# Kadoorie Website

A Laravel-based web application built with Laravel Livewire, Filament Admin Panel, and modern development tools.

## Requirements

- Docker Desktop
- Docker Compose
- Git

## Tech Stack

- Laravel 12.x with Sail
- Livewire with Flux and Volt
- Filament 3.x Admin Panel
- TailwindCSS 4.x
- Vite 6.x
- PostgreSQL 17
- Redis
- Mailpit (for local email testing)

## Getting Started

1. Clone the repository:

   ```
   git clone <repository-url> cd kadoorie-website
   ```

2. Create a copy of the environment file:

   ```
   cp .env.example .env
   ```

3. Set up Composer authentication:

   ```
   cp auth.json.example auth.json
   ```

   Then edit the `auth.json` file with your premium plugin license credentials:

   - **Media Library Pro V4** (€99.00) - Purchase
   - **Record Finder Pro** (€59.00) - Purchase

   Add your license credentials:

   ```
   { "http-basic": { "satis.ralphjsmit.com": { "username":
   "your-license-username", "password": "your-license-key" } } }
   ```

   **Note**: The username is usually your email address, and the password is your license key from the purchase confirmation email. See

DEPLOYMENT.md for detailed instructions. Once you have purchased both you should only need to use one Licence key.

4. Install Composer dependencies using a temporary container:

   ```
   docker run --rm \ -u "$(id -u):$(id -g)" \ -v "$(pwd):/var/www/html" \ -w /var/www/html \ laravelsail/php84-composer:latest \ composer install --ignore-platform-reqs
   ```

5. Start the Docker containers:

   ```
   ./vendor/bin/sail up -d
   ```

6. Generate application key:

   ```
   ./vendor/bin/sail artisan key:generate
   ```

7. Install Node.js dependencies:

   ```
   ./vendor/bin/sail npm install
   ```

8. Build assets:

   ```
   ./vendor/bin/sail npm run build
   ```

9. Run database migrations:

   ```
   ./vendor/bin/sail artisan migrate
   ```

□ **Tip**: You can create a shell alias for sail to make commands shorter:

```
alias sail='[ -f sail ] && sh sail || sh vendor/bin/sail'
```

# Development

## Available Commands

- `sail artisan` - Run Laravel Artisan commands
- `sail composer` - Run Composer commands
- `sail npm` - Run NPM commands
- `sail npm run dev` - Start Vite development server
- `sail npm run build` - Build production assets
- `sail npm run build-filament-theme` - Build Filament admin theme
- `sail test` - Run tests
- `sail artisan migrate:fresh --seed` - Reset and seed database

## Docker Services

The following services are available:

- **Web Server**: `http://oxtec.test`
- **PostgreSQL**: Port 5432
- **Redis**: Port 6379

- **Mailpit**: Port 1025 (SMTP) and 8025 (Web Interface)

## Common Tasks

### Development services

To get up and running with all the development services needed run

`sail artisan solo`

this will run vite, queue and other needed services

you can edit the services run but editing `config/solo.php`

Current running tools

- Logs - View the laravel.log file
- Vite - Runs the front end toolings
- Queue - Runs all queue and displays output
- Dumps (Lazy) - Will catch and display any `dump` in code
- Pint (Lazy) - Will run code through PINT to show any coding misteps in code sytles
- Test (Lazy) - Run Test Suite
- Make - simple tool to allow making laravel code.

- Lazy means that they don't auto start and you have to manual start by going to the the tab in question and hiting `s`

### Working with the Database

```
# Fresh migration with seeders sail artisan migrate:fresh --seed
# Run specific seeder sail artisan db:seed --class=UserSeeder
```

### Cache Management

```
# Clear various caches sail artisan cache:clear sail artisan
config:clear sail artisan route:clear sail artisan view:clear
```

### Maintenance Mode

```
# Enable maintenance mode sail artisan down # Disable maintenance
mode sail artisan up
```

# Production Deployment

For production deployment instructions, please refer to the comprehensive deployment guide:

- **DEPLOYMENT.md** - Complete deployment guide with instructions for:
  - Laravel Forge deployment (recommended)

    - Manual VPS/shared hosting deployment
    - Environment configuration
    - Database setup
    - Security configuration
    - Maintenance procedures

- **DEPLOYMENT_CHECKLIST.md** - Quick reference checklists for deployment steps

## Quick Start for Production

1. Review the DEPLOYMENT.md guide
2. Choose your deployment method (Laravel Forge recommended)
3. Follow the step-by-step instructions
4. Use the DEPLOYMENT_CHECKLIST.md to track progress

## Production Requirements

- PHP 8.4+
- PostgreSQL 17
- Redis 7.x+ (or the drop in replacement ValKey)
- Node.js 22+
- Nginx or Apache web server
- SSL certificate (Let's Encrypt recommended)

## Recommended Server Specifications

**Minimum Recommended (Starter)**:

- RAM: 2 GB
- vCPUs: 2
- Storage: 60 GB SSD
- Bandwidth: 3,000 GB/month
- Example: DigitalOcean Basic Droplet ($18/month)

**Recommended for Production**:

- RAM: 4 GB
- vCPUs: 2-4
- Storage: 80-100 GB SSD
- Bandwidth: 4,000-5,000 GB/month
- Example: DigitalOcean Regular Droplet ($24-48/month)

See DEPLOYMENT.md for detailed specifications and scaling recommendations.

# Testing

Run the test suite using:

```
sail test
```