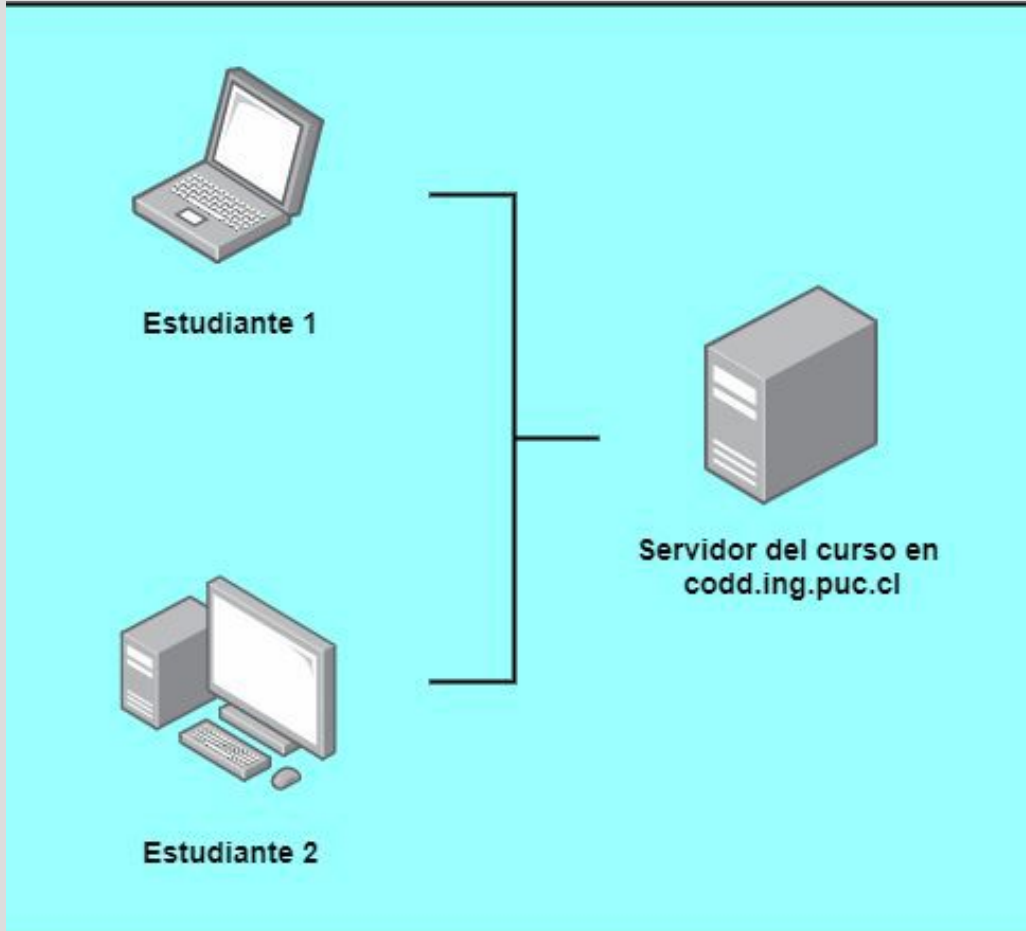




USO DEL SERVIDOR

Ayudantía 2
IIC2413 (2022- 1)
25 de marzo, 2022

Servidor



- Computador que siempre está prendido en el DCC
- TODAS las entregas se harán en el servidor del curso

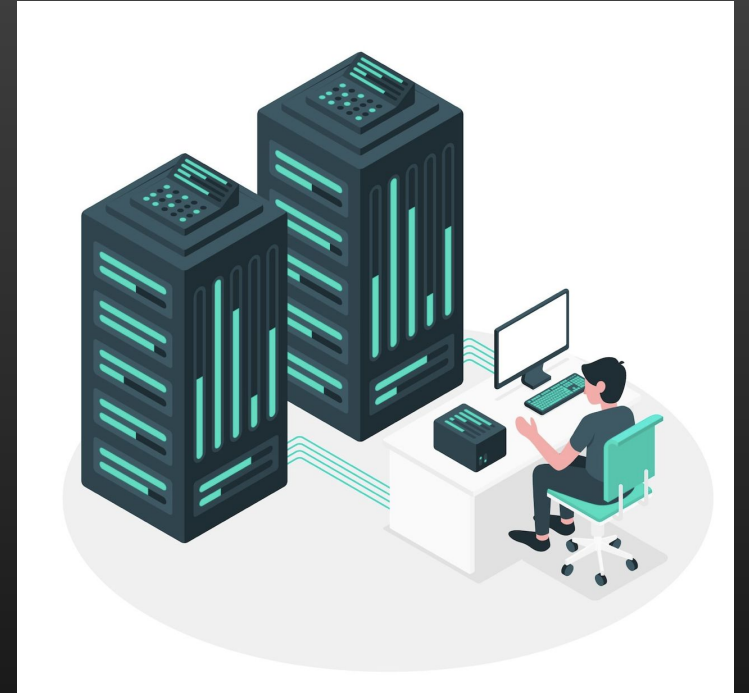
¿Cómo conectarse?

3 cosas que se necesitan

Dirección

Usuario

Contraseña



Comando en consola:
ssh

grupoXX@codd.ing.puc.cl

Contraseña

- Por defecto es el número de grupo
- Comando en consola para cambiarla: `passwd`
 - *Pedirá clave actual*
 - *Pedirá clave nueva 2 veces*

Subir archivos al Servidor



**Opción 1: Utilizar un
Repositorio de Git**



**Opción 2: No utilizar un
repositorio**

Opción 2.1: Vía FileZilla

Opción 1 (Repositorio de Git)

- Ir a repo.new o apretar + □ “new repository” en Github
- Hacer repositorio **PRIVADO**
- Agregar al resto del grupo
 - “Settings” □ “Manage Access” □ “Invite Colaborator”

¿Para qué?

- Trabajar **asincrónicamente** (Es decir, trabajo diferido en el tiempo)
- Poder tener un **Back-up** del avance

Trabajo colaborativo asíncrono

Antes que todo...

¿Qué son las “branches”?

Espacios o entornos independientes para que los desarrolladores de cualquier índole(Back-end, Front-end, Tester, etc) pueda usar y así trabajar sobre un mismo proyecto sin borrar el conjunto de archivos originales.

Ahora si:

Trabajo colaborativo asíncrono

Sin branches

1. Hacer pull antes de hacer cambios

Git pull

2. Luego se crean los commits directo a main

git add index.php + git commit -m "arregla el bug"

3. Inmediatamente se suben al origen(GitHub)

Git push

Trabajo colaborativo asíncrono

Con branches

1. Hacer pull y luego crear una rama de git para implementar algo

git checkout -b arreglar-error

2. Se hacen commits en esa rama y no en main

git add index.php + git commit -m "arregla el bug"

3. Se hacen Pull requests en GitHub y se unen los cambios

git push -u origin arreglar-error + github.com/<usuario>/<repo>/pulls

4. Volver a main y hacer pull

git checkout main y luego git pull

¿Cómo subir archivos al servidor?

Existen muchas maneras de subir archivos. En esta ocasión les mostraremos la siguiente:

- **FileZilla**





¿CÓMO UTILIZAR POSTGRESQL EN EL SERVIDOR?



Comenzamos por...

1. Ingresa a postgres con el comando **psql**
2. Ingresa la contraseña
 1. Por defecto es **'grupoXX'**
 2. Luego debes cambiar la contraseña con el siguiente código:
 1. **ALTER USER <grupoXX> ENCRYPTED PASSWORD 'newpassword';**
3. Ingresa a la base de datos correspondiente a la entrega 1 con **\c grupoXXe1**

Muy importante que trabajen desde esta base de datos!

Comandos útiles (parte 1)

- `\l` □ lista las bases de datos
- `\c <db>` □ ingresar a la base de datos 'db'
- `\dt` □ lista las tablas en una db
- `\d <table_name>` □ describe una tabla
- `\?` □ lista todos los comandos psql
- `\h ALTER TABLE` □ información detallada
- `\q` □ salir (también Ctrl+D)

Otros comandos útiles (parte 2)

- SELECT
- CREATE TABLE
- UPDATE
- ALTER TABLE
- DELETE
- DROP TABLE
- INSERT INTO
- CREATE DATABASE
- \COPY
- DROP DATABASE

Creando tablas...

```
CREATE TABLE users(  
    User_id INT (serial) PRIMARY KEY,  
    username VARCHAR (15) UNIQUE NOT NULL,  
    Password VARCHAR (20) NOT NULL,  
    Email VARCHAR (50) UNIQUE NOT NULL,  
    Birth_date DATE  
);
```


Poblando tablas (Manualmente)

```
INSERT INTO users  
  (user_id, username, password, email, birth_date)  
VALUES (1, 'asc', 'hola123', 'asc@uc.cl', '1999-09-03');
```

Poblando tablas (Manualmente)

```
INSERT INTO users  
(user_id, username, password, email, birth_date)  
SELECT * FROM other_table WHERE condition;
```

*Cualquier declaración SELECT que sea válida

Poblando tablas (Con un csv)

```
\COPY users (columna_names)  
FROM 'relative/path/to/file.csv'  
DELIMITER ',' CSV HEADER;
```



¡MUCHAS GRACIAS!

Más información próximamente en la wiki del
curso