**LMS To-Be System Report**

**Project:** Learning Management System for The Network Co  
**Version:** MVP with AI Enhancements  
**Date:** June 9, 2025

**1. System Overview**

The To-Be LMS is an upgraded version of the existing system, aimed at creating a smarter, more interactive, and analytics-driven learning environment. It is designed to facilitate content delivery, automated assessments, and performance tracking, while introducing AI-powered analytics to personalize learning and support administrative decision-making.

**2. Current Functionality Overview**

**Learner Capabilities**

| **Feature** | **Status** | **Description** |
| --- | --- | --- |
| View Materials | Implemented | Learners can access and download resources uploaded by the instructor. |
| Take Quizzes | Implemented | Learners take multiple-choice quizzes. |
| Auto-Grading | Implemented | The system automatically grades quizzes. |
| Profile Management | Implemented | Learners manage their own personal information. |

**Admin Capabilities**

| **Feature** | **Status** | **Description** |
| --- | --- | --- |
| Upload Content | Implemented | Admins (instructors) upload course materials (PDFs, notes, videos). |
| Quiz Management | Implemented | Admins create multiple-choice quizzes in the system. |
| View Quiz Results | Implemented | Admins can view quiz scores submitted by learners. |

**3. Planned Enhancements (To-Be System)**

**Learner Enhancements**

| **Feature** | **Description** |
| --- | --- |
| Intelligent Assistant | Learners can ask an LLM chatbot questions about lessons, quizzes, or concepts. |
| Personalized Suggestions | The system will suggest next topics or materials based on performance. |
| Performance Dashboard | A visual summary of progress, quiz scores, and AI-generated insights. |

**Admin Enhancements**

| **Feature** | **Description** |
| --- | --- |
| Learner Analytics | Admins can view AI-generated insights about learner behavior and struggles. |
| Quiz Analytics | View per-question performance, drop-off rates, and most failed topics. |
| LLM Feedback Engine | Admins receive feedback on quiz/question clarity and difficulty via the LLM. |

**System-Wide Enhancements**

| **Feature** | **Description** |
| --- | --- |
| Daemon Thread for Analytics | A background process runs every 10–15 minutes, analyzing user data and sending it to an LLM for insight generation. |
| LLM Integration | An embedded LLM (DeepSeek ) analyzes learning patterns, behavior, and quiz data. |
| Data Collection Engine | Frontend scripts track time spent on pages, interaction logs, and learning progress. |
| Security & Privacy Layer | Cookie and session tracking comply with data privacy laws (GDPR/POPIA). |

**4. LLM Role in the System**

| **Module** | **Function** |
| --- | --- |
| Learner Support Chatbot | Natural language interface for questions and tutoring. |
| Quiz Generator | AI generates new MCQs from content automatically. |
| Behavioral Analytics | Detects unusual patterns (e.g., low engagement, skipping) and suggests interventions. |
| Feedback Engine | Summarizes quiz feedback or recommends review content for learners. |

**5. Example Use Cases (AI)**

| **Scenario** | **LLM Response** |
| --- | --- |
| Learner skips 3 quizzes | "We've noticed you missed quizzes on Topic A and B. Would you like a quick review or practice test?" |
| Admin uploads a topic | "Suggest 3 MCQs and a short summary for this lesson." |
| Learner scores low on repeated quizzes | "It seems you're struggling with Data Structures. Try this simpler explanation." |

**6. Technical Plan**

| **Component** | **Stack** |
| --- | --- |
| Backend | Python (Flask or Django) |
| Frontend | HTML, JavaScript, Bootstrap or React |
| LLM Integration | DeepSeek via Ollama or Hugging Face (initially free) |
| Background Processing | Python daemon (thread or celery task queue) |
| Storage | Local or cloud-hosted database |
| Hosting (demo) | Localhost or free-tier services (e.g., Render, Vercel, Hugging Face Spaces) |

**7. Cost Overview (Demo Phase)**

| **Resource** | **Tool** | **Cost** |
| --- | --- | --- |
| LLM API | DeepSeek/Ollama | Free |
| Hosting | Local / Render (free tier) | Free |
| Storage | Mongodb | Free |
| Domain / SSL (if needed) | Optional | Low |

**8. Next Steps**

1. Finalize user tracking and cookie handling.
2. Set up LLM with prompt templates for feedback, tutoring, and analytics.
3. Run analytics daemon on learner behavior.
4. Display admin dashboard with AI insights.
5. Test LLM chatbot (tutor) on learner dashboard.

**1. Tasks & Timeline**

| **Task** | **Description** | **Duration (Weeks)** |
| --- | --- | --- |
| **Data Collection Setup** | Implement user behavior tracking (time on page, clicks, cookies) | 1 |
| **Analytics Engine Development** | Build background daemon for processing and summarizing data | 1.5 |
| **LLM Integration** | Connect LLM API, develop prompt templates for insights | 2 |
| **AI-Driven Insights & Dashboard** | Create admin and learner dashboards for AI-generated insights | 2 |
| **Testing & Refinements** | Test analytics, chatbot, feedback loops | 1 |
| **Deployment & Demo** | Deploy updated system and prepare demo | 0.5 |

**Total Estimated Time: ~8 weeks (2 months)**