# Deskripsi Projek

Projek : **KAS Masjid**

**Masalah:**

Pencatatan kas masjid masih dilakukan secara manual menggunakan buku kas dan informasi keuangan hanya bisa dilihat di papan informasi masjid.

**Solusi:**

Dibuat suatu system yang dapat digunakan oleh bendahara masjid untuk menginput data transaksi keuangan masjid, system ini juga memiliki opsi untuk mempublkasikan keuangan masjid agar lebih transparant

**Aktor:**

Pengurus/Bendahara Masjid

1. Malakukan pendaftaran
2. Mendaftarkan Masjid yang diurus
3. Meng-input pemasukan keuangan masjid
4. Meng-input pengeluaran keuangan masjid
5. Mencetak Laporan pemasukan keuangan
6. Mencetak laporan pengeluaran keuangan
7. Mencetak laporan kas
8. Melihat Data donatur (optional)

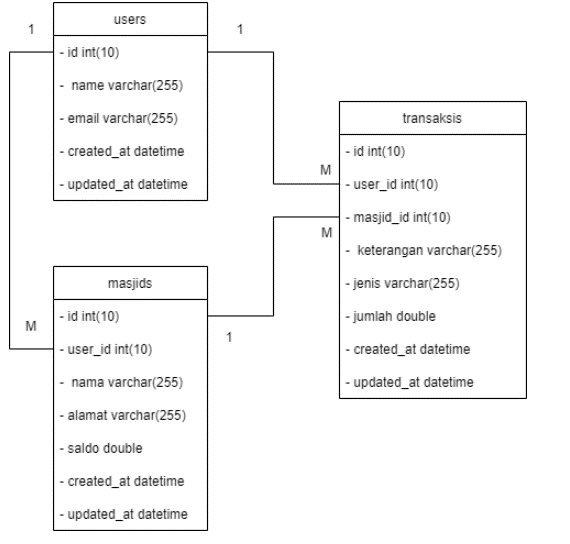
Pengunjung

1. Melihat informasi masjid
2. Melihat informasi keuangan masjid
3. Mendaftar Sebagai Donatur Masjid (optional)

Donatur

1. Login
2. Update Profil
3. Melakukan Donasi (optional)

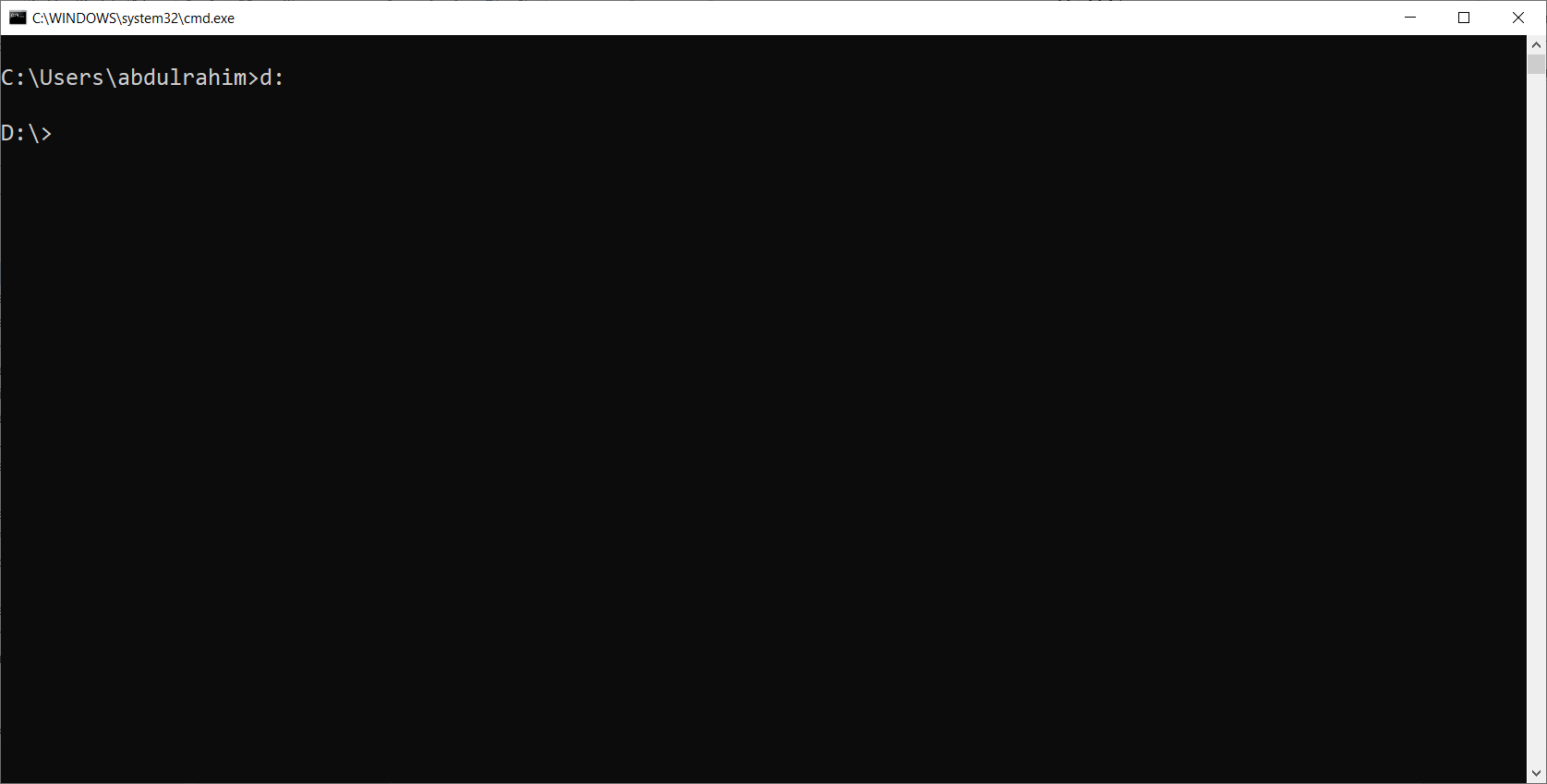
# Relasi Tabel (belum termasuk donasi)



# Instalasi Laravel dan Packages pendukung

Jalankan xampp, aktifkan services apache dan mysql.

Buka command line lalu arahkan ke drive d dengan perintah :



Gambar 1 Windows Command Line

Buat Projek laravel dengan perintah composer :

* composer create-project laravel/laravel kasmajid

Buka Project KasMasjid, lalu di vscode tekan Ctrl+Shift+C untuk membuka command line lalu install package berikut:

Package laravel ide-helper untuk membantu proses developer.

* composer require --dev barryvdh/laravel-ide-helper
* php artisan ide-helper:generate

Install Package collective html untuk membuat form

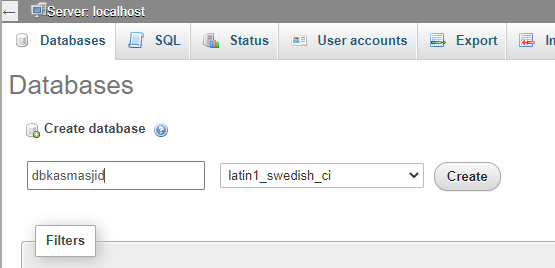
* composer require laravelcollective/html

Install laravel Starter Kits untuk membuat login

* composer require laravel/ui

Buka phpmyadmin <http://localhost/phpmyadmin>

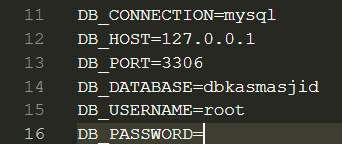
Buat database dengan nama **dbkasmasjid**



Gambar 2 phpmyadmin membuat database

Buka vscode dan open folder projek.

Buka file .env lalu ubah nama database seperti berikut.



Gambar 3 File .env laravel

Generate page login, buka command line di vscode lalu jalankan perintah berikut :

* php artisan ui bootstrap --auth

Lanjutkan dengan perintah untuk generate asset menggunakan npm:

* npm install && npm run dev

Jika terdapat tulisan NPM ERRR maka jalankan perintah berikut :

* npm install resolve-url-loader@^4.0.0 --save-dev --legacy-peer-deps

lalu ulangi perintah npm:

* npm install && npm run dev

Buat tabel login dengan perintah migrate, jalankan perintah berikut di command line

* php artisan migrate

Jalankan laravel dengan php artisan serve dan pastikan sudah ada halaman login dan register

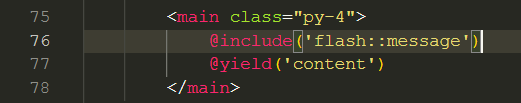
Install packages flash notification agar kita mudah menampilkan pesan,

* composer require laracasts/flash

Buka file app.blade.php lalu ketik / copy paste perintah sebelum @yield

* @include('flash::message')

Sehingga menjadi seperti berikut :



Gambar 4 File app.blade.php

# Percobaan Register

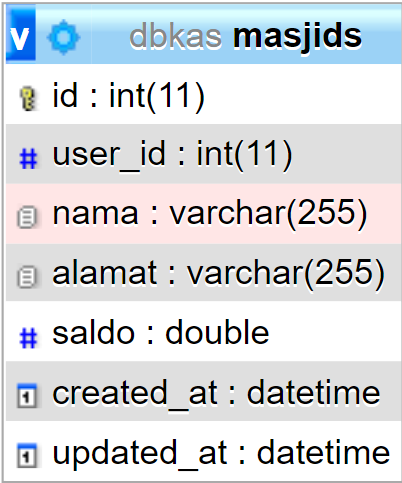
Lakukan percobaan register pada laravel.

email : operator@gmail.com

password : adminadmin

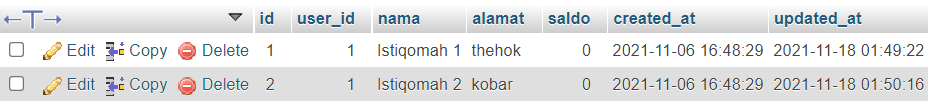
# Tampil data masjid

Buka phpmyadmin, pilih database dbkasmasjid lalu buat tabel dengan keterangan berikut:



Gambar 5 phpMyAdmin tabel masjids

Lalu di menu insert (bagian atas) isi sampel data seperti berikut :



Gambar 6 phpMyAdmin insert sampel data

## Membuat Model Masjid

Di laravel, model digunakan untuk berinteraksi dengan database, 1 file model akan mewakili 1 tabel yang ada di database. Misalnya kita punya tabel dengan nama **masjids** maka nama model kita menjadi **Masjid** (Camelcase).

Kita dapat membuat model dengan mudah menggunakan perintah-perintah yang dijalankan di command line laravel. Silahkan buka command line di vscode menggunakan Ctrl+Shift+C lalu jalankan perintah berikut:

* php artisan make:model Masjid

Di laravel versi >= 8 perintah diatas akan membuat file dengan nama Masjid.php di folder app\Models sedangkan di php <= 7 file Masjid.php ada di folder app

## Membuat MasjidController resource

* php artisan make:controller MasjidController --resource

## Membuat Route Resource

Buka file web.php lalu tambahkan kode berikut dibagian paling bawah:

*Route*::resource('masjid', *MasjidController*::class);

Arahkan cursor ke MasjidController lalu tekan Ctrl+Alt+I agar class MasjidController terimport secara otomatis.

## Membuat Menu Data Masjid

Buka file app.blade.php lalu buat menu di baris ul navbar-nav

<ul class="navbar-nav mr-auto">

    @auth

    <li>

        <a href="{{ route('masjid.index', []) }}" class="nav-link">

Data Masjid

</a>

    </li>

    @endauth

</ul>

## Membuat fungsi index di MasjidController

Buka file MasjidController.php lalu pastekan kode berikut diantara kurawal class

public *function* index()

    {

        $data['models'] = *Masjid*::where('user\_id', *Auth*::user()->id)

            ->latest()

            ->paginate(10);

        return view('masjid\_index', $data);

    }

## Membuat view masjid\_index.blade.php

Buat file view dengan nama masjid\_index.blade.php lalu tambahkan kode berikut:

@extends('layouts.app')

@section('content')

<div class="container">

    <div class="row justify-content-center">

        <div class="col-md-12">

            <div class="card">

                <div class="card-header">DATA MASJID YANG ANDA URUS</div>

                <div class="card-body">

                    <a href="{{ route('masjid.create', []) }}" class="btn btn-info mb-2">

                        <i class="fas fa-file"></i> Tambah Data</a>

                    <table class="table table-bordered">

                        <thead class="table-dark">

                            <tr>

                                <th>NO</th>

                                <th>NAMA</th>

                                <th>ALAMAT</th>

                                <th>TGL BUAT</th>

                                <th>AKSI</th>

                            </tr>

                        </thead>

                        <tbody>

                            @forelse ($models as $item)

                            <tr>

                                <td>{{ $loop->iteration }}</td>

                                <td>{{ $item->nama }}</td>

                                <td>{{ $item->alamat }}</td>

                                <td>{{ $item->created\_at->format('d F Y') }}</td>

                                <td class="align-right">

                                    {!! Form::open(['route' => ['masjid.destroy', $item->id],'method' => 'DELETE','onsubmit' => "return confirm('Anda Yakin ingin menghapus data ini ?')"]) !!}

                                    <a href="{{ route('masjid.show', $item->id) }}" class="btn btn-primary">

                                        <i class="fas fa-edit"></i> Detail </a>

                                    <a href="{{ route('masjid.edit', $item->id) }}" class="btn btn-info ml-2">

                                        <i class="fas fa-edit"></i> Edit</a>

                                    <button type="submit" class="btn btn-danger ml-2"><i class="fa fa-trash"></i> Delete</button>

                                    {!! Form::close() !!}

                                </td>

                            </tr>

                            @empty

                            <tr>

                                <td colspan="4">Data tidak ada</td>

                            </tr>

                            @endforelse

                        </tbody>

                    </table>

                    {!! $models->links() !!}

                </div>

            </div>

        </div>

    </div>

</div>

@endsection

# Tambah data masjid

* 1. Membuat fungsi create di MasjidController
  2. Membuat view masjid\_form.blade.php
  3. Membuat Form tambah masjid
  4. Membuat Fungsi store di MasjidController
  5. Membuat kode simpan

# Menambahkan font awesome | icon

* 1. Mendownload font awesome
  2. Menambahkan font awesome ke app.blade.php

# Edit Data masjid

* 1. Membuat menu Edit di masjid\_index.blade.php
  2. Membuat fungsi edit di MasjidController
  3. Membuat fungsi update di MasjidController
  4. Membuat fungsi update
  5. Membuat kode update

# Hapus data

* 1. Membuat menu hapus di masjid\_index.blade.php
  2. Membuat fungsi destroy di MasjidController
  3. Membuat kode hapus

# Membuat route group

* 1. Menggunakan middleware
  2. Menggunakan prefix

# Scope dan authorize

* 1. Membahas tentang authorize
  2. Membuat scope data masjid

# Update Profil user

* 1. Membuat UserController resource
  2. Membuat Route Resource masjid
  3. Membuat fungsi edit
  4. Membuat view user\_form.blade.php
  5. Membuat fungsi update
  6. Membuat kode update

# Masjid show

* 1. Membuat menu detail di masjid\_index.blade.php
  2. Membuat fungsi show
  3. Membuat view masjid\_show.blade.php
  4. Menampilkan Data masjid

# Session dan Form transaksi, plugin jquery

* 1. Membuat table transaksi di database
  2. Membuat model Transaksi
  3. Membuat TransaksiController
  4. Membuat Route resource transaksi
  5. Membuat menu Input Transaksi di masjid\_show.blade.php
  6. Membuat fungsi create di TransaksiController
  7. Membuat form transaksi
  8. Menggunakan Plugin JQuery Mask untuk format rupiah

# Simpan transaksi

* 1. Membuat fungsi store
  2. Membuat kode simpan transaksi
  3. Update saldo