



**IPL**  
escola superior  
de tecnologia e gestão  
instituto politécnico  
de leiria

DEPARTAMENTO DE ENGENHARIA INFORMÁTICA

[HTTP://WWW.DEL.ESTG.IPLEIRIA.PT/](http://www.del.estg.ipleiria.pt/)

---

TESP EM PROGRAMAÇÃO DE SISTEMAS DE INFORMAÇÃO

---

## Plataformas de Sistemas de Informação

---

### Ficha Prática Nº1

---

#### Objetivos:

- Instalar, configurar e compreender as componentes base de uma plataforma de desenvolvimento Web (Parte 1 de 3):
    - *Stack Web* (Servidor Web, Linguagem de Scripting e SGBD)
    - IDE
    - Cliente de Gestão de Bases de Dados (opcional)
    - Gestor de dependências
    - Framework Web
  - Automatizar instalações e configurações via um gestor de dependências.
  - Analisar e reconhecer as principais componentes MVC da *framework* Yii2.\*.
- 

1. Instalação da *Stack Web*. Cada estudante poderá optar entre uma das seguintes opções:
  - i. LAMP, WAMP ou MAMP (dependendo do SO)
    - a. Cada estudante deverá aceder a página web da respetiva *Stack* e seguir as instruções de instalação.
    - b. **Atenção!** Na escolha da *Stack* o estudante deverá ter em conta a arquitetura da sua plataforma (32 ou 64 bits).

- c. Após instalação, esta deverá ser verificada acedendo ao endereço *localhost*, através de um *browser* web. Este acesso irá aceder à *Root Web* que é mapeada pelo servidor HTTP (por exemplo diretoria *www*, *htdocs*, entre outras possibilidades, dependendo da *stack* instalada).
- d. **Troubleshooting!** Caso a *Root Web* se encontre vazia, o estudante deverá criar um ficheiro com o nome *index.php* e dentro do mesmo deverá chamar a função *phpinfo()*.
- e. Deverá voltar a testar a instalação da *stack*, de acordo com as instruções na alínea c).

## 2. Instalação do IDE.

- a. O IDE de utilização obrigatória nesta UC é o PHPStorm da JetBrains (<https://www.jetbrains.com/phpstorm/>)
- b. Os estudantes devem efetuar o registo no site da JetBrains usando o endereço de *email* da escola, o que lhes permitirá obter uma licença de educação com a duração de 1 ano.

## 3. *Composer* (gestor de dependências para PHP)

- a. Descarregar o instalador do composer (<https://getcomposer.org/>)
- b. Após instalação deverão abrir a consola para linha de comandos (CLI) e escrever o comando *composer*, verificando se o comando é ou não reconhecido na CLI.
- c. Ficheiros importantes do composer (definições):

- i. *composer.json*:

---

---

---

ii. Exemplo de um ficheiro composer.json:

```
{  
    "require": {  
        "vendor/package": "1.3.2",  
        "vendor/package2": "1.*",  
    }  
}
```

iii. Definições das componentes do composer.json:

1. Keyword *require*:

---

---

---

2. Package (vendor/package):

---

---

---

3. Versão após a package:

---

---

---

iv. composer.lock:

---

---

---

v. composer.phar:

---

---

---

- d. Comandos mais usuais no composer:
  - i. `composer update`
  - ii. `composer update <package>`
  - iii. `composer create-project <package> <nome da diretorio>`

4. Instalação da framework Yii2.x usando o composer

- a. Ir à página <http://www.yiiframework.com/>
- b. Navegar para Guide2.0 → Getting Started → Installing Yii
- c. Ir para a parte da documentação que descreve o processo de instalação da framework Yii através do *composer*.
- d. O que é o *Bower* e o *npm*?:

---

---

---

---

- e. Crie um projeto Yii2 baseado no *template* básico. Este comando deverá ser executado pela CLI, na diretorio raiz de acesso web do servidor HTTP (por exemplo; WWW, htdocs, etc...).

Nota: Antes de proceder à execução do comando deverá criar uma conta no GitHub e gerar um token de utilização (personal access token).

```
composer create-project --prefer-dist yiisoft/yii2-app-basic basic
```

f. Explicação do comando (e respetivos parâmetros) da alínea anterior:

---

---

---

---

g. Por que motivo devo criar uma conta GitHub e respetivo token?

---

---

---

---

h. Teste o projeto criado, usando um *browser*, com o seguinte URL

<http://localhost/basic/web>.

i. Analise e explique o Profiler/Tracer/Debugger da Yii que se encontra no canto inferior direito do ecrã inicial (homepage) da aplicação.

i. Request:\_\_\_\_\_

1. Route:\_\_\_\_\_

2. Mapeamento da Route\_\_\_\_\_

---

3. Outros:\_\_\_\_\_

---

---

---

ii. Logs:\_\_\_\_\_

---

---

iii. Profiling:\_\_\_\_\_

---

---

---

iv. Database:\_\_\_\_\_

---

\_\_\_\_\_

\_\_\_\_\_

v. Asset

Bundles: \_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

vi. Mail: \_\_\_\_\_

5. Análise e adapte a aplicação criada.

a. Localize a pasta das vistas no projeto criado:

\_\_\_\_\_

b. Altere a vista 'About' e introduza 3 parágrafos, cada um com o seu título. Recorra a elementos CSS *Bootstrap* (que já se encontra integrado na aplicação).

c. Localize a pasta dos controladores no projeto criado:

\_\_\_\_\_

d. A aplicação dispõe de apenas 1 Controlador. Como se chama a classe e de onde descende?

\_\_\_\_\_

e. Neste tipo de projeto, o que representa o controlador?

\_\_\_\_\_

f. Enumere cada um dos métodos do controlador e explique a funcionalidade de cada um: \_\_\_\_\_

---

---

---

---

---

---

---

g. Localize a pasta dos modelos no projeto:

---

---

h. Explique cada um dos modelos gerados:

---

---

---

---

---

---

---

---

---

---

---

---