

## Desarrollo de Aplicaciones I – FIUBA

### Ejercicio 1

1) Crear el archivo prueba.php y hostearlo mediante el servidor web built-in de PHP \* ejecutando:

```
php -S localhost:8080
```

\*En el caso de utilizar el container de Docker, copiar el script “serve\_php\_app.sh” junto con prueba.php y ejecutarlo con:

```
./serve_php_app.sh . 8080
```

2) Agregar en el archivo prueba.php la impresión de un mensaje utilizando “echo()”.

3) Abrir el navegador en localhost:8080/prueba.php y verificar la ejecución del script.

### Ejercicio 2

1) Definir la clase “User” con los atributos privados “id” (numérico) “name”, “email” (textos) y “isLogged” (booleano). El constructor de la clase debe recibir el ID, el email y el nombre.

2) Agregar el método “printInfo” el cual imprime los tres atributos.

3) Definir un array de 3 objetos “User”. Luego iterar el array y ejecutar el método printInfo() de cada objeto.

### Ejercicio 3

1) Realizar un servicio web “usuario.php” que reciba el argumento por GET “id”. Utilizar la clase del ejercicio anterior para crear un objeto “User”. Cargarle el id recibido por GET y el resto de los atributos con valores fijos.

2) Agregar a la clase el método “getJSONInfo()” el cual devuelva un string en formato JSON con el formato:

```
{"id":39,"email":"juanperez@gmail.com","name":"Juan","isLogged":true}
```

NOTA 1: Utilizar la función “json\_encode()” la cual recibe un diccionario.

NOTA 2: Utilizar la variable global \$\_GET para obtener los parámetros de la URL.

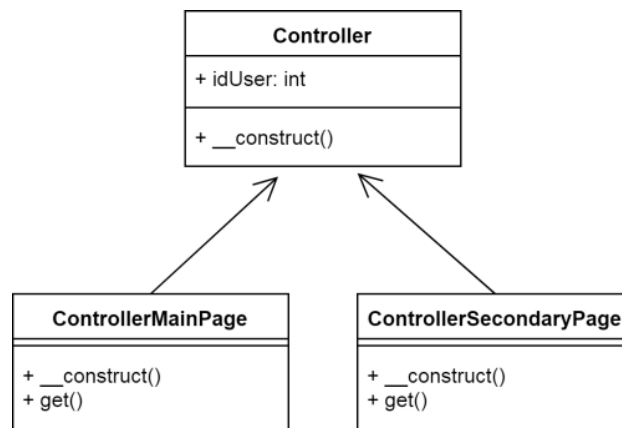
3) Probar el servicio web con la url: “localhost:8080/usuario.php?id=39”

#### Ejercicio 4

1) Crear el archivo index.php el cual se encargará de “rutear” el request GET según el valor de la variable “page” que viene como parte de la URL.

Ejemplo URL: localhost/index.php?page=0&idUser=39

2) Se deberán crear 3 clases según el siguiente diagrama de clases:



El “ruteo” se encargará de evaluar el contenido del campo “page” en la url (se obtiene leyendo la variable global `$_GET`), y según este valor (cero o uno) instanciará un objeto de la clase **ControllerMainPage** o **ControllerSecondaryPage** respectivamente y luego llamará al método “get”.

3) Dentro del constructor de la clase **Controller**, se guardará el campo “idUser” de la url en el atributo “idUser” leyéndolo de la variable global `$_GET`.

4) Dentro de los método “get” de cada clase, imprimir un mensaje diferente junto con el id del usuario que vino como parámetro en la URL.