MOBILE COMPUTING & APPLICATIONS LAB REPORT

Advanced OOP-Based Quiz Web Application (Student–Teacher System)

Team Members
1. Abu Bakkar Siddik – ID: 233003122E (All Coding & System Development)

2. Tipu Sultan – ID: 2330029222E (Documentation, Report, Diagrams)

Project Overview
This project is an advanced quiz management web application designed using HTML, CSS, JavaScript, and OOP principles. It includes Student Login, Teacher Login, dynamic subjects, time customization, randomized questions, and a teacher-only backend marks section similar to Google Forms.

Technologies Used

- HTML5 – structure of pages
- CSS3 – styling & responsive UI
- JavaScript (ES6) – logic, OOP classes, role system,
- Local Storage – storing student quiz results backend-like for the teacher
- Browser DOM API – dynamic UI rendering

Why OOP Was Used
OOP makes the system modular, scalable, and maintainable.

- Class Question – stores question text, options, and correct index

- Class Quiz – handles question flow, navigation, and answer storage
- Class Timed Quiz (extends Quiz) – adds timer functionality
- Class Result – stores score, grade, analytics
- Class Subject Manager – dynamic subject creation
- Class QuizService – central logic to load quiz, randomize questions, calculate score

Why JavaScript

JavaScript allows building full logic inside the browser without a backend server. It was used to:

- Handle login system (student/teacher)
- Create OOP classes
- Store marks in local Storage like backend
- Dynamically render UI like Google Forms.

Major Code Components

class Question {...}

class Quiz {...}

class TimedQuiz extends Quiz {...}

class Result {...}

class SubjectManager {...}

class QuizService {...}

Each class performs specific tasks, making system modular.

Student Features

- Login with Name & ID
- Choose subject
- Choose number of questions
- Custom or fixed timer
- Randomized questions & options

- Submit & get detailed result
- Full answer review

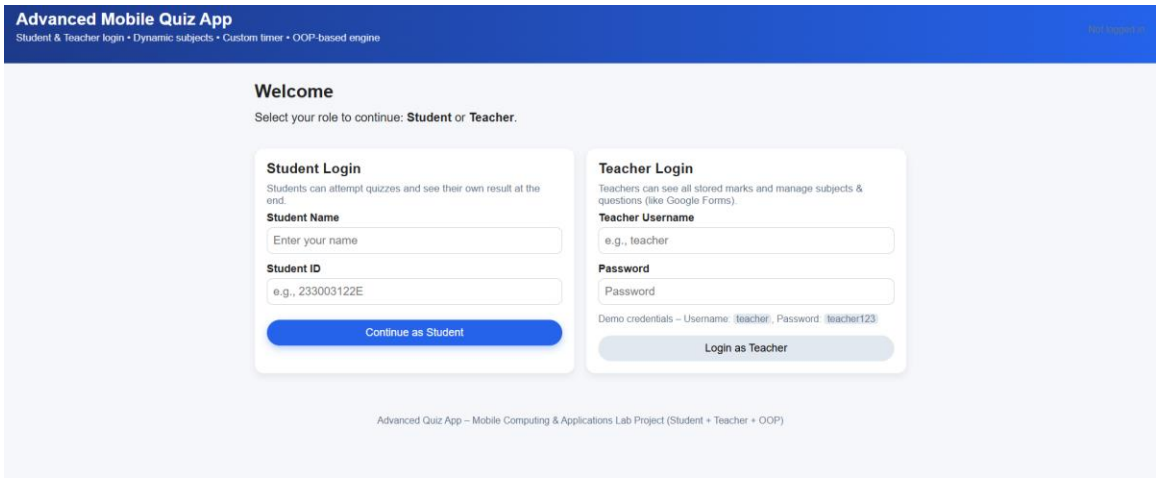
Teacher Features

- Login (username: teacher, password: teacher123)
- View backend Marks Table (stored using localStorage)
- Add subjects dynamically
- Add questions dynamically (like Google Forms)
- Review performance of all students

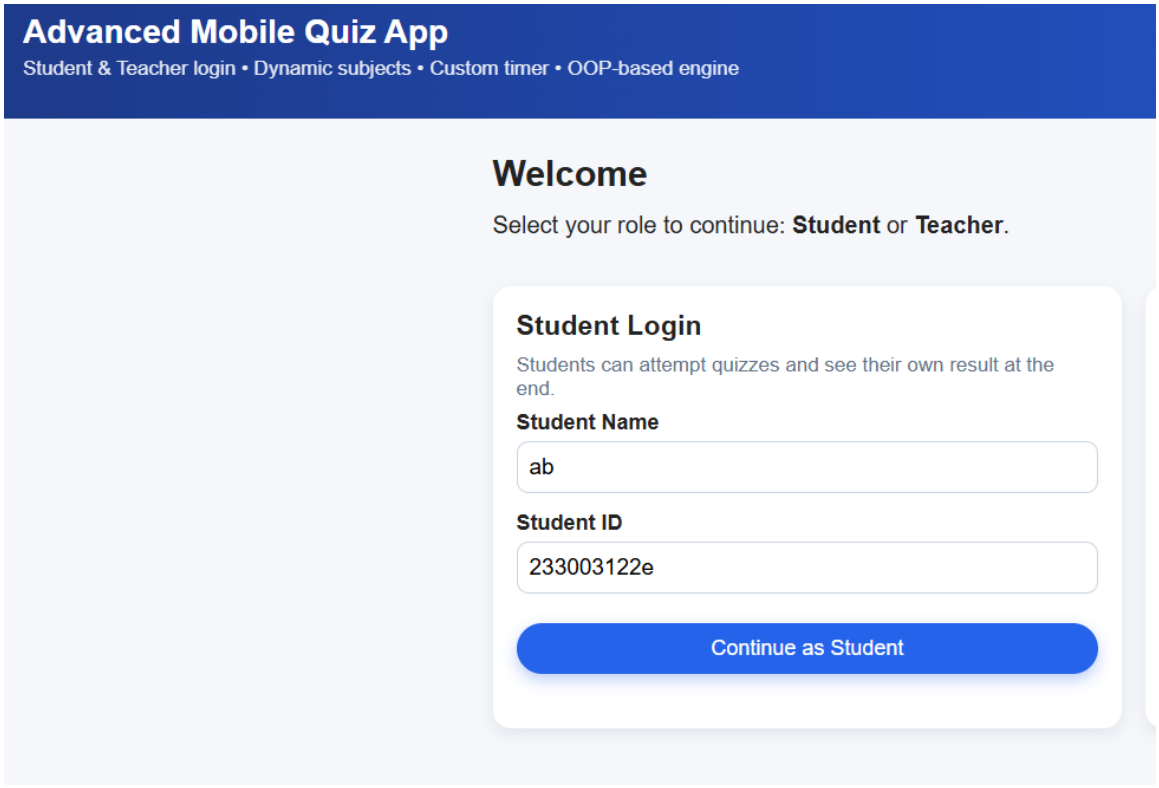
Why This System Is Advanced

- Multiple roles implemented (Student + Teacher)
- Dynamic subject & question management
- Timer customization
- Randomization logic
- Full OOP implementation
- Teacher-only backend marks section
- Works fully offline

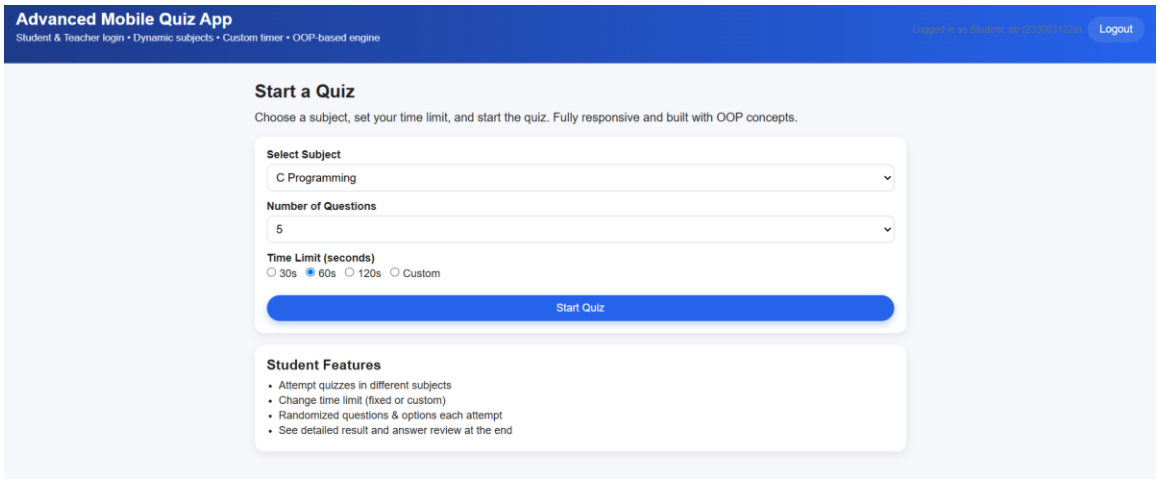
<u>The over view:</u>



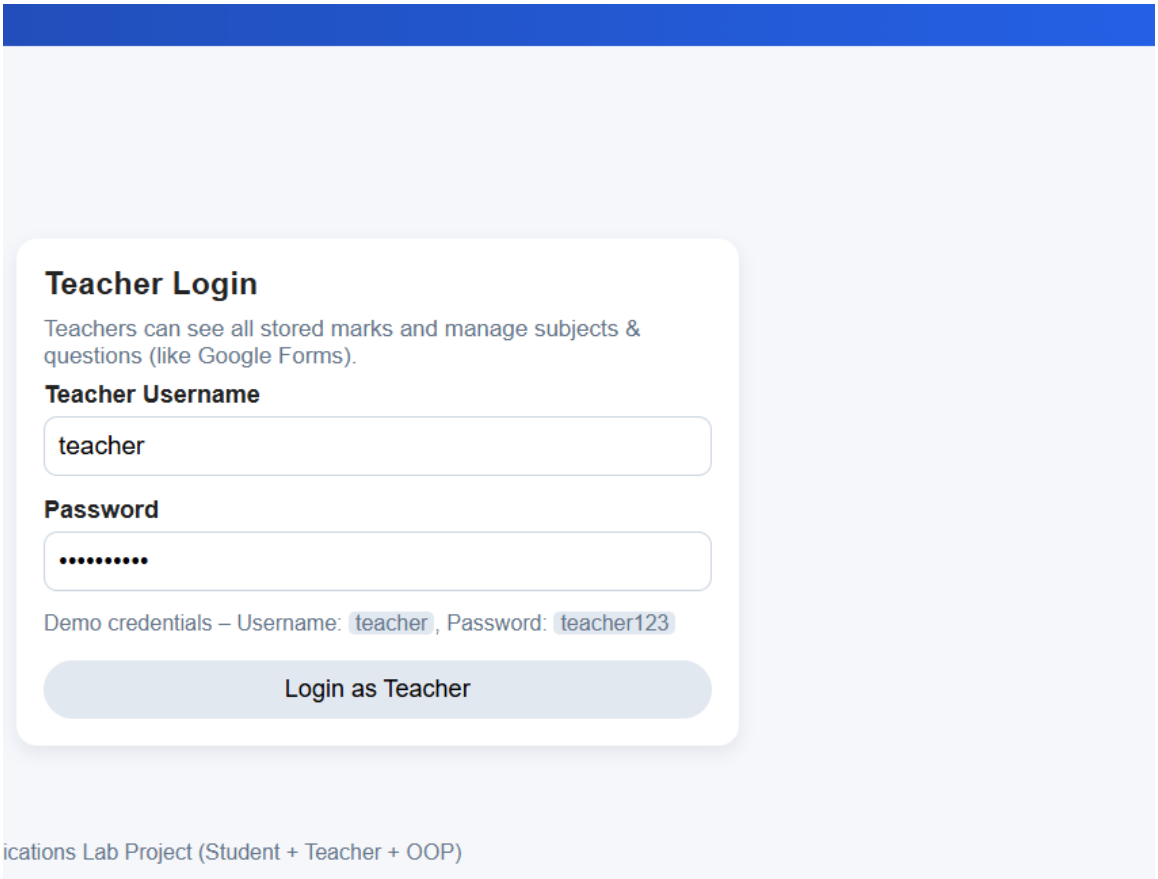
Student login:



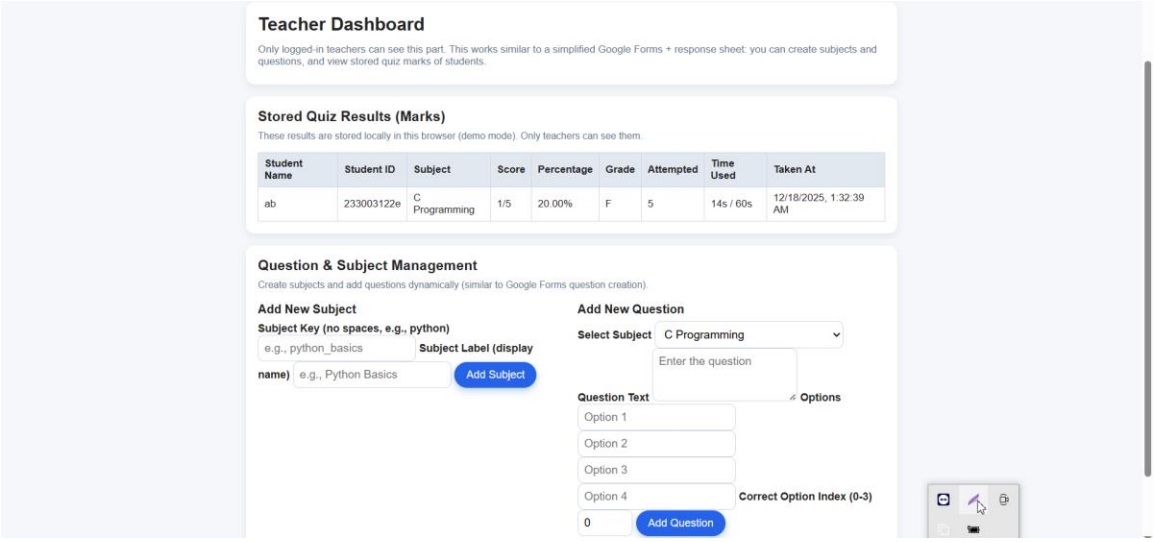
Student Quiz view:



Teacher login:



Teacher Dashboard:



Conclusion

This project successfully meets Mobile Computing lab standards with advanced features. Team effort:

1. Abu Bakkar Siddik – Developed full codebase
2. Tipu Sultan – Created documentation & report

The system is scalable, secure for demo, and similar to Google Forms.