be in Demand

Email: akshatag395@gmail.com

Password: ••••••••••••••

Login

- ==
- !=
- equals()

7. What is the default value of boolean?
- true
- false
- null
- 0

8. What is used to inherit a class?
- extend
- extends
- implement
- inherits

9. Which loop is used for fixed iteration?
- while
- do-while
- for
- switch

10. What is used to define constants?
- const
- define
- final
- static

DLithe™
Be Agile..Be in Demand

2.Background

The goal behind developing this project was to implement Java Swing in a hands-on application that mimics real educational platforms. By combining login functionality and quiz logic, this project replicates the flow of many technical test portals. It also builds confidence in GUI programming — a common skill tested in Java-based development roles.

3.Technologies used

- Java: Used as the core programming language to build the application using object-oriented principles.
- Java Swing: Provided GUI components such as JFrame, JPanel, JLabel, JButton, JRadio-Button, etc., to design the login and quiz interface.
- AWT Event Handling: Enabled interaction with the application through listeners like Action-Listener for button clicks.
- JRadioButton & ButtonGroup: Used to allow users to select only one option per multiple-choice question.
- Arrays: Used to store the questions, options, and correct answers.
- Layouts (BoxLayout & GridBagLayout): Helped structure the GUI components neatly across login and quiz screens.
- JScrollPane: Added scrolling support to the quiz panel for better user experience when dis-playing multiple questions.
- JOptionPane: Used to display the final score in a dialog box after quiz submission.
- (IntelliJ IDEA / VS Code): Used for coding, compiling, and debugging the application.

4.Training Experience

- Gained hands-on experience with Java Swing layouts.
- Learned about organizing large GUI projects using multiple classes.
- Understood how to use ButtonGroup, JRadioButton, and JScrollPane.

- Practiced validating user inputs and GUI event handling.
- Built a modular and expandable structure for future quiz versions.

5.Key Learnings

- Swing UI Design: Learned how to layout forms and manage components using BoxLayout, GridBagLayout, etc.
- OOP Principles: Used class-based structure to separate login and quiz logic.
- Event Handling: Implemented listeners for button actions and scoring logic.
- UI-UX Concepts: Improved user interaction through scrollable quizzes and clean layouts.
- Error Handling: Debugged layout and logic errors in real time using IntelliJ and VS Code.

6.Challenges Faced

- Understanding layout managers (GridBagLayout, BoxLayout) was initially confusing.
- Creating reusable quiz question panels required experimentation.
- Managing long quiz forms without scrollbars caused UI issues.
- Login logic had to be adjusted to accept any credentials (as per requirement).
- Initial attempts caused labels to be cut off — fixed by replacing titled borders with JLabel.

7.Applications of My Project

- Acts as a basic Java-based exam portal simulation.
- Can be extended to include:
    - User database
    - Timer for quiz

    - Answer review and feedback
- Useful as a technical demo for resumes or internships.
- Great practice for building Java GUI projects with real-world structure.

8.Conclusion

The Java Quiz Application demonstrates how to design auser-friendly technical test using Java Swing. It provides anexcellent introduction to GUI-based programming and covers foundational Java skills like classes, event handling, user input, andlayout management.

This project has deepened understanding of bothGUI structure and interactive software design in Java.