Title

SkillForge – AI-Driven Adaptive Learning and Exam Generator

Table of Contents

---

1. Project Overview

SkillForge is an AI-driven adaptive learning platform that delivers personalized learning paths and automatic exam generation. The system adapts difficulty based on learner performance, recommends topics and remedial lessons, and provides assessments with feedback and analytics.

---

2. Objectives

- Provide personalized learning paths using AI adaptivity.
- Generate exam papers dynamically using NLP and performance data.
- Track, assess, and report learner performance.
- Deliver notifications, tips, and feedback to learners.
- Offer admin/instructor dashboards to manage content and users.

---

3. Scope

In-scope: all components you provided — presentation layer, user & management, learning content, adaptive learning engine, exam generator, assessment/evaluation, notifications/feedback, analytics, AI/ML models, data storage, integration layer, and infrastructure.
Out-of-scope: any feature not explicitly listed in the provided content.

---

4. System Requirements

Functional Requirements

- Student, Instructor, Admin dashboards (web).
- Authentication and Role-Based Access Control (RBAC).
- CRUD for learning content (videos, PDFs, MCQs).
- Tagging of content by topic and difficulty (easy/medium).
- Adaptive learning engine: performance tracking, dynamic difficulty, topic recommendation.
- Exam generator: NLP-based blueprint + question paper generation based on performance.

- Assessment & evaluation: scoring, grading, and result storage.
- Feedback generation and notification service (tips, reminders, test feedback).
- Analytics & ML-driven weak-area analysis and performance prediction.

Non-functional Requirements

- Scalable message processing (message queue for heavy tasks).
- Secure data storage (encryption at rest and in transit).
- Monitoring and logging for system health.

---

5. System Architecture (Layered — using only your components)

Presentation Layer

- Web Interface
- Student Dashboard
- Admin / Instructor Dashboard

User & Management Module

- Authentication
- Role-Based Access Control (Student / Instructor / Admin)

Learning Content Module

- CRUD operations
- Videos, PDFs, MCQs
- Tagging: topics, difficulty (easy / medium)

Adaptive Learning Engine Module

- Learner performance tracking
- Dynamic difficulty adjustment
- Topic recommendation using AI

Exam Generator Module

- NLP-based blueprint anatomy
- Generates question papers based on performance

Assessment & Evaluation Module

- Assessment processing and evaluation logic

Feedback Module

- Generate feedback from assessments

Note: You listed “Feedback Module” and also “Notification & Feedback Service” separately. For clarity in architecture diagrams and implementation, combine them into a single module with two responsibilities: (a) generate feedback and (b) deliver notifications (tips, reminders, test feedback). Keep both names in the documentation but implement as one logical component.

Notification & Feedback Service

- Sends learning tips, reminders, and test feedback

Analytics & Insights Module

- Weak-area analysis using ML

AI / ML Layer

- Adaptivity models
- Exam generation models
- Recommendation models

Recommendation Engine

- Suggest next topic, learning path, remedial lessons

Performance Prediction Module

- Aggregation / classification to estimate student success rate

Data Layer

- Databases: user info, assessments, exam results, learning material, content metadata
- File storage: PDFs, videos, statistics content

Integration & Middleware Layer

- Communication between modules
- AI model generation API (OpenAI, custom LLMs, third-party LLMs)
- Message queue for scalability and syncing

Infrastructure Layer

- Cloud platform (AWS or Azure)
- Monitoring & logging (ELK stack or equivalent)
- Security: encryption at rest & in transit

---

6. Critical Interactions / Data Flows (fill these into your diagram tool)

Use these lines in the diagram — they are critical to show:

- Student Dashboard → Authentication → RBAC → User Profile DB
- Student Dashboard → Learning Content Module → File Storage (videos/PDFs)
- Adaptive Learning Engine ↔ Assessment Data (reads results, writes recommended difficulty)
- Exam Generator → AI/ML Layer (NLP model) → Generated Paper → Assessment Module
- Assessment & Evaluation → Feedback Module → Notification Service → Student Dashboard / Email
- Analytics & Insights Module ↔ Data Layer (reads assessments and interaction logs)
- Recommendation Engine → Student Dashboard (suggest next topic / remedial lessons)
- Integration Layer → External LLMs / AI APIs (for exam generation & recommendations)
- Message Queue mediates heavy tasks (exam generation, analytics jobs) for scalability

---

7. Module Descriptions (1–2 lines each — copy into report)

- Web Interface: Entry point for users; hosts dashboards and learning UI.
- Student Dashboard: Displays assigned content, progress, recommendations, upcoming tests.
- Admin / Instructor Dashboard: Manage users, content, view analytics, configure exams.

- Authentication: Verifies user identity; issues session tokens.
- Role-Based Access Control (RBAC): Grants permissions by role (Student/Instructor/Admin).
- Learning Content Module: CRUD for videos, PDFs, MCQs; tagging by topic and difficulty.
- Adaptive Learning Engine: Tracks performance and adjusts content difficulty; recommends topics using AI.
- Exam Generator: Uses NLP/LLM to create question papers tuned to learner performance.
- Assessment & Evaluation: Grades responses and computes skill-level metrics.
- Feedback Module: Produces performance feedback for learners and instructors.
- Notification & Feedback Service: Sends tips, reminders, and test feedback to users.
- Analytics & Insights: Runs ML to find weak areas and produce reports.
- AI/ML Layer: Houses adaptivity, exam generation, recommendation, and prediction models.
- Recommendation Engine: Suggests next topics and remedial lessons.
- Performance Prediction Module: Predicts learner success rates from aggregated data.
- Data Layer: Stores all persistent data and media files.
- Integration & Middleware: API gateway, message queues, and connectors to external LLMs.
- Infrastructure Layer: Cloud environment, monitoring, logging, and security settings.

---

8. Data Layer — details to populate

- User DB: user profiles, roles, authentication metadata
- Assessment DB: test attempts, question-level responses, scores
- Content DB: metadata for videos/PDFs/MCQs, tags, difficulty

- Results & Analytics DB: aggregated metrics for analytics modules
- File Storage: PDFs and video files (object store)
- Cache (optional): frequently-accessed content or session data

---

9. Integration & Middleware — details to populate

- API Gateway: routes requests from front-end to modules.
- Message Queue: handle exam generation, batch analytics, long-running ML tasks.
- AI / LLM APIs: OpenAI/custom LLM endpoints for exam generation and NLP tasks.
- Syncing & Event Stream: used for analytics pipelines and notifications.

---

10. Infrastructure & Security — short notes

- Use AWS or Azure (per your earlier note) for cloud hosting.
- Monitoring & Logging: ELK stack / Cloud-native logging.
- Security: encryption at rest & in transit; secure API keys for LLM services; RBAC enforcement.
