



Breakdown Harian - Minggu 11: Testing & Deployment



Tujuan Minggu Ini

- Memahami pentingnya testing dalam pengembangan aplikasi.
 - Mampu melakukan testing API menggunakan Postman atau tool serupa.
 - Mampu mengidentifikasi dan menangani validasi serta error handling dengan baik.
 - Memahami proses deployment aplikasi backend dan frontend ke platform cloud.
 - Mampu mengkonfigurasi CORS dan environment variables untuk produksi.
-



Hari 1 (Senin) - Testing API dengan Postman

Materi Inti (2 Jam)

- Pengenalan Postman: Fungsi dasar, membuat request (GET, POST, PUT, DELETE).
- Mengorganisir request dengan Collections dan Environments.
- Melakukan testing endpoint API yang sudah ada (misal: autentikasi, produk, keranjang).
- Memahami status kode HTTP (2xx, 4xx, 5xx) dan artinya.

Praktik Mandiri (8 Jam)

- Instal Postman (jika belum ada).
 - Buat Collection baru untuk proyek Anda.
 - Tambahkan semua endpoint API backend yang sudah Anda buat ke Collection tersebut.
 - Buat Environment untuk development (misal: `localhost:PORT`).
 - Lakukan pengujian manual untuk setiap endpoint: buat user, login, dapatkan token, akses protected routes, tambah produk, dll.
 - Pastikan setiap request mengembalikan response yang diharapkan dan status kode yang benar.
-



Hari 2 (Selasa) - Validasi & Error Handling

Materi Inti (2 Jam)

- Pentingnya validasi input di backend untuk keamanan dan integritas data.
- Menggunakan library validasi (misal: `express-validator` untuk Express.js).
- Strategi error handling yang efektif di backend (middleware error handling).
- Mengirim pesan error yang informatif namun tidak membocorkan detail sensitif ke frontend.

Praktik Mandiri (8 Jam)

- Identifikasi endpoint-endpoint di backend yang memerlukan validasi input (misal: register, login, tambah produk).
- Implementasikan validasi menggunakan library pilihan Anda (misal: cek format email, panjang password, tipe data).

- Buat middleware error handling global di Express.js untuk menangani error yang tidak tertangkap.
 - Uji coba validasi dengan mengirim data yang salah dari Postman dan pastikan pesan error yang sesuai dikembalikan.
 - Pastikan error yang terjadi di backend ditangani dengan baik dan tidak menyebabkan aplikasi crash.
-



Hari 3 (Rabu) - Deploy Backend (Railway/Render)

Materi Inti (2 Jam)

- Konsep Platform as a Service (PaaS) untuk deployment backend.
- Pengenalan Railway/Render: Fitur, kelebihan, dan cara kerjanya.
- Persiapan kode backend untuk deployment (misal: konfigurasi port, database connection string).
- Proses deployment dari GitHub repository.

Praktik Mandiri (8 Jam)

- Buat akun di Railway atau Render.
 - Siapkan repository GitHub backend Anda.
 - Pastikan semua environment variables (misal: `DATABASE_URL`, `JWT_SECRET`) dikonfigurasi dengan benar untuk lingkungan produksi.
 - Ikuti panduan deployment Railway/Render untuk menghubungkan repository Anda dan deploy aplikasi backend.
 - Konfigurasi database PostgreSQL di platform cloud (jika belum ada) dan hubungkan dengan aplikasi backend Anda.
 - Setelah deploy, uji coba endpoint API yang sudah di-deploy menggunakan Postman.
-



Hari 4 (Kamis) - Deploy Frontend (Vercel/Netlify) & CORS

Materi Inti (2 Jam)

- Konsep Static Site Hosting untuk deployment frontend.
- Pengenalan Vercel/Netlify: Fitur, kelebihan, dan cara kerjanya.
- Persiapan kode frontend untuk deployment (misal: konfigurasi `API_URL`).
- Mengatasi masalah CORS (Cross-Origin Resource Sharing) antara frontend dan backend.

Praktik Mandiri (8 Jam)

- Buat akun di Vercel atau Netlify.
- Siapkan repository GitHub frontend Anda.
- Konfigurasi environment variable di frontend untuk `REACT_APP_API_URL` yang menunjuk ke URL backend yang sudah di-deploy.
- Ikuti panduan deployment Vercel/Netlify untuk menghubungkan repository Anda dan deploy aplikasi frontend.
- Jika terjadi masalah CORS, implementasikan middleware CORS di backend Anda (misal: `cors` package di Express.js) dan konfigurasikan `origin` yang sesuai.
- Setelah deploy, uji coba aplikasi frontend yang sudah di-deploy di browser dan pastikan semua fitur berfungsi dengan baik.



Hari 5 (Jum'at) - Review & Tugas Mingguan

Materi Inti (2 Jam)

- Review menyeluruh proses testing dan deployment.
- Membahas strategi monitoring dan maintenance aplikasi setelah deployment.
- Persiapan Tugas Mingguan.

Tugas Mingguan (Waktu Pengerjaan: Sisa Minggu + Akhir Pekan)

Tema Tugas: Testing, Deployment, dan Finalisasi Aplikasi E-commerce.

Persyaratan Fungsional:

1. **Testing API:** Lakukan pengujian menyeluruh semua endpoint API backend menggunakan Postman atau tool serupa.
2. **Validasi & Error Handling:** Pastikan semua input divalidasi dengan benar di backend dan error ditangani secara graceful.
3. **Deployment Backend:** Deploy aplikasi backend Anda ke Railway atau Render.
4. **Deployment Frontend:** Deploy aplikasi frontend Anda ke Vercel atau Netlify.
5. **Konfigurasi CORS:** Pastikan tidak ada masalah CORS antara frontend dan backend yang sudah di-deploy.
6. **Environment Variables:** Semua konfigurasi sensitif (API Keys, database URL) harus menggunakan environment variables.
7. **Dokumentasi (README.md):** Perbarui file **README.md** dengan instruksi deployment dan cara menguji aplikasi yang sudah di-deploy.

Fitur Opsional (Nilai Tambah):

- Implementasi unit/integration testing menggunakan Jest/React Testing Library.
- Menambahkan CI/CD pipeline sederhana (misal: GitHub Actions) untuk otomatisasi deployment.
- Konfigurasi custom domain untuk aplikasi yang sudah di-deploy.
- Menambahkan monitoring tool (misal: Sentry, New Relic) untuk memantau performa aplikasi.

Output Tugas:

- Kode sumber aplikasi React dan backend yang sudah siap deploy.
- Aplikasi backend dan frontend yang sudah berhasil di-deploy dan dapat diakses publik.
- File **README.md** yang diperbarui.
- Push semua perubahan ke repositori GitHub pribadi Anda.



Sesi Presentasi (Waktu disesuaikan)

Siapkan diri Anda untuk sesi presentasi singkat di awal minggu berikutnya. Fokus presentasi adalah:

- **Demo Aplikasi:** Tunjukkan aplikasi e-commerce yang sudah di-deploy dan berfungsi penuh.
- **Penjelasan Proses Deployment:** Jelaskan langkah-langkah deployment yang Anda lakukan untuk backend dan frontend.

- **Penjelasan Testing & Error Handling:** Jelaskan bagaimana Anda memastikan kualitas kode dan menangani error.
- **Tantangan & Solusi:** Ceritakan tantangan yang Anda hadapi selama proses testing dan deployment, serta bagaimana Anda menyelesaikannya.



Tips Belajar Tambahan

- **Automated Testing:** Pertimbangkan untuk mulai belajar automated testing untuk meningkatkan kualitas kode jangka panjang.
- **CI/CD:** Pahami konsep CI/CD untuk mengotomatisasi proses build, test, dan deploy.
- **Scalability & Security:** Selalu pikirkan aspek skalabilitas dan keamanan saat mendesain dan mendeploy aplikasi.

Minggu ini adalah puncak dari perjalanan pengembangan aplikasi Anda: memastikan aplikasi berfungsi dengan baik melalui testing dan membuatnya dapat diakses publik melalui deployment. Fokus pada detail konfigurasi dan penanganan masalah yang mungkin muncul di lingkungan produksi. Selamat belajar dan mengerjakan tugas!