PROJECT_PLAN

Project Plan: Abdullah's Course Enrollment & Grade Management System

Overall Start Date: May 13, 2025
Estimated Full Completion: September 7, 2025 (17 weeks, 119 days)

PHASE 1 — PLAN FINALIZATION & INITIALIZATION (May 13–16)

Goals:

- Finalize architectural blueprint
- Set up GitHub repo and branch structure
- Establish dev/staging/prod environment separation
- Write `PROJECT_PLAN.pdf`, `GOALS_AND_ROADMAP.md`

Output:

- Branches: `dev`, `staging`, `main`
- Docs: `docs/dev`, `docs/staging`, `docs/prod`
- CI stubs for all 3 environments

PHASE 2 — DEV ENVIRONMENT BUILD (May 17 – June 7)

Tech Stack Updates:

- JHipster (microservices generator)
- Java 21 + Spring Boot (generated by JHipster)
- H2 (in-memory) or local PostgreSQL
- Maven or Gradle
- JUnit, Mockito
- React or Vue + Vite
- Axios
- GitHub Actions CI
- Docker + Docker Compose
- Kubernetes (minikube for local orchestration)
- Ingress Controller + NGINX
- Local shell scripts: `dev.sh`, `reset-db.sh`

Architecture Adjustments:

- Use JHipster to generate a **gateway app** and **multiple microservices**, each representing a domain:
  - `student-service`
  - `course-service`
  - `enrollment-service`
  - `grade-service`
- Each microservice will be a standalone Spring Boot app
- All services communicate via REST through the gateway app

Milestones:

- May 17–20: Scaffold gateway and microservices using JHipster
- May 21–24: Implement REST APIs in each service
- May 25–30: Integrate frontend with gateway
- May 31–June 4: Unit testing + frontend API communication
- June 5–7: Local orchestration polish using Docker Compose + Kubernetes (minikube), doc update

---

PHASE 3 — STAGING / QA ENVIRONMENT BUILD (June 8 – July 1)

Tech Stack:

- Dockerized JHipster apps (gateway + services)
- Dockerized frontend
- AWS RDS (PostgreSQL – staging instance)
- GitHub Actions staging CI/CD
- Playwright or Cypress for E2E
- Shell scripts: `startup.sh`, `deploy.sh`
- Terraform for EC2 + DB provisioning
- Vault or GitHub Secrets for config mgmt
- Kubernetes (EKS or k3s for staging)
- NGINX ingress setup for routing + domain masking

Milestones:

- June 8–14: Staging infra setup + Kubernetes manifests
- June 15–19: Build Docker images + configure CI/CD pipelines
- June 20–27: Write E2E tests, simulate user flows across services
- June 28–July 1: Validate full stack deployment and service-to-service comms

---

PHASE 4 — PRODUCTION ENVIRONMENT BUILD (July 2 – August 1)

Tech Stack:

- JHipster microservices (Spring Boot) containers on EC2
- React/Vue frontend hosted separately (EC2 or S3 + CloudFront)
- Amazon RDS (PostgreSQL – prod)
- Vault or AWS SSM (for secrets)
- Terraform for infra provisioning
- GitHub Actions CD pipelines
- TLS via Certbot + NGINX
- DNS: DuckDNS or custom domain
- Kubernetes (EKS or self-hosted) with production-grade Ingress

Milestones:

- July 2–10: Infra config, DNS masking, Kubernetes cluster live
- July 11–18: TLS setup and secrets injection
- July 19–26: Full deployment (gateway + services + frontend)
- July 27–Aug 1: Production walkthrough, metrics capture, doc update

---

PHASE 5 — DOCUMENTATION & QUALITY PASS (August 2 – August 16)

Outputs:

- `/docs/dev`, `/docs/staging`, `/docs/prod`
- `ERRORS_AND_LIMITATIONS.md`
- `EDGE_CASES.md`
- `SPRINT_LOG.md`, `RETROS.md`
- `infra-diagram.jpg/png` (hand-drawn and digital with microservice layout)

PHASE 6 — FINAL AUDIT & DEMO READINESS (August 17 – September 7)

- Review of all 3 environments
- Final deployment dry run: dev → staging → prod
- Showcase live app on GitHub, resume, and LinkedIn
- Optional: Add monitoring/logging (Prometheus/Grafana, Loki)
- Document load testing results (if time permits)

Overall Timeline Summary

Phase	Date Range	Duration
Plan & Setup	May 13 – May 16	4 days
Dev Env	May 17 – June 7	3 weeks
Staging QA	June 8 – July 1	3 weeks
Production	July 2 – August 1	1 month
Docs & Polishing	August 2 – August 16	2 weeks
Final Audit	August 17 – Sept 7	3 weeks

Total Duration: May 13 – Sept 7 = 17 weeks