

Software Requirements Specification (SRS)

Project: Dompetku – Aplikasi Manajemen Pengeluaran Pribadi

Tech Stack: Laravel 12, Filament v3, MySQL, Swagger, SMTP, Docker, Nginx, Livewire

Pendahuluan

1. Latar Belakang

Aplikasi Dompetku dibuat untuk memfasilitasi pengelolaan pengeluaran pribadi dengan cara yang mudah dan efisien. Aplikasi ini dikembangkan oleh PT Keuangan Cerdas Indonesia, yang bertujuan untuk membantu pengguna dalam mencatat, mengelola, dan memantau pengeluaran sehari-hari. Dengan menggunakan teknologi Laravel 12 dan Filament v3, aplikasi ini dapat diakses kapan saja dan di mana saja melalui perangkat yang mendukung. Selain itu, aplikasi ini juga menyediakan fitur untuk integrasi API guna mempermudah pengelolaan data pengeluaran dan sistem manajemen keuangan secara menyeluruh.

2. Tujuan

Dokumen ini menyajikan spesifikasi kebutuhan sistem untuk aplikasi **Dompetku**, sebuah sistem manajemen pengeluaran pribadi berbasis web yang menggunakan Laravel 12 sebagai backend, Livewire untuk frontend, dan Filament v3 untuk panel admin. Sistem ini juga mengintegrasikan **Swagger** untuk API dan SMTP untuk notifikasi email. Tujuan dari sistem ini adalah untuk membantu pengguna dalam mengelola pengeluaran secara efisien dan memberikan kemudahan akses ke data keuangan pribadi.

3. Ruang Lingkup

Sistem ini mencakup beberapa fitur utama yang saling terintegrasi, yaitu:

- **Manajemen Pengeluaran:** Sistem memungkinkan pengguna untuk menambah, mengedit, menghapus, dan mengelola pengeluaran mereka.
- **API:** Sistem menyediakan API untuk mengakses data pengeluaran yang dapat digunakan untuk integrasi dengan aplikasi lain.
- **Dashboard Admin:** Admin dapat mengelola data pengguna dan kategori pengeluaran melalui panel admin berbasis Filament v3.
- **Notifikasi Email:** Sistem mengirimkan notifikasi kepada pengguna setelah mereka melakukan pengeluaran atau pencatatan data.

4. Batasan Masalah

Batasan masalah diisi dengan:

- **Sistem hanya mencatat pengeluaran pribadi**, tidak mendukung pencatatan pemasukan atau pengelolaan utang-piutang.
- **Aplikasi berbasis web**, dan tidak tersedia untuk platform mobile (aplikasi hanya dapat diakses melalui browser).
- **Pembayaran hanya dilakukan melalui integrasi dengan API** untuk pencatatan pengeluaran dan tidak ada transaksi pembayaran langsung melalui aplikasi.
- **Data hanya tersedia dalam bentuk digital**, tidak ada fitur untuk mencetak laporan fisik dari pengeluaran.
- **Aplikasi tidak mendukung multi-user untuk pengelolaan bersama**, artinya hanya satu pengguna per akun yang dapat mengelola data pengeluaran.
- **Kategori pengeluaran terbatas pada jenis kategori yang sudah ditentukan** (misalnya: Makanan, Transportasi, dll), tidak ada fitur untuk membuat kategori baru oleh pengguna.

- Sistem tidak menyediakan fitur analisis atau perhitungan lanjutan selain ringkasan total pengeluaran.

5. Definisi, Akronim, dan Singkatan

| No | Akrionim/Singkatan | Definisi |
|----|------------------------|---|
| 1 | Dometku | Aplikasi manajemen pengeluaran pribadi yang dikembangkan oleh PT Keuangan Cerdas Indonesia. |
| 2 | Laravel | Kerangka kerja backend berbasis PHP yang digunakan untuk membangun aplikasi Dometku . |
| 3 | Filament | Kerangka kerja untuk membuat antarmuka admin berbasis Laravel, mempermudah pengelolaan aplikasi. |
| 4 | Livewire | Kerangka kerja untuk frontend di Laravel yang memungkinkan interaksi dinamis tanpa JavaScript eksternal. |
| 5 | MySQL | Sistem manajemen basis data relasional yang digunakan untuk menyimpan data dalam aplikasi Dometku . |
| 6 | Docker | Platform untuk menjalankan aplikasi dalam container, memastikan lingkungan pengembangan yang konsisten. |
| 7 | RBAC | Role-Based Access Control – Sistem kontrol akses berbasis peran pengguna. |
| 8 | API | Application Programming Interface – Antarmuka komunikasi antar sistem. |
| 9 | Swagger | Alat untuk dokumentasi dan pengujian API secara interaktif, digunakan untuk integrasi API Dometku . |
| 10 | SMTP | Simple Mail Transfer Protocol – Protokol untuk pengiriman email otomatis (misalnya, untuk notifikasi transaksi). |
| 11 | Payment Gateway | Sistem yang memfasilitasi transaksi pembayaran secara online antara pengguna dan aplikasi. |
| 12 | MySQL | Sistem manajemen database relasional yang digunakan untuk penyimpanan data. |
| 13 | Panel Admin | Antarmuka backend berbasis web untuk admin dalam mengelola konten, data pengguna, dan sistem. |
| 14 | Frontend | Bagian sistem yang berinteraksi langsung dengan pengguna, biasanya berupa antarmuka visual. |
| 15 | Backend | Bagian sistem yang berjalan di sisi server dan menangani logika, database, dan API. |
| 16 | Nginx | Web server yang digunakan untuk menjalankan aplikasi web dan menangani request dari pengguna. |

Deskripsi Umum

1. Perspektif Produk

Sistem Dompetku adalah aplikasi berbasis web yang dibangun dengan Laravel 12 sebagai backend dan Livewire untuk frontend. Livewire memungkinkan interaksi dinamis tanpa memerlukan JavaScript eksternal. Untuk pengelolaan sistem yang mudah dan efisien, Filament v3 digunakan sebagai antarmuka admin.

Produk ini dirancang untuk memenuhi kebutuhan berbagai jenis pengguna, seperti pengguna (user) yang akan mengelola pengeluaran pribadi mereka, dan admin yang bertanggung jawab atas pengelolaan data pengeluaran serta kategori. Aplikasi ini juga memungkinkan pengguna untuk mencatat, menyaring, dan melihat transaksi pengeluaran dalam bentuk ringkasan yang jelas.

Selain itu, aplikasi ini terintegrasi dengan Swagger untuk mendokumentasikan dan menguji API secara interaktif, serta menyediakan fitur notifikasi email yang dikirim melalui SMTP setiap kali ada pencatatan pengeluaran atau perubahan data penting.

Sistem ini dibangun dengan fokus pada performa tinggi, keamanan yang solid, dan antarmuka yang user-friendly, sehingga dapat memberikan pengalaman yang efisien dan mudah digunakan oleh semua pihak yang terlibat.

2. Fungsi Produk

Setiap fitur dalam sistem ini akan terintegrasi untuk mempermudah operasional dan penggunaan:

- Expense Management: Menambah, mengedit, dan menghapus pengeluaran.
- Admin Dashboard: Admin dapat mengelola data pengeluaran dan pengguna.
- API Integration: Swagger digunakan untuk mendokumentasikan dan menguji API.

3. Karakteristik Pengguna

Karakteristik pengguna pada aplikasi Dompetku terdiri dari beberapa jenis pengguna:

- Pengguna (User):
Pengguna yang bertanggung jawab untuk mencatat dan melihat pengeluaran pribadi mereka. Pengguna memiliki hak akses terbatas pada data mereka sendiri.
- Admin:
Admin yang mengelola kategori pengeluaran, data pengguna, serta dapat mengakses data pengeluaran seluruh pengguna. Admin memiliki kontrol penuh untuk menambah, mengedit, atau menghapus data pengguna.

4. Hak Akses Pengguna

Hak akses pengguna pada sistem Dompetku dibedakan berdasarkan role atau peran masing-masing pengguna, dengan rincian sebagai berikut:

| Role | Hak Akses |
|------|-----------|
|------|-----------|

| | |
|--------------|--|
| User | Mencatat dan mengelola pengeluaran pribadi, melihat ringkasan pengeluaran mereka. |
| Admin | Mengelola kategori pengeluaran, data pengguna, dan laporan pengeluaran seluruh pengguna. |
| API | Mengakses data pengeluaran menggunakan endpoint yang disediakan oleh API. |

5. Asumsi dan Ketergantungan

Agar sistem Dompetku dapat berjalan optimal dan sesuai dengan desain, terdapat beberapa komponen eksternal yang menjadi asumsi dan ketergantungan utama. Berikut penjelasan mengenai alat yang digunakan, fungsinya, dan risiko yang mungkin timbul jika komponen tersebut tidak tersedia:

- Database MySQL:**
 MySQL dipilih karena performanya yang stabil, ringan, dan kompatibel dengan framework Laravel 12. Seluruh data di sistem, mulai dari data pengguna hingga pengeluaran yang tercatat, disimpan di dalam database ini. Tanpa database, sistem tidak dapat menyimpan atau mengambil data, yang akan menghambat seluruh fungsionalitas aplikasi.
- Framework Laravel, Filament, dan Livewire**
 Laravel 12 digunakan sebagai framework utama untuk backend, memberikan fondasi yang kuat dan aman untuk pengelolaan data dan transaksi. **Filament** digunakan untuk membangun antarmuka admin yang user-friendly, memudahkan pengelolaan data dan kategori pengeluaran. **Livewire** digunakan untuk membuat antarmuka frontend yang dinamis, memungkinkan interaksi tanpa menggunakan JavaScript eksternal. Tanpa Laravel, backend akan membutuhkan framework lain yang mungkin kurang efisien. Tanpa Filament, antarmuka admin harus dibangun dari awal, memerlukan waktu dan usaha lebih banyak. Tanpa Livewire, aplikasi akan membutuhkan lebih banyak JavaScript, menambah kompleksitas.
- Docker**
 Docker digunakan untuk membuat lingkungan pengembangan yang konsisten di semua platform, memastikan bahwa aplikasi berjalan dengan cara yang sama di lingkungan pengembangan dan produksi. Tanpa Docker, pengelolaan lingkungan pengembangan dan produksi akan lebih sulit, dan aplikasi mungkin tidak dapat berjalan konsisten di berbagai platform.
- Nginx (Web Server)**
 Nginx digunakan sebagai web server utama yang menerima permintaan dari pengguna dan meneruskannya ke aplikasi Laravel. Tanpa Nginx, sistem tidak dapat diakses melalui domain publik, dan komunikasi antara frontend dan backend akan terputus.

Kebutuhan Fungsional

1. Company Profile

- CRUD (Create, Read, Update, Delete) untuk konten publik pada halaman Home, About Us, Blog, dan Contact Us.
- Admin dapat memperbarui dan mengelola informasi perusahaan yang akan dilihat oleh publik.

2. Expense Management

- User dapat menambah, mengedit, dan menghapus pengeluaran pribadi mereka.
- User dapat memilih kategori pengeluaran (misalnya: Makanan, Transportasi, dll).
- Admin dapat mengelola kategori pengeluaran.
- Sistem menyediakan fitur untuk menyaring data pengeluaran berdasarkan tanggal dan kategori.
- Menampilkan ringkasan total pengeluaran pengguna.

3. User Management

- User dapat membuat akun, login, dan logout.
- Admin dapat mengelola pengguna (membuat, mengedit, menghapus akun pengguna).

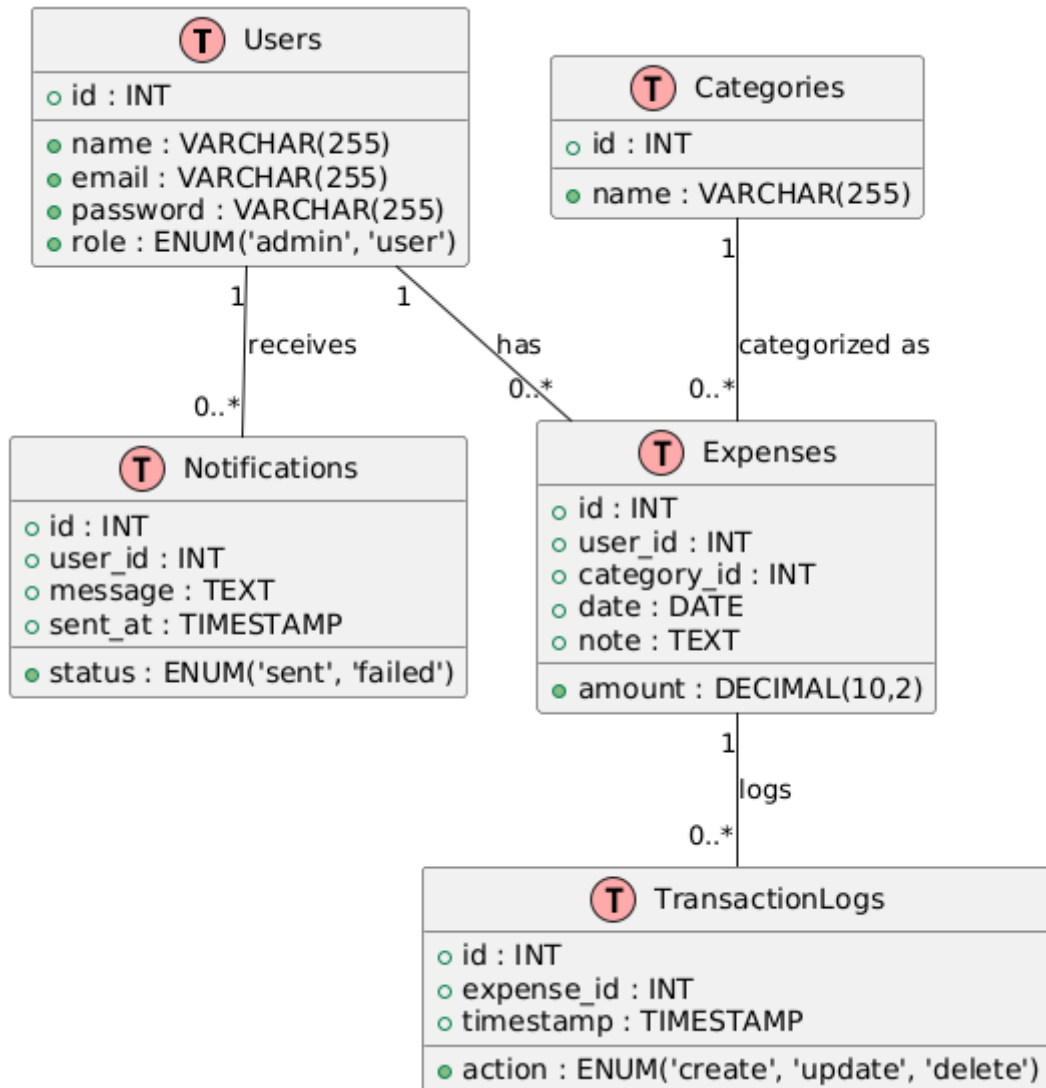
4. Admin Dashboard

- Admin dapat mengelola data pengeluaran seluruh pengguna.
- Admin dapat melihat dan mengelola kategori pengeluaran.
- Admin dapat mengakses statistik pengeluaran seluruh pengguna.

5. API Integration

- Menyediakan endpoint untuk pengeluaran pengguna melalui **Swagger**.
- Endpoint API yang disediakan:
 - **GET**: Menampilkan data pengeluaran.
 - **POST**: Menambahkan pengeluaran baru.
 - **DELETE**: Menghapus pengeluaran tertentu.
- API dibatasi dengan autentikasi berbasis **token** dan hanya bisa diakses oleh pengguna yang terautentikasi.

DESAIN DATABASE



Perluasan di Masa Depan

Sistem **Dompetku** dirancang agar dapat terus berkembang sesuai dengan kebutuhan pengguna dan fitur yang relevan dengan pengelolaan keuangan pribadi. Beberapa rencana pengembangan ke depan antara lain:

1. Fitur Pengelolaan Anggaran

- Pengguna dapat menetapkan anggaran untuk kategori pengeluaran tertentu, seperti makanan, transportasi, dll.
- Sistem akan memberikan notifikasi atau peringatan ketika pengguna mencapai atau melampaui anggaran yang telah ditentukan.
- Ringkasan anggaran akan ditampilkan dalam bentuk grafik atau tabel agar lebih mudah dipahami oleh pengguna.

2. Integrasi Pembayaran Digital

- Sistem dapat memperluas integrasi pembayaran digital untuk mempermudah proses pencatatan pengeluaran langsung dari aplikasi.
- Menambahkan Midtrans atau metode pembayaran lainnya, seperti transfer bank atau e-wallet, yang memungkinkan pengguna untuk langsung mencatat pengeluaran mereka setelah melakukan transaksi pembayaran.

3. Laporan dan Analisis Keuangan Lanjutan

- Menyediakan **laporan dan analisis pengeluaran** berbasis waktu (harian, mingguan, bulanan) atau kategori pengeluaran.
- Menyajikan **grafik analisis** untuk membantu pengguna memahami pola pengeluaran mereka lebih baik dan memberikan saran untuk **pengelolaan keuangan yang lebih efisien**.

4. Pengelolaan Utang dan Piutang

- Menambahkan fitur untuk mencatat dan melacak utang dan piutang pengguna.
- Pengguna dapat menetapkan pengingat terkait pembayaran utang atau menerima notifikasi saat ada perubahan status piutang.

5. Sertifikat Pengeluaran Digital

- Mengintegrasikan QR code pada sertifikat pengeluaran yang memungkinkan pengguna untuk memverifikasi atau mengonfirmasi transaksi pengeluaran secara otomatis.
- QR code ini dapat digunakan sebagai bukti pembayaran yang valid untuk keperluan pajak atau pelaporan lainnya.

6. Integrasi dengan Aplikasi Keuangan Lainnya

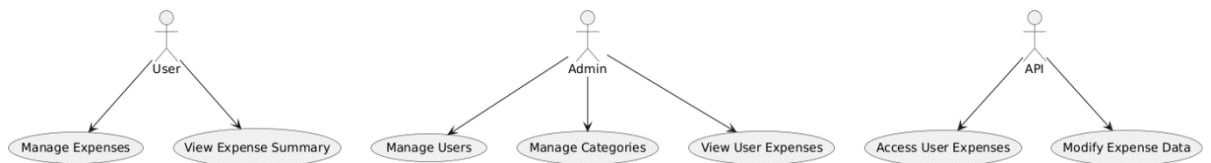
- Menambahkan integrasi dengan aplikasi keuangan lain atau dompet digital untuk mengimpor data pengeluaran dan menyatukan informasi keuangan di satu tempat.
- Misalnya, menghubungkan dengan aplikasi seperti Ovo, GoPay, atau aplikasi perbankan untuk otomatisasi pencatatan pengeluaran.

LAMPIRAN

1. Diagram Use Case

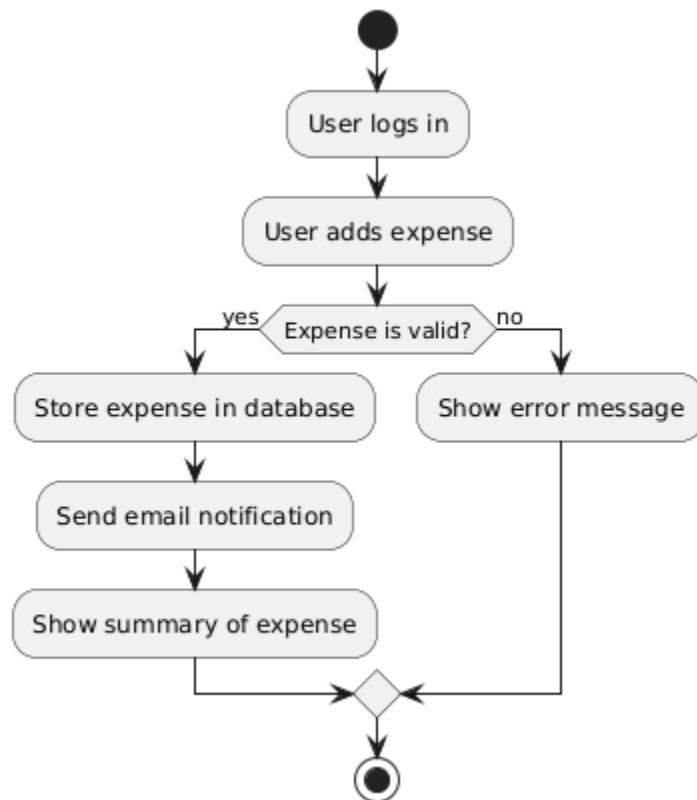
Untuk Dompetku, kita akan membuat Use Case Diagram yang menggambarkan interaksi antara pengguna dan sistem. Berikut adalah elemen utama dalam use case:

- User: Mengelola pengeluaran pribadi mereka (CRUD pengeluaran, melihat ringkasan, dll).
- Admin: Mengelola data pengguna, kategori pengeluaran, dan melihat laporan pengeluaran.
- API: Mengelola akses API untuk sistem pihak ketiga.



2. Diagram Flowchart

Flowchart menggambarkan alur utama dalam sistem Dompetku. Berikut adalah contoh flowchart yang menggambarkan proses pencatatan pengeluaran:



3. Dokumentasi API

← → ↺ uastary.test/api/documentation#/default/get_categories

Dompetku API^{1.0.0}

<https://uastary.test/api/docs-json>

default

GET /expenses Retrieve all expenses

Parameters

Cancel

No parameters

Execute

Responses

Response content type application/json

| Code | Description |
|------|---|
| 200 | A list of expenses Example Value Model <pre>[{ "id": 0, "amount": 0, "category_id": 0, "date": "2025-07-22", "note": "string" }]</pre> |

POST /expenses Create a new expense

Parameters

Try it out

| Name | Description |
|---|--|
| expense * required object (body) | Expense data Example Value Model <pre>{ "id": 0, "amount": 0, "category_id": 0, "date": "2025-07-22", "note": "string" }</pre> <div>Parameter content type application/json</div> |

Responses

Response content type application/json

| Code | Description |
|------|-----------------|
| 201 | Expense created |

GET /categories Retrieve all categories

Parameters

Try it out

No parameters

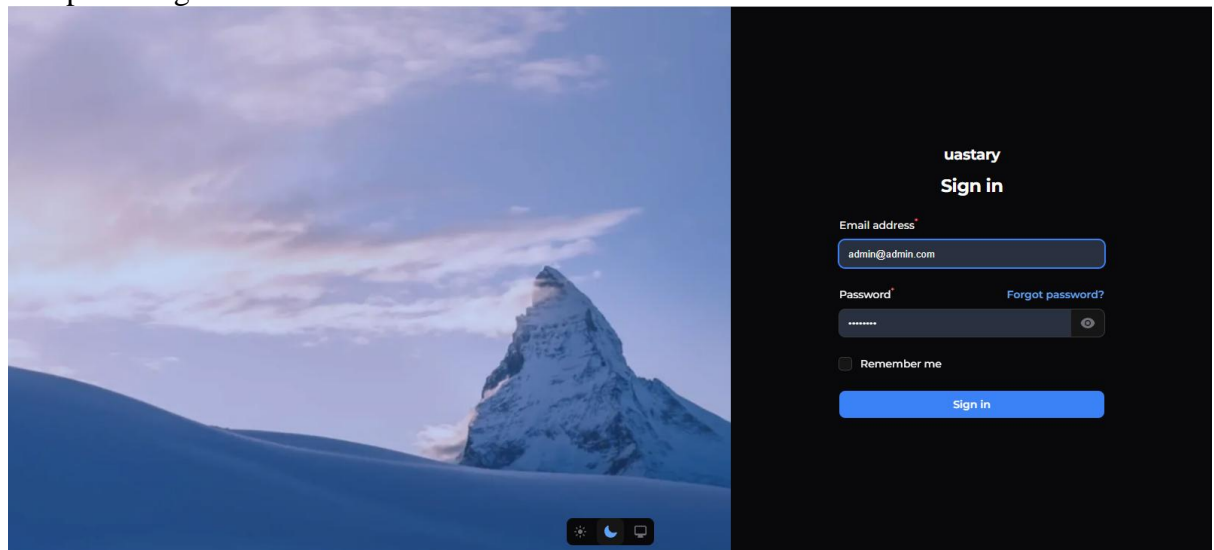
Responses

Response content type application/json

| Code | Description |
|------|--|
| 200 | A list of categories Example Value Model <pre>[{ "id": 0, "name": "string" }]</pre> |

| Models | |
|--------------|--|
| Expense ▾ { | |
| id | integer Expense ID |
| amount | number(\$float) Amount of the expense |
| category_id | integer Category ID linked to the expense |
| date | string(\$date) Date of the expense |
| note | string Additional notes for the expense |
| } | |
| Category ▾ { | |
| id | integer Category ID |
| name | string Category name |
| } | |

4. Tampilan Login



5. Tampilan Backend

uastary

Dashboard

Kategori

Pengeluaran

Administration

Users 1

Roles 1

Activity Log

Dashboard

Users 1

Latest Access Logs

| Type ▾ | Event ▾ | Description | Subject | User | Logged At ▾ |
|----------|---------|--------------------------|-------------|-------|-----------------------|
| Resource | Created | Expense Created by Admin | Expense # 6 | Admin | Jul 22, 2025 11:54:41 |
| Resource | Created | Expense Created by Admin | Expense # 5 | Admin | Jul 22, 2025 11:54:08 |
| Resource | Created | Expense Created by Admin | Expense # 4 | Admin | Jul 22, 2025 11:53:39 |
| Resource | Created | Expense Created by Admin | Expense # 3 | Admin | Jul 22, 2025 11:52:52 |
| Resource | Created | Expense Created by Admin | Expense # 2 | Admin | Jul 22, 2025 11:52:03 |

https://uastary.test/admin

uastary

Dashboard

Kategori

Pengeluaran

Administration

Users1

Roles1

Activity Log

Categories > List

Categories

New category

Search

| Nama Kategori | |
|---------------|---|
| Makanan | Edit Delete |
| Transportasi | Edit Delete |
| Kesehatan | Edit Delete |
| Hiburan | Edit Delete |
| Belanja | Edit Delete |

Showing 1 to 5 of 5 results

Per page10

uastary

Dashboard

Kategori

Pengeluaran

Administration

Users1

Roles1

Activity Log

Pengeluaran > List

Pengeluaran

New expense

Search

| Judul Pengeluaran | Kategori | Jumlah Pengeluaran | Tanggal | |
|---------------------------------|--------------|--------------------|--------------|---|
| Pengeluaran untuk makanan | Makanan | IDR 50,000.00 | Jul 22, 2025 | Edit Delete |
| Biaya Transportasi Pulang Pergi | Transportasi | IDR 80,000.00 | Jul 21, 2025 | Edit Delete |
| Obat-Obatan | Kesehatan | IDR 200,000.00 | Jul 11, 2025 | Edit Delete |
| Menonton Film | Hiburan | IDR 100,000.00 | Jul 18, 2025 | Edit Delete |
| Belanja Pakaian | Belanja | IDR 300,000.00 | Jul 25, 2025 | Edit Delete |

Showing 1 to 5 of 5 results

Per page10