QUEEN'S UNIVERSITY BELFAST

Semester 2 Report

Group 59

Table of Contents

Peer Assessment

Evaluation

Evaluation	Group Number: 59			
Name	Contribution to team-working and motivation[1]	Contribution to PDF Report 2, and the Video Demo [1,2]	Contribution to methodically developed and functioning system code[1,2]	**Peer Score** (Range 85 – 115)
Dean Logan	4	5	5	113
Conor Nugent	4	3	4	104
John Higgins	4	4	4	109
Scott McDonald	4	3	4	107

[1]Values for contribution: 1 = Minimal Contribution; 2 = Reasonable Contribution; 3 = Good Contribution; 4 = Very Good Contribution;
5 = Excellent Contribution

Declaration

Declaration

"I declare that I have read the Queen's University regulations on plagiarism, and that any contribution I have made to the attached submission is my own original work, except for any elements that I have clearly attributed to third parties. I understand that this submission will be subject to an electronic test for plagiarism and will also be subject to the University's regulations concerning late submission if it is received after the deadline."

Name	Date	Confirmation *(use the words shown in the example below!)*
Dean Logan	24/03/2022	I agree to the terms of the declaration
Conor Nugent	24/03/2022	I agree to the terms of the declaration
John Higgins	24/03/2022	I agree to the terms of the declaration
Scott McDonald	24/03/2022	I agree to the terms of the declaration

Personal Statements

Personal statement of	*Dean Logan*
The following were my most significant contributions to the Semester 2 Deliverable (100 words or less):	
I was responsible for coding everything within the Game, File, Location and SafeMode classes. I also participated in some paired programming with Scott and Conor where I was the driver and instructor respectively, while doing this we worked on the CorruptFile class and the Captcha class. also created the interface design along with completing the revisions to the class diagram, sequence diagrams, use case descriptions and diagram, and the board layout. Once the system was finished, I completed some tests. I also completed the write up for Secure System Features, testing and development methodologies, and completed the GitLab Activity screenshots along with compiling everything together to make this report.	

Personal statement	*Conor Nugent*
The following were my most significant contributions to the Semester 2 Deliverable (100 words or less):	
Coded Utility class, coded dice class, coded Captcha class excluding TicTacToe and Wordle Hangman Game within Captcha. Helped with the video demo. I did pair programming with John on fixing some errors within the Captcha mini games where I was the driver. I did pair programming with Dean on fixing some errors to get the text files working for the Captcha class. Test cases for finish game, display title screen, setup game and some land on File Explorer.	

Personal statement of	*John Higgins*
The following were my most significant contributions to the Semester 2 Deliverable (100 words or less):	
I programmed the classes for 'Go', 'DownloadVirus' and 'Player'. I also programmed the 'TicTacToe' captcha game and helped with validation and bug fixing on other classes. I did paired programming with Conor on Captcha (Conor was the driver and I was the instructor) and with Scott on 'RecyclingBin' (I was the driver and Scott was the instructor). I also created templates for testing and interface design for the report that could be utilized by the team. I carried out black-box, white-box and JUnit Testing. I planned and recorded the bulk of the demo video and commentated on it.	

Personal statement of	*Scott McDonald*
The following were my most significant contributions to the Semester 2 Deliverable (100 words or less):	
I wrote the weekly minutes. I paired programming with Dean as the instructor to bug fix corrupt file, paired programming with JB as the driver to bug fix the recycling bin. I wrote the code and JUnits for the recycling bin, file explorer and corrupt file. I contributed to the testing of main game features and invalid tests for the Captcha Hangman game, the display option list, corrupt file, the file explorer and brute force questions. I did the main testing for playing the brute force attack mini game, land on GO, roll dice and land on utility.	

Design Documentation

Text User Interface Design – D.L

ID / Classname	CLI Sample	Purpose of Message	Reason for inclusion in report
Game - Title Screen Menu	Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice:	This is displaying the title screen menu to the user. This will allow the user to pick one of the 2 options displayed.	Included in the report as this is the first impression the user gets of the system.
Game - displaying player menu	<player name> you currently have <number of bits> bits <player name> please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice:	Displays a menu to the user allowing them to select one of the options within the game. It will also display the players current bits.	This is the most important menu within the game as it appears every time it is a users turn. The menu is ordered by the most likely options the user will be selecting.
Game - When player enters an invalid option	Please select a valid option from the list by either entering the number or the name of the option you wish to select	Tells the user that they have entered invalid data and gives a message indicating why it was invalid.	This is an example of an error message given to the user if they enter invalid data. Other designs of error messages can be seen in the appendix
Game - Roll Dice	<player name> you have rolled a <first dice rolled> and a <second dice rolled> meaning you will move <total of dice rolled> places You have moved from position <old position> to <new position> You have landed on <location name>	Informs the current player what the numbers their dice rolled, the total of those numbers and what position they were on to their new position, then the name of their new position. This allows the user to see how many places they have moved and where their current position is.	This is the output of the most common option selected from the player menu and arguably one of the most important aspects of the game. The rolling of the dice and moving the player.

Game - Land On File (player does not own the file)	`You have landed on <file name>. Do you wish to download this file for <bits required for download> (y/n):`	Used to ask the user if they wish to download this file.	This is included here as this is one of the key aspects of the game therefore it is important to have a clear interface.

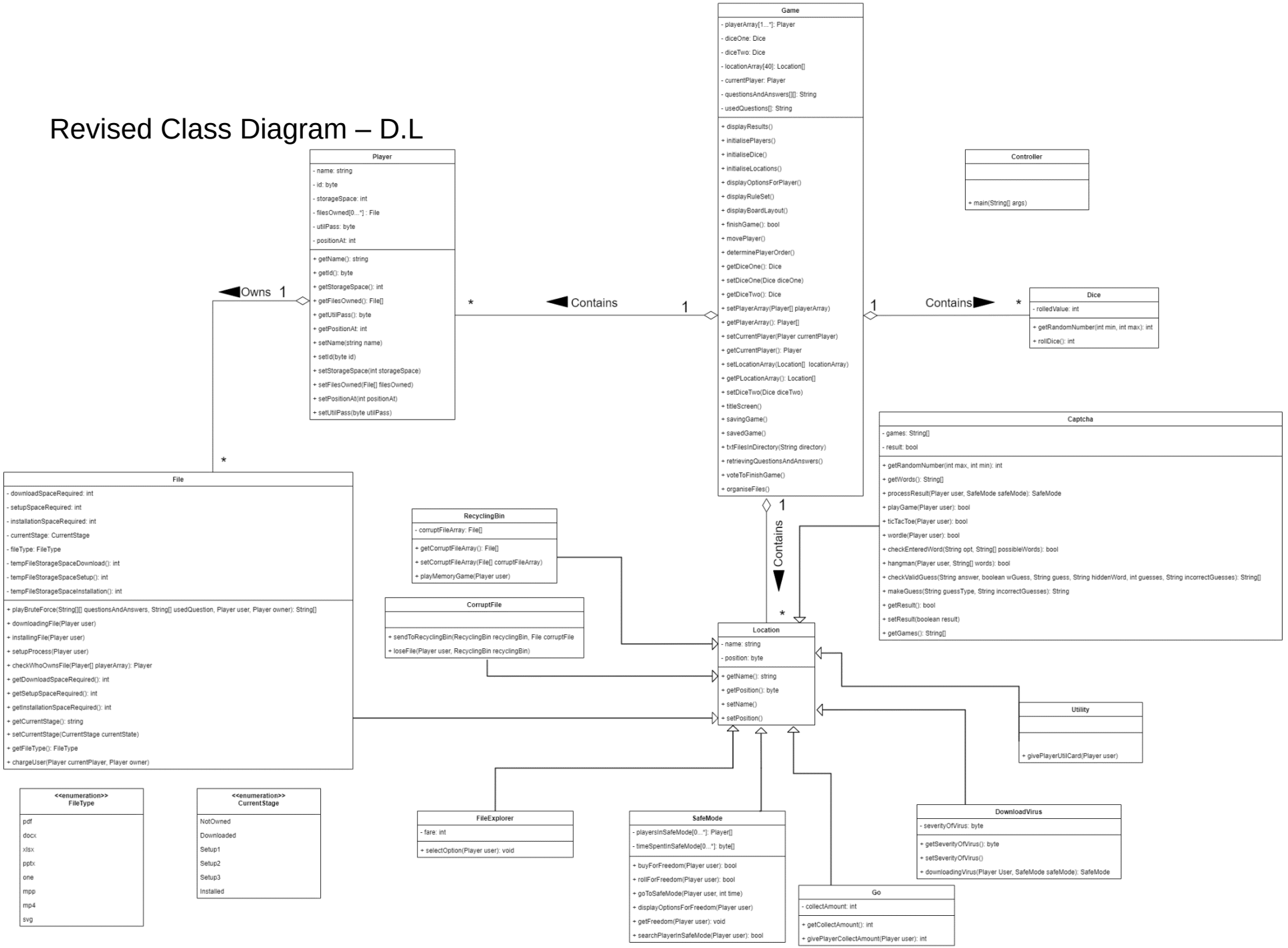
File - Brute Force Game	`The password hint is: <question selected>`
	
	`Please enter password:`
		Gives the player the question which they have to answer and prompts them to enter a password. The player will have 5 attempts at answering the question.	Shown here as this is part of landing on the file (a common occurrence within the game) and one of the value added features within the game.

File - Brute Force Game (answer is correct)	`Well done the password is correct you no longer have to pay the owner of this file`	Tells the player they have answered the question correctly and informs the player that they no longer owe any bits.	Shows that a user doesn't always have to use resources whenever they land on another users file.

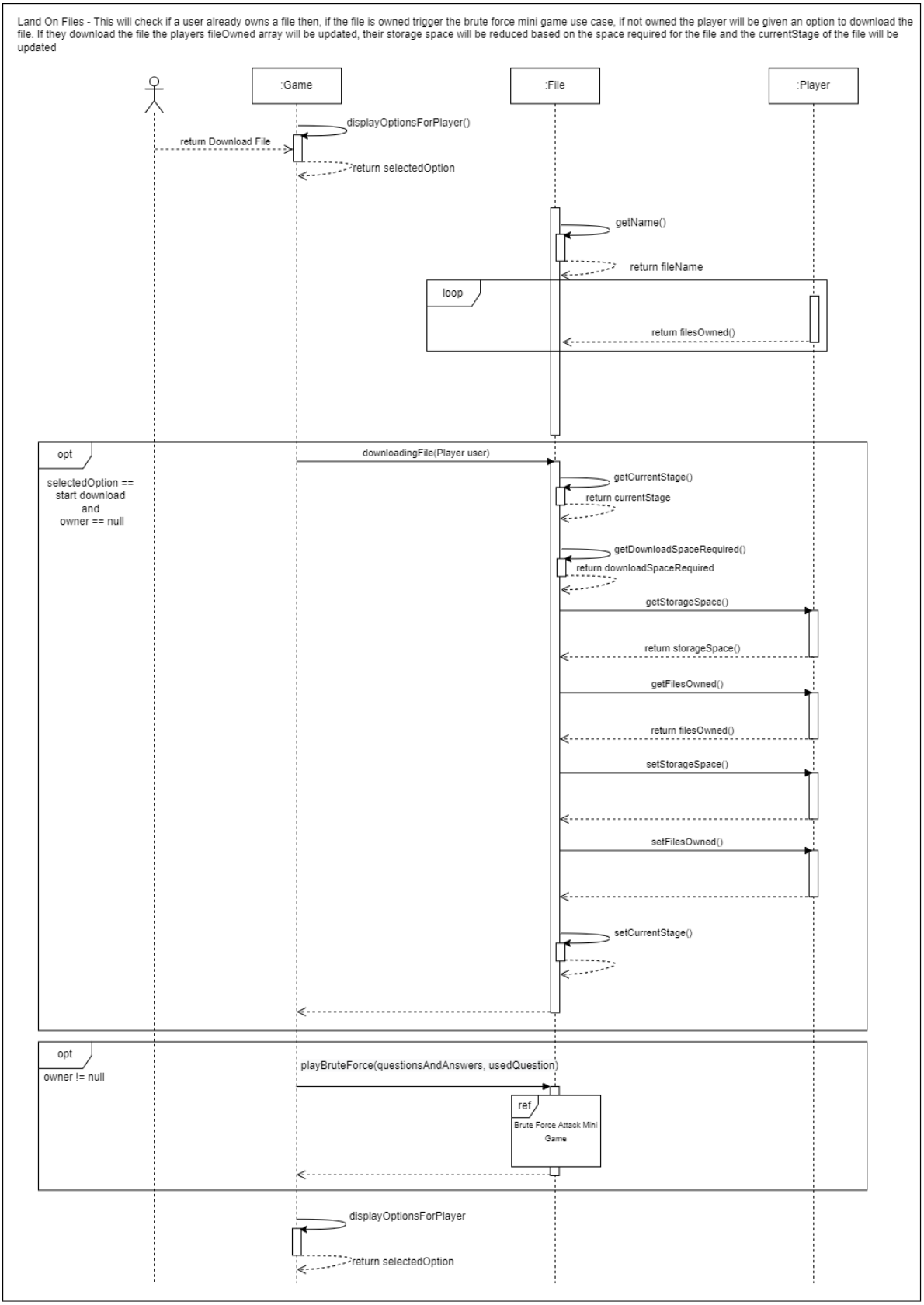
File - Brute Force Game (answer is incorrect)	`You have ran out of tries`
	
	`<current player name>, you have given <owner of file> <price> bits`
		Informs the user that they have failed to guess the password and therefore will be giving the owner of the file bits.	Shows the case whenever a user does use up resources whenever they land on another users property

Note: Additional Text Designs can be seen within the Appendix under the heading "Text User Interface Design -Continued" or if you prefer seeing these two tables combined into one you can view this google document here:
https://docs.google.com/document/d/1UEkNqbWu0a7stCB08oLXAJDmQTGqYvXUhhPUR0BDUKs/edit?usp=sharing
All content within this google document can be found in the report. It is only included if you prefer to see all the interface design within the one table instead of it being separated.

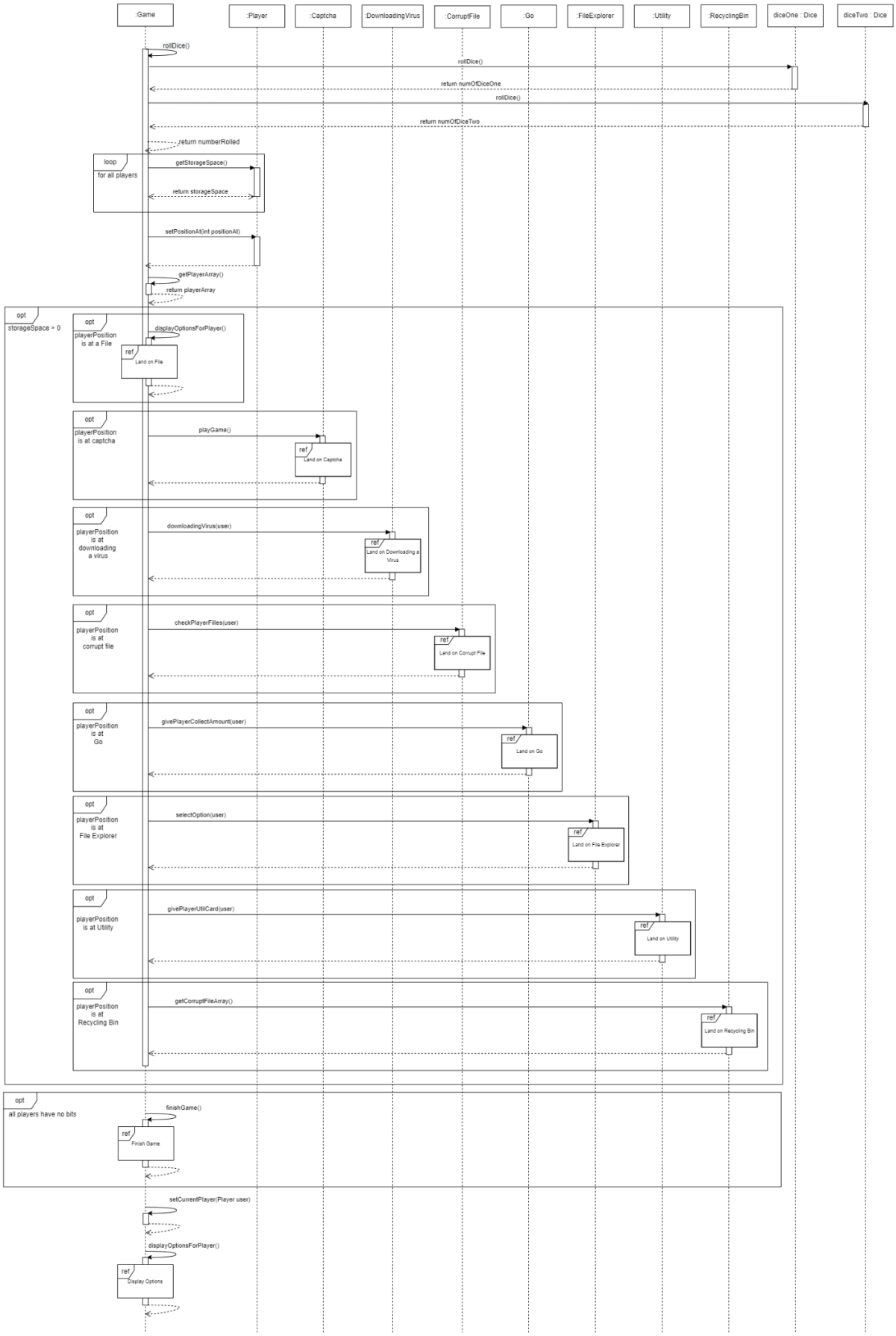
Revised Class Diagram – D.L



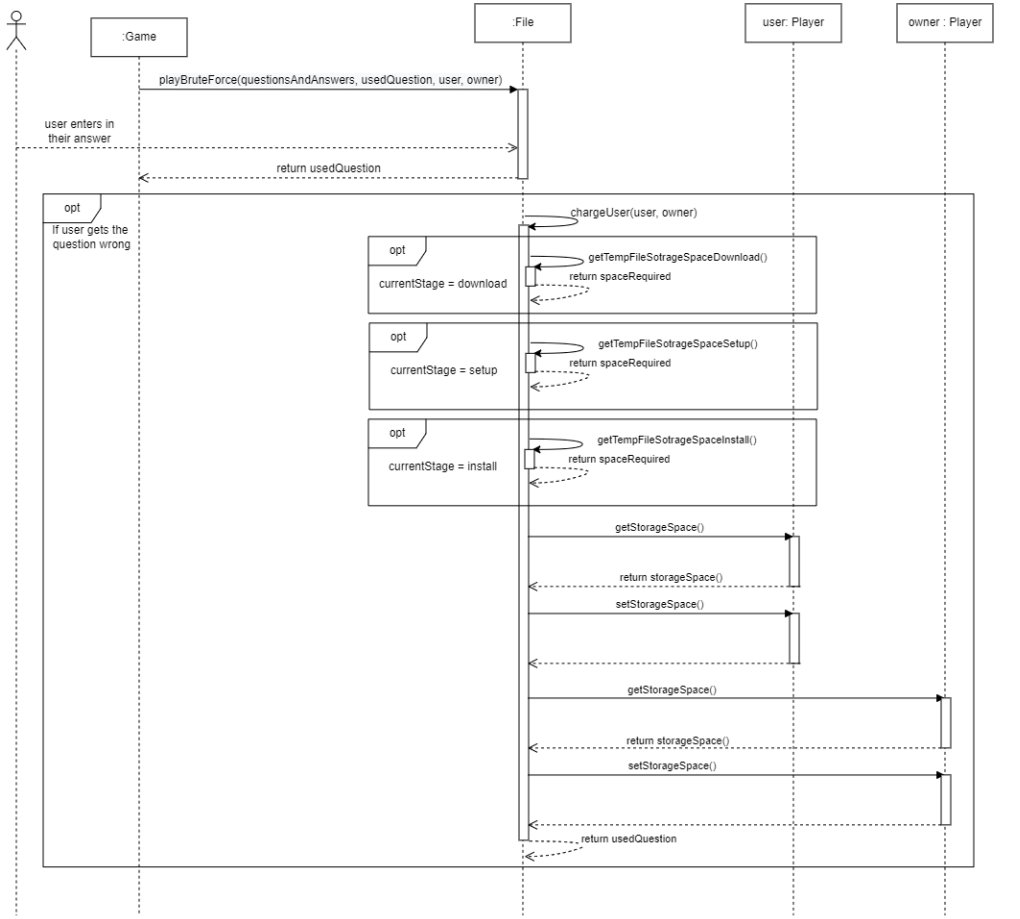
Revised Use Case Realisations (Sequence Diagrams) – D.L



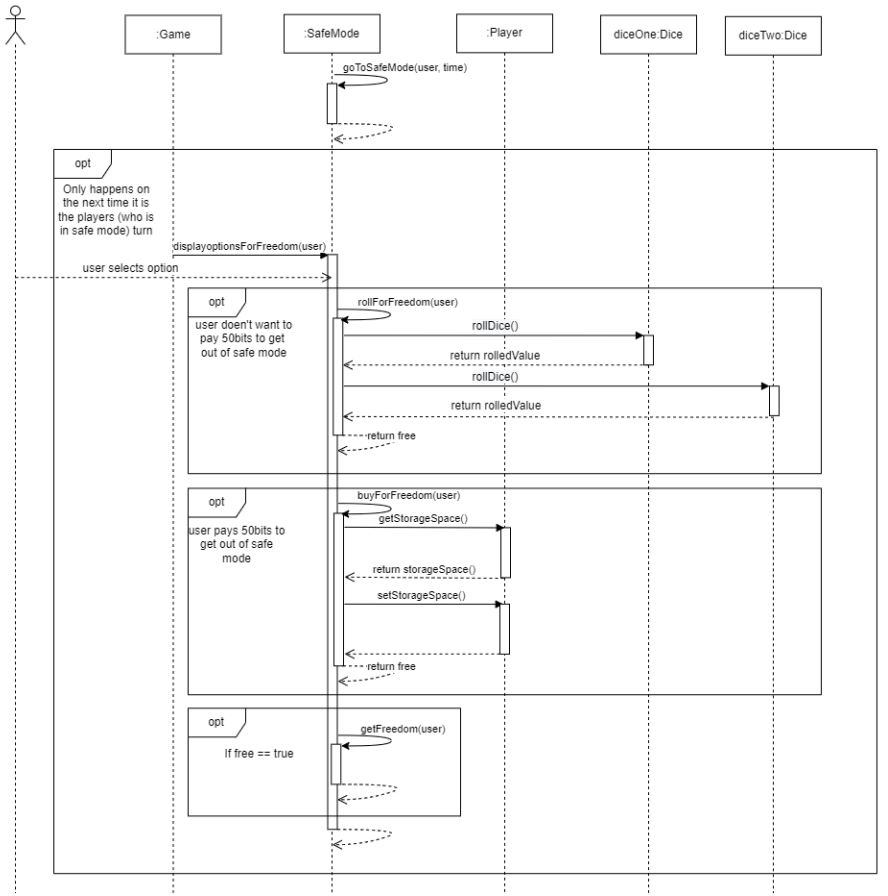
Roll Dice - This gets all of the players storage space checks if they are 0, if a players storage space is 0 then that players position will become -1, if only 1 player has a position which is not -1 then the finish game use case is triggered. Assuming the current players storage space is not 0 then there position will be updated and depending on the location they land on the corresponding use case will be triggered, the currentPlayer will be updated to the next element in the playerArray

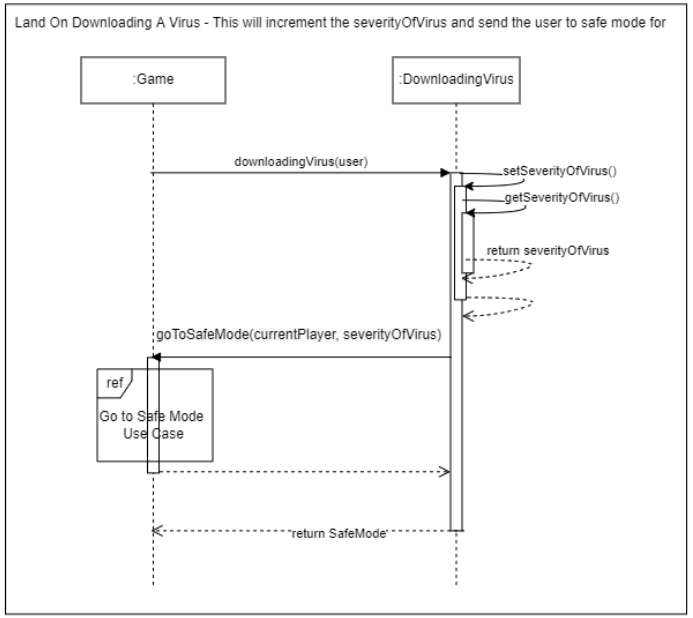


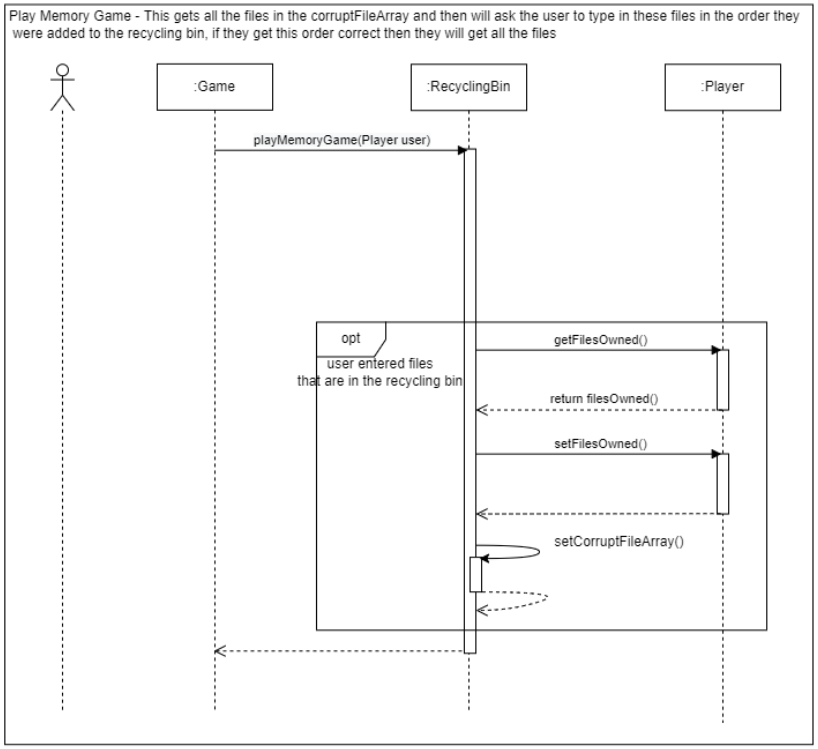
Brute Force Mini Game - This will make the current player answer a quiz question, if the player gets the question wrong then their result will be false. This means that the system will check who owns this file and what stage the file is in, which then allows the system to take away the required amount of bits from the current player to give to the owner of the file. If the result of the game is true then nothing happens

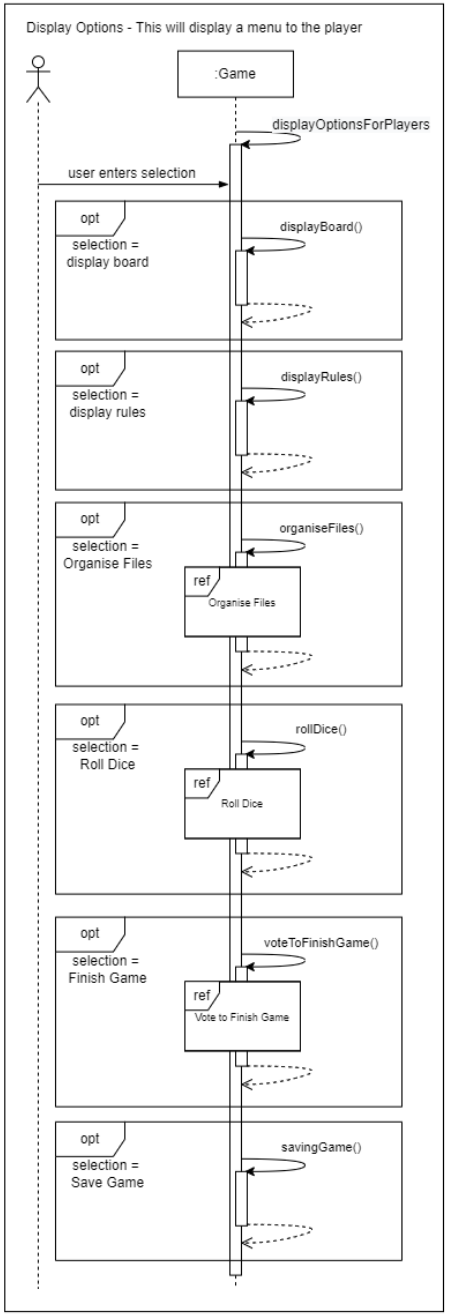


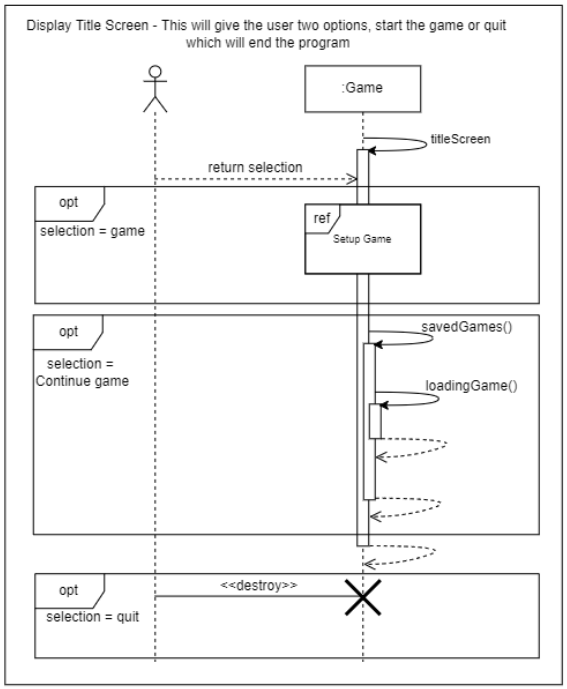
Go To Safe Mode - This will send a player to safe mode for a certain length of time by updating the corresponding array. Then whenever it is a players turn they will roll the dice to see if it is a double to get out of safe mode, if they do not roll a double within their designated time in safe mode then they 50 bits to get out

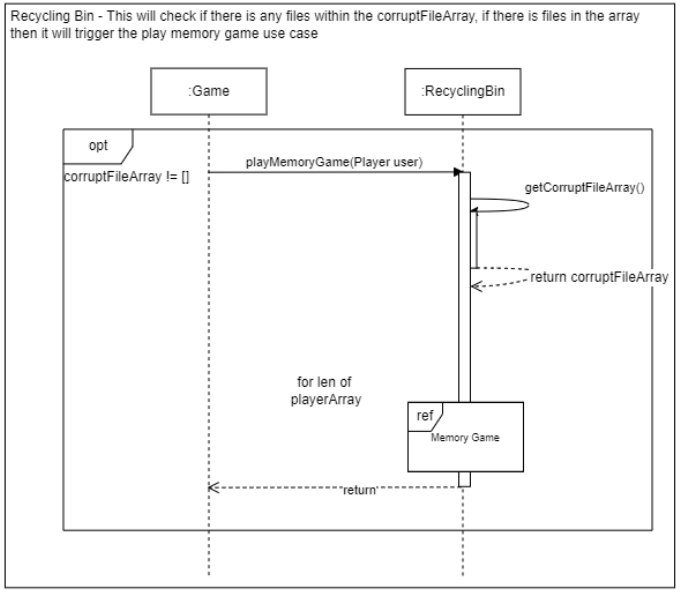



Land On Downloading A Virus - This will increment the severityOfVirus and send the user to safe mode for
:Game
:DownloadingVirus
downloadingVirus(user)
setSeverityOfVirus()
getSeverityOfVirus()
return severityOfVirus
goToSafeMode(currentPlayer, severityOfVirus)
ref
Go to Safe Mode Use Case
return SafeMode


Play Memory Game - This gets all the files in the corruptFileArray and then will ask the user to type in these files in the order they were added to the recycling bin, if they get this order correct then they will get all the files
:Game
:RecyclingBin
:Player
playMemoryGame(Player user)
opt
user entered files that are in the recycling bin
getFilesOwned()
return filesOwned()
setFilesOwned()
setCorruptFileArray()


Display Options - This will display a menu to the player
:Game
displayOptionsForPlayers
user enters selection
opt
selection = display board
displayBoard()
opt
selection = display rules
displayRules()
opt
selection = Organise Files
organiseFiles()
ref
Organise Files
opt
selection = Roll Dice
rollDice()
ref
Roll Dice
opt
selection = Finish Game
voteToFinishGame()
ref
Vote to Finish Game
opt
selection = Save Game
savingGame()


Display Title Screen - This will give the user two options, start the game or quit which will end the program
:Game
titleScreen
return selection
opt
selection = game
ref
Setup Game
opt
selection = Continue game
savedGames()
loadingGame()
opt
selection = quit
<<destroy>>


Recycling Bin - This will check if there is any files within the corruptFileArray, if there is files in the array then it will trigger the play memory game use case
:Game
:RecyclingBin
opt
corruptFileArray != []
playMemoryGame(Player user)
getCorruptFileArray()
return corruptFileArray
for len of playerArray
ref
Memory Game
return

Implementation-Related Documentation

Test Plan

Note: The continuation of the test plan can be found in the appendix under “Test Plan – Continued”, additionally screenshots for each test can be found in the appendix under “Testing Screenshots” (The screenshot for a test and the test case are paired by their ID).
You can also find a google document with all the test cases within a single table here:
https://docs.google.com/document/d/1_R74ik7sKn6RV6LGSIsO3QXTRl7Dhn2Y8DxBifYurmM/edit?usp=sharing
You can also find a google document with all the test case screenshots here:
https://docs.google.com/document/d/1B1451Ia_nxdAjDdZ-WZu1UlPhYdUZIh20qKKts0EUkQ/edit?usp=sharing
All content within these google documents can be found within this report they are only included here if you find it easier to read (e.g have testing screenshots open as you view the test plan).

D.L: 1-9, 13-15, 17-27, 61-66
J.B: 10-12, 28-33, 35-48
C.N: 49-50, 56-60
S.Mc: 16, 34, 51-55, 73-84

ID	Use Case Ref	Description of Test	Test Initialisation	Test Inputs	Test procedures	Expected Results	Passed?
1	Land on ‘Downloading a Virus’	Test the act of landing on the Downloading a Virus location and making sure the severityOfVirus will be incremented.	Making it that the user has landed on the Downloading a Virus location.	A user has rolled the dice and landed on Downloading a virus, then the next user will take their turn and do the same	User has landed on this location.	The severity of virus will be incremented and the user will be moved to Safe Mode	Passed
2	Go To ‘Safe Mode’	Testing if a user can get out of Safe Mode by rolling the dice.	The user is in Safe Mode with severityOfVirus set to 1.	The user will roll the dice. severityOfVirus = 1	The dice will be rolled and will get a double	Whenever the user gets a double they will be moved out of Safe Mode.	Passed

3	Go To ‘Safe Mode’	Testing a user getting out of safe mode if they decide to use bits to get out.	The user is in Safe Mode with severityOfVirus set to 1.	Select option 2 in the menu severityOfVirus = 1	Select option 2 in the menu.	The user will be charged 1500 bits and moved out of Safe Mode.	Passed
4	Go To Safe Mode	Makes sure the system rejects an invalid input	A user has been placed into safe mode	Users enters 42	Users enters 42	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
5	Go To Safe Mode	Makes sure the system rejects an invalid input	A user has been placed into safe mode	Users enters “I am serious… and don’t call me Shirley”	Users enters “I am serious… and don’t call me Shirley”	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
6	Go To Safe Mode	Makes sure the system rejects an invalid input	A user has been placed into safe mode	Users enters 4.2	Users enters 4.2	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
7	Land On ‘Captcha’	Test the act of a user landing on the Captcha location and being given a random mini game to complete.	Making it so that the user has landed on the Captcha location.	User has rolled the dice and landed on Captcha.	A user will land on this location multiple times to ensure a different mini game is selected.	Each time a user lands on this location they will be presented with one of the random mini games. A message will be displayed to the user conveying this.	Passed

8	Land On ‘Captcha’	Test to make sure that whenever the user completes the mini game they will receive a reward.	Making it so that the user has landed on the Captcha location.	User completes the game.	Each of the mini games will be played and completed.	Whenever a mini game is completed the correct reward is given to the user.	Passed
9	Land On ‘Captcha’	Test to make sure that whenever the user does not complete a mini game they will be sent to Safe Mode	Making it so that the user has landed on the Captcha location.	User does not complete the game.	Each of the mini games will be played and not completed.	Whenever a mini game is not completed the user will be sent to Safe Mode and the severity of the virus will be set depending on the mini game. A message will be displayed to the user conveying this.	Passed
10	Land On ‘Captcha’	**Black Box-** Testing invalid input - No Data Entered. Captcha Game - TicTacToe	Land on Captcha and play Tic-Tac-Toe	Test Invalid Data - No Data Entered	Do not enter any data	The game outputs an error message and asks the user to enter compliant data.	Passed
11	Land On ‘Captcha’	**Black Box-** Testing invalid input - Invalid Data Type Entered. Captcha Game - TicTacToe	Land on Captcha and play Tic-Tac-Toe	Test Invalid Data - Invalid Data Type Entered	Enter Data of inavlid data-type	The game outputs an error message and asks the user to enter compliant data.	Passed
12	Land On ‘Captcha’	**Black Box-** Testing invalid input - Invalid coordinate entered. Captcha Game - TicTacToe	Land on Captcha and play Tic-Tac-Toe	Test Invalid Data - Invalid coordinate entered	Enter a non-existent coordinate.	The game outputs an error message and asks the user to enter compliant data.	Passed
13	Land on Captcha	Test to make sure that if a user enters a word that is not on the world list a message will be	The user has landed on a captcha location and the random game is wordle	Users enters 42	Users enters 42	The system will not take any tries away from the user and a message will be	Passed

		displayed telling the user this				displayed telling the user that their input is not in the word list	
14	Land on Captcha	Test to make sure that if a user enters a word that is not on the world list a message will be displayed telling the user this	The user has landed on a captcha location and the random game is wordle	Users enters vengeance	Users enters vengeance	The system will not take any tries away from the user and a message will be displayed telling the user that their input is not in the word list	Passed
15	Land on Captcha	Test to make sure that if a user enters a word that is not on the world list a message will be displayed telling the user this	The user has landed on a captcha location and the random game is wordle	Users enters 4.2	Users enters 4.2	The system will not take any tries away from the user and a message will be displayed telling the user that their input is not in the word list	Passed
16	Land on Captcha	Checking that an invalid response is not accepted in 'Hangman' mini game	A user lands on Captcha	Sandy	User enters incorrect response	The game outputs an error message and the user is asked again	Passed
17	Land on 'Corrupt File'	Testing Corrupt File taking a random file from a user.	Making it so that the user has landed on the Corrupt File location with a number of owned files.	User has rolled the dice and landed on Corrupt File.	The user will land on this location with some files currently owned.	Text will be displayed telling the user that one of their files (name of the file should be displayed) has been corrupted and has been moved to the recycling bin.	Passed
18	Land on 'Corrupt File'	Testing Corrupt File whenever a user has no files	Making it so that the user has landed on the Corrupt File location with no owned files.	User has rolled the dice and landed on Corrupt File.	The user will land on this location no files owned	A message will be displayed telling the user that they have no files to be	Failed - No message displayed explaining to the

						corrupted. Therefore nothing happens.	user what has happened.
19	Land on 'Corrupt File'	This is a re-run of test id 8, same procedure will be followed with the fixed code implemented.	N/A	N/A	N/A	N/A	Passed
20	Land on 'File'	Test to see if a user can download a file.	Making it that the user has landed on a file.	User has selected to download this location.	The user will land on a file which is not owned, then proceed to use the required bits to download the file.	A message will be displayed telling the user that they have successfully downloaded the file.	Passed
21	Land on 'File'	Test to see if a user can land on an unowned file and have the choice not to download it.	Making it that the user has landed on a file.	User selects not to download this location.	The user will land on a file which is not owned and will select the option to not download this file.	A message will be displayed telling the user that they have decided not to download the file	Passed
22	Land on 'File'	Test to make sure a user cannot download a file if it is already owned	Making it that the user has landed on a file.	N/A (This is because the option to download the file should not appear).	The user will land on a file which another user already owns.	The user will be given the option to play the brute force mini game	Passed
23	Land on File	Tests to make sure that if a user has less bits than the required amount the owner of the file will only get the remaining bits the user has. (e.g user who has 50 bits lands on a file	User landing on the file will have 20 bits while the owner will have 40 bits. The file being landed on will have a tempStorageSpace of.	User has landed on this location. Then entering in "42" for every attempt asked by the brute force mini game.	Enter "42" for every attempt asked by the brute force mini game making sure the correct answer to the	The users bits will go from 20 to 40 while the owners bits will go from 40 to 60.	Passed

		which charges them 100 the owner of the file will only get 50 instead of the full 100)			question is never entered.		
24	Land On File	Makes sure the system rejects an invalid input whenever the user is asked if they want to download the file	The game has been started with 2 players and the user has landed on a file.	Users enters 42	Users enters 42	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
25	Land On File	Makes sure the system rejects an invalid input whenever the user is asked if they want to download the file	The game has been started with 2 players and the user has landed on a file.	User enters “Take your stinking paws off me, you damned dirty ape”	User enters “Take your stinking paws off me, you damned dirty ape”	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
26	Land On File	Makes sure the system rejects an invalid input whenever the user is asked if they want to download the file	The game has been started with 2 players and the user has landed on a file.	Users enters 4.2	Users enters 4.2	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
27	‘Display Menu’	Test the users ability to save the game	The game will have started with 4 players with names dean, jb, scott and conor respectively, each with 1000, 230, 400, 300 bits respectively. dean will have 1 file SCRUM.docx, jb will have 2 files UI Patterns.pdf,Software Security.xlsx, scott will	Test that the game will save.	Everything in test initialization will be done. Then on scotts turn (to set him as the current player) the save game option will be selected.	The game will end and all of the relevant information (which can be seen in test initialization) will be saved in a text file.	Passed

			have 5 files Pair Programming.mp4,User Stories.docx,Software Implementation.pptx,UML Class Diagrams.pptx and Software Assurance.pdf and finally conor will have 2 files Software Process.mpp,Practical Project Management.mpp. All the files will be at the stage Downloaded with Software Implementation.pptx being installed. No players will be in quarantine or have any util passes				
28	'Display Menu'	**Black Box -** Test the users ability to interact with a set of options relating to the game.	Set up a game and initialise the players. Select 'Display Menu'	Test that the game will display the game rules.	Input the relevant selection.	The game rules will be displayed.	Passed
29	'Display Menu'	**Black Box -** Test the users ability to interact with a set of options relating to the game.	Set up a game and initialise the players. Select 'Display Menu'	Test that the game will execute the 'Vote to End Game' use case.	Input the relevant selection.	The use case will be correctly executed.	Passed
30	'Display Menu'	**Black Box -** Test the users ability to interact with a set of options relating to the game.	Set up a game and initialise the players. Select 'Display Menu'	Test that the game will display the board layout.	Input the relevant selection.	The game board will be displayed.	Passed
31	'Display Menu'	**Black Box -** Test the users ability to interact	Set up a game and initialise the players. Select 'Display Menu'	Test that the game will execute the 'Organise Files' use case.	Input the relevant selection.	The use case will be correctly executed.	Passed

		with a set of options relating to the game.					
32	'Display Menu'	**Black Box -** Test the users ability to interact with a set of options relating to the game.	Set up a game and initialise the players. Select 'Display Menu'	Test that the game will execute the 'Roll Dice' use case.	Input the relevant selection	The use case will be correctly executed.	Passed
33	'Display Menu'	**Black Box -** Test the users ability to interact with a set of options relating to the game.	Set up a game and initialise the players. Select 'Display Menu'	Test a non-existent option.	Input a non-existent selection	The game will ask the user to select a valid	Passed
34	Display Menu	Checking that an invalid response is not accepted in option menu	The option menu is displayed to the player	Spongebob	User enters incorrect response	The game outputs an error message and the user is asked again	Passed
35	'Vote to Finish Game'	**Black Box -** Testing the users' ability to decide to vote to quit the game.	Set-up a game and initialise the players. Execute the 'Display Options' use case. Select 'Vote to End Game'	Majority of players select 'Yes'.	Cycle through each player and ensure the number of 'Yes' votes outweighs the number of 'No' votes.	The result is displayed and the game returns to the title screen..	Passed
36	'Vote to Finish Game'	**Black Box -** Testing the users' ability to decide to vote to keep playing.	Set-up a game and initialise the players. Execute the 'Display Options' use case. Select 'Vote to End Game'	Majority of Players select 'No'.	Cycle through each player and ensure the number of 'No' votes outweighs the number of 'Yes' votes.	The result is displayed and the game continues as normal.	Passed
37	'Vote to Finish Game'	**White Box -** Testing the game will continue if the vote produces a tie.	Set-up a game and initialise the players. Execute the 'Display Options' use case. Select 'Vote to End Game'	Equal number of players vote 'Yes' and 'No'	Cycle through each player and ensure the number of 'Yes'	The result is displayed and the game continues.	Passed

					votes equals the number of 'No'		
38	'Vote to Finish Game'	Testing if an invalid input was entered (e.g anything that is not y/n or yes/no)	Set-up a game and initialise the players. Execute the 'Display Options' use case. Select 'Vote to End Game'	Enter number 3	Enter the number 3 whenever first prompted for a vote.	A message is displayed to the user telling them to enter either yes or no.	Passed
39	'Organise File'	Test the games ability to tell whether a user has 'downloaded' all files of the same type before attempting to set them up further.	Set up a game and ensure the player has 'dowloaded' no more than 2/3 files of the same file type.	Attempt to execute this use case.	Select a file to attempt to set-up.	The system will output a message telling the player they cannot set up the selected file until all files of the same type have been downloaded.	Passed
40	'Organise File'	Test that the user can advance the status of a file when the required conditions are met.	Set-Up a game and ensure the player has 'downloaded' an entire group of files of the same type.	Test the users ability to set up a file.	Select one of the files and choose to set it up.	The user will be charged bits in exchange for the set up of the file.	Passed
41	'Organise File'	Test the games ability to detect an invalid input.	Set-Up a game and ensure the player has selected 'Organise Files'	Test a non existent option.	Select a non-existent option.	The game should ask the user to pick again.	The system did not perform as expected and thus failed the test - the game crashes.
42	'Organise File'	Test the games ability to detect an invalid input.	Set-Up a game and ensure the player has selected 'Organise Files'	Test a non existent option.	Select a non-existent option.	The game should ask the user to pick again.	FIXED - Added condition to if statement to catch integer values less than 1.

43	‘Play Memory Game’	To test that the user can obtain files held in the ‘Recycling Bin’.	The recycling bin has been initialised with several files.	Test that upon entering an existing file, it is removed from the recycling bin and assigned to the player.	Input an existing file	The file is assigned to the player and removed from the Recycling Bin.	Passed
44	‘Play Memory Game’	To test that the user will not obtain files held in the ‘Recycling Bin’ if the input file is not in the ‘Recycling Bin’	The memory game is ‘running’.	Test that upon entering a non-existent file the game informs the player of this.	Input a non-existent file. E.g The Angel Isligntion.docx	The game tells the player the file isn't in the recycling bin and will accept another input.	The system did not perform as expected and thus failed the test - The game returned no message to the user.
45	‘Play Memory Game’	To test that the user will not obtain files held in the ‘Recycling Bin’ if the input file is not in the ‘Recycling Bin’	The memory game is ‘running’.	Test that upon entering a non-existent file the game informs the player of this.	Input a non-existent file.	The game tells the player the file isn't in the recycling bin and will accept another input.	Passed
46	‘Play Memory Game’	To test that the system will catch if the user hasn’t entered anything.	The memory game is ‘running’.	Test that when entering nothing the game informs the player of this.	Input Nothing.	The game tells the player they haven’t entered anything and will accept another input.	The system did not perform as expected and thus failed the test - the game returned no message to the user.
47	‘Play Memory Game’	To test that the system will catch if the user hasn’t entered anything.	The memory game is ‘running’.	Test that when entering nothing the game informs the player of this.	Input Nothing.	The game tells the player they haven’t entered anything and will accept another input.	Passed
48	‘Play Memory Game’	To test that the user can choose when to quit the memory game.	The memory game is ‘running’.	Test that upon entering the termination case the memory game stops.	Input the termination case.	The memory game terminates.	Passed

49	‘Land on Recycling Bin’	Test that the user can roll the dice to land on the ‘Recycling Bin’ tile.	The game has been set up and players have been initialised.	Test that upon landing on ‘Recycling Bin’ the appropriate information is displayed.	Roll the dice so that the player lands on the ‘Recycling Bin’ tile.	The memory game use case will be executed.	Passed
50	‘Land on Recycling Bin’	Test that the game will not play the memory game if there are no files in ‘Recycling Bin’	The game has been set up and players have been initialised.	Test that upon landing on ‘Recycling Bin’ the appropriate information is displayed.	Roll the dice so that the player lands on the ‘Recycling Bin’ tile - with nothing in the recycling bin.	The memory game method will not run the game due to lack of files in the recycling bin. It will output a message about this.	Passed
51	‘Land on Recycling Bin’	Checking that an invalid response is not accepted in ‘Memory’ game	A user lands on Recycling Bin	Mr Krabs	User enters incorrect response	The game outputs an error message and the user is asked again	Passed
52	Land on 'File Explorer'	Testing the main flow of events after landing on a “File Explorer” location.	The user lands on the “File Explorer”	The user has rolled dice and landed on the “File Explorer” location.	The user has landed on the “File Explorer” location and the locations the user owns are checked.	The user chooses to jump to another location they own. If the user passes go, they will either get the full amount of bits or not.	Passed
53	Land on 'File Explorer'	Testing the alternate flow of events after landing on a “File Explorer” location.	The user lands on the “File Explorer”	The user has rolled dice and landed on the “File Explorer” location.	The locations the user owns are checked and the user chooses not to jump locations.	The player is exited from the menu allowing the next player to make their move.	Passed
54	Land on 'File Explorer'	Testing the alternate flow of events after landing on a “File Explorer” location.	The user lands on the “File Explorer”	The user has rolled dice and landed on the “File Explorer” location.	The locations the user owns are checked.	The user may not own any locations forcing them to let	Passed

						the next player move.	
55	File Explorer	Checking that an invalid response is not accepted for the destination	A user lands on file explorer	Patrick	User enters incorrect response	The game outputs an error message and the user is asked again	Passed
56	Display Title Screen	Test to make sure whenever the user has selected the Continue Game option if there are no saved games no option is presented to the user.	The user runs the program to display the 'Title' screen.	User has selected the Continue Game option on the title screen	User starts the program and selects Continue Game on the title screen.	A message is displayed to the user that there are no game saves	Passed
57	Display Title Screen	Test to make sure whenever the user has selected the Continue Game option if they have saved games they can select one and the game will continue.	The user runs the program to display the 'Title' screen. At least one game has been saved for the user to select	User has selected the Continue Game option on the title screen and selected game save 1	User starts the program and selects Continue Game on the title screen, then the user will select game save 1.	A message is displayed telling the user the game will continue, then the game will proceed to continue	Passed
58	Display Title Screen	Testing the main flow to display the "Title" screen. Ensuring that when the user selects a valid option it is accepted by the system. (either number the option is in the list or by typing out the name of the option).	The user runs the program to display the 'Title' screen.	Enters 1 and then Start Game	The user runs the program to display the 'Title' screen.	The "Title" screen is displayed and the user selects the start game option. First by entering the number then by entering the option phrase	Passed

Final Game Layout – D.L

The Diagram to the right is the draft board layout of our game.

You can also see a list of our different file extensions and their corresponding colour. The .one file extension will consume the most storage space to download while .mpp will consume the least.

There is an arrow indicating the flow of the game.

Each location (square) will be stored within an array called locationArray, it will be stored in the order of the flow of the game. E.g GO is at index 0, then Old Kent Road.mpp is index 1, Corrupt File is index 2, etc.

The files (squares with a coloured bar on them) can be downloaded by a player only whenever they are on that location. Then on any turn the player can start the setup process, before the user can start installing the file they must go through 3 different stages of setup.

The file explorer locations will allow the user to move directly to any of the files that they are currently downloading at the expensive of some storage space.

The Util Pass location means that the player will be able to collect the Go amount if they are moving using file explorer.

Corrupt file will take a random file from the user and add this file to recycling bin.

Recycling Bin will allow the user to get all the files in it if they can remember the order, they were added to recycling bin in.

Downloading a virus will send the player to Safe Mode

Safe Mode is where the player will be until they roll a double or they have reached the amount of time they have to spend in quarantine.

PDF

DOCX

XLSX

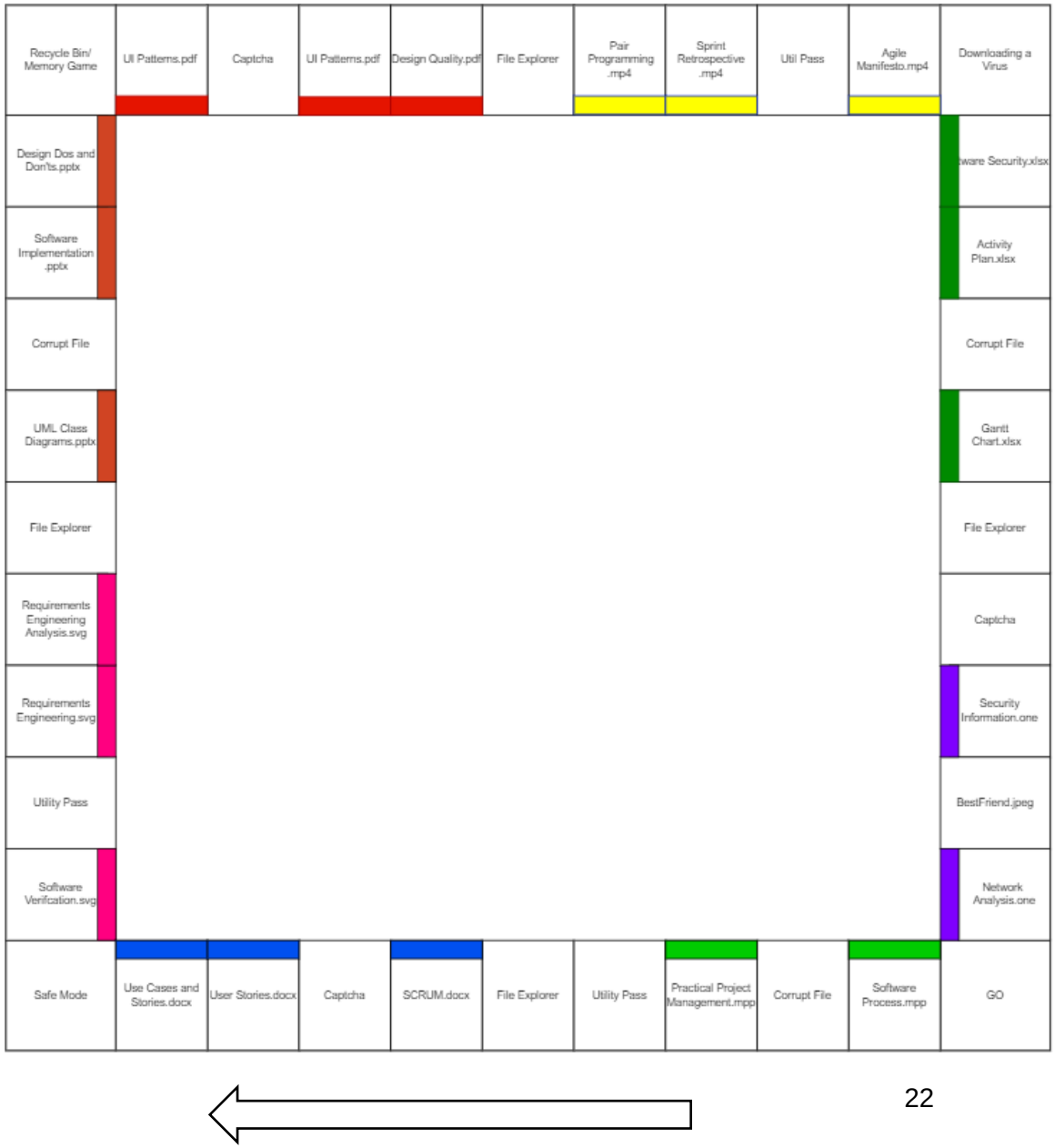
PPTX

ONE

MPP

MP4

SVG



Appendix

Weekly Minutes – S.McD

Minutes for CSC2058 Project 59 Week commencing 23-Jan Date of this minute 28-Jan

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald

- Picked use cases to test
- Decided on approach to coding
- Planned acceptance tests

Name (2): Dean Logan

- Picked use cases to test
- Decided on approach to coding
- Planned acceptance tests

Name (3): Conor Nugent

- Picked use cases to test
- Decided on approach to coding
- Planned acceptance tests

Name (4): John B Higgins

- Picked use cases to test
- Decided on approach to coding
- Created a template for the acceptance testing.
- Planned acceptance tests

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald
- Finish acceptance test plans

Name (2): Dean Logan
- Finish acceptance test plans

Name (3): Conor Nugent
- Finish acceptance test plans

Name (4): John B Higgins
- Finish acceptance test plans

Minutes for CSC2058 Project 59 Week commencing 30-Jan Date of this minute 04-Feb

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald
- Started on interface designs

Name (2): Dean Logan
- Started on interface designs

Name (3): Conor Nugent
- Started on interface designs

Name (4): John B Higgins
- Wrote template for interface designs
- Started on interface designs

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald
- Finish interface designs

Name (2): Dean Logan
- Finish interface designs

Name (3): Conor Nugent
- Finish interface designs

Name (4): John B Higgins
- Finish interface designs

Minutes for CSC2058 Project 59 Week commencing 06-Feb Date of this minute 09-Feb

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald
- Began coding designated classes

Name (2): Dean Logan
- Began coding designated classes

Name (3): Conor Nugent
- Began coding designated classes

Name (4): John B Higgins
- Began coding designated classes

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald

- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create JUnits if appropriate

Name (2): Dean Logan

- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create JUnits if appropriate

Name (3): Conor Nugent

- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create JUnits if appropriate

Name (4): John B Higgins

- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create JUnits if appropriate

Minutes for CSC2058 Project 59 Week commencing 13-Feb Date of this minute 16-Feb

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald

- Continue coding classes
- Continue developing JUnits

Name (2): Dean Logan
- Continue coding classes
- Continue developing JUnits

Name (3): Conor Nugent
- Continue coding classes
- Paired Programming session with JB
- Continue developing JUnits

Name (4): John B Higgins
- Continue coding classes
- Paired Programming session with Conor (Driver)
- Continue developing JUnits

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (2): Dean Logan
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (3): Conor Nugent
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (4): John B Higgins
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Minutes for CSC2058 Project 59 Week commencing 20-Feb Date of this minute 23-Feb

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald

- Continue coding classes
- Continue developing JUnits
- Paired Programming session with Dean

Name (2): Dean Logan

- Continue coding classes
- Continue developing JUnits
- Paired Programming session with Scott (Driver)

Name (3): Conor Nugent

- Continue coding classes
- Continue developing JUnits

Name (4): John B Higgins

- Continue coding classes
- Continue developing JUnits

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald

- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (2): Dean Logan

- Continue coding

- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (3): Conor Nugent

- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (4): John B Higgins

- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Minutes for CSC2058 Project 59 Week commencing 27-Feb Date of this minute 02-Mar

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald

- Continue coding classes
- Continue developing JUnits
- Paired Programming with JB

Name (2): Dean Logan

- Continue coding classes
- Continue developing JUnits
- Paired Programming with Conor

Name (3): Conor Nugent

- Continue coding classes
- Continue developing JUnits

- Paired Programming with Dean

Name (4): John B Higgins
- Continue coding classes
- Continue developing JUnits
- Paired Programming with Scott

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (2): Dean Logan
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (3): Conor Nugent
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Name (4): John B Higgins
- Continue coding
- Update acceptance tests if complete
- Update use case descriptions if required
- Create additional JUnits if appropriate

Minutes for CSC2058 Project 59 Week commencing 06-Mar Date of this minute 09-Mar

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.

Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald

- Finished coding classes
- Began final acceptance testing
- Continued JUnit testing

Name (2): Dean Logan

- Finished coding classes
- Began final acceptance testing
- Continued JUnit testing

Name (3): Conor Nugent

- Finished coding classes
- Began final acceptance testing
- Continued JUnit testing

Name (4): John B Higgins

- Finished coding classes
- Began final acceptance testing
- Continued JUnit testing

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald

- Make any corrections to code
- Take screenshots of testing code
- Continue JUnit testing if needed

Name (2): Dean Logan

- Make any corrections to code
- Take screenshots of testing code
- Continue JUnit testing if needed

Name (3): Conor Nugent

- Make any corrections to code
- Take screenshots of testing code
- Continue JUnit testing if needed

Name (4): John B Higgins

- Make any corrections to code
- Take screenshots of testing code
- Continue JUnit testing if needed

Minutes for CSC2058 Project 59 Week commencing 13-Mar Date of this minute 16-Mar

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald
- Finalise JUnit tests
- Finalise screenshot tests
- Gave code to friends and/or family to test
- Checked adherence to process

Name (2): Dean Logan
- Finalise JUnit tests
- Finalise screenshot tests
- Gave code to friends and/or family to test
- Checked adherence to process

Name (3): Conor Nugent
- Finalise JUnit tests
- Finalise screenshot tests
- Gave code to friends and/or family to test
- Checked adherence to process

Name (4): John B Higgins
- Finalise JUnit tests
- Finalise screenshot tests
- Gave code to friends and/or family to test
- Checked adherence to process

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald
- Make corrections to JUnit tests
- Check work against adherence to process

Name (2): Dean Logan
- Make corrections to acceptance tests
- Check work against adherence to process

Name (3): Conor Nugent
- Check work against adherence to process

Name (4): John B Higgins
- Check work against adherence to process

Minutes for CSC2058 Project 59 Week commencing 21-Mar Date of this minute 24-Mar

The following team members were present on Teams (if not Teams, indicate platform) when minutes were discussed:

Name (printed/typed)	Signature (agreed bitmap or initials)
Scott McDonald	S.McD.
Dean Logan	D.L.
Conor Nugent	C.N.
John B Higgins	J.B.H.

Task Reporting (Briefly list the progress for each team member in the last week.*)

Name (1): Scott McDonald
- Compiled Report
- Make any final changes
- Additional Testing and screenshots

Name (2): Dean Logan
- Compiled Report
- Make any final changes
- Additional Testing and screenshots

Name (3): Conor Nugent
- Compiled Report
- Make any final changes

- Additional Testing an screenshots

Name (4): John B Higgins

- Compiled Report
- Make any final changes
- Additional Testing and screenshots

*Printouts giving an overview of interim deliverables may be added as a supplement to these minutes.

Actions Planned (Briefly list the actions required of each team member for the next week.)

Name (1): Scott McDonald

- Submit

Name (2): Dean Logan

- Submit

Name (3): Conor Nugent

- Submit

Name (4): John B Higgins

Submit

Possible Secure System Features – D.L

Within this project there are plenty of opportunities to implement future security features if this system were to come available online.

The first of these features would be encryption of the various text and csv files used within the system.

For the captcha word list and the questions and answers files it would be possible to use a hashing algorithm as they are only ever used to compare against user input. For example, we could use MD5 to hash all the answers within the questions and answer file then whenever the user enters in their answer you can hash their answer and see if the 2 hash values match to check if their answer is correct. This avoids encrypting and decrypting all the data within the document improving efficiency and is more secure as hashing is a one-way function meaning that a bad actor will not be able to decrypt the answers themselves.

For the save game files it would be possible to use some form of encryption algorithm like RSA to generate a public and private key which can be used by the system to encrypt and decrypt a save file.

Another security feature which could be implemented would be making the files password protected, this means that an unauthorised user would not be able to gain access to these files further preventing any way a malicious user could break the game.

A security feature which is already implemented in the system is checking the length of certain user inputs to prevent buffer overflow. This can be seen when a user enters in their name or whenever deciding the number of players.

Developmental Process Methodology – D.L

Our team's approach to development is best described as an "agile like" approach. We decided it was best to divide up the classes amongst ourselves (Jb taking Player, DownloadVirus and Go. Conor taking Captcha, Dice and Utility. Dean taking File, Game, Location and SafeMode. Scott taking CorruptFile, FileExplorer and RecyclingBin) and have a minimum of 1 meeting a week to catch up on development and discuss any changes that we feel like need to be made, whether it be how certain aspects of the game works, adding additional features or changing output.

During development we also had meetings which were dedicated to fixing bugs as a group allowing the bugs to be found faster. This also led to us having some paired programming sessions.

This "agile like" approach also allowed us to perform unit testing on each of our methods allowing for simple errors to be caught early on in the developmental process.

We avoid taking a waterfall approach to the project as it suited us better to have the flexibility to make changes and iterate on each other's work. We also avoided a traditional agile approach of having daily stand-up meetings, etc in favour of weekly meetings as it would be difficult for everyone to consistently get on a daily call.

Testing Methodology – D.L

Our team's methodology to testing was relatively simple, we decided it was best to split up the use cases evenly, between the group members, and create multiple test cases for each use case to ensure maximum code coverage.

When splitting up the use cases we decided that each group member should have some use cases where they were not involved with the developmental process. We decided this was the best approach as it ensured each member would have an opportunity to perform both white box and black box testing.

We also created JUnit tests for each of the classes to help during testing. However, we decided to avoid creating JUnit tests for any methods which required user input and/or had a random element within the method (e.g rolling the dice). This is because we believe that it would lead to inconsistent JUnit test results. As if someone else tried to run one of these JUnit tests and didn't know the correct input to use for the test it would appear that the method being tested doesn't work, when in actuality, it does. The inconsistent JUnit test result is the same reason why we decided it was not necessary to test any methods which contained a random element to them as simply running the test multiple times will have different results.

Moreover, we decided to perform unit testing throughout the developmental process and focus on integration testing whenever we were completing the test plan. This approach was taken as we found that certain methods would be used across multiple use cases so to avoid repetition of testing the same method, we decided to unit test the methods during development, which also allowed for quick bug fixing.

While carrying out each test case we decided it was best to perform a mixture of different tests before we marked it as passed.
The first thing each member did when approaching a test case was check if there were any corresponding JUnit tests which could be run to check for any errors.
Then we would create a test method which would allow us to only run the section of the code which is being tested (e.g instead of playing through the game and rolling the dice until we land on a certain location the function would just run the corresponding method to this location). This allowed us to check if sections of the code were still working if any changes were made similar to JUnit tests.
Once this was done, we would then play through the game until we reach the point where the test case is. This is to ensure that the section of the code being tested works within the entire system.

If a tested failed it would be documented, then the testing process for the test case would begin again.

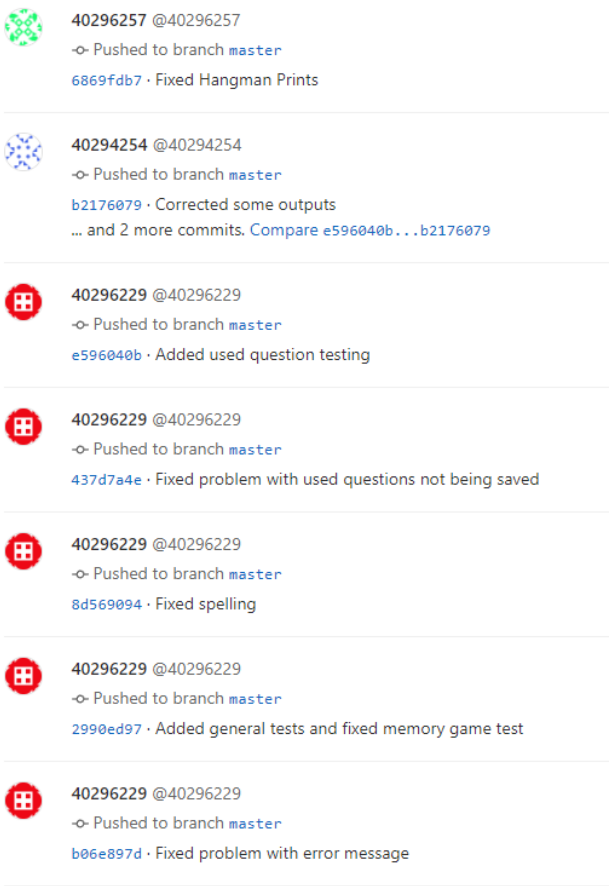
After all test cases were completed to ensure there were no bugs within the code, we decided to give our program to friends and family for further black box testing and to receive feedback allowing us to improve aspects of the game.

We believe this was the best approach to testing as it allowed us to perform all of the different types of testing (unit testing, integration testing, white box, black box and JUnit) at each stage of the developmental process allowing us to have confidence in the systems robustness.

GitLab Activity Records – D.L

Dean Logan – 40294254
John Higgins – 40291622
Conor Nugent – 40296257
Scott McDonald - 40296229

Note: Activity record is sorted from most recent to least recent



40296229 @40296229

Pushed to branch `master`

`cac78ce4` · Changed JUnit to not require user input

40291622 @40291622

Pushed to branch `master`

`7f0d8167` · Fixed Spelling

40291622 @40291622

Pushed to branch `master`

`f7a38b12` · Added Further Validation to Captcha & JUnit Testing

40296229 @40296229

Pushed to branch `master`

`73db882f` · Added comments

… and 1 more commit. Compare `b835883a...73db882f`

40296229 @40296229

Pushed to branch `master`

`b835883a` · Added method to improve cohesion

40296229 @40296229

Pushed to branch `master`

`3fd4847d` · Completed JUnits - removed input tests

40294254 @40294254

Pushed to branch `master`

`9bc01a1a` · Some changes to testFileExplorer

40296229 @40296229

Pushed to branch `master`

`fd2492fd` · Added new methods to improve flow

40294254 @40294254

Pushed to branch `master`

`905b0b35` · corrected some outputs and fixed some problems in saveGame

40294254 @40294254

Pushed to branch `master`

`d290623f` · Correcting some output messages

40294254 @40294254

Pushed to branch `master`

`f312b614` · Added a JUnit test for the Captcha and Utility class

40294254 @40294254

Pushed to branch `master`

`5d9f80a9` · Implemented the Save Game feature

40296229 @40296229

Pushed to branch `master`

`3dd887ff` · Fixed problem with incorrect output message for landing on file exp...

40294254 @40294254

Pushed to branch `master`

`9614a649` · Made it so that the questions for brute force game come from a csv ...

40296229 @40296229

Pushed to branch `master`

`c4fa25a6` · Added outputs to tests for ss

40296229 @40296229

Pushed to branch `master`

`dcdfc4fd` · Added message informing user they've passed GO and got 200 bits

40296229 @40296229

Pushed to branch `master`

`485c8a79` · Matched method with use case alternate flow.

40296229 @40296229

Pushed to branch `master`

`8bcac3ad` · Updated output messages

40296229 @40296229

Pushed to branch `master`

`34c7521c` · Fixed bug with error messages printing twice

40296229 @40296229

Pushed to branch `master`

`0ba210e7` · Player now looses bits when they have no files to corrupt.

40296229 @40296229

Pushed to branch `master`

`89e5e630` · Player now looses bits when they have no files to corrupt.

40294254 @40294254

Pushed to branch `master`

`d61dd56f` · Changed name of locations

… and 1 more commit. Compare `90fd91a8...d61dd56f`

40291622 @40291622

Pushed to branch `master`

`90fd91a8` · Fixed Validation Bug In Game Class

40294254 @40294254

Pushed to branch `master`

`d911dd8d` · Fixed bug in playBruteForceGame and refactored some code within other

40296229 @40296229

Pushed to branch `master`

`ae4d01bb` · Fixed bug with error message

40296229 @40296229

Pushed to branch `master`

`8fe24357` · Fixed error message

40296229 @40296229

Pushed to branch `master`

`327f60c0` · Added test util, test brute force mini game, test land on go, test ...

40296229 @40296229

Pushed to branch `master`

`76478fd2` · Added new JUnit test

40296229 @40296229

Pushed to branch `master`

`bf07b9f0` · Fixed bug that refused correct answers

... and 2 more commits. Compare `ea24f505...bf07b9f0`

40291622 @40291622

Pushed to branch `master`

`ea24f505` · JUnit Test for Go Class

40291622 @40291622

Pushed to branch `master`

`b0f58c06` · Added test case for memory game

40296257 @40296257

Pushed to branch `master`

`860c2dad` · Small part in File Explorer needs changed on line 108

40296229 @40296229

Pushed to branch `master`

`40dcbd81` · Bug Fixed - user can no longer input ID as well as name, fixed issue

---

40296229 @40296229

Pushed to branch `master`

`e0046896` · Finished JUnit testing

40296229 @40296229

Pushed to branch `master`

`099d827f` · Bug Fixed

40296229 @40296229

Pushed to branch `master`

`253949db` · All JUnit tests Completed

40296229 @40296229

Pushed to branch `master`

`3258def2` · All bugs worked out and class fully functioning

40296229 @40296229

Pushed to branch `master`

`266b59c2` · Completed all JUnit tests

40294254 @40294254

Pushed to branch `master`

`8903a898` · Some JUnit tests completed for CorruptFile and bug fixed in the

... and 2 more commits. Compare `dc082a60...8903a898`

40296257 @40296257

Pushed to branch `master`

`dc082a60` · Dice implemented

40294254 @40294254

Pushed to branch `master`

`100915f6` · Fixed some bugs in corrupt file (paired programming with Scott) and

---

40296229 @40296229

Pushed to branch `master`

`fcab5acc` · Index problem should be fixed and user no longer charged for winning

---

40296229 @40296229

Pushed to branch `master`

`25aaacf8` · Now makes file stage NotOwned

40296229 @40296229

Pushed to branch `master`

`279917ba` · Needs tested but potentially finished methods for taking file from

---

40296229 @40296229

Pushed to branch `master`

`c0ee751f` · Most up to date version

... and 3 more commits. Compare `3f6a5f2d...c0ee751f`

40294254 @40294254

Pushed to branch `master`

`3f6a5f2d` · Fixed problem with getWords in the Captcha class

... and 2 more commits. Compare `ade07af4...3f6a5f2d`

40296229 @40296229

Pushed to branch `master`

`ade07af4` · WIP - problem with index

40296229 @40296229

Pushed to branch `master`

`f677f142` · Not been properly changed - just basis for next steps

40296229 @40296229

Pushed to branch `master`

`aa56469a` · Needs tested - no longer loop, fares and util need looked at still

---

40296229 @40296229

Pushed to branch `master`

`b37fc143` · Added while loop and fixed else

40296229 @40296229

Pushed to branch `master`

`a3f03587` · Seeing if I can just copy and paste my code - needs tested

---

40294254 @40294254

Pushed to branch `master`

`f5741472` · Finished more tests with Captcha and fixed wordle game

---

40296257 @40296257

Pushed to branch `master`

`255768f6` · Connected Captcha and safemode but still need to fix hangman

---

40294254 @40294254

Pushed to branch `master`

`b80510c9` · Completed JUnit tests for Location and CorruptFile, also did some w...

40291622 @40291622

Pushed to branch `master`

`76af5dac` · Testing and Altercations to File

40291622 @40291622

Pushed to branch `master`

`25c077d1` · Gave Values to Properties / Display Rules

40294254 @40294254

Pushed to branch `master`

`0a31b3f0` · JUnit tests for the File class and helped fixed Conors problem

40296257 @40296257

Pushed to branch `master`

`1d9eaf26` · Need help on the game class line 432.

40291622 @40291622

Pushed to branch `master`

`8f838c55` · JUnit Tests For DownloadVirus

... and 5 more commits. Compare `541e12d0...8f838c55`

40296257 @40296257

Pushed to branch `master`

`541e12d0` · Words for Captcha games

40296257 @40296257

Pushed to branch `master`

`aed6bd7f` · Added list of words for captcha games

40294254 @40294254

Pushed to branch `master`

`3d50014c` · Some JUnit tests for SafeMode

40294254 @40294254

Pushed to branch `master`

`840af1e2` · Fixed Game class and added in some test cases

40294254 @40294254

Pushed to branch `master`

`8335d9c6` · Added in the test package and created an empty JUnit test case file,

40296257 @40296257

Pushed to branch `master`

`76db540d` · replaceAll(" ", "") instead of strip()

40296257 @40296257

Pushed to branch `master`

`749f8160` · Think hangman is pretty much done now

40296257 @40296257

Pushed to branch `master`

`54146934` · Merge remote-tracking branch 'origin/master'

... and 1 more commit. Compare `ff640877...54146934`

40294254 @40294254

Pushed to branch `master`

`ff640877` · Changed name of fileType back to FileType

40294254 @40294254

Pushed to branch `master`

`dcd29761` · Fixed all bugs which appeared in the latest push meaning the Game c...

40291622 @40291622

Pushed to branch `master`

`fe58b8fe` · Fixed TicTacToe, Added Display Rules, Game Refusing to Run now?

... and 2 more commits. Compare `a9e78c47...fe58b8fe`

40294254 @40294254

Pushed to branch `master`

`a9e78c47` · Fixed some bugs in brute force game

40294254 @40294254

Pushed to branch `master`

`2c782357` · Added enums for the file class and fixed some issues with the organise

40294254 @40294254

Pushed to branch `master`

`6dfdc764` · Fixed some bugs in the Game class and implemented the displayBoardL...

40294254 @40294254

Pushed to branch `master`

`0b9b0359` · Added a test class for all our testing functions

40294254 @40294254

Pushed to branch `master`

`b1ccb26d` · Added in the worlde mini game in captcha

40296229 @40296229

Pushed to branch `master`

`b09177bd` · Updated version - still wanting to tweak some more

40296257 @40296257

Pushed to branch `master`

`9e28977b` · Still need to fix the Captcha.

40294254 @40294254

Pushed to branch `master`

`09f9e671` · Implemented brute force mini game

40294254 @40294254

Pushed to branch `master`

`31063406` · Finished implementing the File class and fixed a few bugs in the Game

40294254 @40294254

Pushed to branch `master`

`49a30447` · Immplemented displayResults, finishGame and voteToFinishGame within...

40294254 @40294254

Pushed to branch `master`

`0522067c` · Just added in the goToSafeMode changes talked about during the meeting

40296229 @40296229

Pushed to branch `master`

`70711e02` · New comments added in

40296229 @40296229

Pushed to branch `master`

`a0497ba2` · My current very unfinished Corrupt file, file explorer, recycling bin

40291622 @40291622

Pushed to branch `master`

`66e2c9b4` · Undid the faulty solution for calling Game in location classes.

40294254 @40294254

Pushed to branch `master`

`00fa67f0` · organiseFiles has been coded but still needs to be tested

40294254 @40294254

Pushed to branch `master`

`14fd1be2` · Completed determinePlayerOrder, movePlayer and displayOptionsForPla...

40291622 @40291622

Pushed to branch `master`

`b4b52bb4` · Completed the 'givePlayerCollectAmount' method within the 'Go' class.

40291622 @40291622

Pushed to branch `master`

`4b4b173c` · Suggested Fix for calling Game class from within location classes.

40294254 @40294254

Pushed to branch `master`

`1f31fa21` · Deleted the "Test" classes, completed the "initialise" methods and the

40294254 @40294254

Pushed to branch `master`

`7d6bf097` · Added in a Controller class that contains a main function.

40296257 @40296257

Pushed to branch `master`

`e5bcba42` · Test message

40291622 @40291622

Pushed to branch `master`

`e1b615e5` · Testing Git on my computer

... and 2 more commits. Compare `97fa1b0d...e1b615e5`

40296229 @40296229

Pushed to branch `master`

`97fa1b0d` · Update src/Classes/File.java

40296229 @40296229

Pushed to branch `master`

`4f12b8d3` · Update File.java

40296229 @40296229

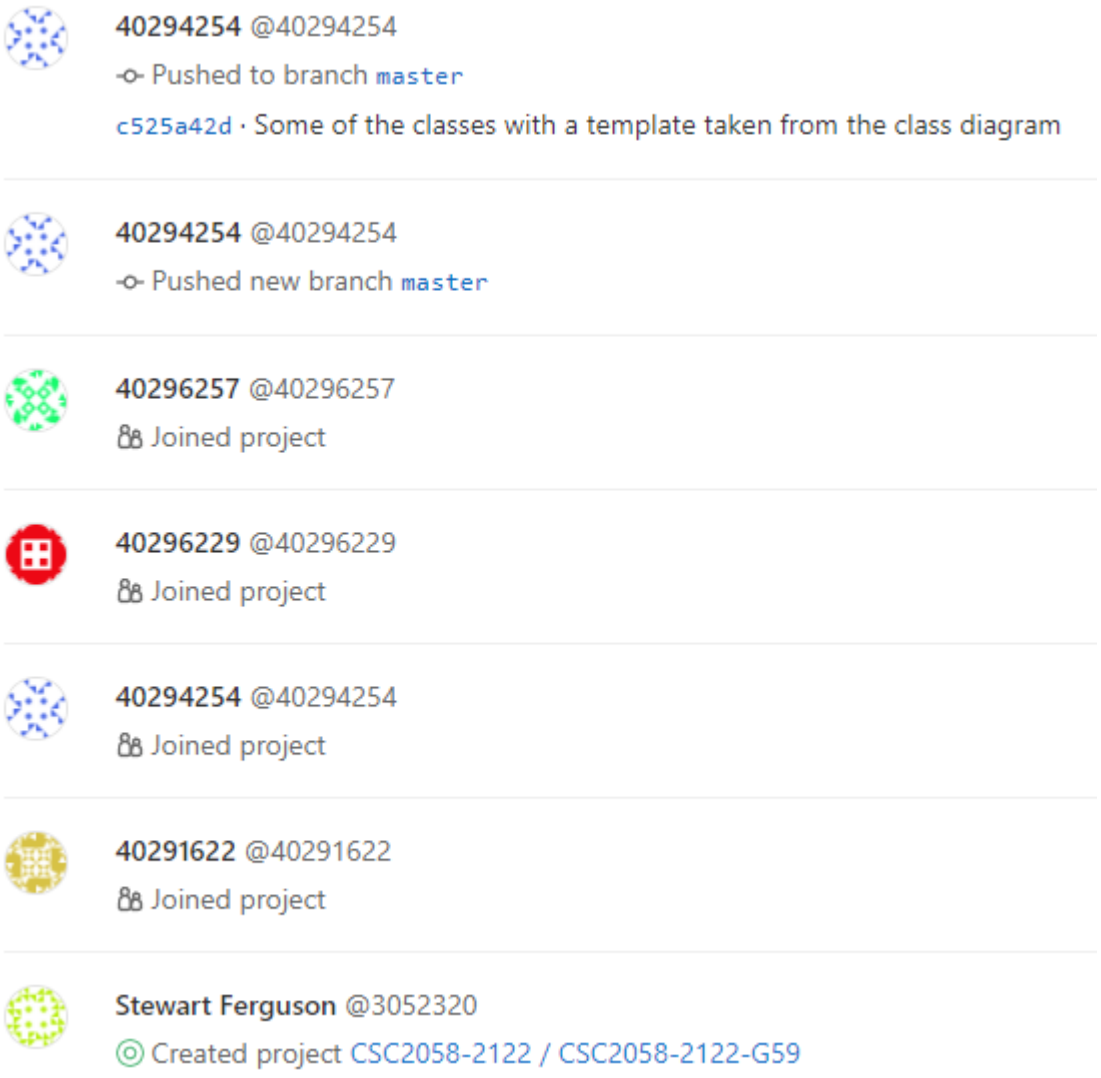
Pushed to branch `master`

`fccaf966` · Update File.java

40294254 @40294254

Pushed to branch `master`

`e18f96b8` · Added in template for the game class and made all the locations inh...



Revised Use Case Descriptions – D.L

Flow of events for the 'Roll Dice'	
Objective	**To Obtain A Randomly Generated Number of Places for the Player to Move.**
Pre-condition	*The Players Have Been Rotated*

Main Flow	1. The Player tells the interface to 'roll the dice'. 2. Two dice objects obtain random numbers between 1 and 6. 3. These numbers are added together. 4. Check the players bits to see if they are out of the game 5. The number rolled will be added onto the players current position 6. Then the use case for the corresponding location which they have landed on will trigger. 7. Then current player will be updated to the next player
Alternative Flows	At 1, the player may have been required to miss a turn, meaning they cannot roll the dice and the players should rotate again. At 4, The current player has no bits so they are out of the game At 4. If only one player has bits left this will trigger the finish game use case
Post-Condition	*Control is passed to the relevant 'Land On' use case.*

Flow of Events for the Set-Up Game	
Objective	**To set up the game on the 'Setup Game' screen and determine the player order.**
Precondition	*The 'Setup Game' option is selected on the 'Title' screen.*
Main Flow	1. The 'Setup Game' screen is displayed. 2. The player names are entered by the user. 3. The system will iterate through the players. 4. For each player the dice is rolled. 5. After every player has rolled the dice, the players are re-ordered based on who rolled the highest value. 6. The resulting order is displayed to the console. 7. Display Options' is called.
Alternative Flows	At 2, the user decides to direct back to the 'Title' screen instead of setting up the game. At 3, if only a single player was declared beforehand, a message is displayed to the console stating that more than one player is required

Post-condition	*The players have been placed into an order that will be referred to throughout the game as this is the order in which they will take their turns.*

Flow of Events for Go To 'Safe Mode'	
Objective	**Send the player from whatever tile they are currently on to the 'Safe Mode' tile as a punishment**
Precondition	*The player has landed on the 'Downloading a virus' tile*
Main Flow	1. Output text telling the user that they are being moved to the 'Safe Mode' file. 2. The player rolls a double to get out of 'Safe Mode'
Alternative Flows	At 2, if the player does not roll a double within the amount of goes specified by severityOfVirus, they have to use 50 bits and move forward by the number of spaces rolled on the dice At 2, the player decides to pay out
Post-Condition	*The next player can take their turn*

Flow of events for the 'Land on Recycle Bin'	
Objective	**To check for 'files' previously marked as being 'corrupt'.**
Pre-condition	*The player's position has been moved and they have landed on 'Recycle Bin'*
Main Flow	. The system checks for any 'files' that have been previously marked as 'corrupt' and loads them into an array that is **not displayed to the player.** . The 'Play Memory Game' use case is executed.

Alternative Flows	At 2, the system may not have found any files marked as corrupt, in which case the players rotate and the game returns to the 'Display Menu' use case.
Post-Condition	*The 'Play Memory Game' use case executes.*

Flow of events for the 'Play Memory Game'	
Objective	To allow the player to play a 'Memory Game' (framed as a 'Search Recycle Bin Function) that will allow them to win some of the 'corrupted files' based on how many they can remember.
Pre-condition	*The game has checked for corrupt files and has called the Play Memory Game use case.*
Main Flow	1. The corrupt file array is obtained. 2. The player is asked to enter the names of any files that they can remember have been marked as 'corrupt' throughout the game. 3. This input request will execute until a termination case is entered. 4. Every input is checked against the array of files. 5. If an input matches a file in an array the file is removed from the array, the 'recycling bin' and assigned to the player. 6. When the termination case is entered control is returned to 'Roll Dice'.
Alternative Flows	At 1, if there is no match in the array for the player's input, a message is displayed to the console stating no match was found. The use case will continue to execute until the termination case is entered.
Post-Condition	*The current player has gained all files that they were able to remember. Other files remain in the 'Recycling Bin'.*

Flow of events for Display Menu	
Objective	Present the user with a list of options relevant to playing the game
Pre-condition	*The game has started, and the player order has been determined*

Main Flow	1. User enters in an option that corresponds with one of the options on the list. 2. The 'Roll Dice' option is then selected and the two dice are rolled.
Alternative Flows	At 1, the user may select the option 'Display Board Layout'. In this case, the game displays a table listing the position, location name, and location of players on the board. The user is then returned to the menu. At 1, the user may select the option 'Display Rules'. In this case, the game displays text clarifying the rules of the game. The user is then returned to the menu. At 1, the user may select the option 'Organise FIles'. This will then run through the flow of events for Organise Files. The user is returned to the menu when they are finished. At 1, the user may select the option 'Vote to FInish Game'. This will then run through the flow of events for the Vote to Finish Game. The user is returned to the menu when it is finished and the users have decided to continue the game. At 1, the user may select the option 'Save Game'. This will create a text file with all the information needed for the game to continue then stop execution of the code.
Post-Condition	*The player takes their turn.*

Flow of events for Organise File	
Objective	Present the user with a list of all viable files that can be organised and then allow them to organise the selected file(s).
Pre-condition	*The game has started, and the player has selected the option to 'Organise Files' from the option list.*
Main Flow	1. The player enters a number that corresponds with one of the options on the list or types in the name of the file. 2. The player is shown the price for organising the file and is asked if they would like to continue. 3. It is checked if the user owns all types of this file. 4. The user is told they do not own all file types and is returned to the list to select another option. 5. The user selects the 'Back' option.

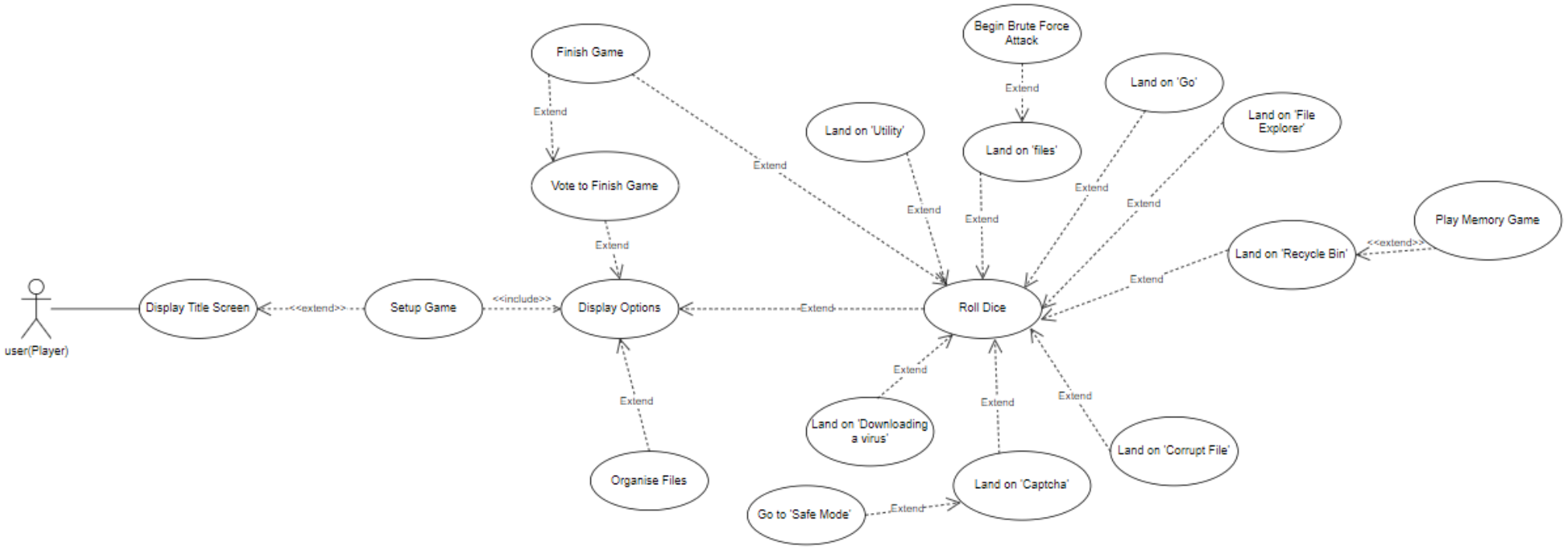
Alternative Flows	At 2, the player may choose not to organise the file. In this case, the user is returned to the list to select another option. At 4, the player may own all of the selected file types. In this case the file will be organised and the user will be returned to the list. At 5, the user may select another option. In this case the flow of events is repeated.
Post-Condition	*The player is returned to the option list.*

Flow of events for Vote to Finish Game	
Objective	Present the users with the option to end the game and return to the main menu.
Pre-condition	*The game has started, and the current player has select the 'Vote to Finish Game' option.*
Main Flow	1. Player 1 is asked if they would like to end the game (yes or no). 2. The following players in order are asked if they would like to end the game. 3. The majority of players select yes to end the game. 4. The game is ended and a leaderboard is displayed.
Alternative Flows	At 3, the majority of players may choose to continue the game. In this case, the current player is returned to the option list and the game continues.
Post-Condition	*The user is returned to the title screen.*

Flow of Events for Go To 'Title Screen'	
Objective	To display a list of options to the user whenever the program is run
Precondition	The user runs the program to display the 'Title' screen.
Main Flow	4. The title screen is displayed 5. The user selects 'Start Game' 6. The Setup Game use case is triggered

Alternative Flows	At 2, the user decides to load the game from a previous save 3. A menu will appear asking the user to select from one of the saved games, then whatever the selection is will load that game. At 2, the user decides to quit the game, this will cause the program to end
Post-Condition	N/A

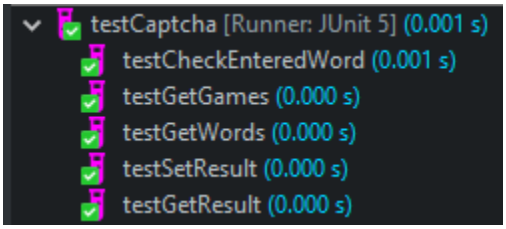
Revised Use Case Diagram – D.L



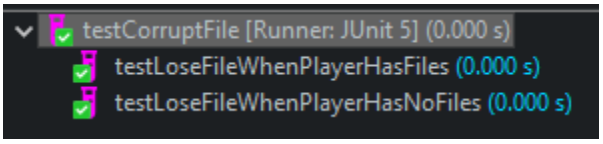
JUnit Test Runs – D.L

Note: As mentioned in Testing Methodology not all classes and methods were tested using JUnit testing

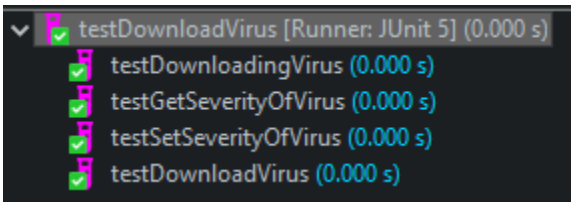
Captcha



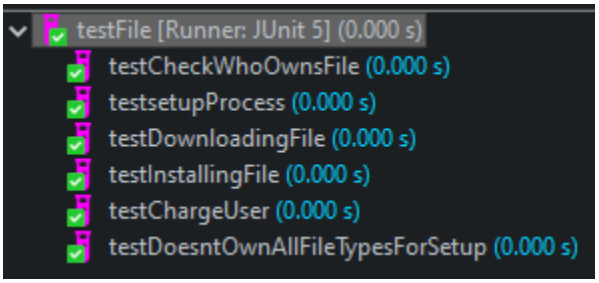
CorruptFile



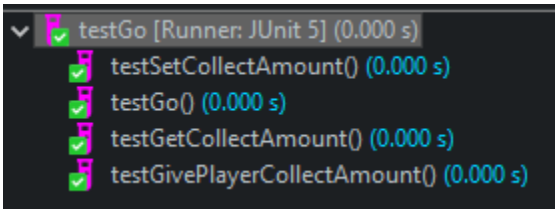
DownloadVirus



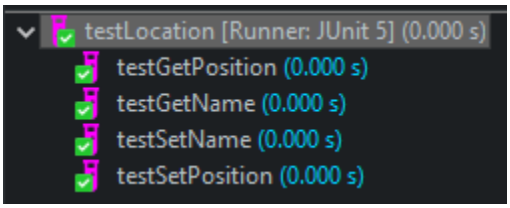
File



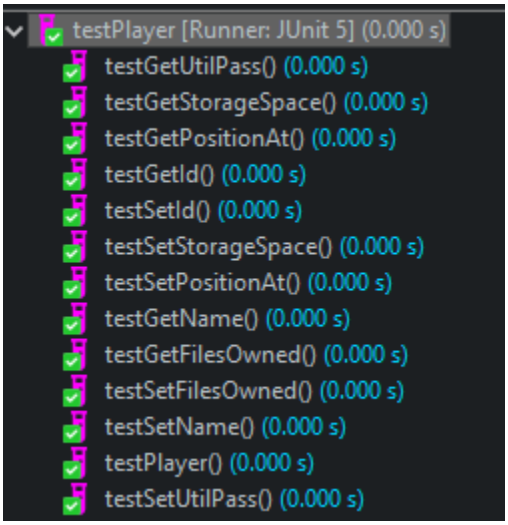
Go



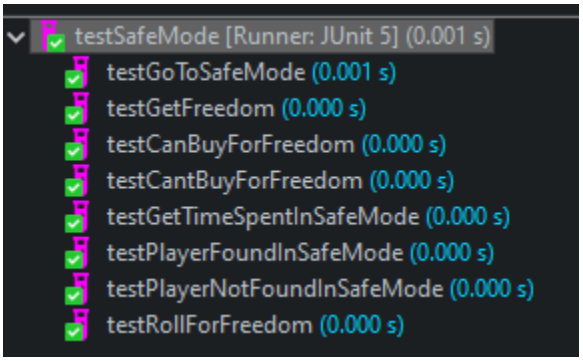
Location



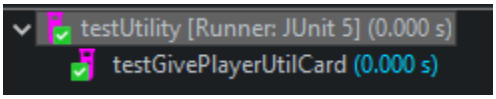
Player



SafeMode

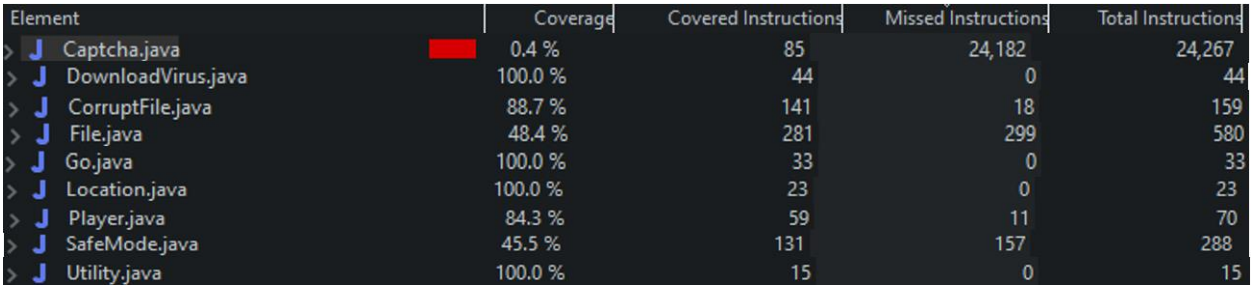


Utility



Code Coverage – D.L

Note: As mentioned in Testing Methodology not all classes and methods were tested using JUnit testing (The coverage for each class shown is for whenever the corresponding JUnit test was run. E.g the coverage for DownloadVirus shown was taken whenever the JUnit test testDownloadVirus was run).



Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
Captcha.java	0.4 %	85	24,182	24,267
DownloadVirus.java	100.0 %	44	0	44
CorruptFile.java	88.7 %	141	18	159
File.java	48.4 %	281	299	580
Go.java	100.0 %	33	0	33
Location.java	100.0 %	23	0	23
Player.java	84.3 %	59	11	70
SafeMode.java	45.5 %	131	157	288
Utility.java	100.0 %	15	0	15

Text User Interface Design – Continued (D.L)

ID / Classname	CLI Sample	Purpose of Message
Arial font for this	`Try and Keep this font (Courier New) if possible for the CLI cos it looks nice :)`	Arial font for this
DownloadVirus	`user.getName() + ", a virus has been detected in your directory.\nYou will now be moved to Safe Mode."`	Let the user know they have been moved to safe mode.
File Explorer - select destination location	`"\n" + user.getName() + ", please choose from the list below : "`	Asks the user to input a value corresponding with one of the options in the list.
File Explorer - no owned files	`"You have no files to travel to."`	Displays an error message to the user informing them that they do not have any files that can be travelled to.
File Explorer - incorrect data entered	`"Please enter a corresponding number of the full name and type of the location."`	Displays an error message to the user informing them of the expected response.
File Explorer - fare price	`"The fare will be" + fare + " bits." + "\n" + "Are you comfortable paying> (y/n): "`	Displays the cost of travel and asks the user if they still wish to travel
File Explorer - accept fare/new position	`"You have moved from " + user.getPositionAt() + " to " + destinationLocation.getPosition() + "." "Your new balance is " + user.getStorageSpace() + "."`	Displays the players old and new positions. The user is then informed of their remaining balance after travelling.
File Explorer - refuse fare	`"Exiting the menu."`	Outputs message informing user that they will no be exiting the menu.
File Explorer - Not enough bits to pay fare	`"You do not have enough bits for this action."`	Outputs message informing the user that they are not able to afford the fare.
File Explorer - travelling with utility pass	`"You have used a Utility pass to get 200 bits when passing Go. You now have " + user.getUtilPass() + " Utility passes remaining."`	Explains to the user that they have used their utility pass to receive the 200 bits for passing Go. It then informs them

		of the amount of utility passes remaining.
File Explorer - travelling without utility passes	`"You don't have a Utility pass." + "\n" + "Therefore, you don't get the 200 bits passing Go."`	Informs the user about why they are not receiving the 200 bits for passing Go.
RecyclingBin - enter response	`"Welcome " + user.getName() + "!" + "\n" + "On this location you have the opportunity to claim all corrupted files for zero cost." + "\n" + "All you have to do is recall the name of the file in full, i.e. 'BestFriends.jpeg'." + "\n" + "Failure to meet these simple requirements will result in the acquisition of no new properties and you majorly sucking." + "\n" + "There are currently " + corruptFilesLen + " files corrupted. " + "\n" + "You can type 'Back' at any time to exit the Memory Game." + "\n" + "Enter your guess: "`	Explains to the user how to play the game and provides a space for them to enter their response.
RecyclingBin - no corrupted files	`"There are no corrupted files." + "\n" + "Maybe next time?"`	
RecyclingBin - exit game	`"Exiting game."`	Informs the user that they are now exiting the memory game after entering back or no.
RecyclingBin - incorrect answer	`"Wrong. Enter both the name and type if the file. It is not case sensitive or particular about spaces`	Outputs an error message to the user informing them of the expected response and it's format.
RecyclingBin - Enter correct value	`"Congratulations! This file is worth " + fileSpaceRequired + " bits." + "\n" + "Are you happy to accept this file? (y/n): "`	Informs the user that they have entered a correct value and asks how they would like to proceed.
RecyclingBin - enter yes	`"You have successfully downloaded this file."`	Informs the user that the file is now downloaded.
CorruptFile - sends file to recycling bin	`"The file " + corruptedFile.getName() + " has been sent to the recycling bin." + "\n" + "The bits invested will not be reimbursed and the file cannot be bought by landing on it." + "\n" + "The only way to retrieve the file is to complete the memory game after landing on the recycling bin." + "\n" + "take a note of the name for " + corruptedFile.getName() + " as you will need to enter the name in full to retrieve it." + "\n" + "Beware however, as any player can play the Memory Game, and any player can take this file."`	Informs the user of the file that has been corrupted and the method to retrieving it, along with the risks involved with other players being able to take the file.

CorruptFile - user has no files	"You have no files to corrupt."	Displays to the user that they have no files to corrupt.
Game - Continuing from a save game	Please select a game to continue with 1. Game 1 2. Back Please select an option:	This is displaying a menu of game saves to the user allowing them to select one to continue with that game.
Game - Exiting the game	Game has ended. Goodbye!	This is telling the user that the game has been closed.
Game - Initialising players	The game is starting :) Please enter how many people will be playing the game: Enter player <iterates through how many players are playing the game> name:	Tells the user the game has started, then will allow one of the players to enter how many people will be playing the game. This will then cause the program to print a message asking the player to enter all of the players names.
Game - Initialising players error message (if the user enters something that is not an int for number of players)	Please enter an integer for the number of players	Tells the user that they have entered invalid data and gives a message indicating why it was invalid.
Game - Determining player order	Now lets roll to see what order everyone will be playing in! Whoever rolls the highest number will go first. <players name> you have rolled a <rolled value> <players name> you will go <place determined by number rolled>	Allows the players to see what order they will be playing in and how this order was determined.
Game - Display Board Layout	Position | Location | Player	This will be a table which will list all the positions, the corresponding locations and the players name (only if the player is at that position). This will help the players visualise their current position.

Game - Organise Files (Player owns no files)	`You have no files to organise`	Gives a message to the user allowing them to see that they have no files instead of just displaying nothing.

Game - Organise Files (Player owns files)	`Please choose from the list below:`
	`1. <name of file> - <files currentStage>`
	`2. Back`
	
	`Please select an option:`
		This will loop through all of the files the player owns then display them to the user with a back option. Allowing the user to see all the files that they own and what their current stage is. Allows the player to select one of their files to upgrade it.

Game - Setting up/Installing a file	`Do you want to continue setting up this file for <price of upgrading file> (y/n):`	Tells the player how many bits that are required to upgrade the file and gives them the option to do so or not.
Game - Setting up/Installing a file (If the player does not own all of the file types)	`You do not own all of the file types of this file`	Informs the player that they do not own all of the correct file types so they cannot start the setup.
Game - Setting up/Installing a file (If the player owns all of the file types)	`You have successfully progressed to <new stage> for the file <file name>`	Informs the player that their file has been upgraded and what the new stage is.
Game - Setting up/Installing a file (If the player does not select a valid option)	`Please select one of the files above`	Tells the user that they have entered invalid data and gives a message indicating why it was invalid
Game - Setting up/Installing a file (If the player owns does not have enough bits)	`You do not have enough space to continue the setup process for <file name>`	Informs the user that they do not have enough storage space to continue setup

Game - Vote To Finish Game	`The vote to end the game has begun`
	
	`<player name> do you wish to end the game? (y/n)`
		This will inform the players that a vote to finish the game has begun, then it will ask all of the players if they wish to end the game.

Game - Vote To Finish Game (Majority vote yes)	`The majority of players have voted to end the game.`	Informs the players that the majority vote was yes, therefore the game will end
Game - Vote To Finish Game (Majority vote no)	`The majority of players have voted to continue the game.`	Informs the players that the majority vote was no, therefore the game will continue
Game - Vote To Finish Game (vote count even)	`There was no majority. Since a Conclusion could not be reached, the game will continue.`	Informs the players that there was no majority, therefore the game will continue
Game - Vote To Finish Game (invalid input)	`Please enter either yes or no`	Tells the player that they have entered invalid data and gives a message indicating why it was invalid
Game - Finish Game	`<player name> you came in <placement in game> place! Congrats`	This will display all of the players' places whenever the game is over. It will print for each player in order of last to first.
Game - Land On File (invalid input)	`Please select either yes or no`	Tells the player that they have entered invalid data and gives a message indicating why it was invalid
Game - Land On File (file downloaded)	`You have successfully downloaded <file name>`	Informs the player that they have downloaded the file .
Game - Land On File (not enough storage space to download the file)	`You do not have enough space to download <file name>`	Informs the player that they cannot download the file at this time.
Game - Land On File (decided not to download file)	`You have decided not to download the file <file name>`	Informs the player that they have not downloaded the file.
Game - Land On File (current player owns the file)	`You own this file`	Informs the player that they already own this file.
Game - Land On File (another player owns the file)	`The user <player name> owns this file`	Informs the player that another player owns this file.

File - Brute Force Game	The password hint is: <question selected> Please enter password:	Gives the player the question which they have to answer and prompts them to enter a password. The player will have 5 attempts at answering the question.
File - Brute Force Game (answer is correct)	Well done the password is correct you no longer have to pay the owner of this file	Tells the player they have answered the question correctly and informs the player that they no longer owe any bits.
File - Brute Force Game (answer is incorrect)	You have ran out of tries <current player name>, you have given <owner of file> <price> bits	Informs the user that they have failed to guess the password and therefore will be giving the owner of the file bits.
Utility - Land On Utility	You have opened up utility and now have gained a util pass!	A message explaining to the user that they have now gained a utility card.
Captcha - Land On Captcha (tictactoe)	Welcome To Tic-Tac-Toe! <player name> you will be competing against the Tic-Tac-Toe AI. You will be 'X', the AI will be 'O. You will take your turn first, simply use the co-ordinates shown on the grid to place an 'X' 0 - - - 1 - - - 2 - - - 0 1 2 Enter your desired Vertical Co-ordinate: Enter your desired Horizontal Co-ordinate:	This is the design layout of the tic-tac-toe mini game. The “_” on the board indicates the different spaces where an X or an O can be placed
Captcha - Land On Captcha (hangman)	Welcome To Hangman! <player name> you will be figuring out the chosen 5 letter word by guessing 1 letter at a time. But be careful because you only have 6 incorrect guesses until a final guess of the word must be made. _____ Guess a letter:	This explains the rules of the hangman game to the user. The “____” will be filled up whenever a user guesses a correct letter of the word. With each individual “_” indicating an unknown letter of the word.
Captcha - Land On Captcha (Wordle)	Welcome To Wordle! <player name> you will be figuring out the chosen 5 letter word.	This explains the rules of wordle and prompts

	Guess the WORDLE in six tries. Each guess must be a valid five-letter word. After each guess, the colour of the guessed letters will change to show how close your guess was to the word. Grey meaning the letter is not in the word. Yellow meaning the letter is in the word but in the wrong place. Green meaning the letter is in the word and in the correct place. Please enter word 1: hello h Grey e Grey l Yellow l Yellow o Grey	the user to enter their first guess. Once they guess, each letter will be printed on a different line along with the associated colour as per the rules of the game.
Captcha - Land On Captcha (Game completed successfully)	Congratulations, you gained 100 bits.	A message is displayed to the player informing them they have gained 100 bits
Captcha - Land On Captcha (Game not completed)	Unlucky on not getting the word: <correct answer>	A message telling the user the correct answer indicating that they have failed the game.

Test Plan – Continued

Note: You can find a google document with all the test cases within a single table here:
https://docs.google.com/document/d/1_R74ik7sKn6RV6LGSIsO3QXTRl7Dhn2Y8DxBifYurmM/edit?usp=sharing
You can also find a google document with all the test case screenshots here:
https://docs.google.com/document/d/1B1451Ia_nxdAjDdZ-WZu1UlPhYdUZIh20qKKts0EUkQ/edit?usp=sharing
All content within these google documents can be found within this report they are only included here if you find it easier to read (e.g have testing screenshots open as you view the test plan).

D.L: 1-9, 13-15, 17-27, 61-66
J.B: 10-12, 28-33, 35-48
C.N: 49-50, 56-60
S.Mc: 16, 34, 51-55, 73-84

ID	Use Case Ref	Description of Test	Test Initialisation	Test Inputs	Test procedures	Expected Results	Passed?
59	Display Title Screen	Testing the main flow to display the “Title” screen. Ensuring that when the user selects a valid option it is accepted by the system. (either number the option is in the list or by typing out the name of the option).	The user runs the program to display the ‘Title’ screen.	Enters 2 and then Continue game	The user runs the program to display the ‘Title’ screen.	The “Title” screen is displayed and the user selects the continue game option. First by entering the number then by entering the option phrase	Passed
60	Display Title Screen	Testing the alternate flow to display the “Title” screen. Ensuring that when the user selects a	The user runs the program to display the ‘Title’ screen.	Enters 3 and then Exit	The user runs the program to display the ‘Title’ screen.	The “Title” screen is displayed but the user selects the “Quit Game” option which kills the program.	Passed

		valid option it is accepted by the system. (either number the option is in the list or by typing out the name of the option).				First by entering the number then by entering the option phrase	
61	Display Title Screen	Makes sure the system rejects an invalid input	The user runs the program to display the 'Title' screen.	Users enters 32	Users enters 32	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
62	Display Title Screen	Makes sure the system rejects an invalid input	The user runs the program to display the 'Title' screen.	Users enters "you can't handle the truth"	Users enters "you can't handle the truth"	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
63	Display Title Screen	Makes sure the system rejects an invalid input	The user runs the program to display the 'Title' screen.	Users enters 4.2	Users enters 4.2	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
64	Display Title Screen	Makes sure the system rejects an invalid input within the continue game menu	The user runs the program to display the 'Title' screen. Then selects continue game	Users enters 42	Users enters 42	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed

65	Display Title Screen	Makes sure the system rejects an invalid input within the continue game menu	The user runs the program to display the 'Title' screen. Then selects continue game	Users enters "You talking to me?"	Users enters "You talking to me?"	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
66	Display Title Screen	Makes sure the system rejects an invalid input within the continue game menu	The user runs the program to display the 'Title' screen. Then selects continue game	Users enters 4.2	Users enters 4.2	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
67	Setup Game	Test to make sure that whenever the user enters the number of players they are prompted for the names of each player	The user runs the program to display the 'Title' screen. Then selects Start Game	Enters 3	Enter 3 when prompted for the number of players that will be playing the game. Then when prompted for the names of the players enter random names to check how many times you were prompted for names.	User should be prompted to enter 3 names for players one after each other	Passed
68	Setup Game	Testing the main flow to set up the game.	The 'Setup Game' option is selected on the 'Title' screen.	The player names are entered by the user.	The system will iterate through the players and place them in a random order.	The players are placed in a random order and "Display Options" is called.	Passed

69	Setup Game	Testing the alternate flow to set up the game.	The ‘Setup Game’ option is selected on the ‘Title’ screen.	A player name was entered by the user.	The system will iterate through the players and place them in a random order.	Only one player was declared. Therefore, a message is displayed to the console stating that more than one player is required.	Passed
70	Setup Game	Test to make sure the user can only enter numbers that are in the range of 2-9 for the number of players	The user runs the program to display the ‘Title’ screen. Then selects Start Game	Enters -1 then 10	Enter -1 then enter 10 whenever prompted for the number of players	The system will display a message asking the user to enter a number within the range of 2-9 for each of the numbers being entered	Passed
71	Setup Game	Test to make sure that if the user enters in a string for the number of players the system does not crash	The user runs the program to display the ‘Title’ screen. Then selects Start Game	Enters “I love the smell of napalm in the morning”	Enters “I love the smell of napalm in the morning”	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
72	Setup Game	Test to make sure that if the user enters in a float number for the number of players the system does not crash	The user runs the program to display the ‘Title’ screen. Then selects Start Game	Users enters 4.2	Users enters 4.2	The system will display a message telling the user that they have entered invalid data and will ask the user to enter the data again	Passed
73	Finish Game	Testing the main flow to finish the game.	All the players, apart from one, have lost all of their bits or the majority of players vote to finish the game.	The majority of the players vote to finish the game.	The game will be finished.	The game is finished and the final game results are displayed.	Passed

74	'Finish Game'	Testing the final leaderboard is constructed correctly.	Set-up a game and initialise the players. End the Game and View the results.	Keep track of players bit amounts etc. so we know if the leaderboard is correct.	Keep track of players bit amounts etc. so we know if the leaderboard is correct.	The results displayed in the leaderboard should be correct.	Passed
75	Finish Game	Checking that when the player loses, their files are returned to the board	A users storage space has been set to 0.	User rolls dice	Another player lands on one of the previously owned files	The player will be able to download the file	Passed
76	Play 'Brute Force Attack' Mini Game	Testing that a brute force mini game is started.	User lands on an already owned property and starts to play the Brute Force mini game.	User rolls dice	The mini game is started	A random (and different) question is shown to the user with an entry box for the user to input an answer and a list of multiple answers with A, B, C before them..	Passed - user landed on file and game was started.
77	Play 'Brute Force Attack' Mini Game	Testing the results of the user answering the question correctly	Land on an already owned property and start Brute Force mini game.	User types in the correct answer	Enter an answer into the entry box.	The player does not have to give up bits after answering correctly	Passed
78	Play 'Brute Force Attack' Mini Game	Testing the results for when the user answers the question wrong.	Land on an already owned property and choose the option to play the brute force mini game.	User types in an incorrect answer.	Enter an answer into the entry box.	The player is given another chance.	Passed
79	Play 'Brute Force Attack' Mini Game	Testing the results of the user failing the mini game.	User has already failed 5 attempts.	User enters an incorrect answer.	Enter an answer into the entry box.	The player loses bits.	Passed
80	Play 'Brute Force Attack' Mini Game	Checking that the question is added to the used questions	Start Brute Force game	Expected response	User enters correct response	Question is added to the used questions array	Passed

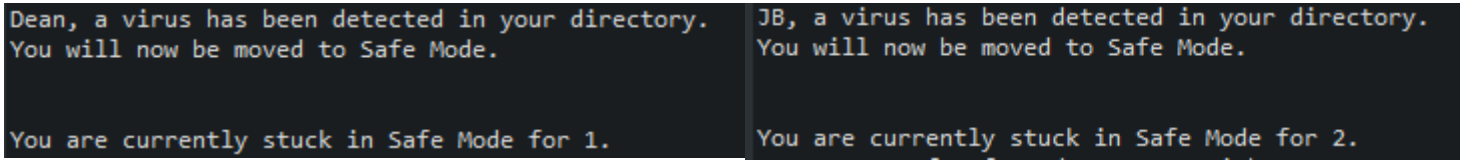
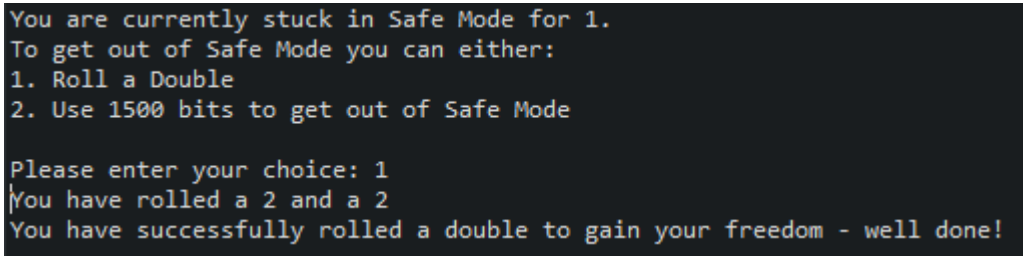
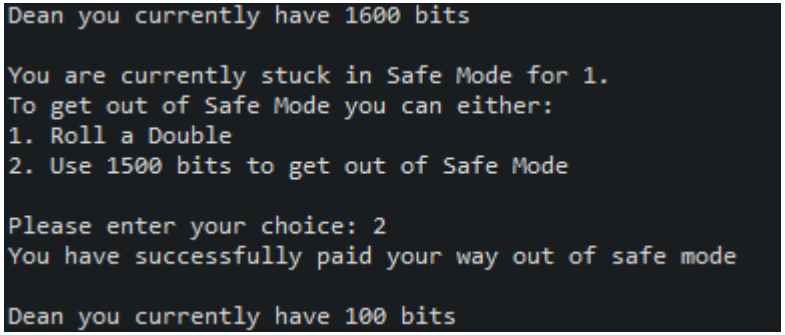
		when the user enters the correct answer					
81	Play ‘Brute Force Attack’ Mini Game	Checking that the question is added to the used questions when the user enters the incorrect answer	Start Brute Force game	N/A	User enters incorrect response	Question is not added to the used questions array	Passed
82	Land on “Go”’	Testing the flow of events for landing on/passing the ‘Go’ location.	User lands on or passes the ‘Go’ location.	User has rolled the dice and passed the ‘Go’ location.	User rolls the dice and passes/lands on the location.	The user is given an extra 200 bits.	Passed
83	Roll Dice	Testing the flow of events for rolling the dice and moving the player.	User selects to roll the dice from the list of options.	User selects ‘Roll Dice’	Two numbers are randomly generated between 1 and 6 and are then added up. This is then added onto the players current position.	User’s position is moved by that amount.	Passed
84	Land on ‘Utility’	Testing the flow of events after landing on a ‘Utility’ location.	User lands on a ‘Utility’ location.	User has rolled the dice and landed on the location.	The player is given a pass for use on the search engines.	The user is given one util pass.	Passed

Testing Screenshots

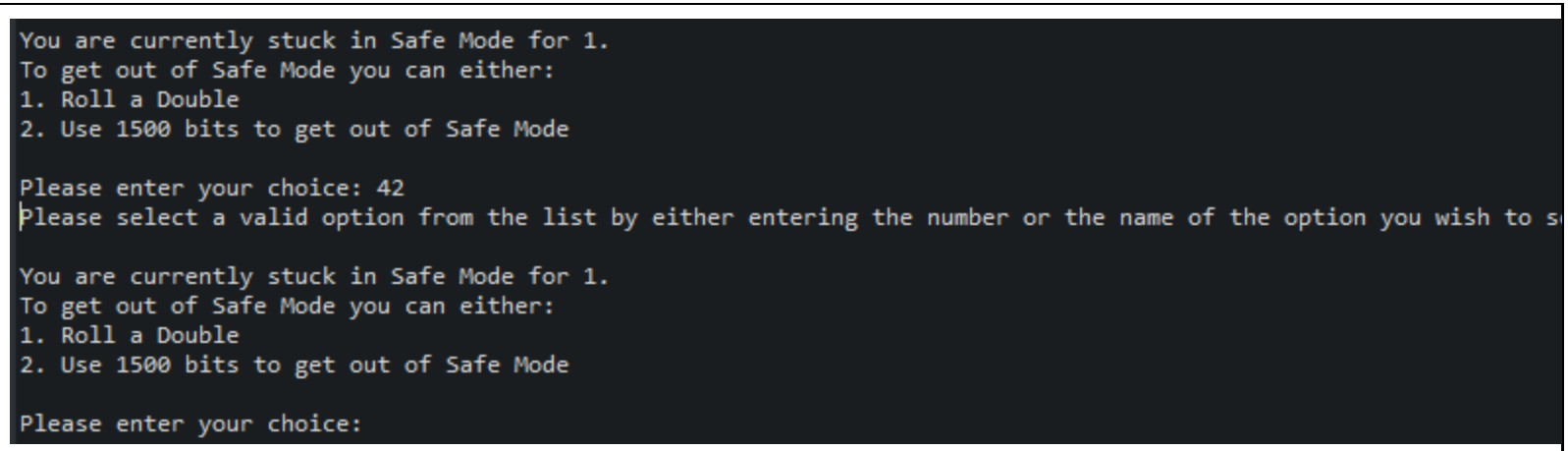
Note: You can find a google document with all the test cases within a single table here:
https://docs.google.com/document/d/1_R74ik7sKn6RV6LGSIsO3QXTRl7Dhn2Y8DxBifYurmM/edit?usp=sharing
You can also find a google document with all the test case screenshots here:
https://docs.google.com/document/d/1B1451Ia_nxdAjDdZ-WZu1UlPhYdUZIh20qKKts0EUkQ/edit?usp=sharing

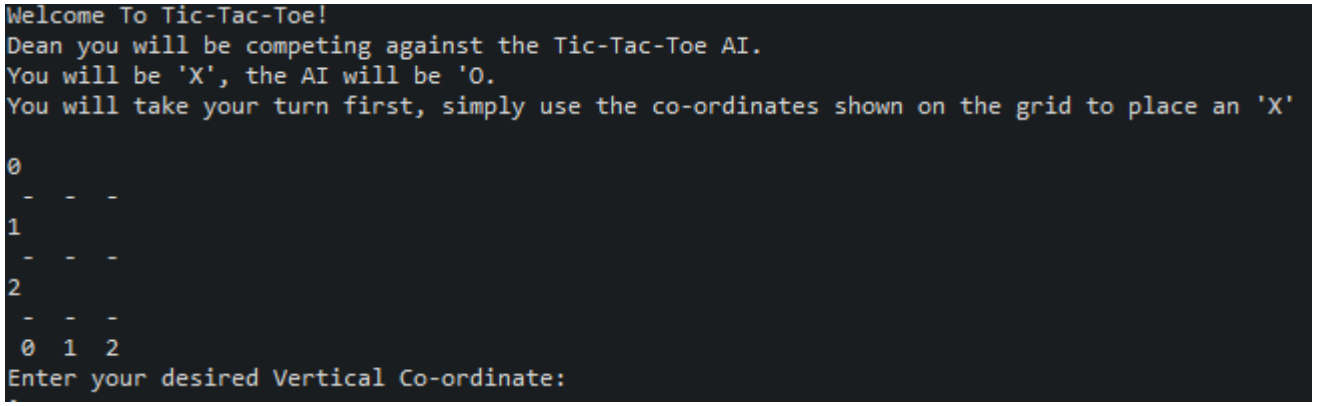
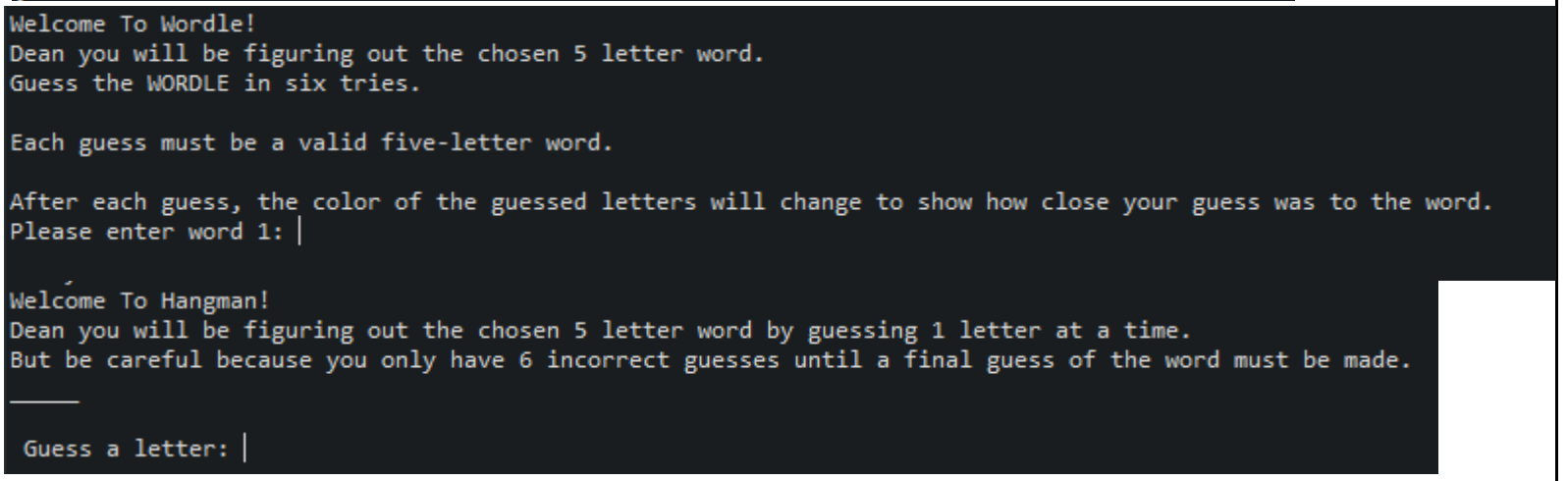
All content within these google documents can be found within this report they are only included here if you find it easier to read (e.g have testing screenshots open as you view the test plan).

D.L: 1-9, 13-15, 17-27, 61-66
J.B: 10-12, 28-33, 35-48
C.N: 49-50, 56-60
S.Mc: 16, 34, 51-55, 73-84

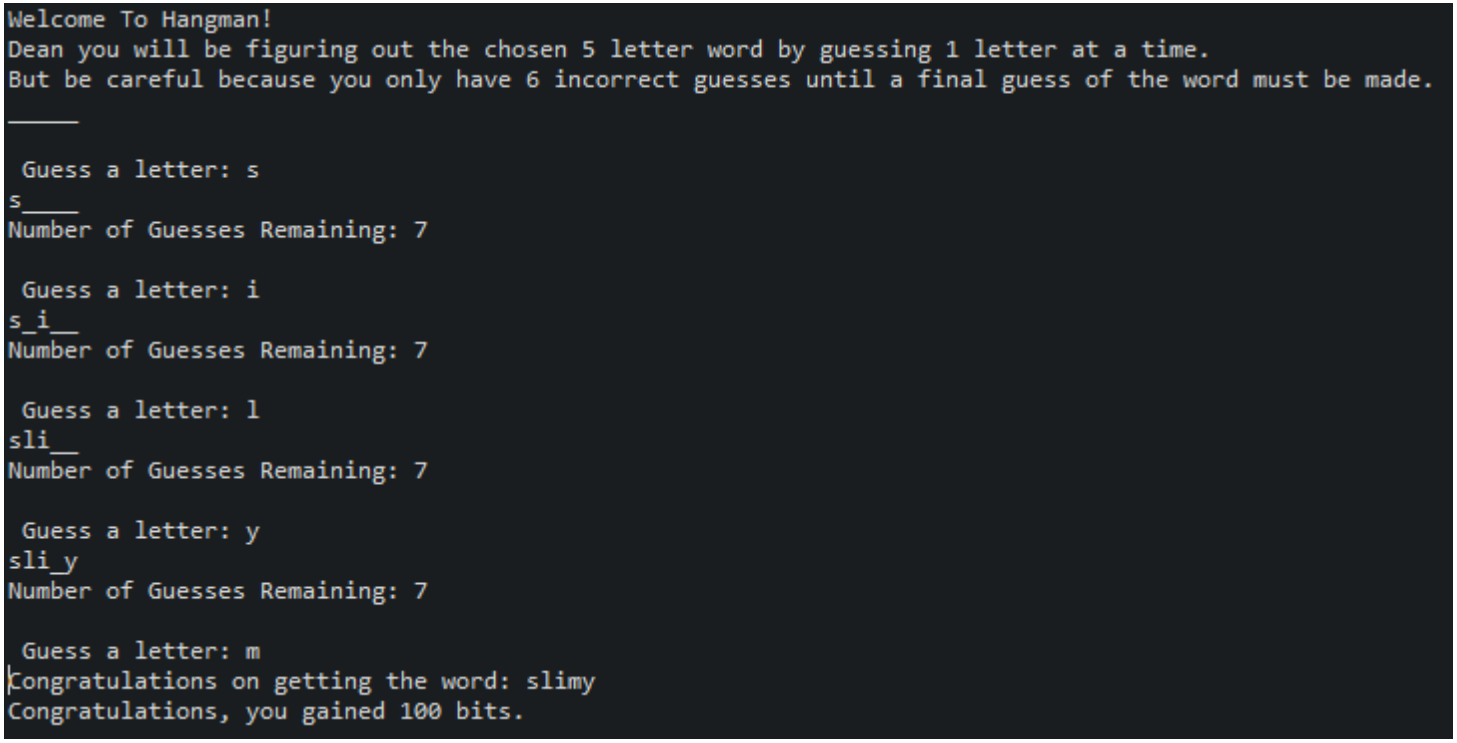
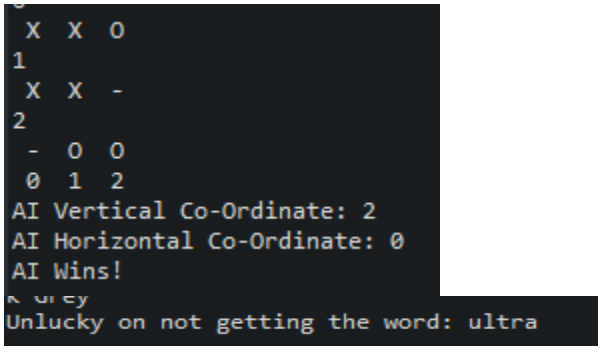
ID	screenshots
1	 Note: the screenshots were taken in order from left to right showing that the severityOfVirus has been incremented.
2	
3	

4	You are currently stuck in Safe Mode for 1. To get out of Safe Mode you can either: 1. Roll a Double 2. Use 1500 bits to get out of Safe Mode Please enter your choice: 42 Please select a valid option from the list by either entering the number or the name of the option you wish to se You are currently stuck in Safe Mode for 1. To get out of Safe Mode you can either: 1. Roll a Double 2. Use 1500 bits to get out of Safe Mode Please enter your choice:
5	You are currently stuck in Safe Mode for 1. To get out of Safe Mode you can either: 1. Roll a Double 2. Use 1500 bits to get out of Safe Mode Please enter your choice: I am serious… and don’t call me Shirley Please select a valid option from the list by either entering the number or the name of the option you wish to se You are currently stuck in Safe Mode for 1. To get out of Safe Mode you can either: 1. Roll a Double 2. Use 1500 bits to get out of Safe Mode Please enter your choice:

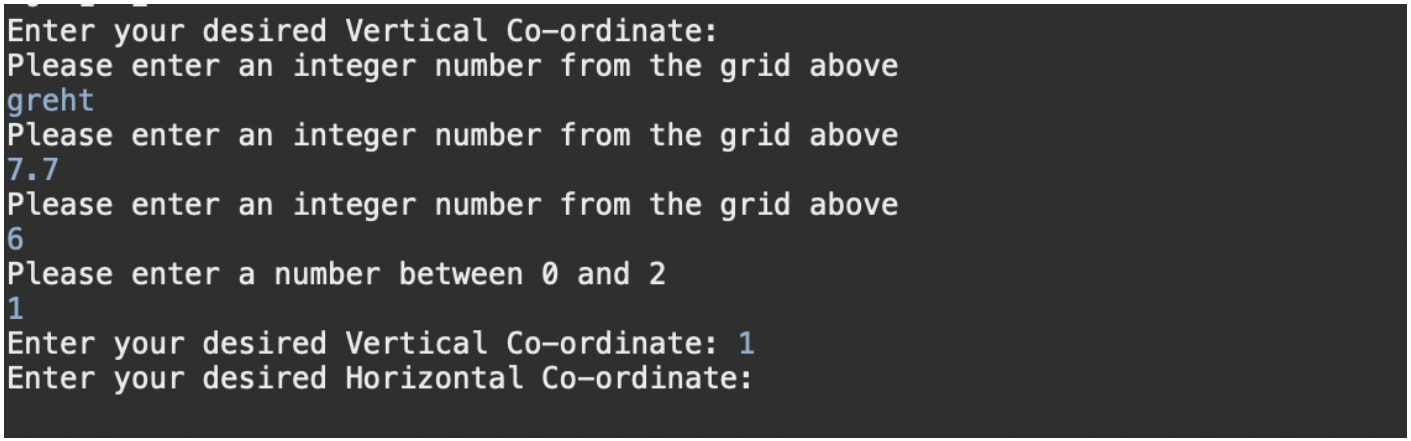
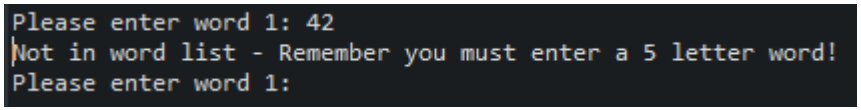
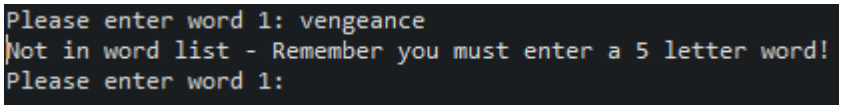
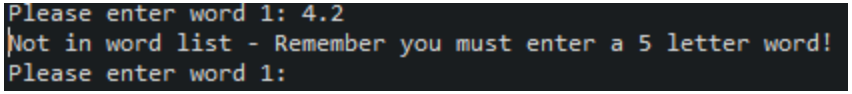
6	
7	Note: Screenshots in the order of Tic-Tact-Toe, Wordle and then Hangman, these screenshots were taken over multiple runs of the game to insure that a random one of these games were selected whenever a user landed on Captcha

	 
8	Note: Screenshots in the order of Tic-Tact-Toe, Wordle and then Hangman

```
0
 X  -  X
1
 O  X  O
2
 O  -  -
 0  1  2
Enter your desired Vertical Co-ordinate:
0
Enter your desired Horizontal Co-ordinate:
1
You Win!
Congratulations, you gained 100 bits.
Welcome To Wordle!
Dean you will be figuring out the chosen 5 letter word.
Guess the WORDLE in six tries.

Each guess must be a valid five-letter word.

After each guess, the color of the guessed letters will change to show how close your guess was to the word.
Please enter word 1: hello
h Grey
e Grey
l Yellow
l Yellow
o Grey
Please enter word 2: ultra
u Green
l Green
t Green
r Green
a Green
CORRECT
```

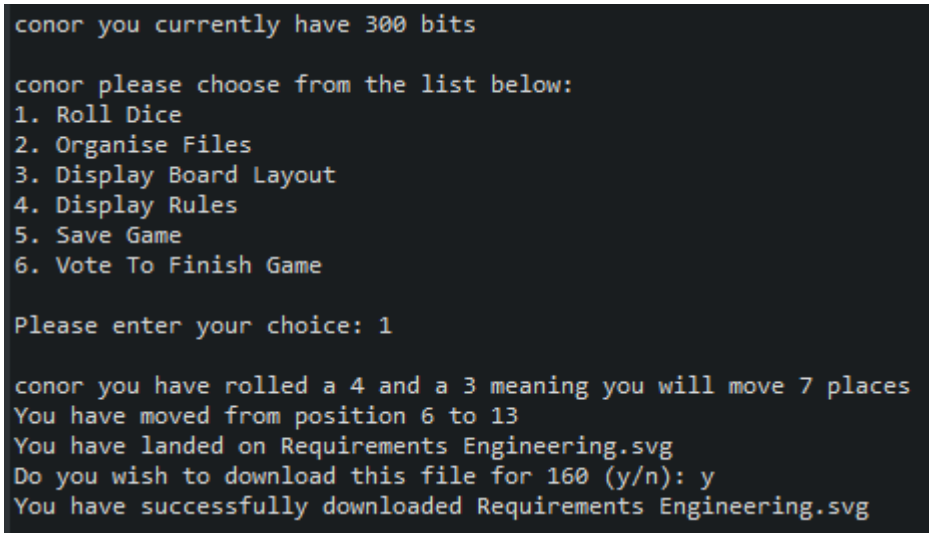
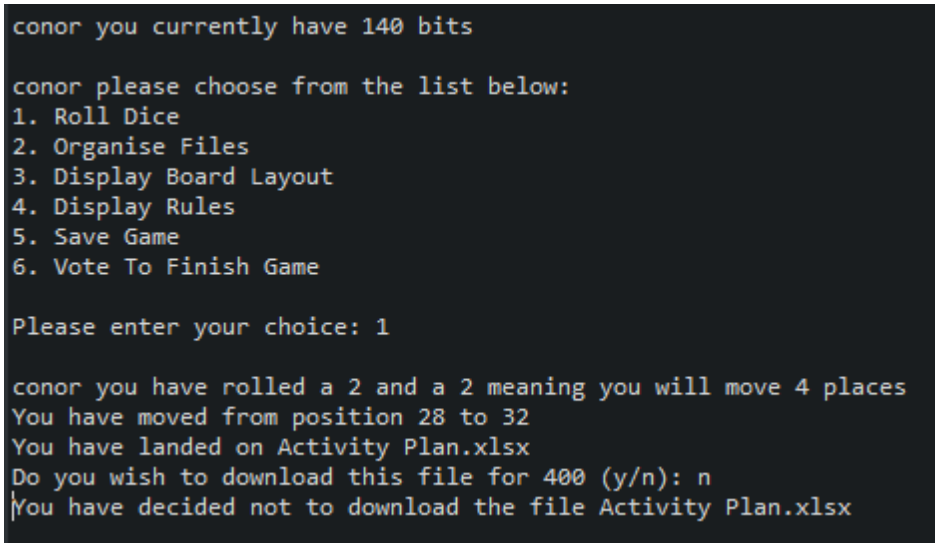
	
9	Note: Screenshots in the order of Tic-Tact-Toe, Wordle and then Hangman 

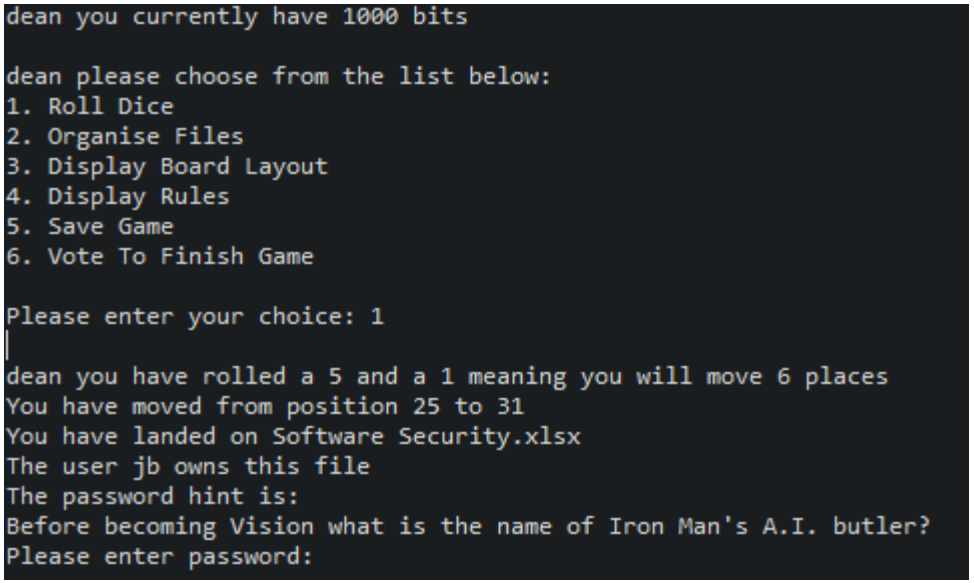
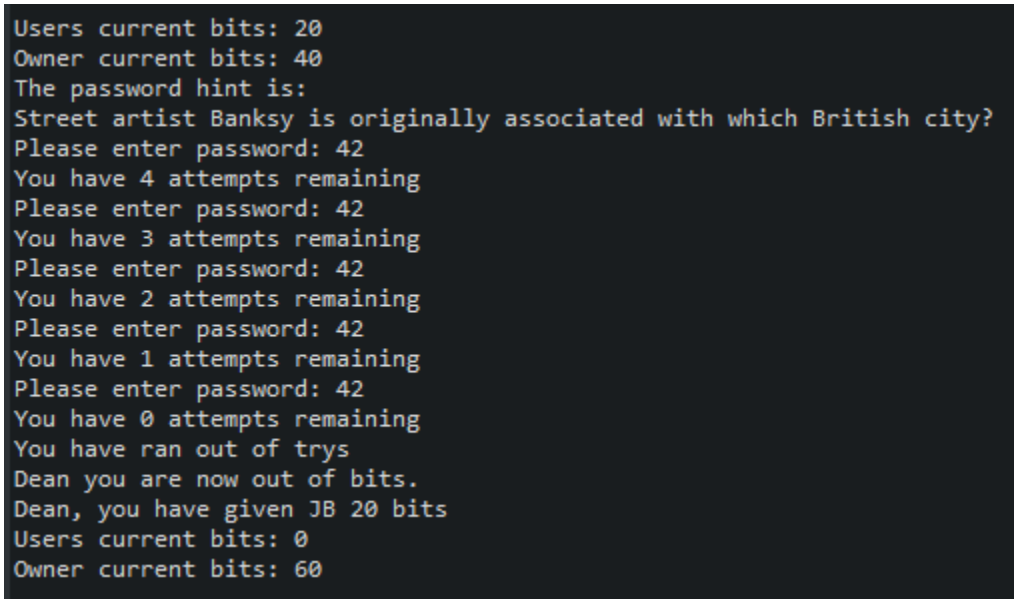
```
Welcome To Hangman!
Dean you will be figuring out the chosen 5 letter word by guessing 1 letter at a time.
But be careful because you only have 6 incorrect guesses until a final guess of the word must be made.
_____

 Guess a letter: k

_____
Number of Guesses Remaining: 6
Incorrect Guesses: k

 Guess a letter: m
m____
Number of Guesses Remaining: 6
Incorrect Guesses: k

 Guess a letter: g
m____
Number of Guesses Remaining: 5
Incorrect Guesses: k g

 Guess a letter: h
m____
Number of Guesses Remaining: 4
Incorrect Guesses: k g h

 Guess a letter: s
m___s
Number of Guesses Remaining: 4
Incorrect Guesses: k g h

 Guess a letter: l
m___s
Number of Guesses Remaining: 3
Incorrect Guesses: k g h l

 Guess a letter: v
m___s
Number of Guesses Remaining: 2
Incorrect Guesses: k g h l v

 Guess the word: n
Not a valid guess.


 Guess the word: maids
m___s
Number of Guesses Remaining: 0
Incorrect Guesses: k g h l v n
Unlucky on not getting the word: mrads
```

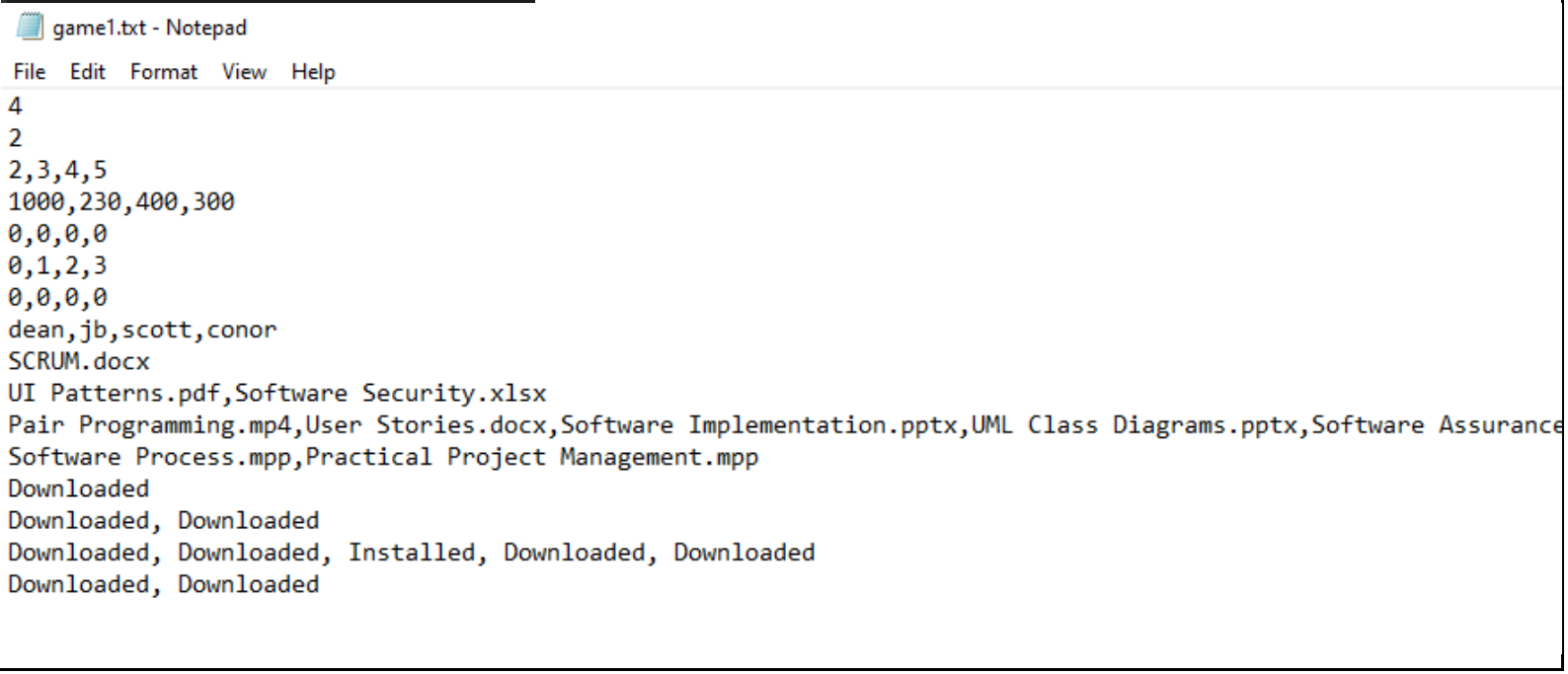
10,11,12	 Note: Screenshot includes the results of test cases with ids of 7, 8 and 9
13	
14	
15	

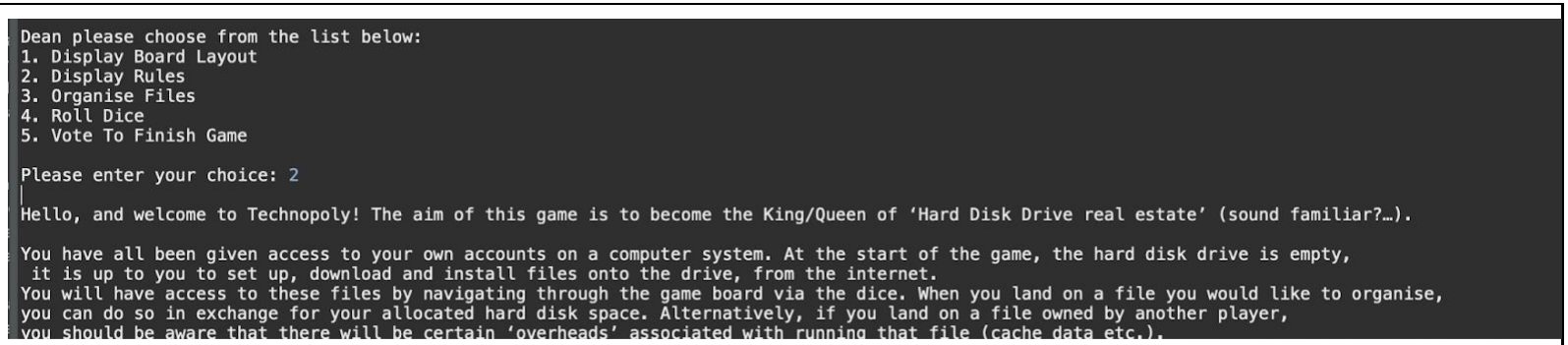
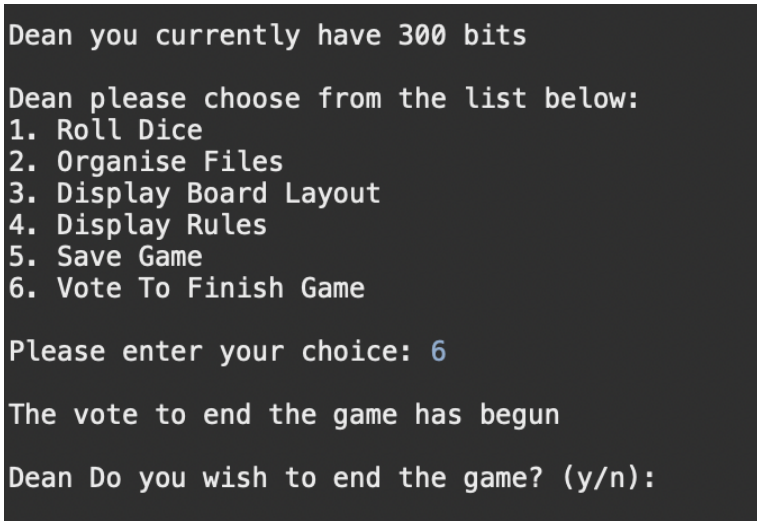
16	Welcome To Hangman! Dean you will be figuring out the chosen 5 letter word by guessing 1 letter at a time. But be careful because you only have 6 incorrect guesses until a final guess of the word must be _____ Guess a letter: Sandy Not a valid guess. Guess a letter:
17	Note: Screenshots taken over multiple runs to ensure a random file was taken away from the user and added to recycling bin Files player owns: Pair Programming.mp4, User Stories.docx, Software Implementation.pptx, UML Class Diagrams.pptx, Software Assu Files in Recycling Bin: No files in bin The file Software Assurance.pdf has been sent to the Recycling Bin. The bits invested will not be reinbursed and the file cannot be bought by landing on it. The only way to retrieve the file is to complete the memory game after landing on the Recycling Bin. Take a note of the name for Software Assurance.pdf as you will need to enter the name in full to retrieve it. Beware however, as any player can play the Memory Game, and any player can take this file. length of player owned files: 0 Files player owns: Pair Programming.mp4, User Stories.docx, Software Implementation.pptx, UML Class Diagrams.pptx, Files in Recycling Bin: Software Assurance.pdf,

	Files player owns: Pair Programming.mp4, User Stories.docx, Software Implementation.pptx, UML Class Diagrams.pptx, Software Ass Files in Recycling Bin: No files in bin The file User Stories.docx has been sent to the Recycling Bin. The bits invested will not be reinbursed and the file cannot be bought by landing on it. The only way to retrieve the file is to complete the memory game after landing on the Recycling Bin. Take a note of the name for User Stories.docx as you will need to enter the name in full to retrieve it. Beware however, as any player can play the Memory Game, and any player can take this file. length of player owned files: 0 Files player owns: Pair Programming.mp4, Software Implementation.pptx, UML Class Diagrams.pptx, Software Assurance.pdf, Files in Recycling Bin: User Stories.docx,
18	Files player owns: Player owns no files Files in Recycling Bin: No files in bin Files player owns: Player owns no files Files in Recycling Bin: No files in bin
19	Files player owns: Files in Recycling Bin: No files in bin You have no files to corrupt length of player owned files: 0 Files player owns: Files in Recycling Bin:

20	 conor you currently have 300 bits conor please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: 1 conor you have rolled a 4 and a 3 meaning you will move 7 places You have moved from position 6 to 13 You have landed on Requirements Engineering.svg Do you wish to download this file for 160 (y/n): y You have successfully downloaded Requirements Engineering.svg
21	 conor you currently have 140 bits conor please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: 1 conor you have rolled a 2 and a 2 meaning you will move 4 places You have moved from position 28 to 32 You have landed on Activity Plan.xlsx Do you wish to download this file for 400 (y/n): n You have decided not to download the file Activity Plan.xlsx

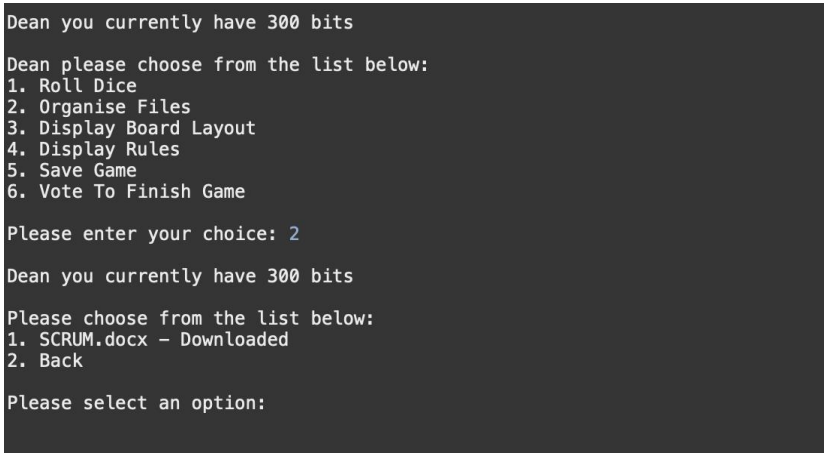
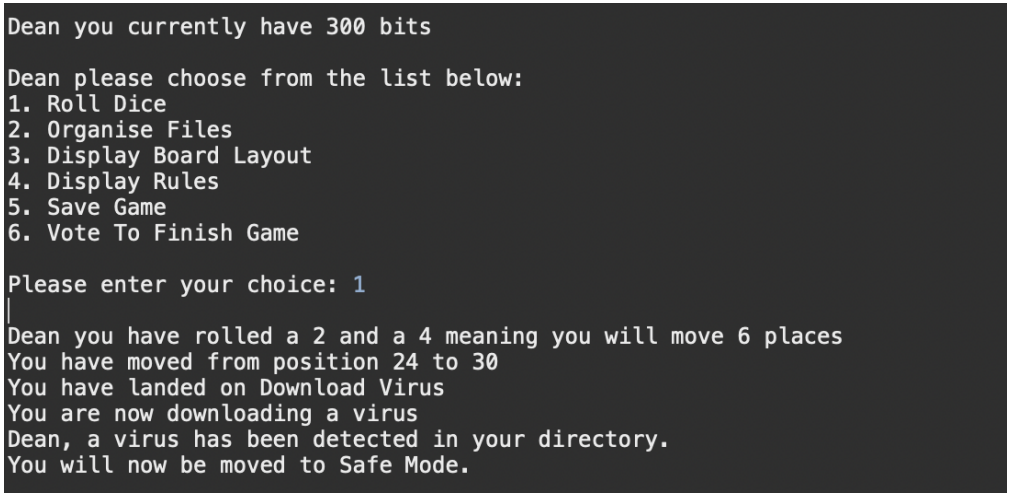
22	 dean you currently have 1000 bits dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: 1 dean you have rolled a 5 and a 1 meaning you will move 6 places You have moved from position 25 to 31 You have landed on Software Security.xlsx The user jb owns this file The password hint is: Before becoming Vision what is the name of Iron Man's A.I. butler? Please enter password:
23	 Users current bits: 20 Owner current bits: 40 The password hint is: Street artist Banksy is originally associated with which British city? Please enter password: 42 You have 4 attempts remaining Please enter password: 42 You have 3 attempts remaining Please enter password: 42 You have 2 attempts remaining Please enter password: 42 You have 1 attempts remaining Please enter password: 42 You have 0 attempts remaining You have ran out of trys Dean you are now out of bits. Dean, you have given JB 20 bits Users current bits: 0 Owner current bits: 60

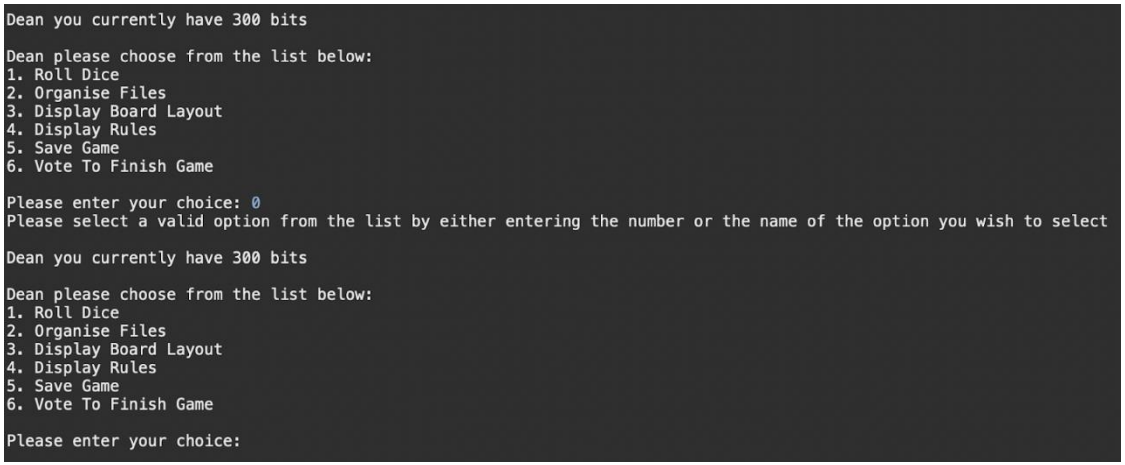
24	Do you wish to download this file for 100 (y/n): 42 Please select either yes or no Do you wish to download this file for 100 (y/n):
25	Do you wish to download this file for 100 (y/n): Take your stinking paws off me, you damned dirty ape Please select either yes or no Do you wish to download this file for 100 (y/n):
26	Do you wish to download this file for 100 (y/n): 4.2 Please select either yes or no Do you wish to download this file for 100 (y/n):
27	Note: First screenshot is taken from the console output and the second is of the text file that has been created The game has been successfully saved 

28	 Dean please choose from the list below: 1. Display Board Layout 2. Display Rules 3. Organise Files 4. Roll Dice 5. Vote To Finish Game Please enter your choice: 2 Hello, and welcome to Technopoly! The aim of this game is to become the King/Queen of 'Hard Disk Drive real estate' (sound familiar?…). You have all been given access to your own accounts on a computer system. At the start of the game, the hard disk drive is empty, it is up to you to set up, download and install files onto the drive, from the internet. You will have access to these files by navigating through the game board via the dice. When you land on a file you would like to organise, you can do so in exchange for your allocated hard disk space. Alternatively, if you land on a file owned by another player, you should be aware that there will be certain 'overheads' associated with running that file (cache data etc.).
29	 Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: 6 The vote to end the game has begun Dean Do you wish to end the game? (y/n):

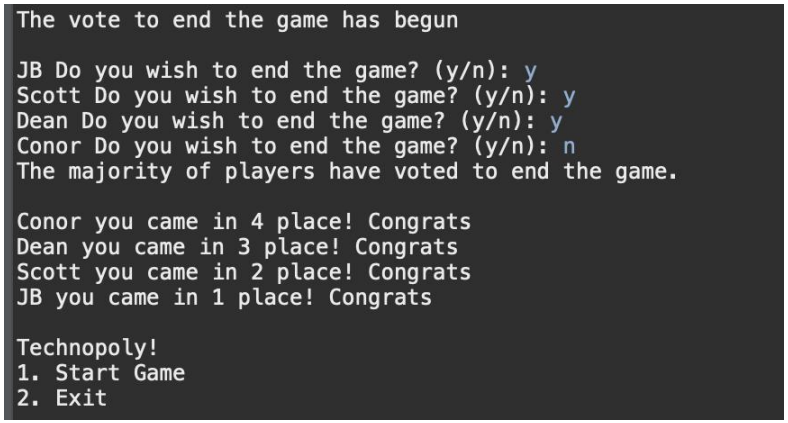
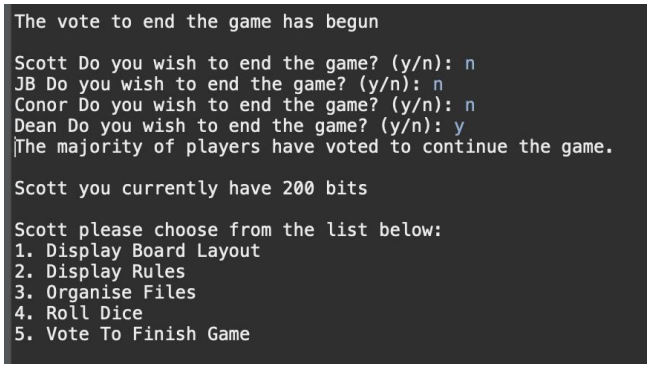
30

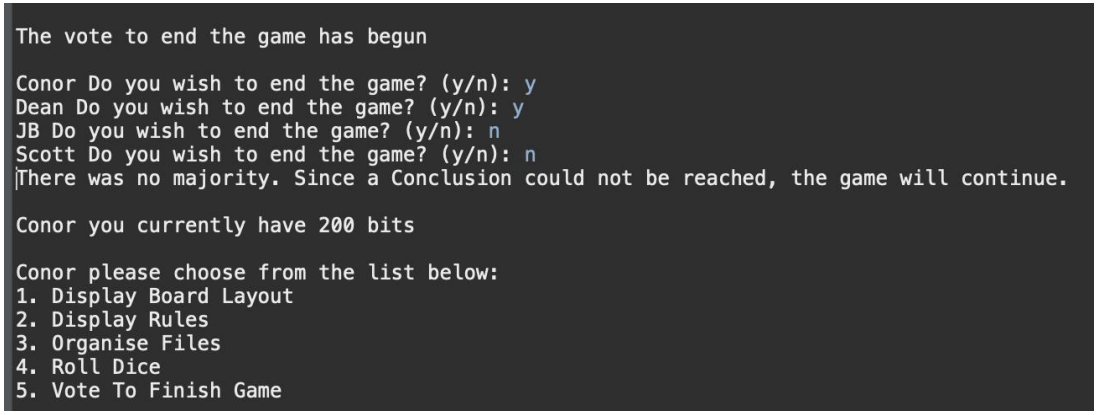
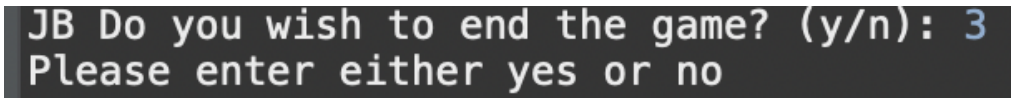
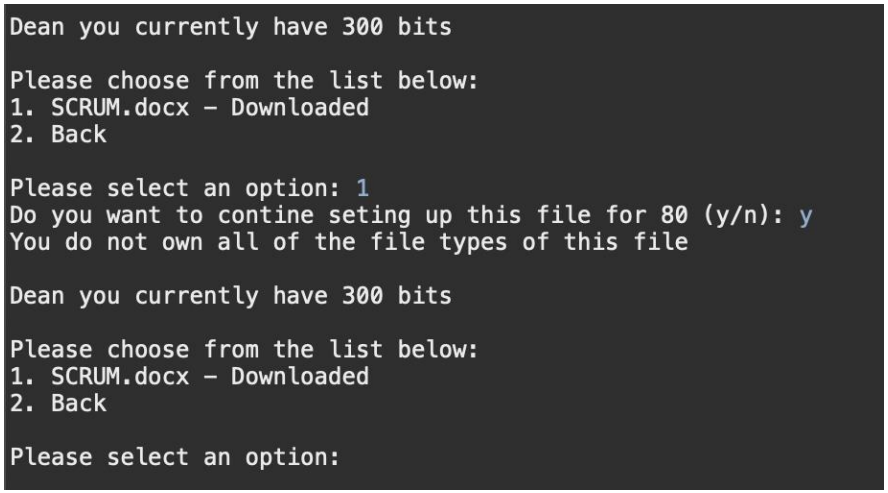
```
Dean please choose from the list below:
1. Roll Dice
2. Organise Files
3. Display Board Layout
4. Display Rules
5. Save Game
6. Vote To Finish Game

Please enter your choice: 3

Board Layot
-----------
 Position | Location                                | Player
-------------------------------------------------------------------
 0        | Go                                      |
 1        | Software Process.mpp                    |
 2        | Corrupt File                            |
 3        | Practical Project Management.mpp        | JB
 4        | Utility                                 |
 5        | File Explorer 1                         |
 6        | SCRUM.docx                              |
 7        | Captcha                                 |
 8        | User Stories.docx                       | Conor
 9        | Use Cases and Stories.docx              |
 10       | Safe Mode                               |
 11       | Software Verifcation.svg                |
 12       | Utility                                 |
 13       | Requirements Engineering.svg            |
 14       | Requirements Engineering Analysis.svg   |
 15       | File Explorer 2                         |
 16       | UML Class Diagrams.pptx                 |
 17       | Corrupt File                            |
 18       | Software Implementation.pptx            |
 19       | Design Dos and Don'ts.pptx              |
 20       | Recycle Bin                             |
 21       | UI Patterns.pdf                         |
 22       | Captcha                                 |
 23       | Software Assurance.pdf                  |
 24       | Design Quality.pdf                      | Dean
 25       | File Explorer 3                         |
 26       | Pair Programming.mp4                    |
 27       | Sprint Retrospective.mp4                |
 28       | Utility                                 |
 29       | Agile Manifesto.mp4                     |
 30       | Download Virus                          |
 31       | Software Security.xlsx                  |
 32       | Activity Plan.xlsx                      |
 33       | Corrupt File                            |
 34       | Gantt Chart.xlsx                        | Scott
 35       | File Explorer 4                         |
 36       | Captcha                                 |
 37       | Security Information.one                |
 38       | Best Friend.jpg                         |
 39       | Network Analysis.one                    |
```

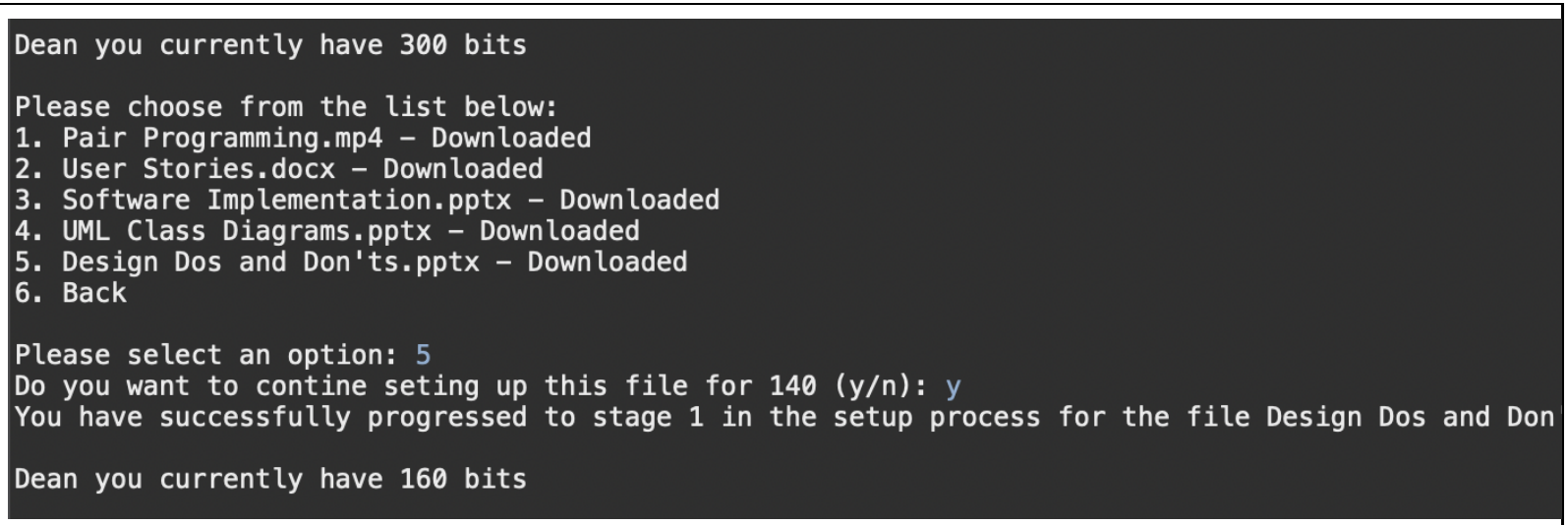
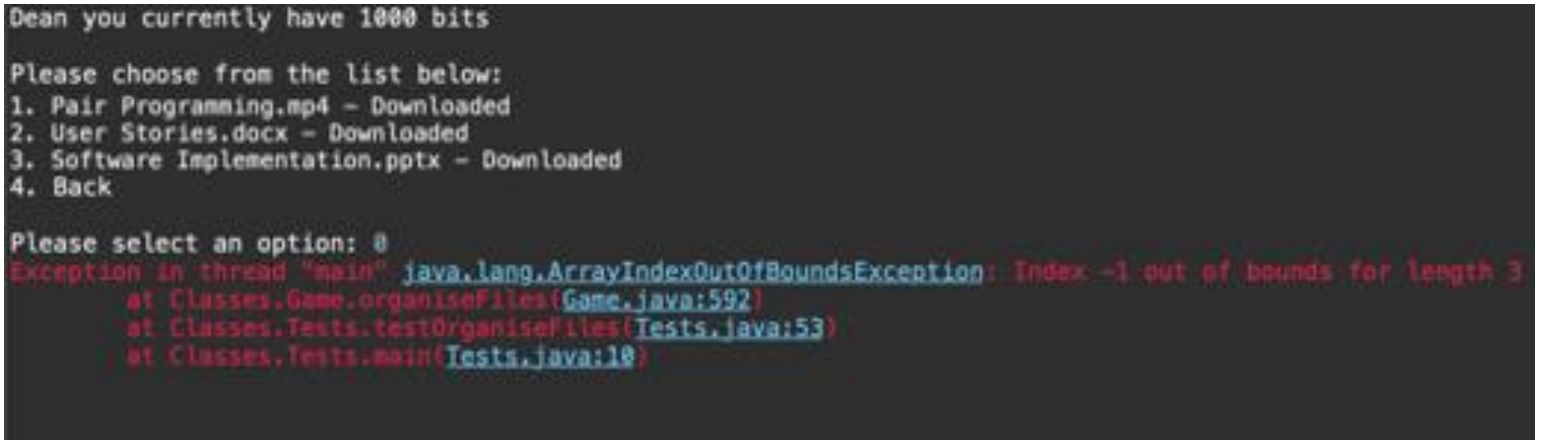
31	 Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: 2 Dean you currently have 300 bits Please choose from the list below: 1. SCRUM.docx - Downloaded 2. Back Please select an option:
32	 Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: 1 Dean you have rolled a 2 and a 4 meaning you will move 6 places You have moved from position 24 to 30 You have landed on Download Virus You are now downloading a virus Dean, a virus has been detected in your directory. You will now be moved to Safe Mode.

33	 Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: 0 Please select a valid option from the list by either entering the number or the name of the option you wish to select Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice:

34	The following are tests Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: Spongebob Please select a valid option from the list by either entering the number or the name of the option you wish to s Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice:

35	 The vote to end the game has begun JB Do you wish to end the game? (y/n): y Scott Do you wish to end the game? (y/n): y Dean Do you wish to end the game? (y/n): y Conor Do you wish to end the game? (y/n): n The majority of players have voted to end the game. Conor you came in 4 place! Congrats Dean you came in 3 place! Congrats Scott you came in 2 place! Congrats JB you came in 1 place! Congrats Technopoly! 1. Start Game 2. Exit
36	 The vote to end the game has begun Scott Do you wish to end the game? (y/n): n JB Do you wish to end the game? (y/n): n Conor Do you wish to end the game? (y/n): n Dean Do you wish to end the game? (y/n): y The majority of players have voted to continue the game. Scott you currently have 200 bits Scott please choose from the list below: 1. Display Board Layout 2. Display Rules 3. Organise Files 4. Roll Dice 5. Vote To Finish Game
37	

	 The vote to end the game has begun Conor Do you wish to end the game? (y/n): y Dean Do you wish to end the game? (y/n): y JB Do you wish to end the game? (y/n): n Scott Do you wish to end the game? (y/n): n There was no majority. Since a Conclusion could not be reached, the game will continue. Conor you currently have 200 bits Conor please choose from the list below: 1. Display Board Layout 2. Display Rules 3. Organise Files 4. Roll Dice 5. Vote To Finish Game
38	 JB Do you wish to end the game? (y/n): 3 Please enter either yes or no
39	 Dean you currently have 300 bits Please choose from the list below: 1. SCRUM.docx - Downloaded 2. Back Please select an option: 1 Do you want to contine seting up this file for 80 (y/n): y You do not own all of the file types of this file Dean you currently have 300 bits Please choose from the list below: 1. SCRUM.docx - Downloaded 2. Back Please select an option:

40	 Dean you currently have 300 bits Please choose from the list below: 1. Pair Programming.mp4 - Downloaded 2. User Stories.docx - Downloaded 3. Software Implementation.pptx - Downloaded 4. UML Class Diagrams.pptx - Downloaded 5. Design Dos and Don'ts.pptx - Downloaded 6. Back Please select an option: 5 Do you want to contine seting up this file for 140 (y/n): y You have successfully progressed to stage 1 in the setup process for the file Design Dos and Don Dean you currently have 160 bits
41	 Dean you currently have 1000 bits Please choose from the list below: 1. Pair Programming.mp4 - Downloaded 2. User Stories.docx - Downloaded 3. Software Implementation.pptx - Downloaded 4. Back Please select an option: 0 Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index -1 out of bounds for length 3 at Classes.Game.organiseFiles(Game.java:592) at Classes.Tests.testOrganiseFiles(Tests.java:53) at Classes.Tests.main(Tests.java:10)

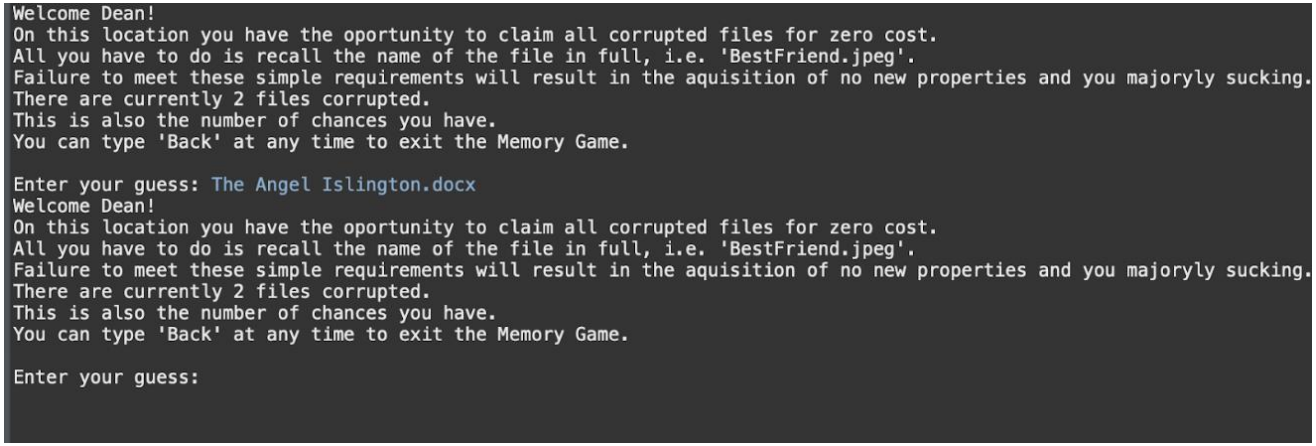
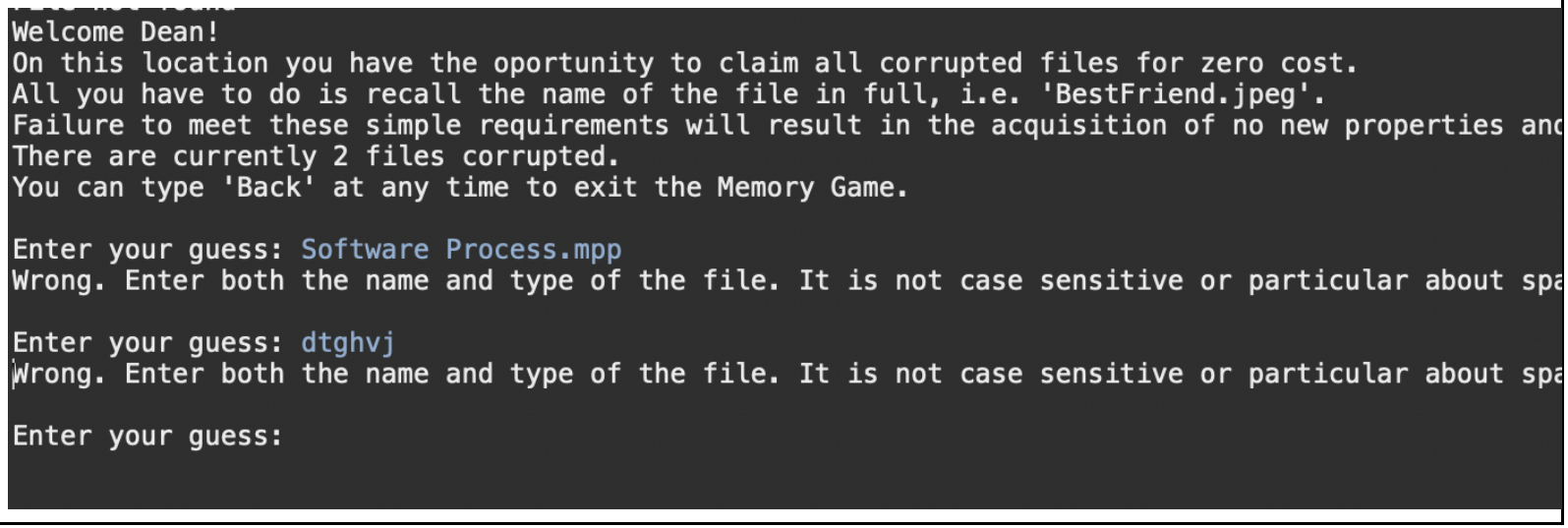
42

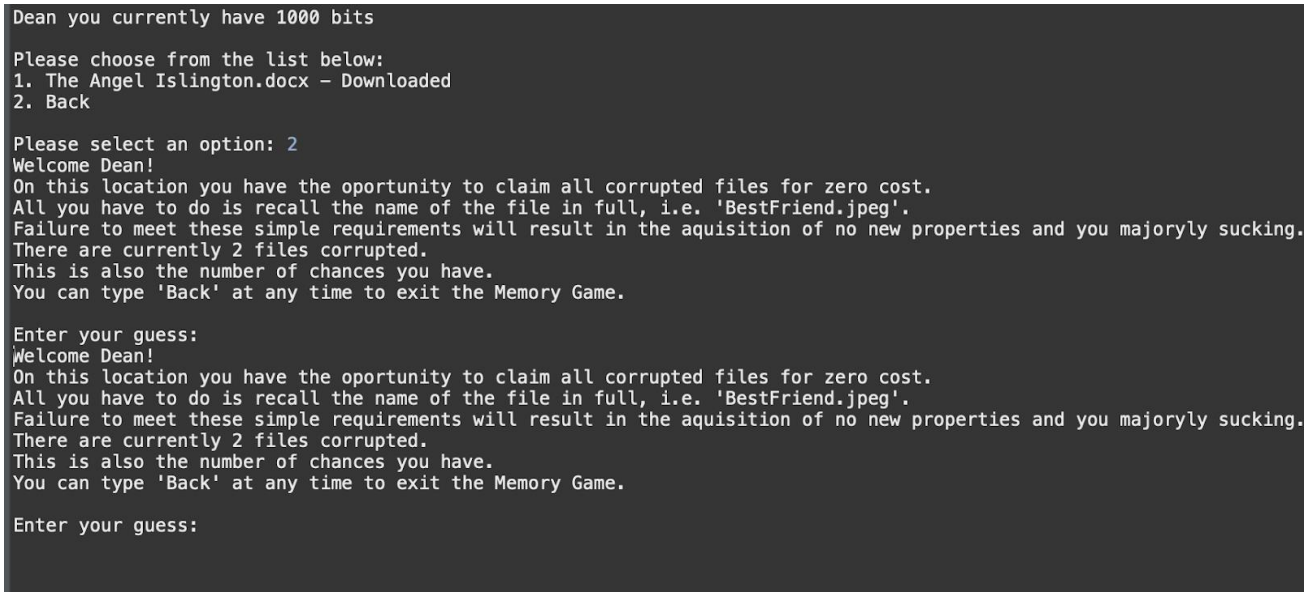
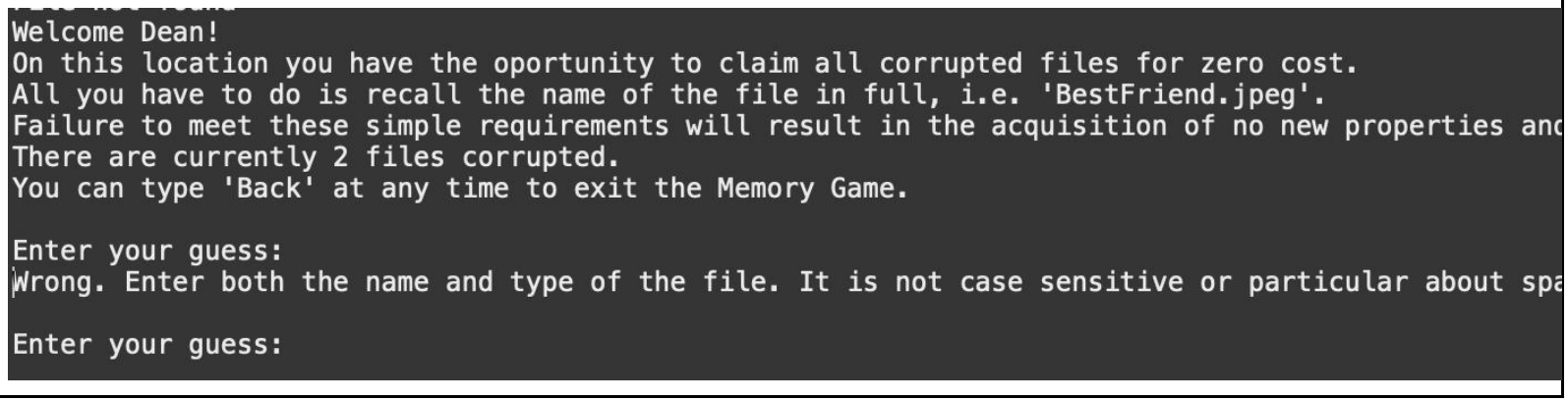
```
Please choose from the list below:
1. Pair Programming.mp4 - Downloaded
2. Back

Please select an option: 0
Pleas select one of the files above

Dean you currently have 1000 bits

Please choose from the list below:
1. Pair Programming.mp4 - Downloaded
2. Back
```

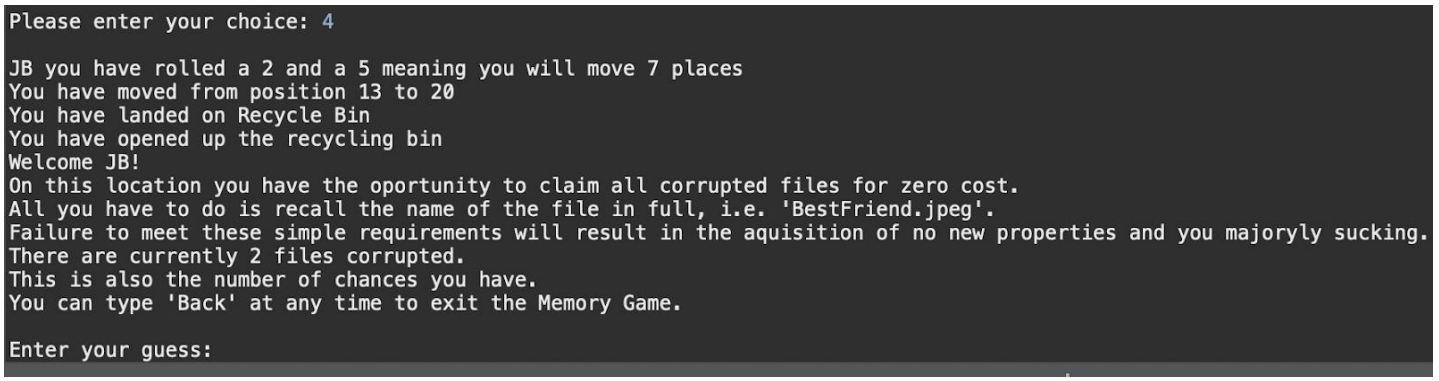
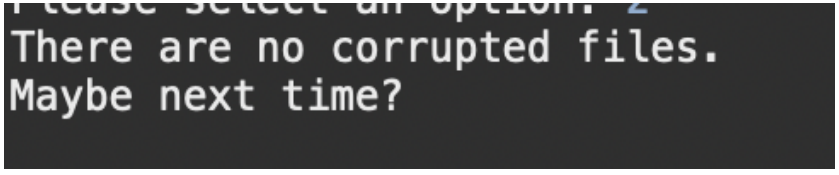
```
if(optionNum <= filesOwnedLen && optionNum > 0) {
```

43

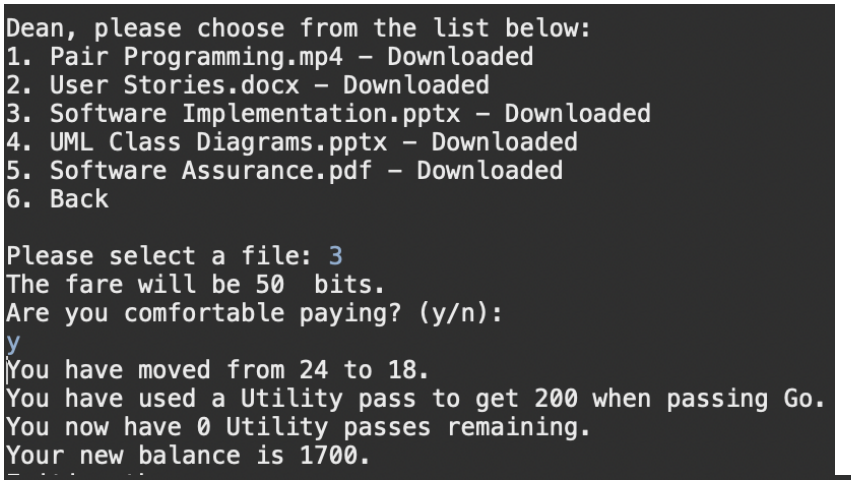
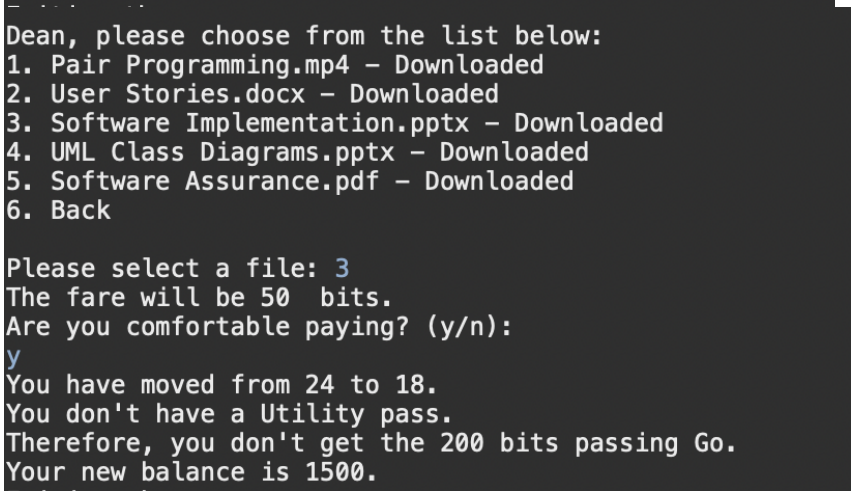
```
Welcome Dean!
On this location you have the oportunity to claim all corrupted files for zero cost.
All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'.
Failure to meet these simple requirements will result in the acquisition of no new properties and you majoryl
There are currently 2 files corrupted.
You can type 'Back' at any time to exit the Memory Game.

Enter your guess: Use Cases and Stories.docx
Congratulations! This file is worth 80  bits.
Are you happy to accept this file? (y/n):
y
You have successfully downloaded this file.


Dean you currently have 1600 bits

Please choose from the list below:
1. SCRUM.docx - Downloaded
2. Use Cases and Stories.docx - Downloaded
3. Back
```

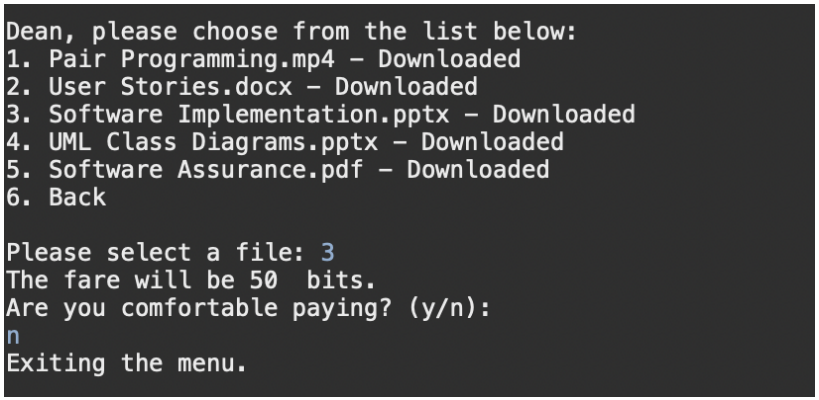
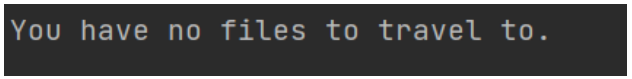
44	 Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the aquisition of no new properties and you majoryly sucking. There are currently 2 files corrupted. This is also the number of chances you have. You can type 'Back' at any time to exit the Memory Game. Enter your guess: The Angel Islington.docx Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the aquisition of no new properties and you majoryly sucking. There are currently 2 files corrupted. This is also the number of chances you have. You can type 'Back' at any time to exit the Memory Game. Enter your guess:
45	 Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the acquisition of no new properties and There are currently 2 files corrupted. You can type 'Back' at any time to exit the Memory Game. Enter your guess: Software Process.mpp Wrong. Enter both the name and type of the file. It is not case sensitive or particular about spa Enter your guess: dtghvj Wrong. Enter both the name and type of the file. It is not case sensitive or particular about spa Enter your guess:

46	 Dean you currently have 1000 bits Please choose from the list below: 1. The Angel Islington.docx - Downloaded 2. Back Please select an option: 2 Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the aquisition of no new properties and you majoryly sucking. There are currently 2 files corrupted. This is also the number of chances you have. You can type 'Back' at any time to exit the Memory Game. Enter your guess: Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the aquisition of no new properties and you majoryly sucking. There are currently 2 files corrupted. This is also the number of chances you have. You can type 'Back' at any time to exit the Memory Game. Enter your guess:
47	 Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the acquisition of no new properties and There are currently 2 files corrupted. You can type 'Back' at any time to exit the Memory Game. Enter your guess: Wrong. Enter both the name and type of the file. It is not case sensitive or particular about spa Enter your guess:

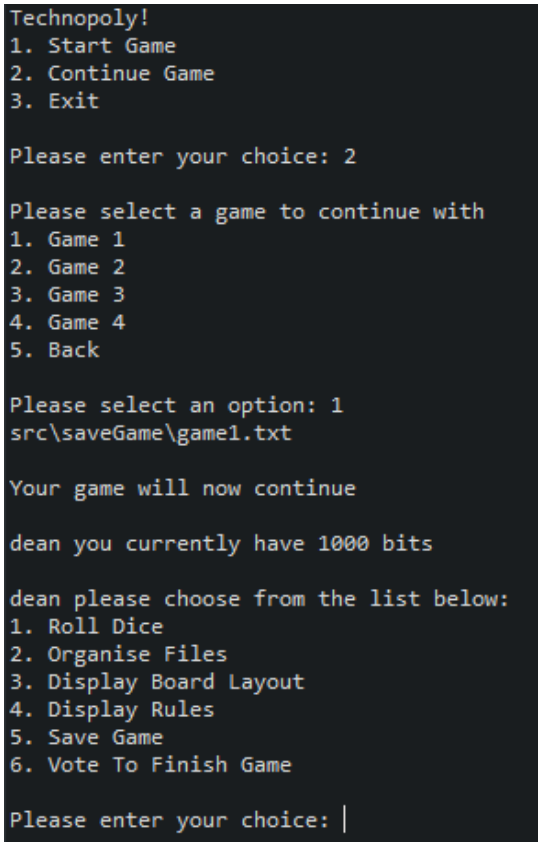
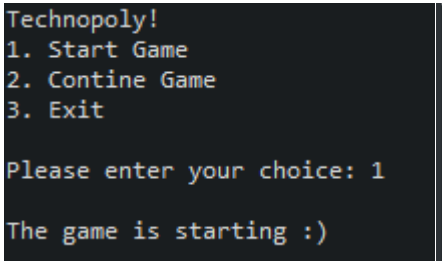
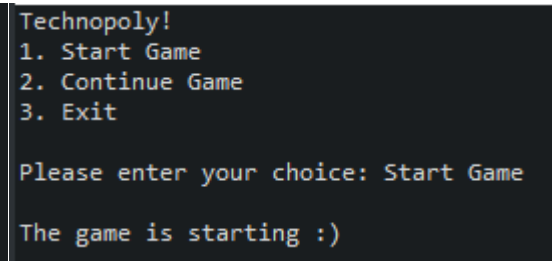
48	 Please select an option: 2 Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the aquisition of no new properties and you majoryly sucking. There are currently 2 files corrupted. This is also the number of chances you have. You can type 'Back' at any time to exit the Memory Game. Enter your guess: Back Exiting Game. Dean you currently have 1000 bits Please choose from the list below: 1. The Angel Islington.docx - Downloaded 2. Back Please select an option:
49	 Please enter your choice: 4 JB you have rolled a 2 and a 5 meaning you will move 7 places You have moved from position 13 to 20 You have landed on Recycle Bin You have opened up the recycling bin Welcome JB! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the aquisition of no new properties and you majoryly sucking. There are currently 2 files corrupted. This is also the number of chances you have. You can type 'Back' at any time to exit the Memory Game. Enter your guess:
50	 There are no corrupted files. Maybe next time?

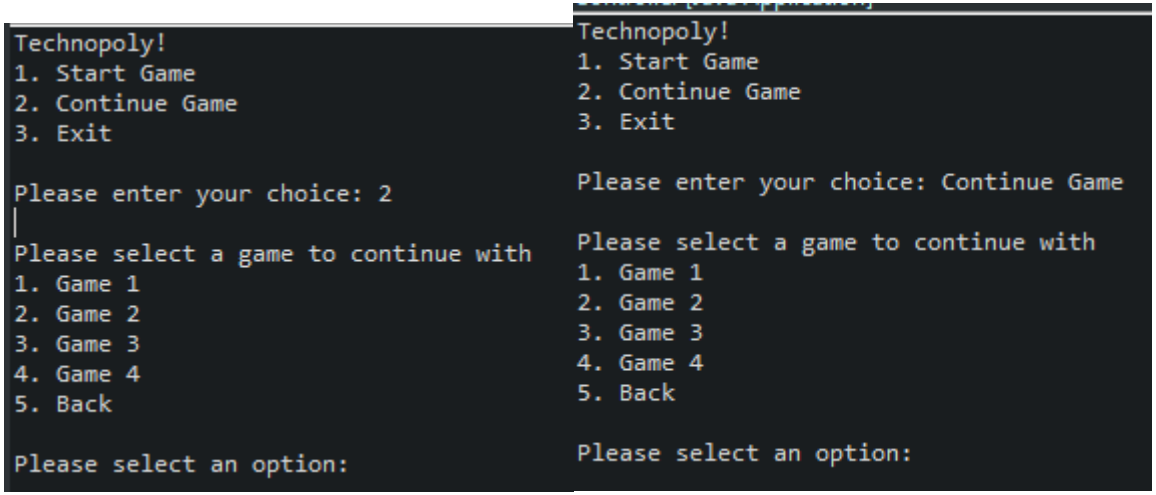
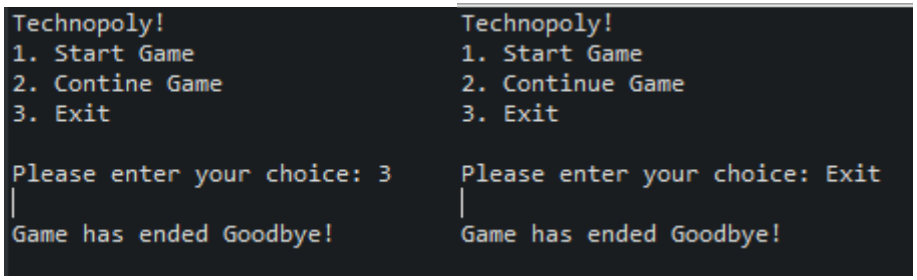
51	The following are tests Welcome Dean! On this location you have the oportunity to claim all corrupted files for zero cost. All you have to do is recall the name of the file in full, i.e. 'BestFriend.jpeg'. Failure to meet these simple requirements will result in the acquisition of no new properties and you majoryly suc There are currently 2 files corrupted. You can type 'Back' at any time to exit the Memory Game. Enter your guess: Mr Krabs Wrong. Enter both the name and type of the file. It is not case sensitive or particular about spaces. Enter your guess:

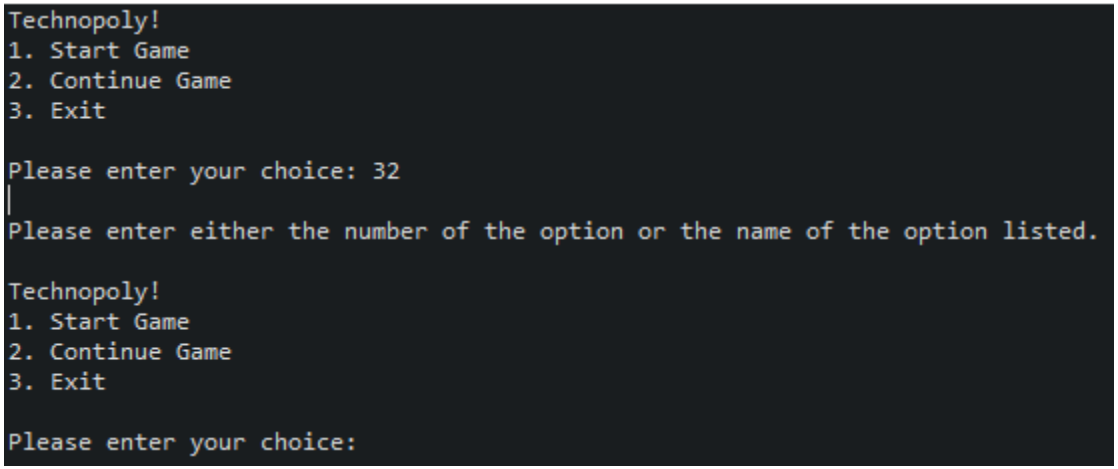
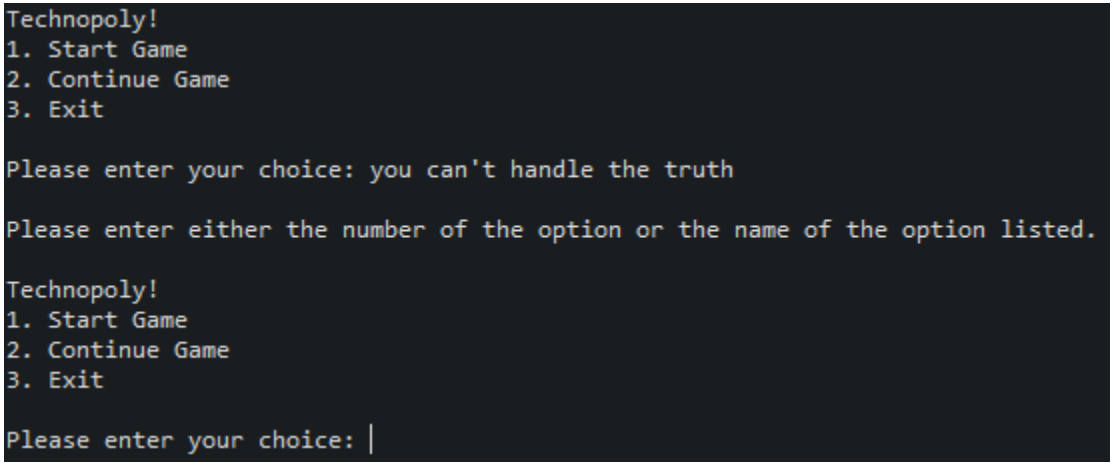
52	 Dean, please choose from the list below: 1. Pair Programming.mp4 - Downloaded 2. User Stories.docx - Downloaded 3. Software Implementation.pptx - Downloaded 4. UML Class Diagrams.pptx - Downloaded 5. Software Assurance.pdf - Downloaded 6. Back Please select a file: 3 The fare will be 50 bits. Are you comfortable paying? (y/n): y You have moved from 24 to 18. You have used a Utility pass to get 200 when passing Go. You now have 0 Utility passes remaining. Your new balance is 1700.  Dean, please choose from the list below: 1. Pair Programming.mp4 - Downloaded 2. User Stories.docx - Downloaded 3. Software Implementation.pptx - Downloaded 4. UML Class Diagrams.pptx - Downloaded 5. Software Assurance.pdf - Downloaded 6. Back Please select a file: 3 The fare will be 50 bits. Are you comfortable paying? (y/n): y You have moved from 24 to 18. You don't have a Utility pass. Therefore, you don't get the 200 bits passing Go. Your new balance is 1500.

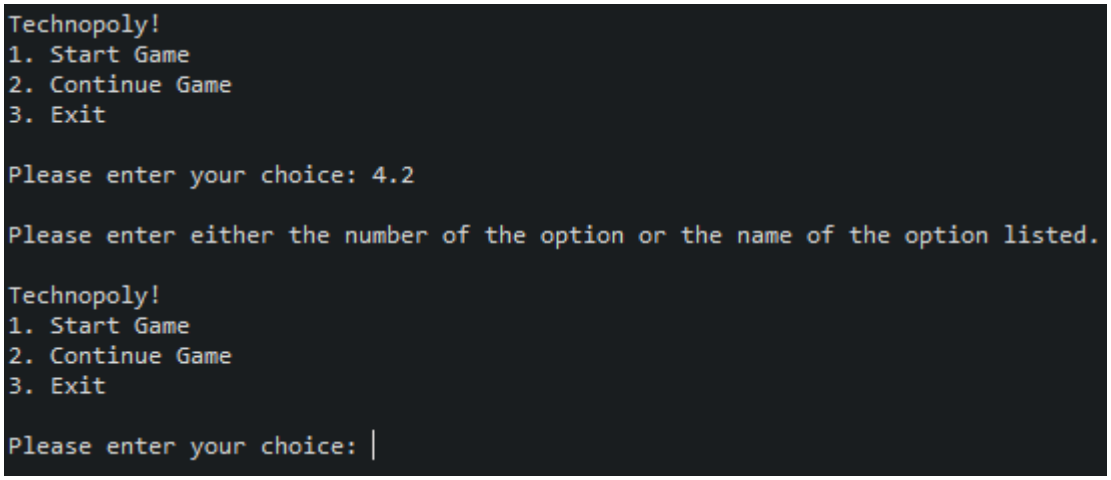
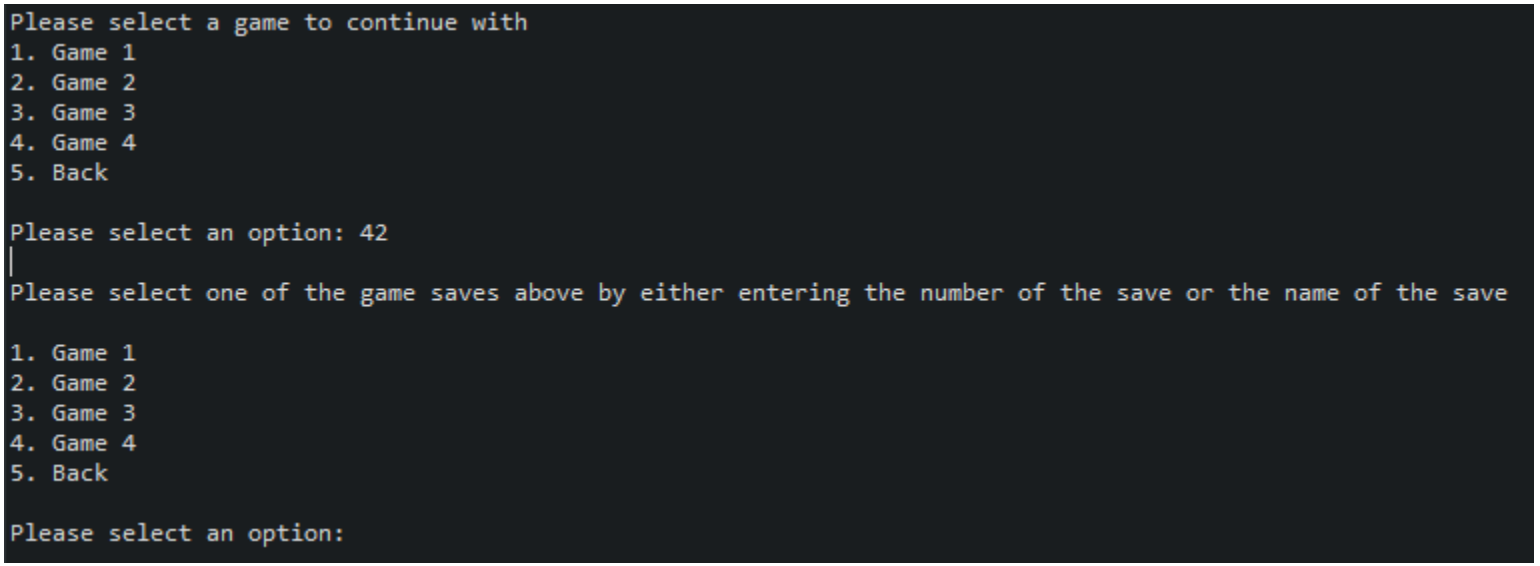
53	 Dean, please choose from the list below: 1. Pair Programming.mp4 - Downloaded 2. User Stories.docx - Downloaded 3. Software Implementation.pptx - Downloaded 4. UML Class Diagrams.pptx - Downloaded 5. Software Assurance.pdf - Downloaded 6. Back Please select a file: 3 The fare will be 50 bits. Are you comfortable paying? (y/n): n Exiting the menu.
54	 You have no files to travel to.

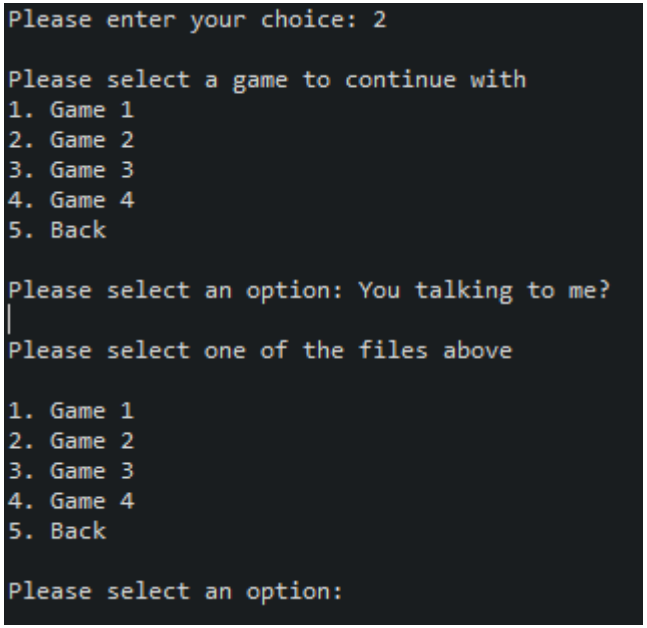
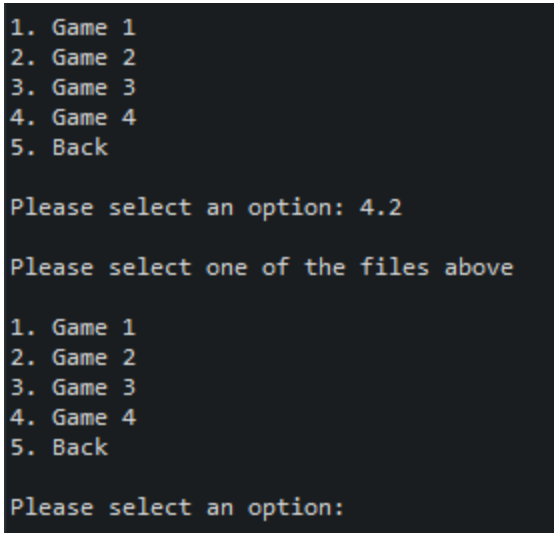
55	The following are tests Dean, please choose from the list below: 1. SCRUM.docx - Downloaded 2. Back Please select a file: Patrick Please select one of the files above Please enter a corresponding number of the full name and type of the location. Dean, please choose from the list below: 1. SCRUM.docx - Downloaded 2. Back Please select a file:
56	There are no saved games

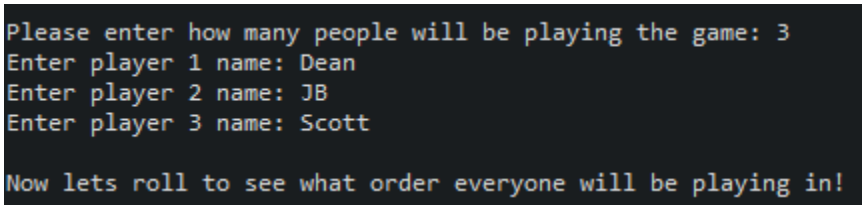
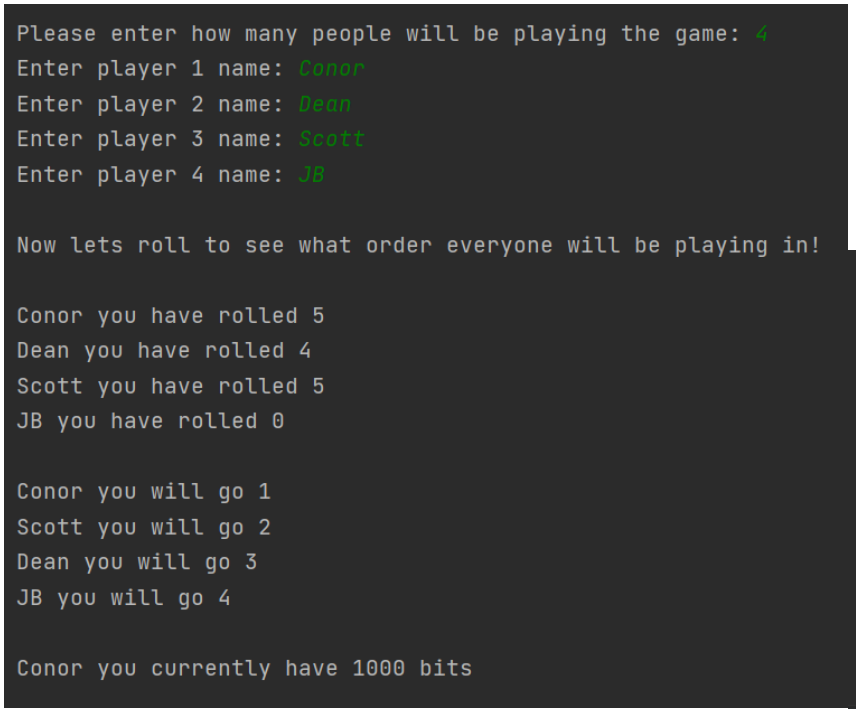
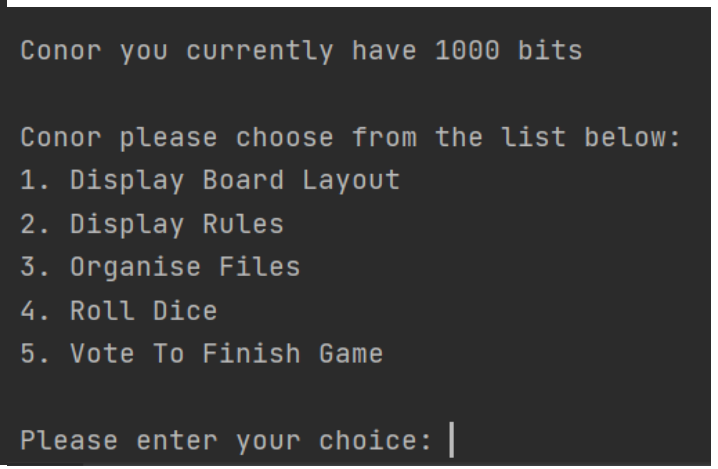
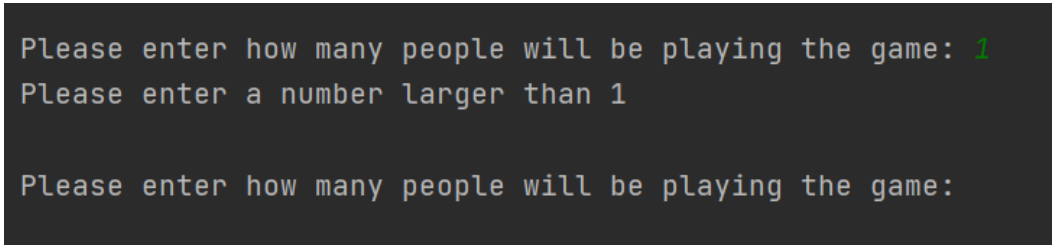
57	 Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: 2 Please select a game to continue with 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option: 1 src\saveGame\game1.txt Your game will now continue dean you currently have 1000 bits dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice:
58	 Technopoly! 1. Start Game 2. Contine Game 3. Exit Please enter your choice: 1 The game is starting :)  Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: Start Game The game is starting :)

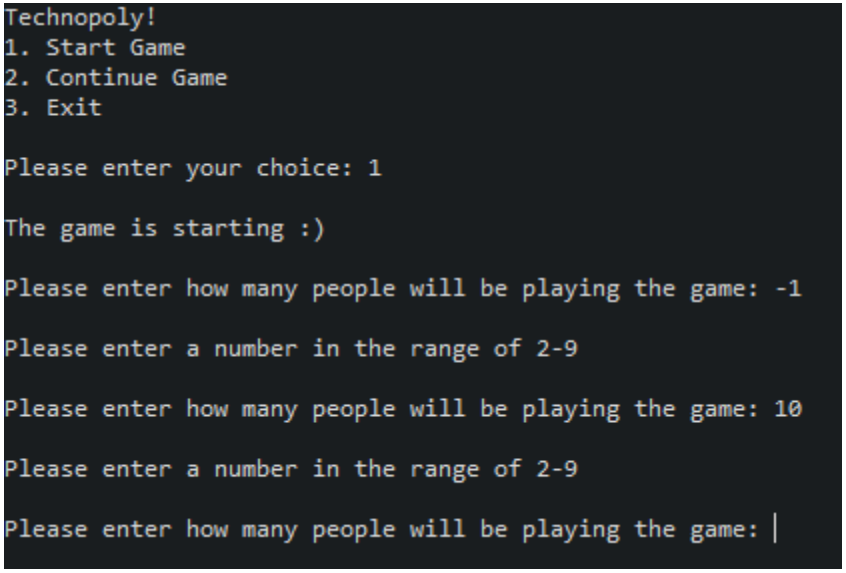
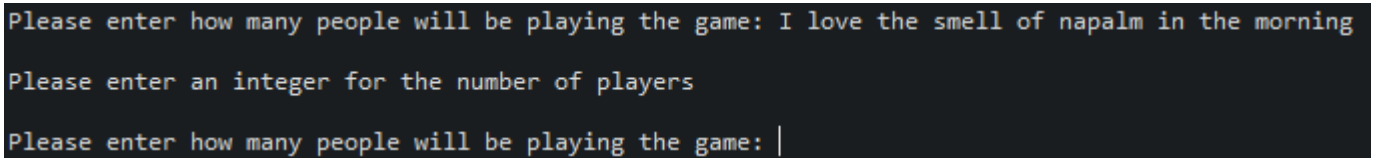
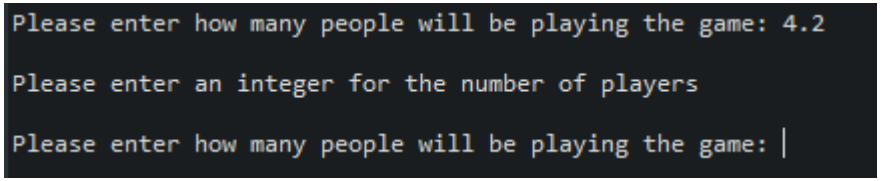
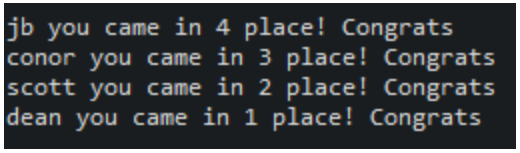
59	 Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: 2 Please select a game to continue with 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option: Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: Continue Game Please select a game to continue with 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option:
60	 Technopoly! 1. Start Game 2. Contine Game 3. Exit Please enter your choice: 3 Game has ended Goodbye! Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: Exit Game has ended Goodbye!

61	 Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: 32 Please enter either the number of the option or the name of the option listed. Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice:
62	 Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: you can't handle the truth Please enter either the number of the option or the name of the option listed. Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice:

63	 Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: 4.2 Please enter either the number of the option or the name of the option listed. Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice:
64	 Please select a game to continue with 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option: 42 Please select one of the game saves above by either entering the number of the save or the name of the save 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option:

65	 Please enter your choice: 2 Please select a game to continue with 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option: You talking to me? Please select one of the files above 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option:
66	 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option: 4.2 Please select one of the files above 1. Game 1 2. Game 2 3. Game 3 4. Game 4 5. Back Please select an option:

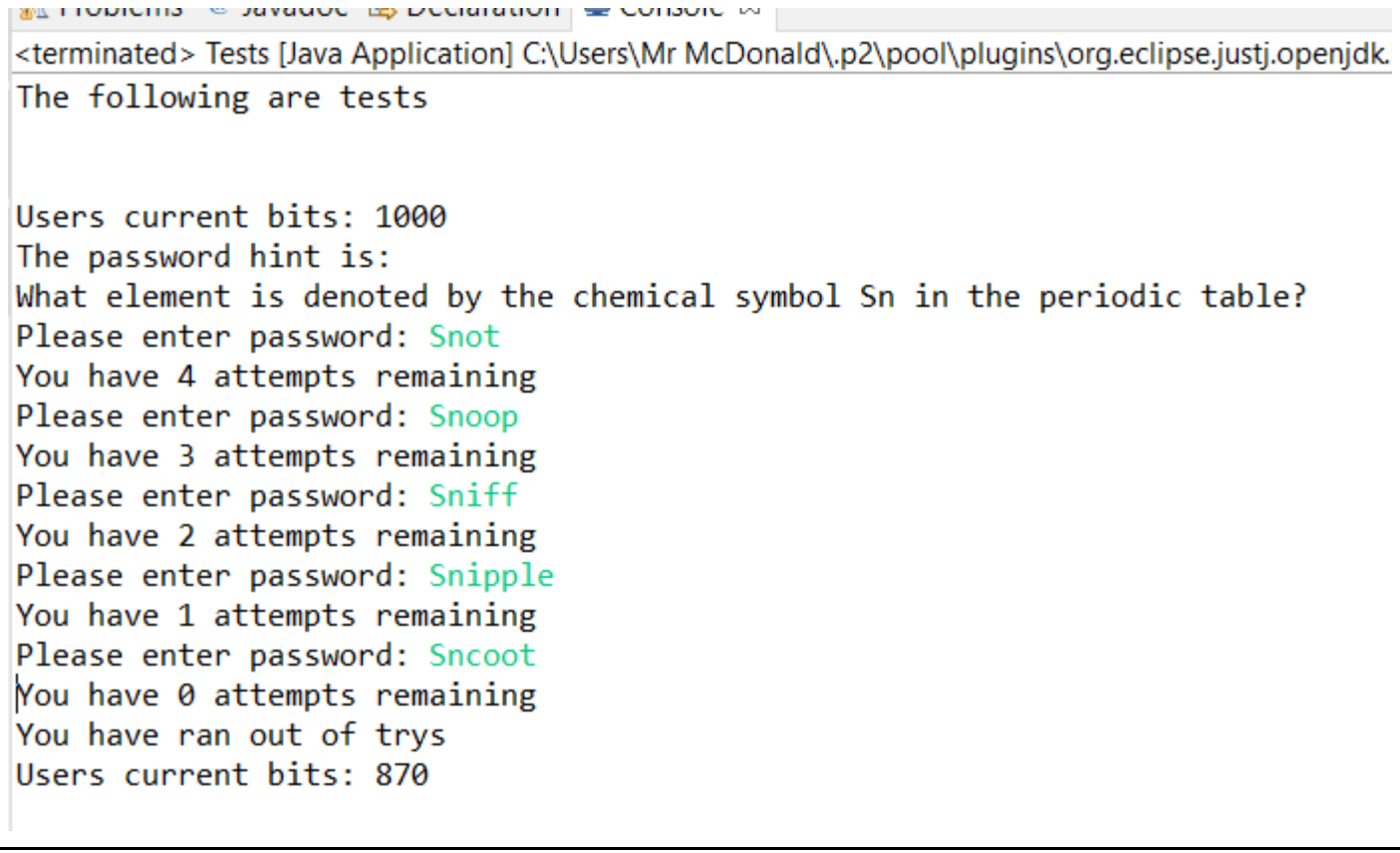
67	 Please enter how many people will be playing the game: 3 Enter player 1 name: Dean Enter player 2 name: JB Enter player 3 name: Scott Now lets roll to see what order everyone will be playing in!
68	 Please enter how many people will be playing the game: 4 Enter player 1 name: Conor Enter player 2 name: Dean Enter player 3 name: Scott Enter player 4 name: JB Now lets roll to see what order everyone will be playing in! Conor you have rolled 5 Dean you have rolled 4 Scott you have rolled 5 JB you have rolled 0 Conor you will go 1 Scott you will go 2 Dean you will go 3 JB you will go 4 Conor you currently have 1000 bits  Conor you currently have 1000 bits Conor please choose from the list below: 1. Display Board Layout 2. Display Rules 3. Organise Files 4. Roll Dice 5. Vote To Finish Game Please enter your choice:
69	 Please enter how many people will be playing the game: 1 Please enter a number larger than 1 Please enter how many people will be playing the game:

70	 Technopoly! 1. Start Game 2. Continue Game 3. Exit Please enter your choice: 1 The game is starting :) Please enter how many people will be playing the game: -1 Please enter a number in the range of 2-9 Please enter how many people will be playing the game: 10 Please enter a number in the range of 2-9 Please enter how many people will be playing the game:
71	 Please enter how many people will be playing the game: I love the smell of napalm in the morning Please enter an integer for the number of players Please enter how many people will be playing the game:
72	 Please enter how many people will be playing the game: 4.2 Please enter an integer for the number of players Please enter how many people will be playing the game:
73	 jb you came in 4 place! Congrats conor you came in 3 place! Congrats scott you came in 2 place! Congrats dean you came in 1 place! Congrats

74	jb you came in 4 place! Congrats conor you came in 3 place! Congrats scott you came in 2 place! Congrats dean you came in 1 place! Congrats
75	The following are tests Files player owns: Software Process.mpp, 1, Practical Project Management.mpp, 3, Setting player storage spac Conor you are out of bits and therefore out of the game Dean you currently have 300 bits Dean please choose from the list below: 1. Roll Dice 2. Organise Files 3. Display Board Layout 4. Display Rules 5. Save Game 6. Vote To Finish Game Please enter your choice: You have recieved 200 bits for passing Go! Dean you have rolled a 2 and a 5 meaning you will move 7 places You have moved from position 34 to 1 You have landed on Software Process.mpp Do you wish to download this file for 60 (y/n): Note: Second screenshot was taken once Conor was out of the game on Deans go. Showing that Conor owned Software Process.mpp before he was out of the game, then once he was out of the game Dean can now purchase this file proving that Conor no longer owns it.

76	The following are tests The password hint is: How many of Henry VIII's wives were called Catherine? Please enter password:
77	The following are tests The password hint is: In which part of your body would you find the cruciate ligament? Please enter password: knee Well done the password is correct you no longer have to pay the owner of this file

78	The following are tests The password hint is: How many of Henry VIII's wives were called Catherine? Please enter password: Patrick You have 4 attempts remaining Please enter password:

79	 <terminated> Tests [Java Application] C:\Users\Mr McDonald\.p2\pool\plugins\org.eclipse.justj.openjdk. The following are tests Users current bits: 1000 The password hint is: What element is denoted by the chemical symbol Sn in the periodic table? Please enter password: Snot You have 4 attempts remaining Please enter password: Snoop You have 3 attempts remaining Please enter password: Sniff You have 2 attempts remaining Please enter password: Snipple You have 1 attempts remaining Please enter password: Sncoot You have 0 attempts remaining You have ran out of trys Users current bits: 870

80	The following are tests Current Questions: [In which part of your body would you find the cruciate ligament?, Knee][What is the na Current Used Questions: The password hint is: What is Captain America's shield made out of? Please enter password: Vibranium Well done the password is correct you no longer have to pay the owner of this file Current Used Questions: What is Captain America's shield made out of?

81	The following are tests Current Questions: [In which part of your body would you find the cruciate ligame Current Used Questions: The password hint is: Who killed Tony Stark's parents? Please enter password: You have 4 attempts remaining Please enter password: You have 3 attempts remaining Please enter password: You have 2 attempts remaining Please enter password: You have 1 attempts remaining Please enter password: You have 0 attempts remaining You have ran out of trys Dean, you have given Scott 170 bits Current Used Questions:

82	Tests [Java Application] C:\Users\Mr McDonald\.p2\pool\plugins\org.eclipse.justj.openjdk.h The following are tests Users current position: 39 Dean you currently have 1000 bits Dean please choose from the list below: 1. Display Board Layout 2. Display Rules 3. Organise Files 4. Roll Dice 5. Vote To Finish Game Please enter your choice: 4 You have recieved 200 bits for passing Go! Dean you have rolled a 3 and a 1 meaning you will move 4 places You have moved from position 39 to 3 You have landed on Practical Project Management.mpp

83	Tests [Java Application] C:\Users\Mr McDonald\.p2\pool\plugins\org.eclipse.justj.openjdk.hot The following are tests Dean you currently have 1000 bits Dean please choose from the list below: 1. Display Board Layout 2. Display Rules 3. Organise Files 4. Roll Dice 5. Vote To Finish Game Please enter your choice: 4 Dean you have rolled a 0 and a 5 meaning you will move 5 places You have moved from position 24 to 29 You have landed on Agile Manifesto.mp4 Do you wish to download this file for 300 (y/n):

84	The following are tests Number of Util Pass before: 0 Users current position: 24 Number of Util Pass after: 1 Users new position: 4