Ayudantía 2

IIC2413 2021-2

Miguel Martínez - Amparo Stevens - Benjamín Vicente

¿Qué veremos hoy? 👀

1. ¿Cómo funciona el servidor?
    a. Ingresar con usuario y contraseña
    b. Ubicación archivos
    c. Subir archivos con FileZilla

2. ¿Cómo trabajar de forma colaborativa?
    a. Repositorio de GitHub
    b. Trabajo simultáneo con LiveShare
    c. Trabajo asíncrono con ramas de Git

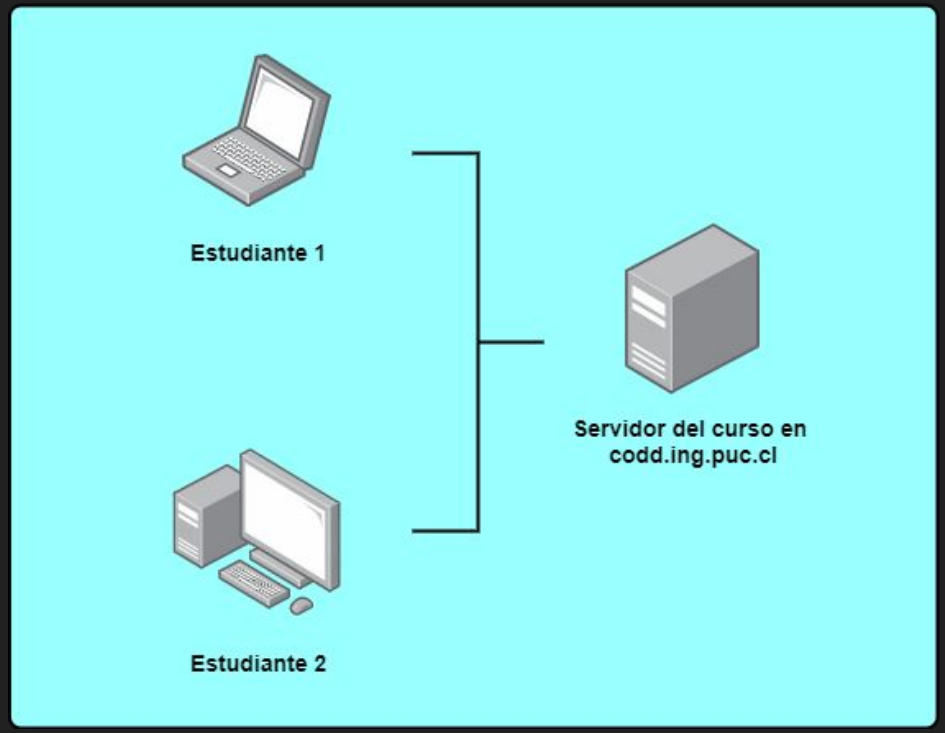
3. ¿Cómo utilizar PostgreSQL en el servidor?
    a. Navegación
    b. Comandos útiles
    c. Creación de tablas

1. ¿Cómo funciona el servidor?

Servidor

Todas las entregas del proyecto se harán en el **servidor** del curso.

Este no es más que un computador que siempre está prendido, en el DCC.



¿Cómo nos conectamos al servidor?

Para conectarnos necesitamos la **dirección** del servidor, un **usuario** y una **contraseña**.

Ejecutamos el siguiente comando en consola:

ssh grupo<número de grupo>@codd.ing.puc.cl

La contraseña por defecto es nuevamente el **número de grupo**.

¿Cómo subir archivos al servidor?

Hay muchas maneras. Las más recomendadas se encuentran detalladas en la **Wiki** del repositorio del curso.

En esta oportunidad les vamos a mostrar cómo hacerlo a través de un programa llamado **FileZilla**.

2. ¿Cómo trabajar de forma colaborativa?

Repositorio de Git

¿Para qué?

1. Backup del avance
2. Trabajar asincrónicamente

¿Cómo?

1. Ir a repo.new o apretar + → “new repository” en GitHub
2. ¡Hacer el repositorio privado!
3. Agregar al resto del grupo en
   “Settings” → “Manage Access” → “Invite Colaborator”

Trabajo colaborativo simultáneo

Live Share en VSCode al rescate!

- Requiere que una persona tenga el código
- El resto se une en una sesión (como Google Docs)

Se puede obtener en aka.ms/vsls

Trabajo colaborativo asíncrono

Se puede ir uniendo avances con un repositorio

1. Sin branches:
    a. Hacer pull antes de hacer cambios
    `git pull`
    b. Luego se crean los commits directo a main
    `git add index.php` + `git commit -m "arregla el bug"`
    c. E inmediatamente se suben al origin (GitHub)
    `git push`

Trabajo colaborativo asíncrono

2. Con branches (recomendado):
    a. Hacer pull y luego crear una rama de Git para implementar algo
    `git checkout -b arreglar-error`
    b. Se hacen commits en esa rama y no en main
    `git add index.php` + `git commit -m "arregla el bug"`
    c. Se hacen Pull Requests en GitHub y se unen los cambios
    `git push -u origin arreglar-error` + `github.com/<usuario>/<repo>/pulls`
    d. Volver a main y hacer pull
    `git checkout main` y luego `git pull`

3. ¿Cómo utilizar PostgreSQL en el servidor?

Antes de comenzar...

1. Ingresa a postgres con el comando `psql`
2. Ingresa contraseña:
    a. Por defecto es `'grupoXX'`
    b. Cambia la contraseña con

    ```
    ALTER USER <grupoXX> ENCRYPTED PASSWORD 'newpassword';
    ```

3. Ingresa a la base de datos correspondiente a la entrega 1 con `\c grupoXXe1`

... algunos 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)

... algunos comandos útiles (parte 2)

```
SELECT
UPDATE
DELETE
INSERT INTO
\COPY
```

```
CREATE TABLE
ALTER TABLE
DROP TABLE
CREATE DATABASE
DROP DATABASE
```

... ahora sí, ¡manos a la obra! 🔨

```
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
);
```

¿Cómo poblar mis tablas? 🤔

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

¿Cómo poblar mis tablas? 🤔

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

* Se puede utilizar cualquier declaración SELECT que sea válida

¿Cómo poblar mis tablas? 🤔

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

* La ruta relativa comienza en la carpeta que la que abrieron psql

¡Muchas gracias!

Para más información visita la wiki del curso