

## 2. Pemilihan Arsitektur Aplikasi

### 2.1. Tujuan Pemilihan Arsitektur

- Web harus mudah digunakan baik oleh pemilik bisnis untuk mengelola inventaris, pemesanan, dan keuangan, maupun oleh pelanggan untuk melakukan pemesanan dan pembayaran.
- Web harus menyediakan fitur-fitur yang diperlukan untuk bisnis rental camp dengan Keamanan data, Kinerja yang baik, skalabilitas, integrasi serta pemeliharaan dalam web

### 2.2. Pilihan Arsitektur

#### 2.2.1. Model-View-Controller (MVC)

- **Deskripsi:** arsitektur perangkat lunak yang populer digunakan dalam pengembangan web. Dengan membaginya menjadi tiga bagian utama : Model: (Mengelola data dan logika bisnis), View (Menampilkan antarmuka pengguna user interface)
- **Keuntungan:** Perubahan pada satu komponen (model, view, atau controller) cenderung tidak berdampak besar pada komponen lainnya, sehingga memudahkan dalam melakukan perubahan dan perbaikan.
- **Kekurangan:** MVC mungkin tidak cocok untuk aplikasi yang sangat kecil atau sederhana, di mana kompleksitas tambahan yang diberikan oleh MVC mungkin tidak sepadan dengan manfaatnya.

#### 2.2.2. Microservices

- **Deskripsi:** pendekatan pengembangan perangkat lunak di mana aplikasi dipecah menjadi layanan-layanan kecil yang independen. Setiap layanan memiliki tanggung jawab yang spesifik dan berkomunikasi dengan layanan lain melalui API. Dengan menggunakan layanan **katalog, pemesanan, pengguna, dan pembayaran**
- **Keuntungan:** Setiap layanan dapat diskalakan secara independen sesuai kebutuhan. Misalnya, jika layanan pemesanan mengalami lonjakan permintaan, hanya layanan tersebut yang perlu ditingkatkan kapasitasnya.
- **Kekurangan:** Mengelola banyak layanan kecil membutuhkan infrastruktur yang lebih kompleks dan keahlian yang lebih tinggi.

#### 2.2.3. Single Page Application (SPA)

- **Deskripsi:** adalah jenis aplikasi web yang memuat satu halaman HTML tunggal dan kemudian memperbarui konten secara dinamis tanpa memuat ulang seluruh halaman. Hal ini memberikan pengalaman pengguna yang lebih mulus dan responsif.
- **Keuntungan:** Navigasi antara halaman terasa lebih mulus, tanpa adanya jeda waktu yang signifikan saat memuat ulang halaman. Dengan pemorfa yang lebih cepat
- **Kekurangan:** Mesin pencari mungkin kesulitan mengindeks konten dinamis dalam SPA. Teknik seperti server-side rendering atau pre-rendering dikarenakan pemuatan yang bisa lebih lama

### 2.3. Rekomendasi Arsitektur

- **Rekomendasi:** Berdasarkan analisis kebutuhan, proyek ini akan direkomendasikan untuk menggunakan SPA (Single Page Application). Hal ini dikarenakan SPA memberikan navigasi yang halus dan interaktif yang fokus pada fitur-fitur seperti pemesanan online yang seamless dengan performa yang baik. Untuk

## 2. Desain Struktur Database Awal

### 2.1. Tujuan Desain Struktur Database

- Mengorganisir data secara efisien: Menyimpan informasi mengenai peralatan camping, pelanggan, transaksi penyewaan, dan data-data terkait lainnya dengan struktur yang jelas dan terstruktur.
- Mempermudah pengelolaan data: Memungkinkan pengelola situs untuk dengan mudah menambahkan, mengubah, menghapus, dan mencari data yang dibutuhkan.
- Menunjang pengembangan fitur-fitur baru: Menyediakan fondasi yang kuat untuk pengembangan fitur-fitur tambahan pada situs rental camp di masa mendatang, seperti sistem pelaporan, analisis data, dan integrasi dengan sistem pembayaran online.

### 2.2. Desain Struktur Database

#### 2.2.1. Diagram Entity-Relationship (ER)

- **Deskripsi:** Untuk mempermudah visualisasi relasi antar tabel, dapat membuat diagram ER. Diagram ini akan memberikan gambaran yang lebih jelas tentang struktur database.

#### 2.2.2. Tabel-Tabel Utama

#### 1. Kategori Peralatan:

- **Tabel Kategori:**
  - id\_kategori (primary key)
  - nama\_kategori (misal: tenda, sleeping bag, masak)

#### 2. Kondisi Peralatan:

- **Kolom Kondisi:**
  - baik, rusak ringan, rusak berat

#### 3. Promosi dan Diskon:

- **Tabel Promo:**
  - id\_promo (primary key)
  - nama\_promo
  - persentase\_diskon
  - tanggal\_mulai
  - tanggal\_berakhir

#### 4. Inventaris:

- **Tabel Inventaris:**
  - id\_inventaris (primary key)
  - id\_equipment (foreign key)
  - jumlah\_stok

## 5. Lokasi Penyimpanan:

- **Tabel Lokasi:**
  - id\_lokasi (primary key)
  - nama\_lokasi
  - alamat

## 6. Pengguna:

- **Tabel Pengguna:**
  - id\_pengguna (primary key)
  - username
  - password
  - role (admin, karyawan)

## 2.3. Rekomendasi Struktur Database

- **Rekomendasi:** Struktur database diusulkan dengan merancang integrasi data dan efisiensi operasi dari Tabel-Tabel Utama yang telah diidentifikasi dan menggunakan hubungan antar tabel diatur untuk mendukung fungsionalitas yang diinginkan, termasuk juga pengolahan data dari peralatan dan juga pengguna/admin

## 3. Pemilihan Teknologi Frontend dan Backend

### 3.1. Tujuan Pemilihan Teknologi

- Pemilihan yang bersifat stabil dan dapat digunakan dengan mudah untuk mendukung sistem serta memenuhi kebutuhan para konsumen dalam pemesanan dan perentalan

### 3.2. Pemilihan Teknologi Frontend

#### 3.2.1. Framework/Library

- **Pilihan:** React, Vue.js, Angular
- **Deskripsi:** Pemilihan teknologi frontend yang tepat sangat krusial dalam membangun situs rental camp yang user-friendly dan performant.
- **Kriteria Pemilihan:** Dikarenakan performa, komunitas, kemudahan integrasi pada backend, dan dokumen

#### 3.2.2. Teknologi Tambahan

- **State Management:** Redux, Vuex, Pinia
- **Routing:** React Router, Vue Router
- **Styling:** Css Module
- **Testing:** Jest, Cypress
- **Build Tools:** Webpack, Parcel

- **Ui Libraries:** Material UI, Ant Design, Semantic UI
- **Performance Optimization:** Code Splitting, Lazy Load, Minification

### 3.3. Pemilihan Teknologi Backend

#### 3.3.1. Bahasa Pemrograman dan Framework

- **Pilihan:** PHP dengan Laravel
- **Deskripsi:** Kombinasi dinamis ini akan menjadi jantung dari situs rental camp Anda. Laravel, sebagai framework PHP yang populer, menawarkan struktur yang kokoh, kemudahan pengembangan, dan fitur-fitur canggih yang sangat cocok untuk membangun aplikasi web yang kompleks seperti situs rental camp
- **Kriteria Pemilihan:** Kemudahan transaksi yang dimana Laravel menyediakan berbagai tools dan integrasi dengan gateway pembayaran populer, sehingga proses transaksi menjadi lebih lancar dan aman.

#### 3.3.2. Basis Data

- **Pilihan:** MySQL
- **Deskripsi:** MySQL adalah pilihan yang populer untuk aplikasi web karena kemudahan penggunaannya, kinerja yang baik, dan dukungan yang luas. Dengan MySQL, Anda dapat menyimpan berbagai jenis data terkait rental camp, seperti informasi pelanggan, detail peralatan, dan riwayat transaksi.
- **Kriteria Pemilihan:** Model data relasional yang sangat cocok untuk menyimpan data yang saling terkait, seperti hubungan antara pelanggan dan pesanan.

### 3.4. Rekomendasi Teknologi

- **Modern dan responsif:** React akan memberikan tampilan yang menarik dan pengalaman pengguna yang seamless di berbagai perangkat.
- **Kuat dan andal:** Laravel dan MySQL akan memastikan bahwa situs Anda dapat menangani beban kerja yang tinggi dan tetap stabil.
- **Mudah dikembangkan dan dipelihara:** Struktur yang baik dan komunitas yang besar akan memudahkan Anda dalam mengembangkan dan memperbarui situs Anda.
- **Terintegrasi dengan baik:** React dan Laravel dapat diintegrasikan dengan mudah melalui API, sehingga Anda dapat membangun aplikasi yang kompleks dan kaya fitur.

