Coursework 1a SPost Web Application

Samuel Netherway (955794)

Table of Contents

1 Migrations

1.1 2014_10_12_000000_create_users_table.php

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateUsersTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('title');
            $table->string('firstname');
            $table->string('lastname');
            $table->string('profile_image');
            $table->string('cover_image');
            $table->string('username');
            $table->string('phone_number');
            $table->string('email')->unique();
            $table->timestamp('email_verified_at')->nullable();
            $table->string('address');
            $table->string('password');
            $table->rememberToken();
            $table->timestamps();
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('users');
    }
}
```

1.2 2020_11_02_213432_create_posts_table.php

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreatePostsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('posts', function (Blueprint $table) {
            $table->id();
            $table->mediumText('title');
            $table->longText('content');
            $table->string('image');
            $table->timestamps();

            // Foreign Keys:
            $table->bigInteger('user_id')->unsigned();

            // References:
            $table->foreign('user_id')->references('id')->on('users')
                ->onDelete('cascade')->onUpdate('cascade');
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('posts');
    }
}
```

1.3 2020_11_02_213446_create_comments_table.php

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateCommentsTable extends Migration
{
    /**
     * Run the migrations.
     *
     * @return void
     */
    public function up()
    {
        Schema::create('comments', function (Blueprint $table) {
            $table->id();
            $table->longText('content');
            $table->string('image');
            $table->timestamps();

            // Foreign Keys:
            $table->bigInteger('user_id')->unsigned();
            $table->bigInteger('post_id')->unsigned();

            // References:
            $table->foreign('user_id')->references('id')->on('users')
                ->onDelete('cascade')->onUpdate('cascade');
            $table->foreign('post_id')->references('id')->on('posts')
                ->onDelete('cascade')->onUpdate('cascade');
        });
    }

    /**
     * Reverse the migrations.
     *
     * @return void
     */
    public function down()
    {
        Schema::dropIfExists('comments');
    }
}
```

2 Models

2.1 User.php

```php
<?php

namespace App;

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

class User extends Authenticatable
{
    use Notifiable;

    /**
     * The attributes that are mass assignable.
     *
     * @var array
     */
    protected $fillable = [
        'name', 'email', 'password',
    ];

    /**
     * The attributes that should be hidden for arrays.
     *
     * @var array
     */
    protected $hidden = [
        'password', 'remember_token',
    ];

    /**
     * The attributes that should be cast to native types.
     *
     * @var array
     */
    protected $casts = [
        'email_verified_at' => 'datetime',
    ];

    // Returns all of the user's posts
    public function posts()
    {
        return $this->hasMany('App\Post');
    }

    // Returns all of a users comments on all posts
    public function comments()
    {
        return $this->hasMany('App\Comment');
    }
}
```

2.2 Post.php

```php
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Post extends Model
{
    // Returns the user who submitted the post
    public function user()
    {
        return $this->belongsTo('App\User');
    }

    // Returns the comments on a post
    public function comments()
    {
        return $this->hasMany('App\Comment');
    }
}
```

2.3 Comment.php

```php
<?php

namespace App;

use Illuminate\Database\Eloquent\Model;

class Comment extends Model
{
    // Returns the user who posted the comment
    public function user()
    {
        return $this->belongsTo('App\User');
    }

    // Returns the post which was commented on
    public function post()
    {
        return $this->belongsTo('App\Post');
    }
}
```

3 Seeders

3.1 DatabaseSeeder.php

```php
<?php

use Illuminate\Database\Seeder;

class DatabaseSeeder extends Seeder
{
    /**
     * Seed the application's database.
     *
     * @return void
     */
    public function run()
    {
        $this->call(UserTableSeeder::class);
        $this->call(PostTableSeeder::class);
        $this->call(CommentTableSeeder::class);
    }
}
```

3.2   UserTableSeeder.php

```php
<?php

use Illuminate\Database\Seeder;
use App\User;

class UserTableSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        $user1 = new User;
        $user1->title = "Mr";
        $user1->firstname = "Bob";
        $user1->lastname = "Ross";
        $user1->username = "HappyAccidents";
        $user1->profile_image = "https://lorempixel.com/640/480/?58841";
        $user1->cover_image = "https://lorempixel.com/640/480/?77616";
        $user1->phone_number = "07704649888";
        $user1->email = "bobross@gmail.com";
        $user1->address = "1 Apple Park Way, Cupertino, Californa, United States";
        $user1->password = "Painting123";
        $user1->save();

        $user2 = new User;
        $user2->title = "Dr";
        $user2->firstname = "Andrew";
        $user2->lastname = "Clarke";
        $user2->username = "AndyC";
        $user2->profile_image = "https://lorempixel.com/640/480/?32496";
        $user2->cover_image = "https://lorempixel.com/640/480/?48448";
        $user2->phone_number = "07744449223";
        $user2->email = "andyc@outlook.com";
        $user2->address = "74-78 Avon St, Bristol, United Kingdom";
        $user2->password = "Back&Forth112";
        $user2->save();

        factory(App\User::class, 20)->create();
    }
}
```

3.3 PostTableSeeder.php

```php
<?php

use Illuminate\Database\Seeder;
use App\Post;

class PostTableSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        $post1 = new Post;
        $post1->title = "Hello Everyone!";
        $post1->content = "Hi Everyone. My name is Bob Ross and I'm new to SPost.
            Who likes my paintings?";
        $post1->image = "https://lorempixel.com/640/480/?31610";
        $post1->user_id = 1;
        $post1->save();

        factory(App\Post::class, 50)->create();
    }
}
```

3.4 CommentTableSeeder.php

```php
<?php

use Illuminate\Database\Seeder;
use App\Comment;

class CommentTableSeeder extends Seeder
{
    /**
     * Run the database seeds.
     *
     * @return void
     */
    public function run()
    {
        $comment1 = new Comment;
        $comment1->content = "Yes, there great!";
        $comment1->image = "https://lorempixel.com/640/480/?20983";
        $comment1->user_id = 2;
        $comment1->post_id = 1;
        $comment1->save();

        factory(App\Comment::class, 100)->create();
    }
}
```

4 Factories

4.1 UserFactory.php

```php
<?php

/** @var \Illuminate\Database\Eloquent\Factory $factory */

use App\User;
use Faker\Generator as Faker;
use Illuminate\Support\Str;

$factory->define(User::class, function (Faker $faker) {
    return [
        'title' => $faker->title,
        'firstname' => $faker->firstName,
        'lastname' => $faker->lastName,
        'username' => $faker->userName,
        'profile_image'=> $faker->imageURL,
        'cover_image'=> $faker->imageURL,
        'phone_number' => $faker->phoneNumber,
        'email' => $faker->unique()->safeEmail,
        'email_verified_at' => now(),
        'address' => $faker->address,
        'password' => $faker->password(),
        'remember_token' => Str::random(10),
    ];
});
```

4.2 PostFactory.php

```php
<?php

/** @var \Illuminate\Database\Eloquent\Factory $factory */

use App\Post;
use Faker\Generator as Faker;

$factory->define(Post::class, function (Faker $faker) {
    return [
        'title' => $faker->realText(50),
        'content' => $faker->realText(300),
        'image' => $faker->imageURL(),
        'user_id'=>App\User::inRandomOrder()->first()->id,
    ];
});
```

4.3   CommentFactory.php

```php
<?php

/** @var \Illuminate\Database\Eloquent\Factory $factory */

use App\Comment;
use Faker\Generator as Faker;

$factory->define(Comment::class, function (Faker $faker) {
    return [
        'content' => $faker->realText(200),
        'image' => $faker->imageURL(),
        'user_id'=>App\User::inRandomOrder()->first()->id,
        'post_id'=>App\Post::inRandomOrder()->first()->id,
    ];
});
```