

Laravel Assessment Task

Brief

The web application that needs to be developed will be a basic url-shortening service. It will have a basic form that consists of 1 text input field and 1 submit button. The text input field will be used for “destination url” which will basically be the long url that will be shortened. Below the input field, there should be a simple table showing the latest created urls on the system.

The form should be only accessible by logged in users, meaning the application would need to have some sort of basic authentication for login and registration (there’s no need to do this from scratch).

Requirements

1. Create a *Url* model with the following properties:
 - a. Id
 - b. destination (string): it will contain the long url.
 - c. slug (string): it will contain the slug of the shortened url.
 - d. Timestamps (created at, updated at)
2. Add a model accessor to the model above to retrieve the full shortened URL as an attribute and make sure this attribute is appended to the model whenever it is retrieved.
3. Create a migration for the model above.
4. Implement an auth system for login and registration.
5. Create a form that will only be accessible for registered and logged in users.
6. Create an input text field for the “destination” or long-url and a submit button.
7. Add validations to the form so it will not allow the text field to be “empty” and to only allow valid urls.
8. The form once valid and submitted, it should store the destination url in the DB using the Url model. It is also required to generate a 5 characters long random string that will be used as the “slug” property.
9. Create the routes needed and make use of model binding for them when applicable (for Url model).
10. Make sure that the route that will redirect the shortened url to the actual long-url is created.
11. Use of laravel helpers are encouraged.
12. Create a simple POST REST API endpoint that takes “destination” as a required property of the request body. If no “destination” is provided then it should return a validation error. The same is applied for when a non-valid url is provided. Sample of valid payload for the endpoint:

```
{  
  
"destination": "https://google.com"  
  
}
```


Expected json response on success:

```
{
  "destination": "https://google.com",
  "slug": "k9GZw",
  "updated_at": "2021-09-10T23:52:11.000000Z",
  "created_at": "2021-09-10T23:52:11.000000Z",
  "id": 18,
  "shortened_url": "http://url-shortener.test/k9GZw"
}
```

1. Make sure to use a git repository (either bitbucket or github) to store the code of your application. Then share the url of the repository with us. Make sure the repository is public.
2. Laravel scaffolding is welcomed (and encouraged).

Screenshot (sample application)

Here's a screenshot of the overall components required for the application. Styling is optional and does not have to match the screenshot.

 Dashboard test ▾

Url Shortening

Enter the url you want to shorten.

Shorten Url

SLUG	URL	CREATED AT
http://url-shortener.test/585Vd	https://google.com	09/10/2021, 7:07 PM
http://url-shortener.test/vuW1N	https://google.com	09/10/2021, 7:04 PM
http://url-shortener.test/fbOdt	https://twitter.com	09/10/2021, 6:29 PM
http://url-shortener.test/pRSeE	https://facebook.com	09/10/2021, 6:29 PM
http://url-shortener.test/GcUxw	https://google.com	09/10/2021, 6:29 PM