



# SOFTWARE ARCHITECTURE

2024-25

Jose Emilio Labra Gayo

Pablo González

Irene Cid Rico



Escuela de  
Ingeniería  
Informática



Universidad de Oviedo

## Lab 1

- Intro to labs
- Teams organization
- Git
- GitHub
- Wikidata

# ● Intro to labs

What are we going to do in these sessions?

Design and develop an online game of questions and answers that are automatically generated using Wikidata.

## Resources?

- <http://arquisoft.gihub.io>: course documents.
- [Virtual Campus](#)
- WIQ Lab [specification](#)
- Project github [repositories](#) .

## Lab assesment?

70% - Teamwork 

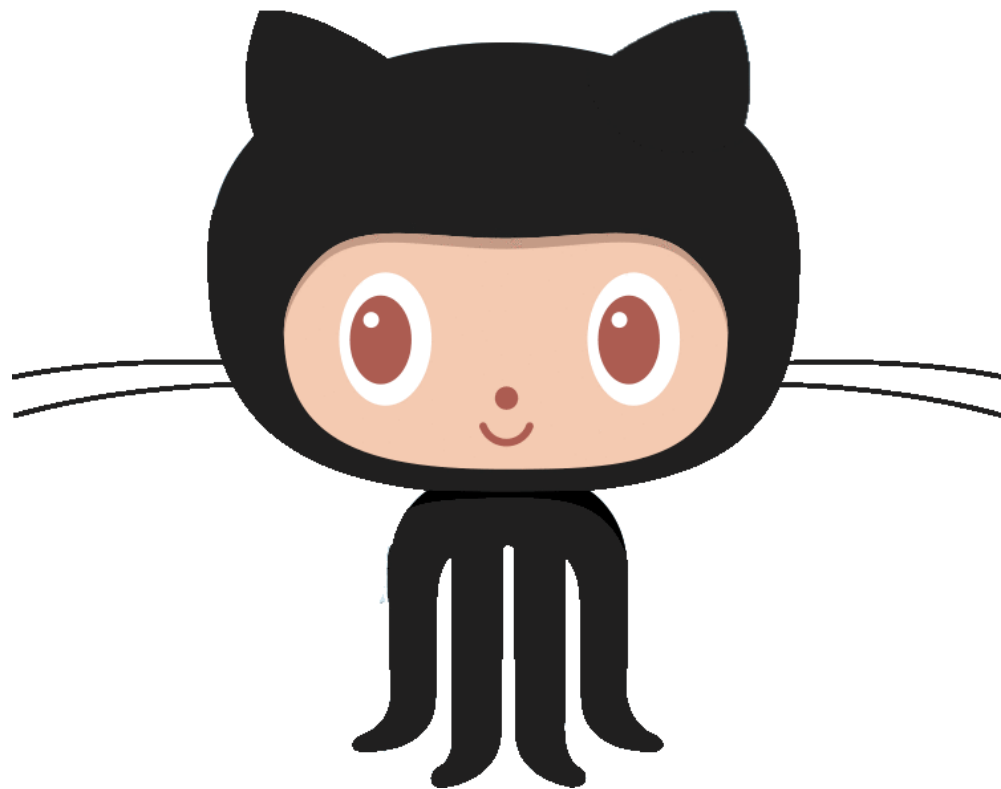
30% - Individual work 

# ● Team work

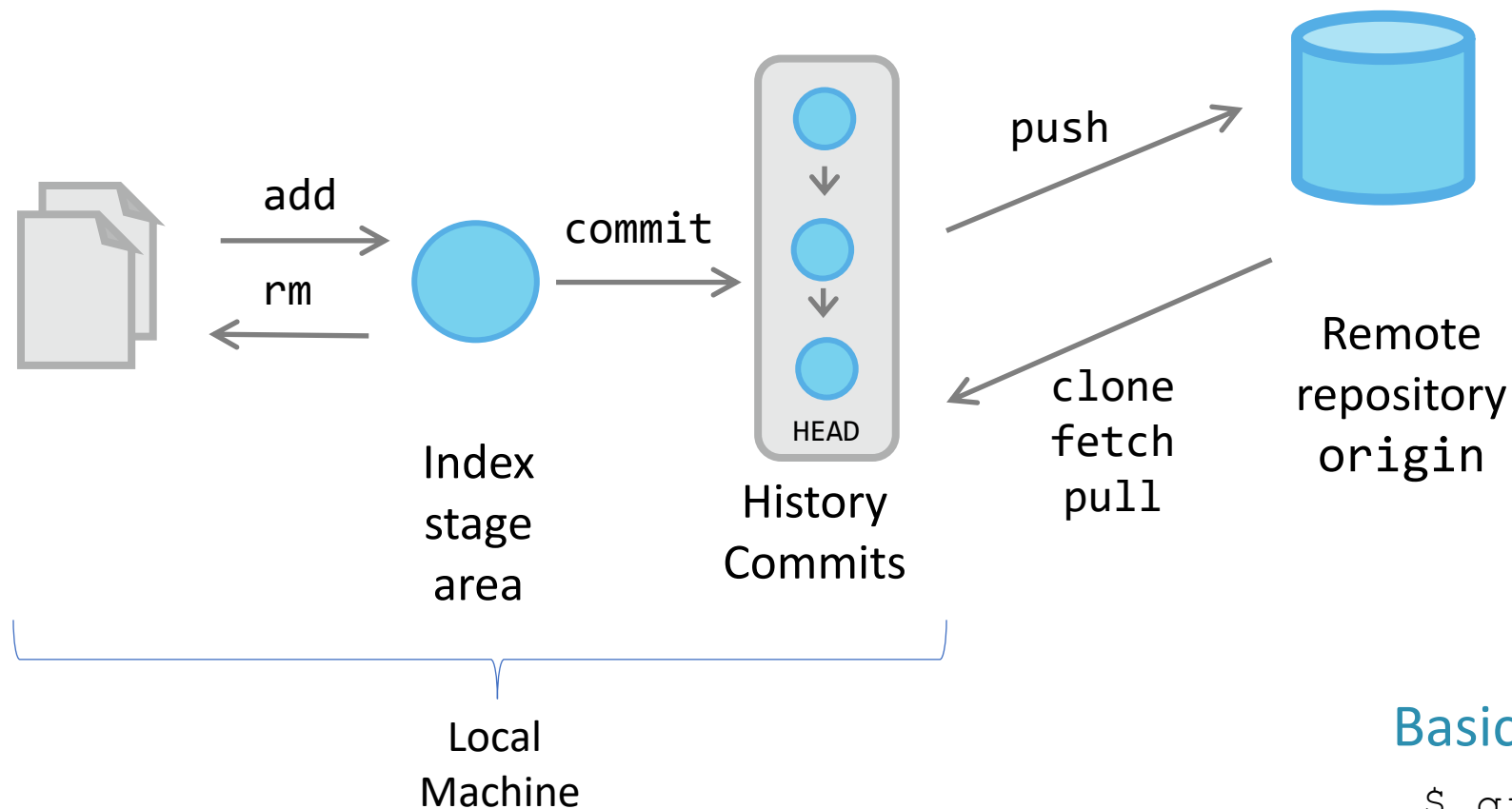
## Meeting minutes

- Each lab session == group meeting.
  - Other meetings allowed out of lab sessions
- **Mandatory** to create minutes of each meeting
- Wiki section of each repository will be used to record minutes
- Minimal mandatory format :
  - Date 📅
  - Participant list 👥
  - Agreements about work assignment for next session (open issues) 🤝
  - Review state of tasks from past meetings ☑
    - Links to **Issues** and **Pull requests** 🔗
  - Short description of decisions taken
    - Preferable to include links architecture decision records (<https://adr.github.io/>)

● Git



# ● Git



## Basic workflow

```
$ git init  
$ git clone urlRepository  
$ git add .  
$ git commit -m "message"  
$ git push origin master
```

# ● Git

## Working with branches

Create a branch:

```
$ git checkout -b branch1
```

Check our current branch:

```
$ git branch
```

Change to another branch:

```
$ git checkout master
```

See differences from branches

```
$ git diff --stat master branch1
```

Merge branch:

```
$ git checkout master
```

```
$ git merge --no-ff branch1
```

Remove branch:

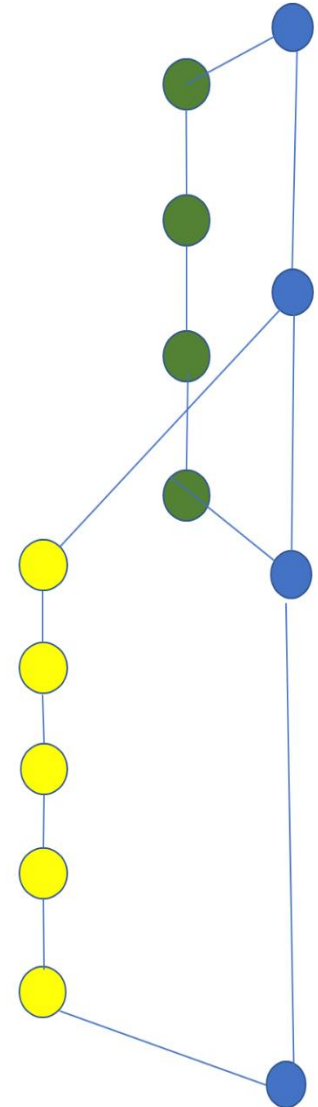
```
$ git branch -d branch1
```

Create the develop branch:

```
$ git checkout -b develop
```

Push it to the remote repository:

```
$ git push origin develop
```



# ● Git      Branching strategies

- Several strategies whose success depends on several factors.
  - See: <https://martinfowler.com/articles/branching-patterns.html>
- Some popular patterns:
  - Git-flow, by Vincent Driessen, 2010: [A successful Git branching model](#)
  - [GitHub Flow](#) strategy
  - Trunk based development: <https://trunkbaseddevelopment.com/>



# Pull Request

## Steps

### New branch

```
$ git flow feature start RE1 develop #option 1  
$ git checkout -b feature-RE1 develop #option 2
```

Add your name inside the *Collaborator* section in your *README.md* file

### Send your local changes

```
$ git add .  
$ git commit
```

### Submit your changes to remote

```
$ git push --set-upstream origin feature-RE1
```

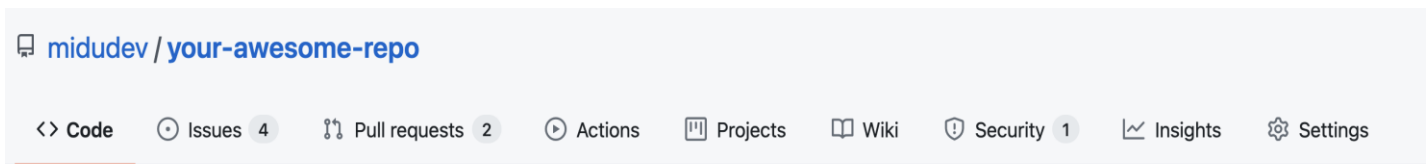
Go to github and **ask for a pull request**

The screenshot shows the GitHub repository interface. At the top, there are statistics: 5 commits, 3 branches, 0 packages, and 0 releases. Below this, a section titled "Your recently pushed branches:" lists two branches: "develop (about 1 hour ago)" and "feature/RE1 (1 minute ago)". Each branch entry has a green button labeled "Compare & pull request".





# Pull request



## Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

1 base: master ← 2 compare: feature-new-cool-thing ✓ Able to merge. These branches can be automatically merged.

3 Fix problem when creating a user

Write Preview H B I

4 # Changes

Use uuid v4 instead a manual generated id with Math.random to avoid problems when using the id in some places.

# Fixed issues

Fixed #30 on generating new users

Attach files by dragging & dropping, selecting or pasting them.

5 Create pull request

Target branch

Branch we want to merge

Add more context

Create pull request

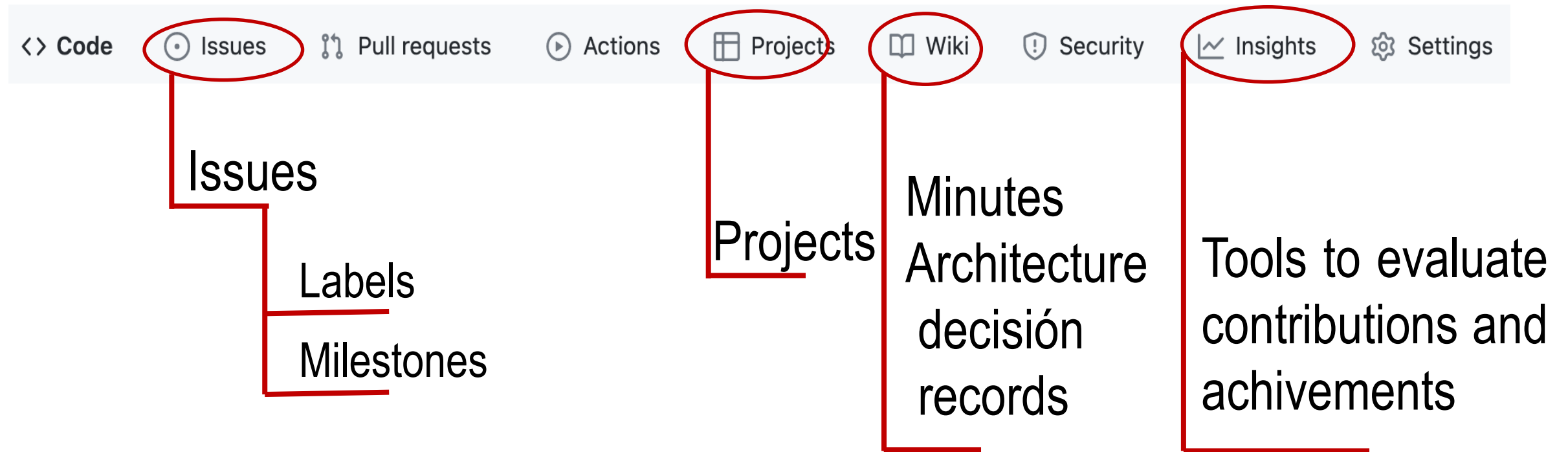
# ● GitHub as a Project management tool

## Advantages of Project management

- Project planning (future)
- More control over current project's state (present)
  - Detect bottlenecks.
  - Share work load between team members.
  - Current problems.
- Report achievements (Past).
  - Evaluate each member's contributions to the project

# ● GitHub

Some tools provided by github for Project management





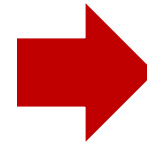
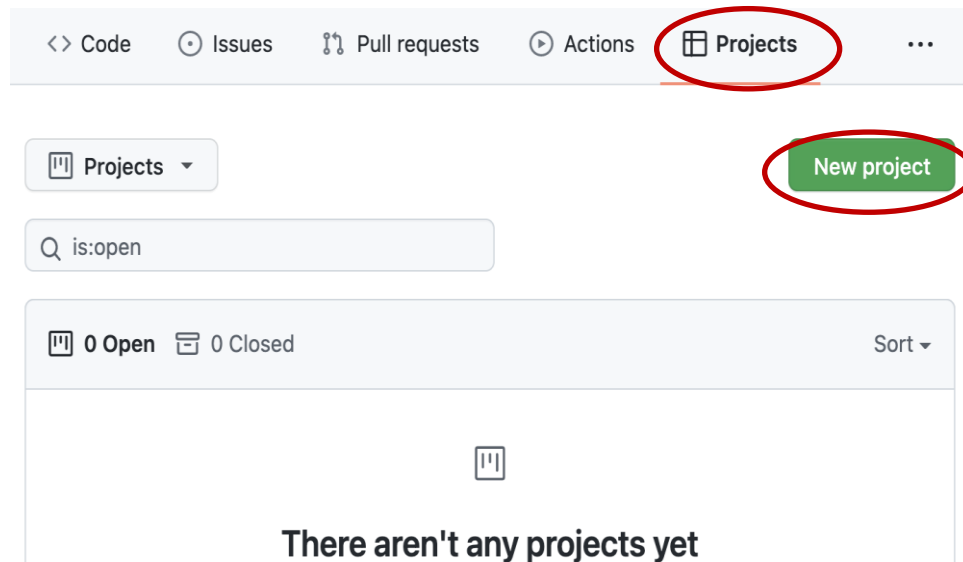
Backlog	To Do	In progress	Done
<ul style="list-style-type: none"><li>Create Docker image for project</li><li>Throw errors for missing dependencies</li><li>Curling results in empty response, server crashes</li><li>Serve aggregations by admin level</li><li>Base API in Swagger Spec Has Extra /</li><li>Use arguments to overwrite default configurations</li><li>Return hasProbsData in API response for schools</li><li>expired token warning gets overridden by filehandler library and return html instead of json</li></ul>	<ul style="list-style-type: none"><li>Colombia shapefile (mpis.json) should be fetched from api rather than be required in react app.</li><li>Fix "file_lines" issue in react-app: public/lib.js</li><li>Comment and refactor code in action-select-country.js</li><li>Add disclaimer popup on load</li><li>Reconfigure login code to work in production</li></ul>	<ul style="list-style-type: none"><li>Add contributing guidelines placeholder (closes #93)</li><li>Determine where to begin writing tests and refactoring</li><li>Map relationships between apps, maps, and azure resources.</li><li>Mobility: Map should display data for mobility being displayed</li></ul>	<ul style="list-style-type: none"><li>Rename test suite (run-tests.sh), add comments</li><li>Refactor WebGLLayer.js</li><li>Create production/staging/dev deployment slots for Mobility Mapping</li><li>Set up automatic CI testing for new pull requests</li><li>Add loading spinner to mobility map</li><li>Add spinner mobility</li><li>Add CodeClimate maintainability + test coverage checks to CI</li><li>Add TravisCI and CodeClimate tests</li></ul>

+ Add column

# GitHub

## Project tab

- It is possible to create KanBan projects
- Automate workflow managing also issues and pull requests
- It is possible to create different dashboards (docs, backend,...)



### Create a new project

Coordinate, track, and update your work in one place, so projects stay transparent and on schedule.

#### Project board name

#### Description (optional)

#### Project template

Save yourself time with a pre-configured project board template

Automation options



## Issues

They are like post-it in a **Kanban** project

👉 Each problem, task or even question related with the Project can have a issue.

They use **Markdown**.

Recommended to write only the necessary to understand the issue. You can add pictures or links

**It is the main part of a Project** and can be used to understand the state of the project.

An Issue can be related with other entities in a github repo.

People 

Labels 

Milestones 



**We only assess information that appears in the github repo**



## Investigate to what extent HTTPS should be mandatory #1091

New issue

Issue status

Open

RubenVerborgh opened this issue on 9 Dec 2021 · 2 comments

Problem explanation



RubenVerborgh commented on 9 Dec 2021

Member

In some previous testing, I have come across preliminary evidence that some Solid-related functionality only works over HTTPS. In particular, when running Mashlib as the on-server UI, authentication seems to break because the server is not running over HTTPS.

Whereas this is actually a question for the bigger Solid ecosystem, we can test some assumptions on CSS and turn them into recommendations.

If the answer is that some functionality only works over HTTPS, then we might want to make CSS start over HTTPS out of the box (e.g., by auto-generating localhost certificates etc.).

Comments



RubenVerborgh added the task label on 9 Dec 2021

Assignees



RubenVerborgh

Assigned to



Labels

task

Projects

None yet

Milestone

No milestone

Milestones



Linked pull requests

Successfully merging a pull request may close this issue.


None yet

Corresponding pull request

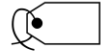
Issue log

# ● GitHub

## Labels

- Labels can be used to catalog issues.
- Generic labels from GitHub
- You can create your own labels (backend, frontend, bbdd, hierarchies)
- You can personalize color and even use emojis  .





Label name



32 labels

worker threads	Making the server multithreaded	5
bug	Something isn't working	17
dependencies	Pull requests that update a dependency file	
developer experience		5
difficulty:high		2
difficulty:low		13
difficulty:medium		9



Short description



Number of labelled issues



Common hierarchies use to give priorities

# You can use arquisoft FAQ for questions

<https://github.com/Arquisoft/faq/issues>

- Share questions about the course
- It is allowed to add any issue that describes some question in either English or Spanish,
  - Anyone can contribute answering the question or adding any comment.
  - Contributors must follow a code of conduct that respects the ethical considerations from a University of Oviedo Course.
- The teachers can remove any issue or contribution that they consider inappropriate for the course.

# Additional links

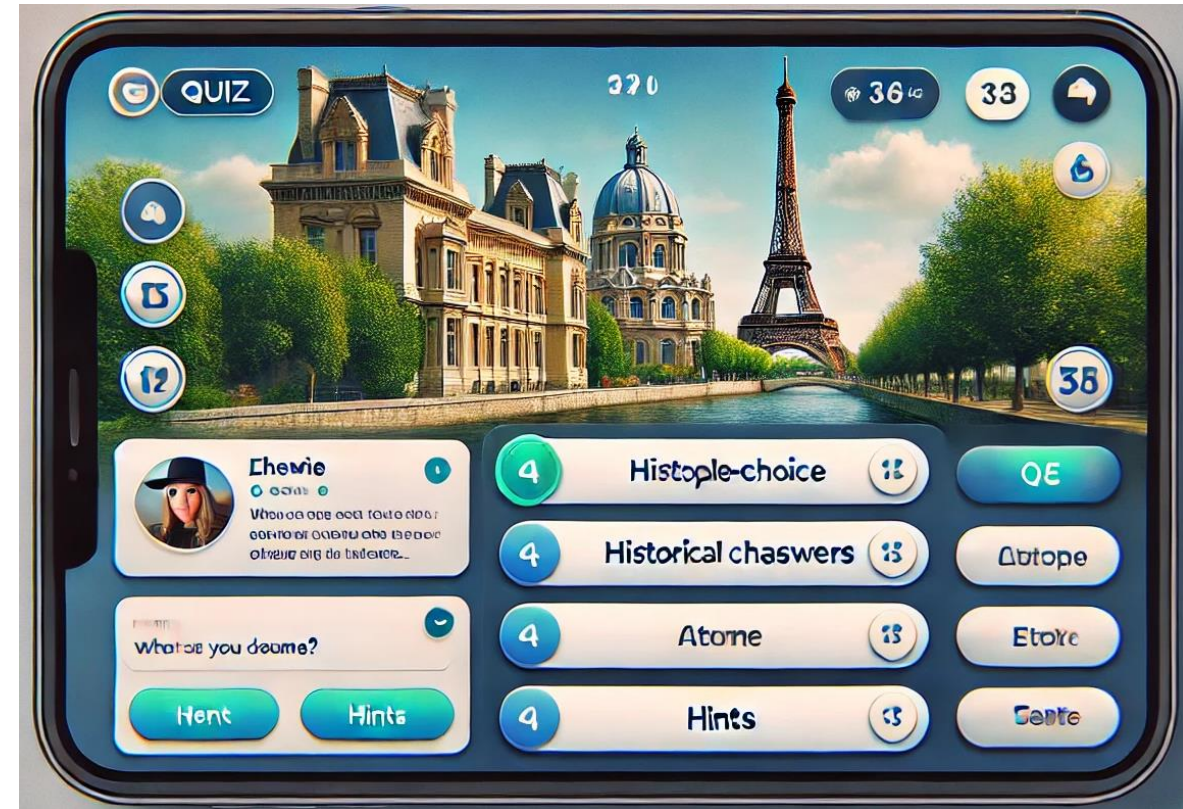
- Introduction to git [Git](#).
- [Quick reference](#) from Pablo Gonzalez
- Short [introducción a git](#) (Hugo)
- Git: the simple guide
  - <https://rogerdudler.github.io/git-guide/index.html>
- [learngitbranching.js.org](http://learngitbranching.js.org)

# Additional Doc

- [Miguel Angel Durán](#) channel and his initial guide [Aprendiendo Git](#).
- Youtube
- [Git explanation in 15 minutes](#)
- [Tutorial](#) for beginners
- [GitHub vs GitLab](#)

# Lab assignment of this year

- WIQ: **W**ikidata based **Q**uizz Show
- Inspired by “Saber y Ganar” quiz show
- It shows a picture and users have to guess the correct answer
- It shows a chat to interact and ask for hints



Description:

<https://arquisoft.github.io/course2425.html#labs>

# About Wikidata

<https://www.wikidata.org/>

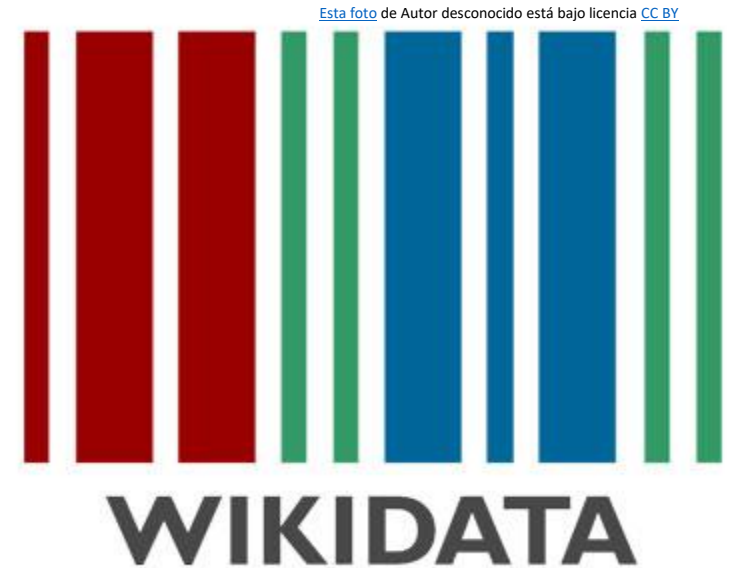
General purpose knowledge base

Supports Wikipedia

Free Access and free code

Part of Wikimedia Foundation

Data access: [https://www.wikidata.org/wiki/Wikidata:Data\\_access](https://www.wikidata.org/wiki/Wikidata:Data_access)



# About LLMs

- Any LLM can be used
- We will give a key to use some LLMs offered by Empathy:
  - Mistral
  - Qwen
  - ...
- Challenge: generate hints based on the picture
  - How to avoid incorrect answers and hallucinations

