Project Report

Project TITLE: **Quiz Master - V2**

Student Details:

- Name: MADHAV SHARMA
- Roll Number: 23f3004142
- Course/Program: Student at Diploma Level.

Project Overview

Introduction: It is a multi-user app (one requires an administrator and other users) that acts as an exam preparation site for multiple courses.

Objective: The system will implement role-based authentication using Flask Security or JWT, ensuring only one admin with elevated privileges. An admin dashboard will be provided for managing subjects, chapters, quizzes, and users, with search functionality and summary charts. Users will be able to attempt quizzes with timers, view previous attempts, and access performance summaries. Scheduled jobs will handle daily reminders and monthly activity reports via email. Additionally, users and admins can export quiz data as CSVs through asynchronous jobs. Performance will be optimized using caching with appropriate expiry settings.

Frameworks and Libraries Used:

- Flask for backend, VueJS for UI
- Redis for caching, Redis and Celery for batch jobs
- Jinja2 templates + Bootstrap for HTML generation and styling
- SQLite for data storage

Database Design:

https://drive.google.com/file/d/1mJCcFXFhRu7Y6Cpeuz-_OI4vKAocwxvm/view?usp=sharing

Drive Link for Presentation Video:

https://drive.google.com/file/d/10NzTW5CF0l4XXSAWAK67UevCMJuX9eFq/view?usp=sharing

CORE FUNCTIONALITY	DESCRIPTION	ROUTES
1. Admin and User Login	A login/register form with fields like username and password for user and admin login.	Frontend: {path: '/', component: Home}, {path: '/login', component: Login}, {path: '/register', component: Register}. Backend: @app.route('/', methods = ['GET']) @app.route('/logout', methods=['GET']), @app.route('/api/register', methods=['POST']) #or @app.post('/api/register'), @app.route('/api/login', methods=['POST']) @app.route('/api/home')
2. Admin Dashboard	Admin login redirects to a dashboard where they can do CRUD on subject,chapter,quiz,question,option	Frontend: {path: '/admin', component: Admin}, Backend: @app.route('/api/adminDASHBOARD')
3. User Dashboard (User)	The user can attempt any quiz of his/her interest, every quiz has a timer, each quiz score is recorded, the earlier quiz attempts are shown, to be able to see the summary charts	Frontend: {path: '/dashboard', component: Dashboard}, Backend:/search/users
4. Summary Charts (User, Admin)	Inferences drawn from quiz data , user_scores, toppers_scores etc	Frontend: {path: '/summary/:userId', component: useRSummary}, {path: '/adminSummary', component: adminSummary}, Backend:@app.route('/admin/summary', methods=['GET'])
5. Search for Available Quizzes,users,chapters,subjects	Admin can search for quiz,user,chapter,subject	Backend: ('/search/users')('/search/subjects')('/search/quizzes')('/search/chapters'),('/search/options')