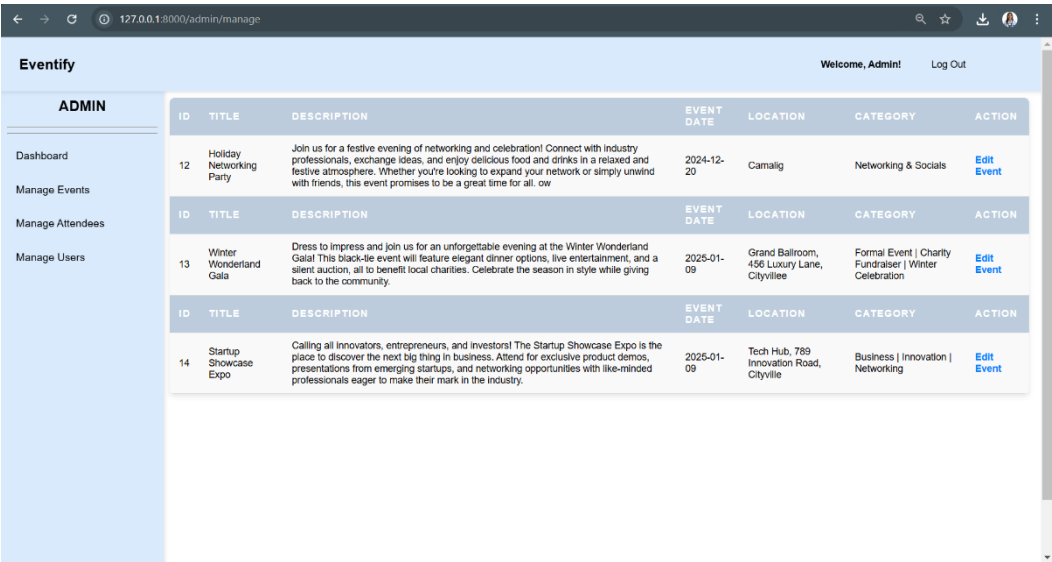


# Making Use of Our Final Eventify System as a Model for Documentation Requirements

## Part 1: Create a database and create a connection to it.

Create controllers that loads the dashboard, feed, or equivalent pages.

```
Http > Controllers > AdminController.php > AdminController
class AdminController extends Controller
{
    1 reference | 0 overrides
    public function show(): View
    {
        $events = Event::all();
        return view('admin.AdminManageEvent', data: compact('events'));
    }
}
```



The show method is to load data from the database and display it on a specific view. It uses the Event::all() function to fetch all the records from the events table, ensuring that the view has access to the complete dataset. The data is then passed to the view (admin.AdminManageEvent) using the compact('events') function, which makes the \$events variable available in the Blade template. Finally, the view() function renders the specified view, separating the backend logic from the frontend presentation.

Register controllers in routes to link methods to URLs.

```
//admin edit attendees
Route::get('/admin/manage-events', [UpdateAttendees::class, 'index'])-
>name('admin.ManageAttendeesAdmin');
Route::get('/admin/attendees/{id}/edit', [UpdateAttendees::class, 'edit'])-
>name('admin.attendees.edit');
Route::put('/admin/attendees/{id}', [UpdateAttendees::class, 'update'])-
>name('admin.attendees.update');
Route::delete('/admin/attendees/{id}', [UpdateAttendees::class, 'destroy'])-
>name('admin.attendees.destroy');
```

You may create and register more controllers if needed.  
= to create controller **php artisan make:controller controllerName**

## Part 2: Make an account registration and login page.

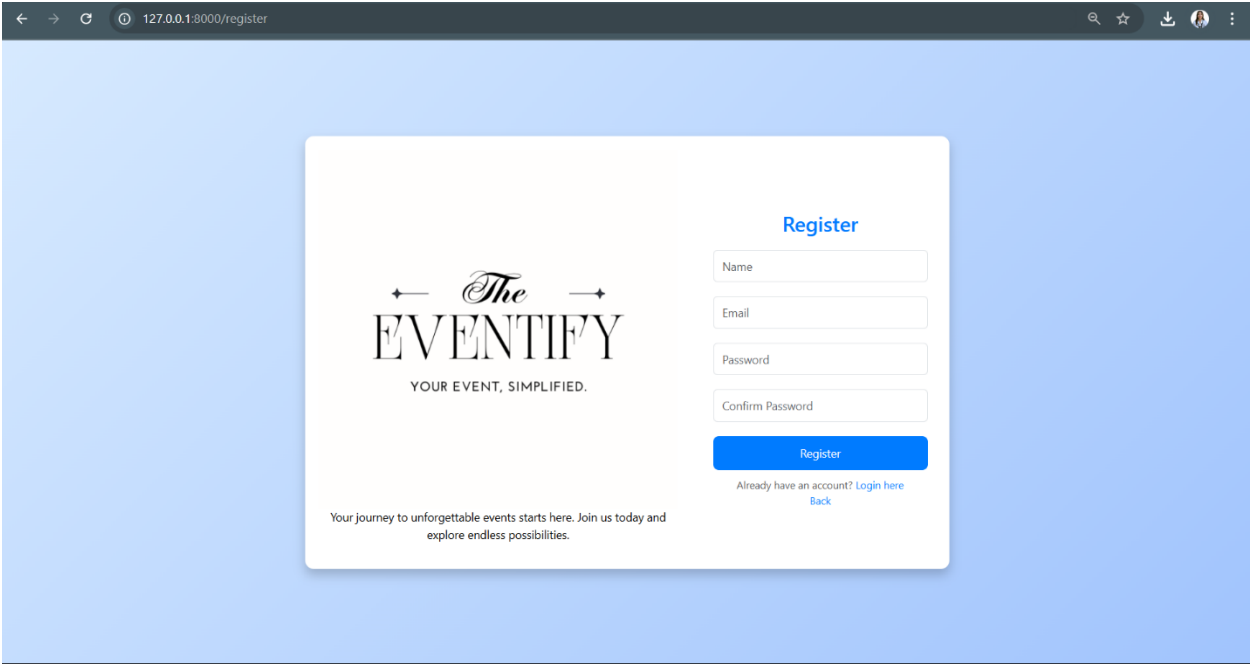
### Registration

```
<body>
  <div class="register-container">
    <!-- Image and Intro Section -->
    <div class="image-section">
      
      <p>Your journey to unforgettable events starts here. Join us today and explore more!</p>
    </div>

    <!-- Form Section -->
    <div class="form-section">
      <h1>Register</h1>
      <form action="{{ url('register') }}" method="POST">
        @csrf
        <input type="text" name="name" placeholder="Name" class="form-control" required />
        <input type="email" name="email" placeholder="Email" class="form-control" required />
        <input type="password" name="password" placeholder="Password" class="form-control" required />
        <input type="password" name="password_confirmation" placeholder="Confirm Password" class="form-control" required />
        <button type="submit">Register</button>
      </form>

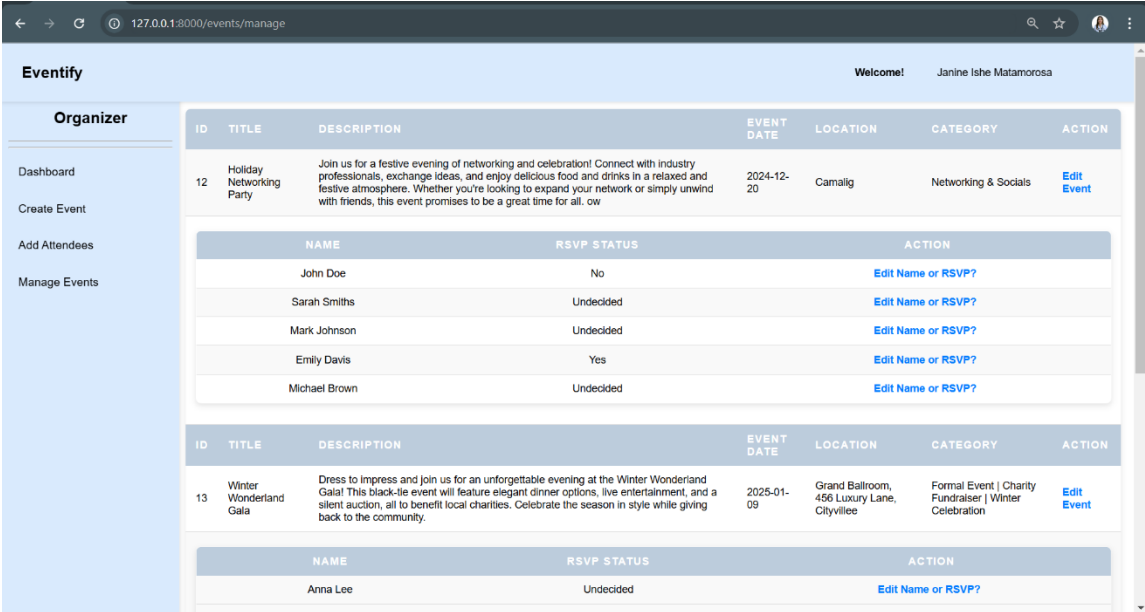
      <div class="footer">
        Already have an account? <a href="{{ url('login') }}">Login here</a>
        <br>
        <a href="javascript:history.back()" class="back-button">Back</a>
      </div>
    </div>
  </div>
</body>
```

This code represents a registration page where users can create an account by providing their name, email, and password.



**Part 3: Populate your pages using data fetched from a database.**  
Populate at least 1 page with content.

Fetch data using a controller and pass it to views.  
Load information such as posts, ratings, comments, pictures, etc.  
Make sure to use a model and a controller.



This code displays a list of events and their attendees using data fetched from a controller and passed to the view. The controller retrieves information about events, including details like ID, title, description, event date, location, and category, and also fetches attendees associated with each event. The events are displayed in a table format, with an "Edit Event" link for modifying event details. For each event, there is a nested table that lists the attendees, showing their names and RSVP status, with an option to edit their information.

Controller:

```
public function index(Request $request)
{
    $query = Event::where('user_id', auth()->id());

    // Check if there's a search query
    if ($request->has('search') && $request->get('search') != '') {
        $search = $request->get('search');

        // Filter events based on title, date, or location
        $query->where(function ($query) use ($search) {
            $query->where('title', 'like', "%{$search}%")
                ->orWhere('event_date', 'like', "%{$search}%")
                ->orWhere('location', 'like', "%{$search}%")
                ->orWhere('category', 'like', "%{$search}%");
        });
    }
    $events = $query->get();
    return view('events.index', compact('events'));
}
```

Prepared by:  
**Janine**

Model:

The model represents the data, handles database interactions, and ensures data integrity.

20 references | 0 implementations | You, 2 hours ago | 1 author (You)

```
class Event extends Model
{
    use HasFactory;

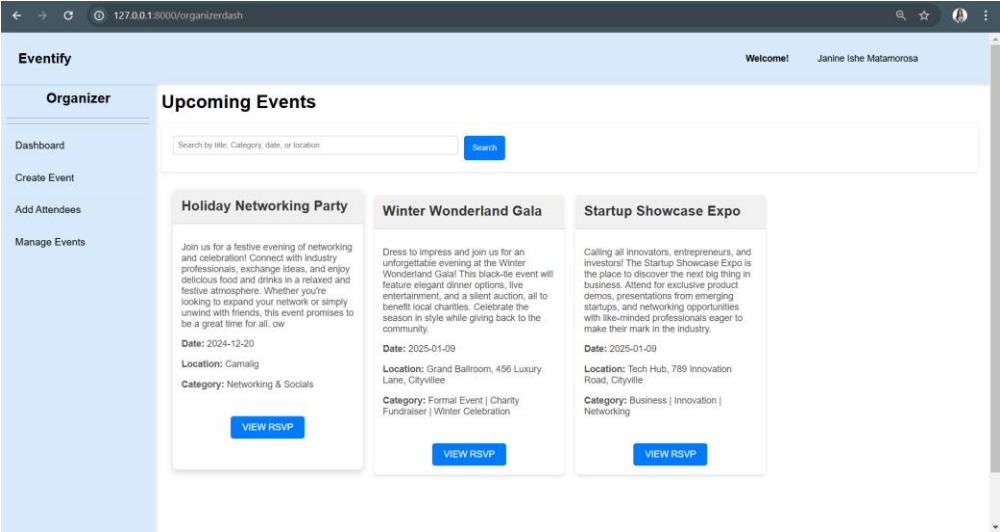
    0 references
    protected $table = 'events';

    0 references
    protected $fillable = [
        'title',
        'description',
        'event_date',
        'location',
        'category',
        'user_id',
    ];

    0 references | 0 overrides
    public function attendees(): HasMany
    {
        return $this->hasMany(related: Attendee::class);
    }
}
```

You, yesterday • log reg create and add attendees show

More data fetch samples:



- the model manages data and logic, the view handles the user interface, and the controller processes user requests, connecting the model and view.

☐

Profiling

[\[ Edit inline \]](#)

[\[ Edit \]](#)

[\[ Explain SQL \]](#)

[\[ Create PHP code \]](#)

[\[ Refresh \]](#)

☐

Show all

Number of rows:

25

Filter rows:

Search this table

Sort by key:

None

Extra options

			id	title	description	event_date	location	category	user_id	created_at	updated_at
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		12	Holiday Networking Party	Join us for a festive evening of networking and ce...	2024-12-20	Camallig	Networking & Socials	1	2024-12-19 14:25:42	2024-12-19 15:15:33
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		13	Winter Wonderland Gala	Dress to impress and join us for an unforgettable ...	2025-01-09	Grand Ballroom, 456 Luxury Lane, Cityvillee	Formal Event   Charity Fundraiser   Winter Celebra...	1	2024-12-19 14:26:50	2024-12-19 15:12:25
<input type="checkbox"/>	<div><div><div><div></div></div><div>Edit</div></div><div><div><div></div></div><div>Copy</div></div><div><div><div></div></div><div>Delete</div></div></div>		14	Startup Showcase Expo	Calling all innovators, entrepreneurs, and investo...	2025-01-09	Tech Hub, 789 Innovation Road, Cityville	Business   Innovation   Networking	1	2024-12-19 14:27:40	2024-12-19 14:27:40

Check all

With selected:

Edit

Copy

Delete

Export

☐

Show all

Number of rows:

25

Filter rows:

Search this table

Sort by key:

None