**Pràctica 5**

**Control de sessions i usuaris registrats (part 2)**

Índex de continguts

[Recuperació de contrasenya i Autenticació Social 2](#_Toc149304078)

[Sobre el codi 2](#_Toc149304079)

[Requeriments per a ser avaluada 3](#_Toc149304080)

[Recomanacions 4](#_Toc149304081)

[Lliurament 5](#_Toc149304082)

### Recuperació de contrasenya i Autenticació Social

* S’ha de donar l’opció de recuperar la contrasenya enviant un correu electrònic que contindrà un enllaç mitjançant token. [2/10]
* En cas de introduir 3 vegades incorrectament la contrasenya, se li demanarà a l’usuari validar-se abans amb reCaptcha. [2/10]

https://developers.google.com/recaptcha/docs/versions

* Una vegada validat, s’ha de mantenir la sessió de l’usuari almenys 25min activa ($\_SESSION).
* Autenticació social
  + Realitzar l’autenticació de l’usuari mitjançant OAuth2 i fent servir un Token o la llibreria PECL si escau. Justifiqueu en cada cas què feu. [3/10]
  + Realitzar l’autenticació de l’usuari mitjançant la classe HybridAuth. [3/10]

### Sobre el codi

* + Treballeu amb funcions
  + Genereu documentació creada amb PHPDcoumentor
  + Les funcions comunes com ara per exemple existeixUsuariBDD() haurien d’estar en un fitxer que es digui controlador.php (p.ex) i que contingui tota la part del controlador comuna al projecte.
  + En aquesta pràctica hauria d’aparèixer els conceptes de les pràctiques anteriors:
    - Creació de formularis
    - Control sessions, cookies, PDO,...
    - Paginació
    - Inici sessions, login, usuaris registrats. En aquest apartat haureu de desenvolupar altres funcionalitats com ara:
      * Donar l’opció a l’usuari de recuperar la contrasenya o gestionar la pèrdua /oblit de la contrasenya.
      * Evitar contrasenyes febles
      * Poder canviar la contrasenya
      * Poder canviar les dades personals
  + Creeu un fitxer php (per exemple errors.php) que gestionarà els errors que es generen a la web.
  + Apliqueu disseny modular sobre el vostre codi. Comenteu-lo i feu servir PREcondicions i POSTcondicions si escau.
  + La crida a la BDD s’ha de fer així:

$nom\_variable\_connexio= ....dbname=pt05\_nom\_cognom', 'root', '');

### Requeriments per a ser avaluada

* A cadascú dels fitxers (php, html, ...) poseu el vostre nom a la capçalera
* S’ha de treballar amb funcions tenint en compte sempre els principis del Refactoring
* No doneu per entès el que feu i sigueu curosos definint que feu en cada cas.
* Cal treballar amb PHPDocumentor
* En cas de crear una BDD, has de lliurar l’esquema de la mateixa amb el següent format: ***pt05\_Nom\_Cognom.sql***
* *La crida a la BDD ha de ser així:*

*$nom\_variable\_connexio= ....dbname=pt05\_nom\_cognom', 'root', '');*

*CREATE DATABASE IF NOT EXISTS `pt05\_Nom\_Cognom`;*

*USE `pt04\_Nom\_Cognom`;*

o fer això:

*DROP DATABASE IF EXISTS pt05\_Nom\_Cognom;*

### Recomanacions

* Crear un fitxer on posareu la lògica i que comprovarà que l’usuari tingui una sessió activa, en cas de no ser així, enviarà a l’usuari a iniciar sessió o que es registri. Si l’usuari te una sessió, carregarem tot el contingut a un fitxer que vindria a ser la vista (...view.php). Aquest control sobre si la sessió està “setejada” o no es recomana fer sobre el fitxer index.php.
* Els fitxers o fitxer de la vista s’hauria de guardar dins d’una carpeta de nom view o vista (p.ex)
* Creeu una carpeta on guardeu el fitxer dels estils.
* Escolliu els estils que creieu convenient. Una web interessant d’on descarregar les icones podria ser Font Awesome. Si no us funciona amb la versió 5 de Font Awesome, comproveu amb aquesta versió:
  + <https://fontawesome.com/v4.7.0/>
* El formulari per a enregistrar-se recomano fer-ho sobre un fitxer tipus registrarse.view.php i registrarse.php que vindria a ser la lògica i des d’on es crida al view.
* El formulari per a logar-se recomano fer-ho sobre un fitxer tipus login.view.php i login.php que vindria a ser la lògica i des d’on es crida als fitxers view.
* Al fitxer login.php hauríeu de controlar si la sessió ja està activa i evitar que l’usuari pugui anar a la pagina de login o de donar-se d’alta. És a dir, una vegada que l’usuari inicia sessió, NO ha de poder accedir a aquests formularis. És de sentit comú...
* Crear una BDD per a enregistrar els usuaris. Aquesta BDD podria constar de:
  + Taula Usuaris (id, usuari, contrasenya)
* Al fitxer registrarse.php hauríeu de controlar si la sessió ja està activa i evitar que l’usuari pugui anar a la pagina de login o de donar-se d’alta. És a dir, una vegada que l’usuari inicia sessió, NO ha de poder accedir a aquests formularis. És de sentit comú...
* Per a validar les dades del formulari de registre, comprovarem:
  + Camps NO buits
  + Contrasenyes idèntiques
* Una vegada iniciada la sessió, podrem entrar al contingut (podríeu fer servir contingut.php p.ex) i a més aquest contingut s’ha de protegir per a que no es pugui entrar directament des de la URL. Com fer-ho? Doncs per exemple comprovant si la sessió està “setejada” i si és així, fem un require de la vista del contingut (contingut.view.php).
* Per a “encriptar” la contrasenya podem fer un hash sobre la contrasenya de la següent manera:
  + $contrasenya = hash(*tipus\_de\_hash,variable\_a\_encriptar*)
  + $contrasenya = hash(‘sha512’,$contrasenya)
* Per a tancar la sessió treballeu amb un nou fitxer php on executarem, entre d’altres coses que creieu convenients, la funció session\_destroy().

### Lliurament

* A cadascú dels fitxers desenvolupats haurà d’aparèixer el vostre nom a la capçalera i el codi comentat amb format PHPDocumentor.
* Adjunteu l’esquema de la vostra BDD.
* Per a qualsevol aclariment, adjunteu un fitxer README
* Al moodle lliureu l’enllaç del vostre Git