

# **Guía de Instalación**

## **OperaCE**

# **OPERACE**



**Guía para realizar el despliegue de la herramienta**

### **Creadores:**

**Dylan Garbanzo Fallas (2021057775)**

**Alejandra Rodríguez Castro (2021131070)**

**Carlos Eduardo Rodríguez Segura (2022437835)**

**Ricardo Borbón Mena (2021132065)**

**Jose Maria Vindas Ortiz (2022209471)**

# Contenido

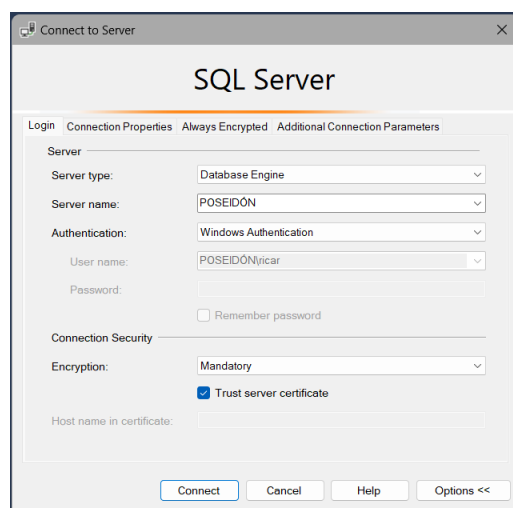
a)	Instalación de la Base de Datos .....	2
a)	Instalación de la API Rest Service .....	5
b)	Instalación de la Aplicación Móvil .....	6
c)	Instalación de la página Web .....	11

## a) Instalación de la Base de Datos

Lo primero que debemos realizar corresponde a la instalación del motor de la base de datos, correspondiente a Microsoft SQL Server, para ello visite el siguiente [sitio web](#)

Siga los pasos de instalación que le proporciona la página web, recuerde que también se necesita instalar el entorno de desarrollo SQL Server Management Studio (SSMS) que se debe de instalar de forma adicional al terminar la instalación de SQL Server.

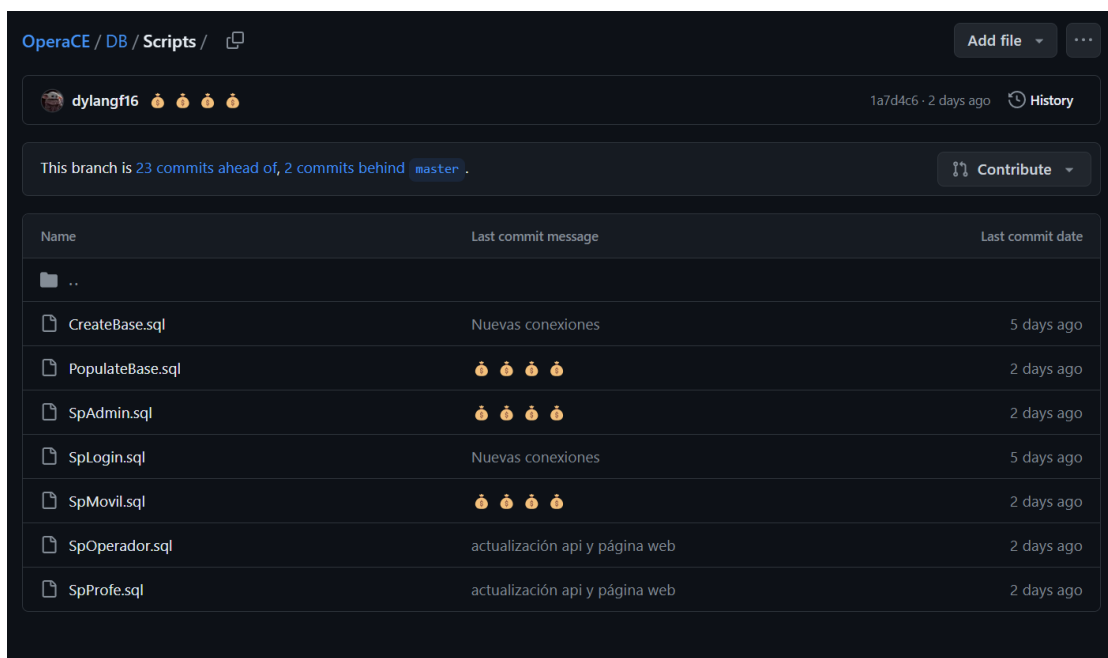
Una vez realizada la instalación, se debe de abrir el SSMS e inicializar nuestra computadora como base de datos. Se desplegará un mensaje similar al que vemos a continuación.



**Figura 1.** Ventana emergente de SSMS para inicializar la base de datos en nuestra computadora

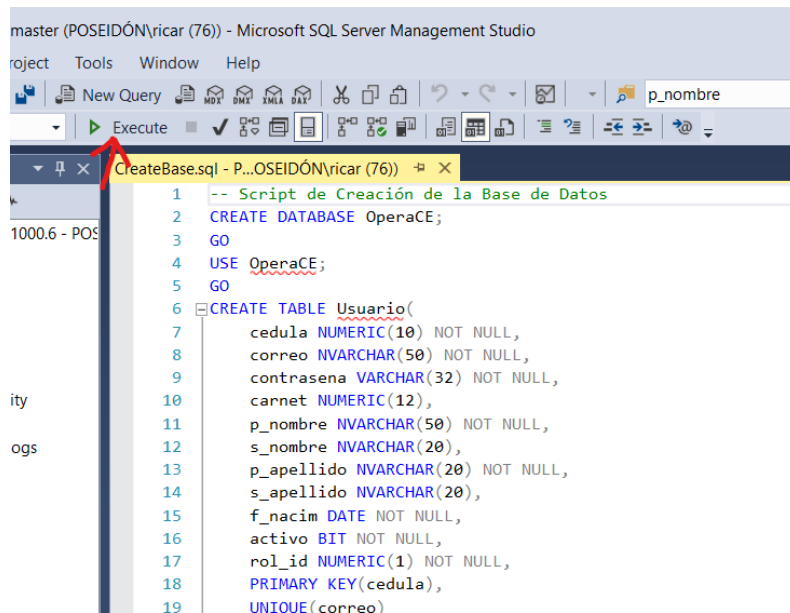
Es importante asegurarse que la casilla del “server name” contenga el nombre de nuestro dispositivo para que el servidor de base de datos se cree correctamente en el computador.

Ahora para poder crear la base de datos correspondiente a nuestro prototipo procederemos a descargar los scripts necesarios en el siguiente [repositorio](#) de github. Nos centramos en los archivos .sql que se encuentra en DB/Scripts:



**Figura 2.** Dirección de la carpeta con los archivos necesarios para la creación de la base de Datos

A continuación, debemos de abrir en nuestra herramienta SSMS el archivo llamado “CreateBase” se nos abrirá en el edito de texto del entorno de desarrollo. A continuación, deberemos de ejecutar su contenido presionando la tecla “F5” o presionando el botón verde “Run” en la caja de herramientas:



**Figura 3.** Botón de ejecución para correr los comandos dentro del archivo seleccionado.

Este archivo creará las estructuras y las relaciones de la base de datos. A continuación, debemos de llenar las tablas con los valores iniciales. Para ello abrimos en nuestro editor en archivo llamado “PopulateBase” y ejecutarlo de la misma forma que realizamos anteriormente.

Por último, debemos de ejecutar los procedimientos de la base, estos corresponden a todos los archivos que inician con “Sp”. No importa el orden en que se ejecuten, pero se debe de asegurar que se ejecutaron todos y cada uno de ellos (existen 4 archivos “SpLogin”, “SpAdmin”, “SpProfe” y “SpOperador”).

Es importante ejecutar los archivos en el siguiente orden para evitar errores en la creación de la base:

*CreateBase.sql → PopulateBase.sql → All SpFiles.sql*

Con esto terminamos la instalación de la base de datos para OperaCE, para confirmar su creación podemos verificar en la sección de databases en la parte izquierda de la herramienta. Asegurarse de refrescar antes de verificar.

## a) Instalación de la API Rest Service

Para la instalación de la API, primero es necesario descargar .NET Core, el cual se puede descargar desde la página oficial: [Download .NET \(Linux, macOS, and Windows\) \(microsoft.com\)](https://dotnet.microsoft.com/download), este es el lenguaje que usaremos para crear y configurar la API. Para el proyecto se descargó la versión .NET 8.0.

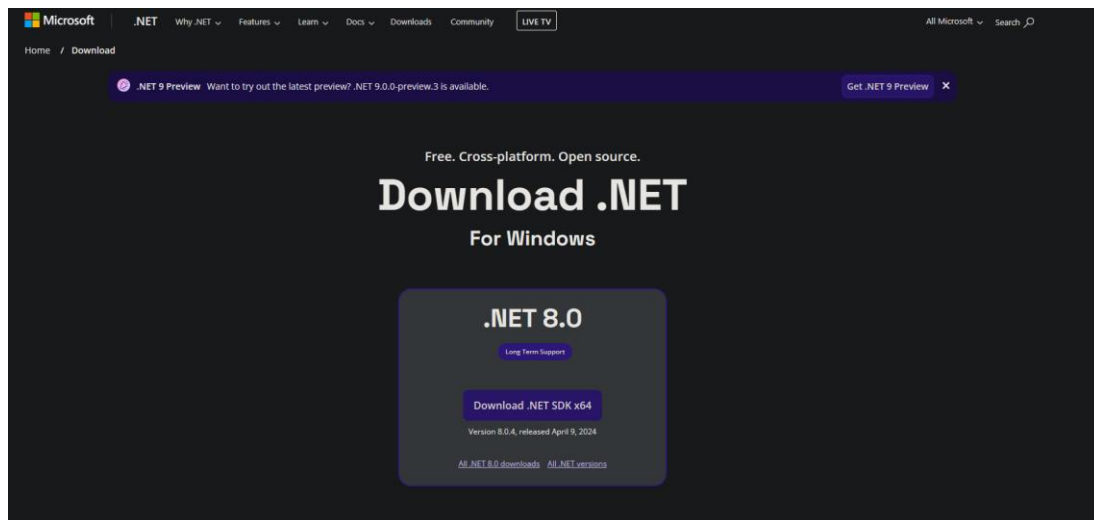


Figura 4. Página oficial de descarga de .NET Core

Además de esto, se descargó Visual Studio Community 2022 como IDE para trabajar con .NET, el cual se puede descargar desde la página oficial: [Visual Studio: IDE and Code Editor for Software Developers and Teams \(microsoft.com\)](https://visualstudio.microsoft.com/)

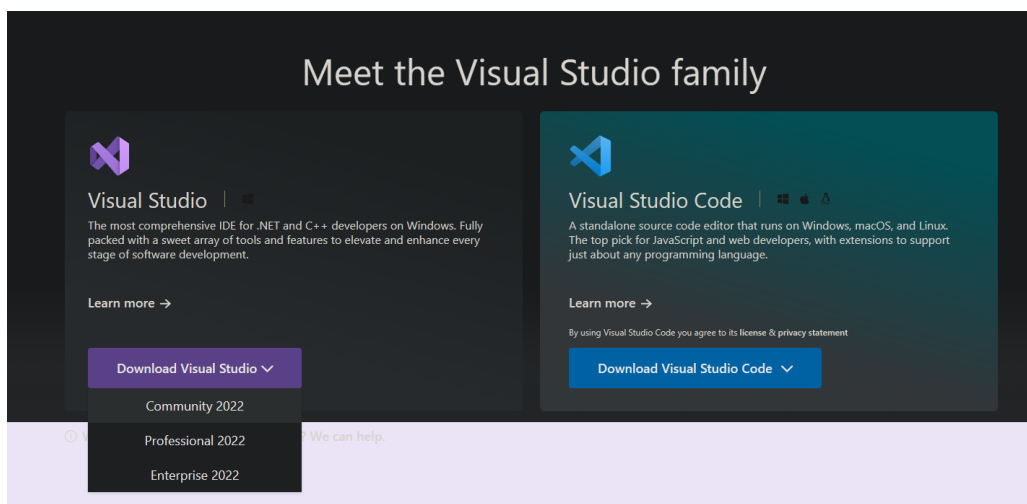


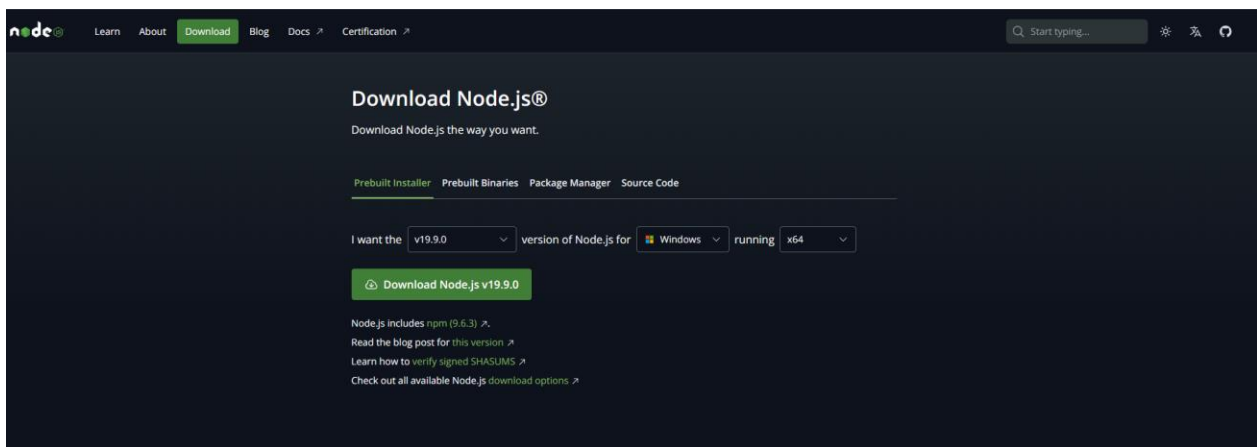
Figura 5. Página oficial de descarga de Visual Studio

Una vez descargado, se va a presentar una ventana, la cual recomienda descargar diferentes paquetes de librerías, no es necesario escoger alguna en específico, pero para la elaboración del proyecto, se descargaron los paquetes: ASP.NET and web development y .NET desktop development.

Una vez descargados ambos, se podrá obtener el código de la API en el repositorio usado para el proyecto, en la rama de API, que contiene la carpeta API con dicho código.

## b) Instalación de la Aplicación Móvil

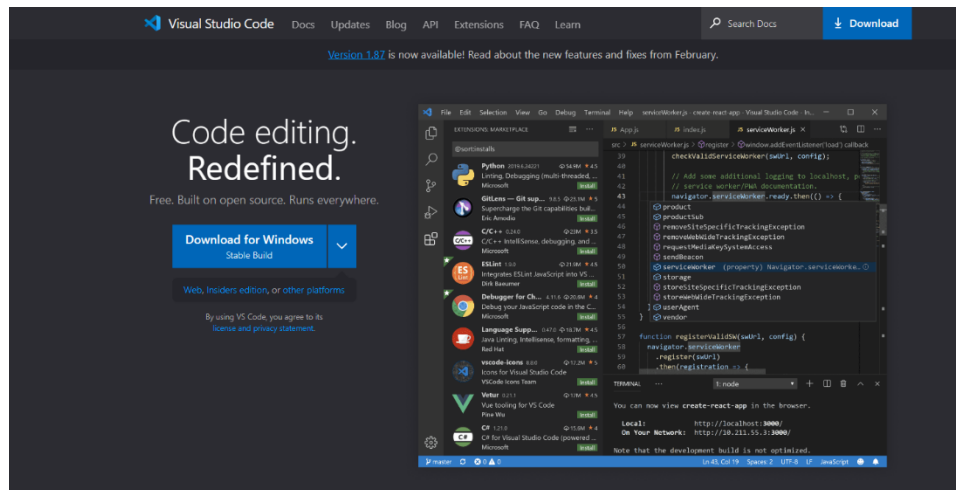
Lo primero que debemos realizar es instalar node.js versión 19.90.0, 18.19.1 o 20.11.1 desde la página oficial: [Node.js — Download Node.js® \(nodejs.org\)](https://nodejs.org/)



**Figura 6.** Página oficial de descarga de Node.js. Elaboración Propia.

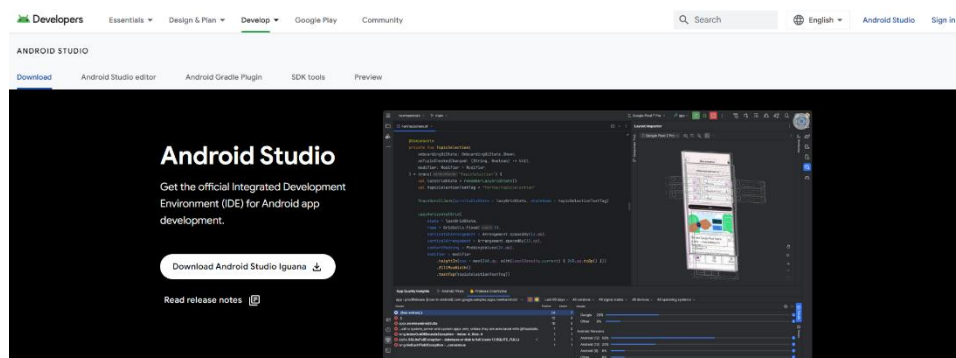
Para confirmar que haya instalado correctamente node.js, puede abrir su consola de preferencia y ejecutar el comando “node -v”, si este le da un mensaje dándole una versión de node.js, el programa se instaló correctamente.

Seguidamente, instalamos Visual Studio Code desde su página oficial: [Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/), no se recomienda el uso de otros IDEs que no sea este, ya que puede generar errores al generar archivos necesarios para que estos funcionen con el programa, y puede que genere errores al ejecutar el programa.



**Figura 7.** Página oficial de descarga de Visual Studio Code. Elaboración Propia.

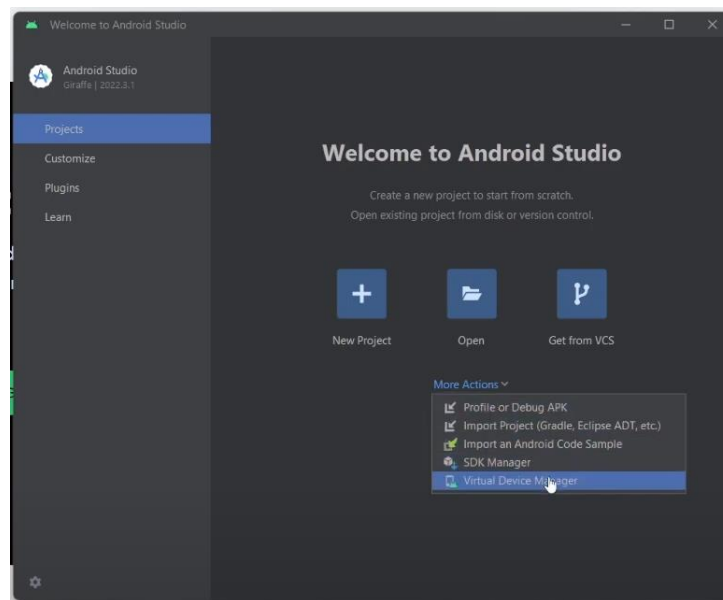
Por último en descargas de aplicaciones necesarias, tenemos que descargar Android Studio Iguana desde su página oficial: [Download Android Studio & App Tools - Android Developers](#), el proyecto se puede ejecutar de manera local en un dispositivo Android físico, pero para esto, es necesario activar el modo Administrador de su dispositivo móvil, y activar el debug por USB, y este proceso varía dependiendo de su modelo de teléfono, además, el proyecto está simulado para un Google Pixel 7 Pro, por ende, puede que muestre inconsistencias en la interfaz en otros dispositivos.



New features

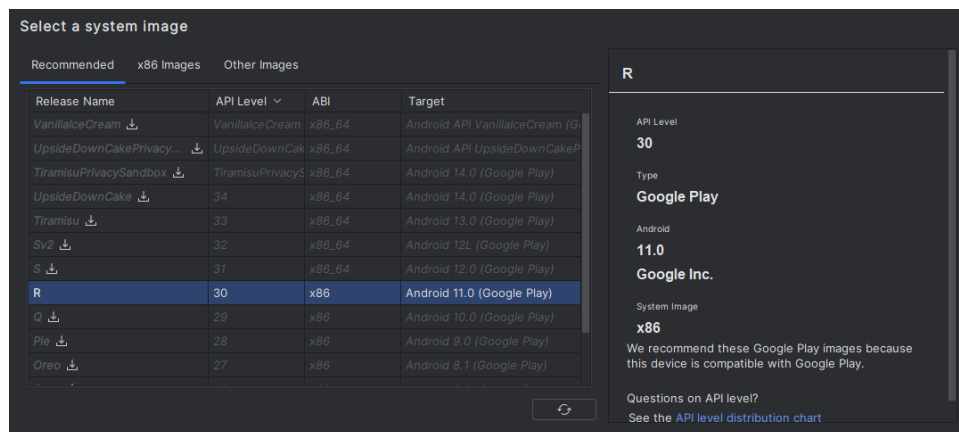
**Figura 8.** Página oficial de descarga de Android Studio. Elaboración Propia.

Seguidamente, pasaremos a la configuración del Android Studio, una vez descargado e instalado el programa, lo abrimos y vamos a “More Actions” y a “Virtual Device Manager” como se muestra en la siguiente figura.



**Figura 9.** Primeros pasos de configuración de Android Studio. Elaboración Propia.

Seguidamente, presionamos el botón “Create Device” ubicado en la esquina superior izquierda, luego elegimos el modelo de teléfono “Pixel 7” o “Pixel 7 Pro” (se recomienda el “Pixel 7 Pro”). Le damos a “Next” y escogemos la opción “R” como se muestra en la siguiente figura:



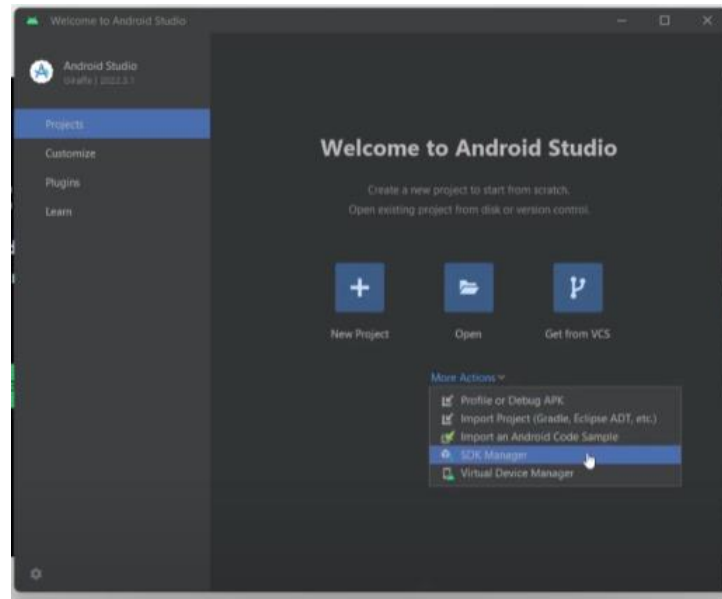
**Figura 10.** Selección de versión de Android. Elaboración Propia.

Una vez descargada, presionamos “Finish”, la seleccionamos, luego presionamos en “Next” y “Finish”. Y habremos instalado correctamente el simulador



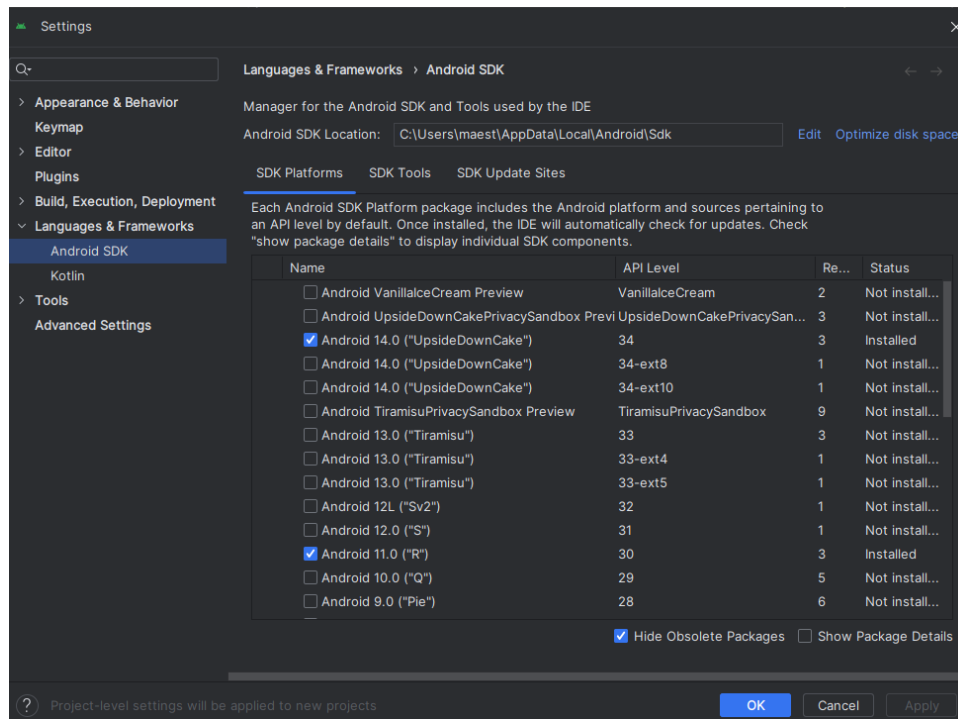
de teléfono Android, el cuál dura cargando, dependiendo de las especificaciones del dispositivo en el cual lo estés simulando.

Ahora, para asegurar su correcto funcionamiento, tenemos que validar unas configuraciones, iremos a “Device Manager” y luego a “SDK Manager” como se muestra en la figura



**Figura 11.** Pasos de verificación de archivo. Elaboración Propia.

En la pestaña de “SDK Plataforms”, tenemos que ver seleccionada nuestra versión de Android instalada anteriormente (se descargó la versión “R”). Si hay alguna otra versión descargada y seleccionada, ignorarla ya que no nos afectará en el uso del programa.

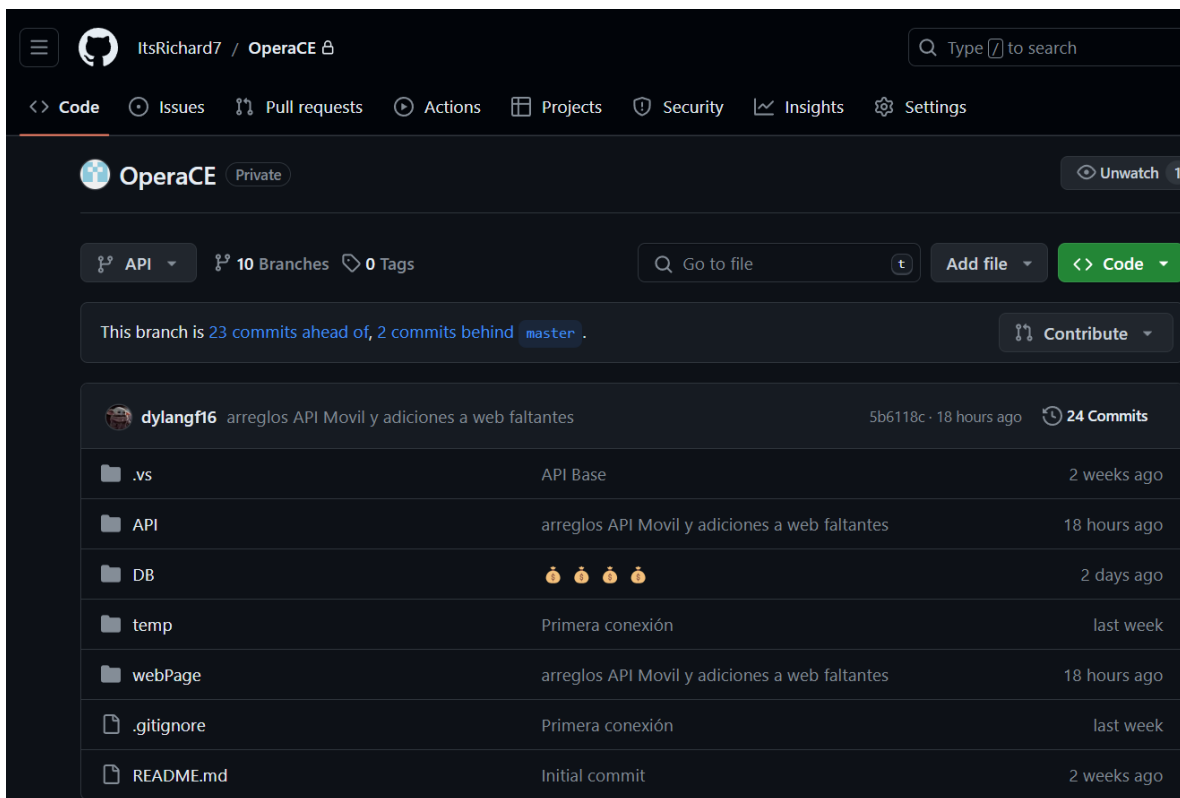


**Figura 12.** Selección de SDK. Elaboración Propia.

## c) Instalación de la página Web

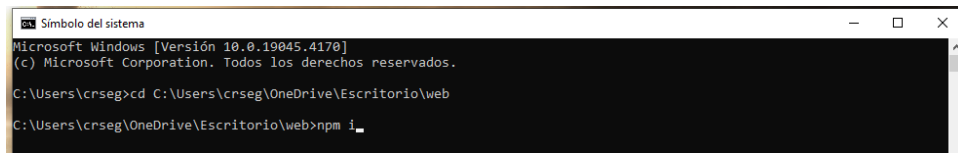
Inicialmente, se tiene que contar con versiones de NPM mayores a 10.2.4 y versiones de node.js mayores a 20.11.1. Una vez hecho esas instalaciones o actualizaciones, se procede a ir al siguiente [repositorio](#) de GitHub

Al estar en esta página, se descarga el .zip y se obtiene el código de la carpeta “webPage” que contiene la información que requerimos para esta sección:



**Figura 13.** Descarga del código de la página referencia en el sitio web Github.  
Elaboración Propia.

Una vez con la carpeta .zip, se descomprime, y se abre en la terminal la carpeta webPage y se coloca el comando npm install, para agregar node modules. Se coloca un ejemplo de cómo se debe de ver.



```
Símbolo del sistema
Microsoft Windows [Versión 10.0.19045.4170]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\crseg>cd C:\Users\crseg\OneDrive\Escritorio\web
C:\Users\crseg\OneDrive\Escritorio\web>npm i
```

**Figura 2.** Comando para agregar la carpeta node\_modules. Elaboración Propia.

Una vez todo instalado, la carpeta debe de verse de la siguiente manera:

node_modules	✖	10/3/2024 13:18	Carpeta de archivos	
public	✔	9/3/2024 13:00	Carpeta de archivos	
src	✔	10/3/2024 10:07	Carpeta de archivos	
.gitignore	✔	9/3/2024 12:59	Archivo de origen ...	1 KB
package	✔	10/3/2024 13:18	Archivo de origen ...	1 KB
package-lock	✔	10/3/2024 13:18	Archivo de origen ...	717 KB
README	✔	9/3/2024 12:59	Archivo de origen ...	4 KB

**Figura 3.** Vista de la carpeta con todos sus componentes. Elaboración Propia.

Finalmente, se abre la carpeta en su IDE de preferencia, en el caso de los autores, se recomienda el uso de visual studio code. Lo importante es poder tener consolas para hacer comandos de descarga y de correr la aplicación.

Para el correcto funcionamiento de la aplicación, se necesitan hacer estos comandos:

- npm i react-data-table-component
- npm i react-bootstrap
- npm i react-router-dom
- npm i react-icons

Finalmente, en la misma consola se corre el comando: **npm start**. Y mostrara la página de login de aplicación web.