

Solucionario Prueba - Desarrollo de la interfaz de usuario Android

Requerimientos

1. Crear o utilizar algún proyecto existente con Android Studio, para luego crear repositorio en GitHub.

The screenshot shows the GitHub 'Create a new repository' interface. It includes fields for 'Owner' (with a user profile icon) and 'Repository name'. A hint suggests repository names should be short and memorable, with an example 'verbose-potato?'. There is an optional 'Description' text area. Under 'Visibility', the 'Public' option is selected, with a note that anyone on the internet can see the repository. The 'Private' option is also available. The 'Initialize this repository with:' section has a checkbox for 'Add a README file'. Below this is the 'Add .gitignore' section with a dropdown menu set to 'None'. The 'Choose a license' section also has a dropdown menu set to 'None'. A note at the bottom states 'You are creating a public repository in your personal account.' A green 'Create repository' button is at the bottom.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * Repository name *

Great repository names are short and memorable. Need inspiration? How about [verbose-potato?](#)

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None ▾

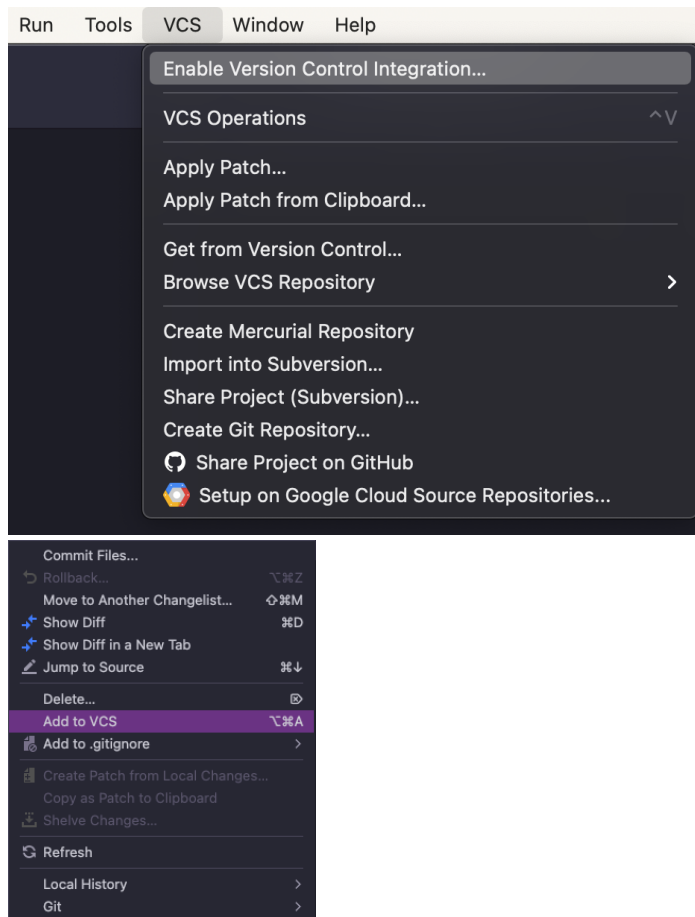
Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

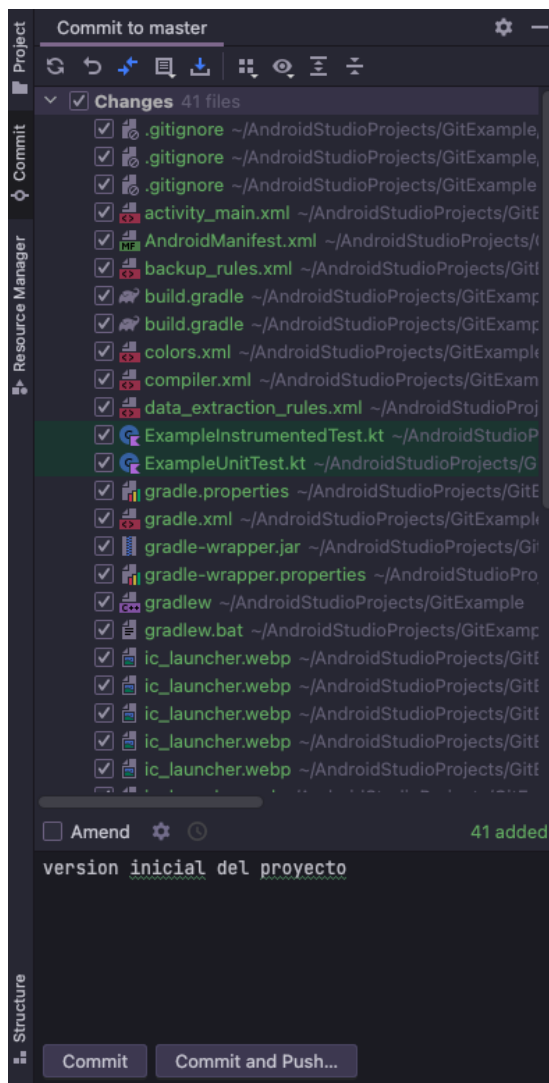
License: None ▾

i You are creating a public repository in your personal account.

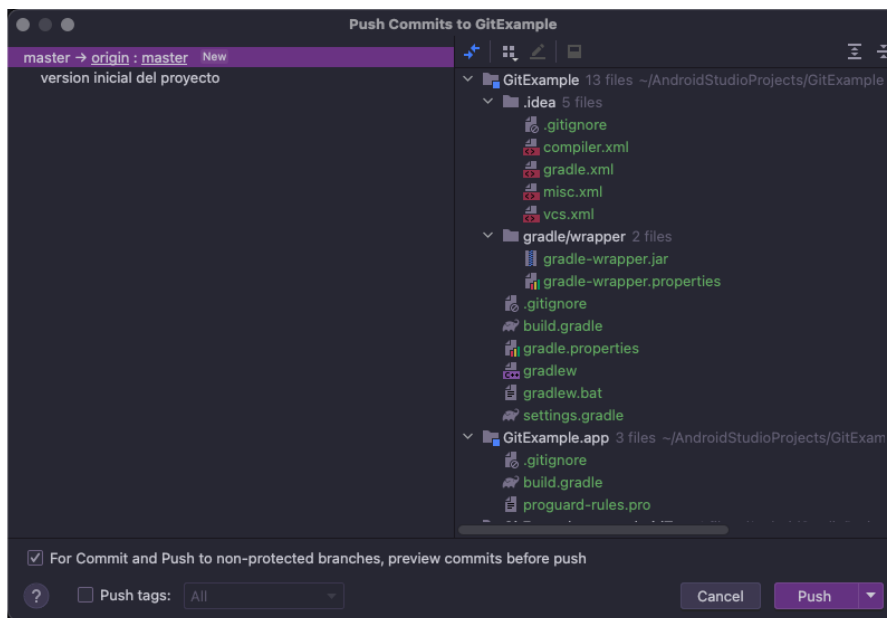
Create repository

2. Inicializar el sistema de control de versiones con Git, y después agregar y confirmar los cambios.





3. Configurar el remoto y empujar los cambios si IDE ya tiene la configuración de GitHub, en caso contrario, agregar el token necesario.



4. Crear la rama **feature1**, para luego hacer un cambio en un archivo de la rama **feature1**, guardar los cambios (commit), y empujar la rama **feature1** al remoto.

```
git checkout -b feature1
vim app/build.gradle (-> hacer un cambio y guardar)
git add app/build.gradle
git commit -m "modificación en la configuración"
git push origin feature1
```

5. Crear PR para mezclar los cambios de **feature1** a master (desde Android Studio o desde consola).

