COMP3385 Assignment 2

Due Date:

February 25, 2024 at 11:59 PM

One of the most basic functions in a web application is the ability to send emails to your users. Laravel provides you with the Mail Facade and Mailable Classes that can be used within your application to send emails.

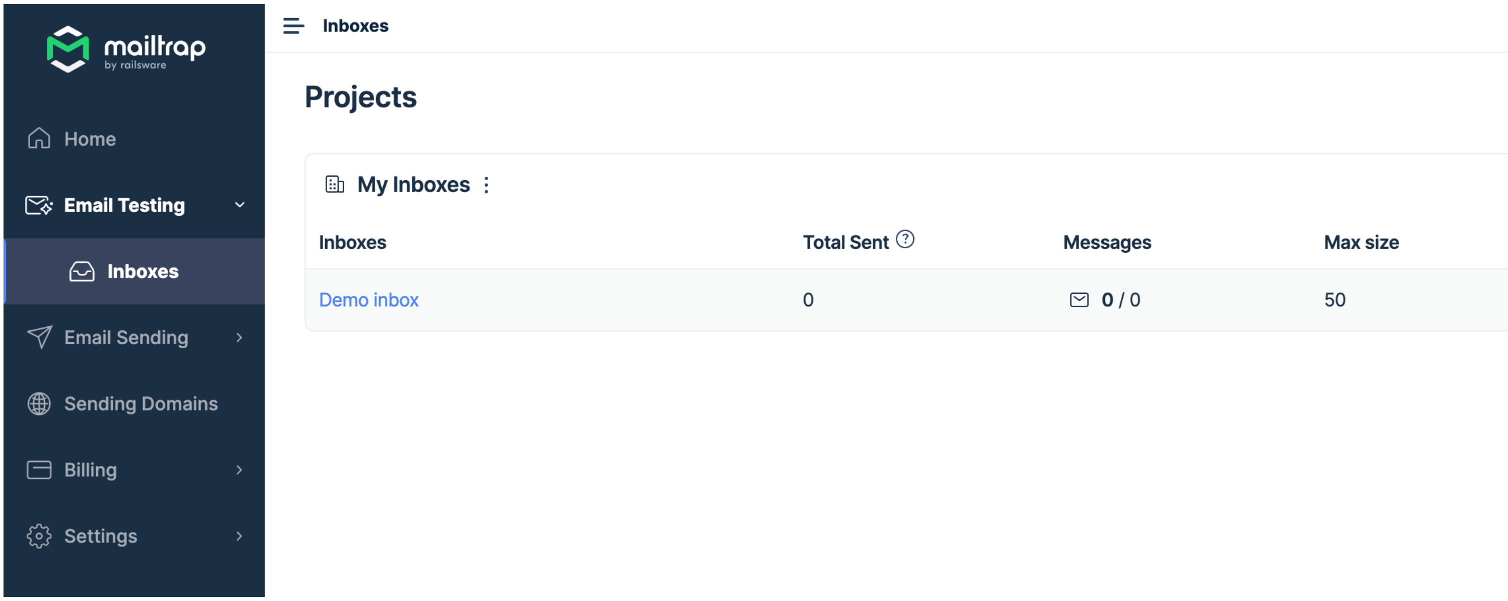
In this assignment you will be creating a form that when submitted will send an email to a Fake SMTP server.

Exercise 1 - Setup a Fake SMTP Server

Typically you would have a proper SMTP server setup or you could use an existing email account like your Gmail or Outlook account. However, in today's lab we will use a fake SMTP server called Mailtrap. Mailtrap allows development teams to test, view and share emails sent from the development and staging environments without spamming real users.

Sign up for a FREE account at https://mailtrap.io/register/signup.

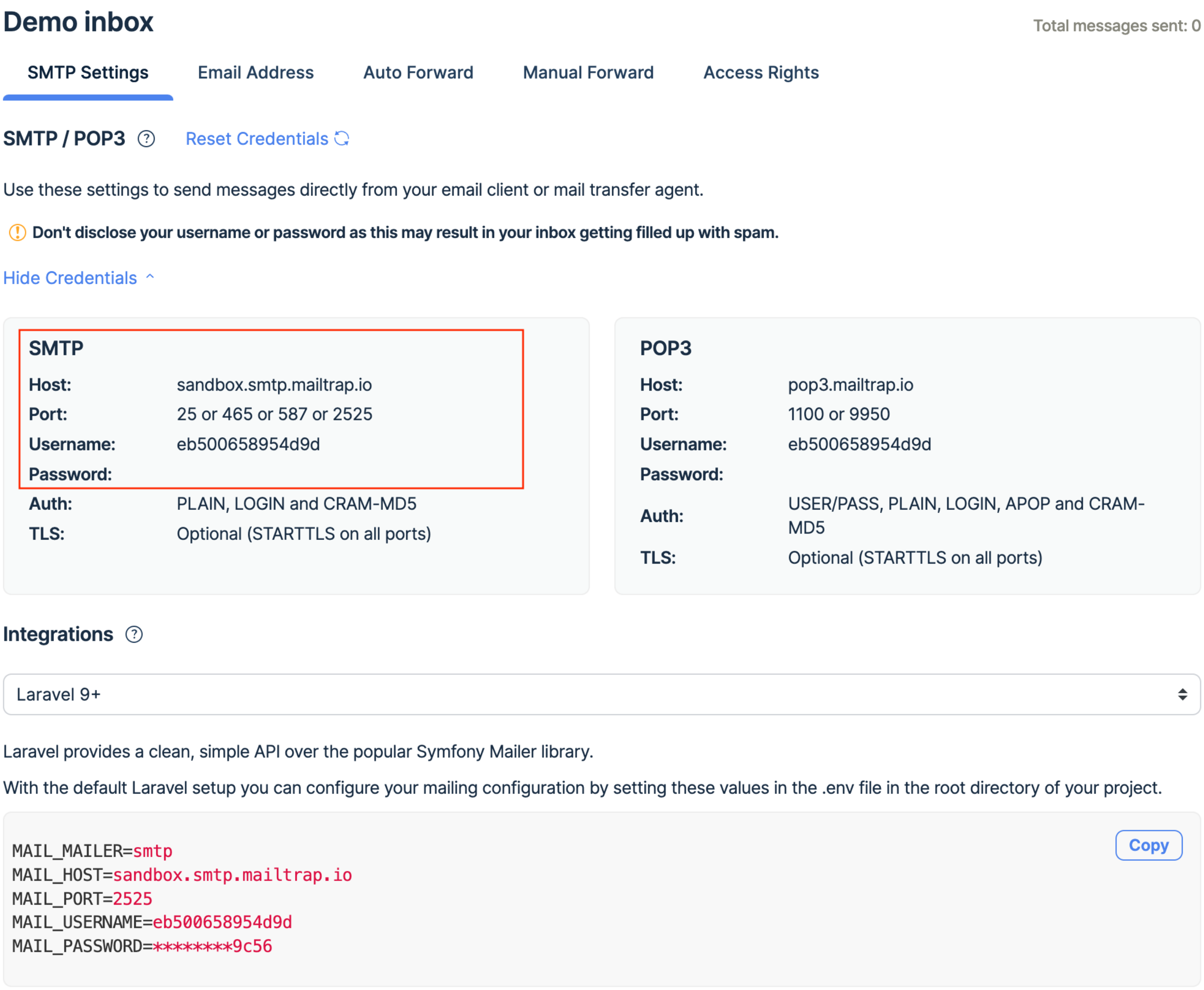
Once you have signed up for an account, you should see your inboxes and there should be one called 'My Inbox' or 'Demo inbox'.



Inboxes	Total Sent	Messages	Max size
Demo inbox	0	0 / 0	50

Figure 1

Select 'My Inbox' or 'Demo inbox' and you should see an Empty inbox. Then click on the "Show Credentials" link under SMTP Settings so that you can see the necessary credentials to connect to this Fake SMTP server.



SMTP / POP3

Use these settings to send messages directly from your email client or mail transfer agent.

Don't disclose your username or password as this may result in your inbox getting filled up with spam.

SMTP	
Host:	sandbox.smtp.mailtrap.io
Port:	25 or 465 or 587 or 2525
Username:	eb500658954d9d
Password:	
Auth:	PLAIN, LOGIN and CRAM-MD5
TLS:	Optional (STARTTLS on all ports)

POP3	
Host:	pop3.mailtrap.io
Port:	1100 or 9950
Username:	eb500658954d9d
Password:	
Auth:	USER/PASS, PLAIN, LOGIN, APOP and CRAM-MD5
TLS:	Optional (STARTTLS on all ports)

Integrations

Laravel 9+

Laravel provides a clean, simple API over the popular Symfony Mailer library.

With the default Laravel setup you can configure your mailing configuration by setting these values in the .env file in the root directory of your project.

```
MAIL_MAILER=smtp
MAIL_HOST=sandbox.smtp.mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME=eb500658954d9d
MAIL_PASSWORD=********9c56
```

Figure 2

NOTE

Ensure you make note of the Host, Port (try to use 465 or 2525) and the Username and Password. We will need these in the next exercise.

Exercise 2 - Get the Starter Code and Setup your .env file

1. Fork the Assignment 2 repository at https://github.com/uwi-comp3385/comp3385-assignment-2.git

2. Then at the command line clone your repository using: `git clone https://github.com/{yourusername}/comp3385-assignment-2.git`

3. Change directories into your new comp3385-assignment-2 folder using `cd comp3385-assignment-2`.

4. Create copy `.env.example` and rename it to `.env`.

5. Generate a new `APP_KEY` in your `.env` file by running the artisan command `php artisan key:generate`.

6. Open your `.env` file and edit your `MAIL_*` settings. For example:

```ini
MAIL_MAILER=smtp
MAIL_HOST=sandbox.smtp.mailtrap.io
MAIL_PORT=2525
MAIL_USERNAME="your mailtrap SMTP username"
MAIL_PASSWORD="your mailtrap SMTP password"
```

7. Start the Laravel development server `php artisan serve`.

Exercise 3 - Create a new Route, Template and Form

1. Create a new controller called `FeedbackController.php` in your `app\Http\Controllers` directory.

2. Within that controller create two new methods. One called `create` and another called `send`.

3. In your `routes/web.php` file, create two new routes. One for `/feedback` and the other for `/feedback/send`. The first should be a **GET** request and the 2nd should be a **POST** request pointed to the appropriate controller and methods created in the previous step.

4. In your Controller, your `create` method should load an appropriate template file to display a form that looks like the screenshot in Figure 3. This template file should be called `feedback-form.blade.php` and should extend from the `layouts/app.blade.php` base layout included in your starter code. It should look something like this:

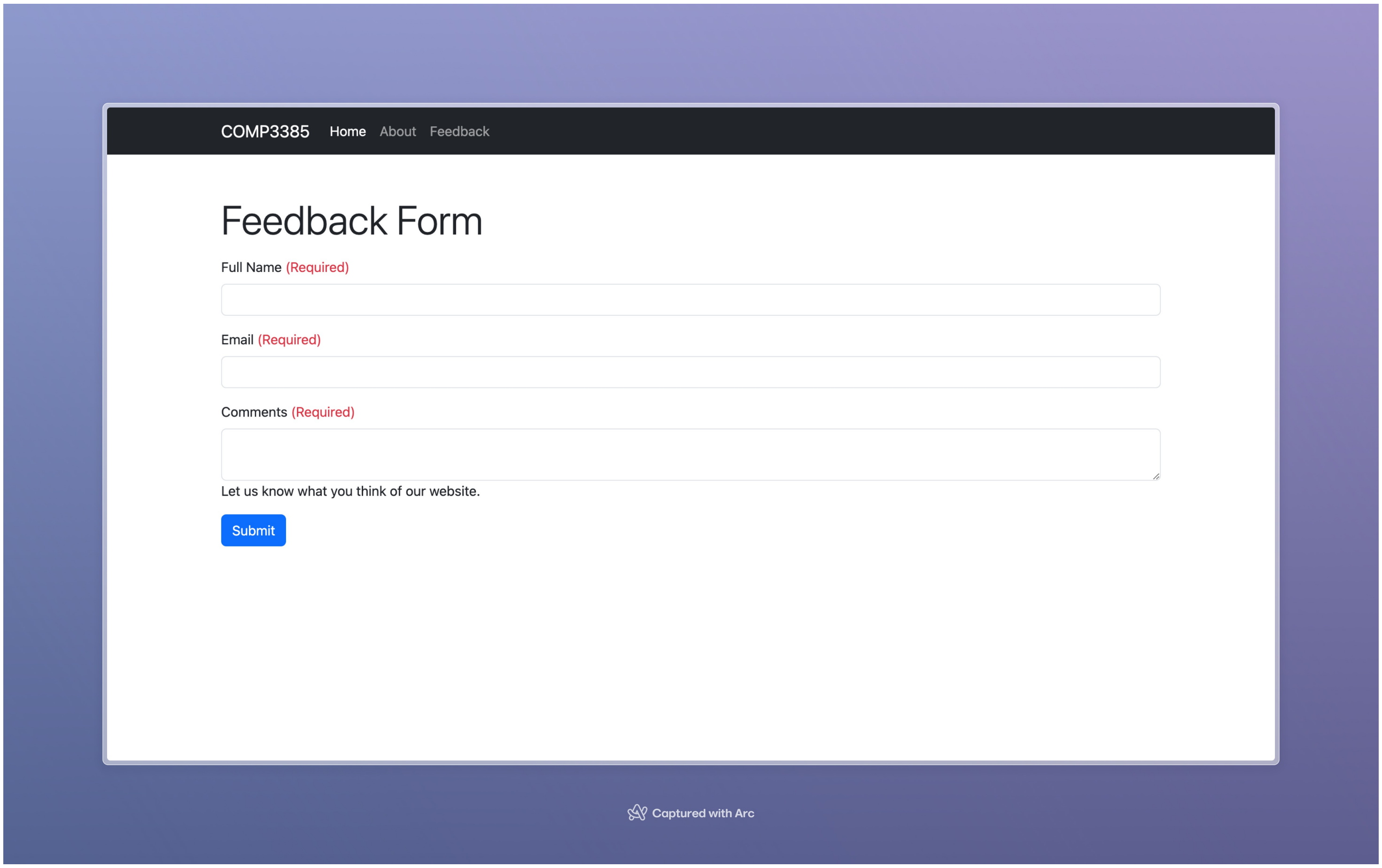


Figure 3

> **Note**
>
> Don't forget to ensure that your form includes a CSRF Token. You do this by printing the following in your template file (within the `<form></form>` tags).
>
> `@csrf`

5. Open your `resources/views/layouts/app.blade.php` and add a link to your newly created Feedback form in the navigation.

> **Checkpoint**
>
> Now would be a good time to commit these changes to your code to your repository and push to Github.

Exercise 4 - Process the form submission to send the email

Now we will process our form submission and have it send an email to our Fake SMTP Server.

1. Get your form input using the `$request->input()` method. Also ensure that you validate each input field first using at least the `required` validator and any other validator necessary based

on the fields of the form.

> **Note**
>
> You should display any form validation error messages if the user does not complete all the form fields with the appropriate information. You can print out the error messages in your template file using the following snippet
>
> ```php
> @if ($errors->any())
>     <div class="alert alert-danger">
>         <ul>
>             @foreach ($errors->all() as $error)
>                 <li>{{ $error }}</li>
>             @endforeach
>         </ul>
>     </div>
> @endif
> ```

2. Next, create a Laravel **Mailable** class to use to send your email. To do this, run the following command `php artisan make:mail Feedback`. This will create a new class that is stored in the `app/Mail` directory of your codebase.

   You should see several methods including the `envelope`, `content` and `attachments` methods.

3. Now update your class `__construct` method to accept the `$fullname`, `$email` and `$comment` from the form.

```php
    public function __construct(
        public string $name,
        public string $email,
        public string $comment
    )
    {
        //
    }
```

4. Then inside the `envelope` method, update it to have the message sent from the User who filled out the form.

```php
    public function envelope(): Envelope
    {
        return new Envelope(
            from: new Address($this->email, $this->name),
            subject: 'Feedback from ' . $this->name,
        );
    }
```

5. And finally in the `content` method, set the first parameter to use a new template file for the body of your email. We can create and store this new template file in `resources/views/mail/feedback.blade.php`.

```php
public function content(): Content
{
    return new Content(
        view: 'mail.feedback',
    );
}
```

Then in your `mail/feedback.blade.php` template, you can do something like the following:

```blade
@extends('layouts.email')

@section('content')
<p>{{ $name }} has sent feedback about your website. Below are the comments left by
<p>{{ $comment }}</p>
@endsection
```

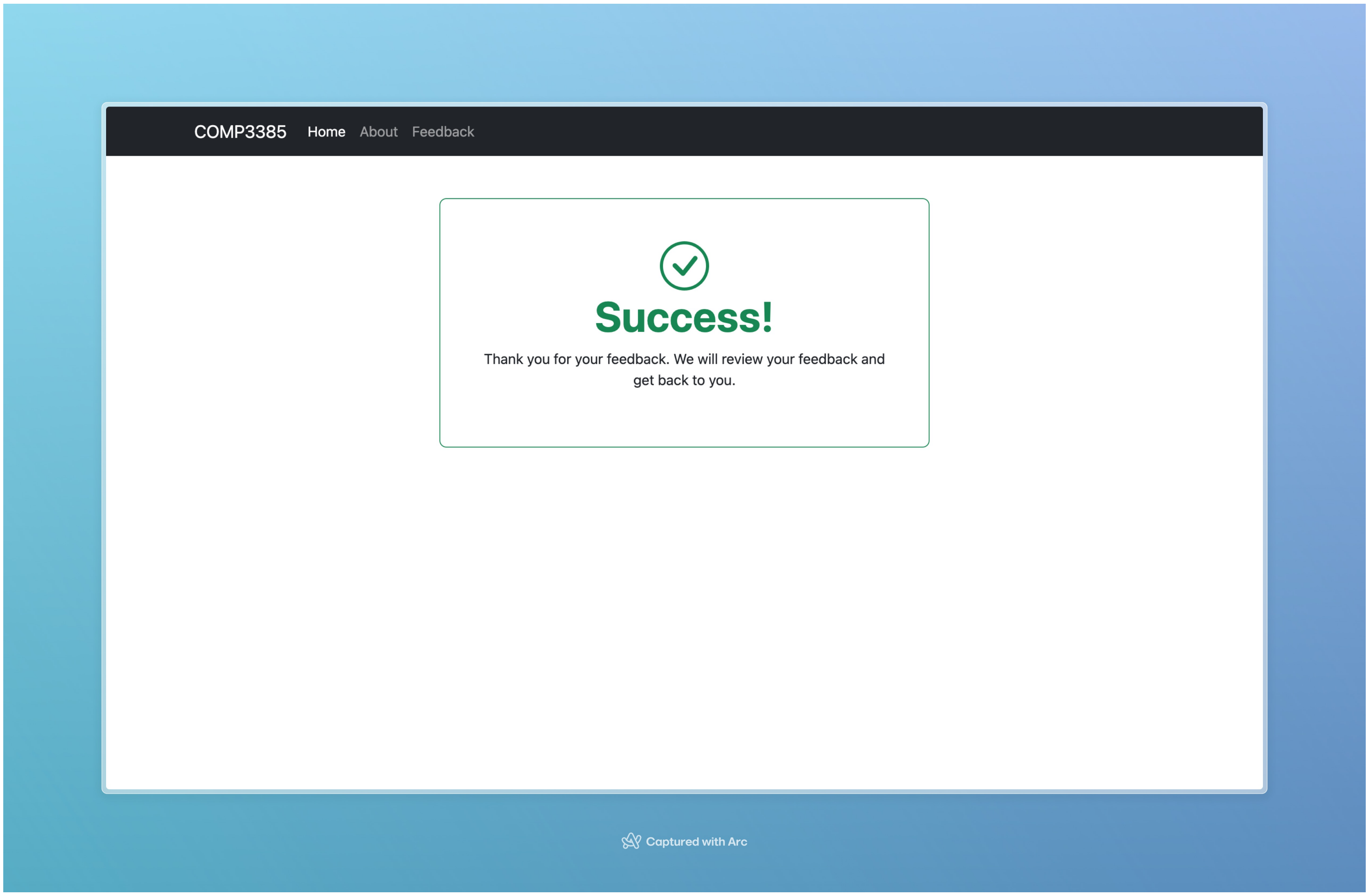
6. Now back in your controller, import your `Feedback` mailable class and pass it as a parameter to the `send` method of the `Mail` facade. e.g.

```php
use App\Mail\Feedback;
use Illuminate\Support\Facades\Mail;
// ...
Mail::to('comp3385@uwi.edu', 'COMP3385 Course Admin')
    ->send(new Feedback($fullname, $email, $comment));
```

7. After you send the message, ensure you `redirect` the user to another page with the URL `/feedback/success` and display a success message. See *Figure 4*.

   Hint

   Remember you will need to create the **route**, a new **method** in your `FeedbackController` and a **template file** that extends your app layout.



Success!

Thank you for your feedback. We will review your feedback and get back to you.

Figure 4

8. Now try filling out your form, tapping the Submit button and see what happens. If all goes well you should be able to look in your Mailtrap Inbox and see the email sent.

Checkpoint

Now would be a good time to commit these changes to your code to your repository and push to Github.

Submission

Submit your code via the "Lab 2" link on the VLE. You should submit the following links:

- Github repository URL for your Laravel Exercise e.g. https://github.com/{yourusername}/comp3385-assignment-2