Amar Seva Aashram Web Application Documentation

Project Overview

The **Amar Seva Aashram** web application is designed to support the organization's mission by providing a platform for managing key content and user interactions. Built using the Laravel framework, it offers a secure, scalable solution with a simplified set of features compared to similar projects.

- **Organization**: Amar Seva Trust
- **Domain**: amarseva.dreamwebies.com
- **Start Date**: 18 December 2024
- **Launch Date**: 4 February 2025 (Planned)

- **Status**: Under Development (as of 18 March 2025)

Technical Details

Framework and Versions

- **Framework**: Laravel
- **Laravel Version**: 10.48.25
- **PHP Version**: (Not specified, assumed 8.1.4 or compatible with Laravel 10; confirm if different)
- **MySQL Version**: 8.0.35
- **JavaScript/TypeScript Version**: 2.4 (Likely Node.js or related dependency; clarify if needed)

Hosting and Deployment

- **Domain**: amarseva.dreamwebies.com

- **Login URL**: amarseva.dreamwebies.com/login
- **Default Admin Credentials**:
  - **Ask to admin**

Development Environment Setup

To set up the project locally:

1. **Prerequisites**:
   - PHP (8.1.4 or compatible with Laravel 10)
   - Composer
   - MySQL 8.0.35

Clone the Repository:
bash

```
git clone https://github.com/shubhambhorge4321/amar-seve-ashram.git
```

2.
```
cd amar-seve-ashram
```

Install Dependencies:
bash

```
composer install
```

3.
```
npm install
```

4. **Environment Configuration**:
   ○ Copy .env.example to .env:
   bash

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

Update .env with database credentials and app settings:
env
CollapseWrapCopy

```
APP_NAME="Amar Seva Aashram"
APP_URL=http://amarseva.dreamwebies.com
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=<your-database-name>
DB_USERNAME=<your-username>
```

- `DB_PASSWORD=<your-password>`

5. **Generate Application Key**:
bash

```
php artisan key:generate
```

6. **Run Migrations**:
bash

```
php artisan migrate
```

7. **Serve the Application**:
bash

```
php artisan serve
```

Access at http://localhost:8000 (or configured URL).

Modules

1. Login Module

- **Description**: Handles admin authentication.
- **Features**: Admin login with email and password.
- **Route**: amarseva.dreamwebies.com/login
- **Status**: Implemented

2. Banner Module

- **Description**: Manages banners displayed on the website. Admins can add, delete, and view banners.
- **Features**:
  - **Add Banner**: Upload a banner with an image, heading, subheading, and page association.
  - **Delete Banner**: Remove a banner.

- ○ **View Banner**: Display banner details.
- ○ **Manage Banners**: Table view with fields:
    - ■ **ID**: Unique identifier.
    - ■ **Image**: Banner image.

    - ■ **Page**: Display page (e.g., "home").
    - ■ **Created Date**: Date added.
    - ■ **Action**: Delete or view options.

3. Volunteer Module

- ● **Description**: Manages volunteer profiles. Admins can add, delete, and view volunteer details.
- ● **Features**:

- **Add Volunteer**: Create a profile with name, designation, photo, and social media links.
- **Delete Volunteer**: Remove a profile.
- **View Volunteer**: Display profile details.

- **Database Table** (Suggested):
  - **Table Name**: volunteers

- **Status**: Development

4. Gallery Module:

- **Description**: Manages a photo gallery. Admins can upload and delete photos, viewable on the frontend.
- **Features**:

- ○ **Add Photo**: Upload a photo.
- ○ **Delete Photo**: Remove a photo.
- ○ **View Photo**: Display photo details.
- ○ **Manage Gallery**: Table view with fields:

    - ■ **Photo**: Uploaded image.
    - ■ **Created Date**: Date added.
    - ■ **Action**: Delete or view options.
- ○ **Frontend Display**: Photos shown on /gallery with configurable widths (e.g., 300px, 400px, 500px, 630px).
- ● **Routes** (Assumed):
  - ○ /gallery (Admin: List photos)
  - ○ /gallery/add (Admin: Add photo)

- **Database Table** (Suggested):
  - **Table Name**: gallery

- **Status**: Development

5. Become a Volunteer Module:

- **Description**: Allows users to register as volunteers via a frontend form. Admins can view and delete submissions.
- **Features**:
  - **Frontend Form**: Submit volunteer registration on /become-volunteer.
  - **View Registration**: Display details in admin panel.
  - **Delete Registration**: Remove a registration.

- ○ **Manage Registrations**: Table view with fields:

  - ■ **Created Date**: Date submitted.
  - ■ **Action**: Delete or view options.
- ● **Routes** (Assumed):
  - ○ /become-volunteer (Frontend: Registration form)

- ● **Status**: Development

6. Donations Module

- **Description**: Enables users to submit donation info via a frontend form. Admins can view and delete records.
- **Features**:
  - **Frontend Form**: Submit donation details on /donations.
  - **View Donation**: Display details in admin panel.
  - **Delete Donation**: Remove a record.
  - **Manage Donations**: Table view with fields:
    - **ID**: Unique identifier.
    - **Amount**: Donation amount (assumed).
    - **Created Date**: Date submitted.
    - **Action**: Delete or view options.

- **Routes** (Assumed):
  - /donations (Frontend: Donation form)

- **Status**: Development

7. Contact Us Module:

- **Description**: Allows users to submit inquiries via a frontend form. Admins can view and delete submissions.
- **Features**:
  - **Frontend Form**: Submit contact form on /contact-us.
  - **View Submission**: Display details in admin panel.
  - **Delete Submission**: Remove a submission.

- ○ **Manage Submissions**: Table view with fields:
    - ■ **ID**: Unique identifier.
    - ■ **Name**: Submitter's name (assumed).
    - ■ **Created Date**: Date submitted.
    - ■ **Action**: Delete or view options.
- ● **Routes** (Assumed):
    - ○ /contact-us (Frontend: Contact form)

- ● **Status**: Development

- Compared to the Yuva Parivartan app, this version omits modules like Event, Exhibition, and Dashboard, focusing on a leaner feature set.
- Frontend structure mirrors the previous app but with fewer pages.
- Assumed fields (e.g., name, amount) are placeholders—provide exact form fields for accuracy.

Next Steps

1. **Confirm Fields**: Specify actual form fields for Become a Volunteer, Donations, and Contact Us (e.g., email, phone).
2. **Refine Routes**: Verify if the assumed routes match your implementation.

3. **Additional Details**: Mention any frontend specifics (e.g., CSS framework) or admin panel features.

Let me know if you'd like to adjust anything or add more details!