

PW EXAM - JULIOL - CAT

Enunciat de l'examen de recuperació de Projectes Web II.

Requeriments

Per poder implementar aquest exercici, cal disposar d'un entorn de desenvolupament que contingui:

1. PHP7
2. Composer
3. MySQL
4. Servidor Web (Apache o Nginx)

Introducció

En aquest exercici es demana d'implementar una aplicació web que permeti als usuaris de gestionar els seus contactes de tlf. L'aplicació estarà formada per les seccions següents:

1. Landing page
2. Registre
3. Login
4. Llistat de contactes
5. Afegir contactes

Landing page (0.5 punts)

Ruta	Mètode
/	GET

En aquest pàgina únicament es mostra la *landing page* de l'aplicació. A més a més, als usuaris que no han iniciat sessió, se'ls mostra un parell d'enllaços:

1. Registre: ha de ser un *link* a la pàgina de registre `/register`.
2. Login: ha de ser un *link* a la pàgina de `login`.

Un cop l'usuari ha iniciat la sessió, cal amagar ambdós enllaços. Per fer-ho, únicament cal que passeu una variable **logged** als templates que sigui *true* quan l'usuari tingui la sessió iniciada.

Registre (2 punts)

Ruta	Mètode
/register	GET
/register	POST

Quan els usuaris accedeixen a la URL `/register` han de veure un formulari amb els següents camps:

1. email
 - Ha de tenir un format de correu vàlid
2. password
 - Ha de contenir 6 o més caràcters

Un cop l'usuari faci click al botó de *submit* el formulari s'haurà d'enviar a `/register` fent servir el mètode **POST**. Tots els camps hauran de ser validats al Controlador abans de registrar l'usuari.

Si hi ha cap error, s'haurà de mostrar novament el formulari marcant en cada camp els errors que s'han trobat durant la validació.

Nota: No cal implementar cap tipus de validació fent servir Javascript.

Cal fer servir aquest codi SQL per generar la taula d'usuaris:

```
CREATE TABLE `user` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `email` varchar(100) NOT NULL,
  `password` varchar(12) NOT NULL,
  `created_at` datetime DEFAULT NULL,
  `updated_at` datetime DEFAULT NULL,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Login (1 punts)

Ruta	Mètode
/login	GET
/login	POST

Quan un usuari accedeixi a la pàgina `/login`, haurà de veure un formulari amb els camps següents:

1. email
2. password

Un cop l'usuari faci click en el butó de *submit* el formulari s'haurà d'enviar a `/login` fent servir un mètode **POST**. Tots els camps han de ser validats en el Controlador abans d'iniciar la sessió de l'usuari. Si hi ha cap error, s'haurà de mostrar el formulari tot indicant els errors que s'han produït.

Llistat de contactes (2 punts)

Ruta	Mètode
/contacts	GET

Tots els usuaris que intentin accedir a aquesta pàgina sense haver iniciat sessió, hauran de ser redirigits cap al formulari de `login`. En cas que l'usuari tingui la sessió iniciada, heu de mostrar un llistat amb tots els contactes de l'usuari. Per cada contacte, cal mostrar:

1. El nom i cognom
2. El número de tlf
3. Un *link* per esborrar el contacte (aquest apartat es descriu més endavant)

Afegir nous contactes (3 punt)

Route	Method
/contacts/add	GET
/contacts/add	POST

Tots els usuaris que intentin accedir a aquesta pàgina sense haver iniciat sessió, hauran de ser redirigits cap al formulari de `login`. En cas contrari, cal mostrar un formulari que contingui els camps següents:

1. Nom
 1. Obligatori
2. Cognoms
 1. Opcional
3. Número de tlf
 1. Obligatori
 2. Heu de validar que contingui 9 dígit (poden contenir espais)

En cas que hi hagi cap error, heu de tornar a mostrar el formulari tot indicant els errors que s'hagin produït.

Si no hi ha cap error, heu de redirigir a l'usuari al seu llistat de contactes tot mostrant un mssg que indiqui que el contacte s'ha registrar correctament.

Per tal de crear la taula **contact** heu de fer servir el següent *snippet* de SQL:

```
CREATE TABLE `contact` (  
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,  
  `id_user` int(10) unsigned NOT NULL,  
  `name` varchar(20) NOT NULL,  
  `cognoms` varchar(255) NOT NULL,  
  `tlf` varchar(255) NOT NULL,  
  `created_at` datetime DEFAULT NULL,  
  `activity_time` datetime DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  KEY `user_contact` (`id_user`),  
  CONSTRAINT `user_contact` FOREIGN KEY (`id_user`) REFERENCES `user` (`id`) ON DE  
LETE CASCADE ON UPDATE CASCADE  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Eliminar un contacte (1.5 punts)

Ruta	Mètode
/contacts/delete/{id}	GET

Tots els usuaris que intentin accedir a aquesta pàgina sense haver iniciat sessió, hauran de ser redirigits cap al formulari de `login`. Si l'usuari ha iniciat la sessió, caldrà buscar el contacte corresponent i esborrar-lo. Si el contacte s'ha esborrat correctament, cal redirigir a l'usuari al seu llistat de contactes tot mostrat un mssg que indiqui que l'acció s'ha processat correctament.

Lliurament de l'entrega

- Disposeu de 2h i 30min per implementar l'exercici
- Podeu consultar i fer servir qualsevol recurs
- No es pot modificar l'estructura de les taules que es faciliten en l'enunciat
- Cal fer servir Slim per implementar l'exercici
- Al finalitzar, cal penjar un **.zip** que contingui el codi, esborrant la carpeta *vendor*, a l'Estudy*
- Qualsevol comentari que creieu que és important de cara a la correcció del vostre examen, el podeu afegir en un document README en el mateix ZIP de l'entrega