

2018

# IdentityEYE

PROJECTE M12  
ELOI ROCA PLANA

INS MOLLERUSSA

## Índex

|   |    |
|---|----|
| 1. Introducció .....                          | 2  |
| 2. Objectius.....                             | 3  |
| 3. Disseny del projecte .....                 | 4  |
| 3.1 Disseny Intern .....                      | 4  |
| 3.2 Disseny Extern.....                       | 5  |
| 4. Planning del projecte i organització ..... | 6  |
| 5. Implementació del projecte .....           | 7  |
| 6. Conclusions.....                           | 11 |

## 1. Introducció

Aquest projecte està desenvolupat per Eloi Roca Plana, en aquest document es mostrarà cada una de les parts del projecte, quines funcionalitats tindrà i quines configuracions s'han de seguir per tal de que l'aplicació funcioni d'una manera satisfactòria.

Principalment l'aplicació ens recomanarà un conjunt de peces de roba adients al nostre color de pell, ulls i de pèl.

No hi ha webs que et recomanin quina roba et quedaria bé. Aquesta aplicació cobrirà aquesta necessitat.

Seguint aquest document ens podrem assabentar de quins han sigut els objectius del projecte, el disseny que ha tingut, la organització de la feina desenvolupada en el lloc de desenvolupament i explicació de les d'algunes funcionalitats més complexes de l'aplicació.

## 2. Objectius

- Dissenyar una aplicació la qual segons la tonalitat del teu propi pèl, la dels ulls i la de la pell et pugui recomanar una roba o una altra.
- Disposar d'un perfil propi el qual es editable.
- Buscar altres perfils de l'aplicació.
- Tenir una paleta de colors pròpia, la qual va referenciada amb la teva subestació pertinent.
- Comprar roba que et genera l'aplicació.
- Guardar les combinacions aleatòries de cada usuari
- Consultar novetats de la moda
- Crear pròximes modes i consultar-les.

### 3. Disseny del projecte



#### 3.1 Disseny Intern

IdentityEYE ha estat desenvolupat amb el Framework de PHP Laravel 5.6.

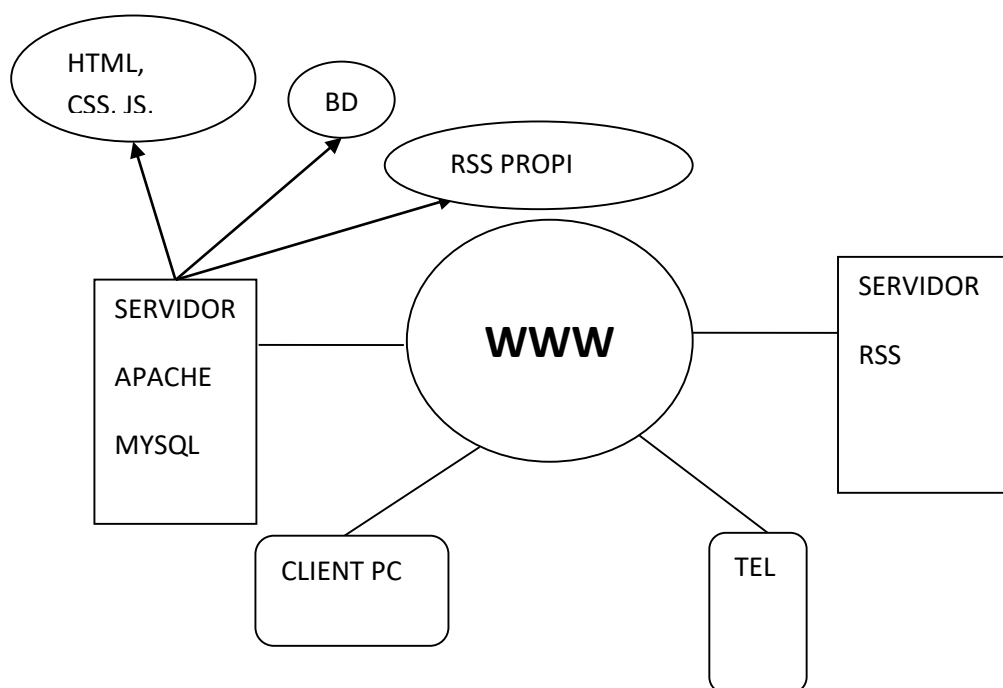
La versió de PHP adient per tal de córrer l'aplicació ha de ser la 7.2.1.

Es pot desplegar damunt de Sistemes Operatius Windows7 i Windows10.

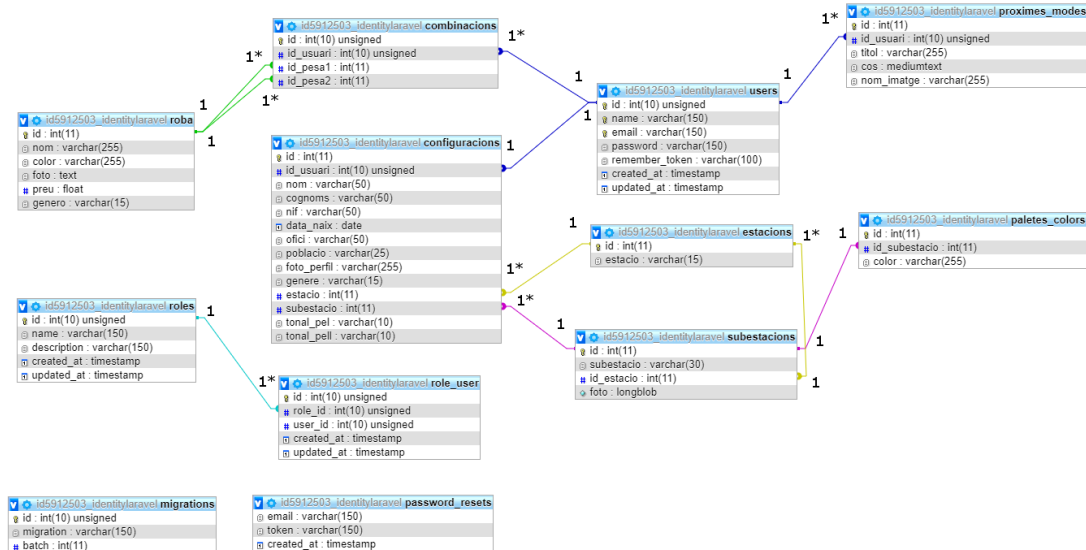
S'ha portat un control de versions amb el GIT de manera que hi haurà versions pujades al [GitHub](#).

IdentityEYE ha estat publicada en un servidor gratuït de [000webhost](#)

El projecte s'ha editat amb el IDE NetBeans i la base de dades ha estat mysql "identitylaravel" amb el servidor de web apache que gestiona el Xampp.



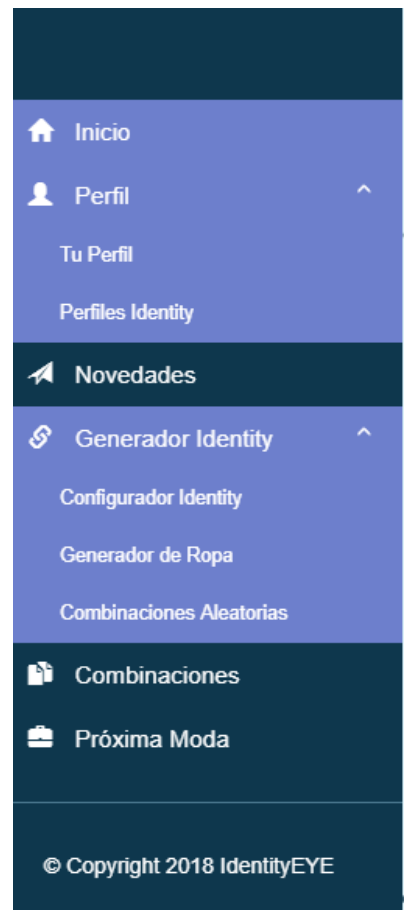
## BASE DE DADES



### 3.2 Disseny Extern

IdentityEYE es basa en els següents apartats per tal de funcionar

- 1r. S'ha de configurar el teu perfil per tal de saber les tonalitats de color
- 2n. Pots visitar els altres perfils, (si ets administrador pots editar i eliminar)
- 3r. Pots veure les novetats en el sector de la moda (RSS Extern)
- 4rt. Pots configurar la teva paleta de color corresponent (configurador)
- 5è. Pots veure i comprar la roba segons la configuració de la paleta anterior
- 6è. Pots guardar una combinació aleatòria que et farà l'aplicació
- 7è. Pots visitar les últimes combinacions guardades
- 8è. Pots veure les pròximes modes que hi hauran sobre la roba (RSS Propi)
- 9è. Si ets administrador pots crear pròximes modes



## 4. Planning del projecte i organització

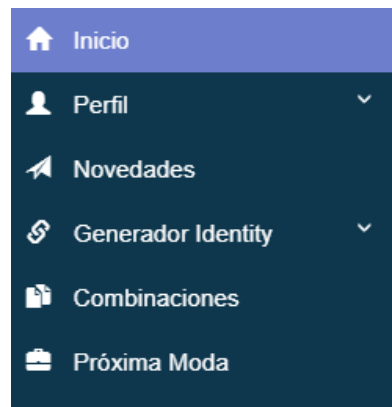
Per tal de desenvolupar aquest projecte han fet falta unes 30h, les quals han estat repartides durant les dos setmanes anteriors.

Primer s'ha desenvolupat la part de la plantilla de l'aplicació on hi ha tres elements bàsics

- a) Una barra per iniciar sessió o registrar-se



- b) Una sidebar amb les opcions de l'aplicació



- c) Un cos de la pàgina anomenat content



## 5. Implementació del projecte

IdentityEYE necessita saber la teva tonalitat de pell i de pel per deixar-te la teva combinació adient. Per tant la primera acció després de registrar-te i iniciar la sessió ha de ser configurar el teu perfil:

Segons la combinació elegida et correspon una subestació determinada.

Tonalidad de la Piel










Fria

Tonalidad del Pelo

Claro

Saber Subestación

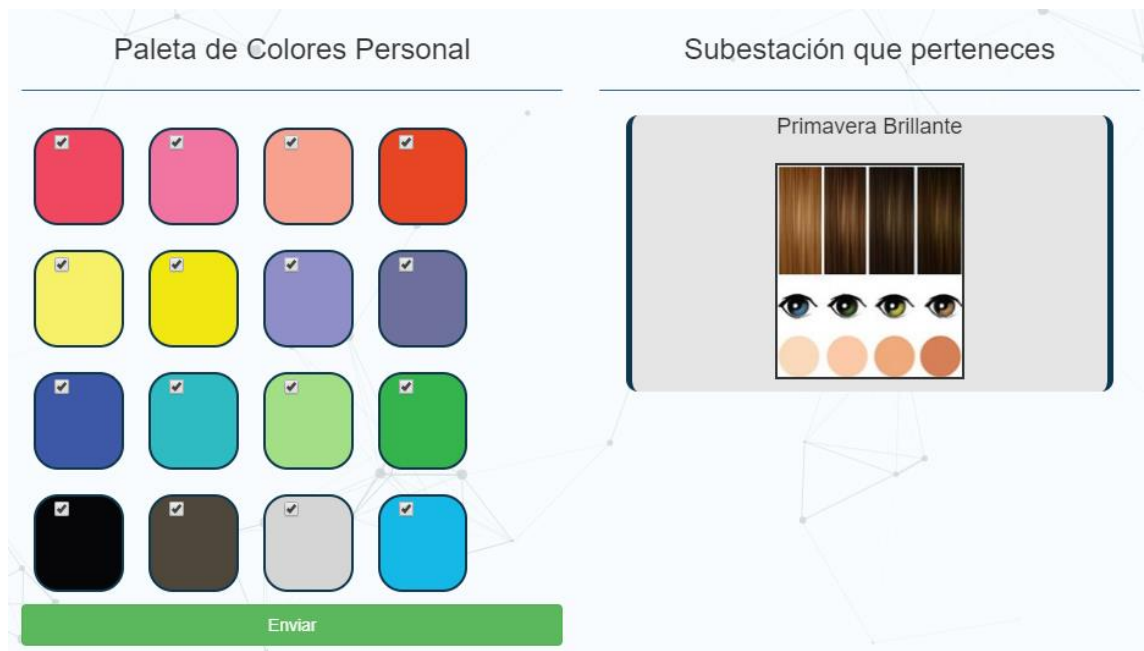
Eres la estación verano! Escoge tu Combinación Perfecta

|   |
|---|
| <br><br> |
| <br><br> |
| <br><br> |

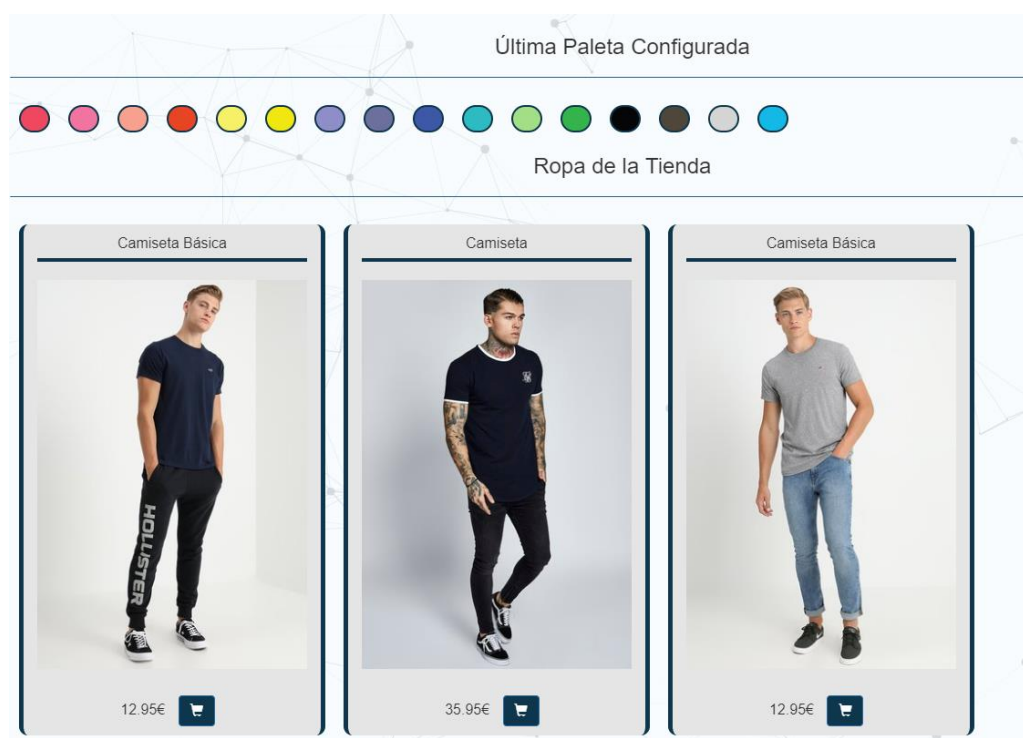


Segons la teva subestació triada en el anterior apartat et corresponent una paleta de colors que et donarà per defecte l'aplicació.

Aquesta paleta serà la que determinarà quina roba mostrar-te. Si un color no et queda bé li pots desactivar a la paleta i no et mostrarà la roba d'aquell color!



Després de configurar-la l'aplicació et mostrarà la roba corresponent



Però com ha arribat fins aquí l'aplicació internament?

Doncs bé mitjançant ajax Identity et mostra un conjunt de combinacions o un altre:

```
function triar_subestacio() {  
    var dades = {  
        "estacio" : estacio  
    };  
    var data = JSON.stringify(dades);  
    $.ajax({  
        data: data,  
        type: 'POST',  
        url: config.rutes[0].ajax_subestacio,  
        cache: false,  
        headers: { 'X-CSRF-TOKEN': $('meta[name="csrf-token"]').attr('content') },  
        datatype: 'JSON',  
        processData: false,  
        success: function (response) {  
            $('#div_subestacio').html(response);  
            $('#div_subestacio').fadeIn(1500);  
            $('#boto_enviar_primer_perfil').fadeIn(1500);  
        },  
        error: function (response) {  
            console.log(response);  
        }  
    });  
}
```

Després d'haver configurat el perfil amb la subestació corresponent el programa ho guarda a la base de dades amb la resta de la teva configuració.

Quan entres en l'apartat de configurar la teva paleta corresponent, Identity recupera la paleta de la teva subestació.

```
//Funcio que retornara la paleta de colors de l'usuari  
public static function obtenir_Colors($id) {  
    $colors = DB::table('paletes_colors')->where('id_subestacio', $id)->get();  
  
    return $colors;  
}
```

Quan el programa obté la paleta la guarda amb una cookie amb el nom de l'usuari registrat, per tal de que no hagi de configurar-la sempre

```
//Funcio que grabara cookie amb els colors del usuari  
public function grabar_cookieColors($colors) {  
    if (isset($_COOKIE['cookie_colors_'.Auth::user()->name])) {  
        setcookie('cookie_colors_'.Auth::user()->name, '', time()-100);  
        setcookie('cookie_colors_'.Auth::user()->name, serialize($colors), time()+3600);  
    } else {  
        setcookie('cookie_colors_'.Auth::user()->name, serialize($colors), time()+3600);  
    }  
}
```

I segons la paleta actual et mostrarà una roba o una altra:

```
//Funcio que retornara la roba amb les dades de la cookie
public static function obtenirRoba($colors, $genero){

    $where = "";
    for($i = 1; $i<count($colors); $i++){
        $where .= " or color = '". $colors[$i]. "'";
    }
    $sql = "select * from roba where (genero = '". $genero. "' AND (color = '". $colors[0]. "'". $where. "));";

    $roba = DB::select($sql, [1]);
    return $roba;
}
```

## 6. Conclusions

Realitzant aquest projecte he arribat a la conclusió de que realment molt poques webs tenen aquesta funcionalitat. He pogut programar totes les combinacions de roba gràcies a [aquesta pàgina](#), la qual m'ha servit de molta ajuda per tal de seguir cada i una de les estacions i subestacions existents.

Avui en dia hi ha moltes revistes que t'aconsellen de la roba que es porta, modes i altres, però no tot et pot quedar bé com les models que surten en elles.

També m'he donat compte que les hores les quals han estat destinades al desenvolupament del projecte han estat satisfactòries. I em sento orgullós del treball realitzat.

Triar el tema del projecte no ha estat una decisió fàcil i s'ha d'agrair l'ajuda de Leny Ares per centrar-lo en aquest món de la moda.