

Kally Tang Project Plan  
Project Plan Final Project:

I wasn't sure what kind of story line at first but I looked at the birthday card my friend sent me a last month which has a cat saying "Cake farts are like little puffs of marshmallow dreams". Made a story inspired by the card. The user can pick the kind of cat the user would like to play for the game.

For the space class, I will create a print function so that it can print what it represents with either words or as an image inside a box, to represent the square.

The character class will be an abstract class, and will have 3 child classes.

- Includes an image to print out of itself,
- Include some catch phrase
- 
- 1. Black tuxedo cat
  - a. Can name cat
- 2. White Cat(maybe adding another character )
  - a. When user attempts to name it, the cat refuses and names itself, its name is fluffy the III
  - b. Can't name
  - c. cat makes a messy cake
- 3. Rival cat(cat will appear in living room of neighbor's house)

I wasn't sure at first what method to use to represent spaces, whether to create an array or make them linked structures. If I had used a 2d array, some spaces would be nullptrs, or null. But if I created a linked structure of my own, then there would be areas that I wouldn't have to manage. I ended up deciding to make them into linked structures so that it was easier to manage spaces. I could create a function that would print an x in the location of the character within the space class. That way it would allow me to keep track of the character. The space class, will allow user to interact with the space and the function will print out a message from the cat saying it found a coin or it found a vending machine. If the user takes all the coins or all the ingredients, then there will be a different speech bubble that says that there's "there's nothing here" .

Since some places shouldn't be connected, for example the bedroom to the back yard, it doesn't get printed. On the mini map I can create a mini map as well as a key, the key would show what each line meant. If it was dashed lines or underscore lines, it means that it's not passable("\_\_\_\_\_"), if it's something that is passable use periods ". . . . ." to indicate it's passable. There could be a function that prints out letters to each of the spots, on the mini map, and when the mini map is printed, it'll print the letters and the corresponding places to them on the map.

## Kally Tang Project Plan

Each space class will have a pointer variable with a character pointer, in the neighbor's living room there will be two, one is an object created within the space for the rival cat and the other will be pointer for the character if character is in that space. The game class will control the game, will move the cat and place it into new space.

The space class will have a class called action, so that when it is called it allows user to interact with the room or enter a different part of the room as well as interact with different parts of the rooms, like the couch.

The game class will have a function that allows user to print a mini map of the area. The game class will also print the map of the areas available north, south, east and west of it. If there are no more spaces to the left, right, up or down, the area that doesn't exist will not print.

Since there's a limit to number of items that can be obtained in container, in this case it'll be a backpack. There will be more coins found than the backpack can carry and user must drop item in space, the space class will have a vector of items dropped in the space, space class will have a function called check dropped items, if there's dropped items user will be able to pick up items or leave them.

An object class will also be created so that I am able to create coins, ingredients for cake and allows user to

The game will end once the cat has gotten all ingredients that it needs.

If user chooses to quit or game wasn't completed in time, the game will display a sad cat

Once character has retrieved all ingredients game will complete and show the ending, if user doesn't make it in time, then it shows a sad cat with no cake.

## Sources of energy:

Vending machine can be used to get an energy drink that replenishes 5 health points, can be exchanged with one coin.

To differentiate the types of items, each inventory item will have a get type function.

	Item name	
1.	Key	v
2.	corn	v
3.	Flour(can't be removed from bag	v
4.	Eggs(can't be removed from bag)	v
5.	Baking soda(can't be removed from bag)	v
6.	coin	v

## Kally Tang Project Plan

<p>Space class:</p> <p>private:</p> <p>Character *currentSpace; std::vector&lt;InventoryItem*&gt; droppedItems; std::string name;</p> <p>public:</p> <p>//space pointers Space* top; Space* right; Space* left; Space* bottom;</p> <p>Space(); virtual void actionMenu()=0; virtual void printDescription() = 0; std::string getSpaceName(); void storeDroppedItems(InventoryItem*); void displayDroppedItems(); InventoryItem* pickUpDroppedItem(int); void deleteDroppedItem(int); virtual void printWholeRoom()=0; int validateInput(int, int); virtual ~Space();</p>	<p>Game class:</p> <p>private:</p> <p>Space* CurrentSpace; Space* kitchen1;</p> <p>public:</p> <p>Game(); void constructMap(); void runGame(); void ChooseCharacter(); void moveCharacter(); void menu(); void printMiniMap(); void printMidMap(); void BadEnd(); void GoodEnd(); int validateInput(int, int); ~Game();</p>	<p>Character class:</p> <p>protected:</p> <p>std::string name; int health; int attackC; int defenseC; const int MAX_SIZEBAG = 5; std::vector&lt;InventoryItem*&gt; bag;</p> <p>public:</p> <p>Character(); void setName(); std::string getName(); virtual void printCat()=0; void setHealth(); void decrementHealth(); void getHealthBar(); void printHealthBar(); void useLaser(); virtual void printFightSpeech()=0; virtual void printIntro()=0; void attack(); void defense(); void bagMenu(); void addToBag(InventoryItem*); InventoryItem* takeFromBag(int); void displayBag(); void printSadCatFace(); void useEnergyDrink(); bool bagFull(); virtual ~Character();</p>
--	---	--

Map w/ functions for child classes of space:

	<p>a. Kitchen  North: w/ fridge  South with cupboards  Check fridge and cupboard to see items need to be found</p> <p>Functions:  -check cupboard fx  -check fridge function</p>		
	<p>b. Livingroom  Look through couch for coins  Functions  -check couch function</p>		
	<p>Front porch  Random vending machine for food, can only eat at vending machine.  c.  -use vending machine function</p>	<p>d. Neighborhood</p> <p>Functions:  find metal detector and shovel  use metal detector to find a coin;</p>	<p>e. Rival/Neighbor's living room &amp; Neighbor's kitchen (rival's house)  Go through cabinets</p> <p>Functions:  Fight cat.  raidKitchen</p>
	<p>f. Farm/cornfield  Find key to open mill (in pile of hay)  Corn kernels  Farm house</p> <p>Functions:  Steal flour from flour mill  Farm animal's pens  Hen house to get egg</p>		

## Kally Tang Project Plan

Black cat	White cat	Rival cat
-----------	-----------	-----------

### Test plan:

Check if menus are working properly

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check the start menu works	User chooses 1 to start game	startGame()(Game class)	User is able to start game	User is able to start game
	User chooses 2 to end game	validateInput()(Game class)	User is able to quit game	User is able to quit game
User chooses a character	User chooses tuxedo cat	startGame()(Game class) validateInput()(Game class)	User is able to choose tuxedo cat and name cat	User is able to choose tuxedo cat and name cat
	User chooses the white cat	chooseCharacter()(Game class)	User is able to choose white fluffy cat but can't name	User is able to choose white fluffy cat but can't name
Check if user is able to name the black cat	User enters a name eg "FancyPants"	chooseCharacter() (Game class) setName()(Character class)	User is able to choose black cat and is able to name cat	User is able to choose black cat and is able to name cat
Check if user is able to pick white cat	User enters a name: eg:"Fluffy" Cat rejects name and tells user its name	chooseCharacter() (Game class)	Can't name Fluffy III	Can't name Fluffy III
Check end menu, if there's any memory leaks when user chooses to not play again and quits game	User chooses to quit game	startGame()	User should be able to quit game without memory leak	Memory leak Fixed: while loop in destructor of character child objects and in space class objects, when should have been using an if loop to check if the vector container empty User is able to quit
Check end menu, if there's any memory leaks when user chooses to not play again and quits game	User chooses to play again	startGame()	User is able to play again without any memory leaks	User is able to play again without any memory leaks

### Game class

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if game ends if user runs out of health	User's health is 0	checkHealthRunOut(Game class) getHealth(Character class)	Game ends and user is asked if user wants to play again Shows bad end.	
Check if user is able to end game from main menu	User chooses to quit in menu function	menu()(Game class)	Game ends and user is asked to play again	
Check if user obtains all items game ends	User obtains all items needed	checkItemsCollected	Game ends and user is displayed a cake.	

## Kally Tang Project Plan

### Check menu option(main) /menu in game class

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if user is able to quit game from Main Menu.	User chooses to quit game from main menu(enters 6)	menu()(Game class) validateInput()(Game class)	User is able to quit game	User is able to quit game
Check if the option to check our area works	User chooses 1	menu()(Game class) actionMenu()(Game class)	User is able to display menu to interact with kitchen	User is able to display menu to interact with kitchen
Check if printing mini map works	User chooses 3	printminiMap()(Game class) menu()(Game class)	Mini map is displayed to user	Mini map is displayed to user
Check if printing health bar works	User chooses 4	printHealthBar(Game class) printHealthBar(Character class)	Health bar is displayed to user	Health bar is displayed to user
Check if moving to another area works.	User chooses 2 and chooses to go to living room	menu()(Game class)	User is able to go to living room	User is able to enter the living room
Check if user is able to print description of room	User enters 5	printDescription(space class/child classes of space class)	User is able to print description of current room	

### Check functionality of kitchen:

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if kitchen menu works	If user chooses to interact with room, and chooses 1 to print room	printWholeRoom()(Kitchen class) actionMenu()(Kitchen class)	Map of room is displayed to user	Map of room is displayed to user
Check If check floors work	If user enters 2 and floor is not empty	actionMenu()(Kitchen class) displayDroppedItems()(space class)	The contents of the dropped items on floor is displayed to user.	The contents of the dropped items on floor is displayed to user.
	If user enters 2 and floor is not empty		User is displayed with the message that there is nothing on the floor	User is displayed with the message that there is nothing on the floor
Check if CheckFridge works	If there aren't any eggs in bag and user chooses to check fridge	findItem();(Character class) actionMenu()(Kitchen class) checkFridge()(kitchen class)	User is displayed a message that there's no eggs and user needs to get some from farm	User is displayed a message that there's no eggs and user needs to get some from farm
	If there is eggs in bag and user chooses to check fridge		User is displayed the message that you have eggs in bag	
Check if Check cupboards work;	If user has no flour and no baking soda, and user checks cupboards,	findItem();(Character class) actionMenu()(Kitchen class) checkCupboards()(Kitchen class)	user is displayed with a message saying that user needs to get flour and baking soda, and to find items in farm and rival's kitchen	user is displayed with a message saying that user needs to get flour and baking soda, and to find items in farm and rival's kitchen
	If user has no flour but has baking soda, and user checks cupboards,		User is displayed with message user has no flour but has baking soda, and to get flour at the farm.	

## Kally Tang Project Plan

	If user has flour but no baking soda, and user checks cupboards,		User is displayed with message user has flour but no baking soda and to get baking soda from rival's house	
Check if user is able to pick up dropped items	User chooses to check floor and pick up item	pickUpDroppedItem()(space class) displayDroppedItems()(space class) actionMenu()(kitchen class)	User is able to pick up dropped item, item can no longer be found on the floor.	Memory leak Edit fixed: User is able to pick up dropped item, item can no longer be found on the floor
Check if user is able to drop item	User chooses to check bag in actionMenu, then chooses to drop an item that's droppable(eg coin)	storeDroppedItem(space class) takeFromBag(character class)	User is able to drop item into room, item is now on the floor.	Memory leak Edit fixed: User is able to drop item into room, item now can be found on the floor
check if character is able to eat food	User chooses to eat food	eatFood()(Kitchen class) addHealth(Character class)	User is displayed with message that user already ate the food	User is displayed with message that user already ate the food
check if character is able to eat food only once	User choses to eat food a second time		User is displayed with message that user already ate the food	User is displayed with message that user already ate the food
Check if user is able to move to the Living room	User enters 2 in main menu to go to a new area and user has checked the cupboards and checked the fridge	moveMenu()(Kitchen class) menu()(game class) validateInput()(space class)	User is able to go to living room	User is able to go to living room
If user doesn't check both cupboard and fridge	User enters 2 in main menu to go to a new area and user has NOT checked the cupboards and checked the fridge	canLeave() (space class) menu()(game class)	User is unable to leave Is prompted to check cupboard	User is unable to leave Is prompted to check cupboard
	User enters 2 in main menu to go to a new area and user has checked the cupboards and has NOT checked the fridge		User is unable to leave and prompted to check fridge	User is unable to leave and prompted to check fridge

Character class options(inherited by BlackCat, WhiteClat, and RivalCat class.

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if findItem function works	An integer value is passed into function;	findItem();(Character class)	Function returns true or false depending if item specified is in bag	Function returns true or false depending if item specified is in bag
Check if bag limit works				
Check if user is able to drop items when function is called	User chooses to drop items in any room	storeDroppedItems()(character class)	User is able to drop item	User is able to drop item

## Kally Tang Project Plan

Check if user chooses to drop flour, baking soda or eggs, user can't drop items	user chooses to drop flour, baking soda or eggs	storeDroppedItems takefromBag	User is unable to drop items	User is unable to drop items
---	---	----------------------------------	------------------------------	------------------------------

## Character class's Status bar:

Test Case	Input Values /condition	Driver Functions	Expected Outcomes	Observed Outcomes
Check if status bars are displayed properly	If Health is between 25 and 20	printHealthBar	I feel great message is displayed	I feel great message is displayed
	If health is between 20 and 7		I feel okay message is displayed	I feel okay message is displayed
	If health is between 7 and 4		Need food message is displayed	Need food message is displayed
	If health is less than 4		Feed me message is displayed	Feed me message is displayed

## Check Living room functionality:

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check menu option works for checking couch	If user enters 3 to check couch cushions for the first time	checkCouch()(LivingRoom class) actionMenu()(LivingRoom class)	User is displayed a menu to take coin	User is displayed a menu to take coin
	User enters 3, and chooses to take coin(enters 1) and bag isn't full	checkCouch() addToBag(); actionMenu()(LivingRoom class)	User is able to see menu and add coin to bag	User is able to see menu and add coin to bag
	User takes the coin from the couch and enters 3 to check couch again	checkCouch() addToBag(); actionMenu()(LivingRoom class)	User does not get option to add coin and user is displayed a message "There's nothing in the couch but some dust bunnies"	User does not get option to add coin and user is displayed a message "There's nothing in the couch but some dust bunnies"
Check if user is able to drop coin/item and find coin on floor and take coin	User chooses 3 and enters 1 to take coin, user then drops coin on floor, and checks floor, and takes coin from floor	pickUpDroppedItem()(space class) displayDroppedItems()(space class) actionMenu()(Living Room class) takeFromBag(character class)	User is able to take coin from couch, drop coin and find coin on floor, and pick up coin from floor.	User is able to take coin from couch, drop coin and find coin on floor, and pick up coin from floor.
Check if user is able to go back up to the kitchen	User chooses 2 in main menu to go to another area, user chooses 1 to go back to the kitchen	moveMenu()(Living room class) menu()(game class) validateInput()(space class) canLeave()	User is able to go back to kitchen	User is able to go back to kitchen



## Kally Tang Project Plan

Check if user is able to go down to the porch	User chooses 2 in main menu to go to another area, user chooses 1 to the porch		User is able to go to porch	User is able to go to porch
Check if user is able to print porch map	User chooses 1 to print area	actionMenu()(Living room class) printWholeRoom()(Living room class)	User is able to print map	User is able to print map
User is able to drop items on floor and floor will have item	User chooses to check bag and drop item on floor	storeDroppedItems() takeFromBag	User is able to drop items and room's floor(droppedItems vector) will store it	User is able to drop items and room's floor(droppedItems vector) will store it

## Porch functionality:

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if user is able to use the vending machine	If user doesn't have coins and tries using vending machine	useVendingMachine()(porch class)	User is given message that user doesn't have any coins.	User is given message that user doesn't have any coins.
	If user does have a coin, and user chooses to use coin		User is able to replenish health by 5 pts	Loop in function doesn't end Fixed:
Check if user is able to go right to go to neighborhood	User chooses option to go to neighborhood(user enters)	actionMenu()(porch class) menu()(game class)	User is able to go to neighborhood	Memory leak Fixed: User is able to go to neighborhood
Check if user is able to go up to living room	User choses to go to living room		User is able to move to living room	User is able to move to living room
Check if user is able to go to the farm	User choses to go farm		User is able to move to farm	User is able to move to farm
Check porch is printed properly	User chooses 1 to print map of room/area	actionMenu()(porch class) printWholeRoom(Porch class)	Image of porch area is displayed	Image of porch area is displayed

## Testing neighborhood functionality

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if user is able to go left to porch	User doesn't use shovel or interact with any objects in room	menu*()	User is able to go to porch	User is able to go to porch
	User picks up shovel and doesn't put it back	actionMenu()		
Check if user is able to go right to neighbor's house	User doesn't pick up shovel or metal detector or interact with any objects in room	canLeave() menu()	User is able to go to rival's house	User is able to go to rival's house
	If user put back shovel and metal detector and chooses to go to rival's house		User is able to go to rival/neighbor's house	User is able to go to rival's house

## Kally Tang Project Plan

Check If user can't leave area if has shovel and metal detector	User picks up metal detector and didn't return it and or user picked up shovel and put back shovel		User isn't able to leave area, and given the message to put away metal detector	User isn't able to leave area, and given the message to put away metal detector
	User picks up shovel and didn't return it and or user picked up metal detector and put back metal detector		User isn't able to leave area, and given the message to put away metal detector	User isn't able to leave area, and given the message to put away metal detector
Check if user is able to take and return metal detector and shovel	If user put back shovel and metal detector and chooses to go to another room	checkLawn2()	User is able to get metal detector	User is able to get metal detector
	User chooses to go to lawn and chooses not to get metal detector		User doesn't get metal detector	User doesn't get metal detector
	User took metal detector and user chooses to go to lawn 2 and chooses to return metal detector		User is able to return metal detector	User is able to return metal detector
	User took metal detector and user chooses to go to lawn and chooses NOT to return metal detector		User is able to keep metal detector	User is able to keep metal detector
	User choose to go to lawn 1 for shovel and chooses to take shovel	checkLawn1()	User is able to get shovel	User is able to get shovel
	User chooses to go to lawn one and not get metal detector		User doesn't get shovel	User doesn't get shovel
	User took shovel and chooses to go to Lawn 1 and return metal detector		User is able to return shovel	User is able to return shovel
	User took shovel and chooses to go to Lawn 1 and does NOT return metal detector		User is able to keep shovel	User is able to keep shovel
	User chooses to find coin in lawn1.	CheckCorner1()	User finds coin and loses two health points	User finds coin and loses two health points
	User chooses to not find coin in lawn1		User finds coin and doesn't lose health points	User finds coin and doesn't lose health points
Check if user is able to take coins from corner1 and corner 2 of lawn1 and lawn 2	User chooses to find coin in lawn.	CheckCorner2()	User finds coin and loses two health points	User finds coin and loses two health points
	User chooses to not find coin		User finds coin and doesn't lose health points	User finds coin and doesn't lose health points
Check when user takes coins, user cannot find anymore coins	User tries finding coin in corner 1 of lawn but coin is already taken	CheckCorner 1()	User can't find any coins	User can't find any coins
	User tries to find coin in corner 2 of lawn		User can't find any coins	User can't find any coins

RivalHouse functionality

## Kally Tang Project Plan

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if cat fight works	User chooses to fight rival	fightRival() battle()	User is able to battle	User is able to battle
Check if user loses, message displayed is that rival won	User loses battle	fightRival() battle()	User is displayed that user's cat' lost and game ends	User is displayed that user's cat' lost and game ends
Check if user wins, character is displayed as winner	User wins battle	printLoserFace	User is displayed that rival lost	User is displayed that rival lost
Check if user is able to leave	User chooses to leave room	actionMenu()	User is able to leave room	User is able to leave room
Check if user is unable to go to kitchen unless user fought rival	User didn't fight rival	raidKitchen()	User can't raid kitchen	
	User fights rival	raidKitchen()	User is able to raid kitchen	

## Farm functionality

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if user is able to get key	User chooses to add key to bag and bag has room	checkHayStack()	User adds key to inventory,	User added flour Fixed: User adds key to inventory,
	User chooses to add key to bag and bag is full		User isn't able to add to bag,	User isn't able to add to bag,
Check if user is able to find key	User chooses to check hay stack		User is able to find key and get key	User is able to find key and get key
Check if user is able to open flour mill	User attempts to open flour mill With key in inventory	openMill();	User is able to get flour	User unable to get flour Edit fixed: User is able to get flour
	User attempts to open flour mill With no key in inventory		User is unable to get key and is told to find key in haystack	User is unable to get key and is told to find key in haystack
Check if user is able to feed chickens and get eggs	User obtains corn and attempts to feed chicken	feedChickens() visitChickens	User is able to feed chickens and get eggs	User is able to feed chickens and get eggs
	User doesn't have corn and didn't feed chicken		User is unable to feed chickens	User is unable to feed chickens
	If user already fed chickens and user tries to visit chickens		User is given a message that they were already fed	User is given a message that they were already fed

## Run game with black tuxedo cat

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if user is able to quit midgame without errors	User chooses to quit mid game	startGame()	User is able to quit without errors	User is able to quit without errors

## Kally Tang Project Plan

Check if user is able to collect all items and win	User collects all items	checkItemsCollected	Game ends user is shown a cake	User was shown cake and end message twice Fixed: Game ends user is shown a cake
--	-------------------------	---------------------	--------------------------------	---

## Run game with white fluffy cat

Test Case	Input Values	Driver Functions	Expected Outcomes	Observed Outcomes
Check if user is able to quit midgame without errors	User chooses to quit mid game	startGame()	User is able to quit without errors	User is able to quit without errors
Check if user is able to collect all items and win	User collects all items	checkItemsCollected	Game ends user is shown a cake	Game ends user is shown a cake made by fluffy cat

## Reflection:

One thing I changed was that I deleted the function to delete dropped item and deleted the dropped item in `pickUpDroppedItemFunction()`. I also decided to condense the game a little more and make the farm just one place. I condensed the game to 6 spaces to make debugging easier. I also reduce the number of inventory items.

I also removed the `useEnergyDrink` function in character class, decided that user could only refill energy while at a vending machine or if they go back home and eat from their cat food.

I had to figure out a way to drop items into rooms and while creating functions, I encountered a problem that it was the object was being deleted instead of added to the floor of the room. I realized I accidentally deleted the item to be dropped and when calling a function to get type of item it would cause an error.

Working on rooms one by one made it easier for me to debug the program.

Found that I whenever I tried deleting the kitchen object it would cause an error, still trying to figure out what is wrong. Whenever I successfully exited the game it had an error message. I figured out what was the cause of the memory leak, my destructor had a while loop that never ends, replaced while with an if statement.

Found a memory leak when user loses and user chooses to play again, found that the loop to delete items doesn't end. Fixed the same way as the other memory leak, using an if statement instead of while statement to delete all objects in bag vector container.