

# Chapter 1

## Pelaksanaan Kerja Praktek

### 1.1 Perancangan Sistem

#### 1.1.1 Desain Antarmuka

Desain antarmuka aplikasi manajemen faktur dirancang dengan mempertimbangkan kemudahan penggunaan dan efisiensi. Antarmuka utama terdiri dari beberapa komponen:

1. **Dashboard:** Menampilkan ringkasan statistik dan informasi penting seperti total faktur, status pembayaran, dan grafik tren. Dashboard ini dirancang untuk memberikan gambaran cepat kepada pengguna tentang kinerja bisnis mereka. Komponen ini menggunakan grafik interaktif yang dapat diubah ukurannya dan disesuaikan dengan kebutuhan pengguna.
2. **Manajemen Pelanggan:** Antarmuka untuk mengelola data pelanggan dengan fitur pencarian, penambahan, dan pengubahan data. Fitur ini memungkinkan pengguna untuk dengan mudah mencari pelanggan berdasarkan nama atau email, serta menambah atau mengedit informasi pelanggan dengan cepat.
3. **Manajemen Faktur:** Tampilan untuk membuat dan mengelola faktur dengan form yang intuitif dan tabel yang informatif. Formulir ini dirancang untuk meminimalkan kesalahan input dengan validasi real-time dan auto-complete untuk beberapa field.
4. **Pengaturan:** Panel untuk mengonfigurasi pengaturan aplikasi dan informasi perusahaan. Pengguna dapat mengubah informasi perusahaan, seperti nama, alamat, dan logo, yang akan tercermin di semua faktur yang dihasilkan.

#### 1.1.2 Desain Database

Aplikasi manajemen faktur ini memiliki beberapa entitas utama yang saling terhubung, yang dapat digambarkan dalam sebuah Entity-Relationship Diagram (ERD). Berikut adalah penjelasan mengenai entitas dan koneksinya:

1. **Users:** Tabel users menyimpan informasi pengguna yang dapat mengakses aplikasi. Setiap pengguna memiliki atribut seperti name, email, dan password. Tabel ini juga berhubungan dengan tabel sessions untuk melacak sesi pengguna dan password\_reset\_tokens untuk mengelola reset kata sandi.
2. **Customers:** Tabel customers menyimpan data pelanggan, termasuk nama, email, dan phone. Setiap pelanggan dapat memiliki banyak faktur (invoices), yang dihubungkan melalui customer\_id.
3. **Invoices:** Tabel invoices menyimpan informasi faktur, seperti invoice\_date, due\_date, email\_receiver, dan status. Setiap faktur terhubung ke satu pelanggan melalui customer\_id dan dapat memiliki banyak item (items), yang dihubungkan melalui invoice\_id.
4. **Items:** Tabel items menyimpan detail item yang termasuk dalam faktur, seperti name, description, quantity, price\_rupiah, dan price\_dollar. Setiap item terhubung ke satu faktur melalui invoice\_id.
5. **Exports:** Tabel exports mencatat informasi tentang ekspor data, termasuk file\_name, exporter, dan user\_id. Setiap ekspor terhubung ke pengguna yang melakukan ekspor melalui user\_id.

6. **Settings:** Tabel settings menyimpan pengaturan aplikasi, seperti company\_name, company\_email, dan company\_phone. Pengaturan ini digunakan di berbagai bagian aplikasi untuk menampilkan informasi perusahaan.
7. **Sessions:** Tabel sessions melacak sesi pengguna, menyimpan informasi seperti user\_id, ip\_address, dan last\_activity. Ini memungkinkan aplikasi untuk mengelola sesi pengguna dengan lebih baik.
8. **Password Reset Tokens:** Tabel password\_reset\_tokens menyimpan token reset kata sandi untuk pengguna, memungkinkan mereka untuk mengatur ulang kata sandi mereka jika diperlukan.

## 1.2 Flowchart Basis Data

Flowchart basis data menggambarkan alur data dalam aplikasi manajemen faktur:

### 1. Input Data Pelanggan:

- Pengguna memasukkan data pelanggan baru
- Data disimpan dalam tabel customers
- Sistem melakukan validasi data

### 2. Pembuatan Faktur:

- Pemilihan pelanggan dari database
- Input detail faktur
- Penambahan item-item faktur
- Kalkulasi otomatis total dalam IDR dan USD

### 3. Penyimpanan dan Pemrosesan:

- Data faktur disimpan dalam tabel invoices
- Item-item disimpan dalam tabel items
- Update status pembayaran

## 1.3 Sequence Diagram

Sequence diagram menunjukkan interaksi antar komponen sistem dengan lebih rinci sebagai berikut:

### 1. Proses Autentikasi:

- **User melakukan login:**
  - Pengguna memasukkan username dan password ke dalam form login.
  - Form mengirimkan data kredensial ke server melalui permintaan HTTP POST.
- **Sistem memverifikasi kredensial:**
  - Server menerima data kredensial dan memulai proses verifikasi.
  - Sistem mencari username di tabel users dalam database.
  - Jika username ditemukan, sistem memverifikasi password yang dimasukkan dengan password yang disimpan menggunakan hashing.
  - Jika verifikasi berhasil, sistem melanjutkan ke pembuatan sesi. Jika gagal, sistem mengirimkan respons kesalahan.
- **Membuat sesi pengguna:**
  - Sistem membuat sesi baru dengan menyimpan user\_id, ip\_address, dan last\_activity ke tabel sessions.
  - Sistem mengirimkan cookie sesi ke browser pengguna.
  - Pengguna diarahkan ke dashboard utama aplikasi.

## 2. Pembuatan Faktur:

- **User memilih create invoice:**
  - Pengguna mengklik tombol "Create Invoice" di dashboard.
  - Sistem menyiapkan form pembuatan faktur.
- **Sistem menampilkan form:**
  - Sistem merender form dengan field untuk memilih pelanggan, memasukkan tanggal faktur, dan menambahkan item.
  - Sistem memuat data pelanggan yang tersedia untuk dipilih.
- **User mengisi detail faktur:**
  - Pengguna memasukkan informasi yang diperlukan, seperti memilih pelanggan, menambahkan item, dan menentukan tanggal jatuh tempo.
  - Sistem melakukan validasi real-time.
- **Sistem memvalidasi dan menyimpan:**
  - Sistem memeriksa semua input untuk memastikan tidak ada kesalahan.
  - Jika validasi berhasil, sistem menyimpan data faktur ke tabel `invoices` dan item ke tabel `items`.
  - Sistem mengirimkan konfirmasi ke antarmuka pengguna.

## 3. Pengiriman Faktur:

- **User memilih send invoice:**
  - Pengguna memilih faktur yang ingin dikirim dan mengklik opsi untuk mengirim faktur.
- **Sistem generate PDF:**
  - Sistem menggunakan library PDF untuk merender tampilan faktur menjadi file PDF.
  - PDF disimpan sementara di server.
- **Sistem mengirim email:**
  - Sistem menyiapkan email dengan lampiran PDF dan detail penerima.
  - Sistem menggunakan layanan email untuk mengirim email ke penerima.
  - Sistem mencatat aktivitas pengiriman email dalam log.
- **Update status pengiriman:**
  - Sistem memeriksa apakah email berhasil dikirim.
  - Jika berhasil, sistem memperbarui status faktur menjadi "sent" di tabel `invoices`.
  - Sistem mengirimkan notifikasi ke pengguna.

# 1.4 Activity Diagram

Activity diagram menggambarkan alur kerja sistem:

## 1. Login dan Autentikasi:

- Input kredensial
- Verifikasi user
- Akses dashboard

## 2. Manajemen Faktur:

- Pembuatan faktur baru
- Pengeditan faktur
- Pengiriman ke pelanggan
- Pemantauan status

## 3. Pengelolaan Data:

- Manajemen pelanggan
- Update pengaturan
- Ekspor data

## 1.5 Sistem Basis Data untuk Faktur

Sistem basis data faktur dirancang dengan mempertimbangkan:

### 1. **Integritas Data:**

- Penggunaan foreign keys
- Validasi input
- Konsistensi data

### 2. **Performa:**

- Indeks optimal
- Query yang efisien
- Caching strategis

### 3. **Keamanan:**

- Enkripsi data sensitif
- Manajemen akses
- Audit trail