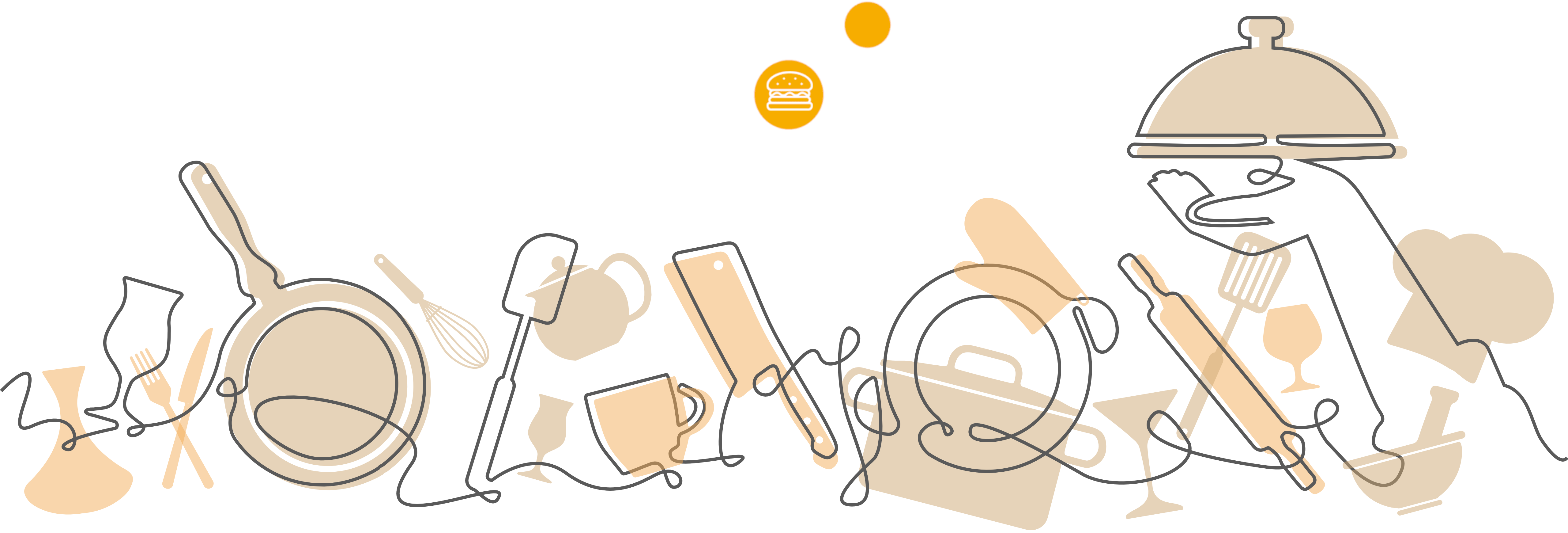
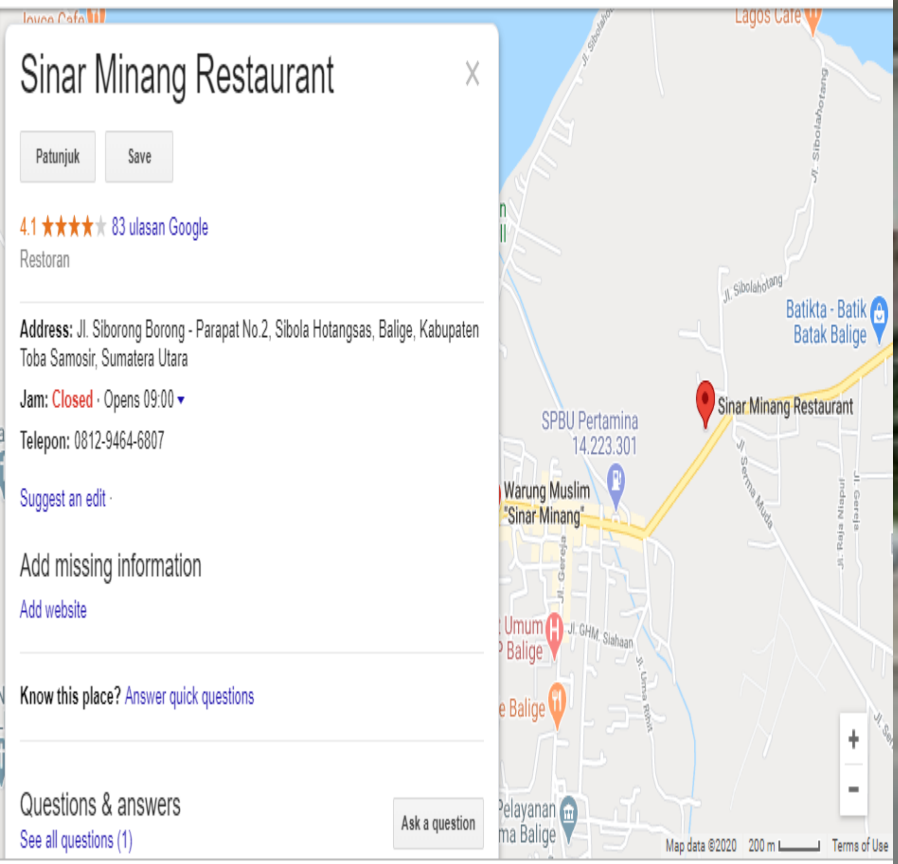


Information system of Rumah Makan Sinar Minang Balige

Created by Kelompok 02



Sinar Minang Restaurant

Patunjuk Save

4.1 ★★★★★ 83 ulasan Google

Restoran

Address: Jl. Siborong Borong - Parapat No.2, Sibola Hotangsas, Balige, Kabupaten Toba Samosir, Sumatera Utara

Jam: Closed · Opens 09:00 ▾

Telepon: 0812-9464-6807

Suggest an edit ·

Add missing information

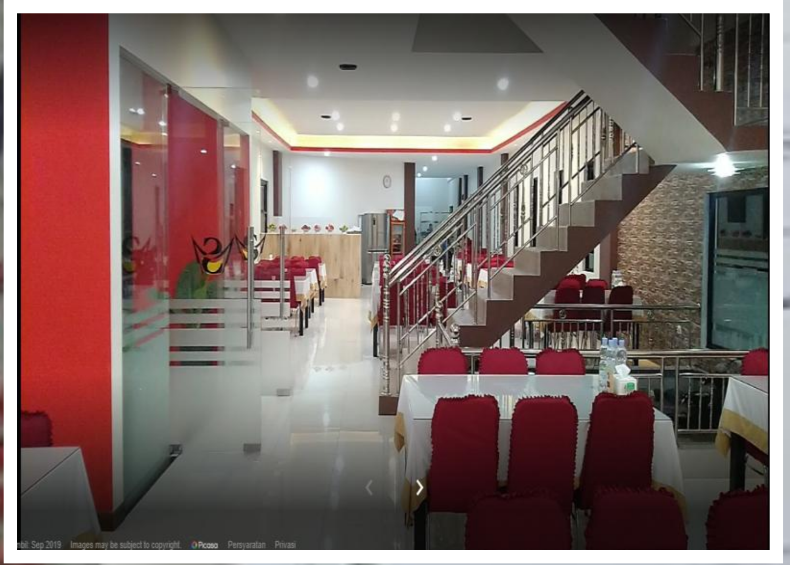
Add website

Know this place? Answer quick questions

Questions & answers

See all questions (1)

Ask a question



Current System

1 Customer used to spend long time to order food

2 Food took long to be prepare

3 Staff difficult to handle a lot of customer

4 Food is wasted

Cooking...

Benefit of New System

1 | Business process much easier and faster

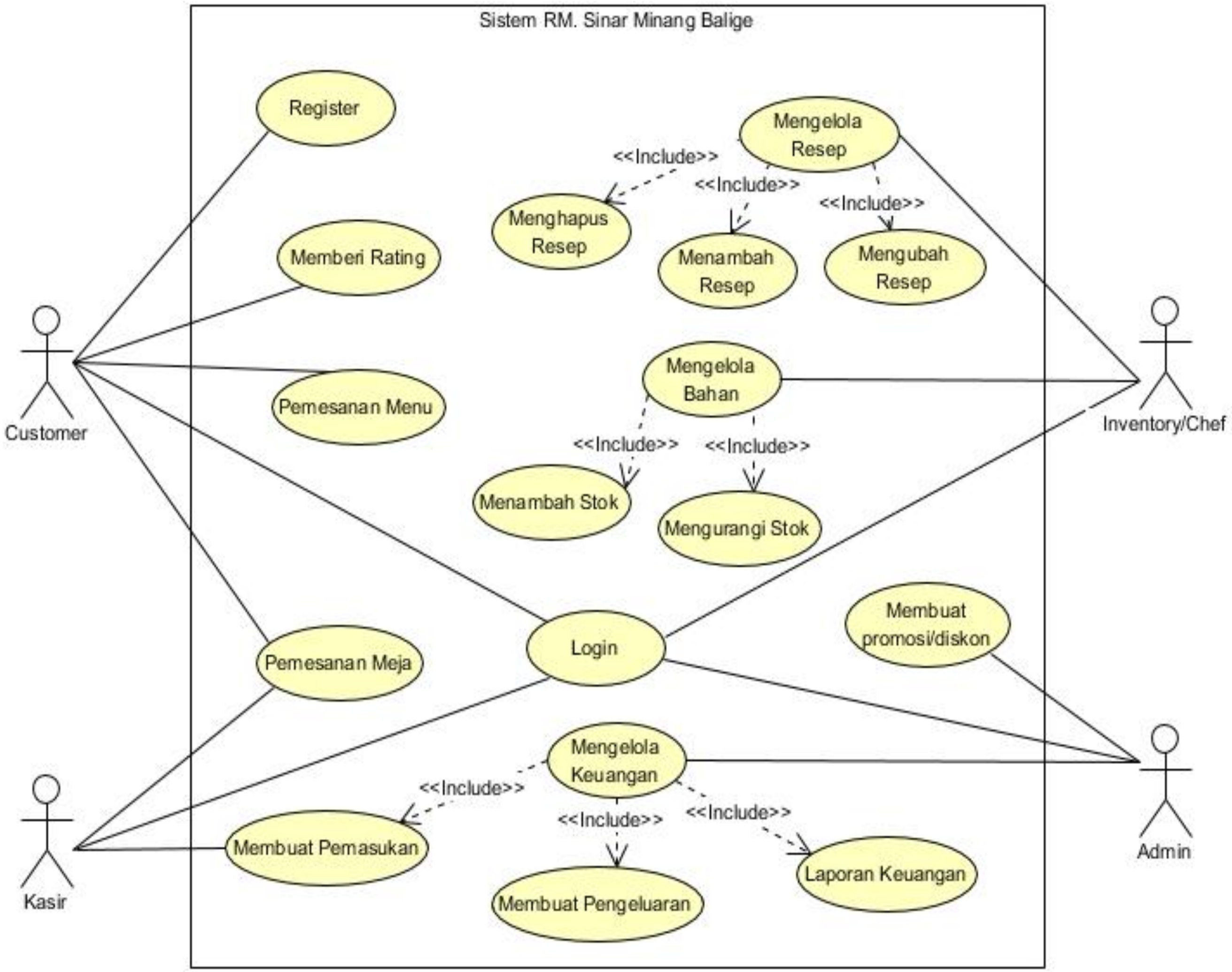
2 | Enable and courage online order

3 | Make it easy to make reservation

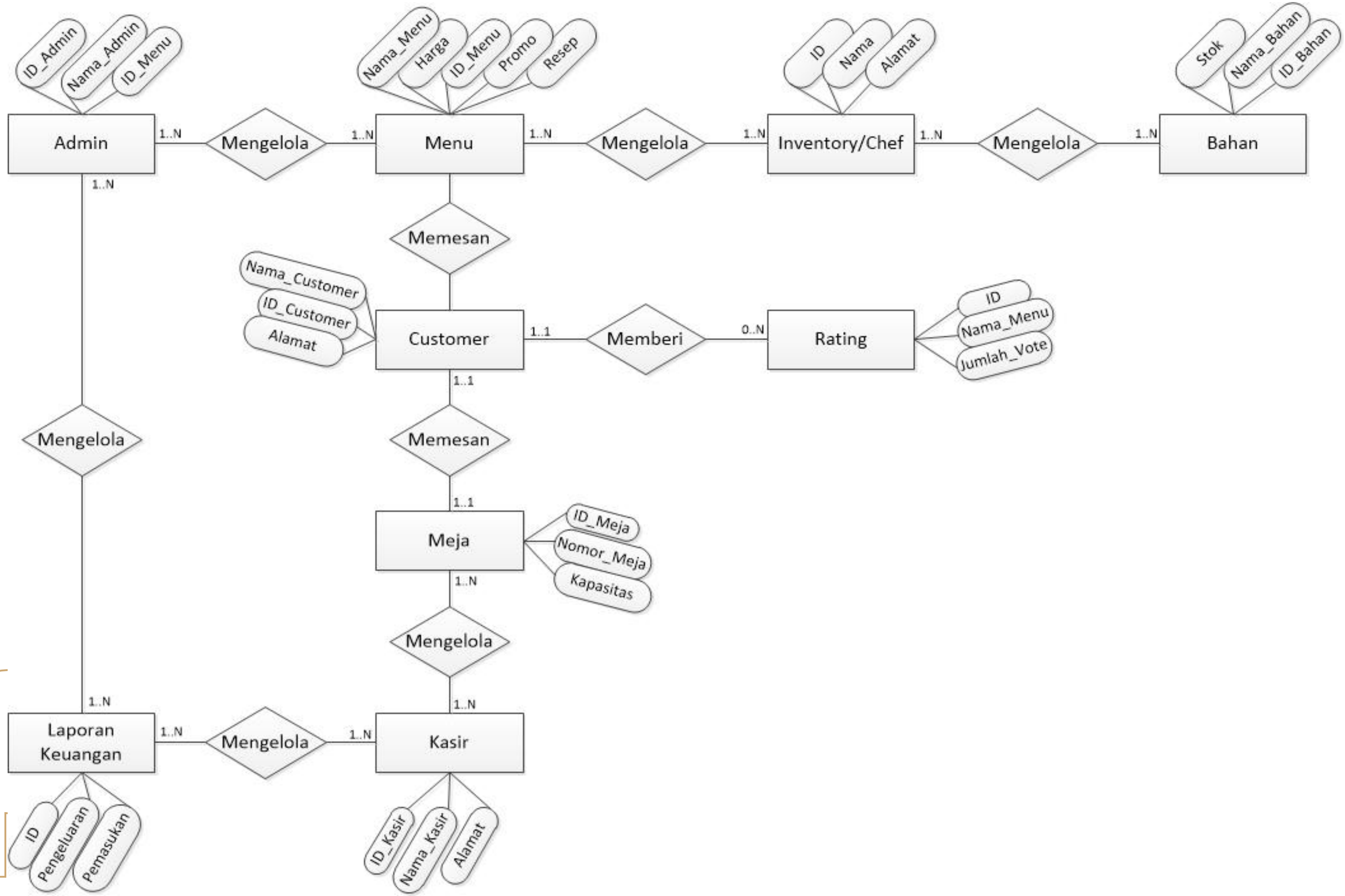
4 | Grow up the business

5 | Recognize inventory system

Use Case Diagram Sistem Informasi Rumah Makan Sinar Minang



Entity Relationship Diagram



The Succedd and Fail Function

Succedd Function

General:

Login & Logout
Menu

Customer:

Register
Gallery
Add to cart
Table Reservation

Inventory/Chef:

Recipe Create, View, Delete
Show Ingredients
Edit Inventory Stock

Cashier:

Show Booked Table
Show Income

Admin

Promo Edit, View
Register employees

Fail Function

Customer:

Cancel reservation

Total Cart

Inventory/Chef:

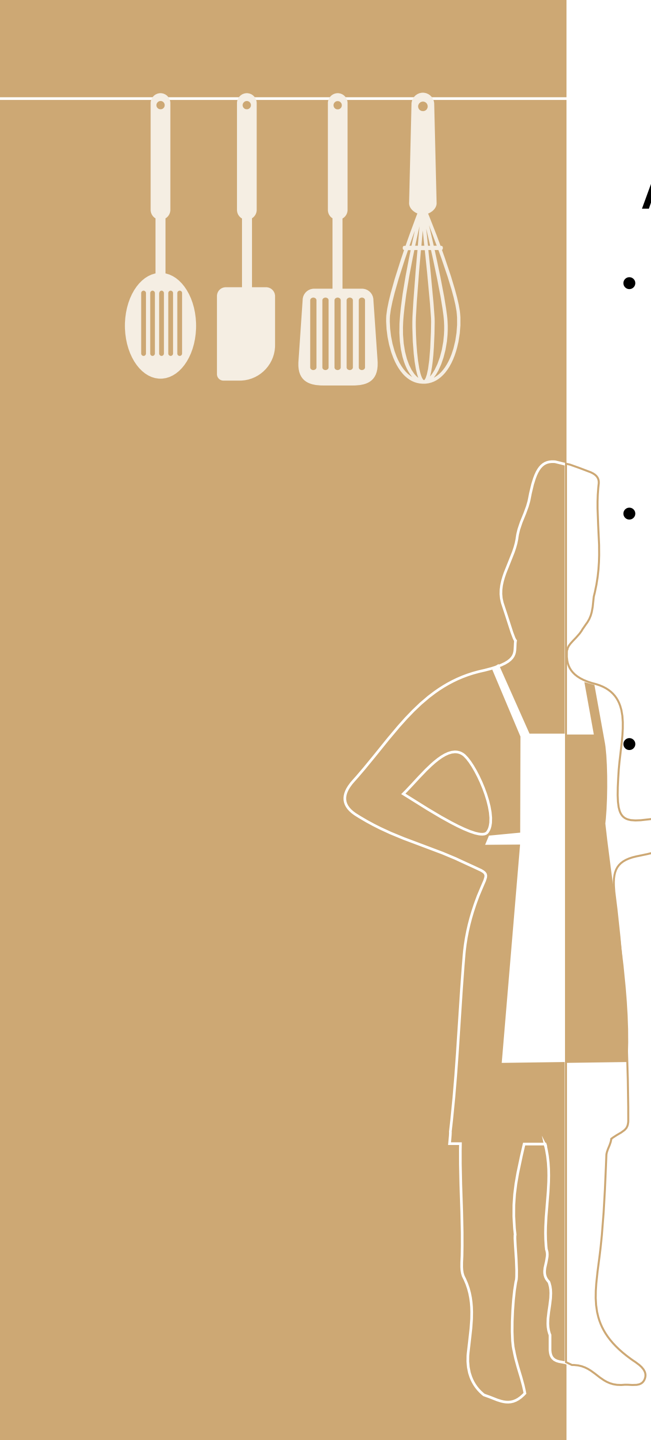
Upload Recipe Picture

Cashier:

Expenses

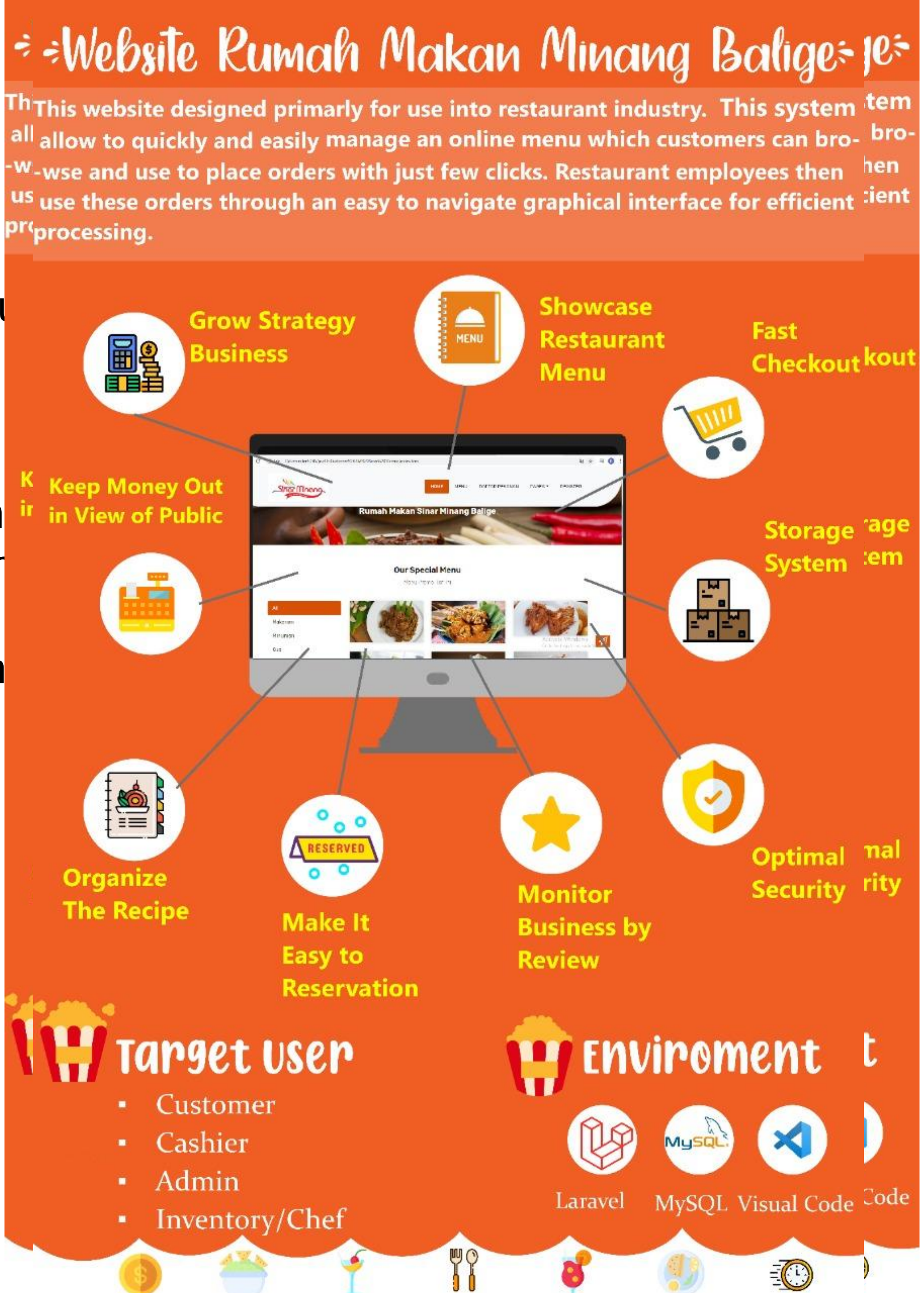
Admin:

Report



Application Architecture

- In Information system, application architecture is one of several architectu domains that form the pillars of an enterprise architecture.
- Application architecture describes the behavior of application used in a busin focused on how they interact with each other and with users.
- In laravel, there are several componen that support application architecture.
  1. Routes
  2. Controllers
  3. Views
  4. Models



Website Rumah Makan Minang Balige

This website designed primarly for use into restaurant industry. This system allow to quickly and easily manage an online menu which customers can browse and use to place orders with just few clicks. Restaurant employees then use these orders through an easy to navigate graphical interface for efficient processing.

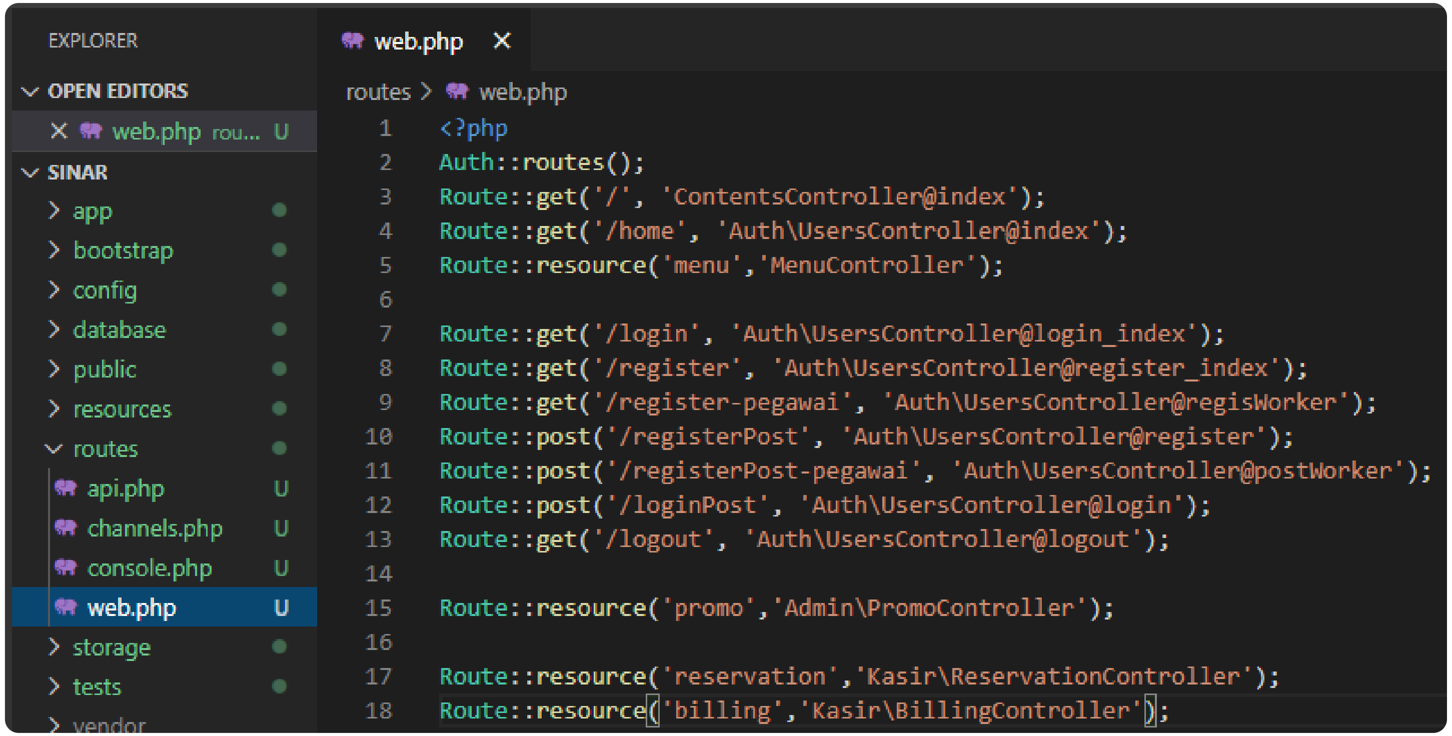
Target user

- Customer
- Cashier
- Admin
- Inventory/Chef

Enviroment

Routes

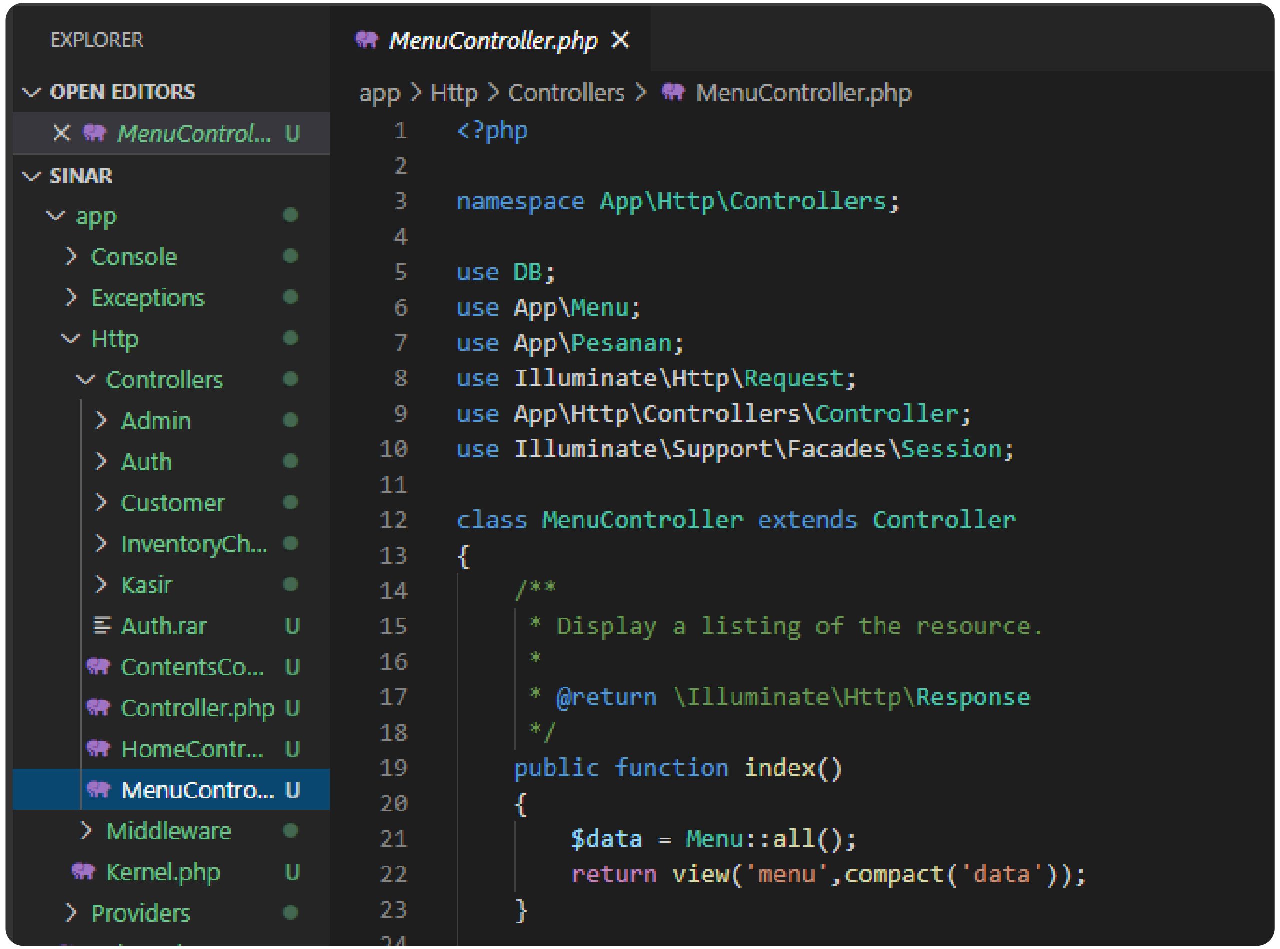
- Route is a function which manage the mapping between URL and controller.
- In Laravel, route written in a file called web.php
- The directory location is in:
  Route > Web.php



```php
<?php
Auth::routes();
Route::get('/', 'ContentsController@index');
Route::get('/home', 'Auth\UsersController@index');
Route::resource('menu','MenuController');

Route::get('/login', 'Auth\UsersController@login_index');
Route::get('/register', 'Auth\UsersController@register_index');
Route::get('/register-pegawai', 'Auth\UsersController@regisWorker');
Route::post('/registerPost', 'Auth\UsersController@register');
Route::post('/registerPost-pegawai', 'Auth\UsersController@postWorker');
Route::post('/loginPost', 'Auth\UsersController@login');
Route::get('/logout', 'Auth\UsersController@logout');

Route::resource('promo','Admin\PromoController');

Route::resource('reservation','Kasir\ReservationController');
Route::resource('billing','Kasir\BillingController');
```

Controllers

- Controller contains methods for commands that mush be performed on the method
- After the route connects controller to the method, then method will return the value for destination URL to go to
- In laravel, the controller directory is in:
  app > Http > Controllers > FileName.php

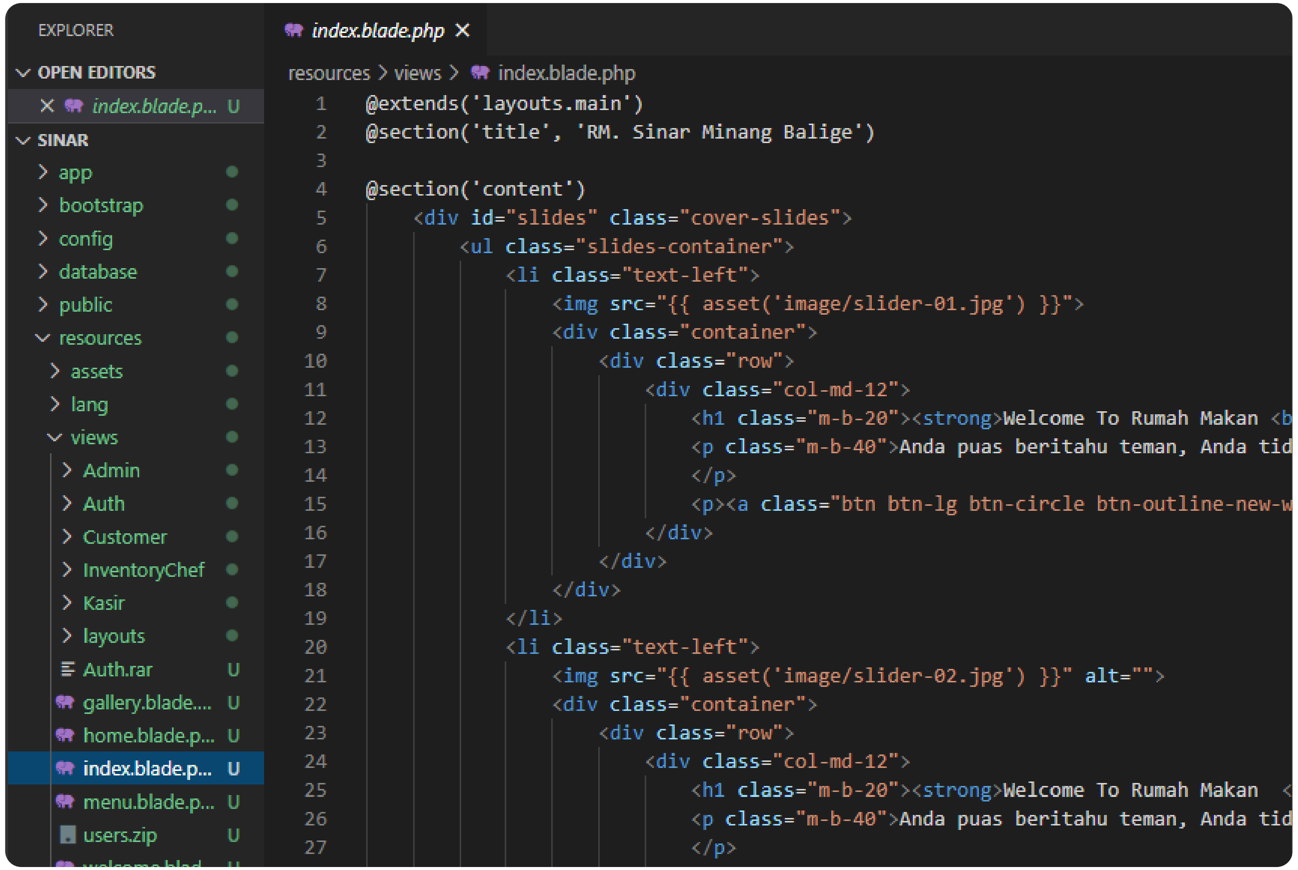


```php
<?php

namespace App\Http\Controllers;

use DB;
use App\Menu;
use App\Pesanan;
use Illuminate\Http\Request;
use App\Http\Controllers\Controller;
use Illuminate\Support\Facades\Session;

class MenuController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $data = Menu::all();
        return view('menu',compact('data'));
    }
```

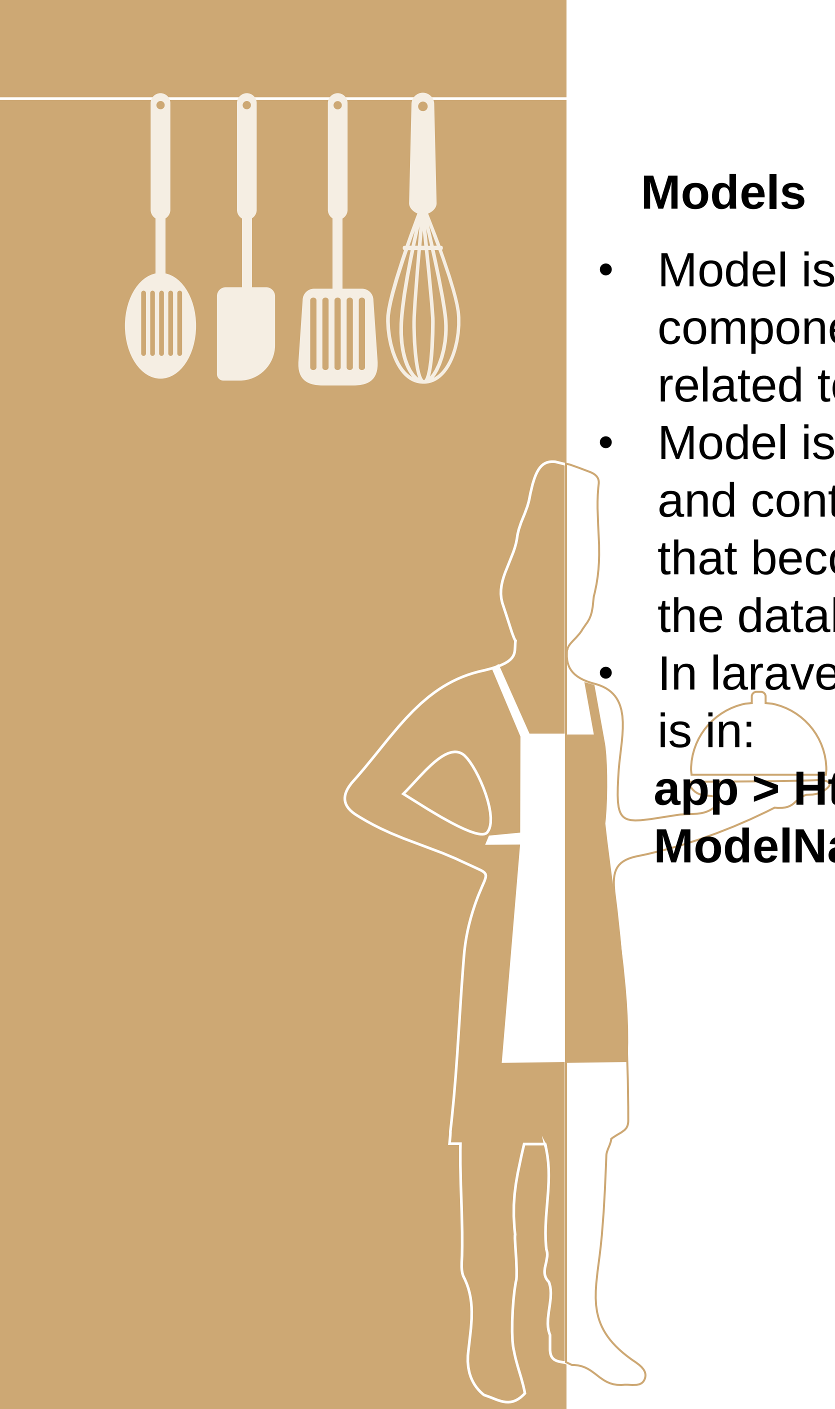
Views

- View is a file contains code that will display the web design
- In laravel, the view directory is inside: **Resources > Views > ViewName.php**

- View can use blade function to make it easier. **Resources > Views > View.blade.php**



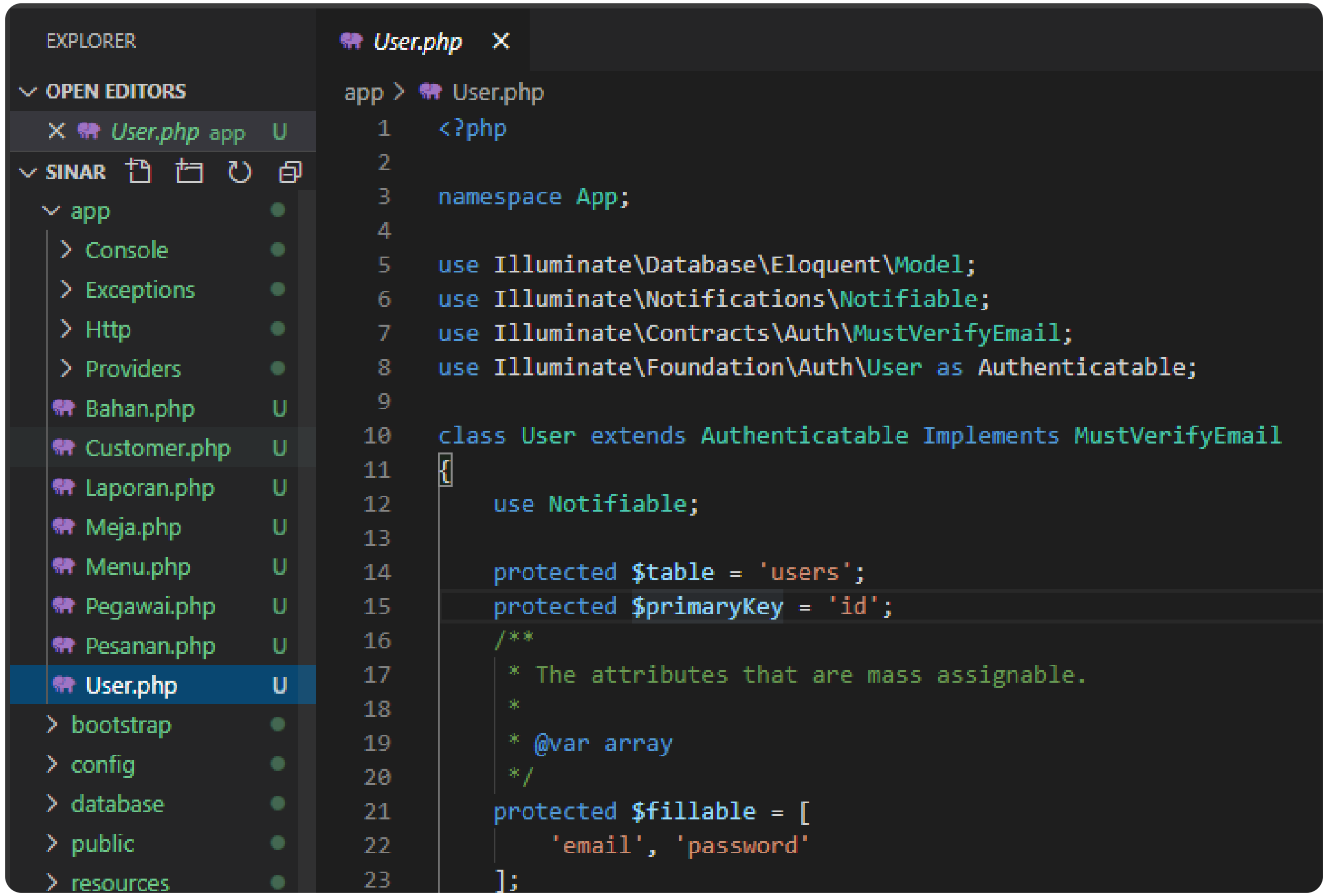
```
@extends('layouts.main')
@section('title', 'RM. Sinar Minang Balige')

@section('content')
    <div id="slides" class="cover-slides">
        <ul class="slides-container">
            <li class="text-left">
                <img src="{{ asset('image/slider-01.jpg') }}">
                <div class="container">
                    <div class="row">
                        <div class="col-md-12">
                            <h1 class="m-b-20"><strong>Welcome To Rumah Makan <b
                            <p class="m-b-40">Anda puas beritahu teman, Anda tid
                            </p>
                            <p><a class="btn btn-lg btn-circle btn-outline-new-w
                        </div>
                    </div>
                </div>
            </li>
            <li class="text-left">
                <img src="{{ asset('image/slider-02.jpg') }}" alt="">
                <div class="container">
                    <div class="row">
                        <div class="col-md-12">
                            <h1 class="m-b-20"><strong>Welcome To Rumah Makan  <
                            <p class="m-b-40">Anda puas beritahu teman, Anda tid
                            </p>
```



Models

- Model is one of the component that is directly related to the database
- Model is presented tables and contains attributes that become columns in the database table
- In laravel, model directory is in:

app > Http > ModelName.php



```php
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Notifications\Notifiable;
use Illuminate\Contracts\Auth\MustVerifyEmail;
use Illuminate\Foundation\Auth\User as Authenticatable;

class User extends Authenticatable Implements MustVerifyEmail
{
    use Notifiable;

    protected $table = 'users';
    protected $primaryKey = 'id';
    /**
     * The attributes that are mass assignable.
     *
     * @var array
     */
    protected $fillable = [
        'email', 'password'
    ];
```

SPECIAL FOOD

Thank you

Link presentasi : https://youtu.be/1TzgjfjSS-g

Link demo : https://youtu.be/jD5zw9EzYpQ