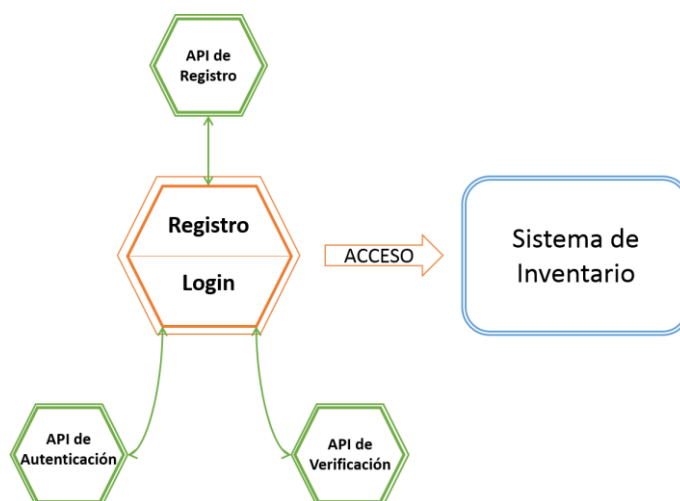


## Creación y uso de Microservicios

**Objetivo:** Crear un sistema de registro y autenticación de usuarios utilizando microservicios, cuyo uso complementará la parte 1 del Proyecto Final.

**Esquema:**



**Pasos:**

1. Crea un microservicio para el **registro de usuarios**. Este microservicio debe recibir una solicitud POST con los datos del usuario (nombre, correo electrónico y contraseña), validarlos y almacenarlos en la base de datos. Si la operación es exitosa, el microservicio debe responder con un código de estado http 201 (Creado).

Las validaciones a realizar son:

- Usuario disponible.
- Correo (expresión regular).
- Contraseña segura.

NOTA: para este registro se utilizará una nueva colección de Firebase, de nombre "usuarios\_sistema".

2. Crea un microservicio para la **autenticación de usuarios**. Este microservicio debe recibir una solicitud POST con las credenciales del usuario (correo electrónico y contraseña), validarlas y generar un *token* de acceso que debe ser devuelto al cliente. Si la autenticación es exitosa, el microservicio debe responder con un código de estado http 200 (OK).



Un *token* de acceso es un *string* aleatorio que identifica a un usuario y puede ser utilizado por la aplicación para realizar llamadas a un API. El *token* incluye información sobre cuándo expirará, para qué aplicación se generó y/o qué aplicación generó el *token*.

3. Crea un microservicio de **verificación de token**. Este microservicio debe recibir una solicitud GET con el token de acceso en el header Authorization. Si el token es válido, el microservicio debe responder con un código de estado 200 (OK) y el nombre del usuario. De lo contrario, debe responder con un código de estado 401 (No autorizado).
4. Prueba el **flujo completo** con un HTTP client de pruebas para asegurarte de que todo funciona correctamente antes de implementar la interfaz. Asegúrate de probar la creación de usuarios, la autenticación y la verificación de *tokens*. Y almacena el resultado de tus pruebas en un POSTMAN-Collection o en un proyecto JSON de INSOMNIA.
5. Crea una **interfaz de usuario** que permita a los usuarios registrarse, iniciar sesión y cerrar sesión. Para la autenticación, debes llamar al microservicio correspondiente y almacenar el *token* de acceso en la “sesión” del usuario, este debe ser validado para permitir el acceso al sistema. Para cerrar la sesión, simplemente elimina el *token* de la sesión.

#### Recursos necesarios:

- Lenguaje de programación para los microservicios:
  - PHP,
  - Python y
  - C# (u otro)
- Framework de microservicios:
  - Slim,
  - Flask y
  - ASP.NET Core (u otro)
- Base de datos para almacenar los usuarios de acceso al sistema:
  - Firebase
- Cliente HTTP de Pruebas:
  - POSTMAN ó
  - INSOMNIA



- Tecnologías para la implementación de la Interfaz del Cliente:
  - Las utilizadas en el Proyecto (parte 1)

Este proyecto es sólo un ejemplo de cómo se pueden utilizar los microservicios en un proyecto real. En esta ocasión, el uso adecuado de este conjunto de microservicios servirá para dar acceso al “sistema de inventario” implementado ya en la parte 1 del Proyecto Final.