**ANALISIS INTEGRACION PYTHON CON SISTEMAS WEBS**

**Seguridad**

* Menús, accesos, datos de usuario y todo lo referente a esto se creará y obtendrá en Python Django, este será el ambiente principal.
* Autentificación del usuario entre aplicaciones se ingresa desde Python y se genera un token que se leerá en las demás aplicaciones utilizando librerías como JWT.
* Seguridad CORS para poder realizar peticiones javascript desde Python a laravel y viceversa (se crea un archivo por cada aplicación)

Escalabilidad

* Crear contenedor para cada aplicación mediante Docker para futuros despliegues en servidores nuevos.

Server Side Includes

* SSI es una tecnología que permite incluir dinámicamente contenido de una página en otra en el servidor antes de que la página sea entregada al navegador. Esto puede ser útil para incluir contenido de un sistema en otro sin utilizar iframes, ya que iframe genera conflictos con las interfaces de usuario o su carga es más lenta.

Fragmento de código para obtener la interfaz mediante del servidor del acceso solicitado o deseado a ingresar



Fragmento de código para definir a que pagina voy a solicitar la interfaz e interpretarla en Python django



Fragmento donde se implementa la interfaz solicitada que es el laravel\_content



Interfaces que tienen opciones de redirecciones a otros sitios o url dentro del mismo

* Tanto la aplicación en laravel o python puede servir para realizar api request de datos específicos.
* También la otra alternativa es que esos botones se reestructuren y mediante Ajax poder solicitar mediante el servidor la interfaz y mostrarla en la vista actual, reemplazando todo el contenido anterior por el nuevo solicitado.

Anotaciones:

composer require firebase/php-jwt

php artisan vendor:publish --tag="cors"

pip install djangorestframework djangorestframework\_simplejwt

pip install PyJWT

*Route*::post('/token', 'UserController@tokenRequest');

use *Firebase\JWT\JWT*;

use *Firebase\JWT\Key*;

public function tokenRequest(*Request* $request)

    {

        $token = $request->token;

        $secretKey = '$$empanada@2024';

        $datos['modulo'] = 0;

        try {

            $decoded = *JWT*::decode($token, new *Key*($secretKey, 'HS256'));

            $user\_id = $decoded->user\_id;

            $username = $decoded->username;

            $datos['user\_id'] = $user\_id;

            $datos['username'] = $username;

            $moduleQuery = *DB*::connection('globalConn')->select('CALL SDK\_GET\_USER\_MODULE\_ACCESS(?,?)', array(16, $user\_id));

            if (count($moduleQuery)) {

                $datos['modulo'] = 1;

            }

            $datos['save'] = 1;

            return response()->json($datos);

        } catch (*\Exception* $e) {

            $datos['save'] = 0;

            $datos['error'] = $e->getMessage();

            return response()->json($datos);

        }

    }

Kernel.php

*\App\Http\Middleware\Cors*::class, // Add this line

Archivo Cors.php

 public function handle($request, *Closure* $next)

    {

        return $next($request)

            ->header('Access-Control-Allow-Origin', '\*')

            ->header('Access-Control-Allow-Methods', 'GET, POST, PUT, DELETE, OPTIONS')

            ->header('Access-Control-Allow-Headers', 'X-Requested-With, Content-Type, X-Token-Auth, Authorization');

    }

*Route*::get('auto/login/{username}/{token}', 'UserController@autoLogin');

 public function autoLogin(*Request* $request, $username, $token)

$request->session()->put('jwtToken\_laravel', $token);

MENU

let TokenJWT = "{{ Session::get('jwtToken\_laravel') }}";

            $.ajaxSetup({

                headers: {

                    'Authorization': 'Bearer ' + TokenJWT

                }

            });