



# The Hidden Keys Game

Team members: Rehahn Qureshi & Hilda Monterrubio  
September 22nd, 2023



# Agenda



- 1 Project Overview
- 2 Technical Challenges
- 3 Improvement Ideas
- 4 Demo
- 5 Questions

# 1. Project Overview



## 1. How is the game's structure organised?

- Imports
- Room and objects definitions & their relationships (Lists and Dictionaries)
- Initial Game State (current room & objects owned)
- Functions for Game Flow (menu with 2 options: examine & pass door + 2 added functions)
- Main

## 2. Which features did you add to the game?

- Removed the repetitive action of displaying "You are now in x room" every time there was a user input.
- Better wording, more game-like & Some emojis to make it more user-friendly.
- Instead of just taking the input "examine" and asking for the item separately, now the user is able to directly type in "examine object x".
- Added the function "pass door" to automatically explore the room when going through a door by typing "pass door door x".
- Drawings of each room are displayed when entering a room, so the user knows what to examine.
- Celebration video that is displayed when/if the player wins.

## 2. Technical Challenges



### 1. What was the most important technical challenge you faced?

- Adding new functions and ensuring that they are handled appropriately within the code.
- Displaying the drawings of each room as intended, plus the video (not that easy to do this within a function!)
- Working with Google Colab notebook (functionalities are not as straightforward as with JupyterLab or VSCode, especially for managing files)

### 2. How did you overcome those challenges?

- A LOT of trial and error. Isolating the issue and working on it separately instead of looking at the whole code.
- Taking the time to understand and research the different errors that we were getting.
- Code research through different channels (StackOverflow, Medium)
- Working as a team! 💪

### 3. Improvement ideas for the future



- Keep improving the wording, make it ever more user-friendly.
- Implement sound or music effects when a key is found or during an escape attempt.
- Implement a scoring “system” in which either amount of keys found, time being spent, or raking is taking into consideration for the scoring.
- Incorporate a menu option to display a list of currently owned keys.
- Implement error handling for potential timing issues.
- Organize the functions in a Python file and import them into Jupyter for better code organization.

## 4. Demo time!



# Questions ?



Thank you !

**GAME  
OVER**