DungeonsAndPythons

By Emilian Spasov, F107681

Inspiration

- Inspiration from Dungeons
 And Dragons
- The idea of the game is that you want to become "Dungeon Master"
- Navigating through a maze full of monsters, treasures and more



Project structure

- The repository is divided into a few components:
- Dungeon contains all the 'business' logic
- Entities all of the 'characters' in the game
- Items all of the usable objects in the game
- Resources all of the static content
- Tests unit and component tests

```
dungeon helpers.pv
 init .py
base entity.py
    base.py
    constants.py
    display factory.py
    emoji.pv
    map transform factory.py
    symbolic.py
enemy.pv
hero.py
 init .py
display info.py
  init_.py
key input.py
print helpers.py
utils.py
verification mixin.pv
 _init__.py
spell.py
weapon.pv
     _init__.py
   names.py
    treasures.pv
    credits.txt
    help.txt
    intro.txt
    level1.txt
    lore.txt
    map kevs.txt
test base entity.py
test class dungeon.py
test dungeons helpers.pv
test hero.py
test invalid map.txt
test spell.pv
test valid map.txt
test verification mixin.py
```

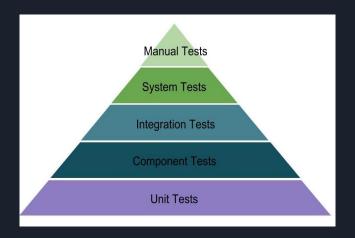
Implementation

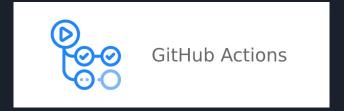
- Written in Python
- Instead of GUI everything is rendered directly in the terminal
- Design patterns Mixin, Factory,
 MVC
- Almost no external dependencies (only argparse and unittest)



Testing and CI

- Good test coverage to ensure functionality of the application
- Unit and component tests
- Used PyTest as test running tool
- Used GitHub Actions to create pipeline for continuous testing





Map Legend

- corridor
 - wall
- 🧝 Hero

- 🔷 Spawn point
- Enemy

- Treasure

- Gate

Gameplay and functionalities

- Use arrows and/ or w,a,s,d to navigate through the maze
- Interactive battles
- Helper commands for easier gameplay
- Respawn mechanism
- Treasures and item/ spell collection



Demo Time