

Azken Lan Praktika

Web-Sistemak



Pablo Ruiz

Laura Marquinez

AURKIBIDEA:

1. Sarrera……………………………………………………………………………………………………………2
2. GitHub biltegiko URLa………………………………………………………………………………….2
3. Aplikazioaren URLa 000webhost-en………………………………………………………...2
4. Eskolaz kanpo taldekideek behar izan d(ut)en denbora garatzeko praktika…………………………………………………………………………………………………………2
5. Garapenaren diren eta zuzen funtzionatzen duten funtzionalitateak….3
   1. Pasahitz kodetua duen erregistro zuzena……………………………………
   2. Pasahitza berrezarri………………………………………………………………………
   3. Kautotutako erabiltzailearen identifikazioa hau irisgarri den orri guztietan………………………………………………………………………………………
   4. Norberaren galdera kopurua eta guztira dauden galderen kopuru bien konputazioa modu asinkronoan gauzatu……………………………

## **WEB-SISTEMAK**

## Azken Lan Praktikoa

## SARRERA

Azken lan praktiko honetan, irakasgaiko laborategietako lorpenetatik abiatuz, gauzatu ez genituen hautazko atazak garatu ditugu.

Gure kasuan, eskatzen ziren oinarrizko funtzionalitateetatik, pasahitz zifratua, pasahitza berrezarri, kautotasuna $\_SESSION objetua erabiliz eta norberaren galdera kopurua zein guztira dauden galderen kopurua gauzatzea sistemara integratu behar izan ditugu.

## 2. GitHub BILTEGIKO URLa

## 3. APLIKAZIOAREN URLa 000webHost

## 

## 

## 4. ESKOLAZ KANPO TALDEKIDEEK BEHAR IZAN D(UT)EN DENBORA GARATZEKO PRAKTIKA

|  |  |
| --- | --- |
| Pasahitz kodetua duen erregistro zuzena: | 3h |
| Pasahitza berrezarri: | 10h |
| Kautotutako erabiltzailearen identifikazioa hau irisgarri den orri guztietan: | 5h |
| Norberaren galdera kopurua eta guztira dauden galderen kopuru bien konputazioa modu asinkronoan gauzatu: | 6h |

## 5. GARAPENAREN DIREN ETA ZUZEN FUNTZIONATZEN DUTEN FUNTZIONALITATEAK

## 

5.1 Pasahitz kodetua duen erregistro zuzena

Ikasleen erregistroan, pasahitza zifratua metatzea eskatzen da, baita pasahitza baliozkoa dela eta Web Sisteman matrikulaturik dagoela egiaztatzea. Kasu honetan pasahitza zifratzearen atala egin dugu, bestea dagoeneko egina genuen.

Pasahitza enkriptatzeko, erregistratzerakoan DB-n pasahitza gorde aurretik *password\_hash()* funtzioa erabiliko dugu zifratzeko.

$newhash = password\_hash($pasahitza, PASSWORD\_BCRYPT);

Aldiz, logeatzerakoan *password\_verify()* funtzioa erabiliko dugu DB-n dagoen pasahitza zifratuaren berdina den egiaztatzeko.

$hash = $user['pasahitza'];

password\_verify($pasahitza, $hash)

5.2 Pasahitza berrezarri

Pasahitzaren berrezarpena, erabiltzaile erregistratuak pasahitza gogoratuko ez balu eta aldatu nahiko balu, pasahitza eguneratzeko aukera emango dio sistemak. Horretarako, demoa jarraitu dugu.

*berrezarriPasahitza.php* sortuko dugu (layout logeatu gabekoen antzekoa) formulario batekin eposta sartzeko.

Hasteko pasahitzaBerrezarri.php fitxategia sortu eta erabiltzaile anonimoentzako layout.php-n gehitu egin dugu. pasahitzaBerrezarri.php orrian email posta sartzeko formulario bat sortu dugu eta AJAX-en bidez pasBerr.php-ri dei egin diogu.

pasBerr.php fitxategian bidaliko den email-aren egitura eta sorrera dago, baita mail() funtzioaren deia.

Beraz, behin AJAX deia amaituta, posta bidalita egongo da, kode eta link batekin.

Eposta heltzen denean pasahitzaBerrezkuratu.php izeneko orrialde batera bidaliko zaio eta han pasahitza berria eta jasotako kodea sartu beharko dira. Kodea egokia bada AJAX bidez pasAldatu.php-ra dei egingo zaio eta han pasahitza aldatu egingo da, eta han pasahitza aldatu egingo da, enkriptatuta sartuz.

5.3 Kautotutako erabiltzailearen identifikazioa hau irisgarri den orri guztietan

$\_SESSION objektua erabiliz, orrialde guztietan identifikatu da nor dagoen aplikazioan nabigatzen emaila bidez. Goikaldean honako hau jarri beharko dugu:

<?php session\_start(); ?>

Erabiltzaileak identifikatzeko id-a erabiliz egingo dugu.

$\_SESSION['id']

Bestetik bi rol izango ditugu, ikaslea eta administratzailea. Hauen arteko desberdinatasuna, ataza orokorrez gain, ikaslea *handlingQuizAJAX* eskura izango du. Aldiz, administratzailea *handilingAccounts*.

$\_SESSION['rola'] = "admin"

$\_SESSION['rola'] = "ikaslea"

Ikasle batek ezingo du handlingAccounts-n sartu. Horretarako,

if(!isset($\_SESSION['rola']))

{

echo '<script> javascript:history.go(1); </script>';

die();

}

else if($\_SESSION['rola'] != "admin")

{

echo '<script> javascript:history.go(-1); </script>';

die();

}

5.4 Norberaren galdera kopurua eta guztira dauden galderen kopuru bien konputazioa modu asinkronoan gauzatu

* Ikasle erabiltzailea *HandlingQuizes* -n galderak sortu eta ikusiko ditu. Kautotuko erabiltzaileak, modu asinkronoan (AJAX erabiliz) zenbat galdera sartu dituen Quiz-en eta zenbat galdera guztira dituen sistemak xmlean (edo datu-basean) erakutsiko du.

*Lehenengo hautazkoa egiteko AJAX eskuratzen dituen funtzionamendua erabili dugu. Hasteko, datu basetik behar diren bi balioak lortzeko behar den php kodea bi fitxategi berrietan sartu ditugu (display.php eta display2.php). Behin prestaturik galderakZenbatu() izeneko funtzio bat sortu eta bi AJAX sartu ditugu barnean, bakoitza dagokion .php fitxategiari dei egingo diona. Azkenik, html kodea barnean balioak idatzi daitezen .html() erabili egin dugu. 20segunduro berriz kalkulatu dezan, setInterval() funtzioa gehitu dugu, laborategiaren enuntziatuan azaltzen zen bezala*

* Funtzionalitatea (adibidez, Galdera kopuruen kontaketa AJAZ bidez, pasahitzaren berrezarpena)
* Inplementazioaren deskribapena (adibidez, pasahitza berrezarpena epostaz ez da lortu, aldiz ...
* handlingAccounts-en erabiltzailearen egoera permutazioa ez da ajax bidez egin eta hortaz freskatzeko egin behar da .... ....