



The PHP Framework for Web Artisans

Laravel is a web application framework with expressive, elegant syntax. We've already laid the foundation — freeing you to create without sweating the small things.

<https://laravel.com/>

Install Laravel 5.4

- Login to the DAW server (IP 10.10.23.183) on the DEEI network using PuTTY and change to the public_html folder

```
a999999@daw:~/public_html/$  
composer create-project --prefer-dist laravel/laravel=5.4.* LAB9_10
```

This command will create a directory named `LAB9_10` containing a fresh Laravel installation with all of Laravel's dependencies already installed

- Give the permissions to the web server to write in selected directories

```
a999999@daw:~/public_html/$ chmod -R g+w LAB9_10/storage  
a999999@daw:~/public_html/$ chmod -R g+w LAB9_10/bootstrap/cache  
a999999@daw:~/public_html/$ mv LAB9_10/server.php LAB9_10/index.php
```

Configure Laravel

- Setup the database in hidden file `.env`.

```
a999999@daw:~/public_html/project$ nano .env
```

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=db_a999999
DB_USERNAME=a999999
DB_PASSWORD=password
```

```
MAIL_DRIVER=sendmail
MAIL_HOST=127.0.0.1
MAIL_PORT=25
MAIL_USERNAME=null
MAIL_PASSWORD=null
MAIL_ENCRYPTION=null
```

Welcome to Laravel !

- Point your browser to

http://all.deei.fct.ualg.pt/~a999999/LAB9_10

Laravel

DOCUMENTATION

LARACASTS

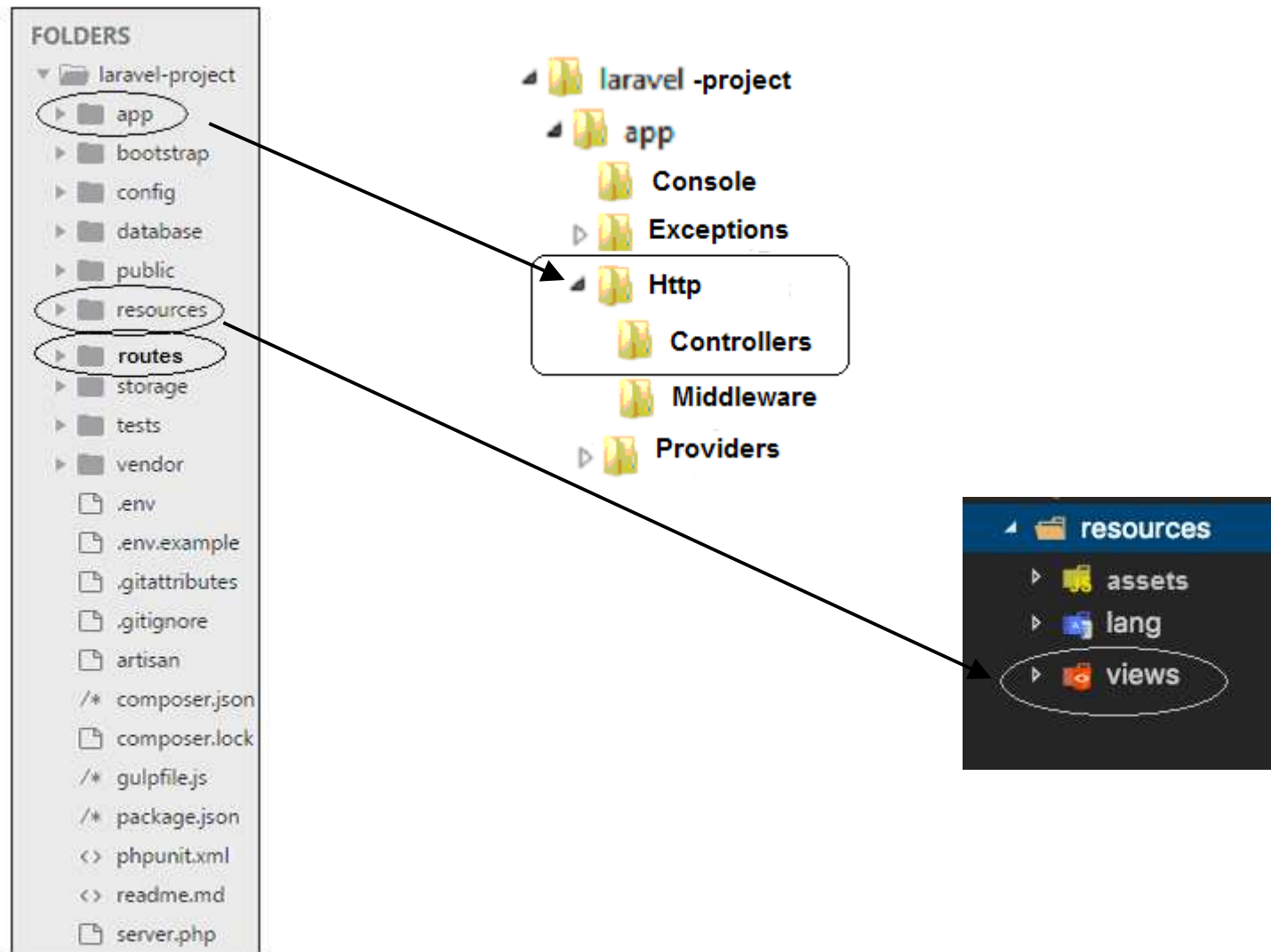
NEWS

FORGE

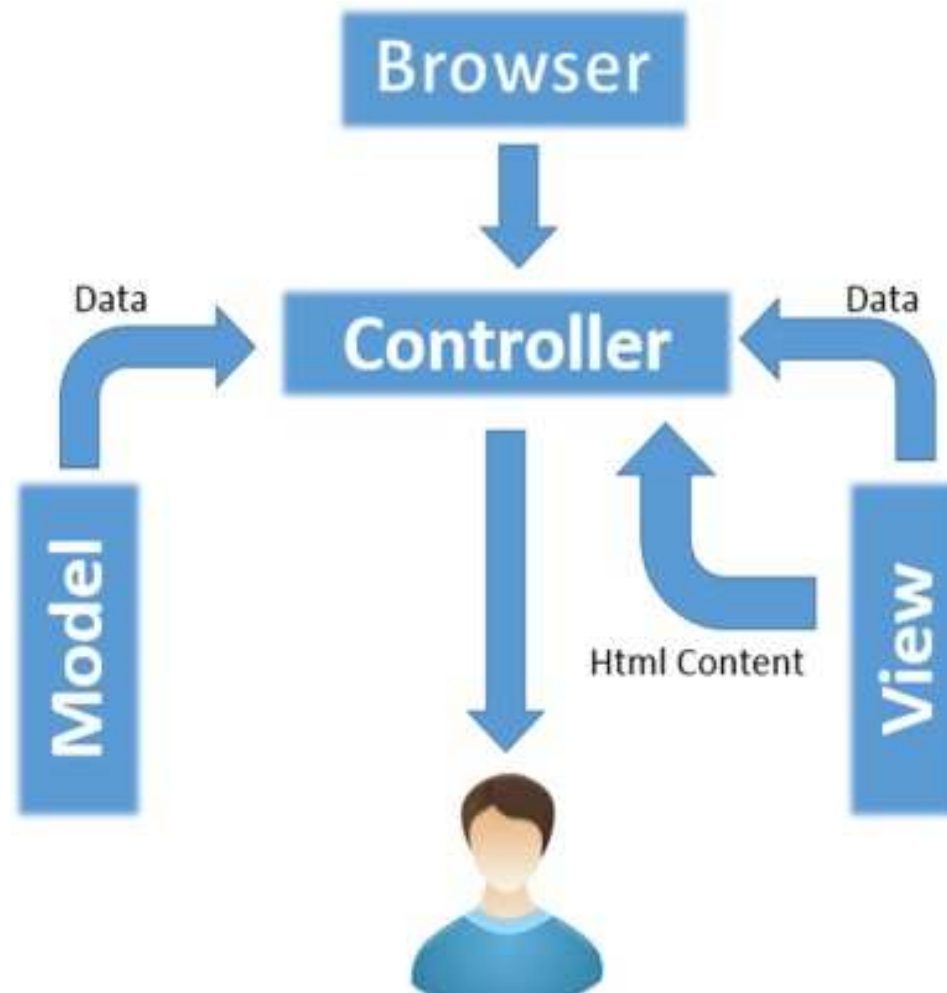
GITHUB

If you see the web page above you have successfully installed Laravel!

Directory Structure



MVC Paradigm



MVC Paradigm ...

- **Controllers** – This folder holds the controllers of the application.
- **app** – The database model is placed in this folder.
- **views** – Application's template files are placed in this folder.

The controller class

```
<?php
namespace App\Http\Controllers;
use Illuminate\Http\Request;
use App\Stud_model;

class Stud extends Controller
{
    public function index()
    {
        $records = Stud_model::get_all();
        return view('stud_view', compact('records'));
    }

    public function delete($roll_no)
    {
        Stud_model::delete($roll_no);
        return redirect() -> route('stud.view');
    }
}
```


The controller class ...

- The general syntax for calling the controller is

`http://www. domain.com/index.php/controller/method/arg`

- If rewriting base is active in `.htaccess`

`http://www. domain.com/controller/method/arg`

URL Rewriting

The segments in a URL normally follow this pattern

```
www.domain.com/class/method/arg1/arg2/...
```

- The **first segment** represents the controller class
- The **second segment** represents the class function (method)
- The **third segment**, and any additional segments, represent any variables that will be passed to the controller.

URL Rewriting ...

- URL `/~a9999/LAB9_10/index.php/stud`

gets method `index()` in file `Stud.php`

```
/users/a9999/public_html/LAB9_10/app/Http/Controllers/  
/Stud.php
```

- URL `/~a9999/LAB9_10/index.php/stud/delete_student/89067`

gets method `“delete_student()”` in file `Stud.php`

```
/users/a9999/public_html/LAB9_10/app/Http/Controllers/Stud.php
```

... and passes value `“89067”` to the `delete_student()` method

Laravel Routes

Routes/web.php

```
Route::get('/stud', 'Stud@index' ) -> name('stud.view');
```

```
Route::get('/stud/add', 'Stud@add' ) -> name('stud.add');
```

```
Route::post('/stud/add_action', 'Stud@add_action');
```

```
Route::get('/stud/update/{number}', 'Stud@update');
```

```
Route::post('/stud/update_action', 'Stud@update_action');
```

```
Route::get('/stud/delete/{number}', 'Stud@delete');
```

The model class

App/Stud_model.php

```
<?php
```

```
namespace App;
```

```
use Illuminate\Support\Facades\DB;
```

```
class Stud_model
```

```
{
```

```
    public static function get_all()
```

```
    {
```

```
        $students = DB::select("SELECT * FROM stud");
```

```
        return $students;
```

```
    }
```

```
    public static function get($roll_no)
```

```
    {
```

```
        $sql = "SELECT * FROM stud WHERE roll_no='$roll_no'";
```

```
        $query = DB::select($sql);
```

```
        return $query;
```

```
    }
```

```
}
```

The views (with BLADE template)

```
<!DOCTYPE html> <html lang = "en">
  <head>
    <meta charset = "utf-8">
    <title>Students Example</title>
  </head>
  <body>
    <a href = "{{route('stud.add')}}">Add</a>
    <table border = "1">

      <tr>
        <td>Sr#</td>
        <td>Roll No.</td>
        <td>Name</td>
        <td>Edit</td>
        <td>Delete</td>
      <tr>
        @foreach ($records as $r)
          <tr>
            <td>{{ $loop->iteration }}</td>
            <td>{{ $r->roll_no }}</td>
            <td>{{ $r->Name }}</td>
            <td><a href = "{{route('stud.update', ['number' => $r->roll_no])}}">Edit</a></td>
            <td><a href = "{{route('stud.delete', ['number' => $r->roll_no])}}">Delete</a></td>
          <tr>
        @endforeach
      </table> </body></html>
```

Helper Functions

[action](#)

[asset](#)

[route](#)

[secure_asset](#)

[secure_url](#)

[url](#)

[abort](#)

[abort_if](#)

[abort_unless](#)

[app](#)

[auth](#)

[back](#)

[bcrypt](#)

[blank](#)

[broadcast](#)

[cache](#)

[class_uses_recursive](#)

[collect](#)

[config](#)

[cookie](#)

[csrf_field](#)

[csrf_token](#)

[dd](#)

[decrypt](#)

[dispatch](#)

[dispatch_now](#)

[dump](#)

[encrypt](#)

[env](#)

[event](#)

[factory](#)

[filled](#)

[info](#)

[logger](#)

[method_field](#)

[now](#)

[old](#)

[optional](#)

[policy](#)

[redirect](#)

[report](#)

[request](#)

[rescue](#)

[resolve](#)

[response](#)

[retry](#)

[session](#)

[tap](#)

[throw_if](#)

[throw_unless](#)

[today](#)

[trait_uses_recursive](#)

[transform](#)

[validator](#)

[value](#)

[view](#)

[with](#)

URL Helper Functions

```
Route::get('/news/{id}', 'News@index' ) -> name('newsIndex');
```

route

```
route('newsIndex', ['id' => 1]);
```

```
http://all.deei.fct.ualg.pt/~a12345/LAB9_10/index.php/news/1
```

action

```
action('News@index', ['id' => 1]);
```

```
http://all.deei.fct.ualg.pt/~a12345/LAB9_10/index.php/news/1
```

url

```
url('/');
```

```
http://all.deei.fct.ualg.pt/~a12345/LAB9_10/index.php
```

asset

```
asset('/');
```

```
http://all.deei.fct.ualg.pt/~a12345/LAB9_10/
```


Form validation

CONTROLLER:

```
public function add()
{
    return view('stud_add');
}

public function add_action()
{
    $this->validate(request(), [
        'roll_no'=>'required|numeric|unique:stud,roll_no',
        'name'=>'required'
    ]);

    $roll_no=request('roll_no');
    $name=request('name');
    Stud_model::add($roll_no,$name);
    return redirect('stud');
}
```

Form validation ...

TEMPLATE:

```
<!DOCTYPE html>
<html lang = "en">
  <head>
    <meta charset = "utf-8">
    <title>Students Example</title>
  </head>
  <body>
    @if(count($errors))
      @foreach($errors->all() as $error)
        <h3>{{ $error }}</h3>
      @endforeach
    @endif

    <form method="post" action="{route('stud.add_action')}}">
      {{csrf_field()}}
      <label>Roll No.</label><input type="text" name="roll_no"
value="{{ old('roll_no') }}" />
      <br/><label>Name</label><input type="text" name="name" value="{{
old('name') }}" />
      <br/><input type="submit" value="Add" />
    </form>
  </body>
</html>
```

Sessions

```
public function session_example(Request $request)
```

- Retrieving session data

```
$name = session('item') OR
```

```
$name = $request->session()->get('item');
```

- Assigning session data

```
session(['item' => $name]); OR
```

```
$request->session()->put('item', $name);
```

- Unset session data

```
$request->session()->forget('item')
```

- Destroy session

```
$request->session()->flush();
```

Cookies

```
use Illuminate\Support\Facades\Cookie;
```

- **Assigning cookie data**

```
Cookie::queue($cookie_name, $cookie_value, $cookie_time);
```

OR

```
$response = new \Illuminate\Http\Response(view('blade_template'));
```

```
$response->withCookie(cookie('$cookie_name ', $cookie_value, $cookie_time));
```

```
return $response;
```

- **Retrieving cookie data**

```
$value = Cookie::get('cookie_name');
```

OR

```
public function cookie_example(Request $request)
```

```
$value = $request->cookie('name');
```

Email

CONTROLLER

```
use Mail;

$email=request('email');
$data = array('reset_digest'=>$reset_digest);

Mail::send(['text'=>'email_template'], $data,
    function($message) use ($email)
    {
        $message->to($email)->subject('Password reset');
        $message->from('webmaster@deei.fct.ualg.pt', 'webmaster');
    }
);
```

Email...

TEMPLATE

'email_template.blade.php'

Para obter uma nova password clique no link

[http://all.deei.fct.ualg.pt/~a999990/LV_exame2/blog/new_password/{{ \\$reset_digest }}](http://all.deei.fct.ualg.pt/~a999990/LV_exame2/blog/new_password/{{ $reset_digest }})

Se NÃO pediu uma nova password IGNORE este email.

Cumprimentos,

webmaster!

Página Web: <http://intranet.deei.fct.ualg.pt/DAW/>

E-mail: webmaster@deei.fct.ualg.pt

NOTA: Não responda a este email, não vai obter resposta!