

1. Judul Project

Sistem Informasi Manajemen Data Pembayaran Berbasis Web dengan Role Admin dan User

2. Latar Belakang

Pengelolaan data pembayaran secara manual rentan terhadap kesalahan pencatatan dan sulit dilacak. Sistem berbasis web dengan hak akses yang jelas antara admin dan user dibutuhkan agar proses pencatatan pembayaran menjadi lebih efisien, aman, dan terorganisir. Dengan sistem ini, admin dapat mengelola seluruh data pembayaran, sementara user dapat melihat status dan riwayat pembayaran mereka sendiri.

3. Rumusan Masalah

- Bagaimana merancang sistem CRUD untuk mengelola data pembayaran secara efisien?
- Bagaimana membedakan hak akses antara admin (pengelola) dan user (pelanggan)?
- Bagaimana user dapat melihat data pembayaran tanpa bisa mengubahnya?

4. Tujuan

- Membangun sistem informasi berbasis web yang mendukung pengelolaan data pembayaran secara CRUD oleh admin.
- Menerapkan sistem login dengan role-based access control (RBAC).
- Menyediakan akses tampilan data pembayaran untuk user tanpa izin edit/hapus.

5. Manfaat

- Mempermudah admin dalam pencatatan dan pengelolaan data pembayaran.
- Meningkatkan akurasi dan keamanan informasi pembayaran.
- Memberikan transparansi data kepada user tanpa risiko manipulasi.

6. Fitur Utama Sistem

- Login Role-Based (Admin dan User)
- CRUD Data Pembayaran (hanya untuk Admin)
- View Data Pembayaran (untuk User)
- Riwayat Pembayaran per User

- Dashboard sesuai peran (Admin/User)
- Logout dan proteksi sesi

7. Alur Kerja Sistem (Use Case)

- **Admin login** → Kelola data pembayaran → Tambah, ubah, hapus, dan lihat data pembayaran
- **User login** → Lihat riwayat pembayaran sendiri → Tidak bisa edit atau hapus

8. Spesifikasi Teknis

- **Bahasa Pemrograman:** PHP (Laravel)
- **Database:** MySQL
- **Frontend:** Blade + TailwindCSS atau Bootstrap
- **Autentikasi & Otorisasi:** Laravel Auth + Spatie Role Middleware

9. Kebutuhan Pengguna

Role	Kebutuhan
Admin	Kelola data pembayaran (CRUD)
User	Melihat riwayat dan status pembayaran

10. Desain Antarmuka

- Halaman login
- Dashboard Admin: tabel data pembayaran + aksi CRUD
- Dashboard User: riwayat dan status pembayaran
- Form tambah/edit pembayaran (admin saja)

11. Rencana Pengembangan

Tahap	Waktu Pelaksanaan	Keterangan
Analisis Kebutuhan	1 hari	Identifikasi data & alur
Perancangan Sistem	1 hari	Desain database dantar muka
Implementasi	1 hari	Koding backend & frontend

Pengujian	1 hari	Debugging dan validasi
Dokumentasi & Final	1 hari	Pelaporan dan dokumentas

Dokumentasi

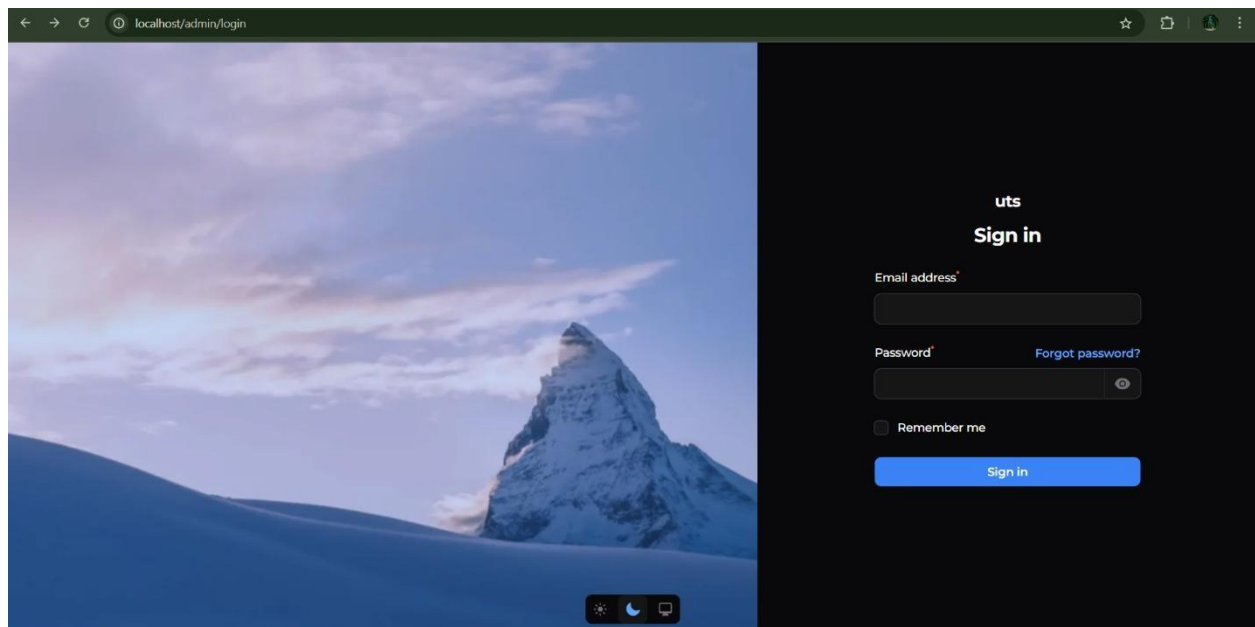
12. Hasil dan Pembahasan

Hasil dari implementasi sistem ini adalah sebuah aplikasi web yang dapat digunakan untuk mencatat, mengelola, dan menampilkan data pembayaran secara terstruktur dan aman, sesuai dengan hak akses pengguna.

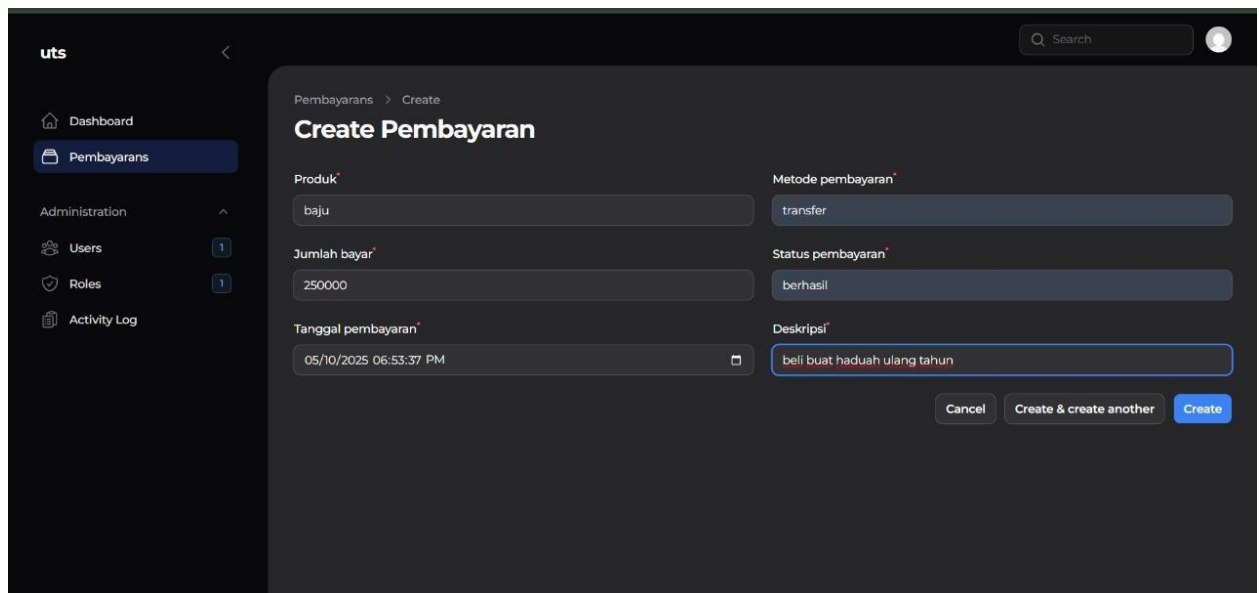
Hasil Implementasi Sistem

Setelah proses pengembangan selesai, sistem manajemen data Pembayaran berhasil diimplementasikan dengan fitur utama sebagai berikut:

Login Role-Based: Sistem berhasil membedakan akses antara **admin** dan **user** berdasarkan akun masing-masing.

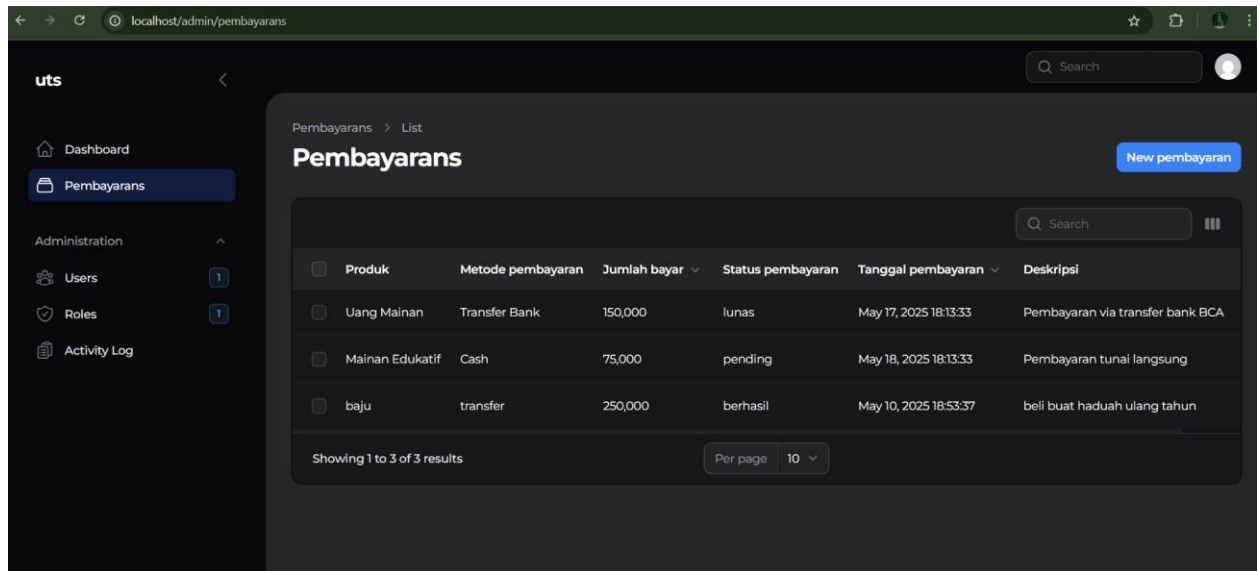


Fitur menambahkan baru untuk Admin: Admin dapat melakukan Create, Read, Update, dan Delete data pembayaran melalui antarmuka web yang mudah digunakan.



The screenshot shows a web application interface for creating a payment record. The left sidebar contains navigation links: Dashboard, Pembayaran (active), Administration, Users, Roles, and Activity Log. The main content area is titled 'Create Pembayaran' and contains several input fields: 'Produk' (baju), 'Metode pembayaran' (transfer), 'Jumlah bayar' (250000), 'Status pembayaran' (berhasil), 'Tanggal pembayaran' (05/10/2025 06:53:37 PM), and 'Deskripsi' (beli buat hadiah ulang tahun). At the bottom right, there are three buttons: 'Cancel', 'Create & create another', and 'Create'.

View Data untuk User dapat melihat data pembayaran mereka sendiri, termasuk status dan deskripsi pembayaran, tanpa bisa mengubah atau menghapus data tersebut. Hal ini menjaga integritas dan keamanan data.



13. Pembahasan

- Sistem ini meningkatkan efisiensi pengelolaan data pembayaran karena admin dapat mencatat dan memperbarui data pembayaran secara langsung tanpa perlu mencatat secara manual.
- Hak akses yang dibatasi untuk user meminimalkan risiko perubahan data oleh pihak yang tidak berwenang, sehingga data tetap valid dan terpercaya.
- Antarmuka yang responsif dan intuitif memudahkan pengguna dari kedua role untuk mengakses sistem tanpa pelatihan khusus.
- Pemanfaatan Laravel dengan fitur otentikasi bawaan dan dukungan middleware mempermudah penerapan **role-based access control (RBAC)**.
- Pengembangan selanjutnya dapat mencakup fitur seperti:
 - **Upload bukti pembayaran** oleh user
 - **Notifikasi status pembayaran** (misalnya, jika sudah dikonfirmasi)
 - **Export laporan pembayaran ke PDF atau Excel**
- Sistem ini meningkatkan efisiensi pengelolaan data supir karena admin dapat mengelola data secara langsung tanpa perlu prosedur manual.
- Pembatasan akses untuk driver meminimalisir risiko data diubah oleh pihak yang tidak berwenang, sehingga menjaga keakuratan data.
- Antarmuka yang sederhana dan intuitif membuat pengguna dari kedua role dapat dengan mudah menggunakan sistem tanpa membutuhkan pelatihan khusus.

- Penggunaan Laravel sebagai framework memberikan kemudahan dalam implementasi fitur otentikasi dan middleware untuk role-based access control.
- Penambahan fitur seperti pengelolaan foto supir atau notifikasi status dapat menjadi pengembangan berikutnya untuk meningkatkan fungsionalitas sistem.

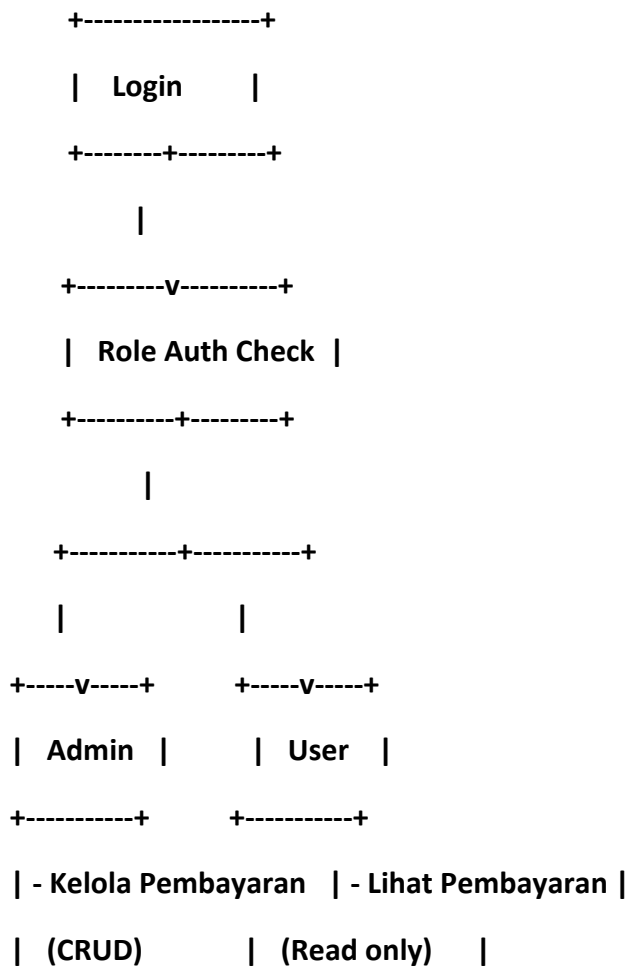
14. Penutup

Dengan adanya sistem ini, diharapkan proses pengelolaan data pembayaran menjadi lebih **terstruktur, aman, dan efisien**.

Admin memiliki kendali penuh dalam mengelola informasi pembayaran, sementara user tetap mendapatkan transparansi tanpa dapat mengubah data.

Sistem ini dapat dikembangkan lebih lanjut sesuai kebutuhan bisnis agar lebih fungsional dan adaptif terhadap pertumbuhan skala operasional.

Use Case Diagram (Teks Deskriptif)



+-----+-----+

Penjelasan Use Case

Actor	Use Case	Deskripsi
Admin	Login	Admin masuk ke sistem menggunakan kredensial.
	Kelola	Admin dapat membuat, membaca, mengubah, dan
	Pembayaran (CRUD)	menghapus data pembayaran.
User	Login	User masuk ke sistem menggunakan
	Actor	akunnya.
	Use Case	Deskripsi
	Lihat Data Pembayaran	User hanya dapat melihat data pembayaran
		yang terkait dengan akunnya, tanpa bisa mengedit atau menghapus.