



PRESTIGE
INSTITUTE OF MANAGEMENT & RESEARCH, GWALIOR
MBA PROGRAMME ACCREDITED BY NBA
UGC NAAC ACCREDITED 'A' GRADE AUTONOMOUS INSTITUTE

MAJOR PROJECT REPORT

ON

RELAX- Instagram clone

Towards Partial Fulfillment of Requirement

Of

Bachelor of Computer Application

Batch 2020-23

SUBMITTED TO

Information Technology Department,
PIMRG

SUBMITTED BY

❁ Divya
❁ Pawan Gupta
❁ Ravi Kishan
❁ Shivani Yadav

BCA III SEM

DECLARATION

We, **Divya, Pawan Gupta, Ravi Kishan and Shivani Yadav** students of BCA III Sem. of Prestige Institute of Management & Research, Gwalior, hereby declare that the Major Project Report entitled “**RELAX- Instagram Clone**” is submitted by us in the line of partial fulfilment of course objectives for the Bachelor of Computer Application.

We assure that this project report is the result of our own efforts and that any other institute for the award of any degree or diploma has not submitted it.

Date: 13-01-2023

Place: Gwalior

Name :

Divya

Pawan Gupta

Ravi Kishan

Shivani Yadav

CERTIFICATE

This is to certify that, **Divya, Pawan Gupta , Ravi Kishan and Shivani Yadav** of BCA III Semester of Prestige Institute of Management & Research Gwalior, have successfully completed their Minor Project. They have prepared this report entitled “**RELAX- Instagram Clone**” under my direct supervision and guidance.

Date: 13-01-2023

Asst.Prof. / Dr. Bharti Gole

Place: Gwalior

(Faculty Guide)

TABLE OF CONTENTS

S.NO	UNIT NAME	PAGE NO
1	INTRODUCTION	1-2
2	PROBLEM STATEMENT	3-5
3	PROJECT DESCRIPTION	6-8
4	PROBLEM ON PRACTICE	9-46
5	CONCLUSION & REFERENCE	46-48

UNIT – 1: INTRODUCTION

We, Divya, Pawan Gupta, Ravi Kishan, and Shivani Yadav are the students of BCA III sem were assigned the task to make a social media app where users can encourage interaction and engagement between users.

Thus, we made RELAX- Instagram Clone which aims to replicate these features and to create a visually appealing and user-friendly application that is easy to use and navigate.

The project will require a solid understanding of web development technologies such as HTML, CSS, JavaScript, and PHP, as well as a proficiency in working with databases, such as MySQL. It will also require a deep understanding of web development best practices and a focus on security and scalability. Additionally, this project will also explore new technologies and trends in web development, such as responsive design, cloud computing and machine learning.

We have used some tools like Visual Studio Code, Xampp, GitHub, etc. We worked using a PHP framework named LARAVEL. It is a MVC (Model view controller) framework for creating new generation and fast applications. We use Tailwind CSS, SASS and Bootstrap for designing purposes, jQuery and JavaScript for action related purposes.

PHP for backend logic and operations including login, register, user creation, updation, deletion and viewing, post creation, updation, deletion and viewing, writing logic about file uploading and managing storage part.

We use Apache as a server provider which makes website work. In the project, we are thinking about to use MySql database for performing necessary CRUD operations which is hosted on local server.

This project is hosted on GitHub for CI/CD operations. For the creation of the project, we have divided the project into separate parts:

1. UI Designs
2. Database management or schema designing
3. Backend development & logical building

4. Report and presentation work

We have divided these works in our teammates, each have to prepare his/her own work and compile all the work in a single unit. After the compilation of the project, every teammate has to teach others about their contribution in the project.

We have faced some challenges while working on this project and resolved some by participating in the discussions with our teammates. Everyone worked hard for this project by giving their hundred percent.

In this project, there can be many future enhancements that can be made in this project such as scalability and performance optimization, security enhancements, advance UI design and machine learning integration for post and user recommendations.

Thankyou Prestige Institute of Management & Research, Gwalior for giving this opportunity to work as a team and showcase our learning. It taught us a lot about web development technologies and how to work as a team.

UNIT - 2: PROBLEM STATEMENT

Q. What is the purpose and requirements of it?

The primary purpose of Relax is to allow users to share photos and videos with their followers, as well as to discover and interact with content from other users. It can be used for a wide range of purposes, such as sharing personal photos with friends and family, promoting a business or product, or simply keeping up with the latest trends and developments in a particular field or community.

To use Relax, you will need a smartphone or tablet , access to the internet through a computer. You will also need to create an account by providing some basic information and setting up a username and password. Once you have an account, you can start sharing your own content and engaging with others on the platform.

Q. What problems will this help in resolving?

Relax can help people in a number of ways, depending on how they use it. Some possible benefits include:

- Staying connected with friends and family: Relax allows users to share their photos and videos with a large audience, which can help people stay connected with their loved ones, even if they are physically distant.
- Promoting a business or product: Relax can be a powerful marketing tool for businesses, as it allows them to showcase their products and services to a large audience in a visually appealing way.

- **Discovering new interests and communities:** Relax has a wide range of content, covering a variety of topics and interests. By following accounts that align with their interests, users can discover new hobbies, ideas, and communities on the platform.
- **Self-expression and creativity:** Relax can be a place for users to share their personal interests, passions, and creativity with others. It can be a platform for self-expression and a way to connect with like-minded individuals.
- **Keeping up with current events and trends:** Relax can also be a source of information and entertainment, as users can follow accounts that provide updates on current events, trends, and popular culture.

Advantages of Instagram over other social media sites?

1. **High engagement rates:** Instagram's visual nature and emphasis on creative expression can lead to higher engagement rates than other social media platforms.
2. **Strong visual branding:** Instagram is a visual-first platform, allowing businesses and individuals to showcase their products and services through visually appealing images and videos.
3. **Targeted advertising:** Instagram offers a range of advertising options that allow businesses to target specific demographics, interests, and behaviors.
4. **Audience insights:** Instagram's business accounts have access to audience insights, which provide information on followers such as location, gender, and age.
5. **E-commerce integration:** Instagram allows direct integration of e-commerce features, it allows users to make purchase directly in the app, making it easy for businesses to drive sales.
6. **Strong community-driven focus:** Instagram has a lot of features to encourage engagement and interaction, such as IGTV, IG Stories and IG live, which allows users to create and participate in shared experiences with a wider community of people.

Q. Why people find Instagram helpful?

People use Instagram for a variety of reasons, including:

1. **Social connection:** Instagram allows people to connect with friends and family, as well as meet new people who share similar interests.
2. **Self-expression:** Instagram is a platform for people to share their experiences, thoughts, and interests through photos and videos. It allows people to express their creativity, and to showcase their talents and passions.
3. **Personal branding:** Instagram has become a powerful tool for individuals and businesses to build their personal and professional brand, by creating a visual identity that showcases their personality, values and skills.
4. **Entertainment:** Instagram is a source of inspiration, entertainment, and education, people use it to discover new content, like funny memes, educational infographics, inspiring stories, etc.
5. **Staying informed and up-to-date:** Instagram also serves as a way to stay informed about the latest news, trends and events, by following influencers, news outlets, or public figures
6. **Professional and Business reasons:** Many businesses and professionals use Instagram as a way to reach a wider audience, build relationships with customers, and grow their sales.

UNIT – 3 : PROJECT DESCRIPTION

Q. What is Relax?

Relax is a social media platform that allows users to share photos and videos, as well as to interact with others through likes, comments, and direct messaging. It can be accessed on the web.

Q. What is Instagram and how it makes people engaged in its activity?

Instagram is a popular social media platform that allows users to share photos and videos, as well as connect with friends, family, and other users. The app is built around the idea of visual storytelling, and it uses creative filters and editing tools to enhance users' photos and videos.

One of the key reasons why Instagram is so engaging is that it is a very visual platform. People are naturally drawn to visual content, and Instagram allows users to easily create, edit and share beautiful and interesting photos and videos. This, combined with Instagram's use of creative filters, allows users to make their content stand out from the crowd and be more engaging.

Instagram also emphasizes engagement through its use of likes, comments, and hashtags. Users can like and comment on each other's photos and videos, as well as search for content using hashtags. This creates a sense of community and encourages users to interact with each other and with the content they see on the platform.

Another reason why Instagram is so engaging is the emphasis on discovery. Instagram uses an algorithm-based feed that shows users content they are likely to be interested in. This makes it easy for users to discover new and interesting content, which can be very engaging. Additionally, Instagram has features like IGTV, IG stories and IG live which allows users to create and participate in shared experiences with a wider community of people.

Instagram is also a great platform for businesses and professionals, it offers targeted advertising that allow businesses to reach specific demographics, interests and behaviours, and provides

insights about audience's characteristics. Instagram also allows direct integration of e-commerce features, making it easy for businesses to drive sales.

In conclusion, Instagram is a highly engaging social media platform that allows users to share visual stories, connect with others, and discover new and interesting content. Its use of creative filters, emphasis on engagement, and algorithm-based feed make it a powerful tool for personal and professional expression, community building and business growth.

Following acts can be performed on Relax

There are many different actions that users can perform on Instagram. Some of the most common ones include:

- Sharing photos and videos: Users can share photos and videos or upload them from their device's camera roll. They can then add captions, and other enhancements to their content before sharing it with their followers.
- Interacting with others: Users can like, comment on, and share other users' content, as well as send private messages to individual users.
- Discovering new content: Users can explore the app to discover new accounts to follow, and explore the "Explore" tab to find content that aligns with their interests.
- Customizing their profile: Users can update their profile information, such as their username, profile picture, and bio, as well.

Editing their preferences: Users can customize their settings like which accounts they want to follow or block.

Q. What are the main features of RELAX which needs to be focused on?

The main features of RELAX are:

1. Authenticated and email verified login system:

2. Real-time conversation
3. Create, update, delete and read operations on different type of data
4. Advance UI design
5. Mobile Responsive design
6. Advance Animation and transition
7. File Uploading and manipulation
8. Fast and futuristic model

Q. What makes Relax different from other social media apps?

Ans: Relax is a photo and video sharing website that is built around the idea of visual storytelling. It also has a focus on engagement through the use of likes, comments, and hashtags. It also has a more passive browsing experience, with a feed that is based on an algorithm that shows users content they are likely to be interested in. Additionally, it has a more community-driven focus, with features like RX stories to encourage interaction and engagement between users.

UNIT – 4 : PROJECT ON PRACTICE

In the way of describing the structure and functions of this project, I would like to use an approach of jinja's explainer technique. In this technique, I break things into sequences and explain one after another. We break the concept into some chapters and explain it.

These are the chapters:

1. Tools
 - a) Language Used
 - b) Framework and Libraries used
 - c) Architecture of Project
 - d) Tools used
2. Project Directory
3. Modules
 - a) Auth Module & User Module
 - b) Post Module
 - c) Profile Module
 - d) Follows Module
 - e) Like Module
 - f) Comment Module
 - g) Story Module
 - h) Chat Module
4. ER-Diagram
5. Database relational table

Every Module contains some elements:

- i. Abstract
- ii. Model
- iii. Migration
- iv. Controller
- v. Views

CHAPTER 1: TOOLS

a) Language Used

In this project, we use a lot of programming language and scripting languages. These are the names of used programming language and scripting languages:

- I. **HTML** – HTML is an acronym that stands for Hyper Text Markup Language which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and a Web page.

Hyper Text: HyperText simply means "Text within Text." A text that has a link within it, is a hypertext. Whenever you click on a link that brings you to a new webpage, you have clicked on a hypertext. HyperText is a way to link two or more web pages (HTML documents) with each other.



Markup language: A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes the text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page: A web page is a document that is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of a static or dynamic type. With the help of HTML only, we can create static web pages.

Hence, HTML is a markup language that is used for creating attractive web pages with the help of styling, and which looks in a nice format on a

web browser. An HTML document is made of many HTML tags and each HTML tag contains different content.

II. **CSS** – CSS stands for Cascading Style Sheets. It is a style sheet language



that is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML document including plain

XML, SVG, and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

III. **JAVASCRIPT** – JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side scripts to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities.

JavaScript was first known as **LiveScript**, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.



- IV. **SCSS** - Sass (Syntactically Awesome Style Sheets) is an extension of CSS. Style sheet languages control where and how text appears on a webpage, from frame size and color to menu positions.

CSS is used all over the web, but that doesn't make it the smoothest coding experience. It was designed to help developers write instructions on how to present text on a screen rather than to work with variables or perform complex decision-making tasks.

As you'll find out, Sass addresses some of these serious shortcomings of CSS, saving developers time and effort on projects.



- V. **PHP** – (PHP: Hypertext Preprocessor) is An extremely popular scripting language that is used to create dynamic Web pages.



Combining syntax from the C, Java, and Perl languages, PHP code is embedded within HTML pages for server-side execution. It is commonly used to extract data out of a database on the Web server and present it on the Web page.

Originally known as "Personal Home Page," PHP is supported by all Web servers and widely used with the MySQL database. See MySQL.

- VI. **SQL** – Structured Query Language (SQL) is a standardized programming language that is used to manage relational databases and perform various operations on the data in them. Initially created in the 1970s, SQL is regularly used not only by database administrators but also by developers writing data integration scripts and data analysts looking to set up and run analytical queries.



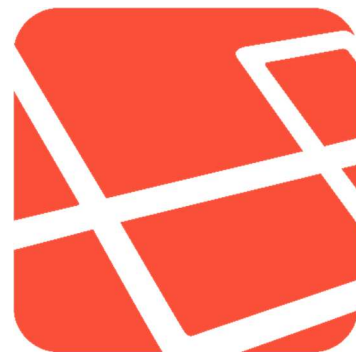
The term SQL is pronounced ess-kew-ell or sequel.

b) Framework and Libraries used

In this project, we use a PHP framework and multiple design and utility frameworks. These are the names of libraries and frameworks:

- I. **LARAVEL** – Laravel is an open-source PHP framework, which is robust and easy to understand. It follows a model-view-controller design pattern. Laravel reuses the existing components of different frameworks which helps in creating a web application. The web application thus designed is more structured and pragmatic.

Laravel offers a rich set of functionalities that incorporates the basic features of PHP frameworks like CodeIgniter, Yii, and other programming languages like Ruby on Rails. Laravel has a very rich set of features that will boost the speed of web development.



If you are familiar with Core PHP and Advanced PHP, Laravel will make your task easier. It saves a lot of time if you are planning to develop a website from scratch. Moreover, a website built in Laravel is secure and prevents several web attacks.

- II. **MYSQL** – A database is a structured collection of data. It may be anything from a simple shopping list to a picture gallery or a place to hold vast amounts of information in a corporate network. In particular, a relational database is a digital store collecting data and organizing it according to the relational model. In this model, tables consist of rows and columns, and relationships between data elements all follow a strict logical structure. An RDBMS is simply a set of software tools used to implement, manage, and query such a database.

MySQL is integral to many of the most popular software stacks for building and maintaining everything from customer-facing web applications to powerful, data-driven B2B services. Its open-source nature, stability, and rich feature set, paired with ongoing development and support from Oracle, have meant that internet-critical organizations such as Facebook, Flickr, Twitter, Wikipedia, and YouTube all employ MySQL backends.



- III. **BOOTSTRAP** - Twitter developed Bootstrap in 2011 and got released it on GitHub in the same year. Bootstrap is a free open source front-end-framework to develop faster, easier, responsive web pages, and mobile-first websites. Bootstrap is the most popular framework for its compatibility with all modern browsers like Firefox, Chrome, Opera, Safari, Edge, etc.



Bootstrap has pre-defined classes that make a developer's job easy. However, a basic understanding of HTML & CSS is required to work with Bootstrap. You can make responsive designs for different devices with Bootstrap. It contains various HTML & CSS-based design templates, especially for typography, buttons, tables, forms, models, navigation, and image carousels as well as optional JavaScript plugins.

IV. **TAILWINDCSS** – Tailwind CSS can be used to style websites in the fastest and easiest way.

Tailwind CSS is a utility-first CSS framework for rapidly building custom user interfaces. It is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to fight to override.



The beauty of this thing called tailwind is it doesn't impose design specifications or how your site should look, you simply bring tiny components together to construct a unique user interface. What Tailwind simply does is take a 'raw' CSS file, process this CSS file over a configuration file, and produces an output.

- v. **JQUERY** – jQuery is a small, lightweight, and fast JavaScript library. It is cross-platform and supports different types of browsers. It is also referred to as ? write less do more. because it takes a lot of common tasks that require many lines of JavaScript code to accomplish and binds them into methods that can be called with a single line of code whenever needed. It is also very useful to simplify a lot of the complicated things from JavaScript, like AJAX calls and DOM manipulation.



c) **Architecture of Project**

This project is created on the **Laravel framework**. For that, This project follows the architecture of **MVC (Model View Controller)**.

Q. What is MVC?

Ans: The **Model-View-Controller (MVC)** framework is an architectural/design pattern that separates an application into three main logical components Model, View, and Controller. Each architectural component is built to handle specific development aspects of an application. It isolates the business logic and presentation layer from each other. It was traditionally used for desktop graphical user interfaces (GUIs). Nowadays, MVC is one of the most frequently used industry-standard web development frameworks to create scalable and extensible projects. It is also used for designing mobile apps.

MVC was created by Trygve Reenskaug. The main goal of this design pattern was to solve the problem of users controlling a large and complex data set by splitting a large application into specific sections that all have their purpose.

Features of MVC :

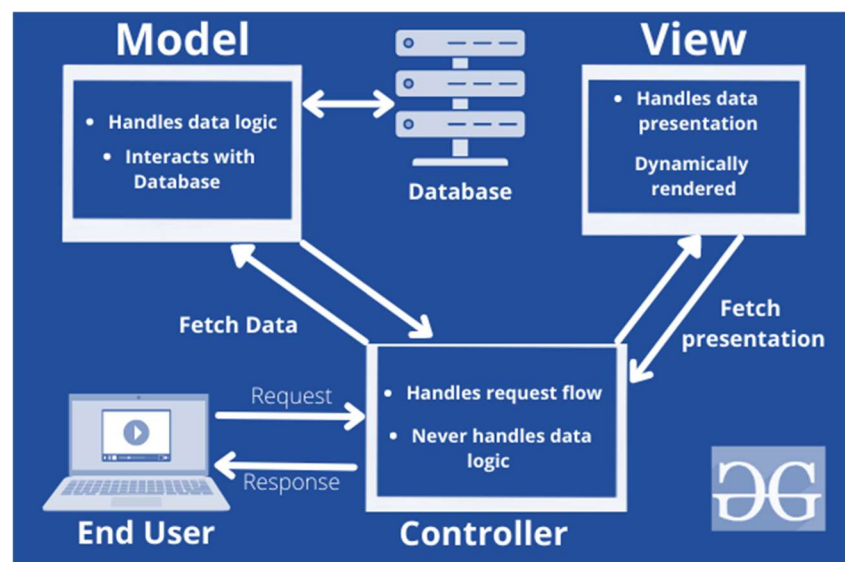
- i. It provides a clear separation of business logic, UI logic, and input logic.

- ii. It offers full control over your HTML and URLs which makes it easy to design web application architecture.
- iii. It is a powerful URL-mapping component using which we can build applications that have comprehensible and searchable URLs.
- iv. It supports Test Driven Development (TDD).

Components of MVC :

The MVC framework includes the following 3 components:

- i. Controller
- ii. Model
- iii. View



A. Controller:

The controller is the component that enables the interconnection between the views and the model so it acts as an intermediary. The controller doesn't have to worry about handling data logic, it just tells the model what to do. It processes all the business logic and incoming requests, manipulates data using the Model component, and interacts with the View to render the final output.

B. View:

The View component is used for all the UI logic of the application. It generates a user interface for the user. Views are created by the data which is collected by the model

component but these data aren't taken directly but through the controller. It only interacts with the controller.

C. Model:

The Model component corresponds to all the data-related logic that the user works with. This can represent either the data that is being transferred between the View and Controller components or any other business logic-related data. It can add or retrieve data from the database. It responds to the controller's request because the controller can't interact with the database by itself. The model interacts with the database and gives the required data back to the controller.

Advantages of MVC:

- i. Codes are easy to maintain and they can be extended easily.
- ii. The MVC model component can be tested separately.
- iii. The components of MVC can be developed simultaneously.
- iv. It reduces complexity by dividing an application into three units. Model, view, and controller.
- v. It supports Test Driven Development (TDD).
- vi. It works well for Web apps that are supported by large teams of web designers and developers.
- vii. This architecture helps to test components independently as all classes and objects are independent of each other
- viii. Search Engine Optimization (SEO) Friendly.

Disadvantages of MVC:

- i. It is difficult to read, change, test, and reuse this model
- ii. It is not suitable for building small applications.
- iii. The inefficiency of data access in view.

- iv. The framework navigation can be complex as it introduces new layers of abstraction which requires users to adapt to the decomposition criteria of MVC.
- v. Increased complexity and Inefficiency of data

NOTE: MVC is generally used on applications that run on a single graphical workstation. The division of logical components enables readability and modularity as well it makes more comfortable for the testing part.

d) Tools Used

For the development of this project, we use a set of tools. There are described:

A. VISUAL STUDIO CODE – Visual Studio Code (famously known as VS Code) is a free open-source text editor by Microsoft. VS Code is available for Windows, Linux, and macOS. Although the editor is relatively lightweight, it includes some powerful features that have made VS Code one of the most popular development environment tools in recent times.

VS Code supports a wide array of programming languages from Java, C++, and Python to CSS, Go, and Dockerfile. Moreover, VS Code allows you to add on and even create new extensions including code linkers, debuggers, and cloud and web development support.



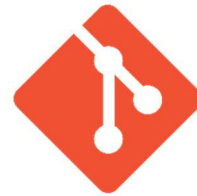
B. XAMPP – XAMPP is an abbreviation where X stands for Cross-Platform, A stands for Apache, M stands for MYSQL, and Ps stand for PHP and Perl, respectively. It is an open-source package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB, PHP, and Perl.



XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself. Among these technologies, Perl is a programming language used for web development, PHP is a backend scripting language, and MariaDB is the most vividly used database developed by MySQL.

C. GIT AND GITHUB –

Git is an open-source distributed version control system. It is designed to handle minor to major projects with high speed and efficiency. It is developed to coordinate the work among the developers. The version control allows us to track and work together with our team members in the same workspace.



GitHub is a for-profit company that offers a cloud-based Git repository hosting service. Essentially, it makes it a lot easier for individuals and teams to use Git for version control and collaboration. GitHub's interface is user-friendly enough so even novice coders can take advantage of Git. Without GitHub, using Git generally requires a bit more technically savvy and the use of the command line.

CHAPTER 2: PROJECT DIRECTORY

Here, we are going to explain every directory of the project and their consisted files with brief explanations:

If you're new to Laravel or frameworks in general then you might be slightly overwhelmed the first time you run `laravel new new-project` in your development environment and see the file and folder structure it has created. Don't worry though, because to begin with you'll only need to focus on a few key files and folders, and we'll go through what they're for in this article.

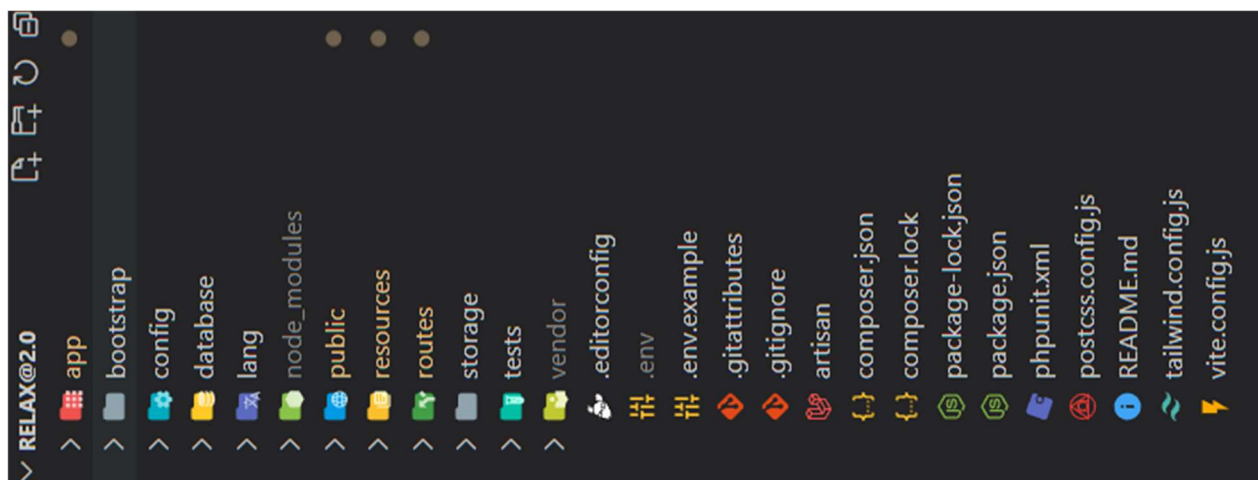
Overview:

Laravel project architecture follows a model-view-controller (MVC) pattern, which essentially means the logic part of the application is separated from the user interface. This is done by utilising a “model”—the application data, which is displayed in a “view”—the user interface, which is updated by the “controller”—which makes calls on the model and the view to display the appropriate information.

In addition to this, Laravel has a lightweight templating engine, a powerful set of routing features, support for additional libraries and packages, the **Eloquent ORM** database query language, **artisan command line** shortcuts and testing tools.

So let's take a look where we can find the files that we need to modify to make use of these features, we'll use our Welcome project from our previous article to go through the key files and folders.

Root:



We'll dive into the folders in a moment, but the key file at this level that you should be aware of from the start is the `.env` file, which will contain your database connection details, as well as other login details or API keys for third party services or email accounts. You also set your app URL, set the app debug level and specify the environment for the app, e.g. local, staging or production.

app/:

By default you should see a `User.php` file in this folder. This defines the default User model. Laravel deliberately does not create a Models folder due to the range of definitions of a “Model” in this context, leaving it up to the developer. See at the end of the article for a quick guide on how to set up a Models folder in your project if you'd prefer.

The model is usually a class with properties that match the columns in your database, and the relationship between models can also be defined within these files.

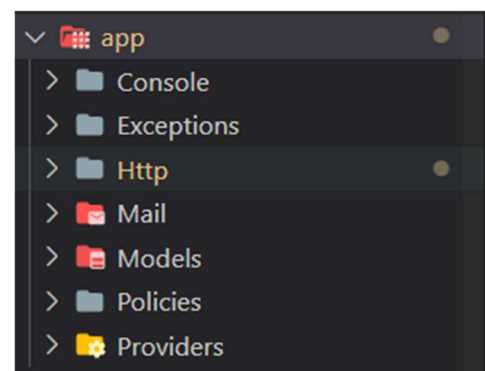
To create a model in Laravel, run this command in your terminal:

```
$ php artisan make:model ModelName
```

And an `app/ModelName.php` file will be generated.

You can create a database migration at the same time by adding `-m` to the end of the command (see database/ below).

We'll take a look at the folders within the app directory next. Typically, additional folders for Notifications, Mail, Transformers for API data and Event Listeners, to name a few will go in the `app/` folder, but we're going to skip everything in here for now except the `app/Http/Controllers` folder.



Controllers/:

This, unsurprisingly is where the Controller files live.

You can create a controller by running:

```
$ php artisan make:controller ControllerName
```

You can add --resource after this command to create empty CRUD (create, read, update, delete) functions in the file.

You can create functions in the controller files that contain any logic you choose, but in we apps they're commonly used to manipulate data and return a view.

To use a simpler version of the example from our Wink post, this function will simply return this view:
resources/views/home/home.blade.php:

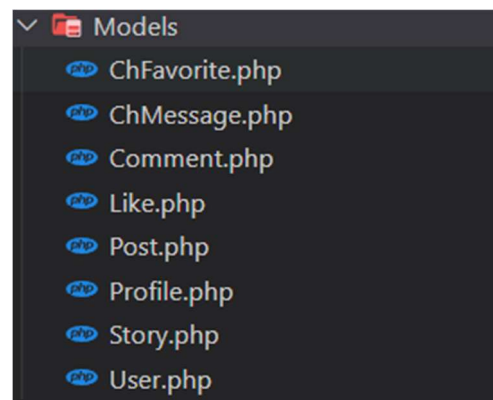
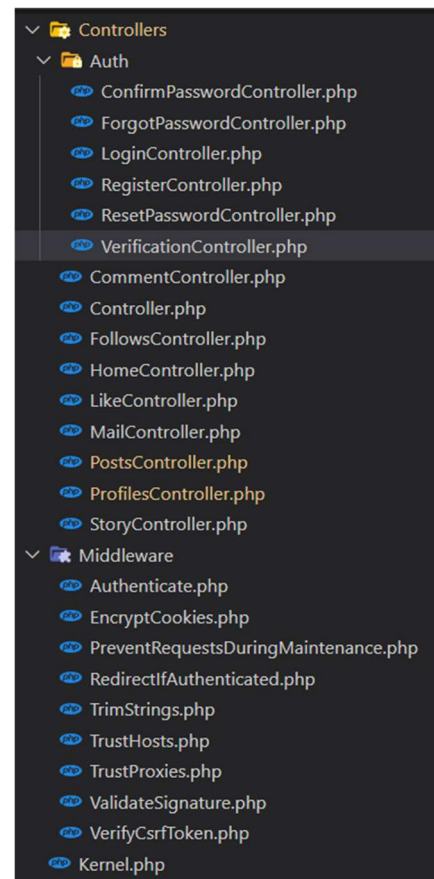
```
public function index()
{
    return view('home.home');
}
```

You can then register a route (see the routes/web.php below) to access that function in your application.

For the time being, that's all you need to know about the app/ folder, so let's move on.

model/:

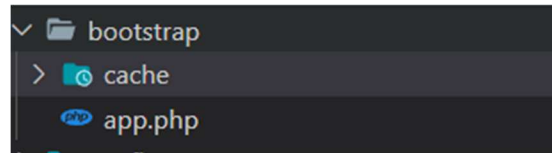
This model/ folder contain all the object of database which is use as a class in controller and every where. It contain every database entity and connected to migrations to perform all type of task with ORM.



```
$ php artisan make:model ModelName
```

bootstrap/:

The bootstrap/ folder isn't for Bootstrap CSS (although if you want to learn about Bootstrap CSS check out our getting started guide) but the app.php file that bootstraps (loads) the framework files, and also stores cached files for performance improvements. We don't need to do anything in here for now.



config/:

When getting started with Laravel, there isn't much to change within the config/ folder, but it's good to know where the config files are if you need to change anything. If you've installed a third party package and you need to publish the config file you can usually find the command in the readme but it will be something like:

```
$ php artisan vendor:publish --provider='Provider\Name' --tag="config"
```

This will copy the config file into the config/ folder for you to edit.

database/:

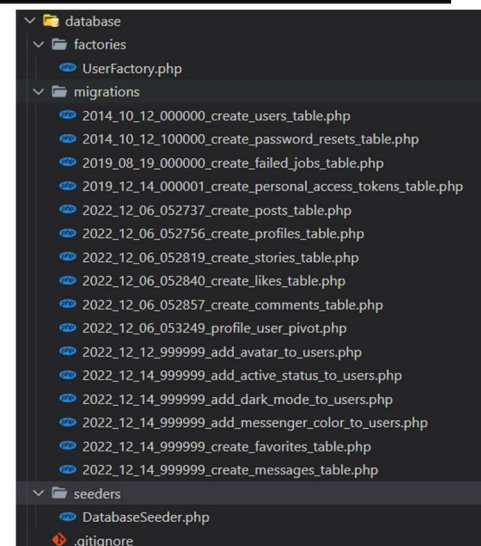
In the database/ folder you'll find a migrations/ folder, this is where files are created when you need a migration file for modifying the database structure of your project. Run the following to create a migration file:

```
$ php artisan make:migration --table='table' --description_of_migration'
```

This will then be created in your database/migrations/ folder. Edit the file to make the necessary changes (check out the 2014_10_12_000000_create_users_table.php file in database/migrations/ for a handy example), then when you run:

```
$ php artisan migrate
```

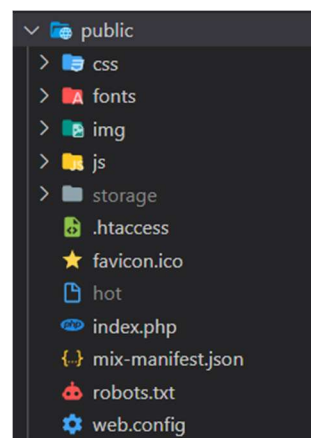
This folder will be checked for pending migrations files that have not yet been run and run them, and update your database accordingly.



public/:

These are the files and folder that will sit in the public folder on your web server when a project is deployed.

You can add and reference your own images, css, javascript etc within this folder. When deploying a site, check the paths in index.php are correct that point to your /vendor/autoload.php and /bootstrap/app.php files.

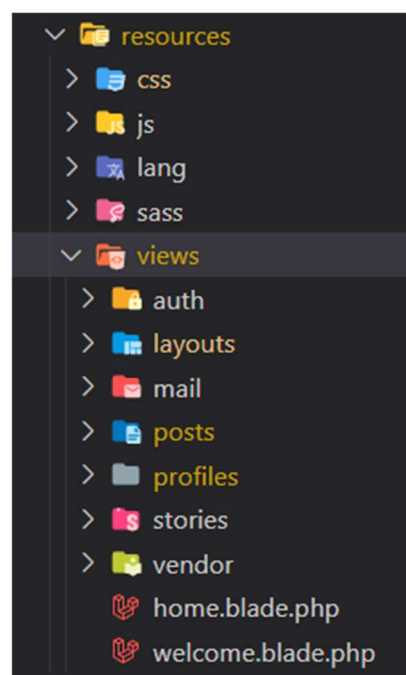


resources/:

In brief, the js/ and sass/ folders contain the uncompiled less, sass and javascript assets, and lang/ contains language files. We'll skip over these for now and focus on just one folder, views/.

views/:

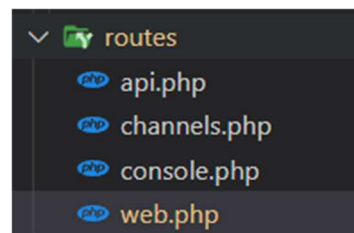
You should see a welcome.php file in this folder, this is the page you will see when you first navigate to "your-project.test". You can create pages and template elements in their own folders within the views/ folder and either return these views directly from the routes/web.php file or from a controller file.



Laravel uses the blade templating engine, which is a simple yet powerful system that allows you to define sections in your views and easily inherit layouts from other files.

routes/:

For a simple web app, we'll just look at the routes/web.php file, this is where you define the URLs for web requests for your app and route particular URLs to controllers in your application, or directly return a view.



You'll see the default entry:

```
Route::get('/', function () {  
    return view('welcome');  
});
```

This returns the welcome view when you visit the root URL of your app.

If you have, for example a blog controller `app/Http/Controllers/Post.php`, you can access the CRUD (Create, Read, Update, Delete) functions by simply adding a resource route to your `routes/web.php`:

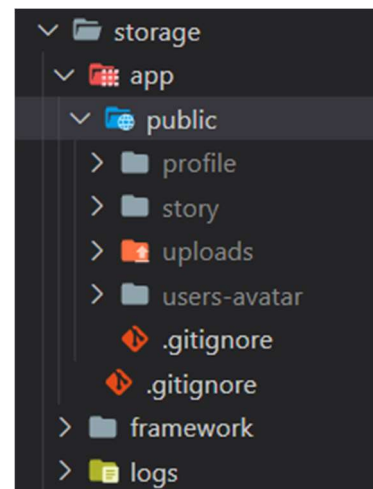
```
Route::resource('user', UserController::class);
```

You can specify particular functions in a controller (in this case the `index()` function mentioned in the `Controllers/` section above) by using:

```
Route::get('/', 'App\Http\Controllers\PostsController@index')-  
>name('post.index');
```

storage/:

The main folder that might be useful from the start is the `storage/logs` folder. This contains any application logs if an error occurs and can be useful for debugging and troubleshooting. Since the overhaul to the logging system in Laravel 5.6, the default behaviour is that date stamped log files are created, making finding errors simpler than searching through single large `laravel.log` file.



tests/:

The `tests/` folder is where you write and store all your automated test files. PHPUnit testing is supported and configured by default. You'll see `Feature` and `Unit` folders within the `tests` folder.

To create a test in the `Feature` directory, run:

```
$ php artisan make:test UserTest
```

To create a test in the Unit directory, run:

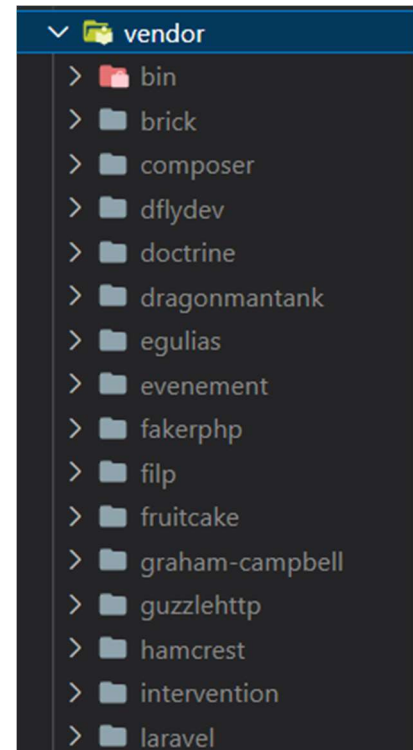
```
$ php artisan make:test UserTest --unit
```

vendor/:

The final folder is the vendor/ folder—this contains all the dependencies installed by composer, as well as optional packages you may have installed. It's important you don't directly edit any files in here as future updates may cause errors in your app. You can publish assets from the vendor folder with a php artisan command, this is usually explained on the package's git readme. This enables you to make changes without affecting the base vendor code.

Wrap up:

So that's a brief look at all the files and folders you should be aware of when starting a new Laravel project. The type of project you are coding will influence how much time you spend in the various files and folders. Typically whilst getting started with a web app you'll become quickly acquainted with the app/Http/Controllers/ and resources/views/ folders.



As this is a quick primer we haven't covered all the files and functions, such as Middleware or Exception handling, but hopefully this will help you not only to understand what the key files and folders do but also appreciate the thought that has gone into the structure of a Laravel project to keep everything organised.

CHAPTER 3: MODULE

a) Auth Module & User Module:

- i. **Abstract** – In this Project, we add an advance **authentication system** with register, login, email verification, forget password, local storage session etc.

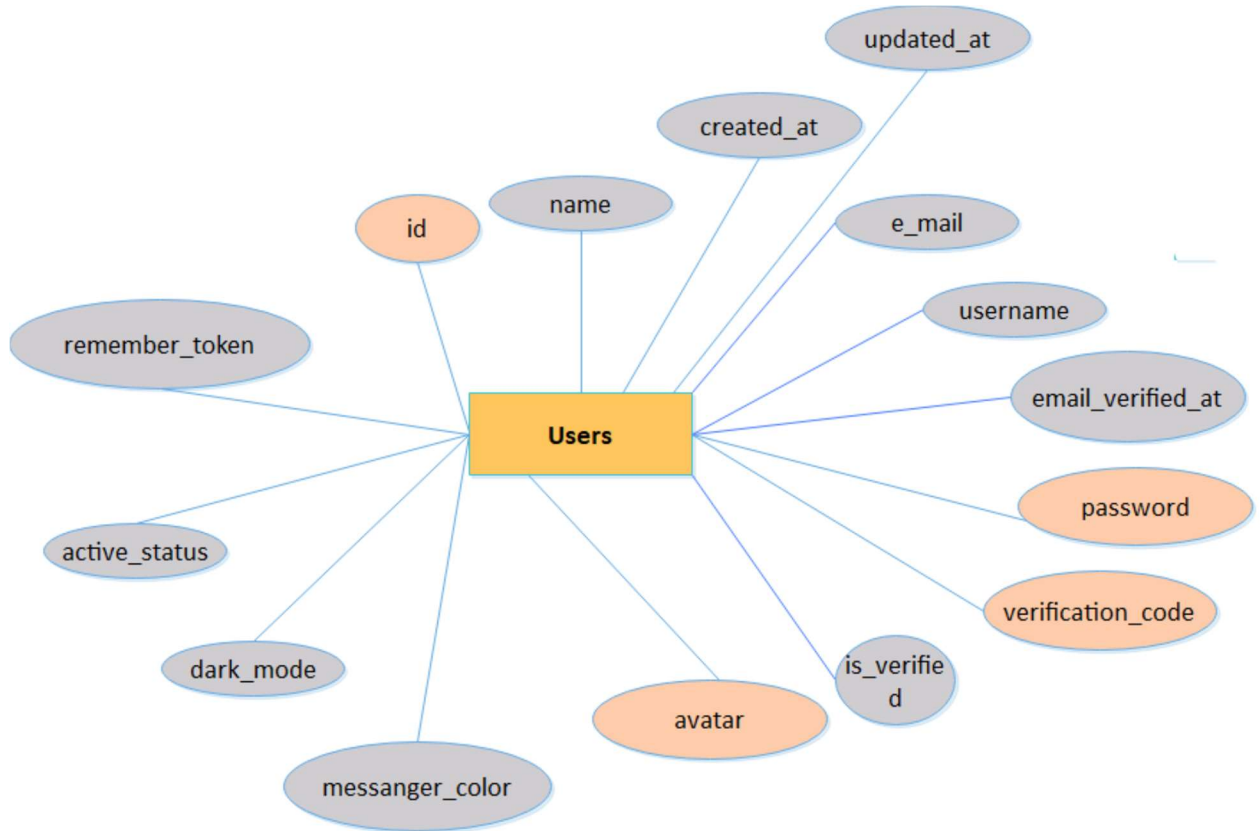
For the start the application, you should to **register** to go further. Fill the all fields and after submitted the form the **verification code** sends to your filled email address. After that user open his email and verify the email and after that user can able to login in site otherwise user can't able login on application.

User can also forget the password and change the password. For that your should to navigate the forget password page and fill the register email and submit. After that user get a link on site and on click of that link, you get the password reset page. And after you can change your password and login with new password.

- ii. **Model** – For the declaring the **User** class, we create a model '**User.php**'. Here we defined the **fillable attributes** and **hidden attributes** for data security. We defined some functions for accessing data from different table using **Eloquent ORM**. In this Model, we defined some of function to make relationship between them.

users	
id	int
name	varchar
email	varchar
username	varchar
email_verified_at	datetime
password	varchar
verification_code	varchar
is_verified	int
avatar	varchar
active_status	int
dark_mode	tinyint
messenger_color	varchar
remember_token	varchar
created_at	datetime
updated_at	datetime

iii. **Migration** – For the structure of User Entity, we need some attributes to struct the entity and relations.

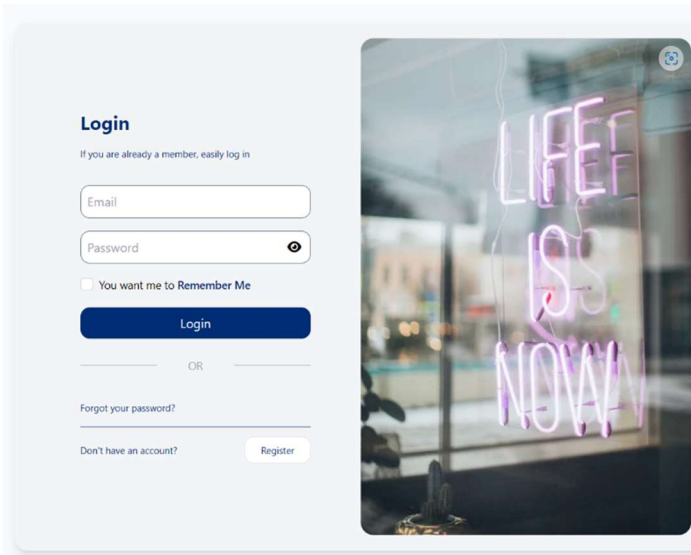


iv. **Controllers** – There are 6 controllers which make the functionality of the login system.

1. LoginController.php – This controller has functions for getting login of an user and logout the user.
2. RegisterController.php – This controller has functions for registering a user and send the verification code to user's email.
3. ResetPasswordController.php – This controller has functions for Reset the password.
4. ConfirmPasswordController.php – This controller has functions for change password to new password after forgetting.
5. ForgetPasswordController.php – This controller has functions for forget the password operations.

6. VerificationController.php – This controller has function for verify the register user and forget password user.

v. **Views –**



Login

If you are already a member, easily log in

Email

Password

☐ You want me to Remember Me

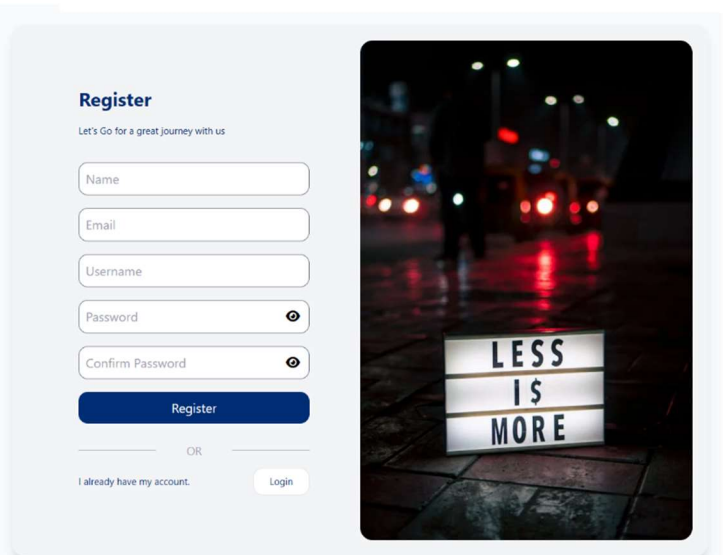
Login

OR

[Forgot your password?](#)

[Don't have an account? Register](#)

Login.blade.php



Register

Let's Go for a great journey with us

Name

Email

Username

Password

Confirm Password

Register

OR

[I already have my account. Login](#)

Register.blade.php

Reset Password

Email Address

E-Mail Address

Send Password Reset Link

[Remembered you password ? Go ahead and login](#)



Email.blade.php

Hello!

You are receiving this email because we received a password reset request for your account.

Reset Password

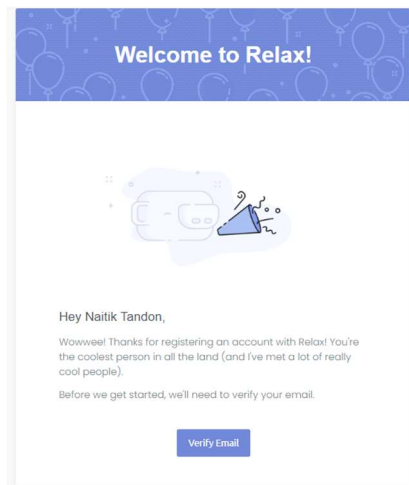
This password reset link will expire in 60 minutes.

If you did not request a password reset, no further action is required.

Regards,
Laravel

If you're having trouble clicking the "Reset Password" button, copy and paste the URL below into your web browser: <http://127.0.0.1:8000/password/reset/77f0becd953179f9868327dd93c357bb7e3136bdf2c1f3d3ded7f5b8da5e0f9?email=diyaa%40gmail.com>

confirm.blade.php

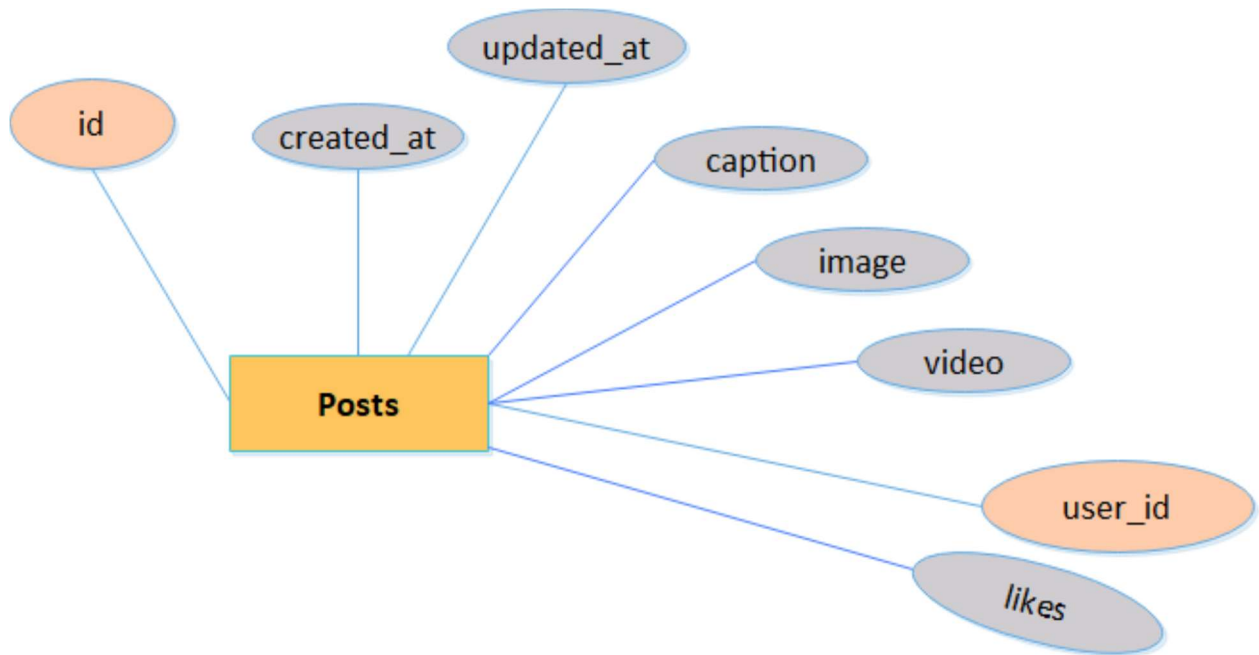


Signup-email.blade.php

b) Post Module:

- i. **Abstract** – In this project, every user can upload image or video file as a post which can see by their followers and it helps to share moments, celebrate joy with others. It basically easy to upload, you need to click on create button and a modal pop up and you can upload any image or video from your file directory and add a caption and simply click on post. Your image or video is now posted. You can see your post in your profile and you post feed.
- ii. **Model** – For the **Post** class, we create a model php file called '**Post.php**' file which contain all the functions and fillable and hidden attributes.
- iii. **Migration** – For the structure of Post Entity, we need some attributes and relations.

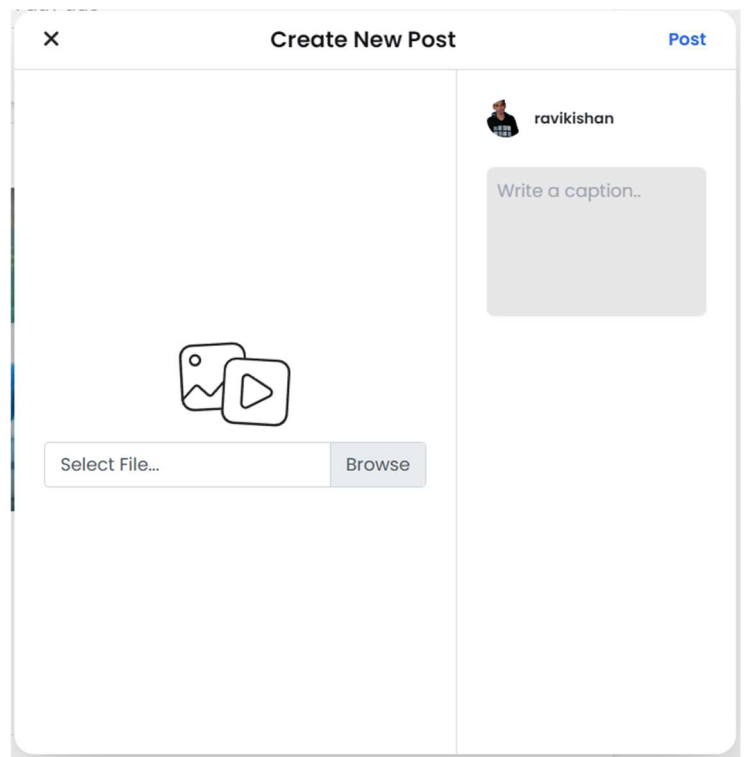
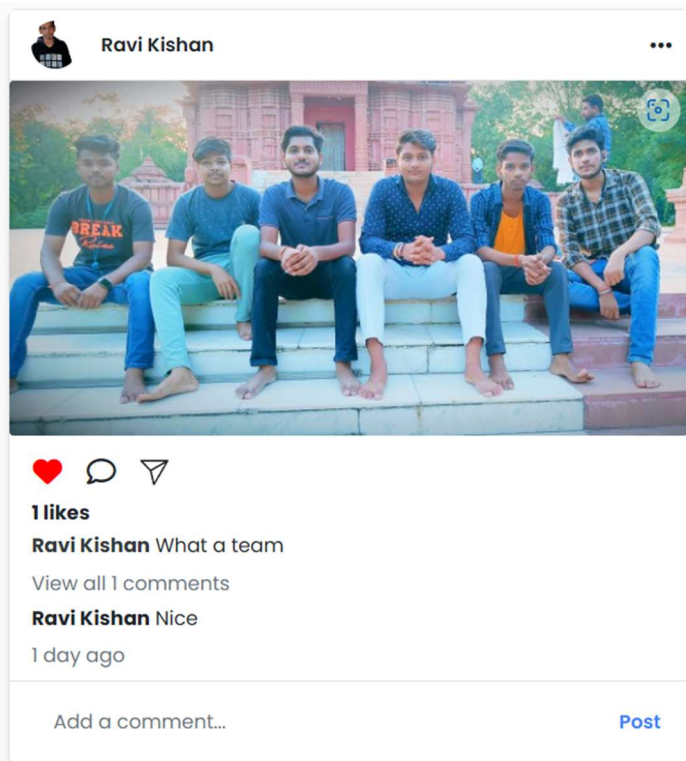
posts	
id	int
user_id	int
caption	text
image	varchar
video	varchar
likes	int
created_at	datetime
updated_at	datetime



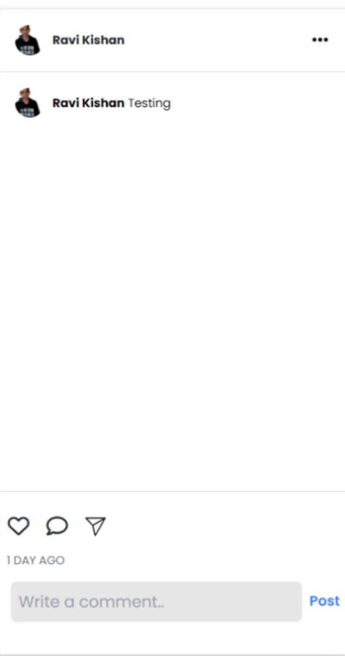
iv. **Controller** – There is a controller defined to control the operations of post manipulation. It contains some of the functions –

<i>Function name</i>	<i>Method</i>	<i>Description</i>
Constructor	ANY	this ensure only the auth user can create, update, and delete the post.
Index	GET	this shows all the posts to user on post feed
Explore	GET	this shows all the post among all users. It helps to find new users and their interesting posts.
Create	GET	this show a page where you can create a post.
Store	POST	It takes data and store the post on database.
Destroy	DELETE	It use to delete the post
Updatelikes	POST	It is use to update like on post.

v. Views –



Post.index.blade.php

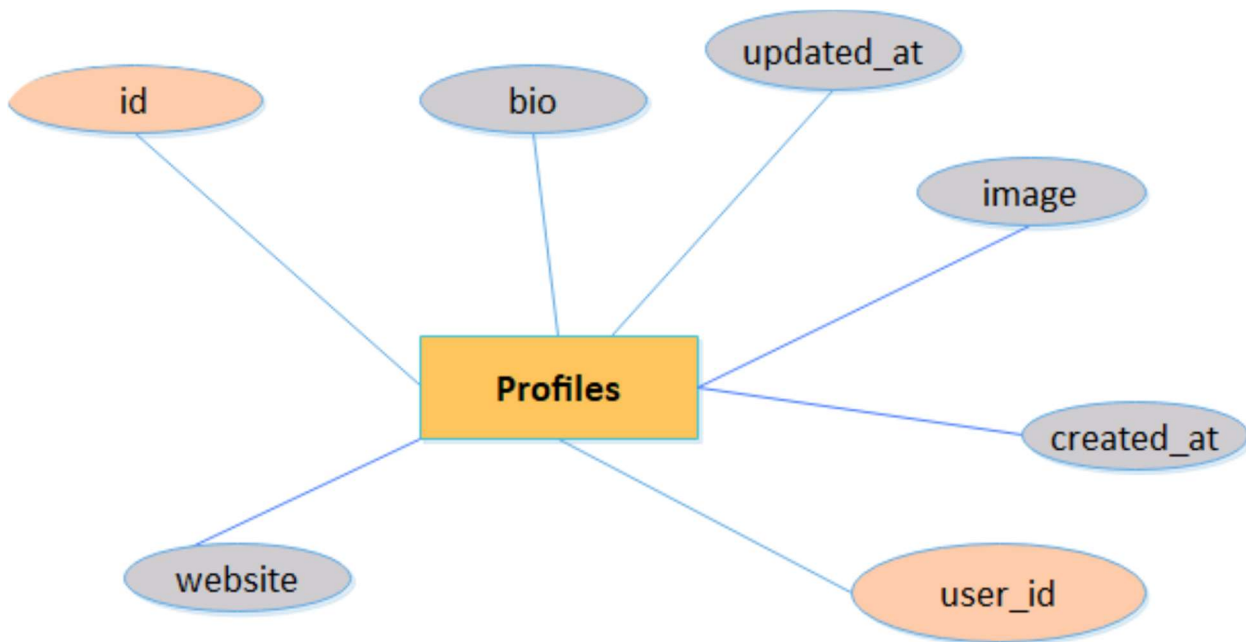


Post.show.blade.php

c) Profile Module:

- i. Abstract* – In this project, every user has a profile and he can see how many post he has, how many followers and following he gets. A user also can edit his profile details. Every user can search any user and follow him/her.
- ii. Model* – For the **Profile** class, we create a model php file called '**Profile.php**' file which contain all the functions and fillable and hidden attributes.
- iii. Migration* – For the structure of Profile Entity, we need some attributes and relations.

profile_user	
id	int
user_id	int
profile_id	int
created_at	datetime
updated_at	datetime



- iv. Controller* – There is a controller defined to control the operations of profile manipulation. It contains some of the functions –

Function name	Method	Description
Index	GET	It helps to see the profile and all details about the user.
Edit	GET	This return a view where you can edit the profile details
Update	POST	This takes data and update the profile.
Search	ANY	This takes the query and return the find matches.
Store	POST	It takes data and store the post on database.

v. Views –

Search

divy

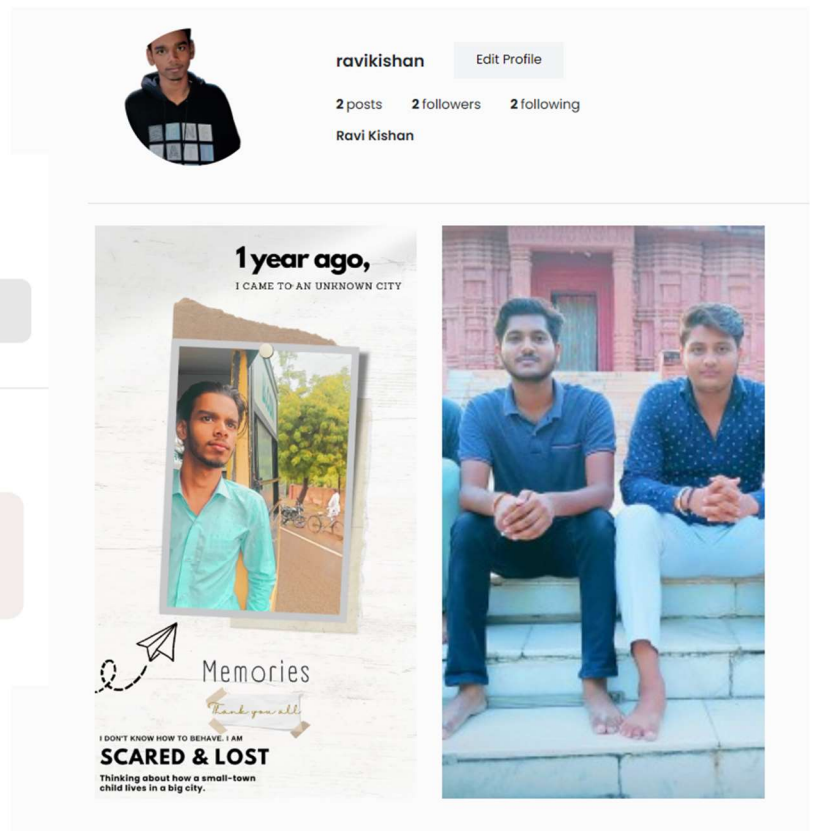
Recent

Clear all



Divya
Kdrama girl

[Profile.search.blade.php](#)



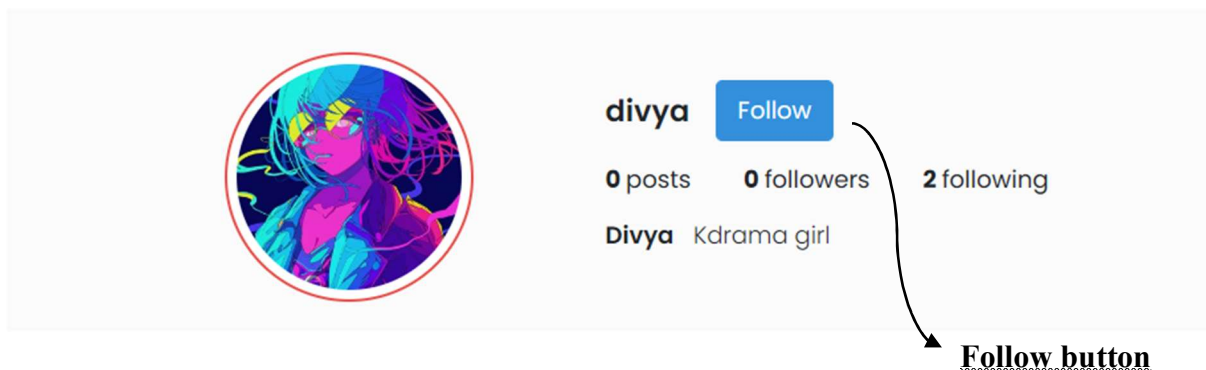
[Profile.index.blade.php](#)

d) Follow Module:

- i. **Abstract** – In this project, every user can follow any user for getting their posts as post feed to his/her application. This is a way to make connection in this application. The person who follows the other one who gets the all posts on their homepage as feed and they can like or share their posts.
- ii. **Model** – For the **Follow** class, we create a model php file called ‘**Follow.php**’ file which contain all the functions and fillable and hidden attributes.
- iii. **Migration** – For the structure of Follow Entity, we need some attributes and relations:
- iv. **Controller** – There is a controller defined to control the operations of follow manipulation. It contains some of the functions –

Function name	Method	Description
Constructor	ANY	It ensure that only auth user can follow or unfollow any other user.
Store	POST	This takes a user and add to current user’s following list.

v. Views –

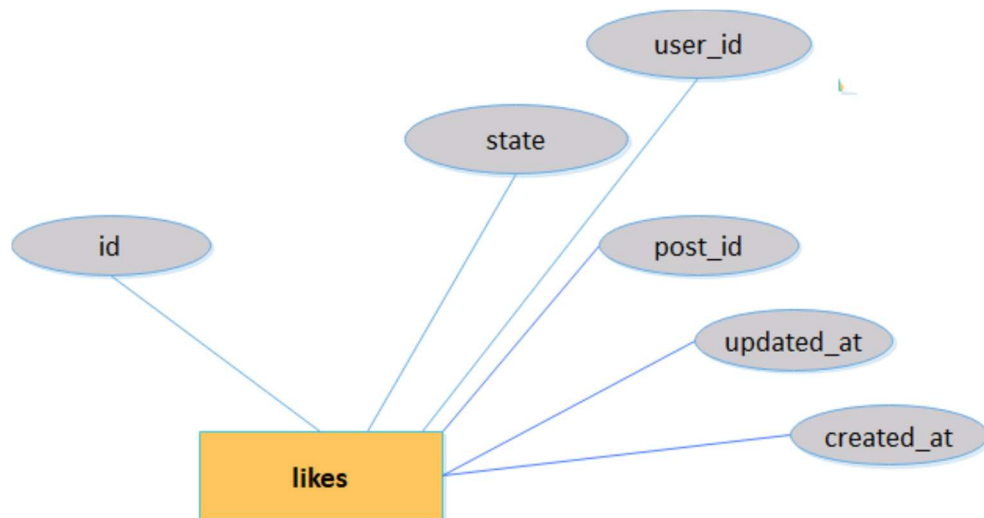


Profile.index.blade.php

e) Like Module:

- i. **Abstract** – In this project, every user can like any posts of any user. It mainly use for the popularity of any post or content. It count as an unit of like by a user means an user can like a post only one or he can dislike that post he likes.
- ii. **Model** – For the **Like** class, we create a model php file called '**Like.php**' file which contain all the functions and fillable and hidden attributes.
- iii. **Migration** – For the structure of Like Entity, we need some attributes and relations.

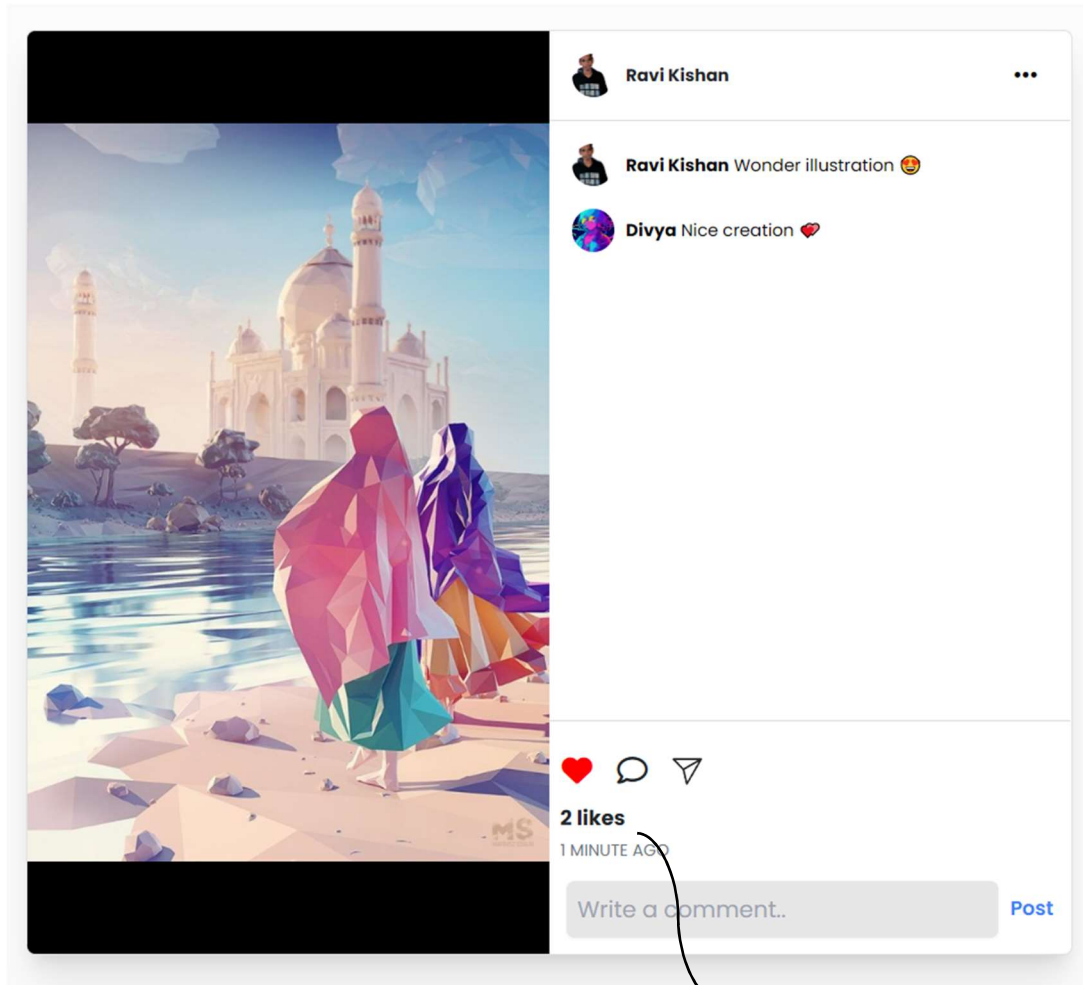
likes	
id	int
State	int
user_id	int
post_id	int
created_at	datetime
updated_at	datetime



- iv. **Controller** – There is a controller defined to control the operations of Like manipulation. It contains some of the functions –

Function name	Method	Description
Update2	POST	It takes data and update the like on post by specific user.

v. Views –



[Post.show.blade.php](#)

Likes Count

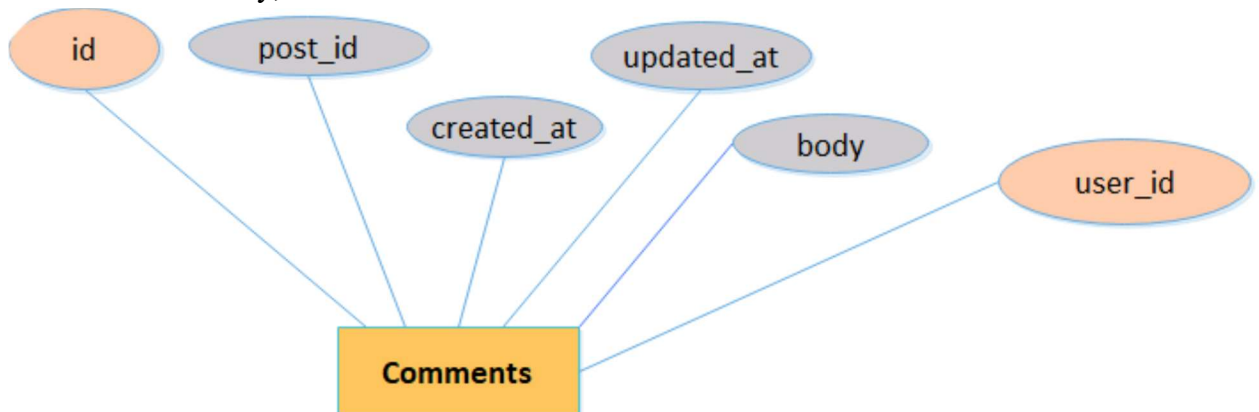
f) Comment Module:

i. **Abstract** – In this project, every user can comment on any post. He can comment any thing at once and they can create multiple comments on a single post. Comments are show with bottom of the post. Comment are shown with username and comment text.

ii. **Model** – For the **Comment** class, we create a model php file called '**Comment.php**' file which contain all the functions and fillable and hidden attributes.

comments	
id	int
user_id	int
post_id	int
body	text
created_at	datetime
updated_at	datetime

iii. **Migration** – For the structure of Comment Entity, we need some attributes and relations.

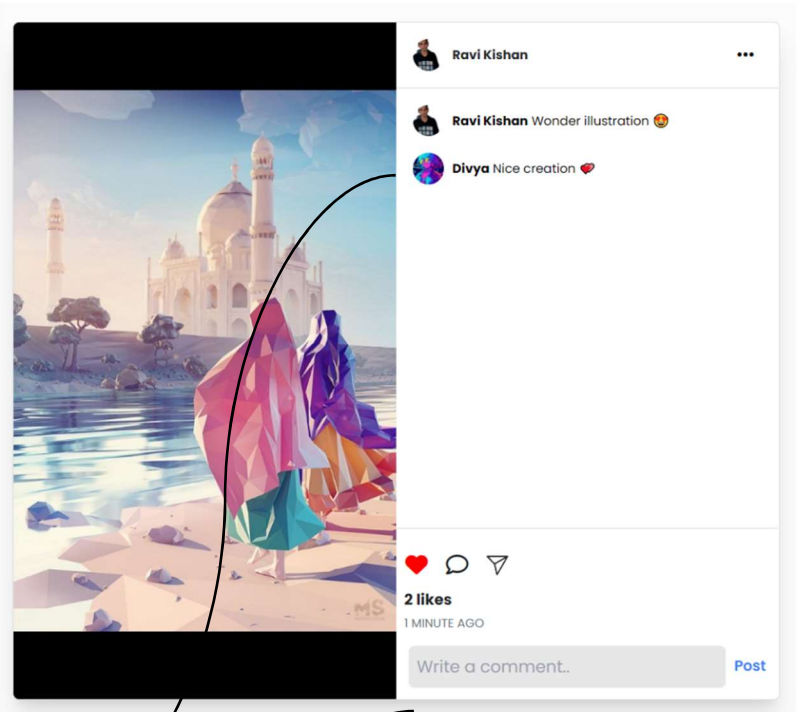


iv. **Controller** – There is a controller defined to control the operations of Comment manipulation. It contains some of the functions –

Function name	Method	Description
Constructor	ANY	It ensure that only auth user can post a comment.
Index	GET	It shows all comments on a post.

Store	POST	It takes data and store on database as a comment on a post.
-------	------	---

v. Views –



[Posts.index.blade.php](#)

[Posts.show.blade.php](#)

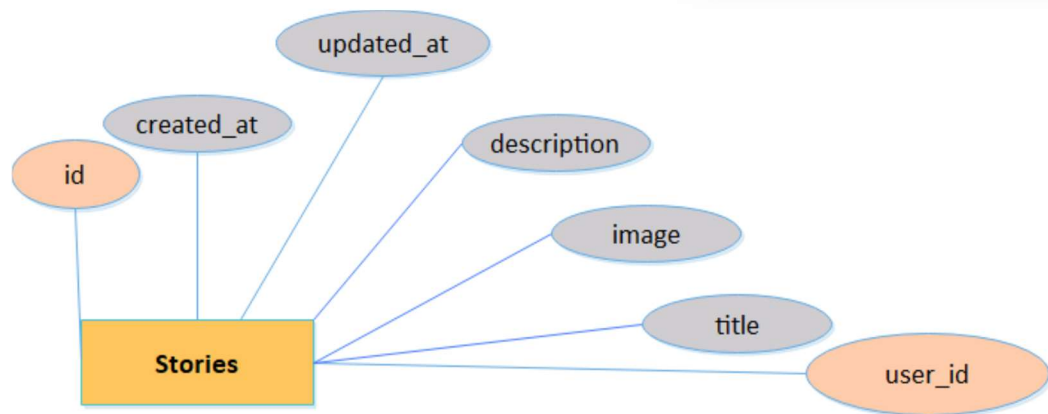
Comments

Comment Input Field

g) Story Module:

- i. **Abstract** – In this project, every user can create story. That story is saved for forever on user's profile. Anyone can see story.
- ii. **Model** – For the **Story** class, we create a model php file called '**Story.php**' file which contain all the functions and fillable and hidden attributes.
- iii. **Migration** – For the structure of Story Entity, we need some attributes and relations.

stories	
id	int
user_id	int
title	varchar
description	varchar
image	varchar
created_at	datetime
updated_at	datetime



- iv. **Controller** – There is a controller defined to control the operations of Story manipulation. It contains some of the functions –

Function name	Method	Description
Constructor	ANY	It ensure that only auth user can create a story.
Create	GET	It shows a page where you can create a story.
Store	POST	It takes data and store on database as a story of a user.
Show	GET	It shows the story to users.

v. Views –

New Story

Caption

Write a caption...

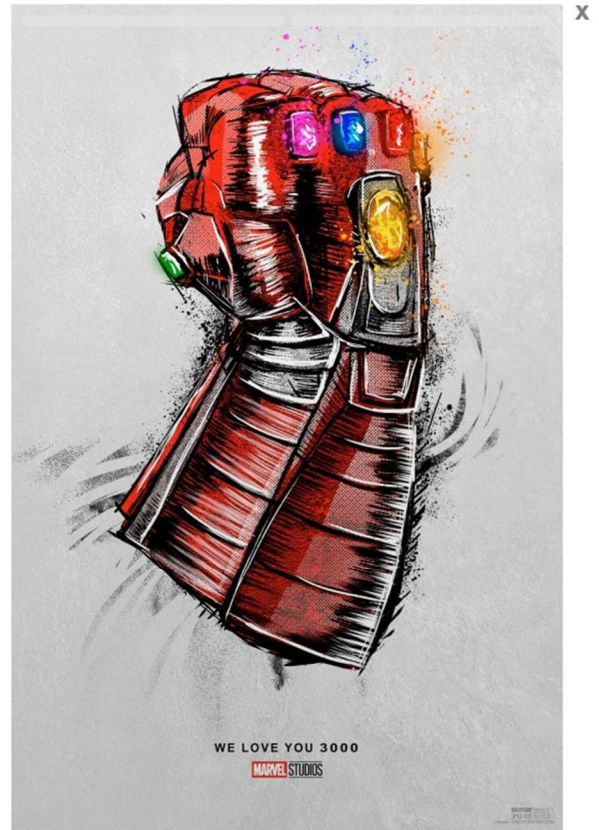
Photo

Select Photo...

Browse

Share

[Stories.index.blade.php](#)



[Stories.show.blade.php](#)

h) Chat Module:

- i. **Abstract** – In this project, every user can chat with another user. They can share any media file and message between them. It has feature to flag for send, seen. It is two-way communication in same chat section. A user can send multiple messages to multiple users

ch_favorites	
id	int
user_id	int
favorite_id	int
created_at	datetime
updated_at	datetime

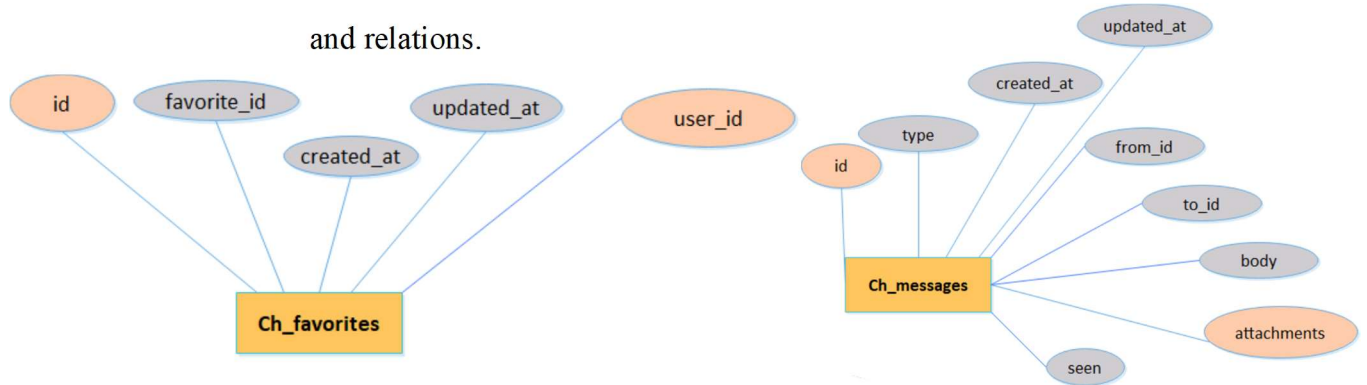
ch_messages	
id	int
type	int
from_id	int
to_id	int
body	varchar
attachment	varchar
seen	int
created_at	datetime
updated_at	datetime

ii. **Model** – For the chat section there are two models –

1. ChMessage.php

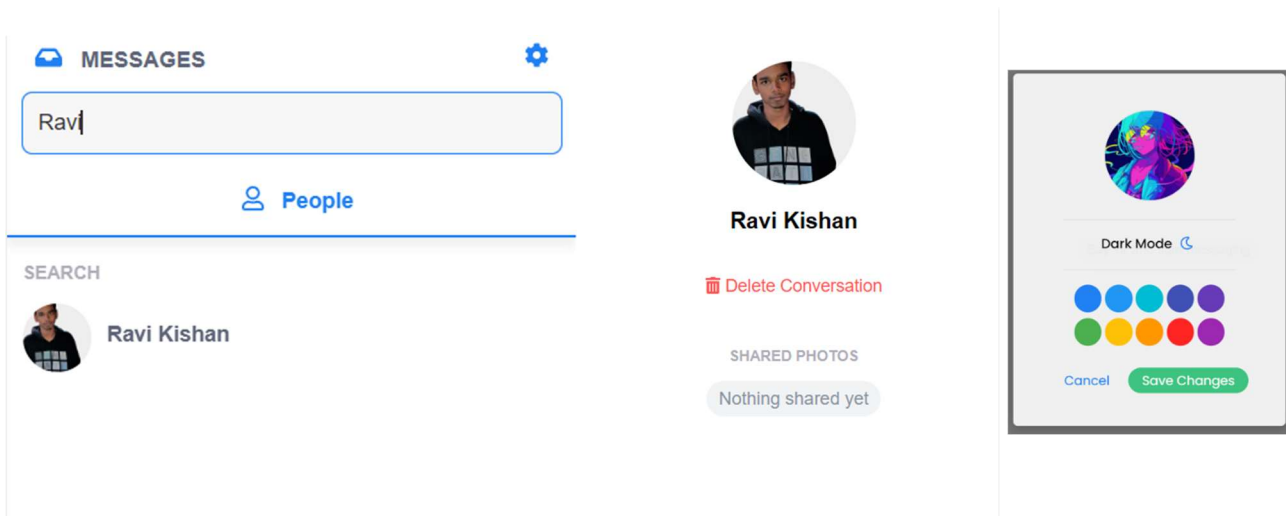
2. ChFavorite.php

iii. **Migration** – For the structure of Chat Module, we need some attributes and relations.



iv. **Controller** – It has a massive number of controllers and their operations which can't be declared in a report file.

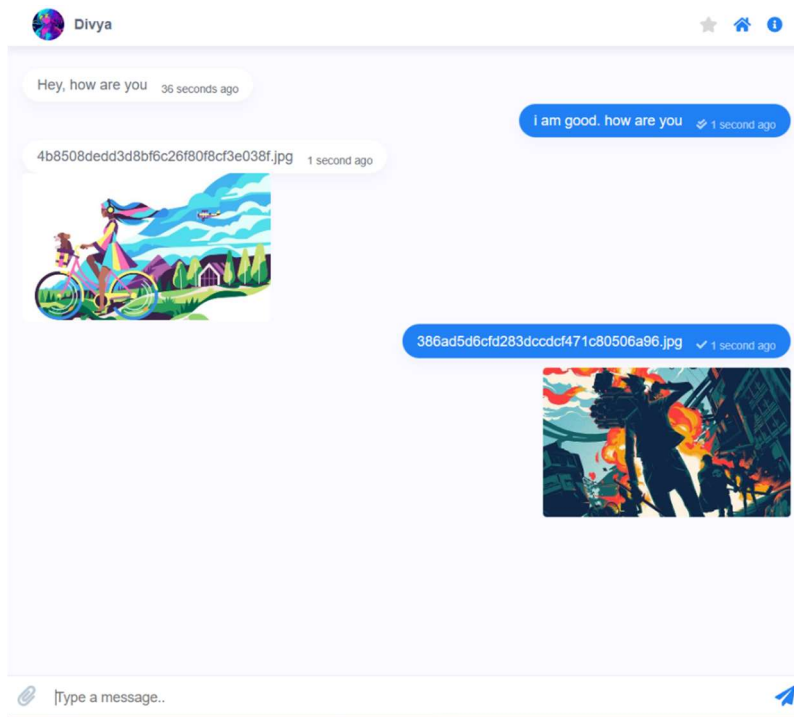
v. **Views** –



User search section

User Details section

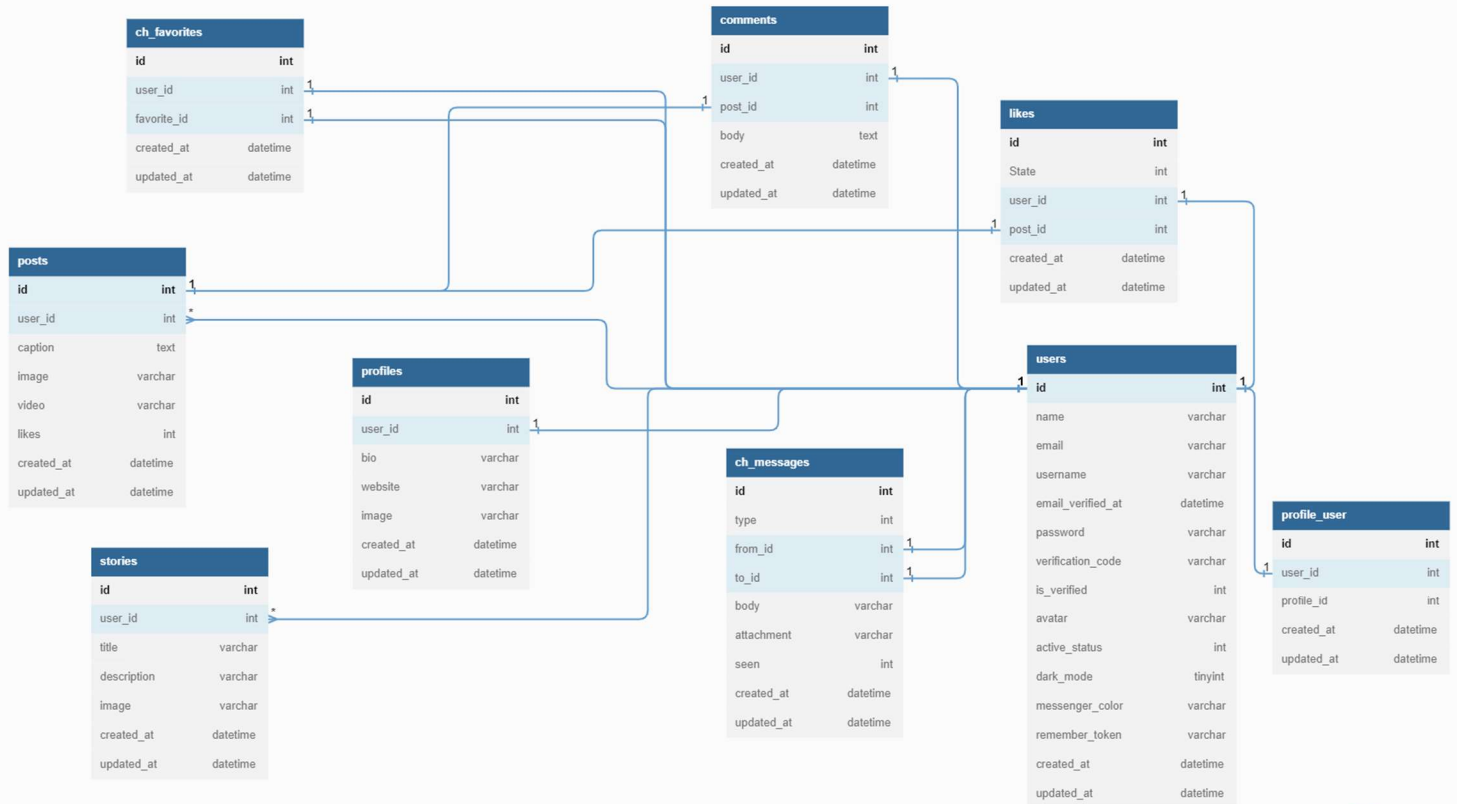
Chat Mode Setting



Chatting view section

CHAPTER 4: ER-DIAGRAM

CHAPTER 5: DATABASE RELATIONAL TABLE



UNIT – 5 : CONCLUSION & REFERENCE

Chapter 1: Conclusion

In conclusion, creating an Instagram clone web development project can be a challenging but rewarding endeavor. It requires a solid understanding of web development technologies, such as HTML, CSS, JavaScript, and PHP, as well as a proficiency in working with databases, such as MySQL. By building an Instagram clone, developers can gain hands-on experience in developing a complex web application that incorporates features like image and video upload, user authentication, and social networking capabilities.

Additionally, the project can be a great opportunity to learn about new technologies and trends in web development, such as responsive design, cloud computing, and machine learning. It can also be an opportunity for developers to explore different ways of building a scalable, high-performance web application that can handle large amounts of data and traffic.

Finally, it should be highlighted that building an Instagram clone is not only about replicating the features and the design, but also thinking about adding new value and new features to the project. It's also important to keep in mind that the development process is only half of the story, and the project will also need to be properly tested, deployed, and maintained in order to be successful.

Chapter 2: Reference

¥ **W3SCHOOL** - <https://www.w3schools.com/>

¥ **LARAVEL** – <https://laravel.com/>

¥ **TAILWINDCSS** – <https://tailwindcss.com/>

¥ **BOOTSTRAP** – <https://getbootstrap.com/>

¥ **MAILTRAP** – <https://mailtrap.io/>

¥ **PUSHER** – <https://pusher.com/>

CODE REPOSITORY: - <https://github.com/Ravikisha/relax>