# Analisis Kebutuhan Perencanaan Proyek Web: Cuci sepatu

## 1. Ringkasan Eksekutif

* Deskripsi: Pengembangan platform system web Cuci Sepatu.
* Tujuan: Meningkatkan aksesibilitas dari klien cuci Sepatu ,melalui web secara online dan efektivitas pelayanan
* Manfaat: Peningkatan engagement mahasiswa, perluasan jangkauan konsumen, dan efisiensi operasional

## 2. Latar Belakang Proyek

* Konteks: Meningkatnya permintaan untuk memudahkan orderan cuci Sepatu secara online dan melihat layanan yang di sediakan serta melihat laporan penjualan yang lebih fleksibel
* Masalah: Keterbatasan untuk mengelola order dan melihat catatan konsumen.
* Peluang: Memperluas jangkauan konsumen dan meningkatkan pengalaman clien

## 3. Tujuan dan Sasaran

* Tujuan Bisnis: Meningkatkan pengalaman pengguna untuk order secara online
* Tujuan Teknis: Membuat platform yang dapat menunjukkan laporan penjualan ,pilihan pelayanan yang bisa di order pengguna
* KPI: Tingkat penyelesaian kursus, skor kepuasan pengguna, jumlah pendaftaran baru

## 4. Ruang Lingkup Proyek

* Fitur Utama: Manajemen pesenan,manajemen laporan,order secara online
* Batasan: Tidak termasuk integrasi dengan keuangan klien laporan hanya menampilkan Riwayat pengguna yang mengorder cuci Sepatu
* Asumsi: pengguna memiliki akses internet

## 5. Analisi Stakeholder

* Stakeholder: Mahasiswa, Klien,konsumen
* Peran: Konsumen sebagai pengguna utama, Klien sebagai pembuat konten
* Ekspektasi: Kemudahan penggunaan, keandalan sistem, dukungan teknis yang responsif

## 6. Kebutuhan Fungsional

## Fitur: Pendaftaran Kursus

## User Story:

## "Sebagai client saya ingin dapat mendaftar ke kursus yang saya minati sehingga saya bisa mengikuti pelajaran dan mendapatkan materi terkait."

## Kriteria Penerimaan:

## Mahasiswa dapat melihat daftar yang tersedia.

## Mahasiswa dapat mendaftar ke kursus dengan mengklik tombol "Daftar".

## Setelah mendaftar, mahasiswa menerima konfirmasi pendaftaran melalui email dan di dashboard mereka.

## Prioritas: Must have (login)

## Fitur: Upload/Download Materi

## User Story:

## "Sebagai dosen, saya ingin dapat mengunggah materi kuliah ke platform sehingga mahasiswa bisa mengakses dan mengunduh materi tersebut untuk persiapan dan belajar di luar kelas."

## Kriteria Penerimaan:

## Dosen dapat mengunggah materi dalam format PDF, DOCX, atau video.

## Mahasiswa dapat mengunduh materi yang telah diunggah.

## Sistem harus mendukung versi kontrol untuk materi yang diunggah.

## Prioritas: Must have (login)(dosen)

## 7. Kebutuhan Non-Fungsional

* Performa: Waktu loading halaman < 3 detik
* Keamanan: Enkripsi end-to-end untuk data sensitif
* Skalabilitas: Mampu menangani peningkatan 50% pengguna tanpa degradasi performa
* Usability: Antarmuka yang intuitif, aksesibel untuk pengguna dengan disabilitas

## 8. Arsitektur Sistem

* Diagram: [Gambar arsitektur sistem]
* Komponen: Frontend (React), Backend (Node.js), Database (PostgreSQL), CDN untuk video
* Integrasi: SSO dengan sistem universitas yang ada

## 9. Desain User Interface

* Wireframes: [Mockup dashboard mahasiswa, halaman kursus]
* Pedoman: Palet warna universitas, font sans-serif, ikon material design
* Responsivitas: Desain responsif untuk desktop, tablet, dan mobile

## 10. Teknologi dan Stack

* Frontend: React.js dengan Next.js
* Backend: Node.js dengan Express
* Database: PostgreSQL
* Server: AWS EC2
* Tools: Git, JIRA, Figma

## 11. Kebutuhan Data

* Model Data: Entitas utama - User, Course, Enrollment, Assignment
* Sumber Data: Sistem informasi mahasiswa yang ada, input manual dosen
* Migrasi: Migrasi data kursus dari sistem lama

## 12. Keamanan dan Privasi

* Keamanan: Implementasi OAuth 2.0, HTTPS, validasi input ketat
* Compliance: GDPR untuk mahasiswa internasional
* Akses: Role-based access control (RBAC)

## 13. Pengujian dan Quality Assurance

* Strategi: Unit testing, integration testing, user acceptance testing
* Jenis Pengujian: Functional testing, performance testing, security testing
* Kriteria: 95% code coverage, zero high-severity bugs

## 14. Deployment dan Maintenance

* Deployment: CI/CD pipeline menggunakan Jenkins
* Hosting: AWS dengan auto-scaling
* Maintenance: Pembaruan keamanan bulanan, backup harian

## 15. Timeline dan Milestones

* Timeline: 6 bulan pengembangan
* Milestones: M1 (Desain UI/UX), M2 (Core Features), M3 (Testing), M4 (Launch)
* Deliverables: Prototype (Bulan 2), Beta Version (Bulan 4), Final Release (Bulan 6)

## 16. Anggaran dan Sumber Daya

* Anggaran: $500,000 total
* Sumber Daya: 4 developer, 2 designer, 1 project manager, 1 QA specialist
* Lisensi: AWS, Zoom integration, design tools

## 17. Risiko dan Mitigasi

* Risiko: Overloading server saat ujian online
* Mitigasi: Implementasi load balancing dan caching

## 18. Metrik Kesuksesan

* “Done”: Semua fitur utama berfungsi, performance test passed, UAT completed
* Metrik: 90% tingkat kepuasan pengguna, peningkatan 30% dalam engagement mahasiswa

## 19. Dokumentasi

* Teknis: API documentation, system architecture document
* Pengguna: User manual, video tutorials
* Proses: Dokumentasi diperbarui setiap sprint

## 20. Penutup

* Ringkasan: Platform e-learning yang komprehensif dan scalable
* Langkah Selanjutnya: Kick-off meeting dengan semua stakeholder, mulai sprint pertama

## Lampiran

* Glossary: LMS (Learning Management System), SPA (Single Page Application)
* Referensi: Standar IEEE untuk Software Requirements Specification
* Hasil Survei: [Ringkasan hasil survei kebutuhan mahasiswa]