**Pràctica 4: Control de sessions i usuaris registrats**

Índex de continguts

[Part bàsica ~~[6/10]~~ 2](#_Toc148196442)

[~~Social Authentication [2,5/10]~~ 3](#_Toc148196443)

[~~Enviar un correu per a recuperar la contrasenya [1,5/10]~~ 3](#_Toc148196444)

[Requeriments per a ser avaluada 4](#_Toc148196445)

[Recomanacions 5](#_Toc148196446)

[Lliurament 6](#_Toc148196447)

### Part bàsica ~~[6/10]~~

* En obrir el web sortiran tots els articles de forma pública sense excepció i sense opció d’inserir, editar o eliminar articles.
* A la mateixa pàgina principal trobarem una opció per a logar-se o enregistrar-se.
* Crear un formulari de inici de sessió i registre d’usuaris en una base de dades i posteriorment verificar quan inicien sessió els usuaris de la base de dades i llavors permetre’ls editar el contingut (els post de la paginació) si no iniciem sessió (usuari anònim) permetrem només veure els articles. Per tant heu de fusionar aquesta pràctica amb la pràctica 3. Mireu com ho faríeu tenint en compte que el tractament de les dades que hi ha a la paginació no serà el mateix si ens validem que si no estem validats.
* S’haurà de comprovar si l’usuari ja existeix en qualsevol dels dos casos (inici sessió o registrar-se) i validar la contrasenya demanant-la dues vegades (registrar-se).
* La contrasenya haurà d’estar encriptada.
* Si l’usuari comet un error a l’hora d’enregistrar-se, s’hauran de mantenir les dades al formulari actual.
* S’ha de donar l’opció de recuperar la contrasenya des del mateix formulari.
* ~~En cas de introduir 3 vegades incorrectament la contrasenya, se li demanarà a l’usuari validar-se abans amb reCaptcha.~~

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

* Una vegada validat, s’ha de mantenir la sessió de l’usuari almenys 30min activa ($\_SESSION).
* En cas de desenvolupar aquesta opció, caldrà comentar el codi de l’opció que es demana a l’enunciat bàsic, és a dir, per a qualsevol opció realitzada, no elimineu la part bàsica demanada a l’enunciat.
* 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

### ~~Social Authentication [2,5/10]~~

~~[1/10]: OAuth2 (Open Authorization)~~

~~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.~~

~~[1,5/10]: HybridAuth (Hybrid Authentication)~~

~~Realitzar l’autenticació de l’usuari mitjançant la classe HybridAuth.~~

### ~~­Enviar un correu per a recuperar la contrasenya [1,5/10]~~

* ~~Implementeu l’opció per a poder recuperar la contrasenya mitjançant un correu electrònic.~~

### Requeriments per a ser avaluada

* Cal treballar amb Git, deixant clar tots els commits que es realitzen.
* Cal adjuntar un README o MD.
* 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: ***pt04\_Nom\_Cognom.sql***
* *La crida a la BDD ha de ser així:*

*$nom\_variable\_connexio= ....dbname=pt04\_nom\_cognom', 'root', '');*

*CREATE DATABASE IF NOT EXISTS `pt04\_Nom\_Cognom`;*

*USE `pt04\_Nom\_Cognom`;*

o fer això:

*DROP DATABASE IF EXISTS pt04\_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. No s’admetrà esquemes que no funcionin.
* Al moodle lliureu l’enllaç del vostre Git