Test Documentation

Note: Each field with a check mark indicates test has been passed

Test - Initialize Game:

When initiating the opening of the game the databases and visuals load .

Approach:

1. After ensuring that your virtual environment is set up. Type “python access.py” into the terminal
2. (Command or Control) + Click on the IP address inside the terminal
3. You should be taken to our game’s homepage
4. Visit each button on the home page

Expected Output:

- Each page should render fully.

Test - Returning Player:

Players should be put into the first level where there is a 0 for their score .

Approach 1 use default data:

5. Upon starting the game go to “NEW GAME” button
6. There will be 2 fields input any of the default players inside the highscores database:
   “Darth Vader”, “Jabba the hutt” or “Alien Invader”
7. Hit returning player button

Expected Output:

- Case “Darth Vader”:
  - Should open up level 2 window
- Case “Jabba the hutt”:
  - Should open up levelEndless window
- Case “Alien Invader”:
  - Should open up levelEndless window

Approach 2 use your own player:

1. Upon opening the game. Hit “NEW GAME” button
2. There will be 2 fields input your username and hit the “Add Player” button.
3. You should be spawned into level1 window.
4. Play a level or a couple (to completion) to generate a score.When finished close the window.
5. Re-open the game and hit “NEW GAME” button , but this time u**sing the same username you used before** hit the “Returning Player” button

Expected Output:

- Depending you should be placed in the level above the last level you played. If the last level you played was the levelEndless, then you will be placed back in levelEndess.

Test - Production System (testing a multitude of scenarios):

When the user is using our grid production system, they have options to place buildings onto the grid. I develop a ruleset and test it to ensure placements are valid and do not interfere with other building placements

- (1) only allow a building to be placed onto the grid
- (2) buildings cannot be placed if they interfere with the location of another building

When the user is using our grid production system, they have options to place Conveyors. I implemented a testing ruleset to prevent bad and conflicting conveyor placements

- (0) only allow conveyor placements if the two selected locations by mouse are valid existing buildings
- (1) cannot connect 2 buildings of same type by conveyor unless they are both constructors
- (2) cannot make copper or steel landing pads an endpoint for a conveyor belt
- (3) cannot make a storage (placeholder for shops and launch pads) be a starting point for a
- (4) prevent conveyor to a constructor that has no recipe
- (5) only let a constructor-constructor if [end constructor] recipe requires [start constructor]
- (6) only allow storage to connect to constructor of a finished product (for launching to David)

Testing the conveyor connections

- (1) 1-to-1 conveyor: ensure a conveyor may send items over from a start building's inventory to an end building's inventory correctly (tested amount)
- (2) 1-to-many conveyor: ensure conveyors can connect one building and branch into several other buildings without a loss of items and a splitting of items from the start building
- (3) many-to-many conveyor: ensure a building acting as both an endpoint and a starting point for multiple conveyors on each side correctly receives each resource and distributes the resources leaving it.

Testing the shop's selling and buying features

- (1) Only allow the selling of existing products in one's inventory

- (2) Allow buying of a product if the funds exist
- (3) Ensure buying a product correctly subtracts from one's money and adds to the shop's inventory
- (4) ensure selling a product correctly subtracts it from the shop's inventory and adds to the user's money

Test Balance and Smoothness

- **In order to make sure that the game is balanced we played through the game.**
  - Fixed all bugs that occurred.
  - Made sure that the game was beatable but still challenging (need several attempts).
  - Made sure that the game ran smoothly.