# Analisis Kebutuhan Perencanaan Proyek Web: Restoran Mie Ayam Gapuro

## 1. Ringkasan Eksekutif

## Deskripsi: Pengembangan website restoran untuk Mie Ayam Gapuro

## Tujuan: Meningkatkan visibilitas online, mempermudah pemesanan, dan memperluas jangkauan pelanggan

## Manfaat: Peningkatan engagement pelanggan, kemudahan akses menu dan layanan, serta otomatisasi pemesanan dan pembayaran

## 2. Latar Belakang Proyek

## Konteks: Banyak pelanggan mencari restoran dan menu secara online sebelum mengunjungi tempat makan, sehingga memiliki website dapat meningkatkan peluang kunjungan.

## Masalah: Restoran Mie Ayam Gapuro saat ini belum memiliki kehadiran digital yang memadai untuk mempromosikan produk dan layanan secara efektif

## Peluang: Memperkenalkan fitur sederhana seperti menu online dan formulir pemesanan dapat meningkatkan daya tarik pelanggan, terutama di era digital ini.

## 3. Tujuan dan Sasaran

* Tujuan Bisnis: Membantu restoran Mie Ayam Gapuro menarik lebih banyak pelanggan lokal dan meningkatkan penjualan melalui pemesanan online
* Tujuan Teknis: Membuat website sederhana yang dapat menampilkan menu, kontak, lokasi, dan formulir pemesanan online yang mudah digunakan.
* KPI: Jumlah kunjungan website, jumlah pemesanan melalui website dan peningkatan kunjungan ke restoran.

## 4. Ruang Lingkup Proyek

* Fitur Utama: Menampilkan Menu, Menampilkan Paket Hemat, Melakukan Pemesanan Online dan melakukan Transaksi, Menerima Pesanan, menampilkan profil restoran.
* Batasan: Fitur-fitur tambahan seperti program membership atau rekomendasi produk otomatis tidak disertakan
* Asumsi: Pengguna memiliki akses internet, restoran siap menerima pesanan online.

## 5. Analisis Stakeholder

* Stakeholder: Pelanggan, pemilik restoran, pengembang website
* **Peran:** Pelanggan sebagai pengguna utama, Pemilik Restoran sebagai pengelola konten, Pengembang Website sebagai penyedia dan pemelihara sistem.
* Ekspektasi: Kemudahan melihat menu dan paket hemat, Manajemen menu dan transaksi mudah, membuat system stabil dan user-friendly

## 6. Kebutuhan Fungsional

## Fitur: Mengelola Menu dan Paket Hemat

## User Story:

## "Sebagai Pemilik Restoran, saya ingin mengelola Menu pada halaman menu dan paket hemat."

## Kriteria Penerimaan:

## Pemilik restoran dapat menababhan, mengahapus dan mengedit menu atau paket hemat

## Prioritas: Must have (login)

## Fitur: Menampilkan Menu

## User Story:

## "Sebagai Pelanggan, saya ingin melihat menu mie ayam yang tersedia di website sehingga saya bisa memilih makanan yang ingin saya pesan."

## Kriteria Penerimaan:

## Pelanggan dapat melihat menu lengkap yang di tawarkan restoran

## Menu harus mencangkup Deskripsi dan harga setiap item

## Prioritas: Must have (login)

## Fitur: Menampilkan Paket Hemat

## User Story:

## "Sebagai pelanggan, saya ingin melihat paket hemat yang ditawarkan restoran untuk mendapatkan penawaran terbaik."

## Kriteria Penerimaan:

## Pelanggan dapat melihat daftar paket hemat dengan deskripsi dan harga.

## Paket hemat harus ditampilkan secara jelas di website.

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

## Fitur: Melakukan Pemesanan Online

## User Story:

## "Sebagai pelanggan, saya ingin memesan mie ayam dan melakukan pemesanan secara online sehingga saya bisa memesan tanpa harus datang ke restoran."

## Kriteria Penerimaan:

## Pelanggan dapat memilih item dari menu dan paket hemat untuk dimasukkan ke dalam keranjang.

## Pelanggan dapat melakukan pembayaran melalui transfer manual atau COD.

## Setelah transaksi selesai, pelanggan menerima konfirmasi pesanan melalui email atau di dashboard mereka.

## Fitur: Melakukan Melakukan Transaksi

## User Story:

## "Sebagai pelanggan, saya ingin memesan mie ayam dan melakukan transaksi secara online sehingga saya bisa memesan tanpa harus datang ke restoran."

## Kriteria Penerimaan:

## Pelanggan dapat melakukan pembayaran melalui transfer manual atau COD.

## Setelah transaksi selesai, pelanggan menerima konfirmasi pesanan melalui email atau di dashboard mereka.

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

## Fitur: Menerima Pemesanan Online dan Melakukan Transaksi

## User Story:

## "Sebagai pemilik restoran, saya ingin menerima dan mengelola pesanan yang masuk melalui website sehingga saya bisa memproses dan menyiapkan pesanan dengan tepat."

## Kriteria Penerimaan:

## Pemilik restoran dapat melihat daftar pesanan yang masuk.

## Pemilik restoran dapat memperbarui status pesanan (diproses, selesai, dll.).

## Prioritas: Must have (login)(Pemilik restoran)

## Fitur: Menampilkan Profil Restoran

## User Story:

## " Sebagai pelanggan, saya ingin melihat profil restoran yang mencakup informasi seperti alamat, nomor telepon, dan jam operasional sehingga saya bisa mendapatkan informasi kontak dan lokasi restoran."

## Kriteria Penerimaan:

## Profil restoran ditampilkan dengan jelas, mencakup alamat lengkap, nomor telepon, dan jam operasional.

## Prioritas: Must have (login)

## 7. Kebutuhan Non-Fungsional

* Performa: Halaman website harus memuat dalam waktu kurang dari 3 detik agar pengguna tidak menunggu lama.
* Keamanan: Data seperti informasi pribadi dan transaksi harus dilindungi dengan enkripsi untuk menjaga kerahasiaan.
* Skalabilitas: Mampu menangani peningkatan 50% pengguna tanpa degradasi performa
* Usability: Desain website harus mudah dipahami dan digunakan, serta dapat diakses oleh semua orang.

## 8. Arsitektur Sistem

* Diagram: [Gambar arsitektur sistem]
* Komponen: Frontend (Blade template( Laravel)), Backend (Laravel (PHP Framework)), Database (MySQL), Server (Apache)
* Integrasi: SSO dengan sistem universitas yang ada

## 9. Desain User Interface

* Wireframes: [Mockup dashboard mahasiswa, halaman kursus]
* Pedoman: Gunakan palet warna yang cocok dengan branding restoran, Font modern yang mudah dibaca (Google Fonts: Poppins, Roboto).
* Responsivitas: Website akan responsif, mendukung berbagai ukuran layar (desktop, tablet, dan mobile)

## 10. Teknologi dan Stack

* Frontend: Blade Template (Laravel)
* Backend: Laravel (PHP Framework)
* Database: MySQL
* Server: AWS EC2
* Tools: Git, Trello/ JIRA, Figma

## 11. Kebutuhan Data

* Model Data: Entitas utama – User(Pelanggan, Menu , Pesanan, Transaksi
* Sumber Data: Input manual dari pemilik restoran dan pelanggan
* Migrasi: Data menu dan pesanan dari sistem manual bisa dimigrasi ke database.

## 12. Keamanan dan Privasi

* Keamanan: Implementasi OAuth 2.0, HTTPS, validasi input ketat
* Compliance: Mengikuti regulasi terkait privasi pelanggan local
* Akses: Role-based access control (RBAC); pemilik restoran memiliki hak akses penuh untuk mengelola menu, pesanan, dan transaksi.

## 13. Pengujian dan Quality Assurance

* Strategi: Unit testing, integration testing, user acceptance testing
* Jenis Pengujian: Functional testing, performance testing, security testing
* Kriteria: 90% code coverage, zero high-severity bugs

## 14. Deployment dan Maintenance

* Deployment: CI/CD pipeline menggunakan GitHub Actions/Jenkins
* Hosting: AWS dengan auto-scaling
* Maintenance: Pembaruan sistem dan keamanan dilakukan setiap bulan, backup data harian otomatis.

## 15. Timeline dan Milestones

* Timeline: 5 bulan pengembangan
* Milestones: M1 (Desain UI/UX), M2 (Fitur Inti), M3 (Pengujian dan Optimasi), M4 (Launch Web)
* Deliverables: Prototype (Bulan 2), Beta Version (Bulan 4), Final Release (Bulan 6)

## 16. Anggaran dan Sumber Daya

* Anggaran: Rp.25.000.000,00
* Sumber Daya: 3 developer, 1 designer, 1 project manager, 1 QA specialist
* Lisensi: AWS, Zoom integration, design tools

## 17. Risiko dan Mitigasi

* Risiko: Overload server saat peak hours.
* Mitigasi: Menggunakan caching dan load balancing jika diperlukan, serta CDN untuk gambar.

## 18. Metrik Kesuksesan

* “Done”: Semua fitur utama (menu, pesanan, pembayaran) berfungsi dengan baik dan telah lulus UAT.
* Metrik: 80% tingkat kepuasan pengguna, peningkatan 20% peningkatan pesanan online dalam 3 bulan pertama setelah peluncuran.

## 19. Dokumentasi

* Teknis: Dokumentasi API dan arsitektur sistem.
* Pengguna: Panduan untuk pemilik restoran tentang cara mengelola website dan pesanan.
* Proses: Dokumentasi diperbarui setiap sprint

## 20. Penutup

* Ringkasan: Website ini akan membantu Restoran Mie Ayam Gapuro meningkatkan kehadiran online, mempermudah pelanggan dalam melakukan pemesanan, dan meningkatkan efisiensi pengelolaan pesanan.
* Langkah Selanjutnya: Kick-off meeting dengan pemilik restoran dan developer untuk memulai sprint pertama.

## Lampiran

* Glossary: CRUD (Create, Read, Update, Delete), RBAC (Role-Based Access Control)
* Referensi: Dokumentasi Laravel, Standar Keamanan OWASP
* Hasil Survei: [Ringkasan hasil survei kebutuhan mahasiswa]