

## Set up a new Laravel project:

Open terminal or command prompt.

Navigate to the desired directory where you want to create the project.

Run the following command to create a new Laravel project:

```
composer create-project laravel/laravel my_blog
```

## Configure your database:

Open the .env file in the root directory of your project.

Update the DB\_CONNECTION, DB\_HOST, DB\_PORT, DB\_DATABASE, DB\_USERNAME, and DB\_PASSWORD values according to your database setup. Here I used my database name as my\_blog in DB\_DATABASE.

## Create the necessary database tables and models:

Run the following commands to generate migrations for the blog post and comment tables:

```
php artisan make:migration create_posts_table --create=posts
```

```
php artisan make:migration create_comments_table --create=comments
```

Open the generated migration files in the database/migrations directory and define the table schemas for the "posts" and "comments" tables.

Run the migrations to create the database tables:

```
php artisan migrate
```

Run the following commands to generate models for the "Post" and "Comment" tables:

```
php artisan make:model Post
```

```
php artisan make:model Comment
```

## Modifying {TimeStamp}\_create\_posts\_table and {TimeStamp}\_create\_comments\_table:

For create\_posts\_table:

```
<?php

use Illuminate\Database\Migrations\Migration;

use Illuminate\Database\Schema\Blueprint;

use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('posts', function (Blueprint $table) {
            $table->id();

            $table->string('title');

            $table->text('content');

            $table->timestamps();

        });
    }
}
```

**For create\_comments\_table:**

```
<?php
```

```
use Illuminate\Database\Migrations\Migration;
```

```
use Illuminate\Database\Schema\Blueprint;
```

```
use Illuminate\Support\Facades\Schema;
```

```
return new class extends Migration
```

```
{
```

```
    /**
```

```
     * Run the migrations.
```

```
    */
```

```
    public function up(): void
```

```
    {
```

```
        Schema::create('comments', function (Blueprint $table) {
```

```
            $table->id();
```

```
            $table->string('title');
```

```
            $table->text('content');
```

```
            $table->timestamps();
```

```
        });
```

```
    }
```

```
    /**
```

```
     * Reverse the migrations.
```

```
    */
```

```
public function down(): void
{
    Schema::dropIfExists('comments');
}
};
```

**Modifying models for the "Post" and "Comment" tables:**

**For "Post" table:**

```
<?php
namespace App\Models;
use App\Models\Comment;
use Illuminate\Database\Eloquent\Model;
class Post extends Model
{
    protected $fillable = ['title', 'content'];

    public function comments()
    {
        return $this->hasMany(Comment::class);
    }
}
```

### For "Comment" table:

```
<?php

namespace App\Models;

use App\Models\Post;

use Illuminate\Database\Eloquent\Model;

class Comment extends Model
{
    protected $fillable = ['content'];

    public function post()
    {
        return $this->belongsTo(Post::class);
    }
}
```

### Necessary routing:

**Open the routes/web.php file and define the necessary routes:**

```
use App\Http\Controllers\PostController;

use App\Http\Controllers\CommentController;

Route::get('/posts', [PostController::class, 'index'])->name('posts.index');

Route::get('/posts/{id}', [PostController::class, 'show'])->name('posts.show');

Route::post('/comments', [CommentController::class, 'store'])->name('comments.store');
```

## Create the corresponding controllers:

Run the following commands to generate the controllers:

```
php artisan make:controller PostController
```

```
php artisan make:controller CommentController
```

## Modifying the controllers:

### PostController:

```
namespace App\Http\Controllers;  
  
use App\Post;  
  
class PostController extends Controller  
  
{  
  
    public function index()  
  
    {  
  
        $posts = Post::all();  
  
        return view('posts.index', compact('posts'));  
  
    }  
  
    public function show($id)  
  
    {  
  
        $post = Post::findOrFail($id);  
  
        return view('posts.show', compact('post'));  
  
    }  
  
}
```

**CommentController:**

```
namespace App\Http\Controllers;

use App\Comment;

class CommentController extends Controller
{
    public function store(Request $request)
    {
        $comment = new Comment();

        $comment->post_id = $request->post_id;

        $comment->content = $request->content;

        $comment->save();

        return redirect()->back();
    }
}
```

## Creating the views using HTML, CSS, and Bootstrap:

Create the following Blade view files in the resources/views directory:

posts/index.blade.php: The blog post list view.

posts/show.blade.php: The blog post details view.

### Here's the code of the posts/index.blade.php view:

```
@extends('layouts.app')
```

```
@section('content')
```

```
<h1>Blog Post List</h1>
```

```
@foreach ($posts as $post)
```

```
<div>
```

```
<h2>{{ $post->title }}</h2>
```

```
<p>{{ $post->content }}</p>
```

```
<a href="{{ route('posts.show', $post->id) }}">Read More</a>
```

```
</div>
```

```
@endforeach
```

```
@endsection
```



Here's the code of the posts/show.blade.php view:

```
@extends('layouts.app')

@section('content')

    <h1>{{ $post->title }}</h1>

    <p>{{ $post->content }}</p>

    <h2>Comments</h2>

    @foreach ($post->comments as $comment)

        <div>

            <p>{{ $comment->content }}</p>

        </div>

    @endforeach

    <form action="{{ route('comments.store') }}" method="POST">

        @csrf

        <input type="hidden" name="post_id" value="{{ $post->id }}">

        <textarea name="content" placeholder="Write a comment"></textarea>

        <button type="submit">Submit</button>

    </form>

@endsection
```