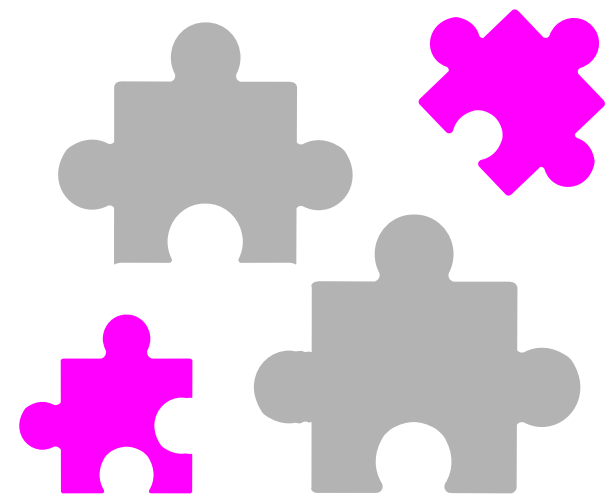


v



wiki_guesser

Try to estimate Wikipedia page popularity as closely as possible. Test your intuition and learn fascinating facts along the way!

presentation_intro

Our Collaborative Coding Journey

presentation_index.csv

index,title

1,presentation_intro

2,python_piranhas

3,collaboration_process

4,game_showcase

5,mvp_and_ideas

7,game_rules

6,github_setup

7,task_management

8,project_structure

9,player_journey

10,functionality_overview

11,biggest_challenges

12,key_learnings

13,wikipedia_discoveries

python_piranhas

Instructor: Maicon



Melody
wiki stats



Martha
cli UI



Alex
git master



Chris
game functions



Denis
storytelling

collaboration_process

Open Zoom Workspace

We kept a live Zoom meeting running, allowing team members to join or leave as needed, with updates on AFK (away-from-keyboard) time.

Flexible Coding Breakouts

We worked solo and used pair-programming for focused (coding) tasks.

Regular Check-Ins

Scheduled full-team meetings ensured everyone was informed and aligned on progress and next steps.

Centralized Communication and Shared Storage

Slack and Github helped us share updates and prevent data silos, ensuring a transparent and streamlined process.

requirements_mvp

setup: Game Setup & Welcome Screen

- > Greeting & brief explanation of the game
- > Play Tutorial Round
- > Enter Player Names
- > Enter Number of rounds

Start of Game == Blueprint for each round

- > Show wikipedia article preview
- > Users guess number of page views
- > Present round winner

end: Winner announcement

- > Replay with same users/no. rounds
- > Option to start all over again

nice_to_have

- > Ask for additional statistics (edits, countries, languages, page size, ...)
- > Improve the Point Reward system (closer estimation, more points)
- > Highscore Dictionary in a .json-file
- > Single player mode: compare the ranking of 2 sites (higher or lower page views)

game_rules

Welcome to Wikiguesser!

How good do you know the interests of other Wikipedia users? Let's have a try!

Whoever estimates better, how many views a random Wikipedia page has, wins the round.

Try to guess better than the other player and become a master Wikiguesser!

If Player 2 enters the same estimation as Player 1, Player 1 will win the point.

So, how do you play Wikiguesser? Here you are:

Chose an uneven amount of rounds you want to compete against each other.

The Players take turns giving their estimates.

Pay close attention to when it's your turn and estimate better than your fellow player to win!

If you are new to Wikiguesser - just start a tutorial round to try it out!

Player 1, please insert your nickname here:

game_showcase

Game Showcase & Repository Link



Try wiki_guesser on your own!

github_setup

Our GitHub Setup


main3 Branches0 Tags


Go to fileAdd fileCode

densenden added content from presentation1b2bd55 · 2 minutes ago40 Commits


.idea	removing url print	yesterday
README.md	added content from presentation	2 minutes ago
main_new.py	Add files via upload	16 hours ago
requirements.txt	contains the libraries used in Project	17 hours ago
wiki_guess_display_menu.py	Hi Python Piranhas, the line colours are changed now to ...	13 hours ago
wiki_guess_game_function.py	Add files via upload	16 hours ago
wiki_statistics_merged.py	Add files via upload	16 hours ago


README

 wiki_guesser


 About the Project

This repository is the result of a collaborative learning experience where we built. Our focus was not only on coding but also on optimizing our workflow, teamwork, and development process.


 How We Worked Together

 Open Zoom Workspace

We maintained a live Zoom session where team members could join and leave as needed, ensuring continuous collaboration. AFK times were communicated to keep expectations clear.

 Regular Check-Ins

Scheduled full-team meetings helped us align on progress, address challenges, and set next steps.

 Flexible Coding Breakouts

About

Repo for our Game based on Wikipedia infos.

Readme

Activity

0 stars

2 watching

0 forks






Releases

No releases published
[Create a new release](#)

Packages

No packages published
[Publish your first package](#)

Contributors 5




Languages

Python 100.0%


Suggested workflows

Based on your tech stack

 Python application

Configure

Create and test a Python application.

 Python package

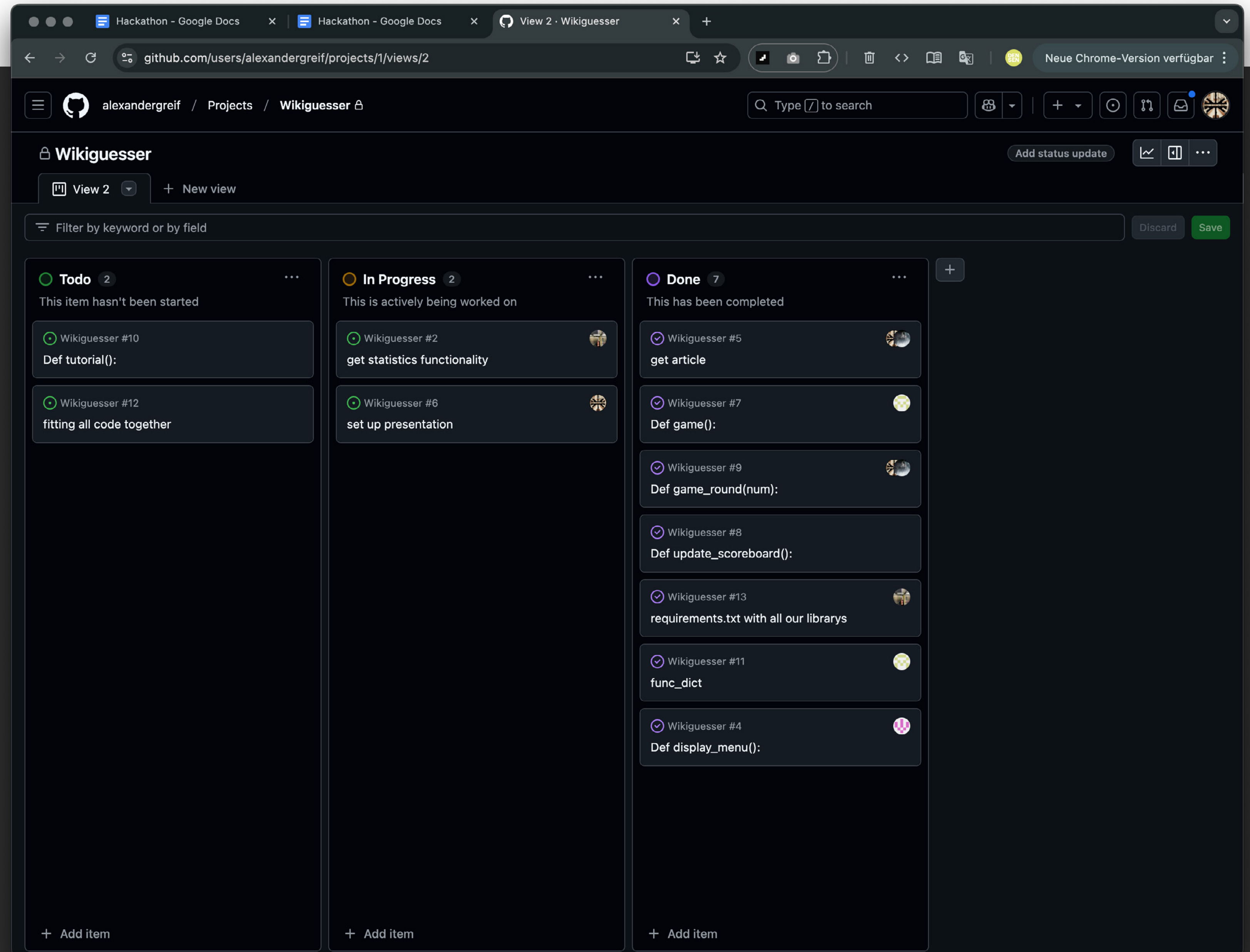
Configure

Create and test a Python package on multiple Python versions.

9

task_management

Task Management in GitHub



The screenshot shows a GitHub project board for the 'Wikiguesser' project. The board is organized into three columns: 'Todo' (2 items), 'In Progress' (2 items), and 'Done' (7 items). Each item is a card representing a task, with a title and a description. The 'Done' column has a plus sign to its right, indicating more items can be added. At the bottom of each column is a '+ Add item' button.

Column	Item	Description
Todo (2)	Wikiguesser #10	Def tutorial():
	Wikiguesser #12	fitting all code together
In Progress (2)	Wikiguesser #2	get statistics functionality
	Wikiguesser #6	set up presentation
Done (7)	Wikiguesser #5	get article
	Wikiguesser #7	Def game():
	Wikiguesser #9	Def game_round(num):
	Wikiguesser #8	Def update_scoreboard():
	Wikiguesser #13	requirements.txt with all our librarys
	Wikiguesser #11	func_dict
	Wikiguesser #4	Def display_menu():

project_structure

Building the Structure

wiki_guesser

setup

game

game_round

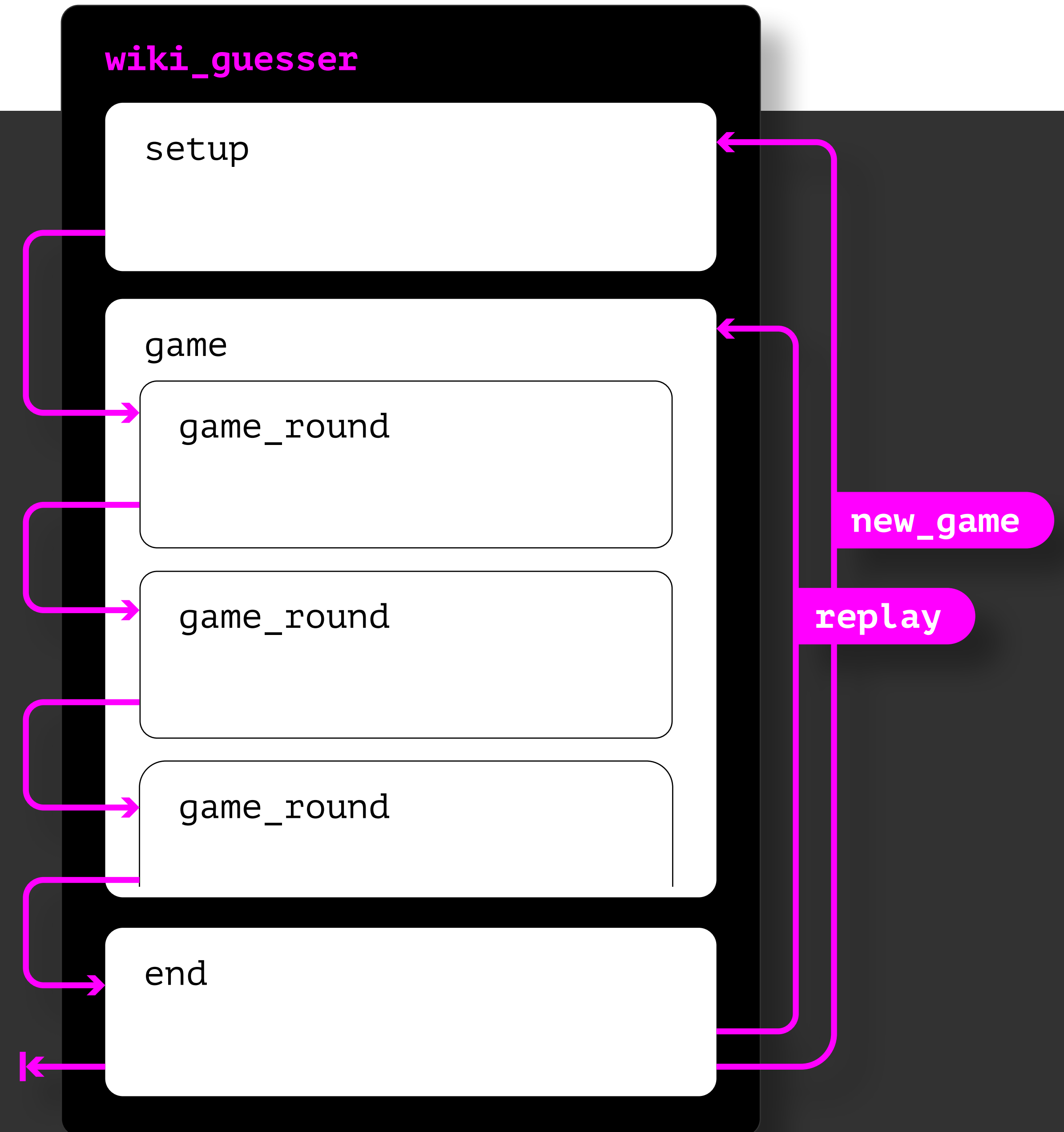
game_round

game_round

end

player_journey

pass



functionality_overview

Program Data Flow overview
and usage of libraries.

wiki_guesser

setup

display_menu

game

game_round

get_article

get_statistics

end

colorama
0.4.6

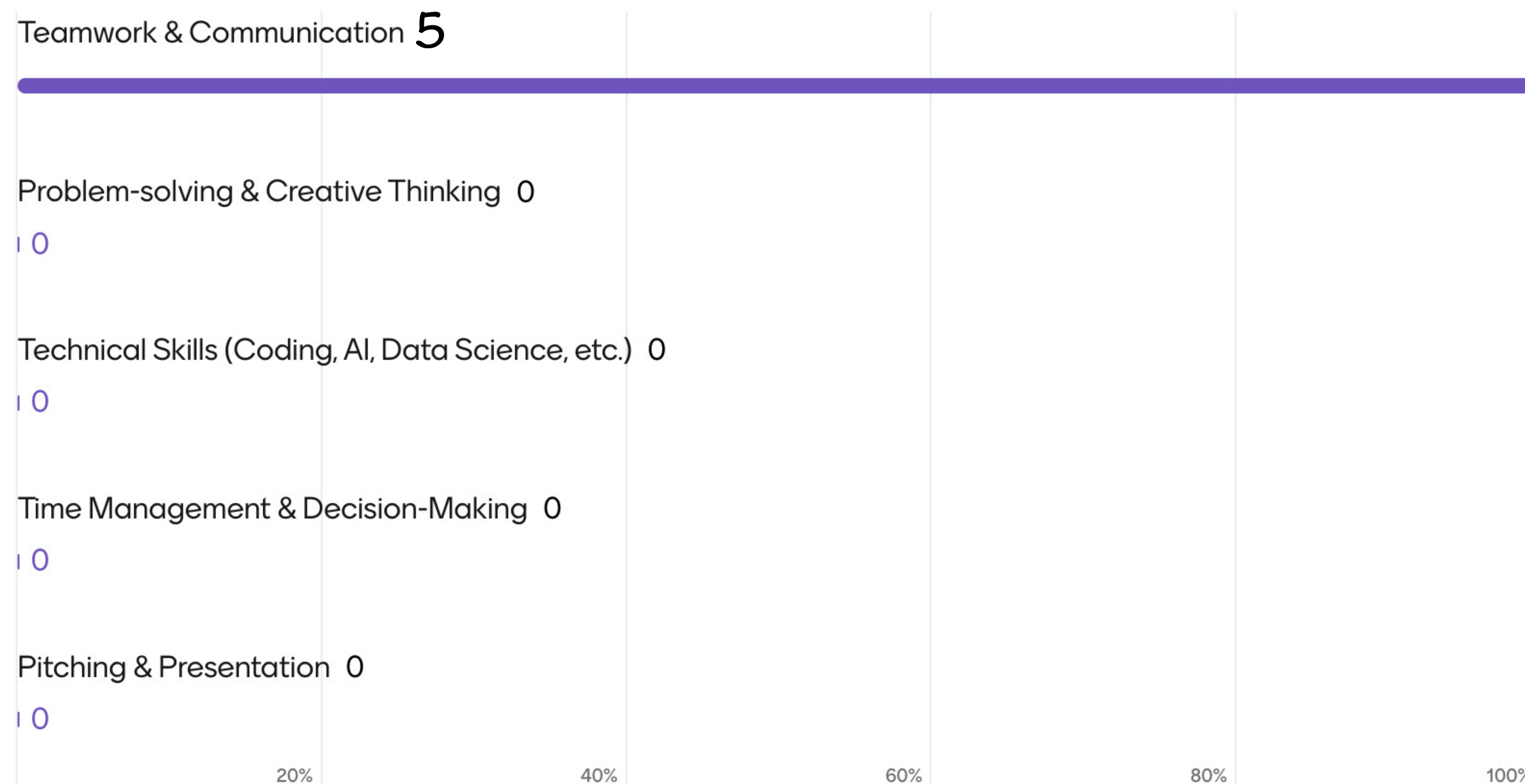
wikipedia
1.4.0

requests
2.32.3

Wikimedia
Analytics
API

biggest_challenges

What was the biggest challenge you faced during this Hackathon?



biggest_challenges

What was the biggest challenge you faced during this Hackathon?

overload by online coop

scoping the feature

computertime

many online hour facetime

real time availability

speed learning git

using git repo in pycharm

letting go of some coding

key_learnings

Key Learnings

modularisation
multiple files functions
connecting code
quick solution to show
using api calls
github
teamwork
setting up github
git vocabulary
communication
planing is key
collaboration

advice_hackathon

What's one piece of advice you'd give to future Hackathon participants?

be open and direct in communication.



take your time to plan the tasks



keep cool

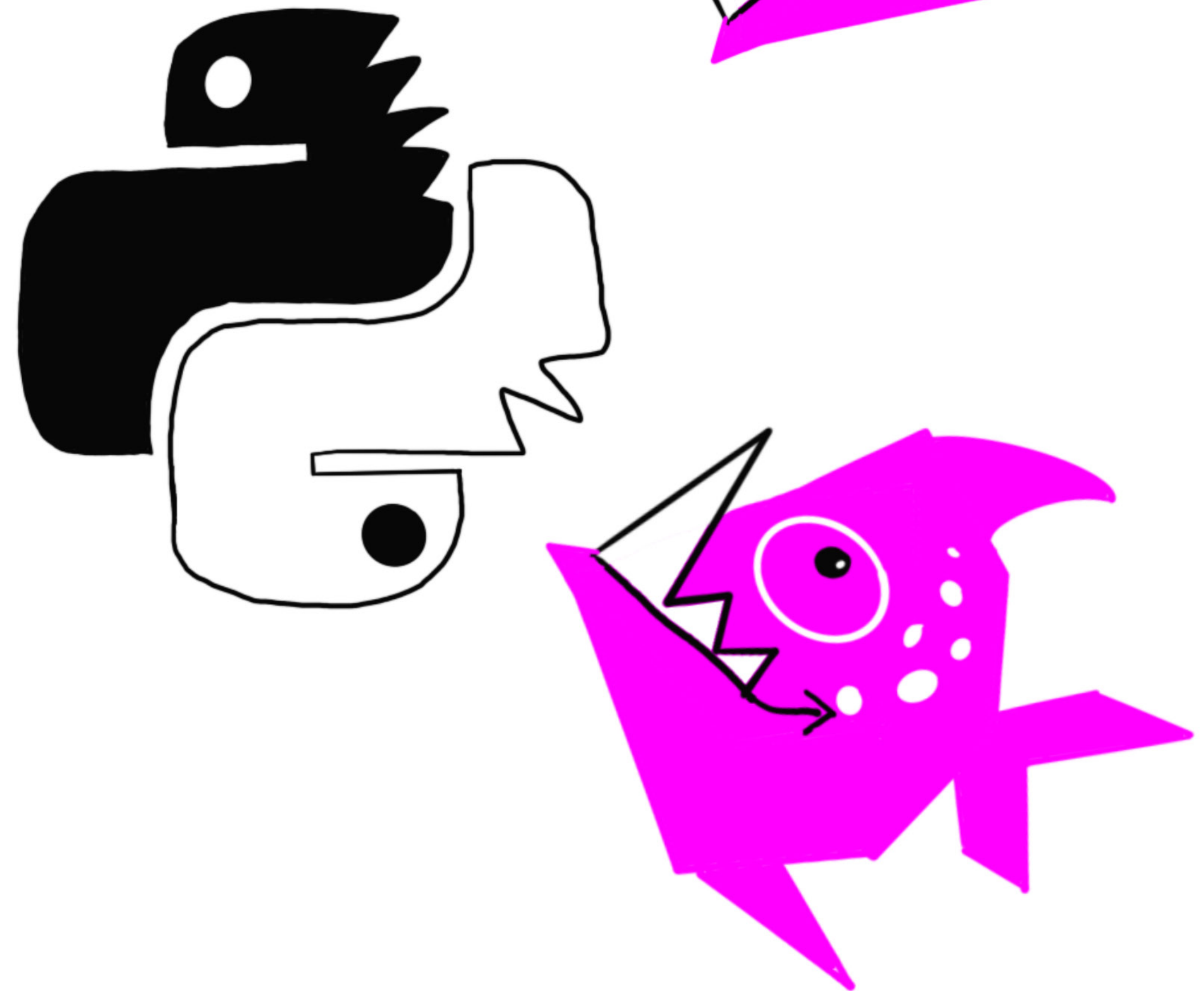


teamwork



experience_share

How was your experience during the Hackathon?



question_time

Feel free to ask the Python Piranhas.



Melody
wiki stats



Martha
cli UI



Alex
git master



Chris
game functions



Denis
storytelling