

Developer Brief

This document outlines the vision, core features, tech expectations, timelines, and team structure — structured for clarity when onboarding a lead developer and assistant.



Developer Brief: CAPS-Aligned LMS Platform

Project Title:

NextGen CAPS-Aligned Digital Learning Platform

1. Project Overview

We are building an intelligent, enterprise-grade LMS (Learning Management System) tailored to the **South African CAPS Curriculum (Grades 1–12)**. The platform must include AI-powered content generation (activities, assessments, report cards, lesson plans), a secure digital library, live class integration, and a robust e-commerce model allowing educators to purchase ready-made resources.

2. Core Objectives

- Reduce paperwork/admin for educators
 - Improve content access through a built-in digital library
 - Offer AI-powered automation for lessons, tests, reports, etc.
 - Create a marketplace for downloadable educational content
 - Support multi-mode learning (video, live, text)
 - Align all academic content with **CAPS curriculum**
-

3. Key Features

LMS Core

- AI-generated activities, quizzes, tests, lesson plans, and reports (OpenAI API)
- Live classes via Zoom/Jitsi + recorded sessions
- Drip-fed, SCORM-compatible courses
- Text, video, and PDF-based courses
- Course analytics and performance tracking
- Homework/assignment system
- PDF viewer and secure file storage

Curriculum Support

- Full database of CAPS subjects (Grade 1–12)
- Each course tagged by phase, grade, and subject
- Automatic generation of CAPS-aligned content per grade

Digital Library

- Built-in digital library with publisher-uploaded PDFs
- Filter/search by subject, grade, publisher
- Secure download or view-only mode
- E-commerce-enabled

Marketplace & E-Commerce

- Educators can purchase bundles (e.g. term 1 assessments)
- Store for virtual + physical goods
- Variable commission per content creator
- Subscriptions, coupons, cashbacks
- 30+ payment gateways supported

Communication & Support

- Class forums + global community
- 1:1 and group messaging
- In-app video calls
- Calendar + reminders (Google Cal integrated)
- Tutor finder system

Administration

- Role-based access: admin, tutor, student, parent
- Built-in accounting and payout management
- Device/session restrictions for security
- GDPR-compliant & multilingual

4. Technical Requirements

Backend

- Laravel or Django (preferred for API control & performance)
- OpenAI/GPT integration for automation

- MySQL or PostgreSQL
- Secure REST API (for mobile expansion)

Frontend

- React (Next.js preferred) or Vue
- Fully responsive UI
- RTL support
- Drag-and-drop CMS home builder

DevOps

- AWS (S3 for storage, EC2, RDS, CloudFront)
- CI/CD pipeline
- Dockerized environment

Mobile

- Optional PWA or native app (Flutter)
- App-only mode toggle

5. 📁 Project Phases

Phase	Timeframe	Deliverables
Phase 1	1 month	Wireframes, system architecture, backend schema
Phase 2	2 months	Core LMS, Digital Library, User Roles
Phase 3	1.5 months	AI Integration (OpenAI), Test/Quiz Generator
Phase 4	1 month	Marketplace, Accounting, Final QA
Phase 5	Ongoing	App (if needed), Support tools, SaaS packaging

6. 👥 Team Structure

- **Lead Developer (Me):** Full-stack, responsible for backend and frontend architecture, feature integration.
 - **Assistant Developer (Freelancer/Junior):** Handles content entry, API testing, styling, plugin setup.
 - **PM (Optional):** Oversees task prioritization and quality assurance.
-

7. 💰 Budget & Tools

- Total Developer Budget: **±R297,000** (6-month development)
- Tools: GitHub, Trello

8. Non-Functional Requirements

- Fully secure login/authentication
 - Load tested for 10,000+ users
 - Scalable and modular codebase
 - Maintainable documentation
-

9. Next Steps

- Developer reviews brief and provides a milestone-based quote
- Begin with wireframes & schema
- Lock in tools and hosting setup