

Game.py attributes

①

dict all_tiles

contains objects of the Tile class
indexed according to their names.
I.e. "Foyer"

dict all_items

contains damage information for
items indexed according to their
names.

ie.

"Board with Nails"	:	1
"Grisly Femur"	:	1
"Golf Club"	:	1
"Chainsaw"	:	3
"Machete"	:	2

~~int health~~

int player-health

health of player default = 6

int player-attack

player attack default = 1

String ~~item~~ ~~player~~ player-item

the item the player is currently holding.
one of

"Board with Nails" "Grisly Femur" "Golf Club"
"Chainsaw" "Machete"

Game.py attributes

②

String player_location

contains current player location
default "Foyer"

can be ~~"Graveyard"~~ "Graveyard"

"Yard"

"Patio"

~~"Dining Room"~~

"Dining Room"

"Evil Temple"

"Foyer"

Boolean has_zombie_totem

true = have totem

false = no totem (Default)

Devcard devcard_controller

contains reference to Devcard object.

int count_cards

used by update_game_time
default = 0

game.py

__init__()

TODO

define and Load Tile objects
into dictionary - all_tiles

Load item information into all_items

create deckard_controller

display game state.

get_item() return Boolean

TODO

check there is an item in the current
↓ tile

No - return False

Set player-items to item from tile

↓

Set new player attack ($1 + \text{item}$)

↓

remove item from ~~Area~~ Tile

↓

Return ~~True~~ true

game.py

move (string direction)

check if there are zombies in the room

No

Yes



↳ Return false

Set player_location to new room



Call display_game_status.



Return True

end_



check_game_condition()

Check if game_time = 12

No

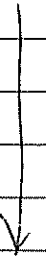
Yes



print "The zombies ate your brain"
end game.

Check if ~~you are~~ player_health ≥ 0

Yes



print "you died"
end game

Return false

game.py

run(direction)

check if there are zombies in Room

Yes

No

↳ Return false



~~check~~ check if direction is ok.

Yes

No

↳ Return false



~~change~~

set zombies in room to zero



set new room to room in direction moved

set player location to new room



health - 1



~~Return True~~

~~Return~~

Call withdraw_card()



Call display_game_status

↳ Return true.

~~Return True~~

cower()

check if there are Zombie in Room

No

Yes

↳ Return False



~~with draw~~

Call withdraw_card()



health + 3



Call display_game_status()



Return true.

game.py

game_display_status()

display this information.

player_location

game_time

if there is an item on the floor

if there is zombies in the room

north

south } is there other Rooms

east }

west

player_health

player_attack

if you have the zombie token.

attack()

~~calculate health lost~~

check if there are zombies in the room

Yes

No

↳ return false



Calculate health lost to game rules

(4 health thing as well)



Set zombies in room to zero



call check_game_end_condition



call game_display_status



return True.

game.py

get_totem()

check player is in the evil temple

Yes

No



Return false

set has_zombie_totem to True



~~Return True~~

call display_game_status,



Return True.

bury_totem()

check player is in "Graveyard"

Yes

No



Return false

Check there is no zombies in room

Yes

No



Return false

Check player has totem

Yes

No



Return ~~True~~ False

Print "You win"



end game

Game.py

withdraw-card()

call Deucard. ~~withdraw-card()~~ pick-card()



returns tuple of 2 values

[0], [1]

explanation

0, an item

1, an number of zombies

2, a message



if 0 set item in current room



if 1 set number of zombies in current room



if 2 print message on screen



call update-game-time()



call check-game-end-condition()



~~end.~~ end. (No return value)

game.py

update_game_time()

increase card_count by 1



if card_count > 8 increase game_time by 1

Yes No.



↳ return.

increase game_time by 1



set card_count to 1



return.