

**Instituto Costarricense de Electricidad**

**Investigación de modelo de autenticación con Keycloak integrado con redes sociales  
  
  
Mayo 2021**

# Tabla de Contenidos

[Tabla de Contenidos 2](#_Toc72484916)

[Introducción 3](#_Toc72484917)

[Alcance del Entregable 4](#_Toc72484918)

[Manual Técnico. 5](#_Toc72484919)

[Requisitos 5](#_Toc72484920)

[Referencia a documentación: 5](#_Toc72484921)

[NOTA: 7](#_Toc72484922)

[Crear un nuevo realm 7](#_Toc72484923)

[Configuración de la conexión hacia LDAP 8](#_Toc72484924)

[Aplicación JAVA Spring Boot para pruebas. 10](#_Toc72484925)

[KeycloakConfig 12](#_Toc72484926)

[SecurityConfig 13](#_Toc72484927)

[IndexController 14](#_Toc72484928)

[Autenticación con Google. 17](#_Toc72484929)

[Desde keycloak: 17](#_Toc72484930)

[Desde google. 17](#_Toc72484931)

[Autenticación con Facebook 22](#_Toc72484932)

[Desde Keycloak. 22](#_Toc72484933)

[Desde Facebook. 22](#_Toc72484934)

[Notas: 27](#_Toc72484935)

# Introducción

En el presente documento se detalla el resultado de la investigación realizada para el modelo de autenticación con Keycloak integrado con redes sociales.

Se detalla y se muestra el uso y conexión de keycloak hacia ldap y uso de Identity Providers con conexión desde Spring. Además, se plantea una infraestructura para realizar conexiones desde keycloack a través de algunos proveedores de identidad y ldap. También de una conexión con java spring boot y los servicios se crean por medio de contenedores que desplegarán las diferentes aplicaciones.

# Alcance del Entregable

Este documento contiene los pasos requeridos para la instalación y configuración de la aplicación, tales como base de datos, contenedores y compilación además los requisitos necesarios del servidor.

# Contenido.

## Requisitos

Para el despliegue de la arquitectura es requerido la instalación de:

1. Docker.
2. Docker compose.
3. JDK (Opcional) se recomienda la versión de adopt openjdk.
4. Apache Maven.

## Referencia a documentación:

Por medio del archivo **docker-compose.yml** se crean los contenedores docker que tendrán los servicios para el desarrollo de la prueba.

A continuación, se debe abrir una terminal en la ubicación del archivo **docker-compose.yml** y ejecutar siguiente comando:

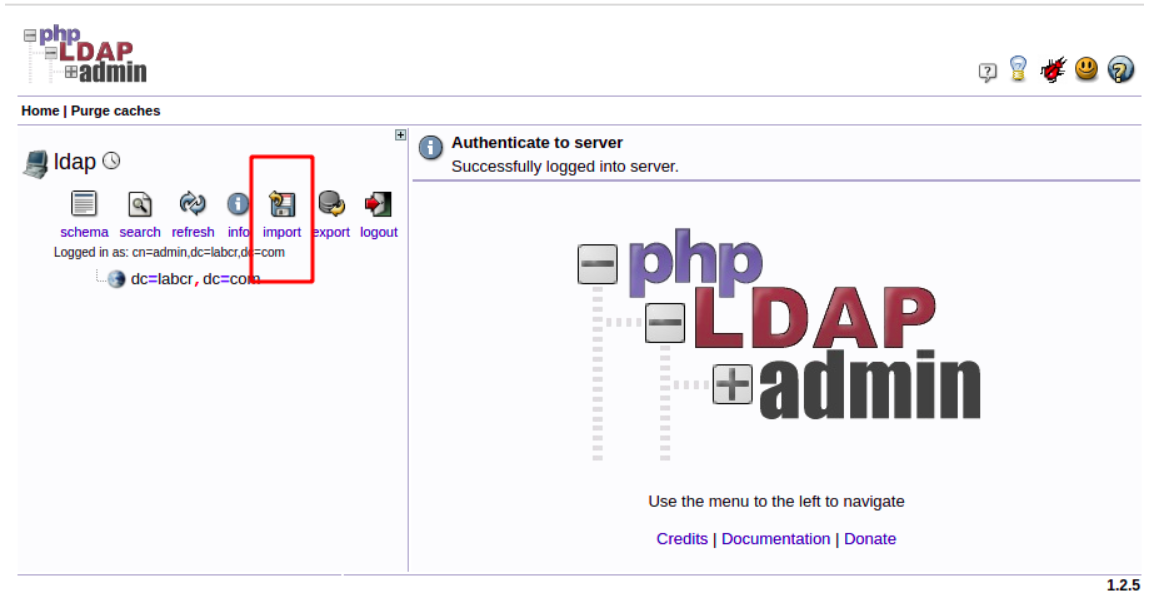


Una vez ejecutado el comando y desplegados los contenedores se debe ingresar a la siguiente ruta:

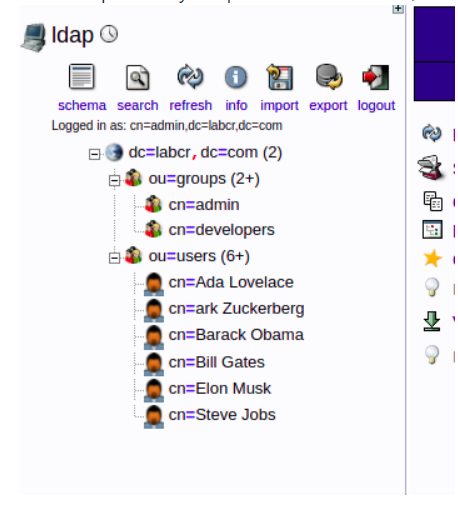
[https://localhost:6443](https://localhost:6443/)

Para ingresar debe dar click sobre el botón **Login** ubicado en la parte superior izquierda de la pantalla (en caso de no mostrar la pantalla para credenciales) y utilizar las siguientes credenciales:

Para el servidor LDAP se puede ingresar cualquier estructura organizacional, sin embargo, para el ejemplo práctico que se está presentando se ofrece una estructura predefinida en la carpeta resources, en el archivo llamado **ldap-import.ldif**, la cual puede ser importada desde el administrador de LDAP indicado en el punto anterior, para lo cual se puede utilizar la opción de importar:



Una vez seleccionado el archivo se presiona el botón **Procced**, luego de finalizar la importación se deberá ver en el panel izquierdo una estructura como la siguiente:



Con esto la configuración del servidor LDAP se encuentra finalizada.

El siguiente paso será ingresar a **keycloak** por medio del siguiente enlace:

http://$(url\_instalacion):9091/auth/

Luego ingresar a la consola de administración, ingresar con los datos asignados en el **docker-compose.yml** por ejemplo:

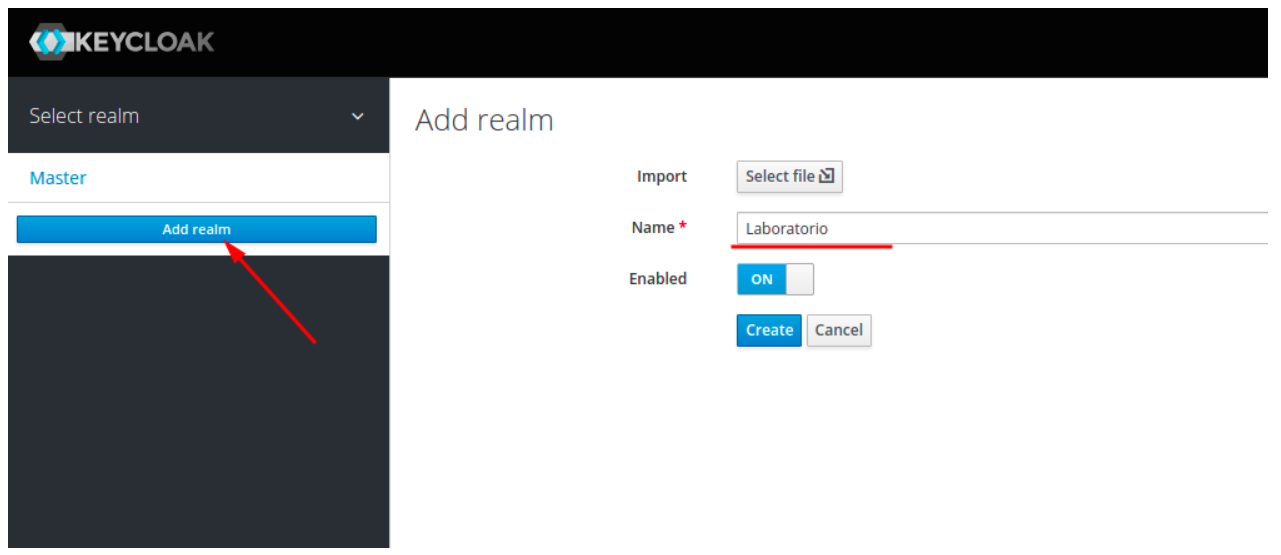


## NOTA:

Para las versiones más nuevas de **keycloak** por defecto se establece la conexión por medio de ssl, por lo que dependiendo del entorno de ejecución puede ser necesario instalar certificados o bien, a nivel de base de datos cambiar en la tabla REALM el valor de la columna **ssl\_required** a "NONE".

## Crear un nuevo realm

En la parte superior izquierda se debe ubicar el mouse sobre "**Master**" para que se despliegue la opción de crear otro realm: hacer click sobre **Add Realm**, como se muestra a continuación:

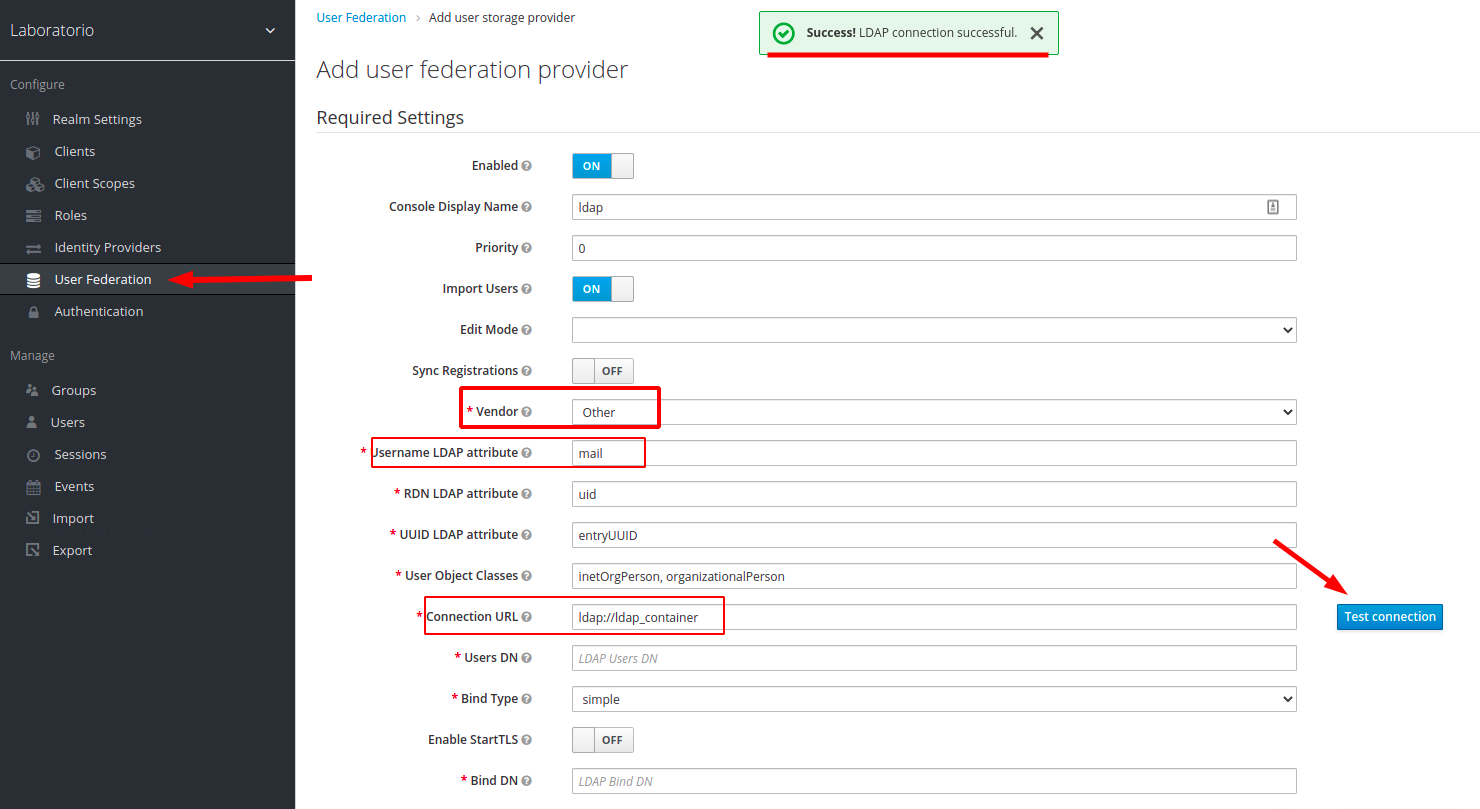


Luego de presionar el botón **Create** se deberá crear un nuevo cliente en el nuevo realm, para ello se debe seleccionar la opción **Clients** en el menú principal y completar los siguientes datos:

* Client ID: laboratorio\_web.
* Client Protocol: openid-conect.
* Click en **save.**
* En la pestaña Settings.
  + Para el campo **Valid Redirect URIs** establecer el valor **http://localhost:9080/\***
  + Guardar los cambios.
* En la pestaña Roles.
  + Agregar un nuevo rol con el nombre **user.**

## Configuración de la conexión hacia LDAP

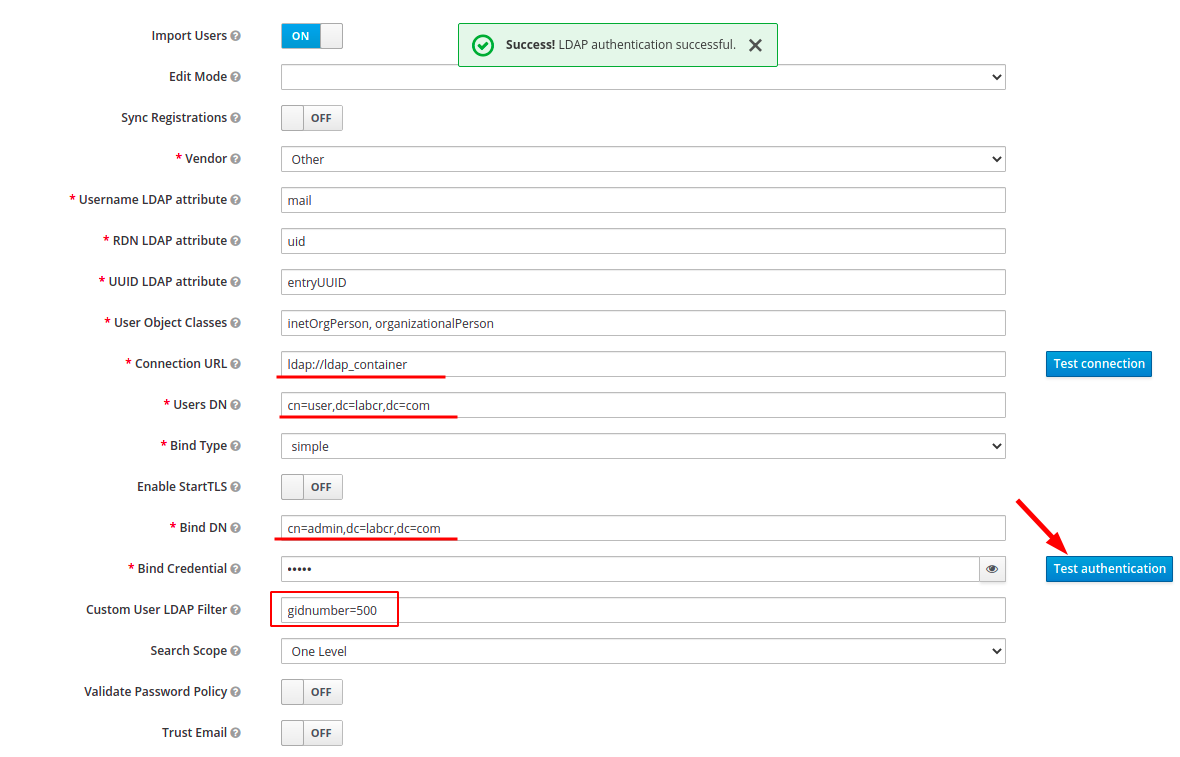
1. En el menú principal seleccionar la opción **User Federation**.
2. Seleccionar el tipo de proveedor **ldap**.
3. Buscar el campo que dice **Vendor** y seleccionar la opción **Other.**
4. Dada la configuración actual, en el campo **Username LDAP attribute** se deberá ingresar **mail**, sin embargo, este campo también se puede configurar con el valor por defecto **uid**.
5. Para el campo **Connection URL** se debe ingresar el valor **ldap://ldap\_container** ya que corresponde al contenedor donde está el servidor de LDAP, este parámetro podrá ser configurado con algún otro servidor LDAP.
6. Al presionar sobre el botón **Test Connection** deberá mostrarse un mensaje de conexión exitosa.



En el campo **Users DN** se debe establecer el valor de acceso definido en la configuración **ou=users,dc=labcr,dc=com**

Retomando el archivo de **ldap-import.ldif** donde se definió la estructura del servidor LDAP, se creó un grupo llamado **developers** con **gidnumber = 500**,para este ejemplo, se realizará una conexión donde únicamente este grupo se listará, por lo tanto en campo **Custom User LDAP Filter** se establecerá el valor **(gidnumber=500)** respetando los paréntesis

El campo **Bind DN** debe quedar establecido con el valor **cn=admin,dc=labcr,dc=com** y en **Bind Credential** la contraseña genérica usada para acceder **admin** y por ultimo guardar los cambios.



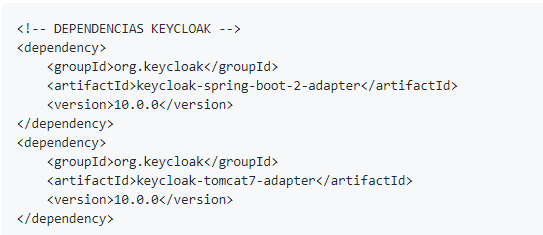
Esta configuración es suficiente para probar que los usuarios puedan hacer login, desde este punto se puede ingresar a la aplicación con una conexión desde Spring o cualquier otra conexión Open ID, sin embargo, para probar el acceso hacia LDAP, se puede presionar el botón **Synchronize all users** ubicado al final de la página y verificar en la sección **Users** del menú principal que efectivamente se hayan importado los usuarios.

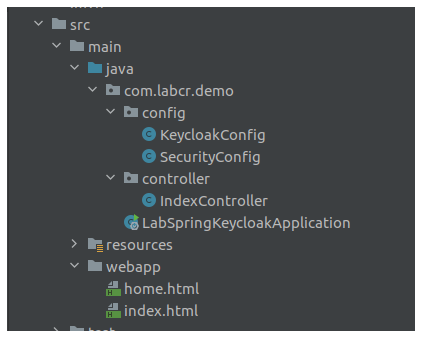
## Aplicación JAVA Spring Boot para pruebas.

1. Debe ingresar a la siguiente dirección: [https://start.spring.io](https://start.spring.io/) y crear un proyecto con las siguientes características:

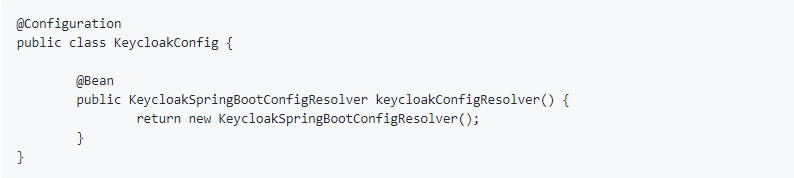


Los demás campos se pueden llenar a con los valores de preferencia. La recomendación es que se descargue con las dependencias de Spring Web y Spring Security, aunque estos pueden ser agregados posteriormente, al completar los datos se debe presionar el botón **GENERATE**.

1. En la carpeta descargada buscar el archivo **pom.xml** y agregar las siguientes dependencias.
2. En **src/main** agregar la carpeta **webapp.**
3. Dentro de la carpeta **wepapp** crear un archivo llamado **index.html.**
4. En la misma ubicación crear un archivo llamado **home.html**, este será utilizado para acceder desde el **index.html.**
5. Se deberán crear los siguientes archivos java:
   1. Archivo de configuración keyclok (KeycloakConfig.java)
   2. Archivo de configuración de seguridad (SecurityConfig.java)
   3. Index controller (IndexController.java)



### KeycloakConfig



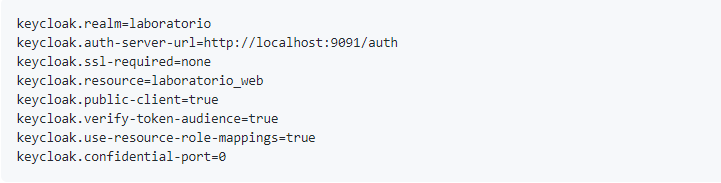
### SecurityConfig



### IndexController

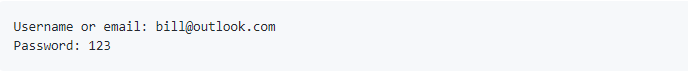


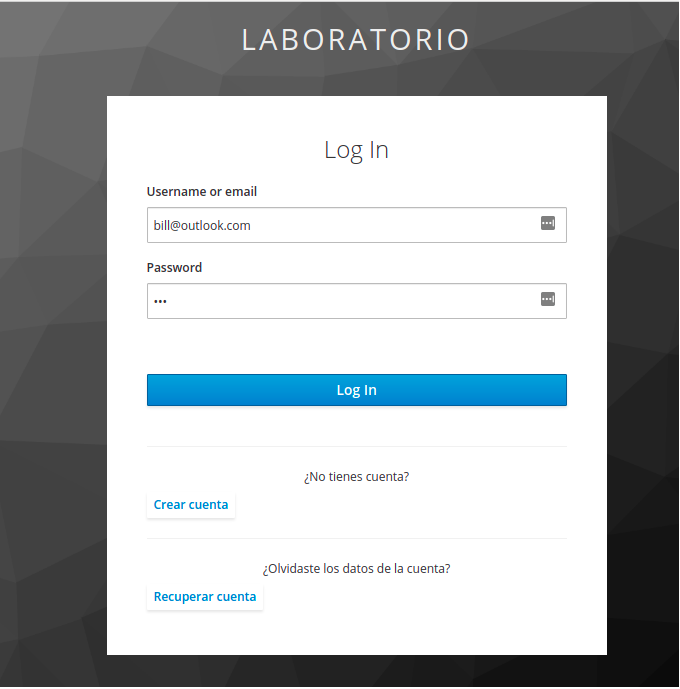
En la ubicación **src\main\resources** configurar en el archivo **aplication.properties** los siguientes valores:



Importante: <http://localhost:9091/auth> debe ser reemplazado por la url donde está el servidor **keycloak** en la configuración del realm, en la pestaña **Login**, se debe ajustar el campo **Require SSL** de acuerdo a si se tiene o no ssl desde el cliente.

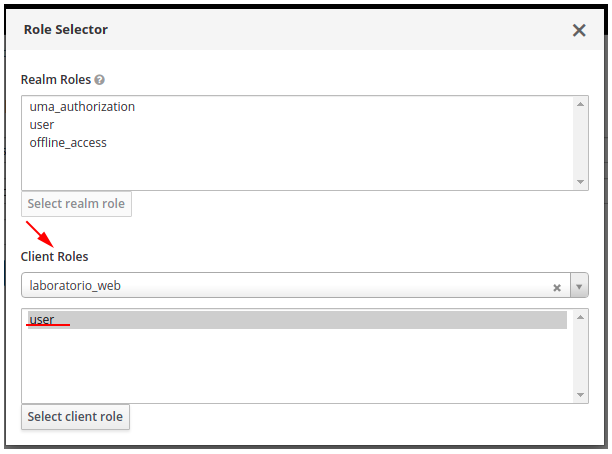
Con lo anterior configurado al dar click sobre **Entrar** el sistema re direccionará hacia **keycloak** para realizar el login, en esta pantalla se pueden utilizar los usuarios existentes en LDAP:





El acceso desde java y conectado a **keycloak** ha sido creado, sin embargo, al iniciar sesión se mostrará un error **403** que indica que no se tiene acceso al recurso solicitado, esto se debe a que en el archivo **SecurityConfig** se ha indicado que a la página **/home.html** tienen acceso los roles **admin** y **user**, pero, al crear estos usuarios desde ldap, los permisos no han sido asignados. Por lo que se realizará un mapeo desde el **User Federation** configurado anteriormente. Para esto se debe ingresar al registro de LDAP creado y configurado y dar click sobre la pestaña **Mappers** y seguir los siguientes pasos:

* Dar click sobre **Create.**
* Asignar un nombre de preferencia.
* Seleccionar la opción **hardcoded-ldap-role-mapper.**
* Dar click sobre la opción **Select Role.**
* En la sección **Client Roles** seleccionar **laboratorio\_web** y seleccionar el rol **user**.



Dar click sobre **save**, para cerrar sesión se debe ingresar a la url (<http://localhost:8080/logout>) y al volver a ingresar se podrá ver que ahora si es permitido el acceso.

## Autenticación con Google.

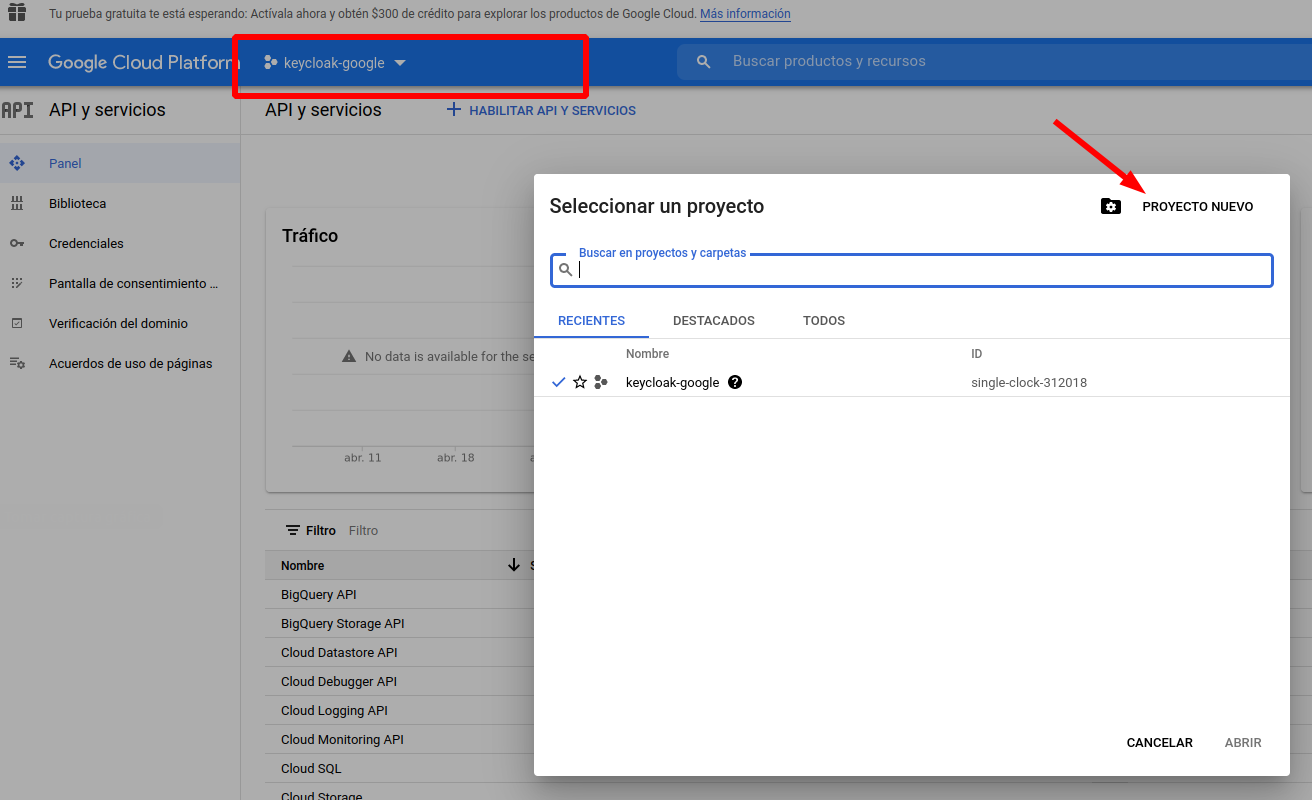
Para poder configurar el acceso con google se requiere contar con una cuenta de desarrollador desde google y se deberá llevar en paralelo la configuración junto con los cambios en **keycloak**.

### Desde keycloak:

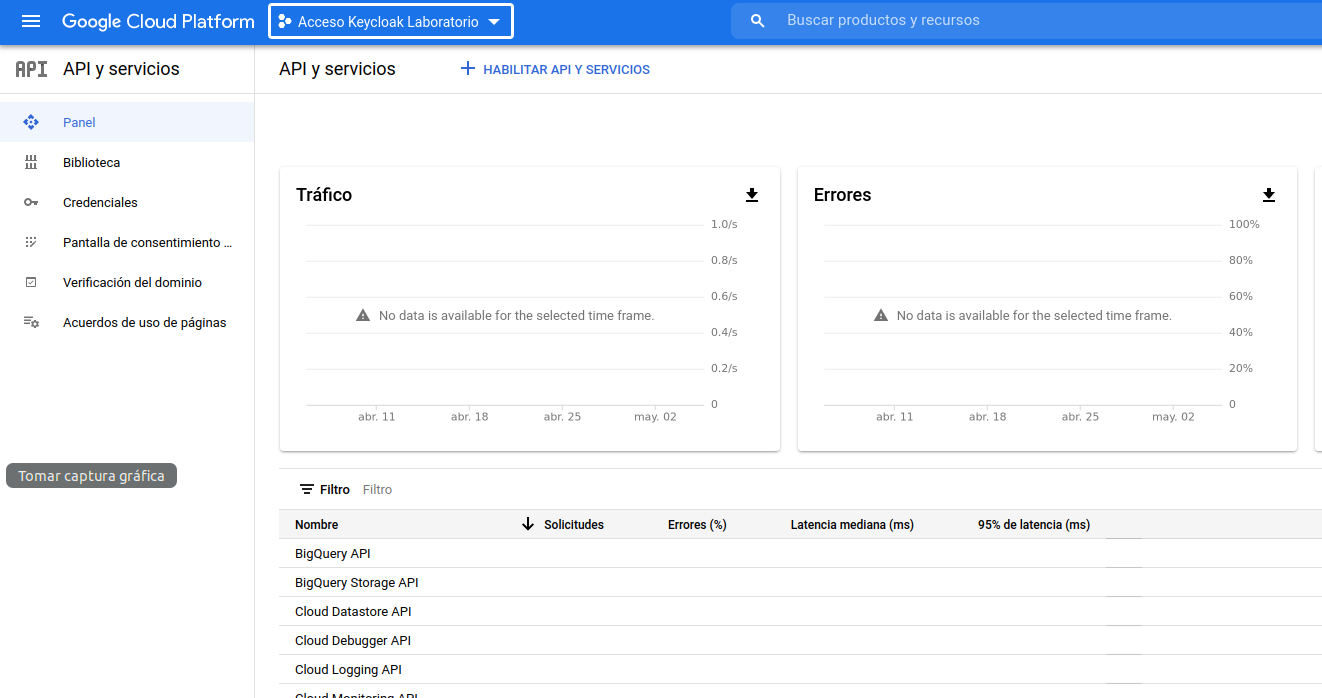
1. Dar click sobre **Identity Providers.**
2. Seleccionar **Google**.
3. **Client ID** y **Client Secret** son dos campos que se llenarán más adelante.
4. El campo **Redirect URI** se utilizará para la configuración desde google.

### Desde google.

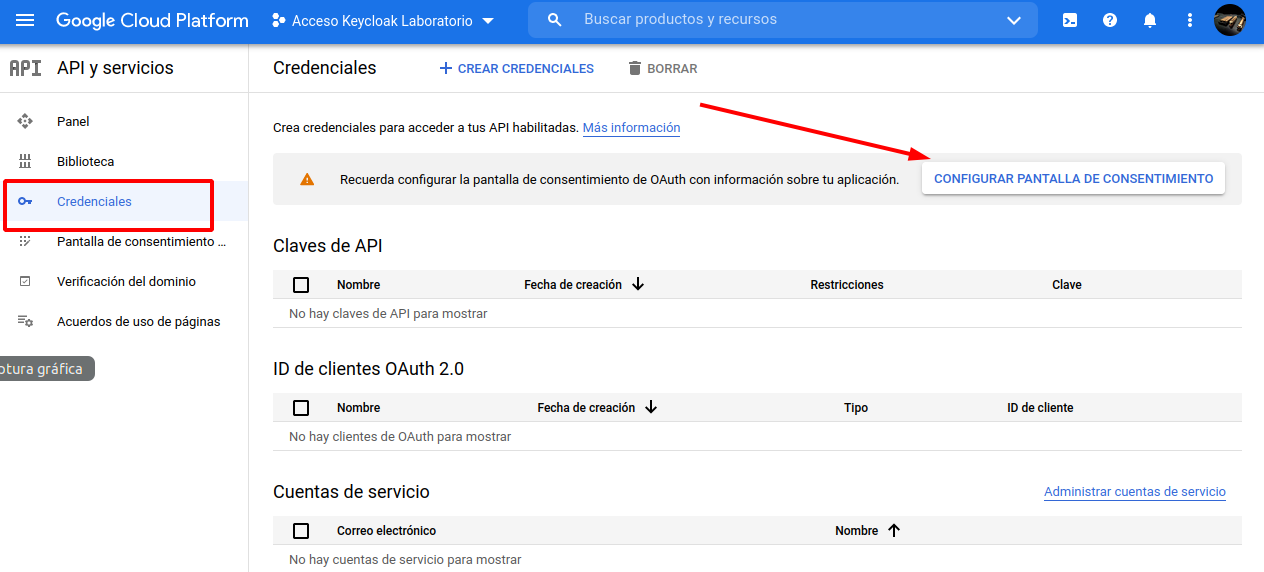
1. Ir a [google api](https://console.developers.google.com/).
2. En la parte superior dar click al lado del logo, donde se desplegará un cuadro de dialogo para crear un API.
3. Seleccionar **Proyecto Nuevo.**



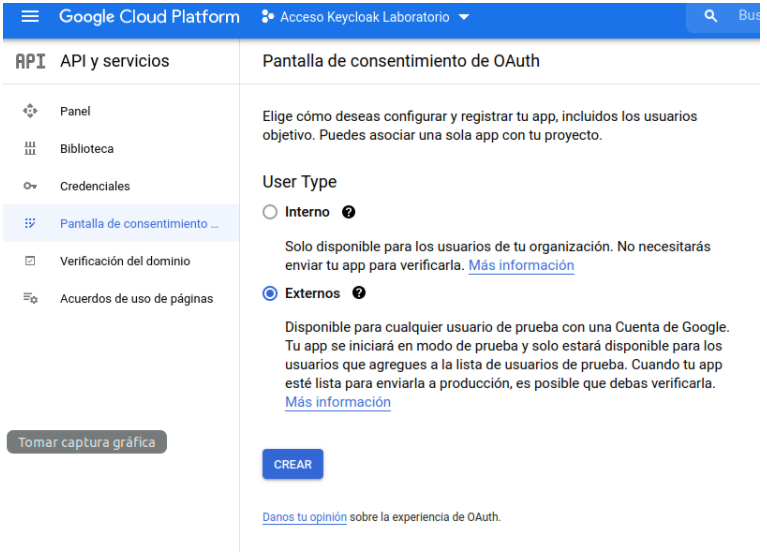
1. Asignar el nombre y dar click sobre el botón **Crear.**
2. Al crear el proyecto se desplegará el menú de administración.



1. Dar click sobre **Credenciales**.
2. En esta pantalla aparecerá un mensaje recordando configurar el consentimiento de **OAuth**, es importante configurar este consentimiento.



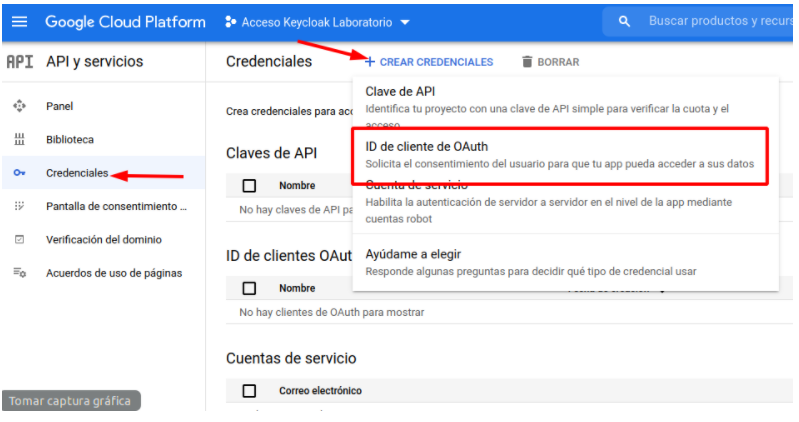
1. En la pantalla de consentimiento se debe seleccionar **Externos** y dar click sobre **CREAR.**



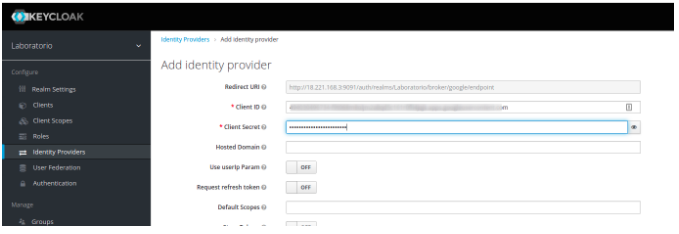
1. En la siguiente pantalla se configurará la información que aparecerá en el cuadro desplegable a la hora de iniciar sesión la primera vez.
2. Seleccionar el nombre de la aplicación que leerán todos los usuarios.
3. Llenar los demás campos de acuerdo a los requisitos de la aplicación.



1. Dar click en **guardar y continuar**.
2. Regresar nuevamente a **Credenciales** y dar click sobre **Crear Credenciales**.
3. Seleccionar la opción **Id** de cliente de **oAuth**.



1. Seleccionar tipo de aplicación web.
2. Ingresar el nombre.
3. En el espacio **URI de redireccionamiento autorizados** se debe ingresar el valor del campo **Redirect URI** de keycloak mencionado anteriormente.
4. Al guardar los cambios, se mostrarán los valores que se deberán ingresar en **keycloak**.
5. Regresar a **Keycloak** y completar los campos.



1. Con esta configuración al ingresar nuevamente a la pantalla de login desde la aplicación java, se mostrará la opción de ingresar con google.

## Autenticación con Facebook

De la misma forma la autenticación con google se deberá configurar en paralelo a **keycloak** con el api de Facebook.

### Desde Keycloak.

1. Dar click sobre **Identity Providers.**
2. Seleccionar **Add provider.**
3. Seleccionar **Facebook**.
4. Tener en cuenta el campo **Redirect URI** ya que será utilizado desde la configuración de Facebook.

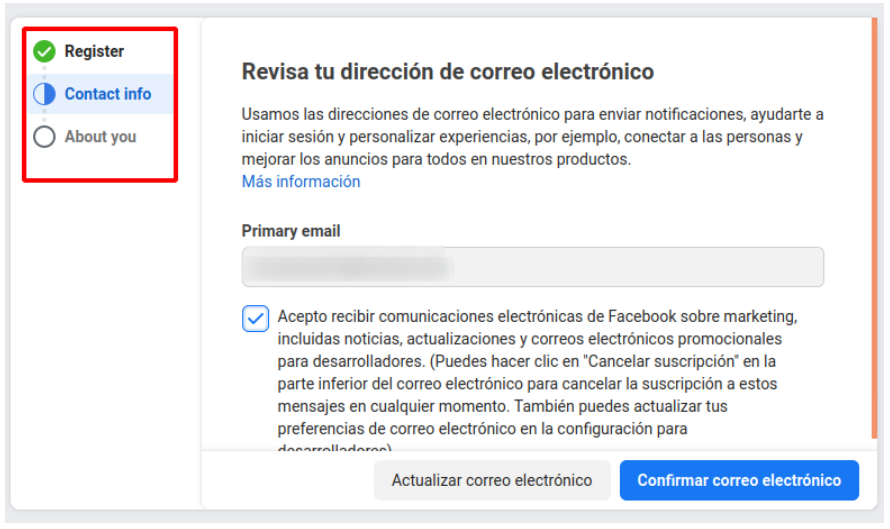
### Desde Facebook.

Se debe seguir los pasos indicados en la página de facebook developers, ya que es normal que esté cambiando la parte visual de la misma, por lo que se ofrece una guía general teniendo en cuenta que esto puede variar a lo largo del tiempo.

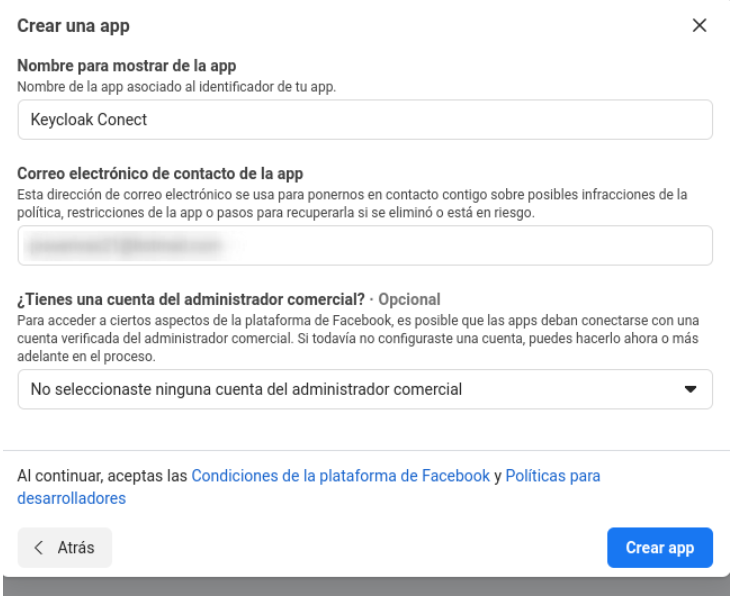
1. Ir a la página de [Facebook developers](https://developers.facebook.com/).
2. Logearse en la consola de administración.
3. Registrar la cuenta como cuenta de desarrollador.



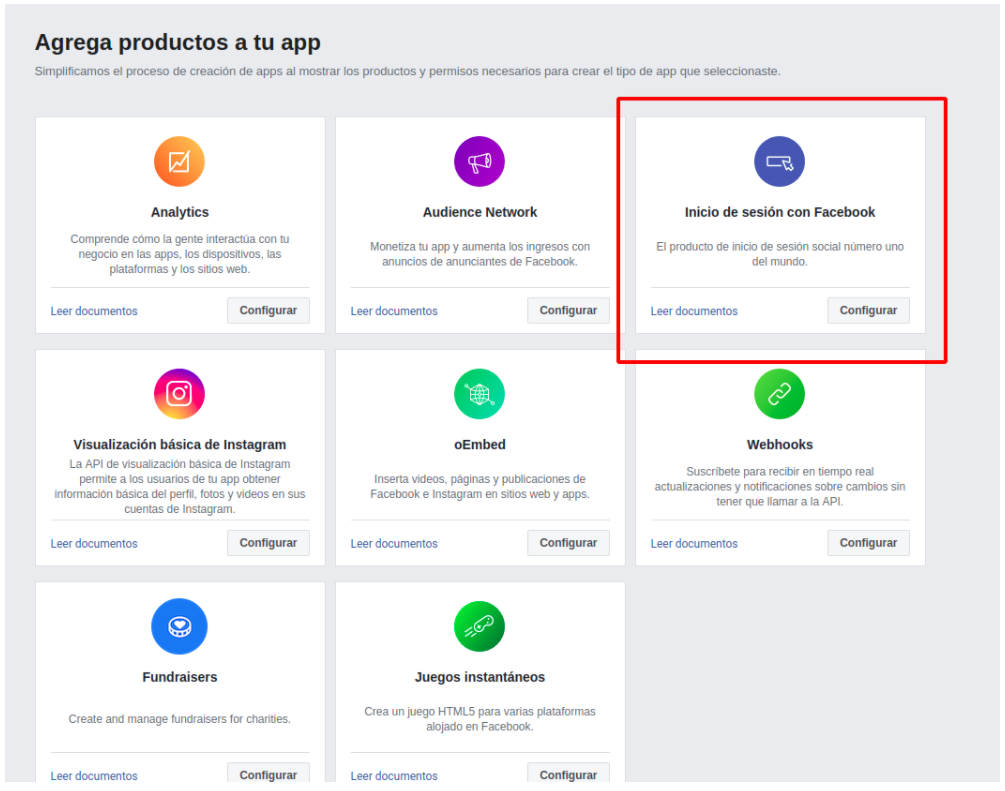
1. Seguir los pasos para completar el registro como desarrollador.



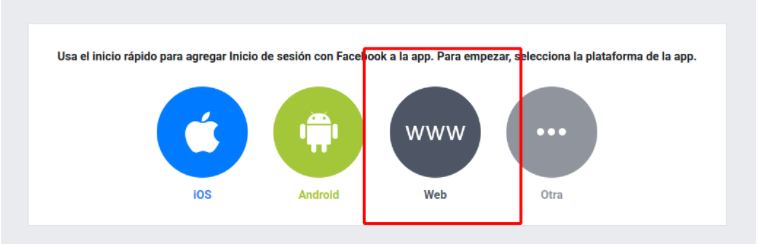
1. Crear nueva aplicación.



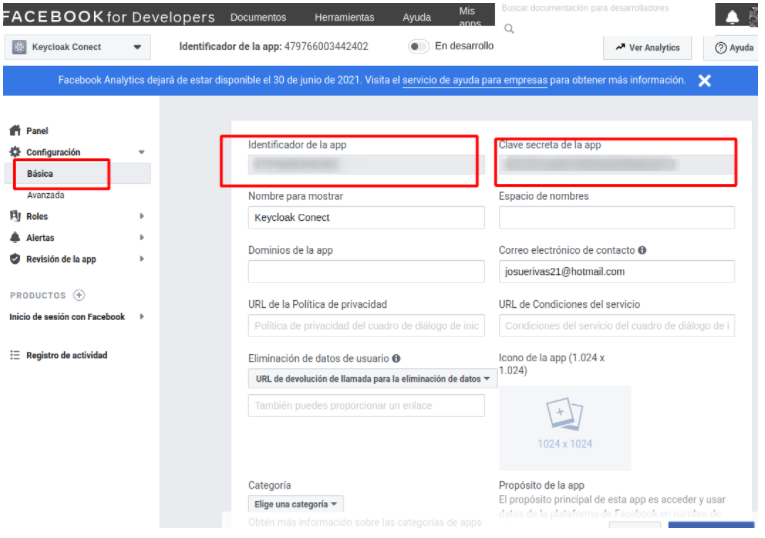
1. Agregar producto de inicio de sesión con Facebook.



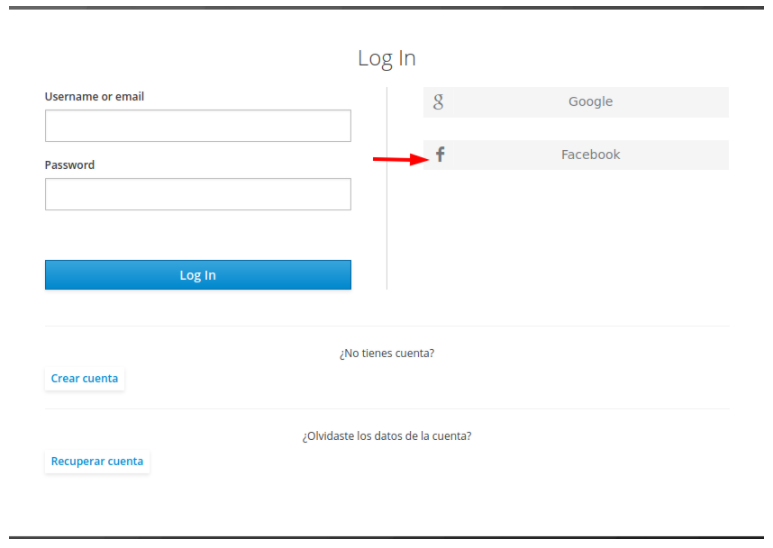
1. Seleccionar tipo de aplicación **web**.



1. Ingresar en el campo URL del sitio web la dirección generada por **Keycloak**.
2. Desde la configuración básica seleccionar los campos de identificador de la app y la clave secreta de la app e incluirlos dentro de **Keyclok**.



1. Después de esta configuración se podrá ingresar a la aplicación y realizar el login, hasta este punto debe aparecer tanto el acceso con google como con Facebook.



## Notas:

* De acuerdo a las configuraciones realizadas, al utilizar Google como proveedor de identidad, este no permite usar ip's por lo que **keycloak** debe estar bajo un nombre de dominio valido.
* Al usar Facebook como proveedor de identidades, a pesar de que permite agregar ip's en los dominios, no se recomienda que **keycloak** se encuentre en una ip diferente a la aplicación, ya que se complica el redireccionamiento, para ello se recomienda que igual estén bajo un mismo nombre de dominio.

Imagen que contiene edificio, jaula

Descripción generada automáticamente