Project Kit

Title of the Project

Online Skill Test Platform

Abstract of the Project

The Online Skill Test Platform is a full-featured web application aimed at evaluating and certifying user skills across various sectors and proficiency levels. The system is designed for students, job seekers, and employers to participate in and benefit from standardized, adaptive assessments. The backend is developed using Python Django and the frontend uses HTML and CSS. The platform allows for secure registration, test delivery, real-time scoring, feedback generation, and administrative control. The goal is to bridge the gap between academic qualifications and actual industry skills, promoting continuous learning and career growth.

Keywords

Generic: Skill Assessment, Adaptive Testing, Sector-Wise Exams, Certification, Online Learning, User Dashboard

Technical: Django, HTML, CSS, SQLite, Python Web Development, User Authentication, REST APIs

Functional Components

1. **User Registration and Login:**
   Provides secure account management using Django's authentication framework.

2. **Sector-Based Exam Selection:**
   Users can select tests based on industry sectors and proficiency levels.

3. **Adaptive Question Flow:**
   Questions adjust dynamically based on previous responses.

4. **Result Calculation and Feedback:**
   Instant result computation and personalized suggestions for improvement.

5. **Employer Interface:**
   Employers can view candidate profiles, certifications, and test scores.

6. **Admin Dashboard:**
   Admin can manage users, questions, view reports, and perform data analytics.

7. **Data Protection:**
   Implements secure handling of personal and assessment data using Django's built-in tools.

Functionality Overview

The platform facilitates the complete lifecycle of online skill testing: registration, test selection, execution, evaluation, and feedback. Users (students/professionals) register, log in, choose from sector-specific tests, and take adaptive assessments. Results are computed immediately and feedback is shown. Admins manage system-wide operations, including adding/removing tests and managing user data. Employers can explore scores to identify top talent.

Steps to Start-Off the Project

1) Study core technologies: Django for backend, HTML/CSS for frontend, and SQLite for database.
2) Analyze similar platforms (like LinkedIn Skill Assessments) to gather feature insights.
3) Define user roles and access levels: Admin, Student, Employer.

4) Design intuitive UI screens: login, test-taking interface, result dashboard.
5) Develop REST APIs for backend operations such as test submission and result handling.
6) Secure all forms of communication and ensure data validation and sanitization.

System Requirements

❖ Hardware Requirements

Processor: Minimum x86-64 bit, Recommended: Intel i5/Ryzen 5 or higher
• RAM: Minimum 4 GB, Recommended 8 GB
• Disk Space: Minimum 5 GB, Recommended SSD with 10+ GB free space

❖ Software Requirements

- Operating System: Windows/Linux/macOS
• Backend: Python 3.10+, Django 4+
• Frontend: HTML5, CSS3
• Database: SQLite (default), PostgreSQL (optional for production)
• Tools: VS Code, Postman, Git, Browser (Chrome/Firefox)

❖ Manpower Requirements

- Team Size: 2–3 students
• Project Duration: 4 to 6 months (full-time)

Milestones and Timelines

Milestone Name	Description	Timeline (Week)
Requirements Specification	Identify target users, test types, scoring mechanism	Weeks 2–3

Technology Familiarization	Setup Django environment and HTML/CSS templates	Week 4
Database Creation	Design models for Users, Questions, Results, Certificates	Weeks 5–6
High-Level Design	Draw data flow diagrams, define modular design	Weeks 7–8
Frontend Implementation	Build UI for login, test taking, feedback	Weeks 9–10
Integration Phase	Connect backend APIs with frontend and test flows	Weeks 11–12
Testing	Manual and automated testing of all features	Weeks 13–14
Final Review	Submit report, project demo, resolve bugs	Weeks 15–16

Guidelines and References

- ReactJS Documentation: https://react.dev/reference/react
- Django Documentation: https://docs.djangoproject.com
- Wikipedia: https://www.wikipedia.com/
- Database Systems – Raghu Ramakrishnan or Elmasri & Navathe
- Object-Oriented Modeling with UML – Michael Blaha, James Rumbaugh
- Software Engineering – Ian Sommerville (7th Edition)