**1. Pemilihan Arsitektur Aplikasi**

**1.1. Tujuan Pemilihan Arsitektur**

* Menyediakan dasar yang solid untuk pengembangan, pemeliharaan, dan skalabilitas aplikasi perpustakaan e-book.
* Memastikan bahwa arsitektur yang dipilih dapat memenuhi kebutuhan fungsional dan non-fungsional seperti peminjaman buku, pencarian, dan pengelolaan pengguna.

**1.2. Pilihan Arsitektur**

**1.2.1. Model-View-Controller (MVC) - Laravel**

* **Deskripsi**: Laravel menggunakan arsitektur MVC yang memisahkan aplikasi menjadi *Model* (logika data), *View* (tampilan), dan *Controller* (pengelolaan input pengguna dan komunikasi antara model dan view).
* **Keuntungan**:
  + Memudahkan pengembangan dan pemeliharaan karena pemisahan logika bisnis dan tampilan.
  + Memanfaatkan berbagai fitur bawaan Laravel seperti routing, autentikasi, dan Eloquent ORM untuk pengelolaan database.
  + Integrasi yang mudah dengan berbagai package Laravel yang bisa mempercepat pengembangan.
* **Kekurangan**:
* Memerlukan pembelajaran bagi pengembang yang baru mengenal Laravel.
* Performa mungkin tidak sebaik arsitektur microservice untuk aplikasi skala sangat besar.

**1.3. Rekomendasi Arsitektur**

* **Rekomendasi**: Laravel dengan arsitektur MVC direkomendasikan karena mampu mengelola logika bisnis perpustakaan e-book dengan baik, mendukung pengembangan cepat, dan mendukung pemisahan tugas dalam tim.

**2. Desain Struktur Database Awal**

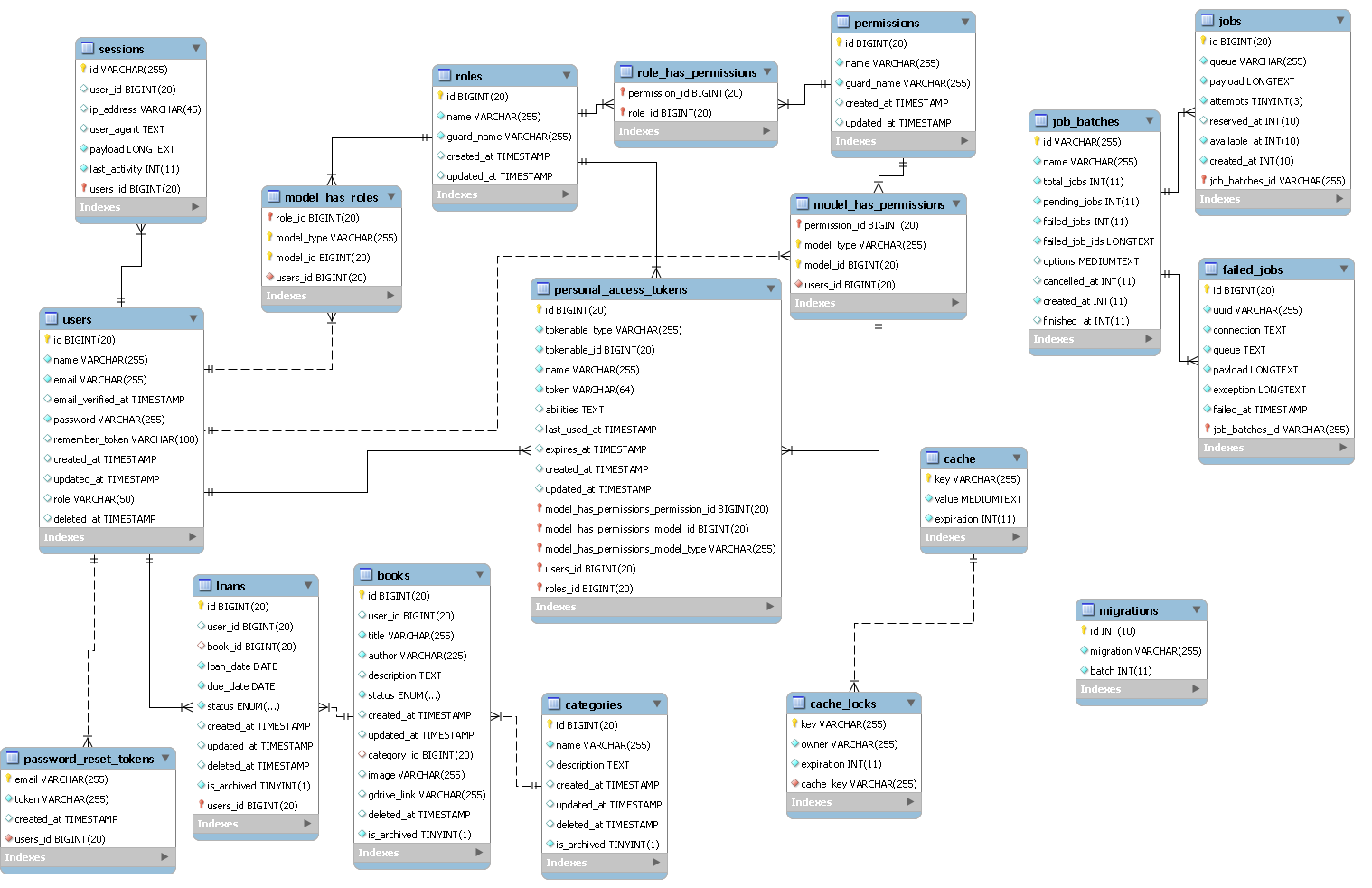
**2.1. Tujuan Desain Struktur Database**

* Menyediakan penyimpanan data yang efisien dan mudah diakses oleh aplikasi perpustakaan e-book.
* Mengoptimalkan performa query dan mendukung pengelolaan peminjaman

**2.2. Desain Struktur Database**

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

* **Deskripsi**: Diagram ER mendefinisikan entitas utama seperti pengguna, buku, peminjaman, dan kategori serta hubungan antar-entitas dalam sistem.
* **Diagram**:



**2.2.2. Tabel-Tabel Utama**

* **Tabel Pengguna**
  + Kolom: : id(PK), nama, email, kata\_sandiDeskripsi: Menyimpan data pengguna perpustakaan.
* **Tabel Admin**
  + Kolom: : id(PK), nama, email, kata\_sandi.
  + Deskripsi: Menyimpan data admin perpustakaan.
* **Tabel Buku**
  + Kolom: id(PK), judul, pengarang, kategori, tersedia, deskripsi, gambar\_sampul, dibuat\_pada, diperbarui\_pada, id\_admin(FK).
  + Deskripsi: Menyimpan data buku e-book yang tersedia di perpustakaan.
* **Tabel Peminjaman**
  + Kolom: id(PK), id\_pengguna(FK), id\_buku(FK), tanggal\_peminjaman, tanggal\_pengembalian, status.
  + Deskripsi: Menyimpan catatan peminjaman buku oleh pengguna.
* **Tabel Kategori**
  + Kolom: id(PK), nama\_kategori, id\_admin(FK).
  + Deskripsi: Menyimpan kategori buku.

**2.3. Rekomendasi Struktur Database**

* **Rekomendasi**: Struktur database menggunakan Laravel Eloquent ORM untuk memudahkan pengelolaan data dan relasi antara tabel pengguna, buku, peminjaman, dan kategori. Hal ini akan mempermudah operasi CRUD serta menjaga integritas data peminjaman dan buku.

**3. Pemilihan Teknologi Frontend dan Backend**

**3.1. Tujuan Pemilihan Teknologi**

* Memilih teknologi yang mampu memenuhi kebutuhan fungsional dan non-fungsional aplikasi e-book perpustakaan, serta mempertimbangkan aspek performa, skalabilitas, dan pemeliharaan jangka panjang.

**3.2. Pemilihan Teknologi Frontend**

**3.2.1. Framework/Library Frontend**

* **Pilihan**: Bootstrap (untuk responsivitas dan UI yang cepat).
* **Deskripsi**: Bootstrap dipilih untuk mempercepat pengembangan antarmuka responsif dan modern.
* **Kriteria Pemilihan**: Bootstrap dipilih karena performa, dokumentasi yang baik, dan kemudahan integrasi dengan Laravel.

**3.2.2. Teknologi Tambahan**

* **CSS Framework**: Bootstrap untuk pengembangan UI yang konsisten dan responsif.
* **Tooling**: Laravel Mix untuk pengelolaan aset frontend dan kompilasi CSS/JavaScript.

**3.3. Pemilihan Teknologi Backend**

**3.3.1. Bahasa Pemrograman dan Framework Backend**

* **Pilihan**: PHP dengan Laravel.
* **Deskripsi**: Laravel sebagai framework PHP dipilih karena fitur out-of-the-box untuk routing, autentikasi, session, dan manajemen database menggunakan Eloquent ORM.
* **Kriteria Pemilihan**: Laravel dipilih karena performanya, dokumentasi yang komprehensif, serta komunitas yang besar.

**3.3.2. Basis Data**

* **Pilihan**: MySQL (dengan Eloquent ORM di Laravel).
* **Deskripsi**: MySQL digunakan untuk manajemen data e-book dan peminjaman.
* **Kriteria Pemilihan**: MySQL dipilih karena skalabilitas, dukungan untuk transaksi, dan kompatibilitas dengan Laravel.

**3.4. Rekomendasi Teknologi**

* **Frontend**: **Bootstrap** dipilih karena kemudahan penggunaan dan integrasi yang baik dengan Laravel.
* **Backend**:
* **Laravel** dipilih karena fleksibilitasnya dalam pengelolaan logika bisnis dan pengelolaan database.
* **Laravel** dengan PHP untuk logika server-side, termasuk penggunaan fitur-fitur seperti Laravel Sanctum untuk autentikasi API jika diperlukan.

**4. Skema Penggunaan Teknologi**

* **Frontend**: HTML, CSS (Bootstrap), JavaScript (Vue.js opsional), Laravel Mix untuk build proses.
* **Backend**:
* Laravel dengan PHP untuk server-side logic dan manajemen data menggunakan MySQL.
* Laravel (PHP) untuk logika server-side
* Laravel Eloquent ORM untuk interaksi dengan database
* Laravel Sanctum untuk autentikasi API (jika diperlukan)
* **Database**:
* MySQL dengan Eloquent ORM untuk pengelolaan relasi dan query.
* MySQL dengan Laravel Migrations untuk version control schema database