

Holaaaaç

[!@#\$\$%^&\*()\_+{}|:"<>?`~;,.\/"\\-]

# DOCKER

## INSTALACIÓN DE DOCKER

sudo apt update

sudo apt install apt-transport-https ca-certificates curl software-properties-common git -y

(pedro) sudo mkdir -p /etc/apt/keyrings

sudo install -m 0755 -d /etc/apt/keyrings

sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o  
/etc/apt/keyrings/docker.asc

sudo chmod a+r /etc/apt/keyrings/docker.asc

echo \

"deb [arch=\$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc]

https://download.docker.com/linux/ubuntu \

\$(. /etc/os-release && echo "\$VERSION\_CODENAME") stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo reboot

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin

\$(A carlos le va al resto no (mentira cochina))docker-compose-plugin -y

sudo docker version

## CREACIÓN CONTENEDORES DE DOCKER

### Estructura

```
administrador@administrador-VirtualBox:~/prueba_examen$ tree
.
├── app
│   ├── index.php
│   ├── docker-app-php
│   └── memoria.md
└── 1 directory, 3 files
administrador@administrador-VirtualBox:~/prueba_examen$
```

### Crear dockerfile php

```
FROM php:8.2-apache
COPY app/ /var/www/html/
EXPOSE 80
```

### Crear dockerfile ubuntu

```
FROM ubuntu:22.04
```

```
#Para que no pida ninguna confirmación al usuario
ENV DEBIAN_FRONTEND=nointeractive
```

```
RUN apt-get update && apt-get install -y \
apache2 php git curl && \
rm -rf /var/lib/apt/lists/*
```

```
RUN rm -rf /var/www/html*
```

```
RUN git clone https://github.com/heroku/php-getting-started.git /var/www/html/
```

```
EXPOSE 80
```

```
CMD ["apachectl", "-D", "FOREGROUND"]
```

## **CONSTRUIR APLICACIÓN**

```
sudo docker build -t nombre_imagen -f dockerfile .
```

## **CREAR CONTENEDOR**

```
sudo docker run -d --name nombre_contendor -p 8080:80 nombre_imagen
```

## **BORRAR Y PARAR CONTENEDOR**

```
sudo docker stop nombre_contendor
```

```
sudo docker rm nombre_contendor
```

## **Script crear contenedores docker con imagen**

```
#!/bin/bash
for i in {1..20}; do
    port=$((8080+i))
    docker run -d --name ubuntu-app-$i -p $port:80 ubuntu-app
done
```

## **Script eliminar contenedores docker**

```
#!/bin/bash
for i in {1..20}; do
    docker stop ubuntu-app-$i && docker rm ubuntu-app-$i
done
```

# **GIT**

## **Iniciar git**

git init

## **Añadir archivos**

git add .

## **Usuario y correo**

git config --global user.name "nombre:usuario"  
git config --global user.email "correo@gmail.com"

## **Crear commit**

git commit -m "Nota del commit"

## **Establecer ruta de subida**

git branch -M main

## **Conectar con el repositorio de github**

git remote add origin ruta\_github.git  
git remote -v

## **Pushear archivos**

git push -u origin main

## **Añadir nuevos archivos**

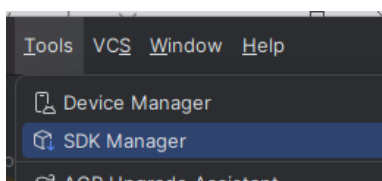
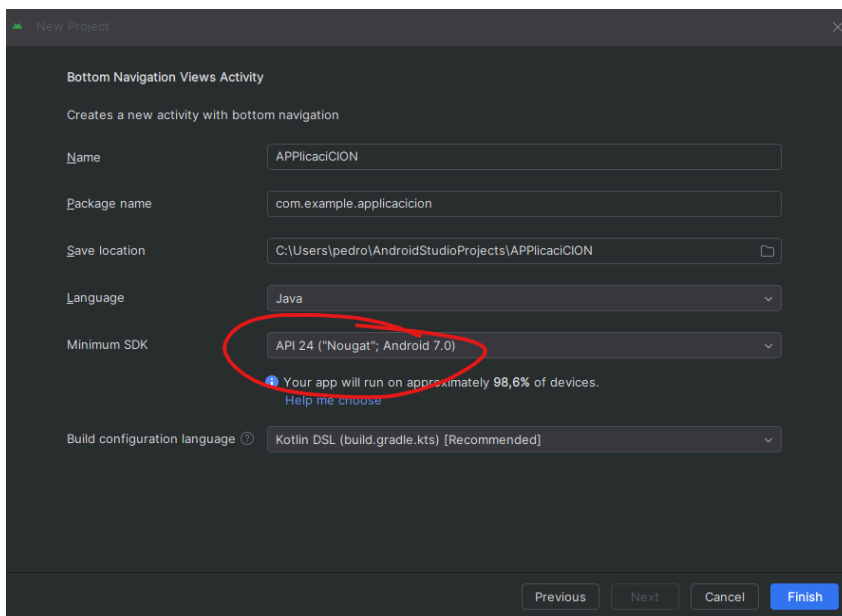
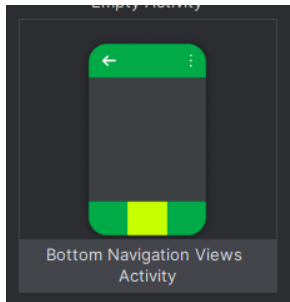
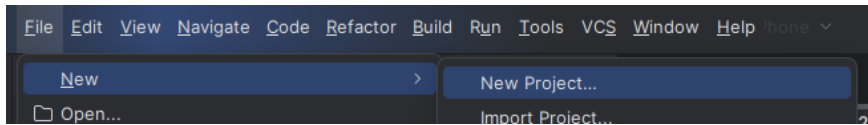
nano loquesea  
nano patata.txt

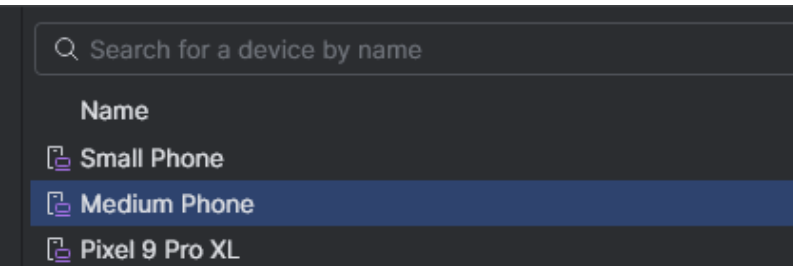
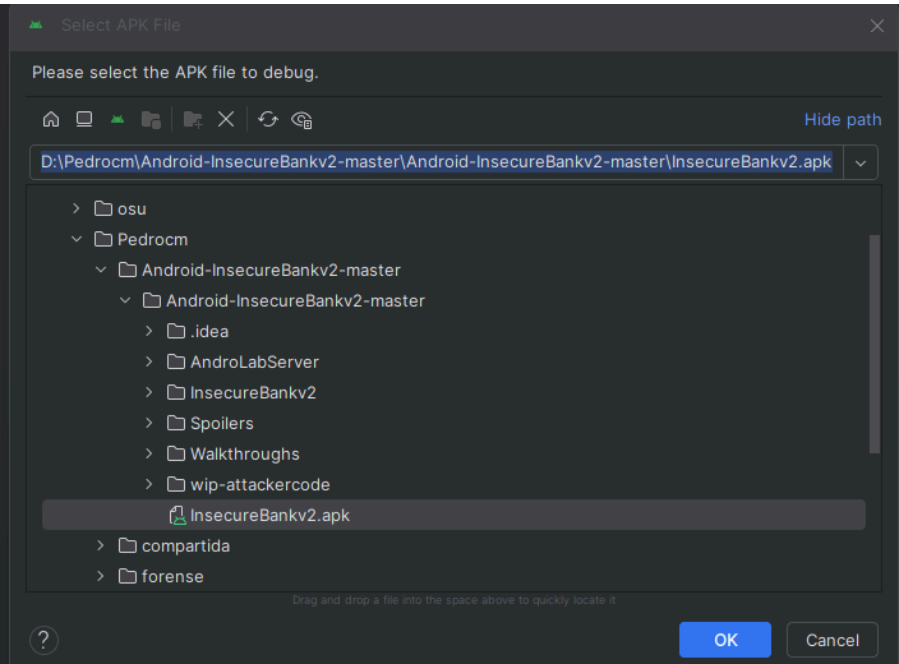
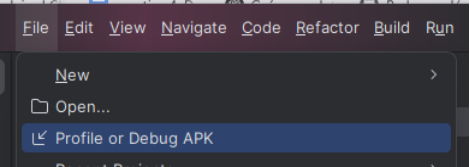
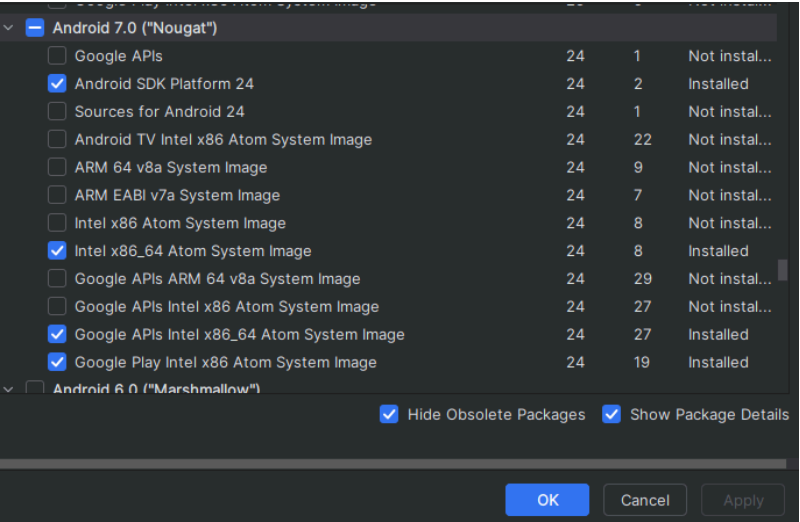
git add .  
git commit -m "nuevo commit con cambios"

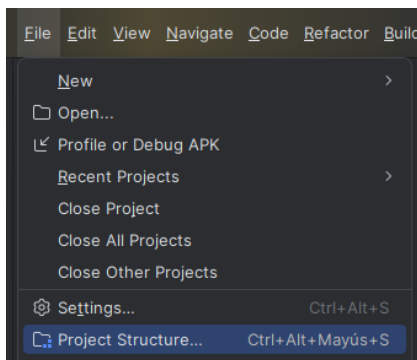
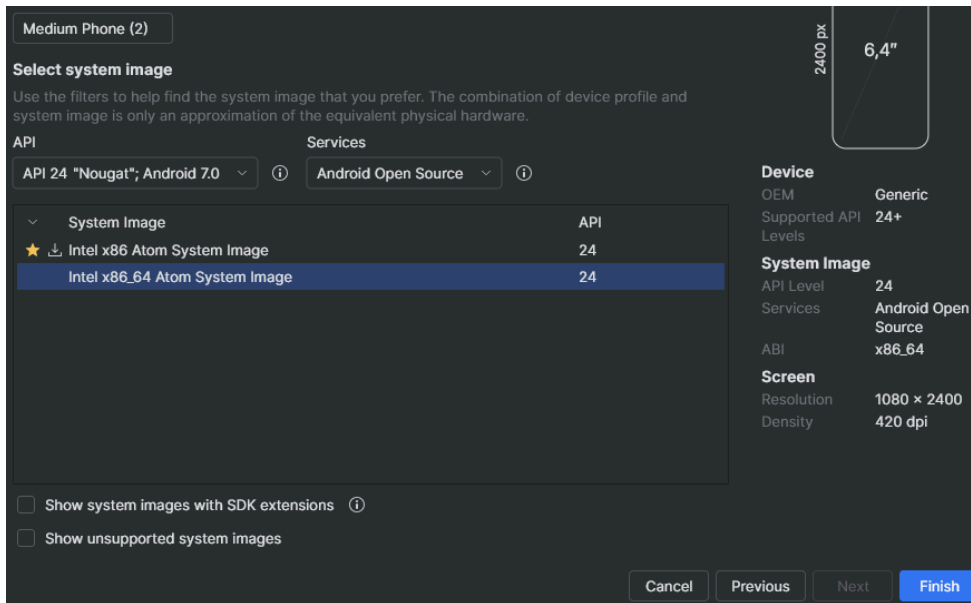
git push

# FORENSE MÓVILES

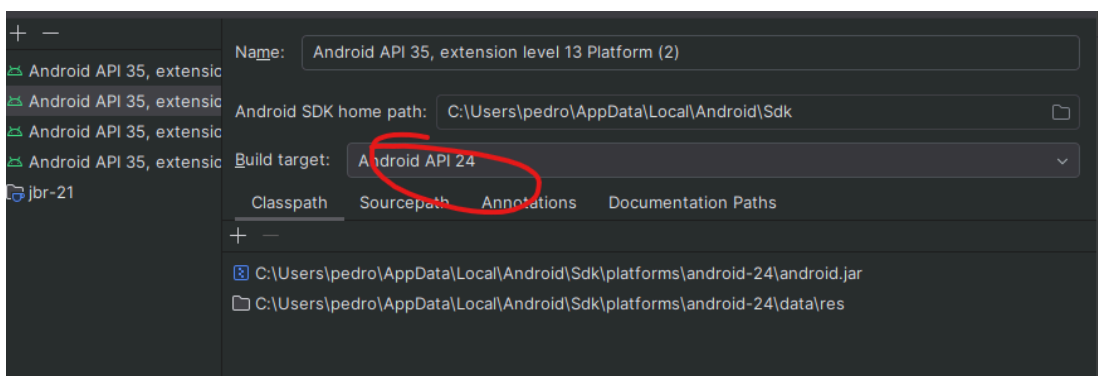
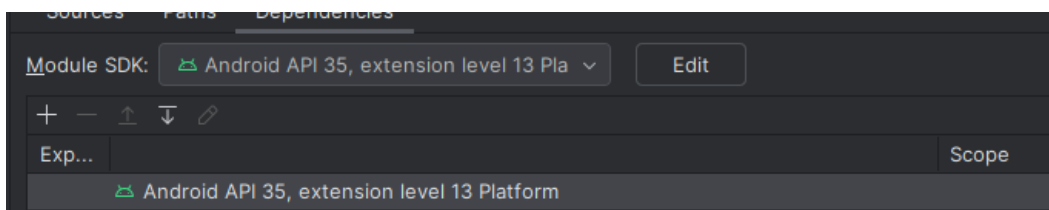
## ANDROID STUDIO







dentro de modules



## **APKTOOL**

# copiarlo antes en otro sitio para desempaquetar

```
apktool d aplicacion.apk  
getid AndroidManifest.xml
```

## **Qark**

# en la carpeta en la que se encuentra el apk  
qark --apk InsecureBankv2.apk

## **Bytecode Viewer**

```
# usuario@pps:~/Descargas  
java -jar Bytecode-Viewer-2.12.jar
```

```
Archivo > Añadir... > aplicacion.apk  
com > android > Insecurebankv2
```