System and Object design document

For Capstone

Project title: Happiness plus 1

By: Jeffrey Redmond

SID 200335711

Table of Contents

Level Design

Lofi level design

The lofi design of levels one through six began at the beginning of the project and was completed before anything else.

Figure 1
LoFi level design of the Second level
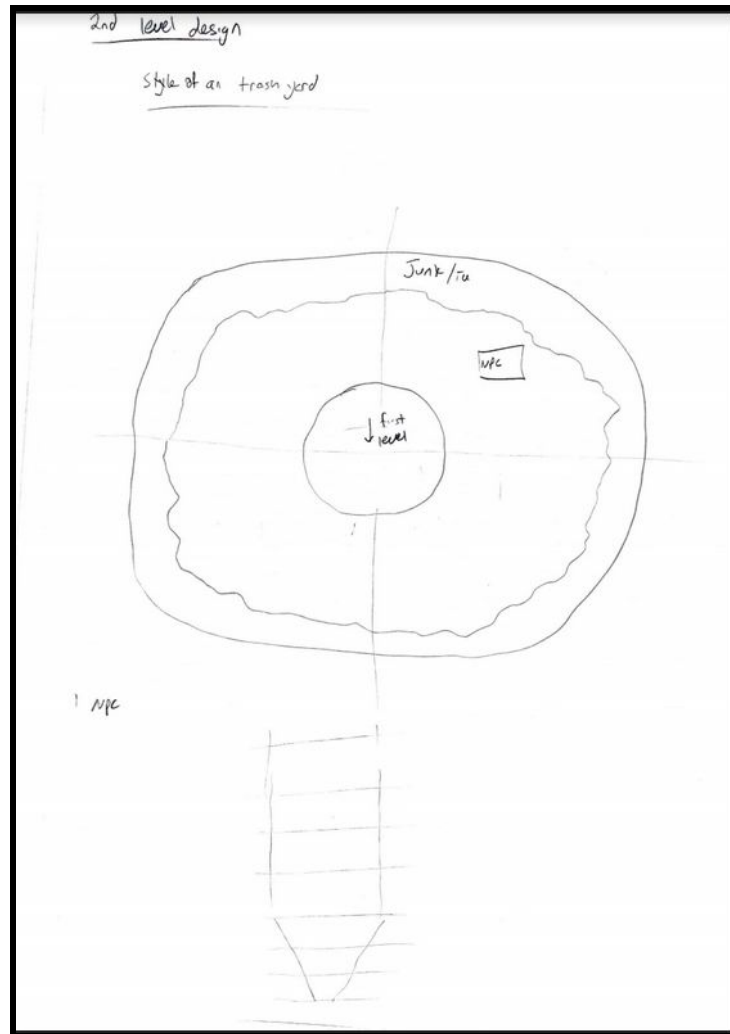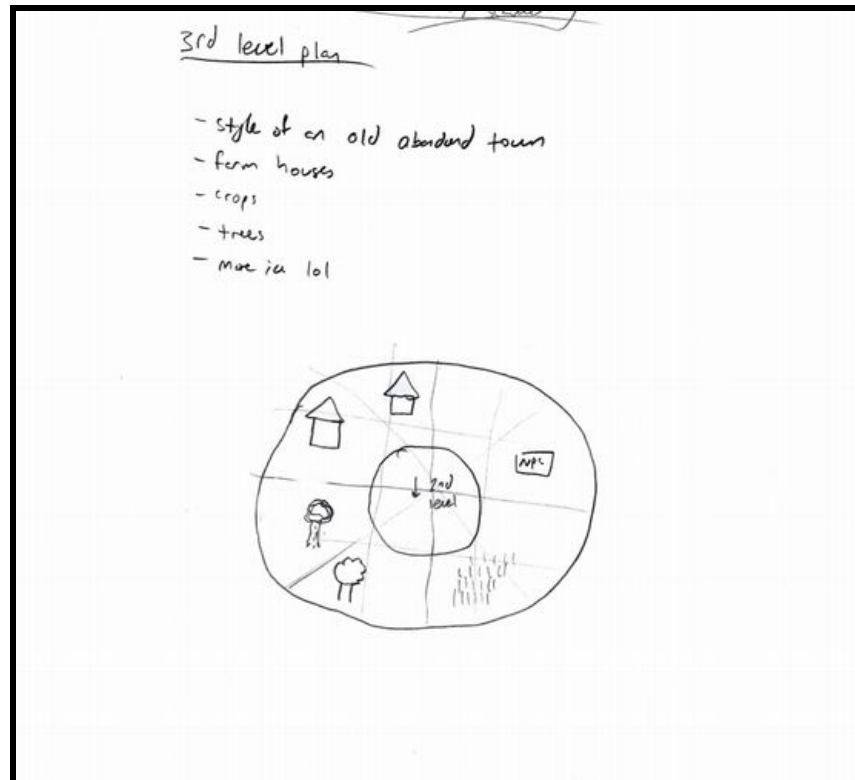
Figure 2
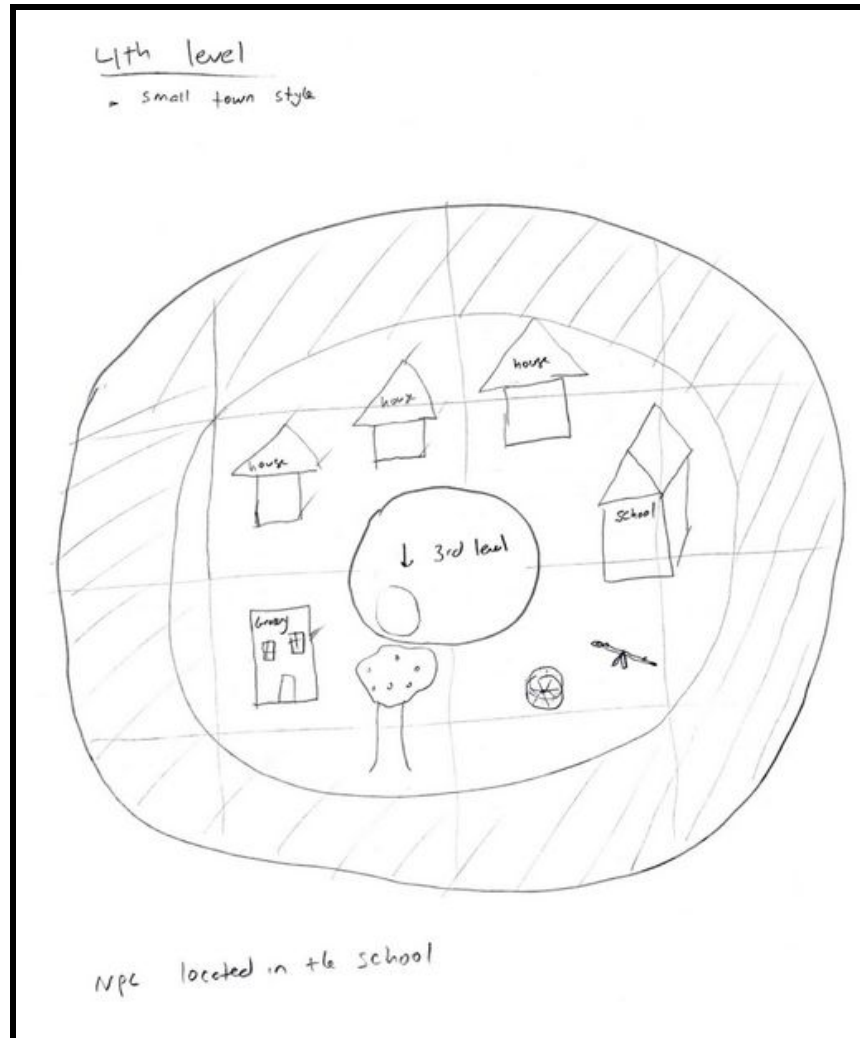LoFi Design of the third level

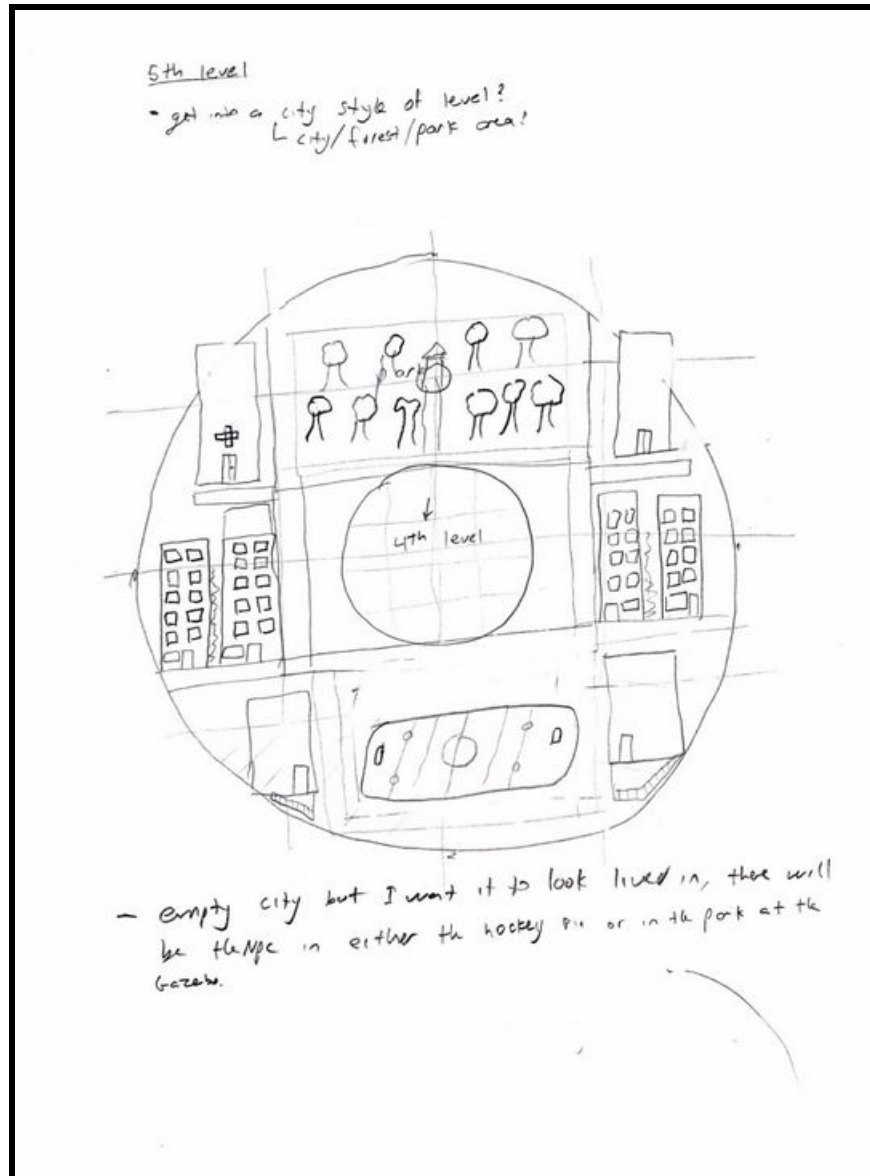Figure 3
LoFi level design of the fourth level

Figure 4
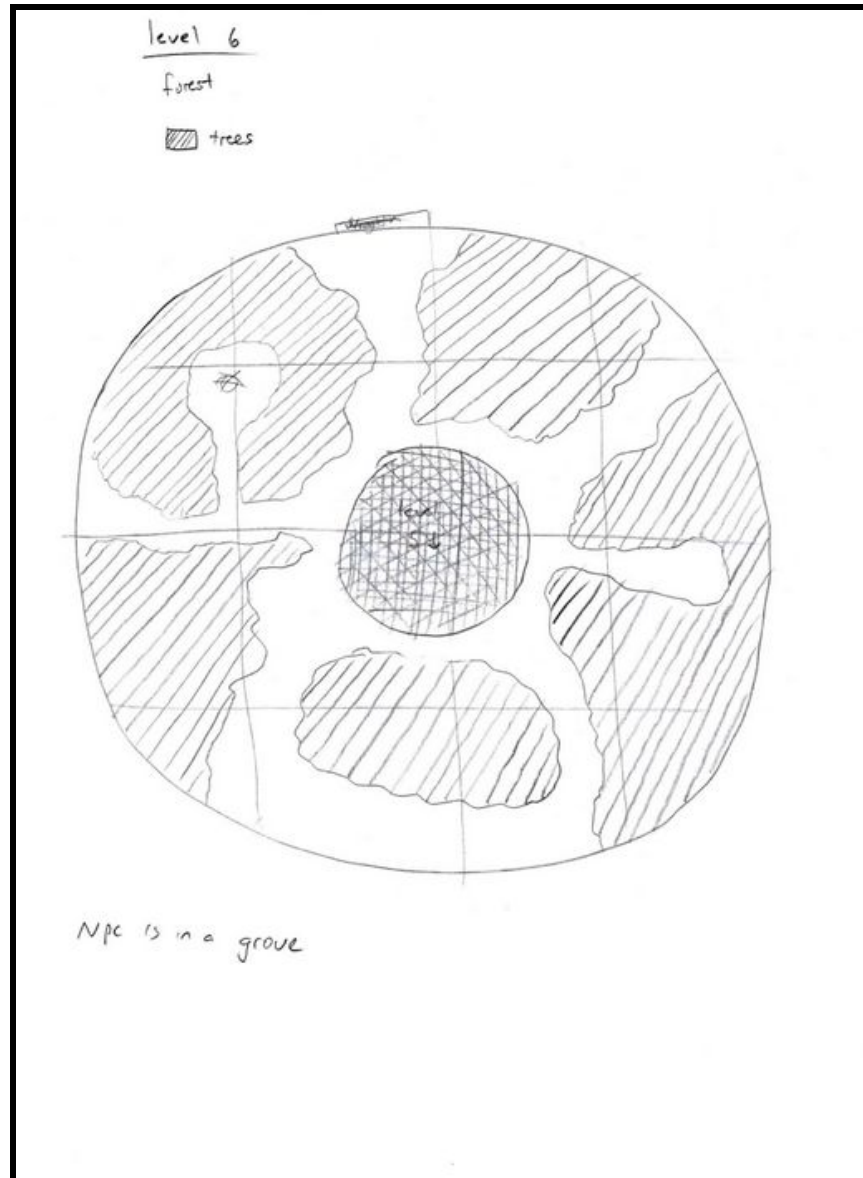LoFi level Design of the Fifth level

Figure 5
LoFi level design of the sixth level

Character Creation and Descriptions

The character creations were done in the fall semester at the same time as the other planning phases.

Character name: Ash
Character description:
- Female

- brown hair (long)
- blue eyes
- Medium build

Character memories: can remember sunshine and the smell of flowers

Character task: Give a flower

Appears in first level.

Character name: Dusty

Character description:
- male
- blonde/brown hair (medium/long)
- orange beard
- Blue eyes
- larger build

Character memories: can remember building with his hands

Character task: Find a knife and a chuck of wood

Character name: Elizabeth

Character description:
- Female
- Blonde hair (long)
- Blue eyes
- Slight build
- Wears a scarf

Character memories: can remember teaching

Character task: bring a apple

Character name: Jerry the bear

Character description:
- Bear

Character memories: can remember learning

Character task: This is an open task that i will figure out when i get to creating the character

Character name: Cat

Character description:
- Female
- Brown dark hair (medium/long)
- Brown eyes

Character memories: can remember someone being there to talk to

Character task: Just have a visit

Character name: Allen

Character description:
- male
- Black hair (medium/short)
- No facial hair
- Brown eyes
- Glasses?
- medium build

Character memories: can remember terrible jokes
Character task: Make someone laugh/find a book of puns

Character name: boss
Character description:
- male
- Bald
- Largest build

Character memories: can remember being being fit
Character task: Find a mirror/and or some weights

Character name: marlon/berry

Character description:
- male
- Black hair (medium)
- Black beard
- Larger build

Character memories: can remember the joys of words
Character task: Find a book

Character name: marvy
Character description:
- Male
- Brown hair (short)
- Blue eyes
- scruff/brown
- Large build

Character memories: can remember having someone to spend time with
Character task: Find a friend

Character name: pil
Character description:
- Male

- Brown hair (short)
- Blue eyes
- Larger build
- No facial hair

Character memories: can remember a cat (pickles)
Character task: Find a stuffed cat

Character name: tina
Character description:
- Female
- Dark brown hair (medium)
- Slight build

Character memories: can remember a cat (peaches)
Character task: Find a stuffed cat

Character name: Kurt

Character description:
- Male
- Dark brown hair (short)
- Dark brown beard
- Medium build

Character memories: can remember a warmth inside from something
Character task: Find some tea

Scripts

After the levels and the non playable characters were roughed out the process started on creating the first version of the script for the non playable characters and the creation of the first version of level puzzles.

Excerpt from the first version of the script listing out the first two levels.

Level 1 script

Screen fades in from black

<a dot of light floats above the characters head in a room that is dark grey> side note ("the light was sent by someone wanting you to feel better")

The player is able to move around now have everything dark blue/ black. Want it to be very cold feeling. There is ice everywhere.

As the player progresses they come to a area that is a bit brighter , there is a npc there that you can talk to, it says all the dialog.

A Character is sitting, holding legs to its chest. The character will be hazy/fuzzy looking colors will be muted.

The character is ash

Interact with ash:

Dialog box pops up
Ash: "...."
Ash: "huh"
Ash: "Are you real"
Ash:"...."
Ash: "you are real"
Ash: "it's been dark so long"
Ash:"..."
Ash: "how long have i been here"
Ash: "huh?"
Ash: "who am i?"
Ash:"... i "
Ash: "i can't remember"
Ash: "i remember..."
Ash:"..."
Ash: "a smell"
Ash: "it was beautiful..."
Ash: "..."
Ash: "wait, what? What was i saying"
Ash: " it's so cold"
Repeat ->Ash: "so cold"
   -    <puzzle one here get flower here>
Flower is given to ash:
Ash: "what's this?"
Ash: "is that"
Ash: *sniffs*
Ash: "ha... " *picture of character gets less fuzzy and they smile with a tear in their eye
Ash: "a flower"

Ash: * dialog box exits and the sprite of ash changes from the sitting character to a clearer one that has more color.
Ash: "Ash… my name was.."
Ash: "My name is Ash"

Ash: "I remember i used to live in the sun, surrounded by flowers, and with friends"
Ash: "then i remember… all my friends… they moved away"
Ash: "why? Why did they have to move away? I was... "
Ash: "I was sad, so sad."
Ash: "then… i don't know, i remember falling, i was cold and sad"
Ash: "for so long. But you helped me remember the sun and flowers"
Ash: *smiles* "thank you"
Ash: *disappears in a pillar of golden light

On screen it pops up
"Happiness plus one"
Your character is engulfed in a pillar of blue light and drifts upwards and the screen goes to black

Dark screen dialog: dunno yet - explain what is going on

Level 2 script

The screen fades in from black and the player is in a new area
Character explores level
Finds npc sitting beside a pile of trash

Allen: "..."
Allen: "hello?"
Allen: "who is there"
Allen: "what do you want?"
Allen: "me?why would you want to know who i am?"
Allen: "if i knew i would tell you"
Allen: "but"
Allen: "i can't remember"
Allen: "i'm sorry i couldn't help"
Allen: "if it would make you feel better i could tell you a joke"
Character: "yes"
Allen: *breath in*
Allen: "......."
Allen: *looks around*
Allen: "Why … can't ….. I ….. Remember any"

Allen: *starts crying"

Allen: *sobs*
Allen: repeat last response
<insert puzzle here get joke book>
Give book to allen

Allen: *sob* "huh?" sniff sniff
Allen: "what is this? Is this for me?"
Allen: *reads the book"
Allen: "heh"
Allen: "he he he"
Allen: *insert joke here* allen reads the joke
Character : chuckles
Allen: *looks up* "did you like that one?"
Allen: *second joke here*
The joke is terrible but you manage to force out a laugh
Allen: "what is this feeling? It's been so long since i've felt this"
Allen transforms into the clearer version of himself
Flash of light or some such shit
Allen: " i remember now"
Allen: "my name is allen"
Allen: "i remember people, i was there and i… everyone was laughing, i was so happy"
Allen: "but as time went on i ran out of jokes and only started saying puns"
Allen: "everyone started hating me and left"
Allen: "then there was darkness… sadness and i was here"
<more dialog>
Allen: "you saved me from this, and for that i can never repay you"
Allen: "thank you"
Looks like he is about to ascend into the golden light. Camera pans over to the right to show a
pizza slice *allen looks over to it*
Allen: "Well i guess that you could say that it's been a slice "HAHAHAHAHA"

Possible idea: the gold pillar turns from gold to red, and allen starts lowering, the pillar burst into
flames and allen goes
Allen: "AAAAAAHHHH FIRE"
Allen: "AAAAAAAAAHHHH OH MY GAWD I TAKE IT BACK IM SORRY"
Fire pauses for a seconds then goes back to gold and allen rises away.

You feel uncomfortable after the display of holy wrath, but are happy that allen is okay

On screen it pops up
"Happiness plus one"

Your character is engulfed in a pillar of blue light and drifts upwards and the screen goes to black

Storyboarding

Lofi storyboarding

The storyboards were made to get a feel for how the main player would interact with each npc and set the stage for how the room would look before any room were created. This was done with the hope of quickly laying out interactions and then as the project developed insert dialog into the scenarios depicted in the storyboards.
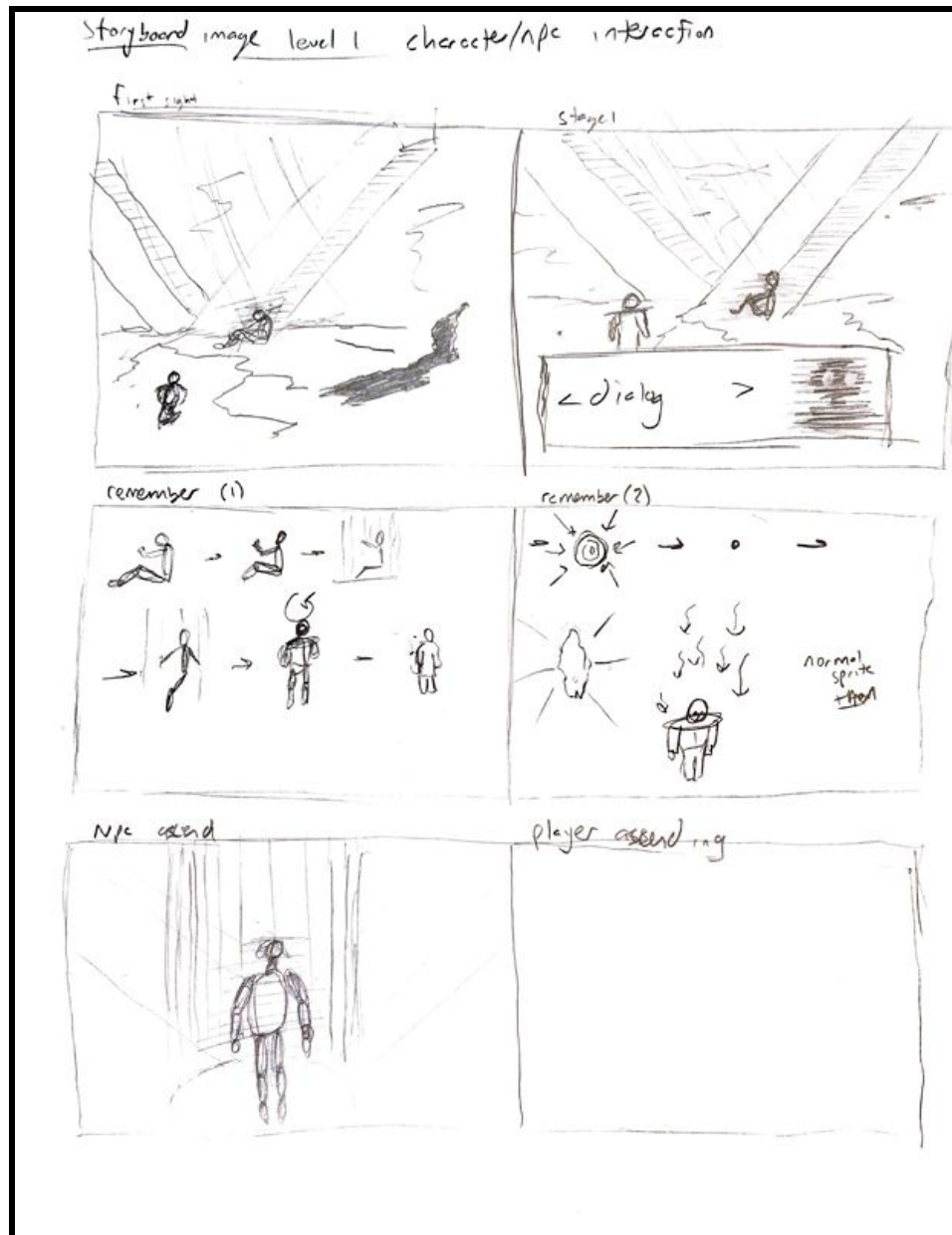
Figure 6
Storyboard for Level one
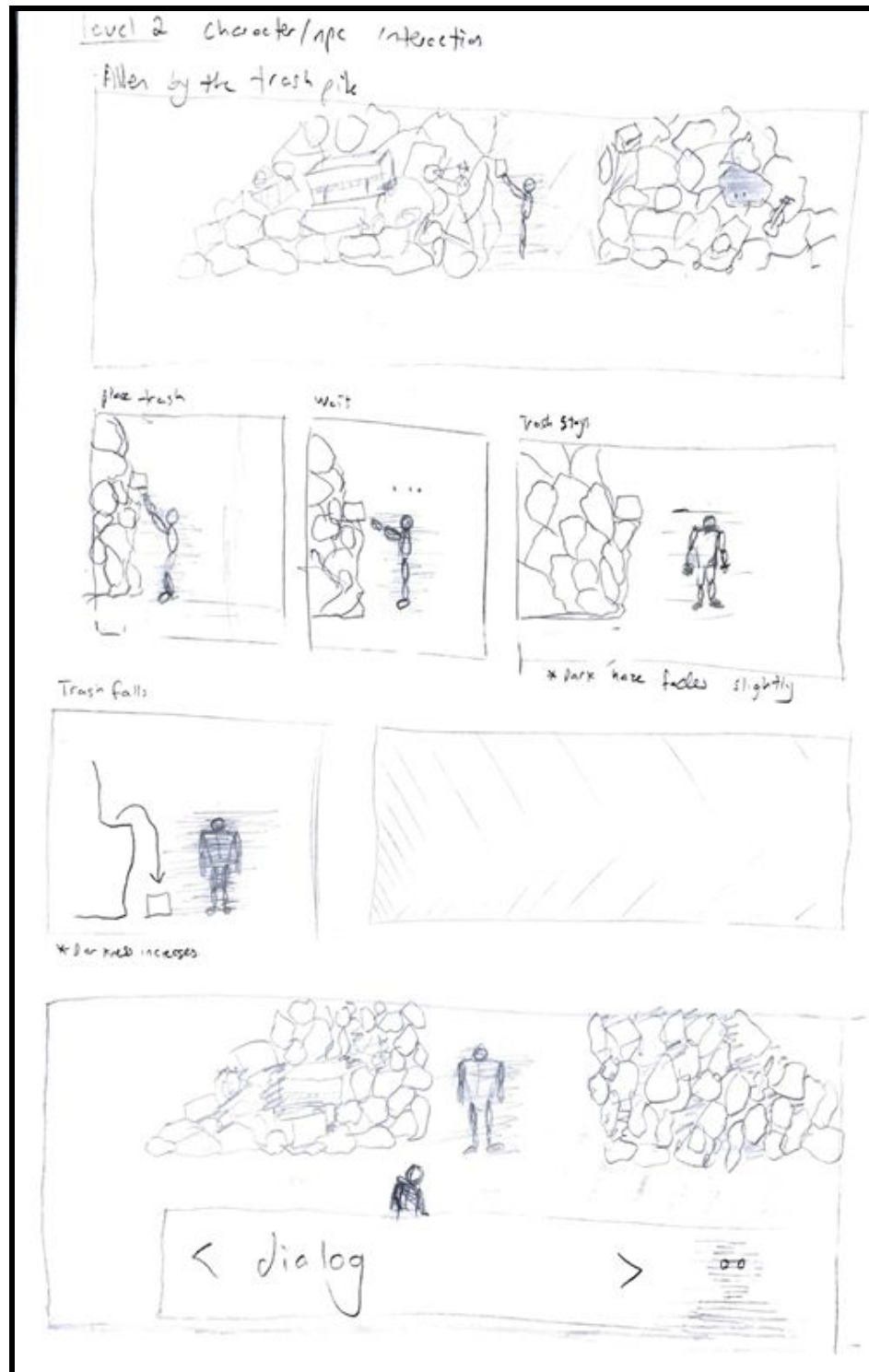
Figure 7
Storyboard for level two

Figure 8
Storyboard for level three

Player then needs to find a crank to work the crank

Level 4

NPC: -elizabeth
      -Beer Bella   > 2

SCHOOL

Top down View
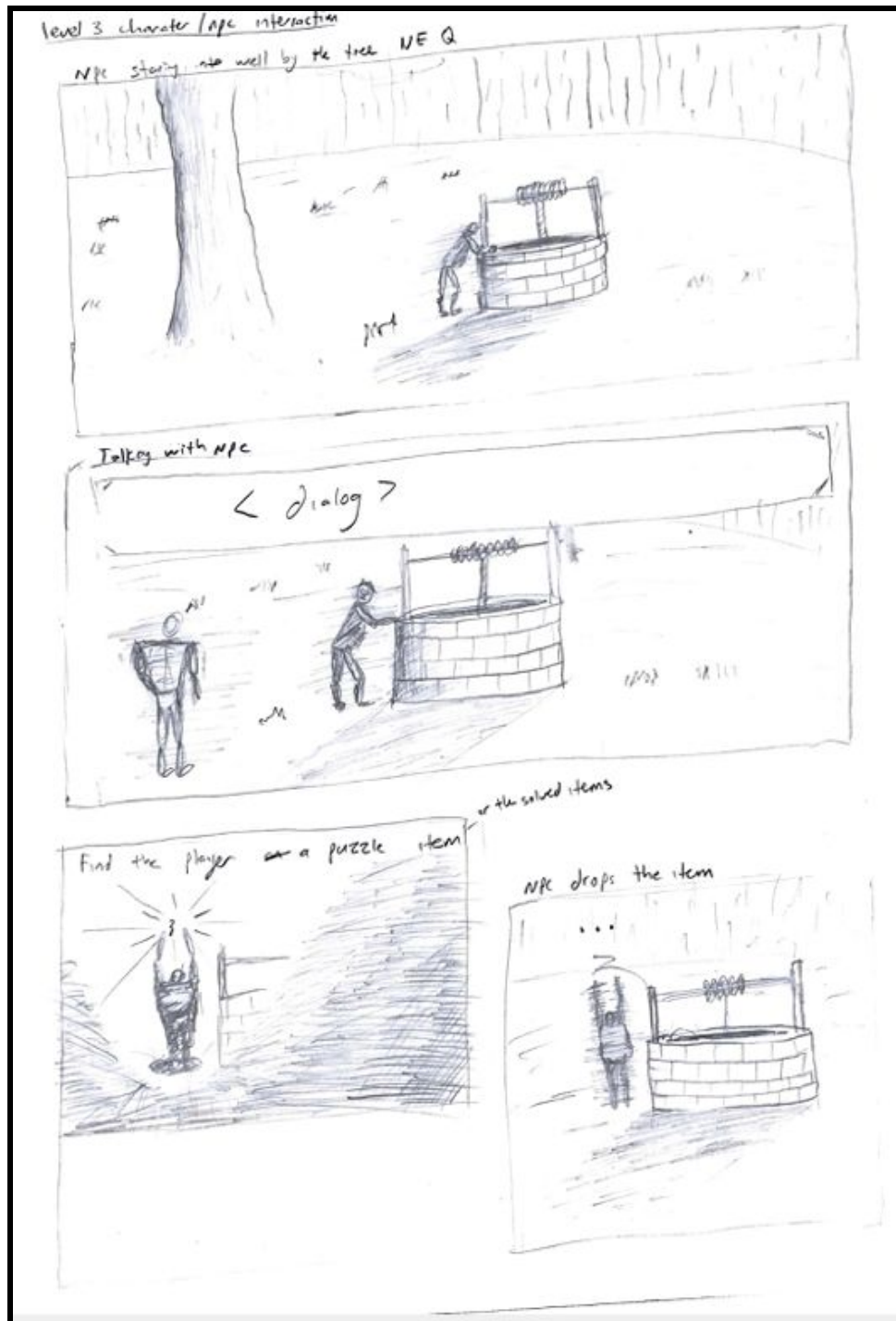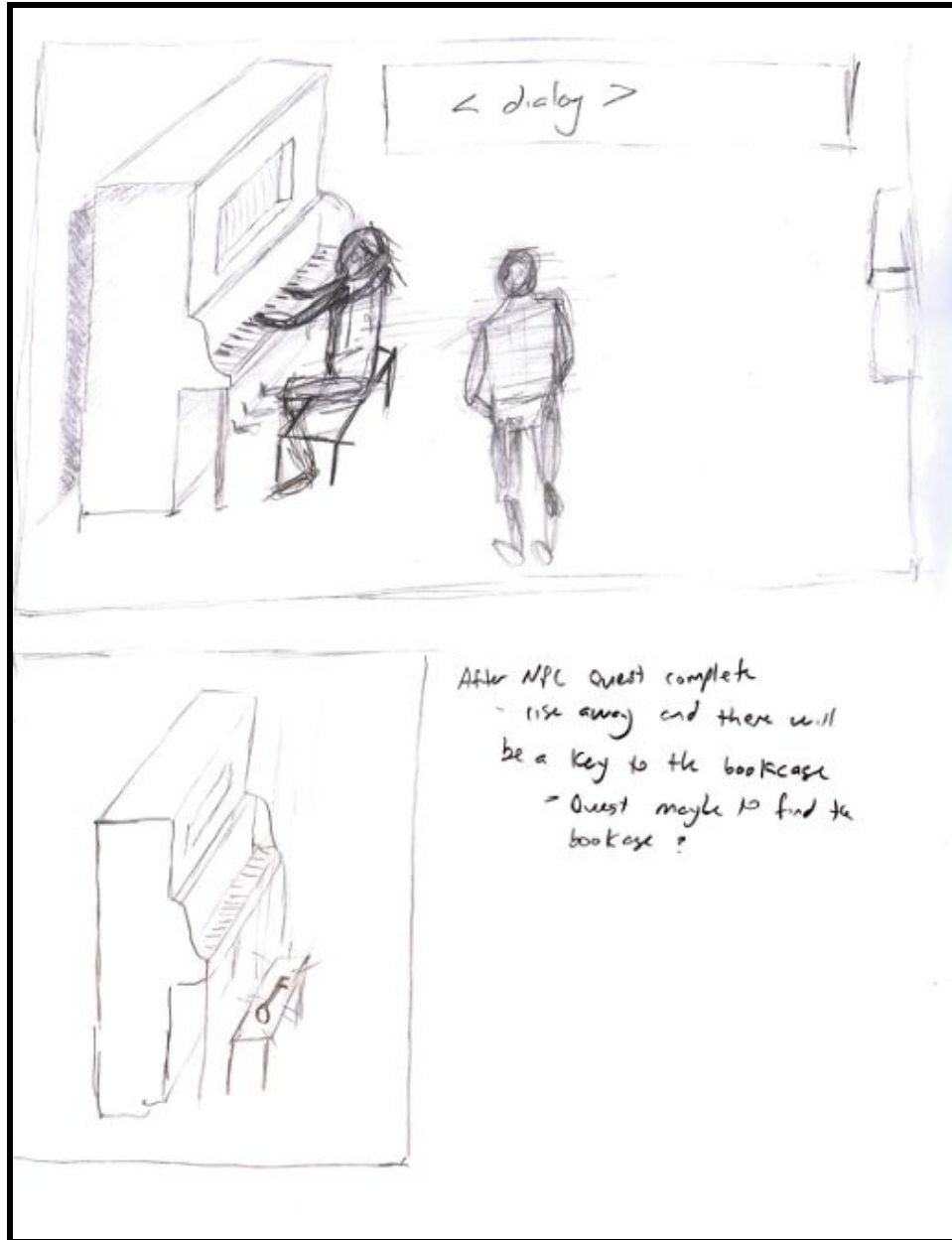
Figure 9
Storyboard for level four part one

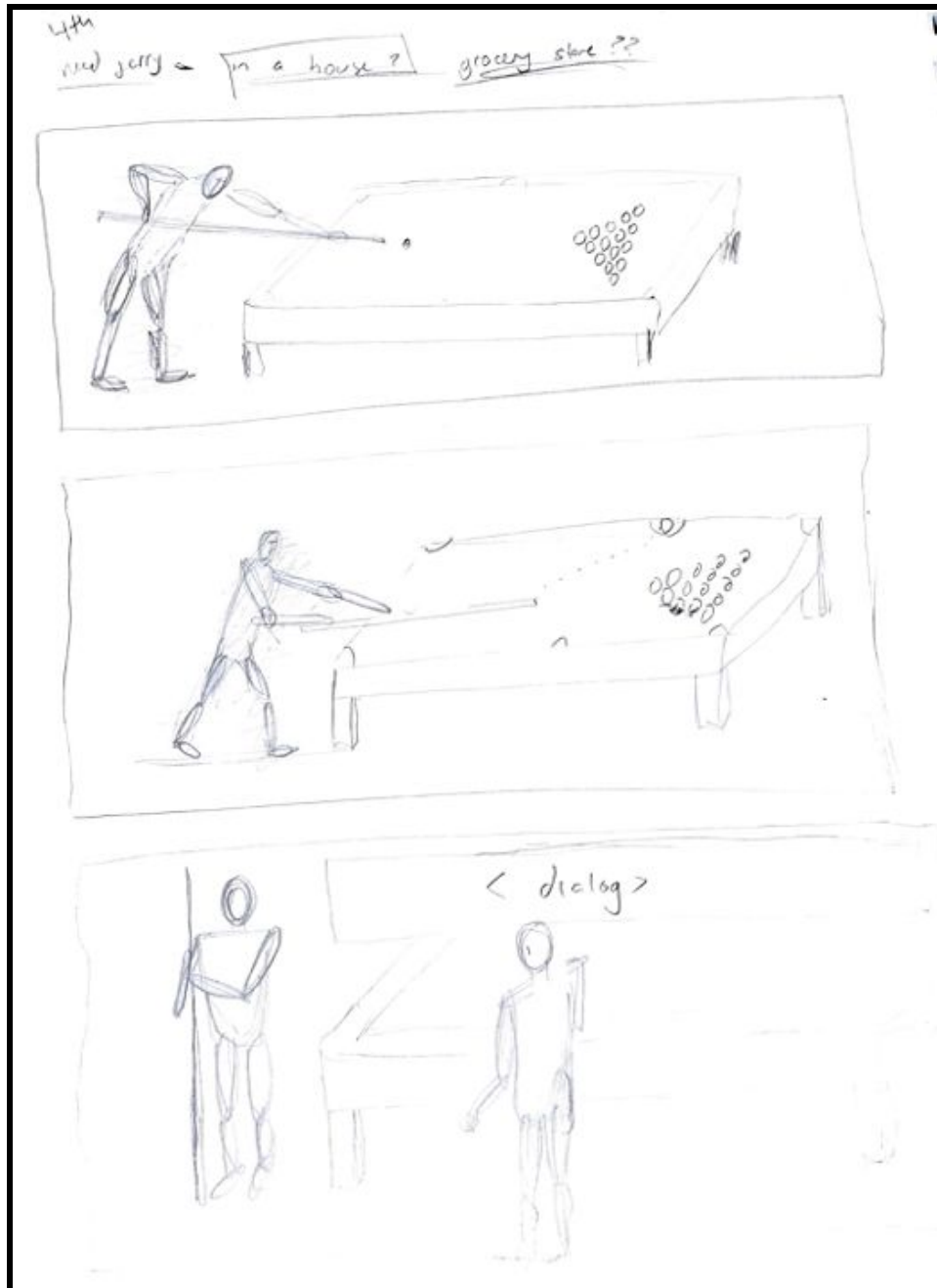Figure 10
Storyboard for level four part two

Figure 11
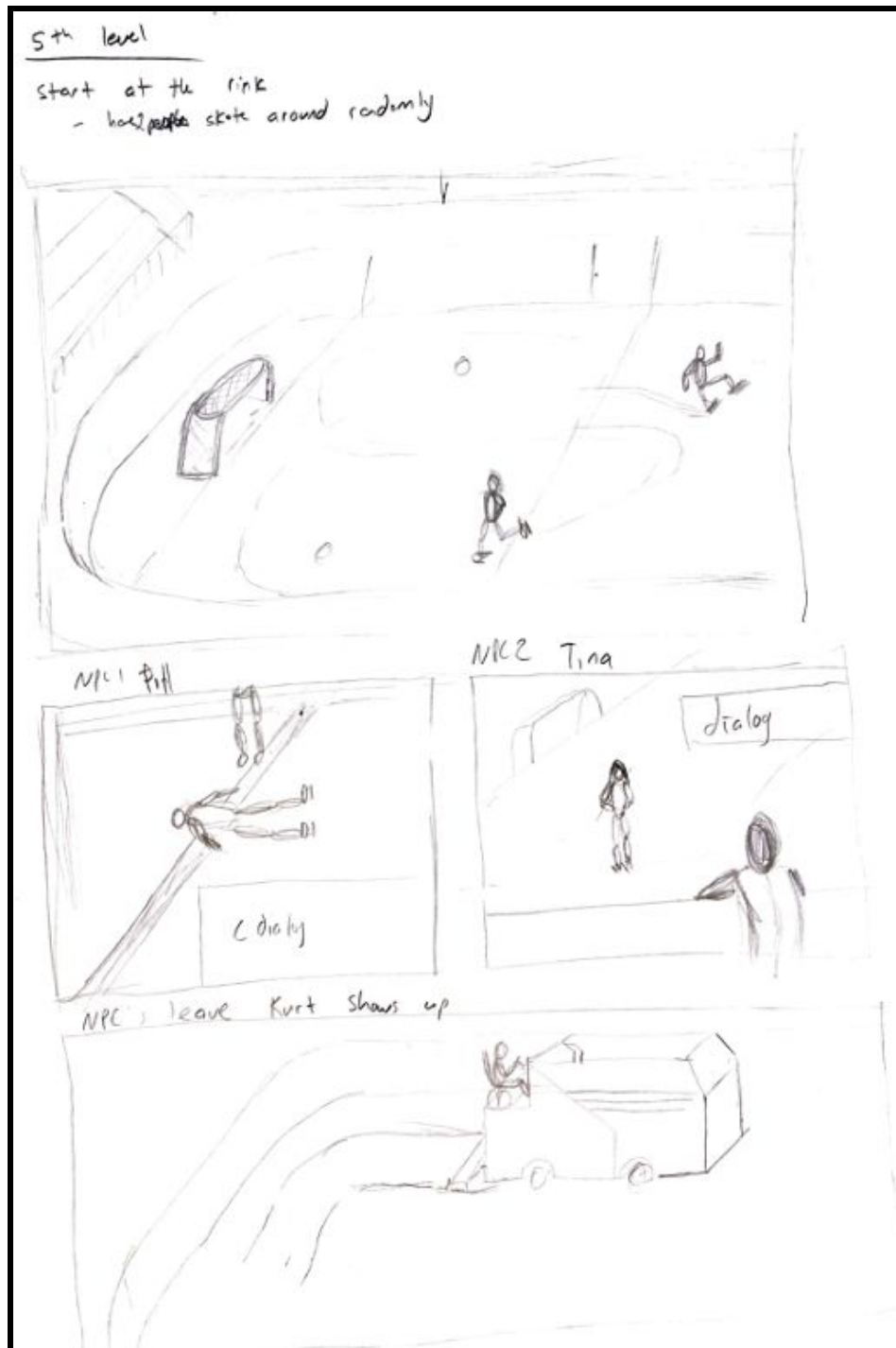Storyboard for level four part three
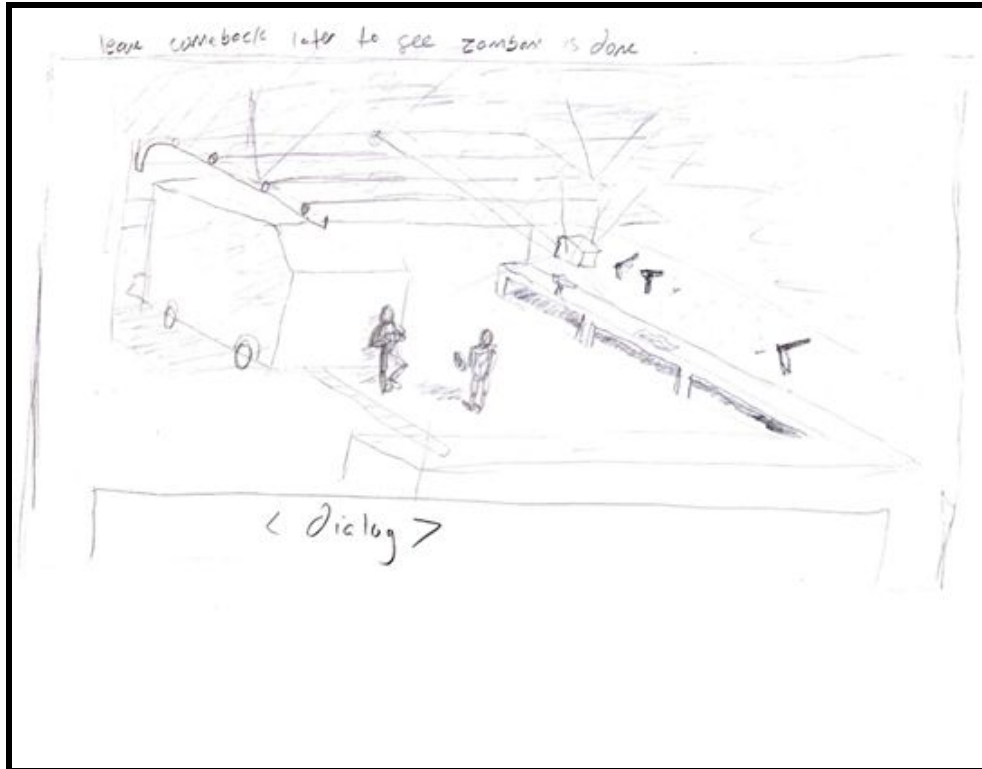
Figure 12
Storyboard for level five part one

Figure 13
Storyboard for level five part two

Figure 14
Storyboard for level six

Puzzle creation

Puzzle writeups

Puzzles began with a jot note version describing the items and the steps to completion.

Puzzle one

Find flower
- Frozen in a ice crystal
- Need to melt ice
- To melt ice need torch
- Need to find torch
- Torch needs stick, rag, fuel
- Get stick from tree
- Get rag from beside ash
- Get fuel from pool that smells bad
- There will be a small fire by ash to start
- Once all items are gathered the torch is built
- Take to crystal, use to melt ice
- Take flower and then give it to ash
  Lofi creation of puzzle items
  Highfi creation of puzzle item

Puzzle two

- Need to get a joke book for allen
- The joke book is located up on a stack of garbage too high for the character to reach
- Need to find a boxing glove arrow and a bow to knock it down.
- Boxing glove
    - Find the boxing glove in a chest?? Or on a statue of mike tyson
- Arrow
    - Find stick
    - Need feathers as well
        - For the feathers find a pillow
            - Find the pillow on a trash pile (it's a body pillow with a skeleton on it .... spooky)
            - To get feathers out of the pillow need to use a knife to cut open pillow

- Find knife sticking out of the snow
- Bow
    - Need bow
        - Find a crossbow
    - Need string
        - Take laces out of a shoe
        - Use second lace to tie to the arrow

- Use the gun mechanism from the last game
- Shoot down the book

Puzzle item LoFi design
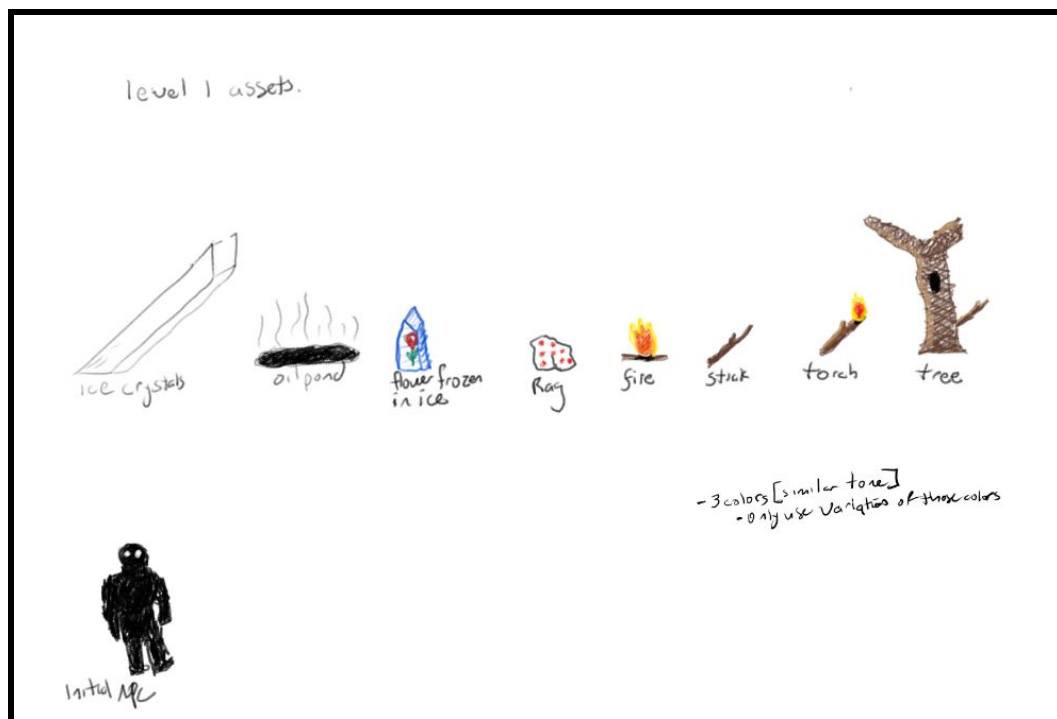
Figure 15
Level 1 LoFi puzzle items

Puzzle item sprite design

Figure 16
Level 1 puzzle item sprites

Game object interactions and structures

This following section attempts to show the different object in the game and how they communicate together to allow the game to work.

Cutscene controller objects

The cutscene controller object allows the developer to quickly create cutscene objects using a list of pre made scripts. The triggers for these cutscene can be either one time use or permanent. As well there is the option to check for certain flags and or inventory conditions and the cutscene will only progress if pre stated conditions are met. This makes the cutscenes quite versatile in what they can do.

Figure 17
Cutscene controller objects

Item description diagram

The item description object allows the developer to place interactable object in the game that will allow the player to read a text box that pops up. This object is also able to have numerous text arrays that can be incremented though the use of either a trigger or a string command typed into the text array.

obj_cutscene_trigger

t_scene_info (array)
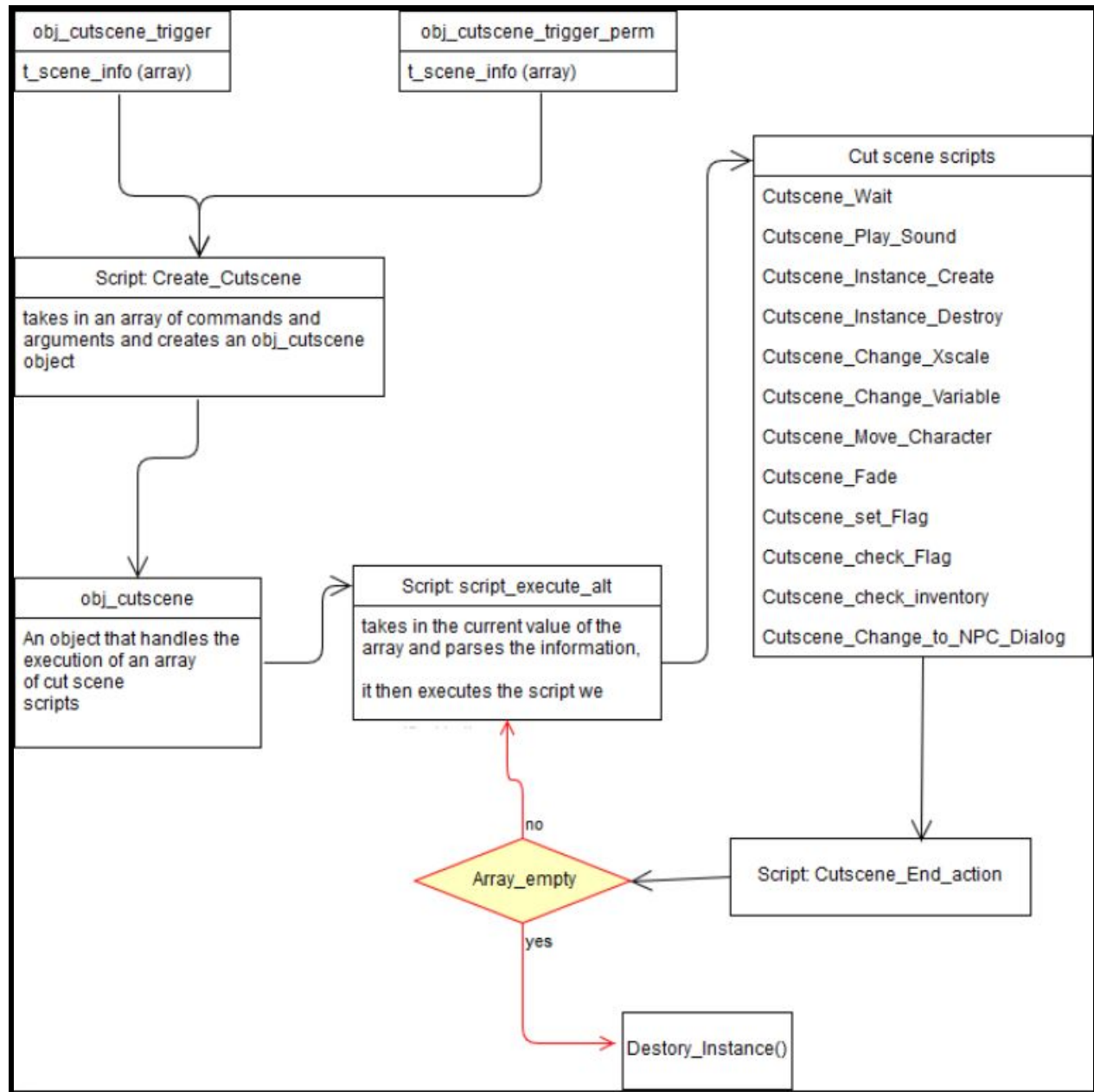
obj_cutscene_trigger_perm

t_scene_info (array)

Cut scene scripts

Cutscene_Wait

Cutscene_Play_Sound

Cutscene_Instance_Create

Cutscene_Instance_Destroy

Cutscene_Change_Xscale

Cutscene_Change_Variable

Cutscene_Move_Character

Cutscene_Fade

Cutscene_set_Flag

Cutscene_check_Flag

Cutscene_check_inventory

Cutscene_Change_to_NPC_Dialog

Script: Create_Cutscene

takes in an array of commands and arguments and creates an obj_cutscene object

obj_cutscene

An object that handles the execution of an array of cut scene scripts

Script: script_execute_alt

takes in the current value of the array and parses the information,

it then executes the script we

no

Array_empty

yes

Script: Cutscene_End_action

Destory_Instance()

Figure 17
Cutscene controller objects

Item description diagram

The item description object allows the developer to place interactable object in the game that will allow the player to read a text box that pops up. This object is also able to have numerous text arrays that can be incremented though the use of either a trigger or a string command typed into the text array.
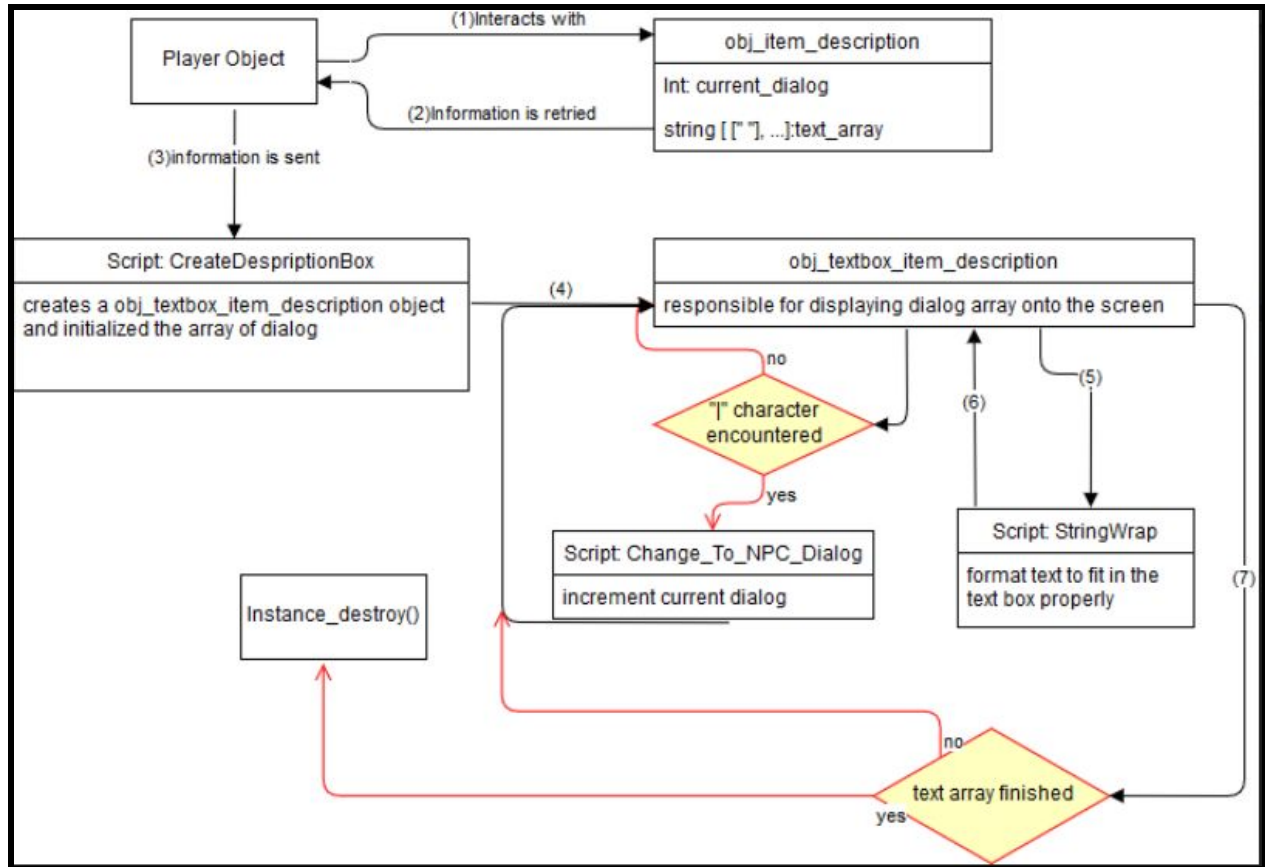
Figure 18
The item description diagram

Item interaction object

The interaction object first displays the description of the object (this can be any number of lines). Second after that message is displayed go into interaction phase depending on the variables set in the item interaction object the game will ask the player if they want to interact. If the player selects yes and there is no required item the inventory will receive the resulting item. If there is an item required then the game will check the players inventory and depending on if they have the required item the will get the resulting item.

Figure 19
Item interaction object diagram
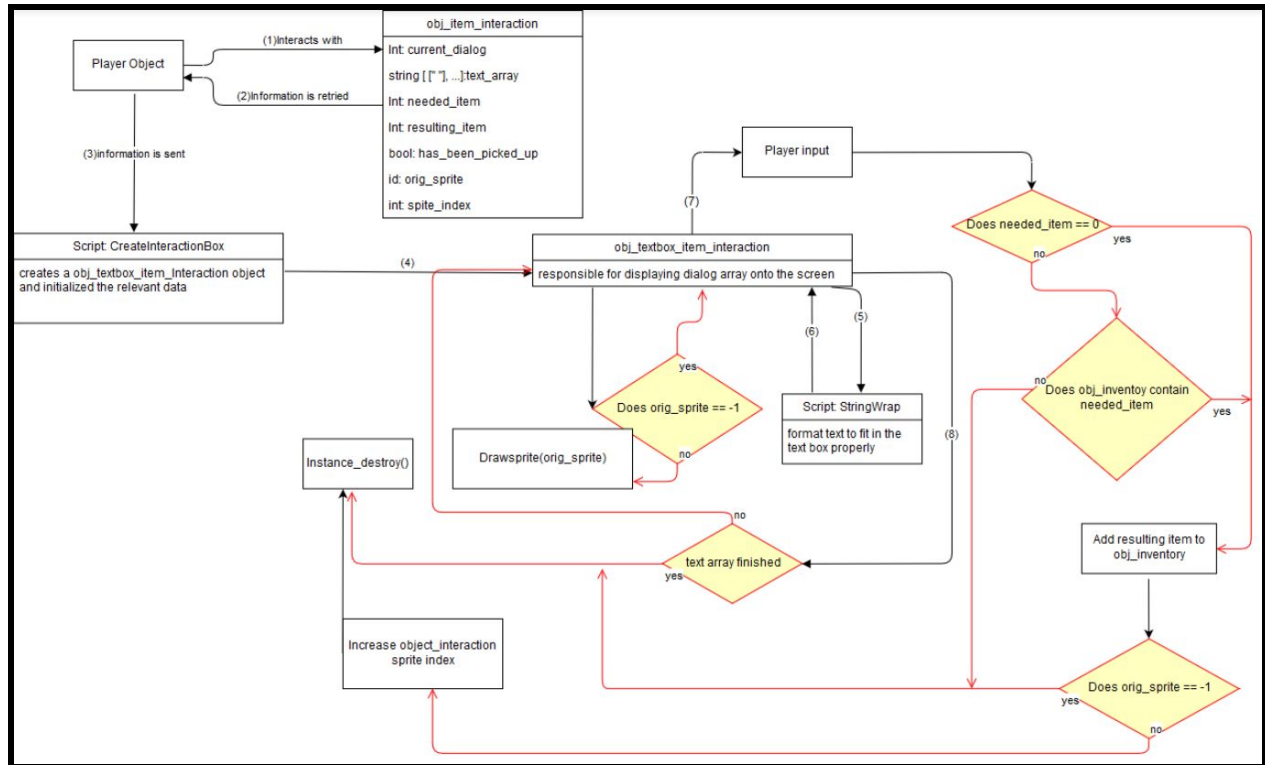
NPC textbox diagram

The obj_textbox is the object in place that is responsible for displaying dialog between the player and NPC's.

Figure 20
NPC textbox object diagram
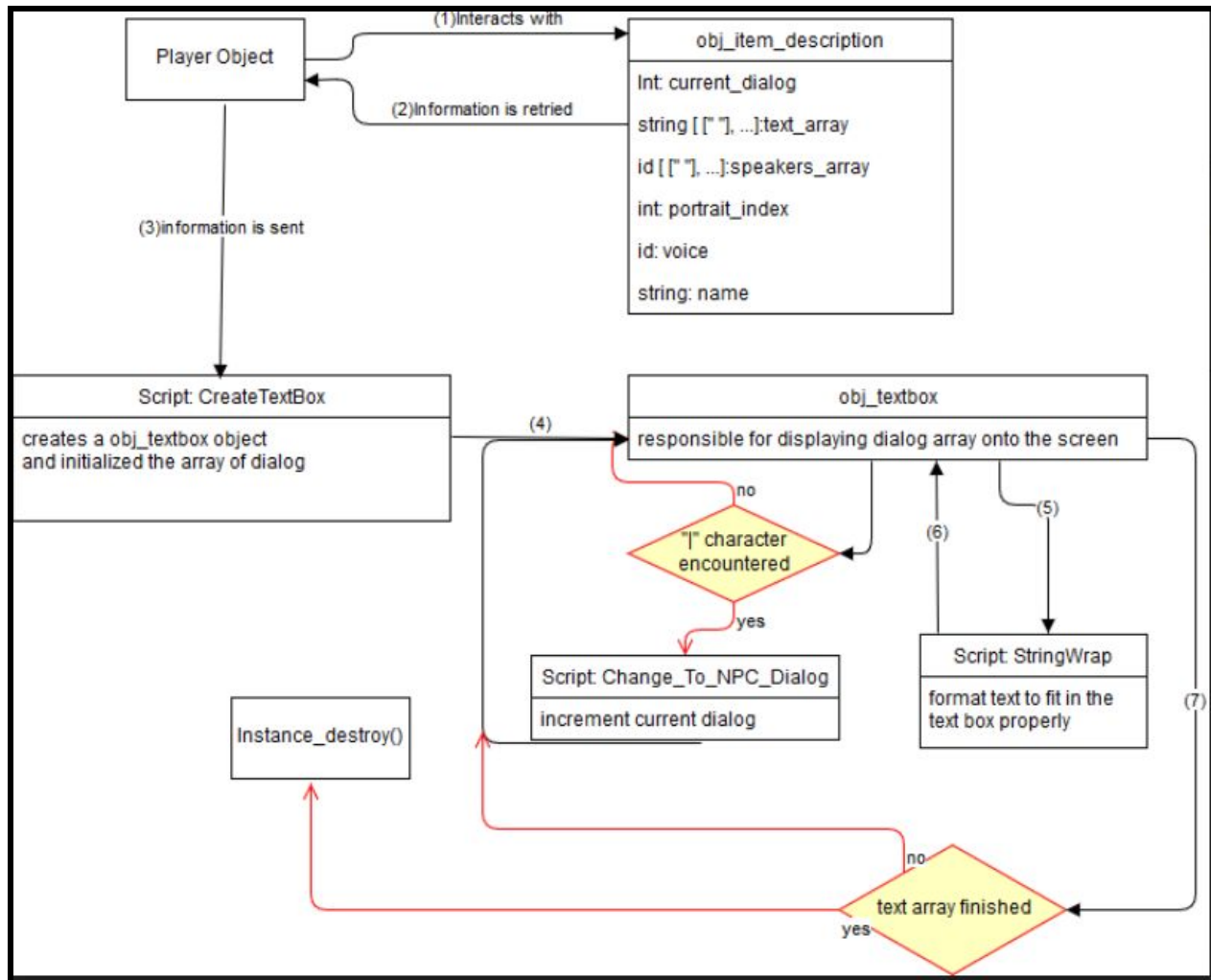
Save and Load diagram

The save system works by fetching information from other game objects and formatts them first into a ds_map and then into a json object that can be saved to a file. The load system works in reverse of the save system, in the event of the game loading from the tile screen the load system will store the other game objects in a buffer in the game object. Once the load system navigates to the desired room in the game each object will initialize its variables using the information from the Game object buffer. The other object will only do this on creation when the game has been loaded from a file.
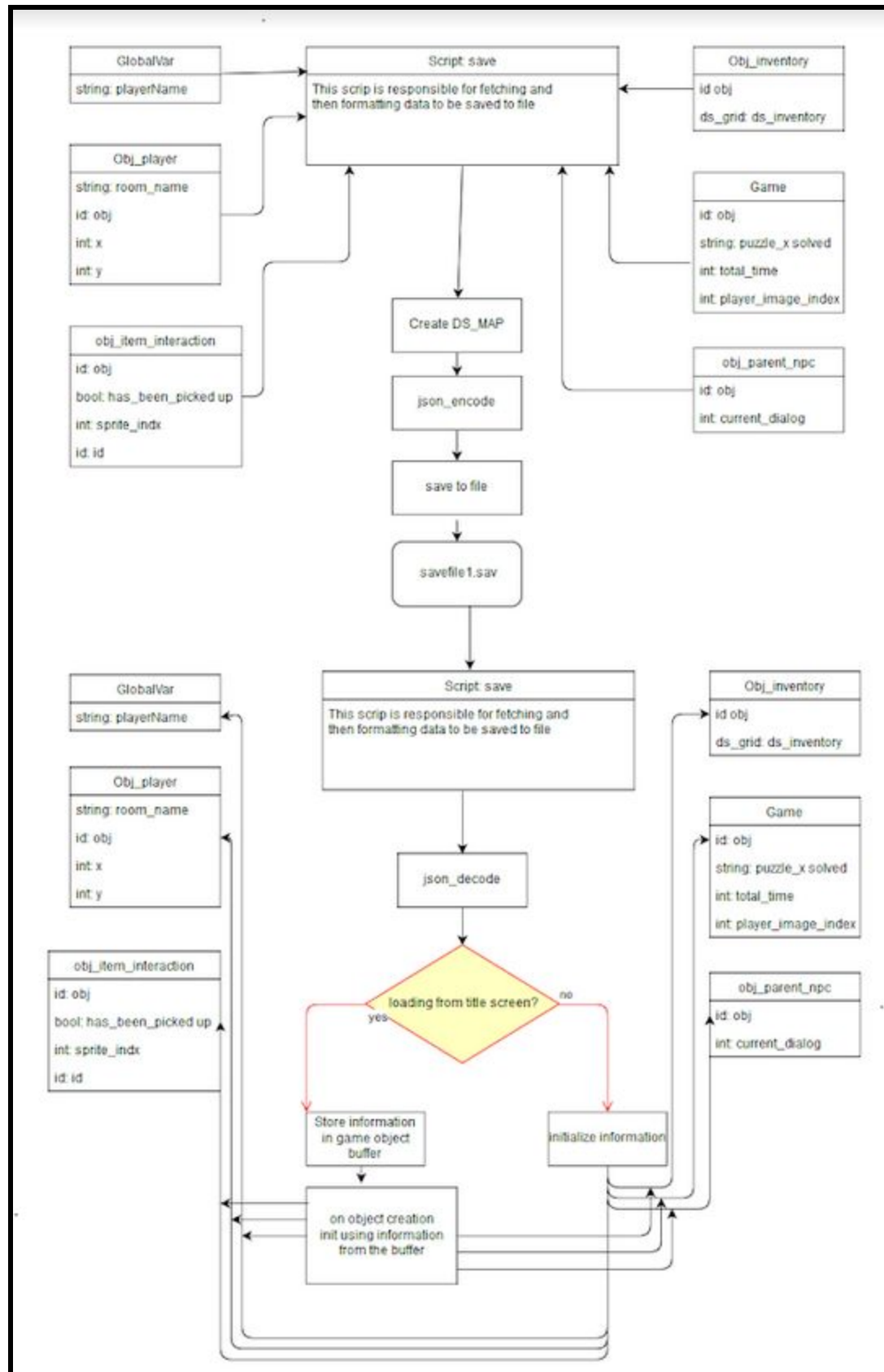
GlobalVar

string: playerName

Script: save

This scrip is responsible for fetching and then formatting data to be saved to file

Obj_inventory

id: obj

ds_grid: ds_inventory

Obj_player

string: room_name

id: obj

int: x

int: y

Game

id: obj

string: puzzle_x solved

int: total_time

int: player_image_index

obj_item_interaction

id: obj

bool: has_been_picked up

int: sprite_indx

id: id

Create DS_MAP

obj_parent_npc

id: obj

int: current_dialog

json_encode

save to file

savefile1.sav

Script: save

This scrip is responsible for fetching and then formatting data to be saved to file

GlobalVar

string: playerName

Obj_inventory

id: obj

ds_grid: ds_inventory

Obj_player

string: room_name

id: obj

int: x

int: y

Game

id: obj

string: puzzle_x solved

int: total_time

int: player_image_index

json_decode

obj_item_interaction

id: obj

bool: has_been_picked up

int: sprite_indx

id: id

loading from title screen?

yes   no

obj_parent_npc

id: obj

int: current_dialog

Store information in game object buffer

initialize information

on object creation init using information from the buffer

Figure 21
Save and load script functionality diagram

Inventory Object

The inventory object will not initialize communication with other objects itself. Instead other objects will retrieve information from the inventory to use. The inventory object is a isolated system that takes care of keeping track and displaying:
1) The players current inventory
2) The names, descriptions, and sprite indexes of all inventory items
3) The current items in the crafting array
4) The list of possible item crafting combinations

When the inventory screen is open the inventory object manages player inputs and allows for item crafting. This object will display the name, descriptions and sprites for current items in the inventory.

Object Inventory
ds_gird: ds_inventory
ds_grid: ds_inventory_item_description
ds_grid: ds_inventory_crafting
ds_grid: ds_inventory_crafted_combos
enum: item
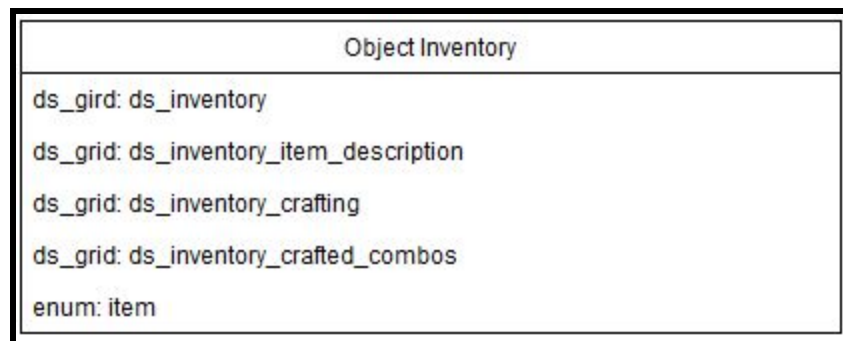
Figure 22
Inventory object variables

Depth controller object relation diagram

The depth controller object works by getting the ID's and y locations of every child of the obj_wall. It then sorts them from smallest to largest, after they are sorted it will go through the DS grid of instances and call the draw event for the objects with the smallest y values first. This allows for the illusion of depth since object further away (higher up on the screen) will always be drawn under the lower objects.
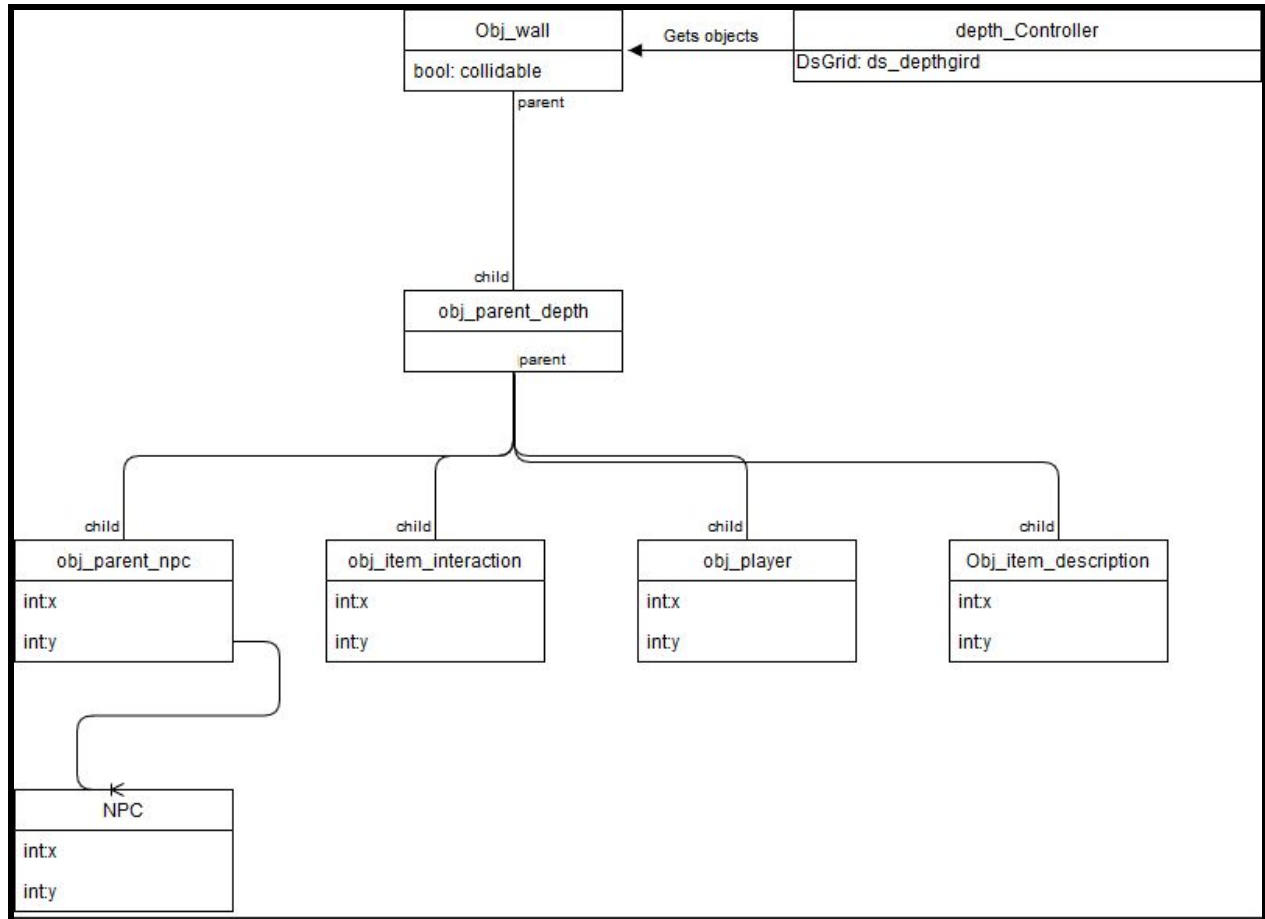
Figure 23
Depth controller relationship diagram

Menu object

The menu object is responsible for
1) Saving and loading while in game
2) Changing settings
   a) Audio
   b) Graphics
   c) Difficulty
3) Quitting the game

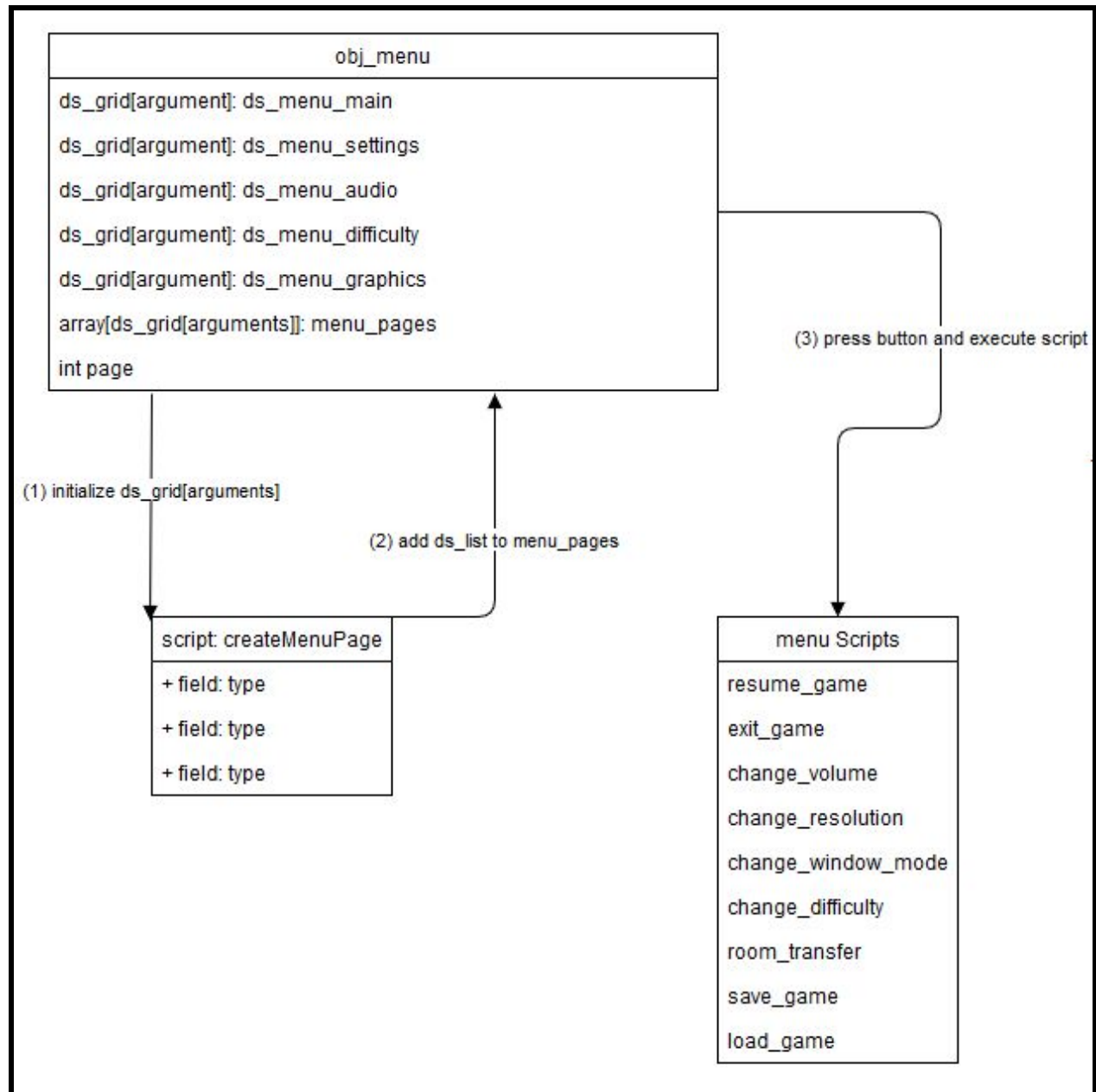It does this by having a ds list of functions and arguments that get called when the item is selected.

Figure 24
Menu object diagram

Player object

The player object contains the information relevant to displaying the player in the game. As well this object is the object that checks for user input and interaction with other items in the world
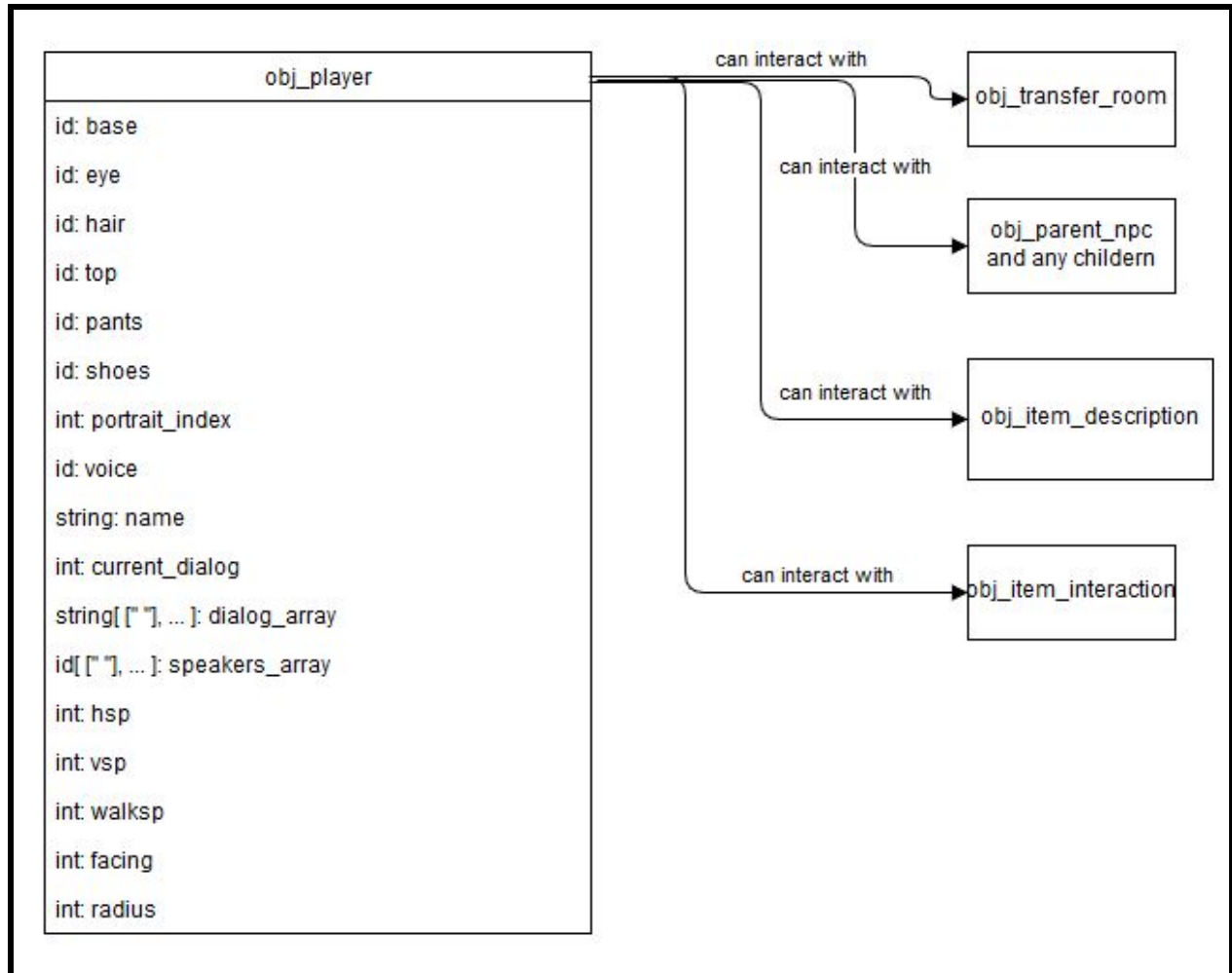
Figure 25
Player object and interactions

Game object

The game object is responsible for keeping track of user input and processing if the player presses the escape button or the inventory button. It is also responsible for acting as a buffer to hold information that will be loaded from file when the game is loaded from the title screen. The game object also has the added functionality of managing the fade in and out animation and keeping track of player x, y, and dir information when transitions between rooms. The reason for this was because it is a persistent object that can act as an in between for transferring information between objects in different rooms.

Figure 26
Game object

NPC parent object

This object is responsible for displaying npc characters, each npc character that is created can be loaded with a dialog array and speakers array. This is used for talking to NPCs
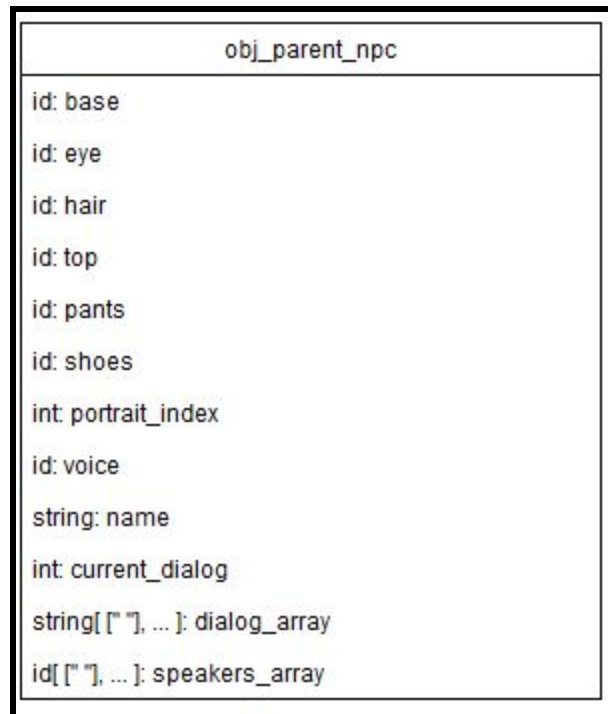
Figure 27
The parent object for the npc object

Room traversal diagram

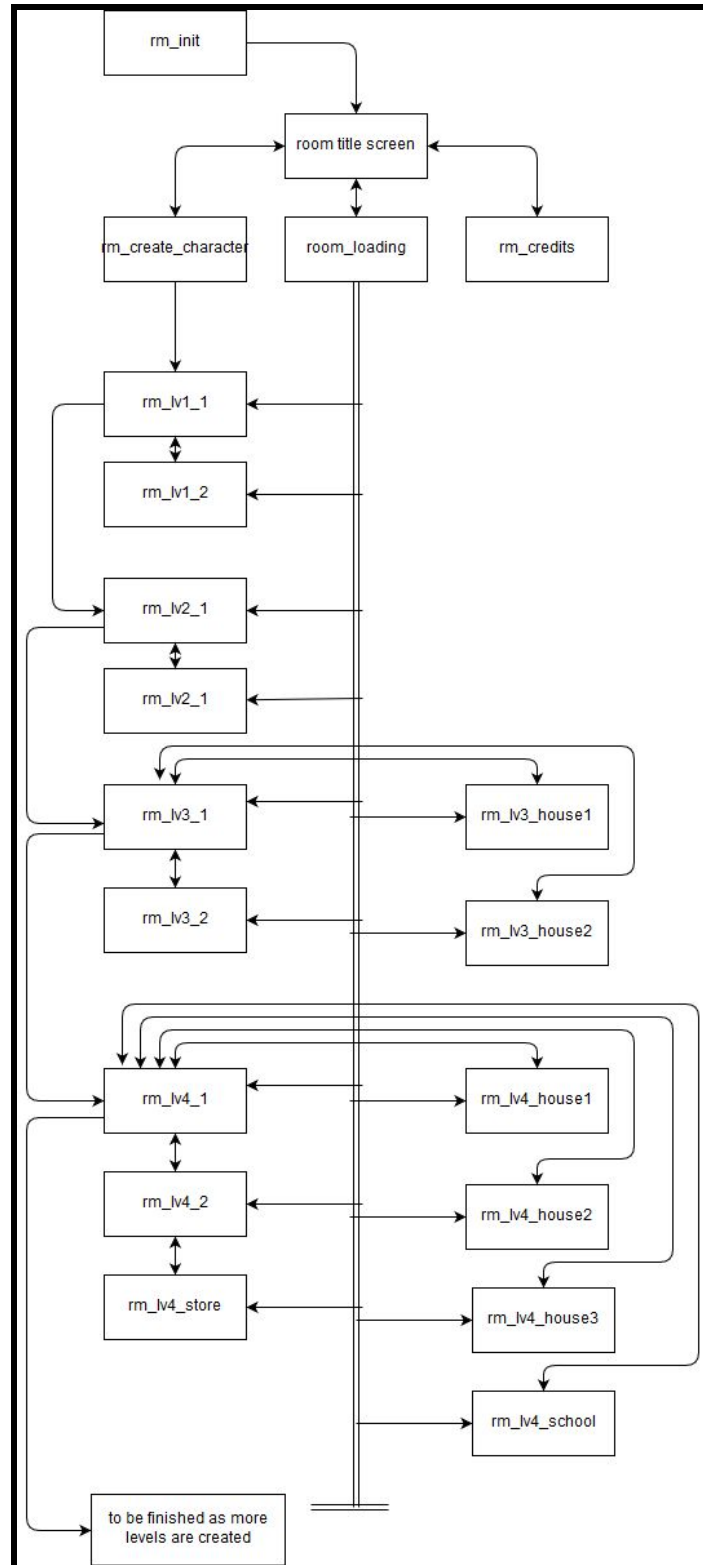This diagram shows how the player can navigate from the tile screen up to level 4 in the game. The diagram does not show the necessary conditions needed for room traversal

Figure 28
Room traversal diagram.