Testing

Test plan

In this stage of the project, I will be conducting a test plan for all areas of the system. I will use a wide range of testing methods, along with normal (data that the system would normally work with), erroneous (data that is invalid and would produce an error message) and extreme data (data at the boundary between normal and erroneous data). Moreover, I will complement the results of each test with a screenshot along with commentaries, including any specific suggestions to refine the system.

Staff side testing

Login screen

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if the login screen works by seeing what happens when valid credentials are presented | Entered username: willb4  Entered password: brúsku | Normal | The system should grant the user access into the system | The system logs in the user to the system |  | It worked as intended and no changes were needed |
| 2 | Testing if the login screen works by seeing what happens when invalid credentials are presented | Entered username: waskb4  Entered password:  blyat | Erroneous | The system should display a message informing the user of incorrect credentials | The system produced the expected error message |  | It worked as intended and no changes were needed |
| 3 | Testing if the login screen works by seeing what happens when extreme credentials are presented | Entered username: willb4  Entered password: brúsk | Extreme | The system should distinguish that the the password is not correct and deny access | The system detects that they don’t match |  | It worked as intended and no changes were needed |

Buying price calculator

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if the buying price calculator works by seeing what happens when valid data is entered | I will enter days since release date 40, condition as 10, popularity as 10 and RRP as £50 | Normal | The system should return a price value of £37.5 | The output was £37.5 |  | It worked as intended and no changes were needed |
| 2 | Testing if the buying price calculator works by seeing what happens when no data is entered | I will leave all of the fields back | Erroneous | The system should display a message informing the user that they need to enter all required fields | The system produced the expected error message |  | It worked as intended and no changes were needed |
| 3 | Testing if the buying price calculator works by seeing what happens when extreme data is entered | I will enter the RRP of said game to be £-1 and leave other fields as default value | Extreme | The system should still use the input against the algorithm to produce the output of £-13.8 | The expected price was calculated, although an unrealistic RRP was given |  | It worked as intended and no changes were needed |

Resell price calculator

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if the resell price calculator works by seeing what happens when valid data is entered | I will enter the Price Paid for the game as £24 | Normal | The system should return a resell price value of £30 | The output was £30 |  | It worked as intended and no changes were needed |
| 2 | Testing if the buying price calculator works by seeing what happens when no data is entered | I will leave the Price paid for game field blank | Erroneous | The system should display a message informing the user that they need to enter a value into the Price Paid field | The system crashed, stating that the input string was given in the incorrect format |  | The test failed as the system crashed instead of producing an error message.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |
| 3 | Testing if the buying price calculator works by seeing what happens when extreme data is entered | I will enter the Price paid for game as £-1 | Extreme | The system should still use the input against the algorithm to produce the output of £-1.25 | The expected price was calculated, although an unrealistic Price Paid was given |  | It worked as intended and no changes were needed |

Adding games into the system

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if adding a game in the system works by entering valid data | Bought from a customer, title rocket league, price £14.99, genre sport, developer Psyonix and CustomerID 30 | Normal | The system should first let the user know that they don’t need to enter the GameID as it didn’t currently exist in the system. It should inform the user that the game has been added to the system and then ask if they wish to log the transaction in the database. | When hovering over the GameID, the user is made aware that input in said textbox is unnecessary when buying a game from a customer as it wouldn’t have been currently existing in the system. When Add/Remove was pressed the system then added the game to the system, informed this to the user, then asked if they would like to log the transaction to the system. After yes was pressed, the user was then told that the game transaction was logged into the system. |  | No changes needed to be made as outputs were as expected. |
| 2 | Testing if adding a game in the system works by entering erroneous data | I will leave all of the fields blank (apart from bought/sold) | Erroneous | The system should display a message explaining that all data fields need to be entered before proceeding | The actual output did not match expected. The form remained blank. |  | The test failed as no error message was displayed to the user.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |
| 3 | Testing if adding a game in the system works by entering extreme data | I will enter the price of the game being £-15 | Extreme | The system should display a message explaining that invalid data has been entered for the price value | The system did not respond as expected. The “game” was added to the system with the invalid information provided. |  | The test failed as the “game” was still added to the system despite the extreme data entered for the price.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |

Removing games from the system

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if removing a game from the system works by entering valid data | Selling to a customer, title rocket league, price £14.99, genre sport, developer Psyonix, GameID 35 and CustomerID 30 | Normal | It should inform the user that the game has been removed from the system and then ask if they wish to log the transaction in the database. | The program did what as expected, and the game was removed from the system successfully and the transaction was recorded to the RecentGames table. |  | No changes needed to be made as outputs were as expected. |
| 2 | Testing if removing a game from the system works by entering erroneous data | I will remove a game from the system that doesn’t exist. Selling to a customer, Game title = test, price = 1, genre = test, developer = test, CustomerID = 33 and GameID = 999 | Erroneous | The system should display a message explaining that the game entered does not exist and thus cannot be sold to a customer | The actual output did not match expected. The game was still sold |  | The test failed as no error the non-existent game was “removed” from the database.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |
| 3 | Testing if removing a game from the system works by entering extreme data | I will edit the “BoughtOrSold” dropdown box to the value “Test” | Extreme | The system should display a message explaining that I must be selling a game to a customer or buying a game from a customer. | The actual output did not match expected. The form remained blank. |  | The test failed as no error message was displayed to the user.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |

View all available games

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will be testing the form to ensure a staff member is able to view all available games on the system | I will click the view all available games button | Normal | I expect the form to display all of the games that are available on the system | The system did as I expected |  | No changes needed to be made as outputs were as expected. |

View games recently sold to/bought from customers

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will be testing the form to ensure a staff member is able to view all games that have been sold and bought | I will click the view games recently sold to/bought from customers | Normal | I expect the form to display all of the games that have recently been sold to or bought by customers | The system did as I expected |  | No changes needed to be made as outputs were as expected. |

Game search

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if searching for a game in the system works by entering valid data | I will input the game im looking for (Game Title) as Half Life: Alyx | Normal | It should find the game (as it exists on the system) and then return its relevant details and then populate them in the corresponding text boxes | The program did what as expected, the game was found and its relevant details were displayed to the user |  | No changes needed to be made as outputs were as expected. |
| 2 | Testing if searching for a game in the system works by entering invalid data | I will search for a game that doesn’t exist on the system: Game Title: Half Life 2 | Erroneous | The system should display a message explaining that the game entered does not exist on the system. | The actual output met the expected. The user was informed that the game didn’t exist on the system |  | No changes needed to be made as outputs were as expected. |
| 3 | Testing if searching for a game in the system works by entering extreme data | I will attempt to search for a game with the Game Title left blank | Extreme | The system should display a message explaining that the game name field must be entered before a game can be searched | The actual output did not match expected. The system explained that there was no game called “” |  | The test failed as the incorrect error message was displayed to the user.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |

Customer search

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if the customer search form works by entering valid data | I will enter the customer username as: smp09, which is a valid customer. I should then be presented with all of their relevant information | Normal | The system should output the details for CustomerID 25 | The expected output was presented |  | It worked as intended and no changes were needed |
| 2 | Testing if the customer search form works by entering invalid data | I will leave the customer username field blank | Erroneous | The system should display a message informing the user that they have left the customer username field blank | The actual output was not as expected. It said “No customer with the name “” found” |  | I have now updated the code to produce the error message shown on the left. |
| 3 | Testing if the customer search form works by entering extreme data | In one instance of the form, I will search for 3 customer’s details: Somnaint, bepis and fishbultz | Extreme | The system should replace the text boxes with the new information each time | The output was as expected |  | It worked as intended and no changes were needed |

View all customers

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will be testing the form to ensure a staff member is able to view all games that have been sold and bought | I will click the view all customers | Normal | I expect the form to display all of the customers and their information that exist on the system | The system did as I expected |  | No changes needed to be made as outputs were as expected. |

Update customer details

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if the update customer details form works with valid data | I will enter the CustomerID as 28 (which is a valid customer), the information to be updated as surname and the new field data to be Brúsku | Normal | The system should update the customer’s record with the information provided | The expected output was presented |  | It worked as intended and no changes were needed |
| 2 | Testing if the update customer details form works with invalid data | I will enter the CustomerID as 28 (real customer), information to be updated as PhoneNum and new field data as 12345 (invalid phone number) | Erroneous | The system should display an error message informing the user that the phone number they entered is invalid | The actual output was not as expected. It updated the customer’s record with the invalid data |  | The test failed as the system allowed the user to edit the customer’s details using invalid data  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |
| 3 | Testing if the customer search form works by entering extreme data | I will leave all of the required fields blank | Extreme | The system should produce an error message informing the user that before any process can be made, each field needs to be populated | The output was not as expected. The form remained blank. |  | The test failed as the form just remained blank, without providing any error message.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |

View customer requests

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will be testing the form to ensure a staff member is able to view all requests that have been sent to the staff team by customers | I will click the view customer requests button from the staffmenu form | Normal | I expect the form to display all of the customer requests and relevant information | The system did as I expected |  | No changes needed to be made as outputs were as expected. |

Send customer request

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if the send customer request form works with valid data | I will enter the user as smp09, requestID as 4 (real customer request) and then the Message: re: Quote request: If you'd bring modern warfare 2 into store, you'd get £4.99 for it. Thanks. | Normal | The system should send the message to the customer, and then ask the staff member if they wish to delete the original request that corresponded to that reply from the database | The expected output did not match the actual output. Instead, |  | The test failed as the system crashed once the message was sent.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |
| 2 | Testing if the send customer request form works with invalid data | I will enter the following data:  Customer username: neem  RequestID: 10  Message: You will get £15 if you bring GTA V in store | Erroneous | The system should display an error message informing the user that the reply cannot be sent as the original request doesn’t exist | The actual output was as expected. |  | No changes needed to be made as outputs were as expected. |
| 3 | Testing if the send customer request form works with extreme data | I will leave all of the required fields blank | Extreme | The system should produce an error message informing the user that before any the message can be sent, all fields need to be filled | The actual output was as expected. |  | The actual output was as expected. |

Customer side testing

Login screen

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will test the system to see if existing customers will be able to login to the system without an issue | Entered username: fishbultz  Entered password: fortnite | Normal | The system should grant the user access into the system as the credentials match. | The system logs in the user to the system |  | It worked as intended and so no changes were needed |
| 2 | Testing if the login screen works by seeing what happens when invalid credentials are presented | Entered username: fishtank  Entered password:  basketball | Erroneous | The system should display a message informing the user that they have presented incorrect credentials and therefore they will not be given access to the system. The text boxes shall then be cleared to facilitate the re-entry of data. They should then be informed of how many attempts they have left. | The system produced the expected error message. |  | It worked as intended and so no changes were needed |
| 3 | Testing if the login screen works by seeing what happens when extreme credentials are presented (password correct but username wrong) | Entered username: fishbultz  Entered password: fortntie | Extreme | The system should distinguish that the the password is not correct and deny access and display the relevant message including their attempts left. The system should also not inform the user that they got one out of the two correct, as this will facilitate brute force entry attacks. | The system detects that they don’t match and denies access as well as informing of the attempts remaining. |  | It worked as intended and no changes were needed |

Account creation

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will test the system to ensure that when user enters valid customer details and attempts to create an account, they are able to. | I will enter the following data:  Username: niki2001  Password:  Dutedracu  First name: Nicolae  Surname: Popescu  Telephone No:  07848574554  Full address:  12 Tree Road  County: London  Town: Barking  Postcode: SW12 9PL | Normal | The system should allow the user to create an account and inform them of their success as well as appending this information to the database for reference when logging in. | The system allowed the user to create the account and added their information o the database. |  | It worked as intended and so no changes were needed |
| 2 | I will test the system to see how it responds when a user tries to create an account with invalid data. | I will enter the following data:  Username: niki  Password:  sugi  First name: Nicolae  Surname: Popescu  Telephone No:  12345  Full address:  12 Tree Road  County: 1337  Town: 999  Postcode: SW19P | Erroneous | The system should not allow the user to create the account. Moreover, they should be informed that their username and password need to be over 5 characters, that their telephone number needs to consist of 11 digits, that their county and town must only be composed of letters and that the postcode should be 7 or 8 characters long (according to UK format) including a space in the middle | The system produced the expected error message. |  | The expected outputs were presented to the user and so no changes were needed.  Reference to this validation has been included in the Developmental Testing section of the project. |
| 3 | I will test how the create account form responds when all fields are left blank | I will not enter any data. | Extreme | The system should detect that not all fields have been filled and a subsequent error message should then be produced by the system informing the user that all fields are required. | The system produced the expected error message. |  | It worked as intended and no changes were needed |

Game search

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | Testing if searching for a game in the system works by entering valid data | I will input the game I’m looking for (Game Title) as buhnroP | Normal | It should find the game (as it exists on the system) and then return its relevant details and then populate them in the corresponding text boxes. | The program did what as expected, the game was found and its relevant details were displayed to the user as well as notifying that their game was found. |  | No changes needed to be made as outputs were as expected. |
| 2 | Testing if searching for a game in the system works by entering invalid data | I will search for a game that doesn’t exist on the system: Game Title: brúsku | Erroneous | The system should display a message explaining that the game entered does not exist on the system. | The actual output met the expected. The user was informed that the game didn’t exist on the system |  | No changes needed to be made as outputs were as expected. |
| 3 | Testing if searching for a game in the system works by entering extreme data | I will attempt to search for a game with the Game Title left blank | Extreme | The system should display a message explaining that the game name field must be entered before a game can be searched. | The actual output matched the expected |  | No changes needed to be made as outputs were as expected. |

Request search

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will test if a customer is able to send a valid request to the staff members. | I will input the following data:  Request type: Buy Request  Message: Hello, I saw you guys had buhnroP in store, I was wondering when I’d be able to come up and make the purchase. Thanks. | Normal | It should allow the customer to send the request, inform them of their success and then add their request to the database. | The program did what as expected, request was sent and saved in the database to allow the staff members to read it. |  | No changes needed to be made as outputs were as expected. |
| 2 | I will test how the request sender form copes when erroneous data is entered | I will edit the request type drop down box to the value 123 and the Message as 123 | Erroneous | The system should display a message explaining that the drop down box value must be on of the determines ones, and the message should be valid | The actual output was not as expected. The request was sent to the database. |  | The test failed as the invalid request was sent.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |
| 3 | I will test how the request sender form copes with extreme data | I will enter a message: Hello, I want some of my details to be updated. Firstly, I wanted my first name to be changed to Master as I have had it changed now by law. I would also want to have my surname changed to Damascus as I have taken my wife's surname (we married last week). I would also want to have my phone number changed to 07283748332 because I was given a new mobile phone. | Extreme | The system should display a message explaining that the message is too long and that it must be shorter. | The actual output did not match the expected. The system produced a message box saying that the message was sent, before crashing. |  | The test failed as the system crashed once the message was sent.  I will include the corrections that I have made to the code in the Developmental Testing section of the project. |

Change credentials

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will test if a customer is able to view all of the replies of their requests | I will click the View request replies button on the Customer menu | Normal | It should show the current logged in user (smp09) all of the replies they’ve received | The program did what as expected, the currently logged in user (smp09) was displayed all of their replies from the database. |  | No changes needed to be made as outputs were as expected. |
| 2 | I will test if a customer is able to delete the oldest reply that they have received from a staff member. | I will click the Delete oldest reply button. | Normal | The oldest reply on the list should be deleted from the customer’s view as well as from the database. They should also be notified of the successful deletion. | The actual output was not as expected. The oldest reply on the Data Grid View was deleted, as well as being deleted from the database. The user was also notified of the success of this process. |  | No changes needed to be made as outputs were as expected. |

View customer requests

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will be testing the form to ensure a customer is able to view all games currently available to be bought from the store | I will click the view all games currently available button from the customer menu | Normal | I expect the form to display all of the games that are available to be bought to the customer. | The system did as I expected. All of the available games were displayed to the customer via the Data Grid View |  | No changes needed to be made as outputs were as expected. |

View details

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will be testing the form to ensure a customer is able to view all of the data that the system has stored about them | I will click the view my details button from the customer menu form | Normal | I expect the form to display all of the information that the system has stored about the user logged in (fishbultz) | The system did as I expected. The user was able to see all of the data stored about them by each piece of data being used to populate the text box labelled as that data. |  | No changes needed to be made as outputs were as expected. |

View details

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test No | Test description | Input test data | Test type | Expected output | Actual output | Screenshot | Comments |
| 1 | I will be testing the form to ensure a customer is able to view all of the data that the system has stored about them | I will click the view my details button from the customer menu form | Normal | I expect the form to display all of the information that the system has stored about the user logged in (fishbultz) | The system did as I expected. The user was able to see all of the data stored about them by each piece of data being used to populate the text box labelled as that data. |  | No changes needed to be made as outputs were as expected. |

Acceptance Testing

Introduction: This test section has been created to illustrate client satisfaction regarding how the tests have met all of the requirement criteria that was specified by him. The client has checked all of the tests that I have made created and has subsequently given me feedback in the form of Yes/No answers, which determine if he thinks that each criteria has been met. The table below shows each test as well as whether or not the client believes the requirement has been met (shown by the Yes/No column). Finally, I have also included a comment column which describes in detail how/how not the requirement has been met.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No | Requirement specification | Test Category | Test No | Test result as expected:  Y/N | Requirement met:  Y/N | Comments |
| 1 | The system will be able to calculate selling prices for games using a set algorithm | Resell price calculator | 1 | Yes | Yes | The resell price calculator worked as it should and provides a business suitable value for games that have been bought from customers. |
| 2 | The system will be able to calculate buying prices for games using a set algorithm | Buying price calculator | 1 | Yes | Yes | The buying price calculator worked as it should and has also proved a large improvement from the prototype version where the member of staff had to manually enter the condition and popularity of the game. Now drop down boxes are used for those fields instead which facilitates the process as well as making it more efficient and accurate since it promotes faster, more restricted data entry. Overall, this part of the system functions as it should and calculates business suitable values for the games that are being bought |
| 3 | The system will allow staff members to login using their pre-set logins | Staff login | 1 | Yes | Yes | This requirement for staff members to be able to login their pre-set account has been met as all 3 staff members are currently able to login to the system using their credentials. |
| 4 | Allow customers to login to their existing account or create a new one in order to allow them to access the features that a customer account offers | Customer Login  Create account | 1  2 | Yes | Yes | This requirement for customers to be able to login to their existing account or to be able to create a new one have both been met. If a new customer wishes to create a new account they are required to fill in all specified information and accurately (checked by validation techniques) before they are allowed to carry out the task. This helps to ensure data integrity. |
| 5 | Allow staff members to buy and sell games to/from customers by entering relevant game and customer details | Adding games to the system  Removing games from the system | 1  2 | Yes | Yes | The system was able to add games to the system successfully using the data entry of the Game title, Price, Genre, Developer and CustomerID of who sold it to the store. Moreover, the system was also able to delete games from the system using the data entry of all of the fields specified above, along with the GameID. Both buying/selling transactions of games are logged in the system if the staff member chooses to, or if not in the case of a game being lost or stolen. |
| 6 | Allow the viewing of games that have been previously sold to/bought from customers | View recent games | 1 | Yes | Yes | The ability to view all games that have been previously bought by / sold to customers is very important as it allows staff members to keep track of what games have been sold/bought and to/from who. |
| 7 | Allow staff members to check the quantity of each game in store | N/A | N/A | N/A | Not entirely | The ability to specifically find out the quantity of a certain game on the system has not been implemented. Although, while viewing all available games, if sorted by Name, it can be manually counted how many of one game exists on the database. |
| 8 | Allow staff members to view and update customer information | View all customer details  Update customer details | 1  2 | Yes | Yes | This requirement of the system has been successfully met, when a staff member wishes to view all customers currently registered on the system, they are able to. Moreover, if they wish to update data on a customer record (as per customer request) they are able to, by entering the CustomerID, the field to be updated and valid new data (checked by validation) to replace the old data. |
| 9 | Allow anyone to view all of the games currently available | Staff: View all available games  Customer: View all available | 1  2 | Yes | Yes | Both staff members and customers are able to view all games on the system. This is an essential feature for both parties, as much as for staff members who want to keep track of what games are in stock as for customers who want to look for games to buy. Previously, when Mongoose Games used the paper based system, staff members had to contain a record of all games on the system, hand written. Moreover, customers had to ask staff members constantly what games were available to avoid having to look around the store. Now, both staff members and customers can carry out this task more efficiently and effectively, as well as reducing the amount of customer requests, saving time for staff. |
| 10 | Allow anyone to search through the games currently available | Staff: Game search  Customer: Game search | 1  2 | Yes | Yes | Both staff members are also currently able to search through the games in stock by their title. This is a very important feature for both parties because as the store will become more popular and profitable, more games will be stocked, meaning searching for one game in a database with thousands would be like finding a needle in a haystack. Therefore, this feature facilities the action for both staff members and customers to search for a game. Staff members could do this if they were verifying that a game they added to the system had been appended, or if asked by a rushed customer. Customers could use this if a staff member was busy, to be able to by themselves look for a certain game that they wish to purchase. This improves customer service as waiting times are reduced and customers could feel more independent and self-sufficient to be able to answer their query without asking a staff member. Again, compared to the old paper-based system, it’s a drastic improvement as finding a certain game from the records in the notebooks was heavily time consuming. |
| 11 | Develop the program so that customers can access the system from home via an application to pre-order a game or set up a sale for a game they have | Send request | 1 | Yes | Not fully | This requirement has not been fully met as the ability for users to access the system is not yet an option. I have discussed this further in my evaluation under the Successful features and suggested improvements section. However the parts of the criteria that have been met are the ability for customers to send pre-order requests as well as setting up a sale for their a game they have although they can only do this while being logged into a computer at the store. |
| 12 | Allow customers to receive a quote for their game | Send request  View request replies | 1  2 | Yes | Yes | This requirement has been met and fits under the send request form, where a customer can send a Quote request, specifying what game they wish to sell. They will then get a reply from a staff member telling them how much they would be given, and this would be viewed from the view request replies form. |
| 13 | Allow customers to be able to change their credentials | Change credentials | 1 | Yes | Yes | This requirement has been successfully met as customers are able to change their username/password whenever they wish as long as they know they current username and password, to be able to log into the system. |
| 14 | Allow customers to view all of the information that the system has stored about them | View your details | 1 | Yes | Yes | The ability for customers to have access to their customer record is essential as it allows customers to verify data integrity which complies with GDPR rules. With the paper-based system, it was a much more troublesome task for a staff member to show the customer what details were stored about them, as they had to find their record first on the notepads and had to physically show it to them which was very unprofessional and provides possibilities of a data breach due to physical records being kept of sensitive customer data. |
| 15 | Allow games that are bought from customers to be put up as available to be bought in the system instantly but for a higher price (determined by a set algorithm) | Adding games to the system  Resell price calculator | 1  2 | Yes | Not fully | The ability for games to be instantly put up for sale after being bought from a customer has not been implemented. However, a more manual and methodical way of carrying out this is possible. This involves the following: after a game has been purchased from a customer, you input the price paid for it into the resell price calculator to get a value of how much you should sell it for, you then go to the adding game to the system function and input the required details, and for the price value you enter what was given by the resell price calculator. |