# SPN APP versión 2.0

Marzo 2022

El proyecto APP está creado con FLUTTER. Herramienta multiplataforma creada por GOOGLE para realizar aplicaciones, compiladas nativamente desde una única base de código.

El alcance del proyecto cubre los sistemas operativos Android y iOS.

Herramientas utilizadas en el desarrollo:

- SDK de Flutter.
- Se puede utilizar Android Studio y/o VSCode (Visual Studio Code) como editores principales, para utilizar Flutter en estos editores se deben habilitar plugins específicos para cada uno, ver configuración.
- XCode, IDE para desarrollar en el entorno Apple. Todo el código se puede crear y modificar en los editores anteriores, pero para realizar las compilaciones y el despliegue al App Store es obligatorio utilizar XCode.

Seguridad en APP

Las compilaciones de la app siguen los pasos de seguridad brindada por cada plataforma.

**iOS:** La compilación es creada bajo un certificado de Developer autorizado por Apple. El entorno Apple es cerrado, y al realizar la compilación el archivo es encriptado, al igual que la data de la app, lo cual no permite que otras apps dentro o fuera del equipo accedan a la data interna del app. Ver más.

Para la colocación de la app en el App Store, Apple evalúa y se asegura de que la app cumpla con sus requisitos de seguridad.

**Android:** La compilación en android genera un archivo APK, para su generación se utiliza una firma única, la cual protege de que no se suplante en el Google Play por otra app.

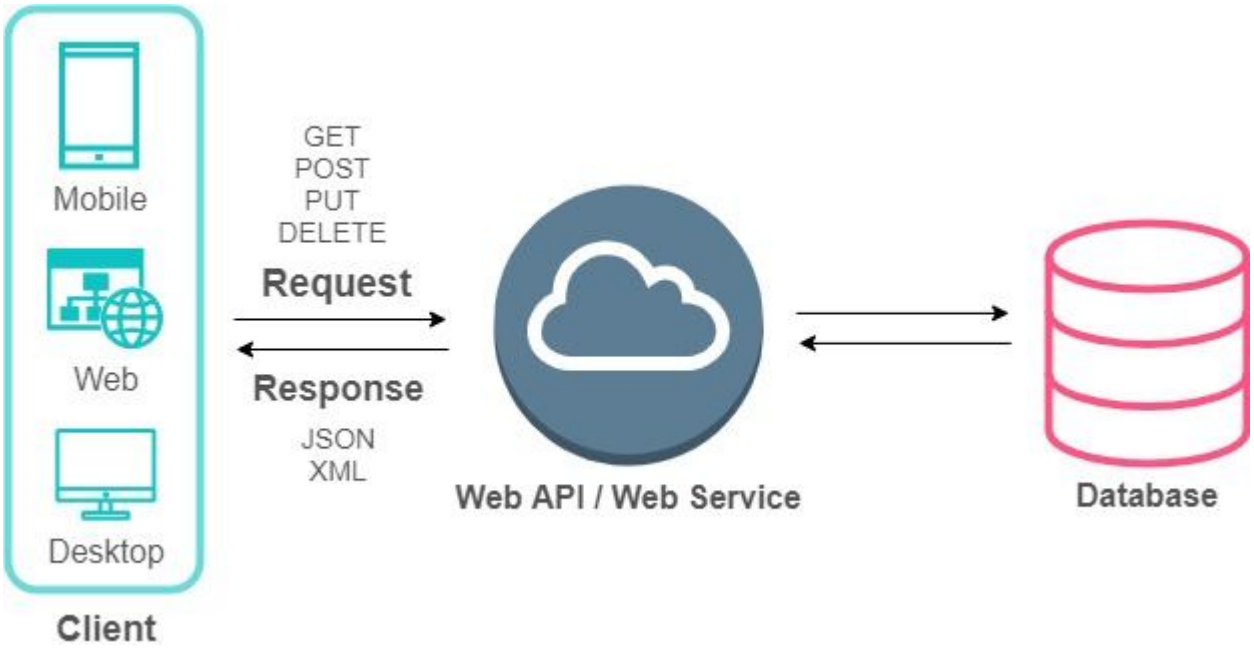
La compilación del app utiliza métodos de ofuscación del código, para proteger de que se obtenga un código fuente legible. Ver más.

Los archivos y datos son encriptados en el app para que otras apps no accedan a los datos.

**API: (Application Programming Interface)**

Este proyecto está desarrollado en C# .NET 4.8, y se basa en un proyecto Web API. Las API permiten que sus servicios se comuniquen con otros, sin necesidad de saber cómo están implementados.

Los recursos son expuestos a través de los métodos HTTP (GET, POST, PUT y DELETE).




Los datos transmitidos son en formato JSON, y mediante el protocolo HTTPS los datos y encabezados viajan de forma segura. De esta forma es recomendable y necesario que los APIs Clientes utilicen certificados SSL.

**Base de Datos:**

La base de datos no está expuesta o accesible desde el app, el API es quien recibe la petición y realiza la consulta a la DB.

Las DB cuentan con métodos de autenticación, y la comunicación entre el API y la DB es mediante una autenticación válida.

**Seguridad para los recursos expuesto:**

El proyecto implementa la autenticación basada en Tokens JWT, El Token se trata de una cadena formada por tres partes (cabecera, payload y firma), que son procesadas por un algoritmo de SHA256.

Esto da protección de que, si se obtiene la URL del API o la URL de un recurso, sea necesario una autenticación válida previamente.

Para obtener un token se debe de realizar una autenticación de usuario válida, y en la respuesta de la llamada el API retorna un token.

Ya con el token en mano, cuando se llama a un recurso, se envía dicho token y el API comprueba que ese token sea válido.

De hacer una llamada al API y no enviar un token o enviar un token invalido retorna un error 401, que indica que la petición (*Request*) no ha sido ejecutada porque carece de credenciales válidas de autenticación.

En el diagrama siguiente se muestra el flujo de cómo se lleva a cabo el proceso: