

PROJECT_PLAN_V1.2

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 with 3 main branches: dev, staging, main
- Establish branch-based working tree logic (no subdirs like /dev or /prod)
- Write PROJECT_PLAN_V1.2, GOALS_AND_ROADMAP.md

Output:

- Branches: dev, staging, main
- Clean working tree: project content is branch-dependent
- Initial planning docs pushed to planning branch and merged to dev

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

Tech Stack:

- JHipster (for gateway + microservices)
- Java 21 + Spring Boot (per service)
- H2 (local dev DB)
- Maven or Gradle
- React or Vue + Vite (frontend)
- Axios (API requests)

- JUnit, Mockito (testing)
- Docker + Docker Compose
- GitHub Actions (CI/CD)
- Minikube (local k8s)
- Local scripts: dev.sh, reset-db.sh

Backend Microservices:

- student-service
- course-service
- enrollment-service
- grade-service
- gateway-app

Milestones:

- May 17–20: Scaffold gateway + services via JHipster
- May 21–24: Build REST APIs
- May 25–30: Connect frontend to gateway
- May 31–June 4: Write unit tests, Dockerize, test local orchestration
- June 5–7: Implement GitHub Actions CI: build, test, dockerize, dev-only deploy

PHASE 3 — STAGING ENVIRONMENT & QA CI/CD (June 8 – July 1)

Staging CI/CD Pipeline:

- Triggered by dev pipeline success or merge to staging
- Deploy via Terraform to EC2 (for app), RDS (PostgreSQL)

- Run integration + end-to-end tests (Cypress or Playwright)
- Inject secrets using Vault or GitHub Secrets
- Automatically assign blame if staging fails: QA config issue vs. Dev logic issue
- Slack/Email notifications with blame report + action items

Milestones:

- June 8–14: Infra setup + Terraform manifests
- June 15–19: GitHub Actions staging CI/CD
- June 20–27: End-to-end tests, smoke validation
- June 28–July 1: Automate error analysis + blame tagging

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

Production Pipeline:

- Triggered by manual approval, release tag, or merge to main
- Deploy containers via Kubernetes (EKS or self-hosted)
- Database: Amazon RDS PostgreSQL (prod instance)
- TLS via Certbot + NGINX
- Monitoring integration (optional: Prometheus, Loki, Grafana)
- Smoke tests + rollback on failure

Milestones:

- July 2–10: Infra provisioning, DNS, TLS
- July 11–18: Vault integration + secrets
- July 19–26: Full production deploy (gateway + services + frontend)

- July 27–Aug 1: Validate metrics, cleanup unused infra

PHASE 5 — DOCUMENTATION, POLISH & EDGE CASE COVERAGE (Aug 2 – Aug 16)

Outputs:

- /docs/dev, /docs/staging, /docs/prod (generated per branch)
- ERRORS_AND_LIMITATIONS.md
- EDGE_CASES.md
- SPRINT_LOG.md, RETROS.md
- Final infra diagram (hand-drawn + digital)

PHASE 6 — FINAL AUDIT & DEMO READINESS (Aug 17 – Sept 7)

Goals:

- Finalize live walk-through of all CI/CD pipelines
- Validate branch-to-environment link (dev → staging → main → prod)
- Showcase app live (GitHub, resume, LinkedIn)
- Optional: Add load testing results, runtime metrics, and GitHub Insights

Updated Overall Timeline

Phase	Date Range	Duration
Planning & Init	May 13 – May 16	4 days
Dev Build + CI	May 17 – June 7	3 weeks

Staging QA + CD	June 8 – July 1	3 weeks
Prod Infra + CD	July 2 – August 1	1 month
Docs + Edge Cases	August 2 – August 16	2 weeks
Final Audit + Demo	August 17 – Sept 7	3 weeks
Total	May 13 – Sept 7	17 weeks