

1. 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
```

1. 1 Comandos básicos de docker

Ver imágenes:

```
sudo docker images
```

Eliminar imágenes:

```
sudo docker rmi numero
```

Construir imágenes a partir de un Dockerfile del directorio actual:

```
sudo docker build -t esto_es_la_tag/nombre_de_la_imagen .
```

Crear contenedor y lanzar (-d instancia separada) (-p puertolocal/puertocontenedor):

```
sudo docker run -d -p 8777:8080 --name cont-ej1 ej1
```

Ver contenedores:

```
sudo docker ps -a
```

Para contenedor:

```
sudo docker stop nombre-cont
```

Iniciar contenedor una vez creado:
`sudo docker start nombre-cont`

Eliminar contenedor
`sudo docker rm nombre-cont`

Ver logs para depurar fallos
`sudo docker logs nombre-cont`

Ej1

Crear estructura de directorios

```
php-app/  
├── Dockerfile  
└── app/  
    └── index.php
```

Contenido del index.php

```
<?php  
echo "Hola desde contenedor PHP!";  
?>
```

contenido del Dockerfile

```
# Copiar imagen con base php y servidor embebido
FROM php:8.2-cli

# Dentro de la imagen crear el directorio de la aplicación
WORKDIR /usr/src/miapp

# Copiar aplicación desde el host hacia la app
COPY aplicacion/ /usr/src/miapp

# Comando para ejecutar la aplicación
CMD ["php", "-S", "0.0.0.0:8080", "-t", "/usr/src/miapp"]
```

docker build -t ej-1 .

docker run -d -p 8777:8080 --name cont-ej1 ej1

Ej2

```
# imagen base
FROM ubuntu:22.04

# evitar instalacion interactiva
ENV DEBIAN_FRONTEND=noninteractive

# instalar php y apache
RUN apt-get update && apt-get install -y apache2 php git curl && apt-get clean

# Activar apache
RUN echo "ServerName localhost" >> /etc/apache2/apache2.conf

# copiar aplicacion simple
RUN rm -rf /var/www/html/*
RUN git clone https://github.com/heroku/php-getting-started.git /var/www/html/

# Exponer puerto 80
EXPOSE 80

# Iniciar apache en segundo plano
CMD ["/usr/sbin/apache2ctl", "-D", "FOREGROUND"]
```

Ej 3

Script iniciar

```
#!/bin/bash
for i in {1..3}
do
    puerto=$((8000 + $i))
    docker run -d -p ${puerto}:80 --name cont-ej3-$i ej2
done
```

sudo ./ script-iniciar-cont.sh

Script parar

```
#!/bin/bash
for i in {1..3}
do
    docker stop cont-ej3-$i
    docker rm cont-ej3-$i
done
```

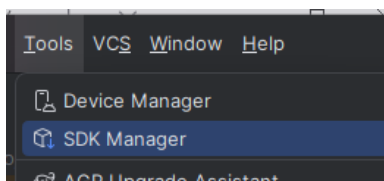
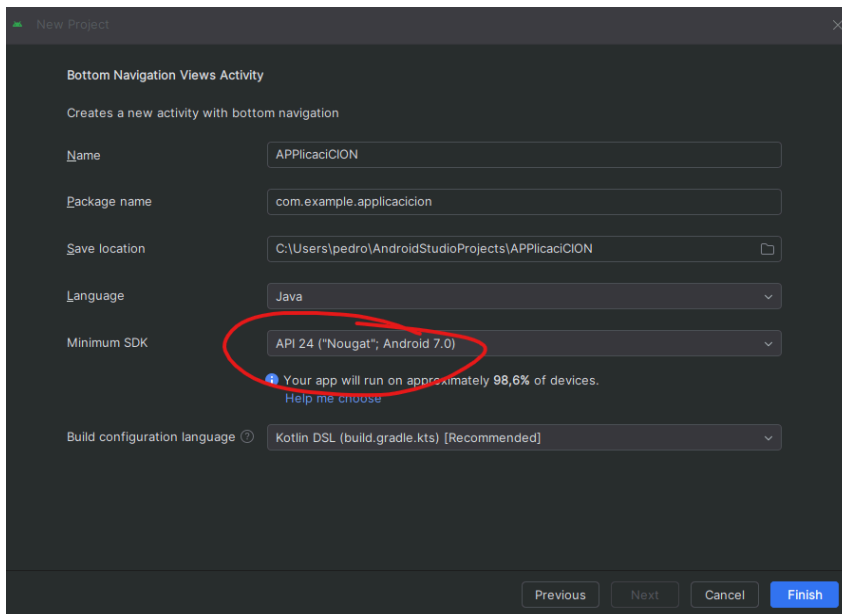
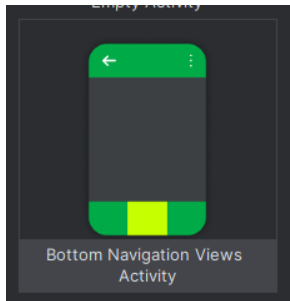
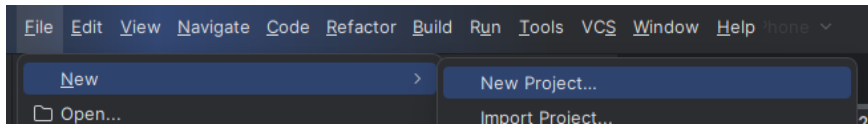
sudo ./ script-parar-cont.sh

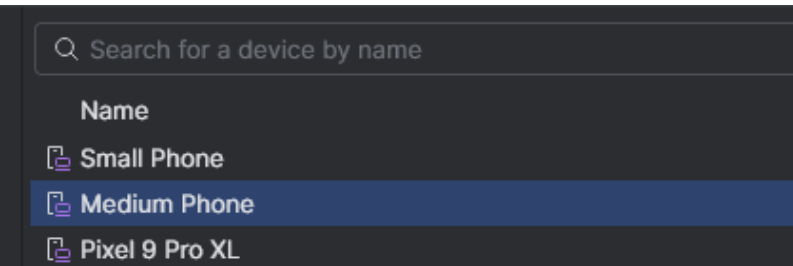
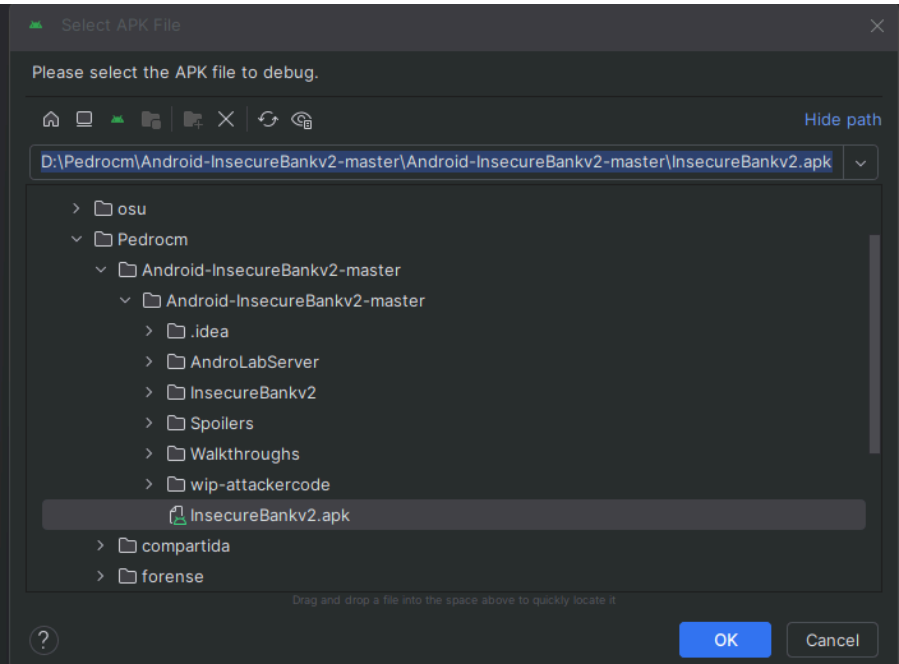
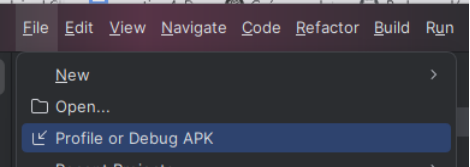
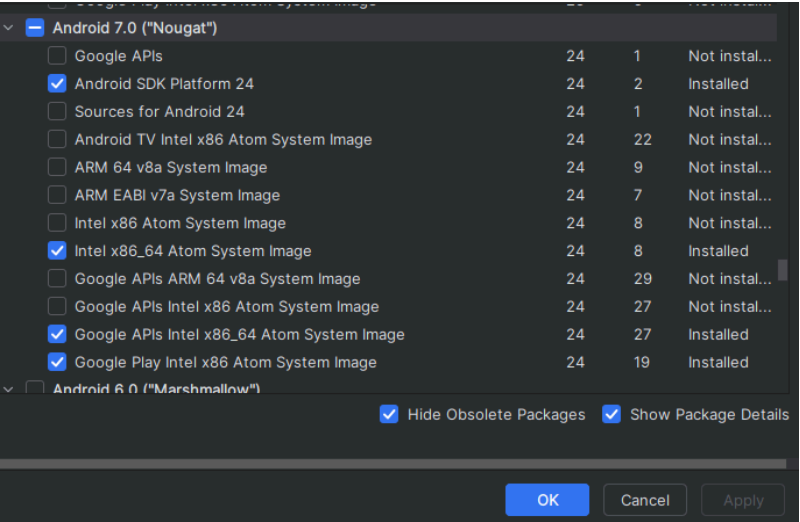
2. GIT

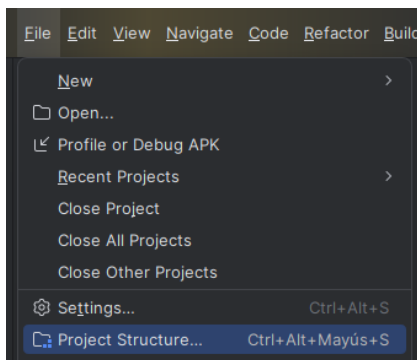
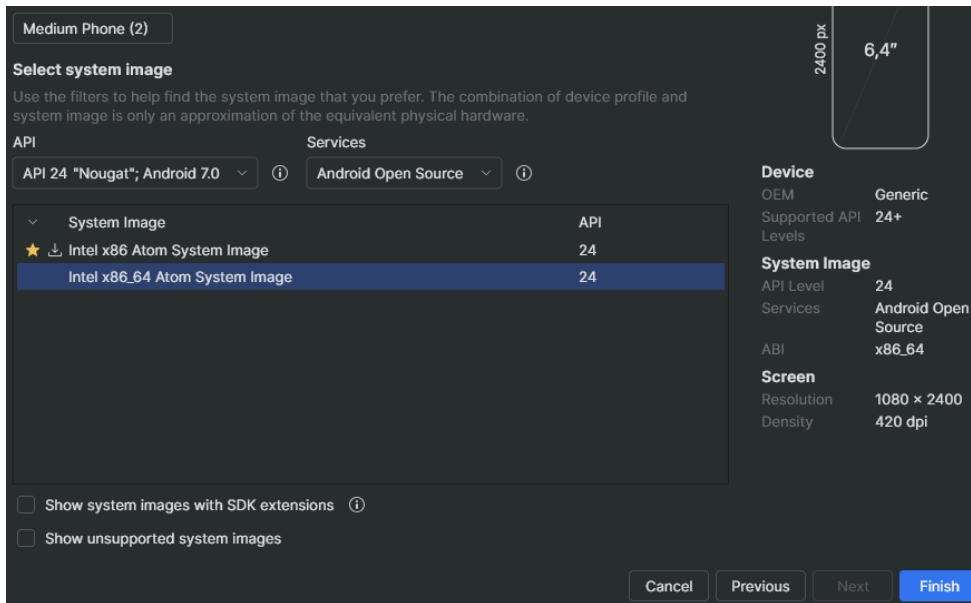
```
sudo apt update
sudo apt install git -y
git config --global user.name "PedroCasal"
git config --global user.email "pedrocasal17@gmail.com"
git config --global --list
sudo apt install ssh -y
ssh-keygen -t ed25519 -C "pedrocasal@git"
eval "$(ssh-agent -s)"
ssh-add ~/.ssh/id_ed25519
cat ~/.ssh/id_ed25519.pub
# Añadir el resultado del cat a las claves ssh de github/Settings -> ssh and gpg keys/
git init
# Crear ficheros
echo "121212" > README.md
#revisar estatus
git status
# añadir todo
git add .
# Hacer commit
git commit -m "COMENTARIO"
git remote add origin git@github.com:Pedrocasal/repaso-git
git remote -v
git branch -M main
ssh -T git@github.com
git push -u origin main
```

móvil

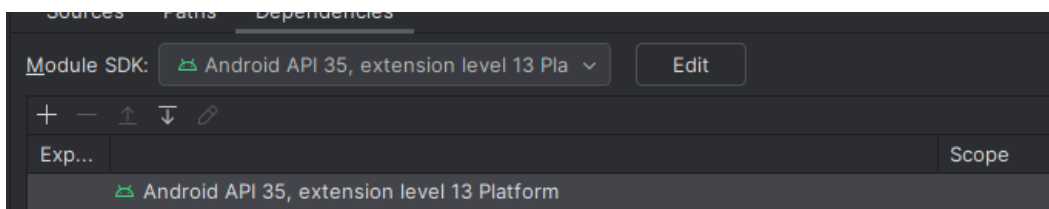
ANDROID STUDIO



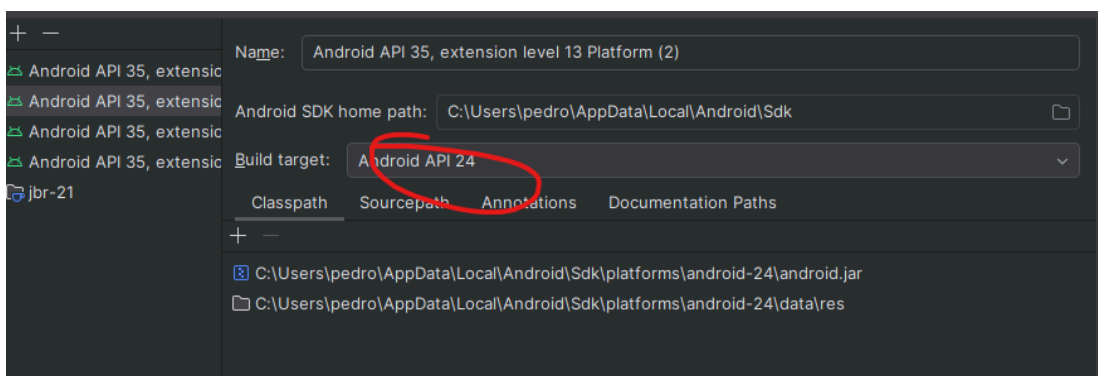




Dentro de modules



En SDKs



/home/usuario/Android-InsecureBankv2/AndroLabServer

./instalar_ejecutar_servidor_v3.sh

Entrar a la ip desde el móvil

/home/usuario/Android-InsecureBankv2

qark --apk InsecureBankv2.apk

/home/usuario/.local/lib/python2.7/site-packages/qark/report/report.html

Copia del apk

apktool d InsecureBankv2.apk

nano /home/usuario/Documentos/InsecureBankv2/AndroidManifest.xml

cd Descargas

java -jar Bytecode-Viewer-2.12.jar

-> archivo -> añadir -> escoger apk

com > android > insecure Bank > están las clases