



<codoo  
codoo/>



Maurisandev



@MauriDeveloper

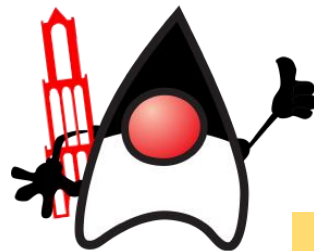


mau.sanchez@bue.edu.ar



# Curso Java FullStack

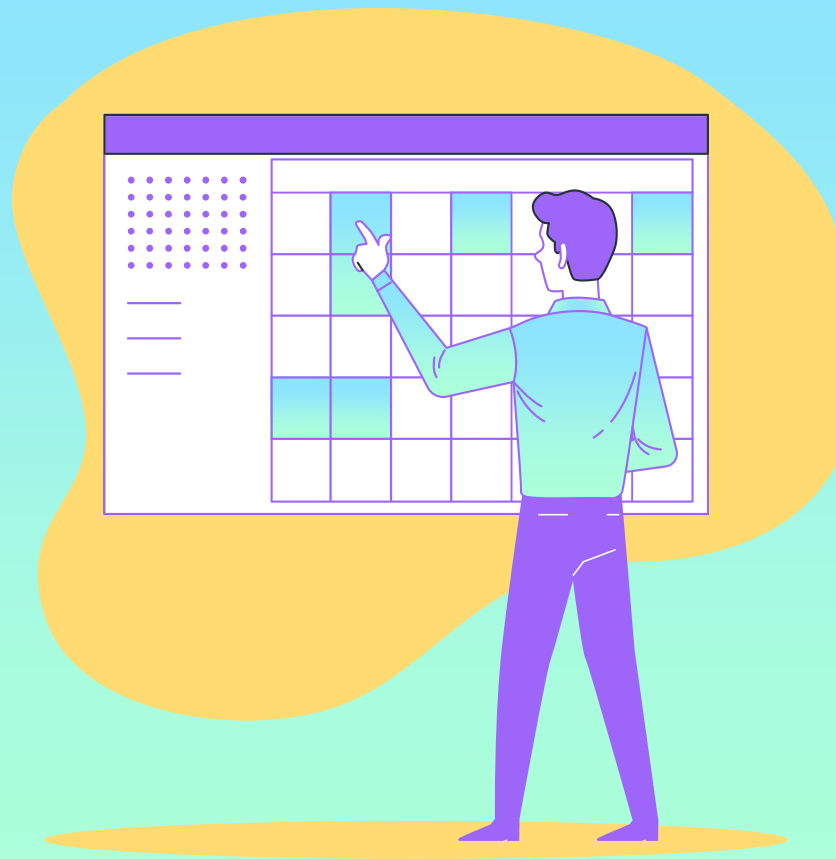
Codo a Codo 4.0  
Clase-21 REFRESH  
SCRUM-GIT





<codoa  
codo/>

# Metodologías Agiles



Maurisandev



@MauriDeveloper



maurisan4011@gmail.com

- **metodologías ágiles** son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno.
- En esencia, las empresas que apuestan por esta metodología consiguen gestionar sus proyectos de forma flexible, autónoma y eficaz reduciendo los costes e incrementando su productividad.





\*\*\*\*\*

MARCO DE  
TRABAJO!

< SCRUM >



<codoo  
codoo/>



# <SCRUM>



Scrum es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio de la manera de trabajar de equipos altamente productivos.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto.



Maurisandev

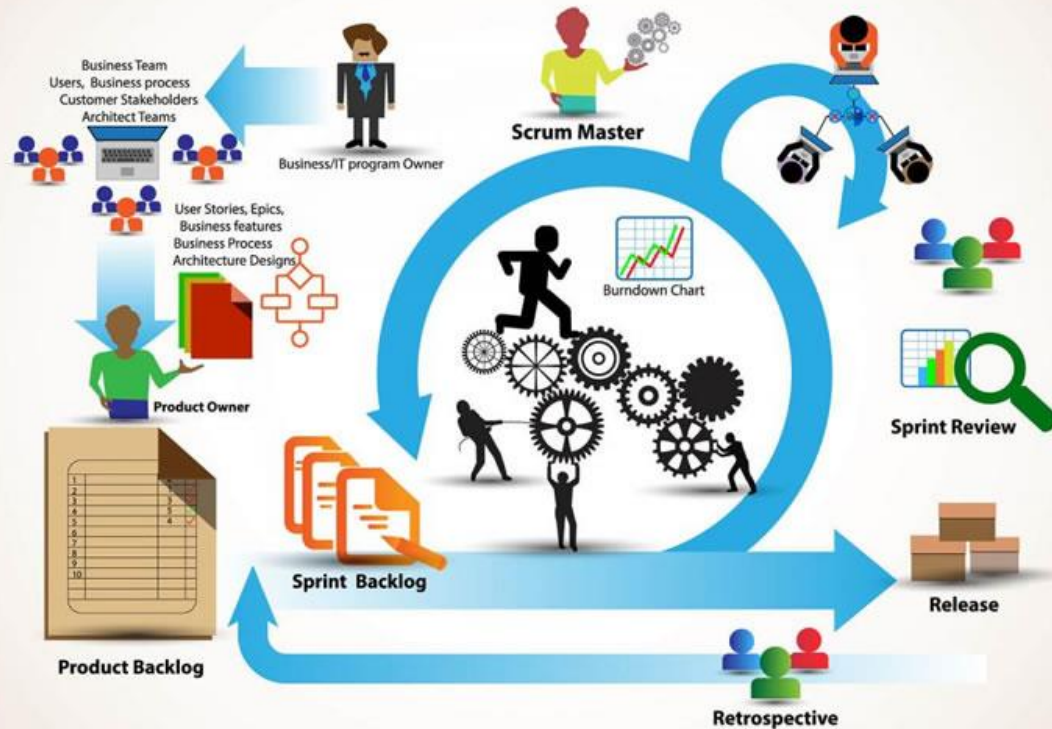


@MauriDeveloper



maurisan4011@gmail.com

## Agile Methodology - Scrum Process



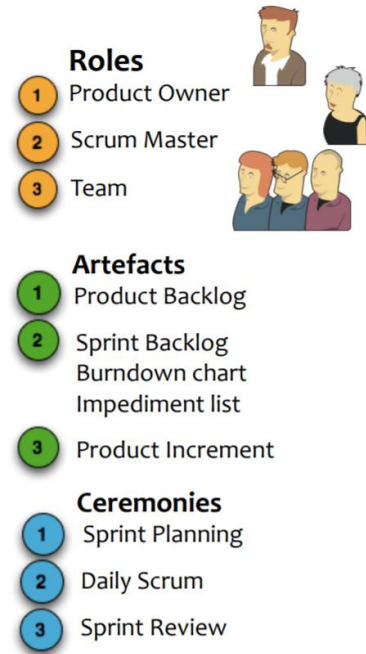
\*\*\*\*\*

**MARCO DE  
TRABAJO!**

**< SCRUM >**



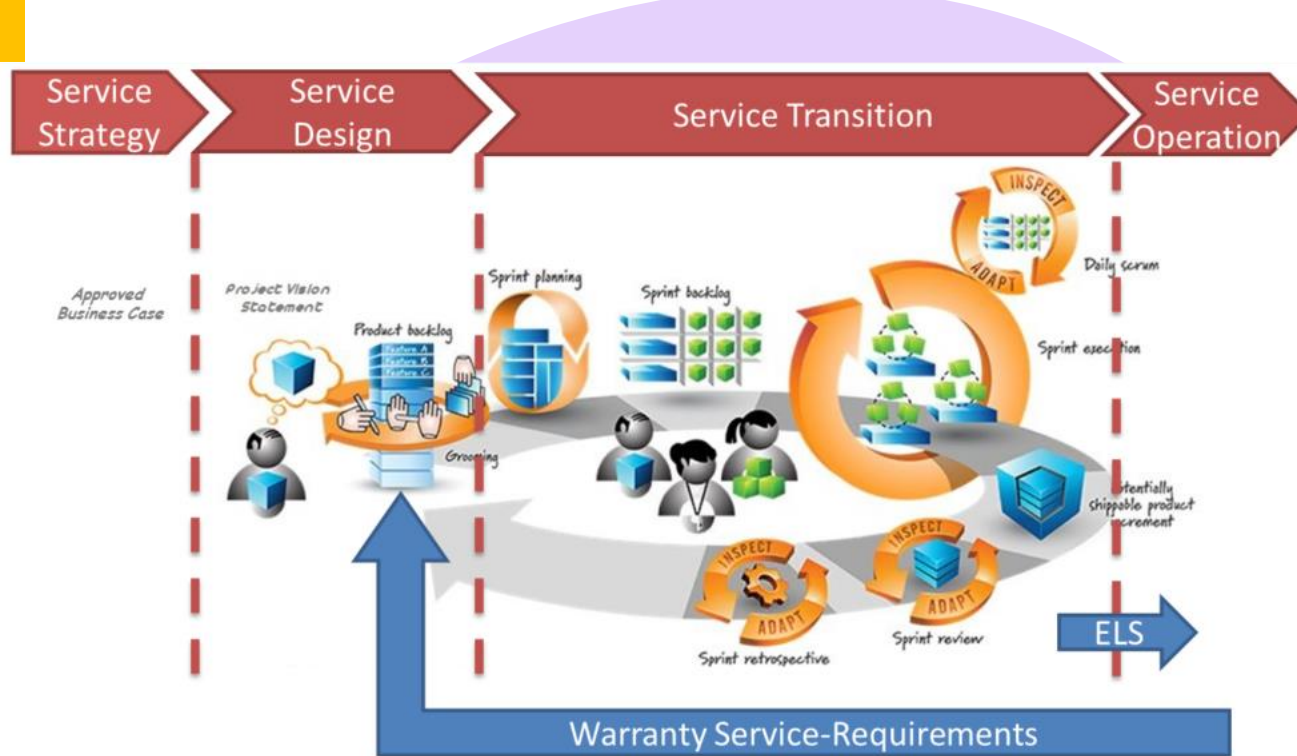
1



# MARCO DE TRABAJO!







\*\*\*\*\*

**MARCO DE TRABAJO!**

**< SCRUM >**

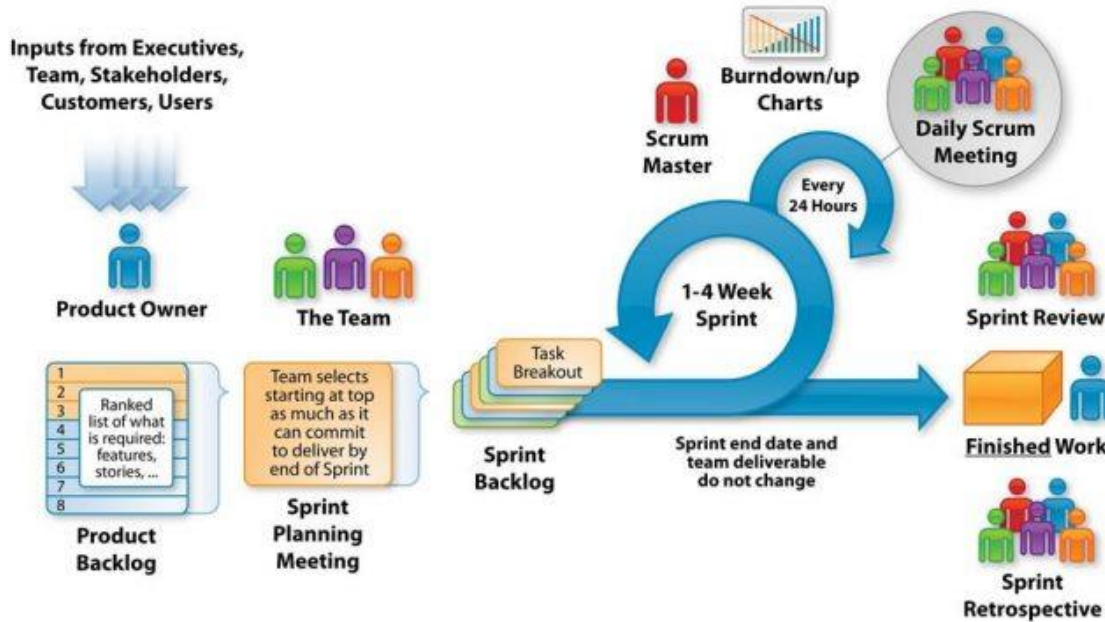


`<codigo  
codigo/>`





# The Agile - Scrum Framework



\*\*\*\*\*

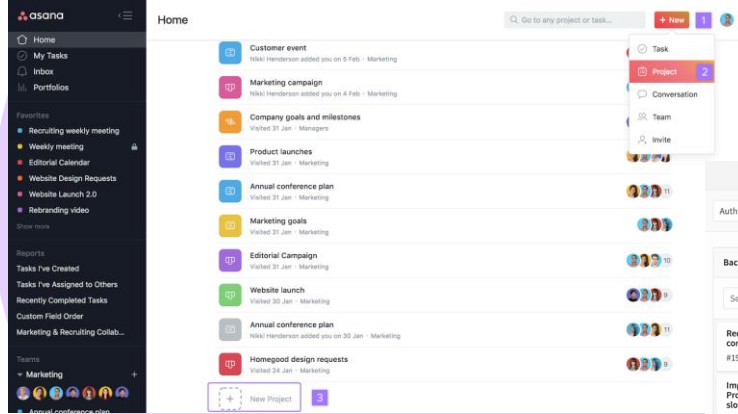
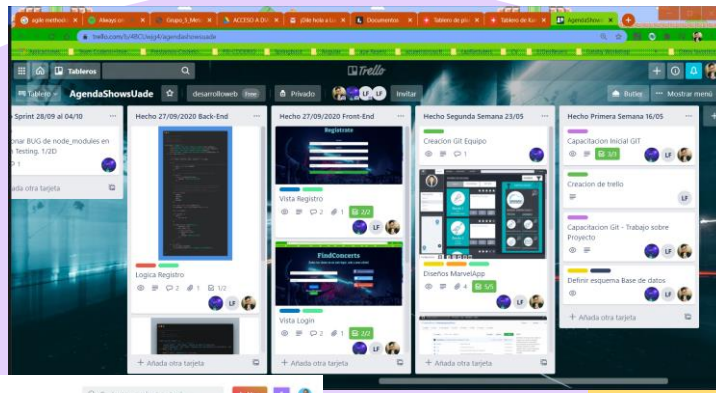
MARCO DE  
TRABAJO!

< SCRUM >



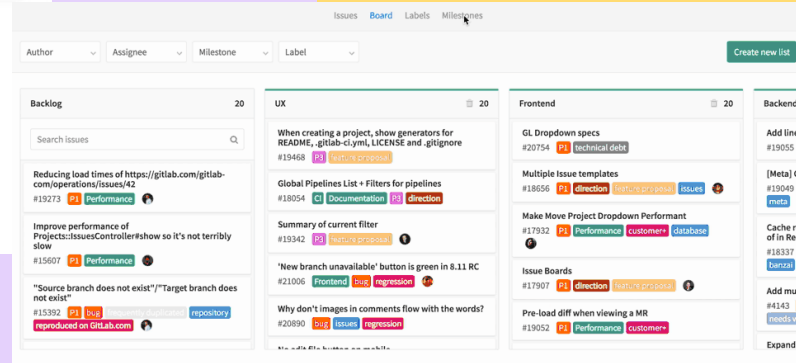
<codoa  
codoa/>





< SCRUM >

\*\*\*\*\*  
AL TABLERO !



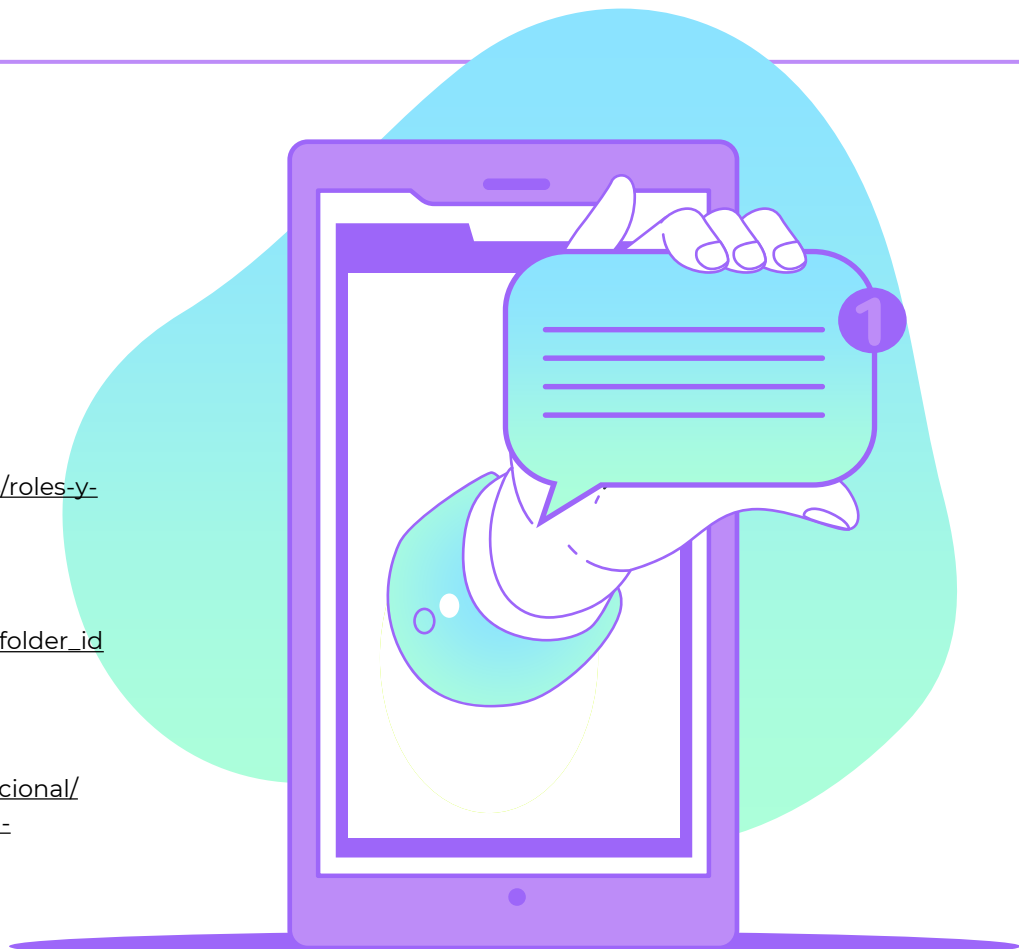
# Links para Reforzar

## Scrum Metodologias Agiles

- <https://integrait.com.mx/blog/roles-de-scrum/>
- <https://www2.deloitte.com/es/es/pages/technology/articles/roles-y-responsabilidades-scrum.html>
- <https://www.projectmanager.com/blog/guide-to-scrum-ceremonies>
- [https://lucid.app/lucidspark/2686cecc-13aa-42f9-9018-77336740b0f3/edit?beaconFlowId=3B26DDDC2B54832#?folder\\_id=home&browser=icon](https://lucid.app/lucidspark/2686cecc-13aa-42f9-9018-77336740b0f3/edit?beaconFlowId=3B26DDDC2B54832#?folder_id=home&browser=icon)

## Pensamiento computacional

- <https://codelearn.es/beneficios-del-pensamiento-computacional/>
- <https://www.ceibal.edu.uy/es/articulo/que-aporta-al-aula-el-pensamiento-computacional>





```
<code>  
code  
</code>
```



**GitHub**



**GitLab**



Maurisandev



@MauriDeveloper



maurisan4011@gmail.com



# ¿Qué es Git?



Git, es un software de control de versiones diseñado por Linus Torvalds. La pregunta es ¿qué es control de versiones? Pues bien, se define como control de versiones a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración del mismo es decir a la gestión de los diversos cambios que se realizan sobre los elementos de algún producto o una configuración, y para los que aún no les queda claro del todo, control de versiones es lo que se hace al momento de estar desarrollando un software o una página web. Exactamente es eso que haces cuando subes y actualizas tu código en la nube, o le añades alguna parte o simplemente le editas cosas que no funcionan como deberían o al menos no como tú esperarías.



Maurisandev



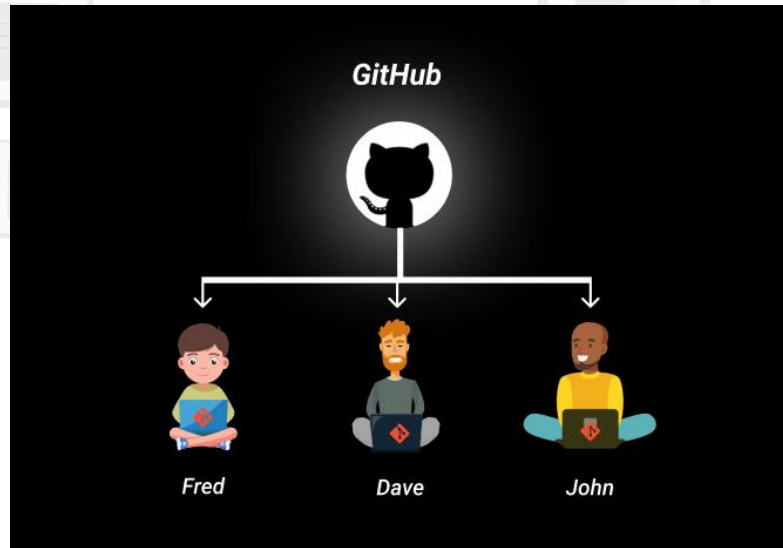
@MauriDeveloper



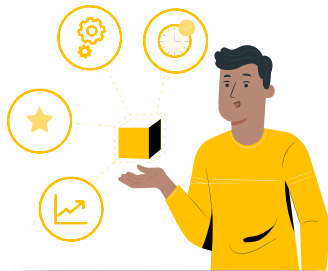
maurisan4011@gmail.com



Y, entonces ¿a que le llamamos sistema de control de versiones? Muy sencillo, son todas las herramientas que nos permiten hacer todas esas modificaciones antes mencionadas en nuestro código y hacen que sea más fácil la administración de las distintas versiones de cada producto desarrollado; es decir Git.







- Control de versiones distribuido.
- Manejar distintas versiones del proyecto.
- Guardar el historia o se guardan todas las versiones de todos los archivos del proyecto.
- Trabajar simultáneamente sobre un proyecto

**TUTORIAL:** <https://try.github.io/levels/1/challenges/1>



# GitHub



```
#Git configuration
git config --global user.name "Your Name"
git config --global user.email "you@gmail.com"
```

```
#Starting A Project
git init [project name]
git clone [project url]
```

```
#Remove file from directory
git rm [file]
```

```
#Status of working directory
git status
```

```
#Add a file to the staging area
git add [file]
```

```
#Discard changes in working directory
git checkout -- [file]
```

## GIT CHEATSHEET

```
#Commit to local
git commit
```

```
#Revert your repository
git reset [file]
```

```
#List all local branches
git branch [-a]
```

```
#Fetch changes from the remote and merge current branch with its upstream
git pull [remote]
```

```
#Join specified [from name] branch
git merge [from name]
```

```
#Create new branch
git branch [branch_name]
```

```
#Push local changes to the remote
git push [--tags] [remote]
```

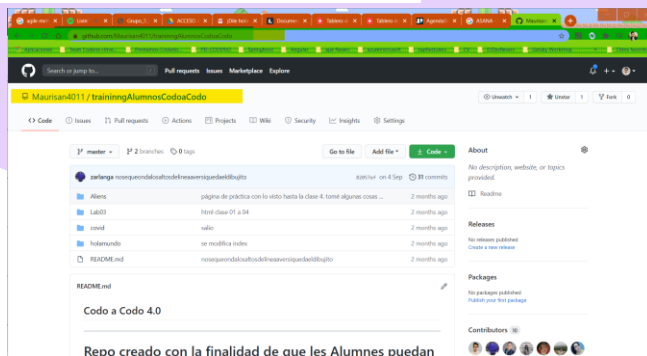
```
#Remove selected branch
git branch -d [name]
```

```
#Fetch changes from the remote
git fetch [remote]
```

```
#Switch current branch to specified branch
git checkout [-b][branch_name]
```



# Vamos a nuestro repo, creamos un Branch con nuestro apellido y subamos algún Proyecto



Repo creado con la finalidad de que les Alumnos puedan practicar sentencias de Git.

OJO con lo que subimos el uso es de practica y publico!!!

Alumnos Java FullStack comision 2011

#ALUMNO: DIAZ

Matias Gutierrez dice:

**FACILITO**

ANDRES ACOSTA DICE:





Maurisandev



@MauriDeveloper



mau.sanchez@bue.edu.ar



*MUCHAS GRACIAS Y QUE LO DISFRUNTEN !!*



\*\*\*\*\*

**NOS VEMOS EL  
jueves!!**

