

Rancangan Pembelajaran Mentor - Minggu 12: Fullstack Integration & Deployment

Berikut rencana pembelajaran untuk Minggu 12 yang berfokus pada Fullstack Integration & Deployment dengan intensitas yang lebih menantang, membangun di atas dasar yang telah dipelajari di minggu sebelumnya.

Senin

Materi (2 jam):

- Integrasi frontend (React) dan backend (Express.js)
- Komunikasi API

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Buat aplikasi React sederhana yang memanggil 3 endpoint API dari Express.js:
 - Implementasi GET/POST/PUT untuk proyek
 - Integrasi dengan backend
 - Testing endpoint
 - Uji komunikasi API dengan Postman
- **Proyek Kecil (4 jam):**
 - Integrasikan frontend React portofolio dengan backend Express.js:
 - Tampilkan data proyek via API
 - Tambah fitur CRUD proyek
 - Dokumentasikan integrasi API
 - Commit ke branch baru di Git
- **Review (1 jam):**
 - Uji komunikasi API di browser
 - Validasi di Postman
 - Periksa log commit

Tugas Harian:

- Submit kode integrasi
- Submit portofolio dengan fitur fullstack
- Submit log branch

Selasa

Materi (2 jam):

- Pengenalan deployment
- Hosting backend (Render)

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Deploy API Express.js ke Render:
 - Setup project untuk deployment
 - Deploy ke Render
 - Uji 2 endpoint dengan Postman
- **Proyek Kecil (4 jam):**
 - Deploy backend portofolio ke Render:
 - Konfigurasi environment variables
 - Pastikan endpoint proyek berfungsi
 - Buat flowchart alur deployment
 - Commit ke Git
- **Review (1 jam):**
 - Uji API di hosting
 - Validasi flowchart
 - Periksa log commit

Tugas Harian:

- Submit link API terdeploy
- Submit flowchart
- Submit portofolio backend

Rabu

Materi (2 jam):

- Deployment frontend (Netlify)
- CI/CD dasar

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Deploy aplikasi React ke Netlify:
 - Setup project untuk deployment
 - Konfigurasi CI/CD dengan GitHub
 - Uji auto-deploy
- **Proyek Kecil (4 jam):**
 - Deploy frontend portofolio ke Netlify:
 - Hubungkan dengan backend di Render
 - Konfigurasi environment variables
 - Testing integrasi fullstack
 - Commit ke branch baru
- **Review (1 jam):**
 - Uji aplikasi fullstack di hosting
 - Periksa CI/CD
 - Validasi log commit

Tugas Harian:

- Submit link frontend terdeploy
- Submit portofolio fullstack
- Submit log branch

Kamis

Materi (2 jam):

- Flowchart aplikasi fullstack
- Optimasi deployment

Belajar Mandiri (8 jam):

- **Latihan (3 jam):**
 - Buat flowchart aplikasi fullstack:
 - Alur frontend-backend
 - Optimasi endpoint dengan caching
 - Implementasi kompresi
- **Proyek Kecil (4 jam):**
 - Optimalkan portofolio fullstack:
 - Tambah loading state di React
 - Implementasi caching di Express
 - Dokumentasi optimasi
 - Push ke GitHub dengan pull request
- **Review (1 jam):**
 - Uji performa di hosting
 - Validasi flowchart
 - Periksa pull request

Tugas Harian:

- Submit flowchart
- Submit kode optimasi
- Submit link pull request
- Submit portofolio

Jumat (Ujian):

- **Ujian Praktik (5 jam):**
 - Buat aplikasi fullstack manajemen proyek dengan:
 - Frontend React
 - Backend Express.js
 - Integrasi database
 - Deploy aplikasi:
 - Frontend ke Netlify
 - Backend ke Render
 - Kelola kode dengan Git:
 - Buat branch

- 5 commit terstruktur
- Push ke GitHub
- Buat pull request
- Sertakan flowchart alur aplikasi
- **Persiapan Presentasi (3 jam):**
 - Validasi kode
 - Validasi flowchart
 - Validasi hosting
 - Siapkan slide untuk demo dan penjelasan
- **Sabtu:** Presentasi hasil ujian

Tugas Mingguan:

- Kembangkan portofolio fullstack dengan fitur lengkap:
 - CRUD proyek via API
 - UI interaktif
 - Deployment ke Netlify/Render
- Kelola dengan Git:
 - Branch
 - Merge
 - Pull request
- Unggah ke GitHub dengan README yang menjelaskan:
 - Arsitektur fullstack
 - Alur deployment
 - Cara instalasi dan penggunaan

Catatan:

- **Koneksi dengan Minggu Sebelumnya:** Portofolio diselesaikan sebagai aplikasi fullstack, mengintegrasikan React, Express.js, database, dan autentikasi dari minggu sebelumnya, dikelola dengan Git.
- **Intensitas Lebih Berat:** Latihan mencakup skenario kompleks (integrasi API, CI/CD, optimasi), proyek kecil menyelesaikan portofolio fullstack, dan pull request untuk kolaborasi.
- **Flowchart:** Digunakan untuk memvisualisasikan alur fullstack dan deployment.
- **Git:** Menggunakan branch, commit terstruktur, dan pull request untuk praktik profesional.