

## Short summary of side projects on my GitHub account

@japinol

Name: Joan A. Pinol

GitHub: <https://github.com/japinol7>

Twitter / X: <https://www.twitter.com/japinol>

Updated: 2025-08-14



Languages: Python, Java, C, C++, Go, Rust

Frameworks: Django, Flask, SQLAlchemy, Odoo ERP, JavaFX...

### Projects implemented in Python

- life:
  - Conway's Game of Life with colors.  
<https://github.com/japinol7/life>
  - Tests: No unit tests, but includes a test suite.
  - Notes: GUI under construction  
For the moment you will have to use the keyboard keys and the line command parameters.  
Apart from these GUI elements, this version is fully operational.
- movies\_lib\_explorer
  - Movies Library Metadata Explorer using Django and Django REST framework  
[https://github.com/japinol7/movies\\_lib\\_explorer](https://github.com/japinol7/movies_lib_explorer)
  - Tests: Unit tests implemented using `django.test` with `unittest`.
  - Notes:  
This is a very simple Movies Library Metadata Explorer using Django.  
It includes a REST API for some catalog models using Django REST framework.  
Its purpose is to be used as a Django web app example.
- teams-generator-services
  - Teams Generator Services  
<https://github.com/japinol7/teams-generator-services>
  - Tests: Some unit tests implemented for the main service using `pytest`:  
<https://github.com/japinol7/teams-generator-services/tree/main/services/generate-teams>
  - Notes:  
Some microservices related to the generation of RPG teams.  
This project uses AWS and Docker containerization.
- maze-solver
  - Maze Solver  
<https://github.com/japinol7/maze-solver>
- flask-api-calls
  - Flask API Calls  
<https://github.com/japinol7/flask-api-calls>
  - Notes:  
This is a very simple Flask web app example that calls some APIs, such as NASA APIs, Marvel Comics, Spotify or OpenAI (ChatGPT).
- music-lib-explorer
  - Music Library Metadata Explorer  
<https://github.com/japinol7/music-lib-explorer>
  - Notes:  
This is a very simple Music Library Metadata Explorer.  
Its purpose is to be used as a Flask + SQLAlchemy web app example.

- **erp-invoice-model**
  - Very Basic Invoice Model Example  
<https://github.com/japinol7/erp-invoice-model>
  - Tests: Unit tests implemented using `unittest`:
  - Notes: Run this example this way:
  - `$ python -m unittest`
- **the-codemaster**
  - Game: The CodeMaster. Nightmare on Bots' Island  
<https://github.com/japinol7/the-codemaster>
  - Tests: End-to-end tests and a test suite framework to run them.  
<https://github.com/japinol7/the-codemaster/tree/master/suiteoftests>
  - Notes: Python game using `pygame` and `pygame-gui`.
- **tictactoe**
  - Game: Tic Tac Toe  
<https://github.com/japinol7/tictactoe>
  - Tests: Some unit tests implemented for the `ComputerPlayer` class using `pytest`:  
<https://github.com/japinol7/tictactoe/tree/main/tests/unit/model>
  - Notes: Python game using `pygame`.  
Computer Players use a simple version of the Minimax decision rule algorithm.
- **sudoku-solver**
  - Sudoku Solver. Solves sudokus  
<https://github.com/japinol7/sudoku-solver>
  - Tests: Test for the backtracking algorithm solution using `pytest`:  
<https://github.com/japinol7/sudoku-solver/tree/main/tests>
- **hanoi-towers**
  - Game: The towers of Hanoi with graphics. A mathematical puzzle game.  
<https://github.com/japinol7/hanoi-towers>
  - Tests: Tests for the iterative and recursive solvers using `pytest`:  
<https://github.com/japinol7/hanoi-towers/tree/main/tests/unit/model>
  - Notes: Python game using `pygame`.
- **simple-tier-list**
  - Simple tier list.  
<https://github.com/japinol7/simple-tier-list>
  - Notes: Python application using `pygame` and `pygame-gui` for the GUI.

## Projects implemented in Java

- retail-kiosk
  - JAP Merch's Retail Kiosk  
<https://github.com/japinol7/retail-kiosk>
  - Tests: Only some tests for the Item model using JUnit 5.
  - Notes:  
Uses JavaFX.

## Additional projects implemented in Python

- small-python-projects
  - Some very basic examples of Python programming  
<https://github.com/japinol7/small-python-projects>
  - Tests: Code katas in cyber-dojos project with TDD (tests first) using `pytest`:  
[https://github.com/japinol7/small-python-projects/tree/master/projects/cyber\\_dojos](https://github.com/japinol7/small-python-projects/tree/master/projects/cyber_dojos)
- small-odoo-addons
  - Small addons examples of Python programming with the Odoo CE 18.0 ERP framework.  
<https://github.com/japinol7/small-odoo-addons>
    - JAP Sales:
      - Duplicate a sale line with the duplication button.
      - Create an Invoice from Selected Sale Lines.
- odoo-integration-examples
  - Odoo integration examples in Python using json-rpc and xml-rpc.  
<https://github.com/japinol7/odoo-integration-examples>
  - odoo-jsonrpc  
<https://github.com/japinol7/odoo-jsonrpc>
  - odoo-xmlrpc  
<https://github.com/japinol7/odoo-xmlrpc>
- poetry-lib-explorer
  - Poetry Library Explorer  
<https://github.com/japinol7/poetry-lib-explorer>
  - Warning:  
The poems in this project were written by myself when I was younger. Read them under your own risk ; )  
Although these poems are not great, they all have copyright.
  - Notes:  
This is a very simple Poetry Library Metadata Explorer.  
Its purpose is to be used as a Flask + SQLAlchemy web app example.
- snakes
  - Game: Snakes  
<https://github.com/japinol7/snakes>
  - Notes: Python game using pygame.
- three-clocks
  - Game: Three Clocks  
<https://github.com/japinol7/three-clocks>
  - Notes: Python game using pygame and pygame-gui.
- fibonacci
  - Fibonacci  
<https://github.com/japinol7/fibonacci>
  - Notes:  
Benchmarks some algorithms that calculate elements of the Fibonacci sequence.

## Additional projects implemented in C++

- snake
  - Game: Classic snake game derivative.  
<https://github.com/japinol7/snake>
  - Notes:  
C++ game using the C graphics library: SDL2.

## Additional very small projects implemented in Python, C, C++, Go, and Rust

- small-python-projects
  - Some very basic examples of Python programming  
<https://github.com/japinol7/small-python-projects>
  - Tests: Code katas in cyber-dojos project with TDD (tests first) using `pytest`:  
[https://github.com/japinol7/small-python-projects/tree/master/projects/cyber\\_dojos](https://github.com/japinol7/small-python-projects/tree/master/projects/cyber_dojos)
- small-c-projects
  - Some very basic examples of C programming  
<https://github.com/japinol7/small-c-projects>
  - Tests: Code katas in cyber-dojos project with TDD (tests first):  
[https://github.com/japinol7/small-c-projects/tree/main/projects/cyber\\_dojos](https://github.com/japinol7/small-c-projects/tree/main/projects/cyber_dojos)  
[https://github.com/japinol7/small-c-projects/tree/main/tests/unit/cyber\\_dojos](https://github.com/japinol7/small-c-projects/tree/main/tests/unit/cyber_dojos)
- small-cpp-projects
  - Some very basic examples of C++ programming  
<https://github.com/japinol7/small-cpp-projects>
  - Tests: Code katas in cyber-dojos project with TDD (tests first):  
[https://github.com/japinol7/small-cpp-projects/tree/main/projects/cyber\\_dojos](https://github.com/japinol7/small-cpp-projects/tree/main/projects/cyber_dojos)  
[https://github.com/japinol7/small-cpp-projects/tree/main/tests/unit/cyber\\_dojos](https://github.com/japinol7/small-cpp-projects/tree/main/tests/unit/cyber_dojos)
- small-go-projects
  - Some very basic examples of Go programming  
<https://github.com/japinol7/small-go-projects>
  - Tests: Code katas in cyber-dojos project with TDD (tests first):  
[https://github.com/japinol7/small-go-projects/tree/main/projects/cyber\\_dojos](https://github.com/japinol7/small-go-projects/tree/main/projects/cyber_dojos)
- small-rust-projects
  - Some very basic examples of Rust programming  
<https://github.com/japinol7/small-rust-projects>
  - Tests: Code katas in cyber-dojos project with TDD (tests first):  
[https://github.com/japinol7/small-rust-projects/tree/master/projects/cyber\\_dojos](https://github.com/japinol7/small-rust-projects/tree/master/projects/cyber_dojos)