Online Quiz System

Submitted By

Student Name	Student ID
Farhan Sadik Kawsar	241-15-462
Yasin Arafat Shibloo	241-15-288
Mohima Hossain Safa	241-15-213
Md. Sazidul Zidan	241-15-268

MINI LAB PROJECT REPORT

This Report Presented in Partial Fulfillment of the course **CSE222: Object-Oriented Programming Lab in the Computer Science and Engineering Department**

DAFFODIL INTERNATIONAL UNIVERSITY
Dhaka, Bangladesh

August 10, 2025

DECLARATION

We hereby declare that this lab project has been done by us under the supervision of **MD. Jubayar Alam Rafi, Lecturer** Department of Computer Science and Engineering, Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere as lab projects.

Submitted To:

MD. Jubayar Alam Rafi
Lecturer
Department of Computer Science and Engineering
Daffodil International University

Submitted by

Farhan Sadik Kawsar 241-15-462 Dept. of CSE, DIU	Yasin Arafat Shibloo 241-15-288 Dept. of CSE, DIU
Mohima Hossain Safa 241-15-213 Dept. of CSE, DIU	Md. Sazidul Zidan 241-15-268 Dept. of CSE, DIU

COURSE & PROGRAM OUTCOME

The following course have course outcomes as following:

Table 1: Course Outcome Statements

CO's	**Statements**
CO1	**Define** and **Relate** classes, objects, members of the class, and relationships among them needed for solving specific problems
CO2	**Formulate** knowledge of object-oriented programming and Java in problem solving
CO3	**Analyze** Unified Modeling Language (UML) models to **Present** a specific problem
CO4	**Develop** solutions for real-world complex problems **applying** OOP concepts while evaluating their effectiveness based on industry standards.

Table 2: Mapping of CO, PO, Blooms, KP and CEP

CO	**PO**	**Blooms**	**KP**	**CEP**
CO1	PO1	C1, C2	KP3	EP1,EP3
CO2	PO2	C2	KP3	EP1,EP3
CO3	PO3	C4, A1	KP3	EP1,EP2
CO4	PO3	C3, C6, A3, P3	KP4	EP1,EP3

The mapping justification of this table is provided in section **4.3.1**, **4.3.2** and **4.3.3**.

Table of Contents

Chapter 1

Introduction

1.1 Introduction

This project is a **Java-based Online Quiz System** that facilitates quiz creation, participation, and evaluation through an interactive CLI interface. The system supports two user types: **Admin** and **QuizParticipant**, simulating an end-to-end quiz environment.

1.2 Motivation

In academic and training settings, evaluating learning outcomes digitally is growing. A simple, object-oriented quiz system fosters automation, reducing manual work and encouraging fair evaluations.

1.3 Objectives

- Design a CLI-based quiz system using OOP in Java
- Allow Admin to create/delete quizzes and view all results
- Let users take quizzes and view their results.
- Store and compute results using objects and basic data structures.

1.4 Feasibility Study

Unlike complex web-based apps, this system focuses on **command-line interactivity**, Java object modeling, and beginner-friendly logic – ideal for foundational software engineering practices.

1.5 Gap Analysis

Unlike complex web-based apps, this system focuses on **command-line interactivity**, Java object modeling, and beginner-friendly logic – ideal for foundational software engineering practices.

1.6 Project Outcome

A working Java-based quiz application that uses OOP principles such as abstraction, inheritance, encapsulation, and polymorphism. It enhances both Java and system design skills

Chapter 2

Proposed Methodology/Architecture

Every chapter should start with 1-2 sentences on the outline of the chapter.

2.1 Requirement Analysis & Design Specification

2.1.1 Overview

The system consists of classes for User, Admin, QuizParticipant and

Result. Admin manages quizzes, while users participate.

2.1.2 Proposed Methodology/ System Design

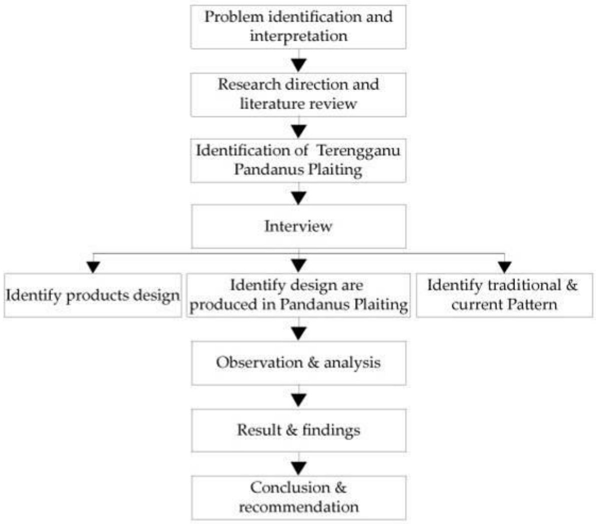
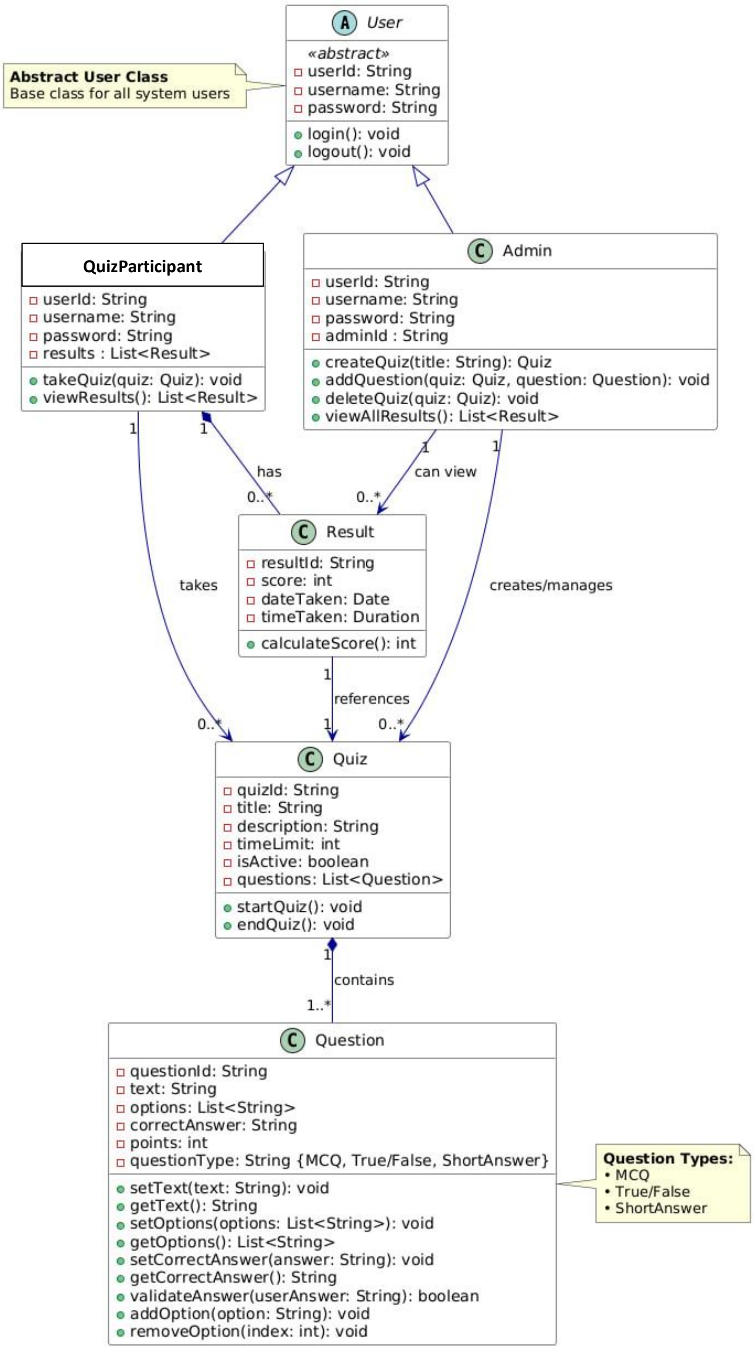

Problem identification and interpretation
Research direction and literature review
Identification of Terengganu Pandanus Plaiting
Interview
Identify products design
Identify design are produced in Pandanus Plaiting
Identify traditional & current Pattern
Observation & analysis
Result & findings
Conclusion & recommendation

Figure 2.1: This is a sample diagram

2.1.3 UML Design



2.2 Overall Project Plan

- Week 1: Requirements & design
- Week 2: Implement base classes
- Week 3: Integrate Admin/User panels
- Week 4: Finalize features and test

Chapter 3

Implementation and Results

Every chapter should start with 1-2 sentences on the outline of the chapter.

3.1 Implementation

The system is implemented in Java using core OOP techniques:

- User is abstract and extended by Admin and QuizParticipant.
- QuizParticipant participates in quizzes.
- Quiz contains multiple Question instances.
- Admin can view all results of users.

3.2 Performance Analysis

Efficient in memory and runtime for small-scale quizzes. Suitable for low-resource systems or learning environments.

3.3 Results and Discussion

After testing with various users and quizzes:

- Admin can successfully create quizzes and monitor results.
- Users can take quizzes and receive immediate feedback.
- System prevents crashes with basic input validation.

Chapter 4

Engineering Standards and Mapping

Every chapter should start with 1-2 sentences on the outline of the chapter.

4.1 Impact on Society, Environment and Sustainability

4.1.1 Impact on Life

Encourages self-assessment and learning reinforcement.

4.1.2 Impact on Society & Environment

Reduces the need for printed quiz papers – paperless testing.

4.1.3 Ethical Aspects

Promotes fair, consistent quiz-taking experiences.

4.1.4 Sustainability Plan

Expandable for more features like user registration, file-based persistence**.**

4.2 Project Management and Team Work

The project was planned weekly. No external budget was required due to the open-source nature of . Java. Tools used: IntelliJ IDEA / VS Code.

4.3 Complex Engineering Problem

4.3.1 Mapping of Program Outcome

In this section, provide a mapping of the problem and provided solution with targeted Program Outcomes (PO's

Table 4.1: Justification of Program Outcomes

PO's	**Justification**
PO1	Applied class modeling and user interaction principles
PO2	Developed with Java OOP concepts like encapsulation and inheritance
PO3	Designed solution using problem decomposition and system modeling

4.3.2 Complex Problem Solving

In this section, provide a mapping with problem solving categories. For each mapping add subsections to put rationale (Use Table 4.2). For P1, you need to put another mapping with

Knowledge profile and rational thereof.

Table 4.2: Mapping with complex problem solving.

EP1 Dept of Knowledge	EP2 Range of Conflicting Requirements	EP3 Depth of Analysis	EP4 Familiarity of Issues	EP5 Extent of Applicable Codes	EP6 Extent of Stakeholder Involvement	EP7 Interdependence
√	√					

4.3.3 Engineering Activities

In this section, provide a mapping with engineering activities. For each mapping add subsections to put rationale (Use Table 4.3).

Table 4.3: Mapping with complex engineering activities.

EA1 Range of resources	EA2 Level of Interaction	EA3 Innovation	EA4 Consequences for society and environment	EA5 Familiarity
√	√			

Chapter 5

Conclusion

Every chapter should start with 1-2 sentences on the outline of the chapter.

5.1 Summary

The project successfully demonstrates the development of a quiz system using core Java OOP. It achieves the basic educational goals of simulation and logic-based evaluation.

5.2 Limitation

- No persistent storage (e.g., file or database)

- Limited to CLI – no GUI

5.3 Future Work

- Add GUI using JavaFX or Swing
- Introduce file/database persistence
- Enable user registration and authentication

References

[1] Jon Kleinberg and Eva Tardos. *Algorithm Design*. Pearson Education India, 2006.
[2] Java SE Documentation – Oracle
[3] Effective Java by Joshua Bloch.

Source Code : github Link