

LER O ENUNCIADO ATÉ AO FIM ANTES DE COMEÇAR!

O objectivo do exame é adicionar uma nova funcionalidade à aplicação web que foi construída ao longo dos laboratórios da disciplina.

É dado acesso a uma implementação do site web com as funcionalidades correspondentes ao LAB 6. A base de dados já se encontra implementada (informação sobre a base de dados encontra-se em ANEXO).

Apenas se solicita ao aluno a alteração dos templates e o código controlador PHP necessário para implementar a nova funcionalidade.

PRELIMINARES

A. Faça login por ssh (com o PuTTY, por exemplo) no servidor com o IP 10.10.23.183

```
a12345@daw:~$
```

B. Faça download para a pasta “exame” já existente, do código do site web arquivado em “Github” com o comando “git”, e ponha as permissões correctas.

```
a12345@daw:~ $  
git clone https://github.com/jmatbastos/exame15.01.2018.git  
~/public_html/exame
```

```
a12345@daw:~$ chmod g+r ~/public_html/exame/*
```

C. (OPCIONAL). Se preferir trabalhar inicialmente no seu portátil pode fazer o download do código que se encontra numa pasta “zip” no URL

```
https://github.com/jmatbastos/exame15.01.2018/archive/master.zip
```

FUNCIONALIDADE “PRIVATE MESSAGE”

Pretende-se implementar a funcionalidade “private message” que permite a utilizadores registados no site trocarem mensagens privadas entre si

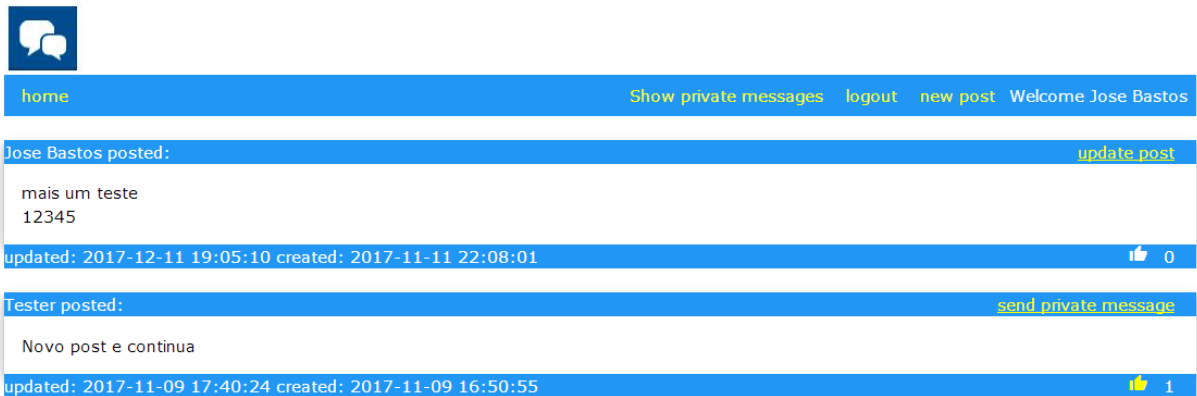
PARTE A – ACTUALIZAÇÃO DA PÁGINA DE ROSTO [6 valores]

1. Actualize o template `index_template.html`

- se o utilizador fez *log in*, para ter um “placeholder” adicional no menu para o *hyperlink*

```
<a href="showprivatemessages.php">Show private messages</a>
```

- se o utilizador é anónimo, o “placeholder” está vazio



2. Altere o ficheiro `index.php` para:

- A. actualizar o “placeholder” da pergunta 1
- B. actualizar o “placeholder” `{UPDATE_PM}`
 - se o utilizador que fez login é o autor do post, com o hyperlink

```
<a href="updateblog.php?micropost_id=' . $tuple['id'] . '">update post</a>
```

se o utilizador que fez login não é o autor do post, com o hyperlink

```
<a href="privatemessage.php?user_id=' . $tuple['user_id'] . '">send private message</a>
```

PARTE B – ENVIO DA PRIVATE MESSAGE [8 valores]

3. Construa o template “privatemessage_template.html”

SUGESTÃO: Copie e adapte o template “blog_template.html”



© 2016 Desenvolvimento de Aplicações Web Designed by [Aluno](#)

O formulário utiliza o método “post” e a acção é

```
method="post" action="privatemessage_action.php"
```

4. Construa o controlador do template “privatemessage.php”

SUGESTÃO: Copie e adapte o controlador “blog.php”

A passagem da variável “user_id” para o controlador da acção pode ser feito pelo método GET no URL da acção ou (recomendado) através de um input do tipo “hidden” no formulário

5. Construa o controlador `privatemessage_action.php`

que actualiza a tabela “messages” com a nova mensagem privada

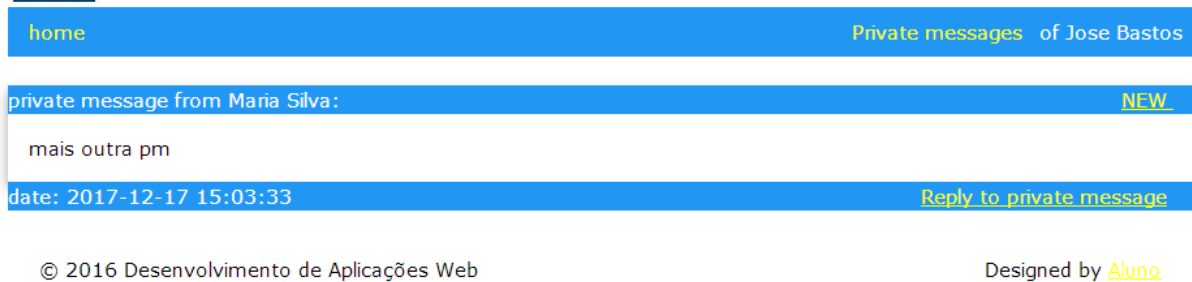
NOTA: o status da mensagem deve ser “0” (não lida)

SUGESTÃO: Copie e adapte o controlador “blog_action.php”

PARTE C – LEITURA DA PRIVATE MESSAGE [8 valores]

6. Construa o controlador `showprivatemessages.php`

Este controlador trabalha com o template `showprivatemessages_template.html` (o template é dado já completo)



que tem um hyperlink

```
<a href="privatemessage.php?user_id=' . $tuple['from_user_id'] . '">Reply to private message</a>
```

que permite o utilizador responder às mensagens privadas

e outro hyperlink

```
<a class="w3-tooltip" href="markread_action.php?m_id=' . $tuple['m_id'] . '">NEW <span class="w3-text">(<em>click to mark read</em></span></a>
```

que permite “marcar” uma mensagem como “lida”

7. Construa o controlador `markread_action.php` que actualiza o status da mensagem correspondente na tabela “messages” como “lida” (`status='1'`)

NOTAS:

- Coloque o seu código entre as linhas “PRINCIPIO DO CODIGO” e “FIM DO CODIGO” (o seu código pode ter qualquer numero de linhas; adicione as linhas que precisar)
- Caso considere absolutamente necessário escrever código noutra local, comente o seu código e justifique-o num ficheiro “README.TXT”. Pode utilizar esse ficheiro também para outra informação que julgue pertinente.

- Caso tenha trabalhado no seu portatil, **é obrigatório fazer o upload de todos os ficheiros** para a pasta “exame” no seu site web pessoal

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

(onde 12345 é o seu número de aluno). Utilize scp (Linux) ou WinSCP (Windows) para fazer a cópia. **Verifique que o site fica operacional** (com as permissões necessárias*).

- SUGESTÃO: caso tenha problemas como seu código e o browser apresente uma página web em branco, lembre-se que pode consultar o log do servidor web com o comando

```
a12345@daw:~$ tail -f /var/log/apache2/error.log
```

* a12345@daw:~\$ chmod g+r ~/public_html/exame/*

ANEXO 1 Acesso à base de dados MySQL

- O acesso à base de dados MySQL pode ser feita utilizando um cliente gráfico à sua escolha (por exemplo <http://www.heidisql.com/>),

ou em linha de comando

```
a12345@daw:~$mysql -u a999999 -p -h 10.10.23.183 db_a999999
```

ou ainda utilizando o software **phpMyAdmin** disponível no URL

<http://all.deei.fct.ualg.pt/phpMyAdmin>

ANEXO 2 : estrutura da base de dados

```
CREATE TABLE `users` (  
  `id` int(11) NOT NULL auto_increment,  
  `name` varchar(255) default NULL,  
  `email` varchar(255) default NULL,  
  `created_at` datetime NOT NULL,  
  `updated_at` datetime NOT NULL,  
  `password_digest` varchar(255) default NULL,  
  `remember_digest` varchar(255) default NULL,  
  `admin` tinyint(1) default NULL,  
  `activation_digest` varchar(255) default NULL,  
  `activated` tinyint(1) default NULL,  
  `activated_at` datetime default NULL,  
  `reset_digest` varchar(255) default NULL,  
  `reset_sent_at` datetime default NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `index_users_on_email` (`email`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
  
CREATE TABLE `microposts` (  
  `id` int(11) NOT NULL auto_increment,  
  `content` text,  
  `user_id` int(11) default NULL,  
  `created_at` datetime NOT NULL,  
  `updated_at` datetime NOT NULL,  
  `likes` int(11) NOT NULL DEFAULT '0',  
  PRIMARY KEY (`id`),  
  KEY (`user_id`),  
  CONSTRAINT FOREIGN KEY (`user_id`) REFERENCES `users` (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;  
  
CREATE TABLE `messages` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `content` text,  
  `from_user_id` int(11) NOT NULL,  
  `to_user_id` int(11) NOT NULL,  
  `created_at` datetime NOT NULL,  
  `status` int(1) DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  KEY (`from_user_id`),  
  KEY (`to_user_id`),  
  CONSTRAINT FOREIGN KEY (`from_user_id`) REFERENCES `users` (`id`),  
  CONSTRAINT FOREIGN KEY (`to_user_id`) REFERENCES `users` (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8
```