# IT/SD TEAM PROJECT

**********

***Team Name: The Java Lamps***

*Members*:

Kirsten Miller (1102394M)

Susie Smart (2286107S)

Stephanie Man ( 2286857M)

Anupam Chakraborty (2284740C)

Hannah Pankow (2284062P)

*Top Trumps*

# Table of Contents

# 1 Details of the Database

Username: m_16_2284062p

Database: m_16_2284062p

Password: 2284062p

# 2 Introduction

The program implements a version of Top Trumps. It reads in a text file listing the categories and values for the individual cards. A GUI allows the user to play against up to four computer players and provides all the relevant information to understand and follow the game-flow. A range of message boxes guides the user in a self-explanatory fashion. The program further connects to a database in order to allow game results to be stored. In addition, the user also has the option to write the information to a text file.

This report explains the user stories and sprint planning while also tracking the group's progress through the use of a Burn Down Chart. All assumptions made regarding the game logic are introduced and a wide variety of tests serves to demonstrate the game's functionality.

# 3 Current Status

The program is working and fulfils all functional requirements.

# 4 User Stories

User stories were generated through a group brainstorming session. Each user story was discussed in turn and the story title, description, and conversation were recorded on the front of the index cards. The team also considered the various confirmations and tests that would be required for each user story. This information was recorded on the back of the cards. The scanned in copies of all the story cards are located in Appendix A. Table 1 shows a summary of the initial user stories generated by the group.

### Table 1: Initial User Stories

| User Story Number | User Story | Estimated Story Points |
|---|---|---|
| 1 | Create an option to start a new game which will launch game play | 16 |
| 2 | Create an option to select how many players will be in the game | 0.5 |
| 3 | Create options for each category so that the player can select a category when it is their turn | 2 |
| 4 | Allow the user to exit the game | 0.5 |
| 5 | Allow the user to save the game statistics to the database | 1 |
| 6 | Provide an option for the user to view the game statistics | 1 |
| 7 | Provide an option to discard the results from the current game so that they will not be added to the database | 1 |
| 8 | Allow the user to save the game statistics to a text file | 1 |
| | Total Estimated Story Points | 23 |

Table 1 also contains the estimated story points for each user story. The story points were again generated through a group session where each participant estimated the number of story points per user story in confidence. The team then discussed the estimations and explained their thought processes until everyone converged on an agreed value.

User story 1 has an estimated story point value of 16 which is significantly higher than any of the other user stories. The team therefore decided that this story could be considered as an epic and should be broken down further into smaller user stories. Table 2 details how the epic was broken down further into user stories 1.1 to 1.7.

### Table 2: Expanded user stories

| User Story Number | User Stories | Estimated Story Points |
|---|---|---|
| 1 | Create an option to start a new game which will launch game play | |
| 1.1 | Read in cards from deck.txt | 1 |
| 1.2 | Shuffle Cards | 2 |
| 1.3 | Deal cards | 1 |
| 1.4 | Pick random player to begin | 1 |
| 1.5 | Allow computer to select a category | 4 |
| 1.6 | Compare scores and pick winner | 4 |

| | | |
|---|---|---|
| 1.7 | Handle draw | 4 |
| 2 | Create an option to select how many players will be in the game | 0.5 |
| 3 | Create options for each category so that the player can select a category when it is their turn | 2 |
| 4 | Allow the user to exit the game | 0.5 |
| 5 | Allow the user to save the game statistics to the database | 1 |
| 6 | Provide an option for the user to view the game statistics | 1 |
| 7 | Provide an option to discard the results from the current game so that they will not be added to the database | 1 |
| 8 | Allow the user to save the game statistics to a text file | 1 |
| | **Total Story Points** | 24 |

# 5 Sprint Planning and Review Reports

## 5.1 Sprint 1: Planning Report

Scrum master: Stephanie Man (SM)

Scrum Team: Anupam Chakraborty (AC), Kirsten Miller (KM), Hannah Pankow (HP), Susie Smart (SS)

Table 3 contains the product backlog for sprint 1. The user stories selected for sprint 1's back log were chosen as they were the main user stories that would need to be completed for the game to function. The order in which the team planned to complete these tasks was based on a logical process where some user stories naturally progress to the next. For example, "Read in cards from deck" had to be completed before "Shuffle cards" could be implemented. The tasks to complete each user story were divided between the scrum team to ensure they would be completed in the time boxed sprint.

**Table 3: Sprint 1 User Stories**

| User Story Number | Sprint 1: User Stories | Estimated Story Points | Allocated Team member |
|---|---|---|---|
| 1.1 | Read in cards from deck.txt | 1 | KM |
| 1.2 | Shuffle Cards | 2 | KM & SS |
| 1.3 | Deal cards | 1 | KM |
| 1.4 | Pick random player to begin | 1 | KM |

| | | | |
|---|---|---|---|
| 1.5 | Allow computer to select a category | 4 | KM & SS |
| 3 | Create options for each category so that the player can select a category when it is their turn | 2 | SS |
| | **Total Story Points** | 11 | |

During this two-week sprint, AC and HP were also tasked with designing the database for the system while SM took on the responsibility of generating the documentation including collating all the user stories.

## 5.2 Sprint 1: Review Report

Throughout the first sprint the team reported their individual progress at regular stand up meetings and the actual story points used for each user story were recorded. The actual story point values are displayed in Table 4.

**Table 4: Sprint 1 Actual Story Points**

| User Story Number | Sprint 1: User Stories | Estimated Story Points | Actual Story Points |
|---|---|---|---|
| 1.1 | Read in cards from deck.txt | 1 | 0.5 |
| 1.2 | Shuffle Cards | 2 | 0.5 |
| 1.3 | Deal cards | 1 | 2 |
| 1.4 | Pick random player to begin | 1 | 1 |
| 1.5 | Allow computer to select a category | 4 | 2 |
| 3 | Create options for each category so that the player can select a category when it is their turn | 2 | 2 |
| | **Total Story Points** | 11 | 8 |

Table 4 illustrates that in some cases the estimated story points were too high, for example in 1.1 and 1.2 the user stories were simpler to implement than predicted. However, there were also user stories which took longer than initially expected for example 1.3 “Deal cards” was more complex and took longer to debug than planned. Overall, all the user stories in sprint 1 were completed in fewer days than estimated and so the team was in a strong position going into sprint 2.

The team also produced a plan for the database design and collated the user story cards during this sprint.

The main lessons learned during this sprint were that the team worked well whilst pair programming. It was therefore decided that this would continue onto the next sprint with SS and KM working in a team and HP and SM working as another pair.

### 5.3 Sprint 2: Planning Report

Scrum master: Stephanie Man (SM)

Scrum Team: Anupam Chakraborty (AC), Kirsten Miller (KM), Hannah Pankow (HP), Susie Smart (SS)

Table 5 contains the product backlog for sprint 2. The user stories selected for sprint 2's back log were the remaining user stories identified in the original brainstorming session. The tasks to complete each user story were divided between the scrum team to ensure they would be completed in the time boxed sprint.

**Table 5: Sprint 2 User Stories**

| User Story Number | Sprint 2: User Stories | Estimated Story Points | Allocated Team member |
|---|---|---|---|
| 1.6 | Compare scores and pick winner | 4 | KM & SS |
| 1.7 | Handle draw | 4 | KM |
| 2 | Create an option to select how many players will be in the game | 0.5 | SS |
| 4 | Allow the user to exit the game | 0.5 | SS |
| 5 | Allow the user to save the game statistics to the database | 1 | HP & SM |
| 6 | Provide an option for the user to view the game statistics | 1 | HP |
| 7 | Provide an option to discard the results from the current game so that they will not be added to the database | 1 | HP |
| 8 | Allow the user to save the game statistics to a text file | 1 | HP & SM |
| | **Total Story Points** | **13** | |

During this two-week sprint, AC was responsible for updating the documentation including the Burn Down chart and product back log. SM continued as the Scrum Master, running the regular stand ups and supporting the team throughout the sprint.

## 5.4 Sprint 2: Review Report

Throughout the second sprint the team continued to report their individual progress at regular stand up meetings and the actual story points used for each user story were again recorded. The actual story point values are displayed in Table 6.

**Table 6: Sprint 2 Actual Story Points**

| User Story Number | Sprint 2: User Stories | Estimated Story Points | Actual Story Points |
|---|---|---|---|
| 1.6 | Compare scores and pick winner | 4 | 4 |
| 1.7 | Handle draw | 4 | 8 |
| 2 | Create an option to select how many players will be in the game | 0.5 | 0.5 |
| 4 | Allow the user to exit the game | 0.5 | 0.5 |
| 5 | Allow the user to save the game statistics to the database | 1 | 4 |
| 6 | Provide an option for the user to view the game statistics | 1 | 2 |
| 7 | Provide an option to discard the results from the current game so that they will not be added to the database | 1 | 0.5 |
| 8 | Allow the user to save the game statistics to a text file | 1 | 2 |
| | **Total Story Points** | 13 | 21.5 |

Table 6 illustrates that in some cases the estimated story points were too high, for example in 7 the user story was simpler to implement than predicted. However, there were also user stories which took longer than initially expected for example 1.7 “Handle draw” was far more complex, taking almost double the story points than were estimated.

At the end of sprint 2 all the user stories had not been completed and user stories 6 to 8 had to be planned and completed in an additional period. It was clear that the two sprints had not been as evenly split as originally thought.

The team also produced continued comments to keep the documentation up to date during this sprint.

The main lessons learned during this sprint were that it is easy to underestimate the story points. The team identified that 1.7 “Handle draw” could have been made more specific and potentially broken down further which might have helped the team predict the story points more accurately.

## 6 Burn Down Chart

The estimated vs actual story points were monitored throughout both sprints with the use of a Burn Down Chart. The Burn Down Chart is shown in Figure 1 and clearly illustrates how the actual project progression compared with the predicted user stories. The actual Burn Down trend line goes below zero as not all of the story points were completed within the second sprint. The remaining stories were completed in the next couple of days as they required more time for debugging than the team had predicted.

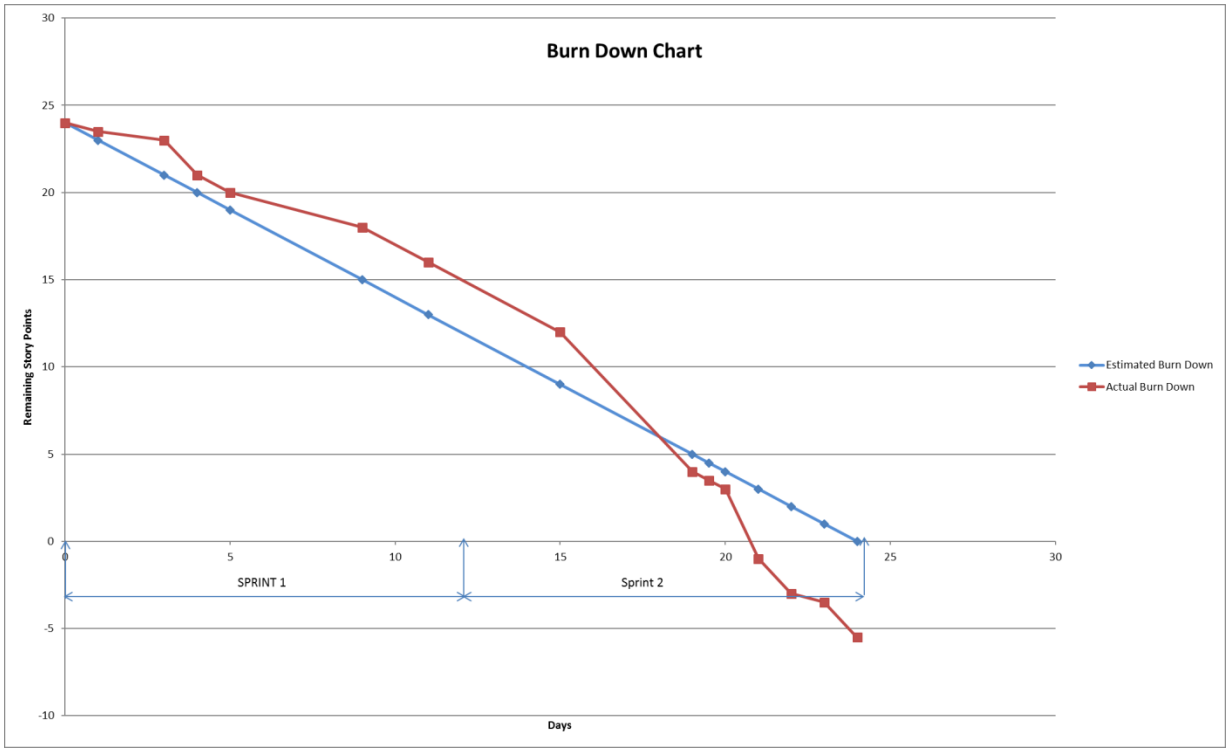



**Figure 1: Burn Down Chart**

## 7 Assumptions

The following assumptions have been made during the design and implementation of the Top Trump's game:

1. A draw won't continue until the point where there are only cards in the communal pile
2. If a draw occurs and the winner of the previous round is not part of the draw we have assumed that the last player (e.g. player 4 is the last player out of players 3 and 4) in the draw round gets to pick the next category
3. The user will not be able to start a new game during game play
4. When cards are dealt and the number of players is uneven, we have assumed that player 1 will always get the extra card
5. We have assumed that there will always be 5 categories and 40 cards in a deck

6. We have assumed that deck.txt will always be formatted correctly
7. We have assumed that the category descriptions will not require a text width greater than 15 and that the card description will not require a text width greater than 30
8. When players are not included in a game, they are given a score of 0 in the database for the number of rounds won

## 8 Testing

The information displayed in Table 7 includes the tests that were carried out and the input data used to carry out the tests. The image column contains a reference to a screen shot in Appendix B which evidences that the test has been completed. The results column is used to mark whether the test was passed. The testing aims to assess every aspect of the functionality of the game.

**Table 7: Test Details**

| Test Number | Test | Input | Result |
|---|---|---|---|
| 1 | Check the deck file is read in and card objects created correctly | Sample input file and new test input file | passed |
| 2 | Check the deck is shuffled | Test for multiple games | passed |
| 3 | Check the cards have been dealt to each player | Test for all possible number of players | passed |
| 4 | Check that the player who starts is randomly selected | Test for all possible number of players | passed |
| 5 | Check that comp player correctly selects the highest category on their top card | Test for multiple cards | passed |
| 6 | Check that user selection of category is registered correctly | Test for each category | passed |
| 7 | Check that the winner of each round is correctly calculated | Test for multiple games | passed |
| 8 | Check that the cards in round are players top cards | Test for multiple rounds | passed |
| 9 | Check that the current card displayed in the GUI correctly updates throughout game play | Test for multiple rounds | passed |
| 10 | Check that for the category | Test for multiple | passed |

| | | | |
|---|---|---|---|
| | selected the corresponding value from each players card is displayed on the GUI | rounds | |
| 11 | Check that the category selected by the current player is displayed on the GUI | Test for multiple rounds | passed |
| 12 | Check that the user is unable to select a radio button unless it is their turn | Test for multiple computer turns | passed |
| 13 | Check that the user is unable to select view statistics or new game during game play | Test for multiple rounds | passed |
| 14 | Check that cards are correctly distributed to the winners hand and the corresponding GUI information is updated | Test for multiple Rounds | passed |
| 15 | Check the winner of the round gets the next choice of the category | Test for multiple rounds | passed |
| 16 | Check that the winner of the round is clearly displayed on the GUI | Test for multiple rounds | passed |
| 17 | Ensure that when a draw occurs that all players in the rounds cards are placed in the communal pile and this is displayed in the GUI | Test for multiple games | passed |
| 18 | Check that only the players who drew are in the consecutive round and the winner gets the cards from the other players and those in the communal pile | Test for multiple games | passed |
| 19 | Check the scenario where multiple consecutive draws occur | Test for multiple games<br>Test using draw test pack | passed |
| 20 | Check that when a player runs out of cards they stay out of the game | Test for multiple games | passed |
| 21 | Check that the player with all the cards in the deck wins the game | Test for multiple games | passed |
| 22 | Check that the user has an option to save the game data or exit or play again | Test for multiple games | passed |

| | | | |
|---|---|---|---|
| 23 | Check that if the user chooses to save the data that this information is correctly stored in the database | Test for multiple games | passed |
| 24 | Check that the user can view the overall game statistics as specified in the brief | Test for multiple games | passed |
| 25 | Check that the overall statistics can be output to a text file | Test for multiple games | passed |
| 26 | At the end of a game check that the player can start a new game | Test for multiple games | passed |
| 27 | Check that the specified information for testing is printed to the console at the correct points during game play | Test for multiple games | passed |
| 28 | Check all database tables | Test for multiple games | |

* The draw test pack has cards with lots of equal values to ensure frequent draws occur.

## Appendix A: User Stories

User story points were recorded on sticky notes during the brainstorming session. They have therefore been added to the user story cards in the blue circles.

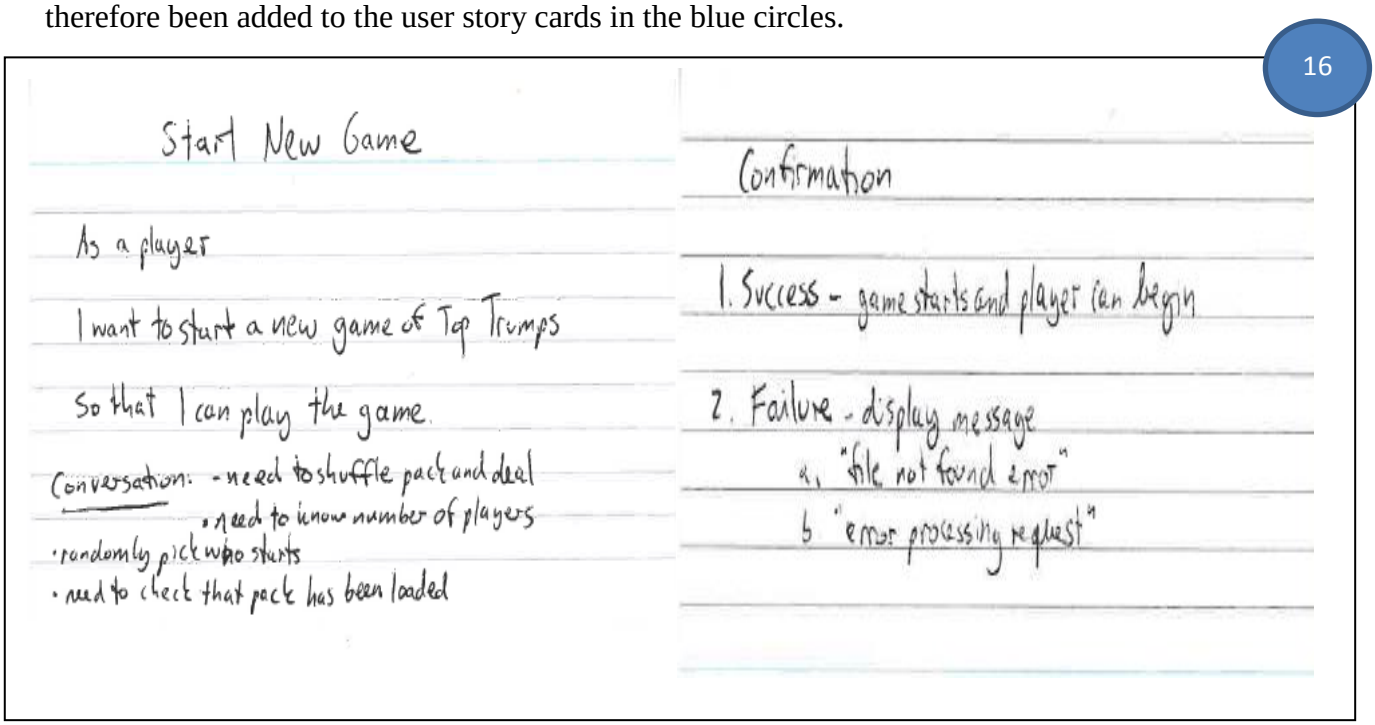



Figure 2: User story 1

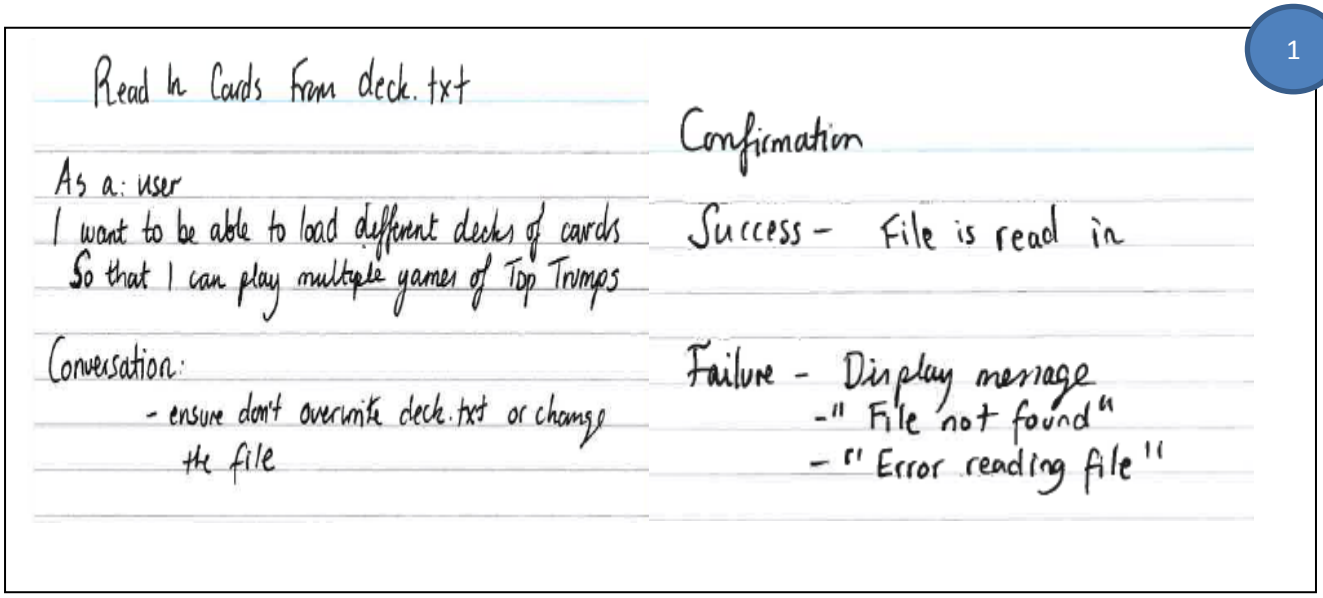



Figure 3: User Story 1.1

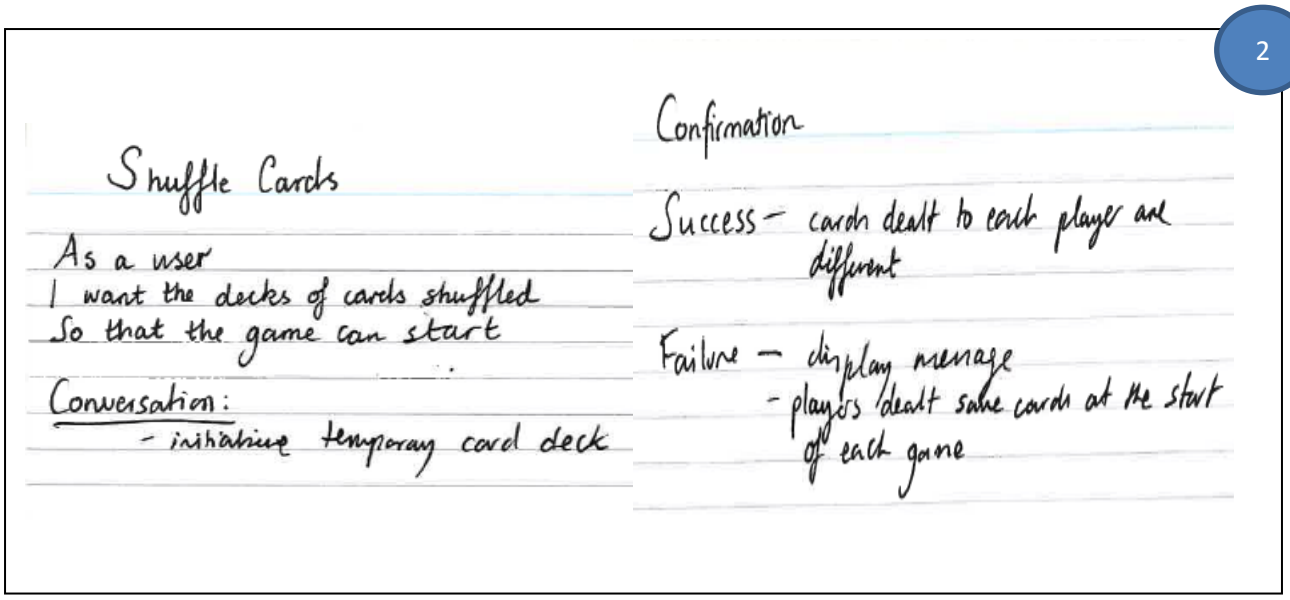



Figure 4: User Story 1.2

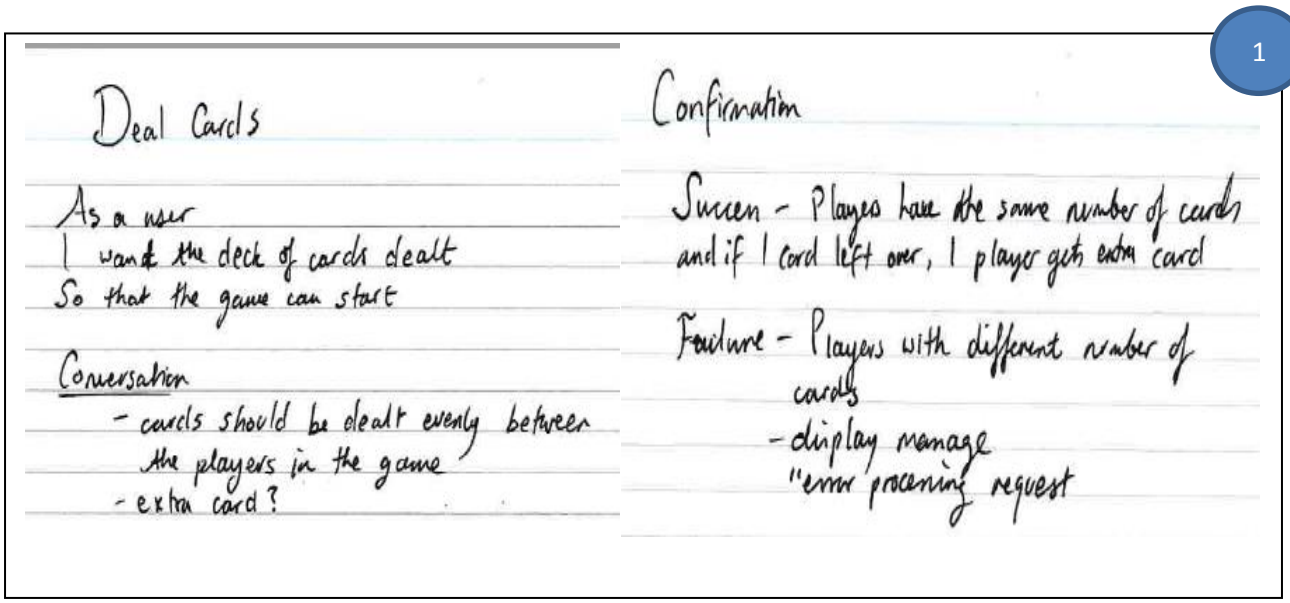



Figure 5: User Story 1.3

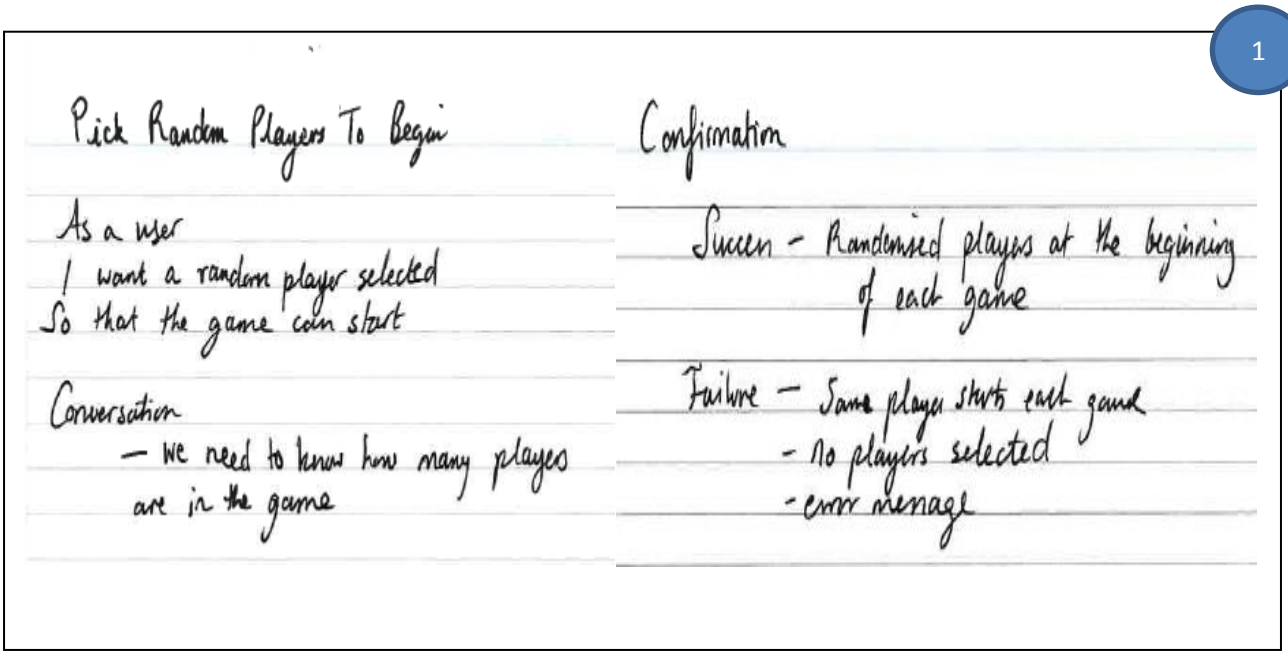



Figure 6: User Story 1.4

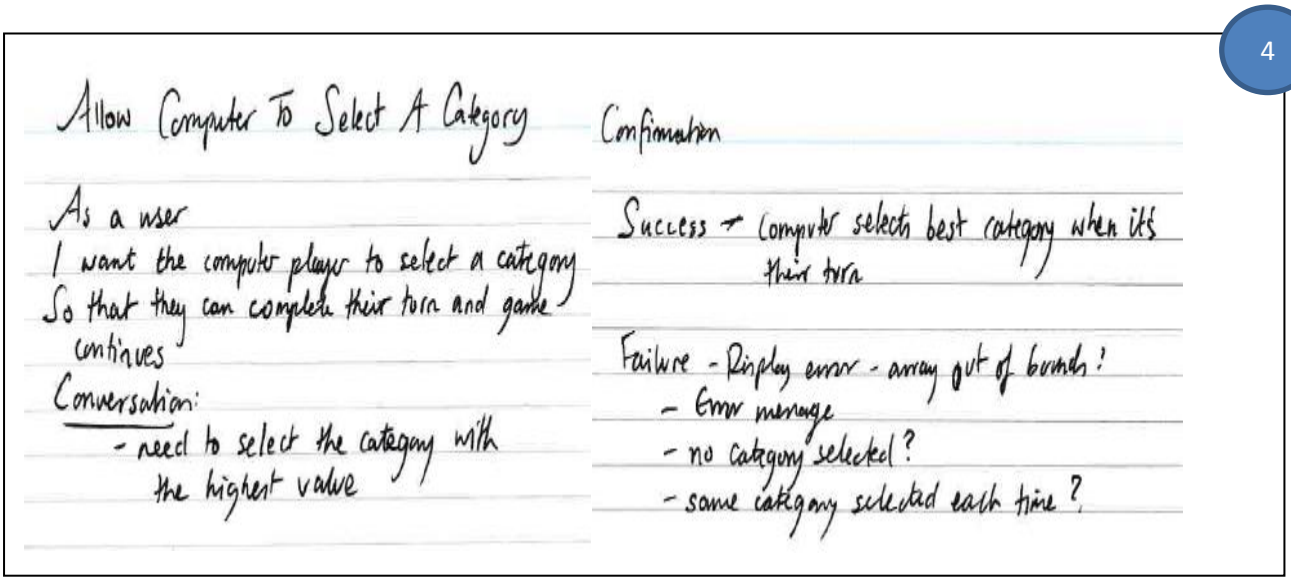



Figure 7: User Story 1.5

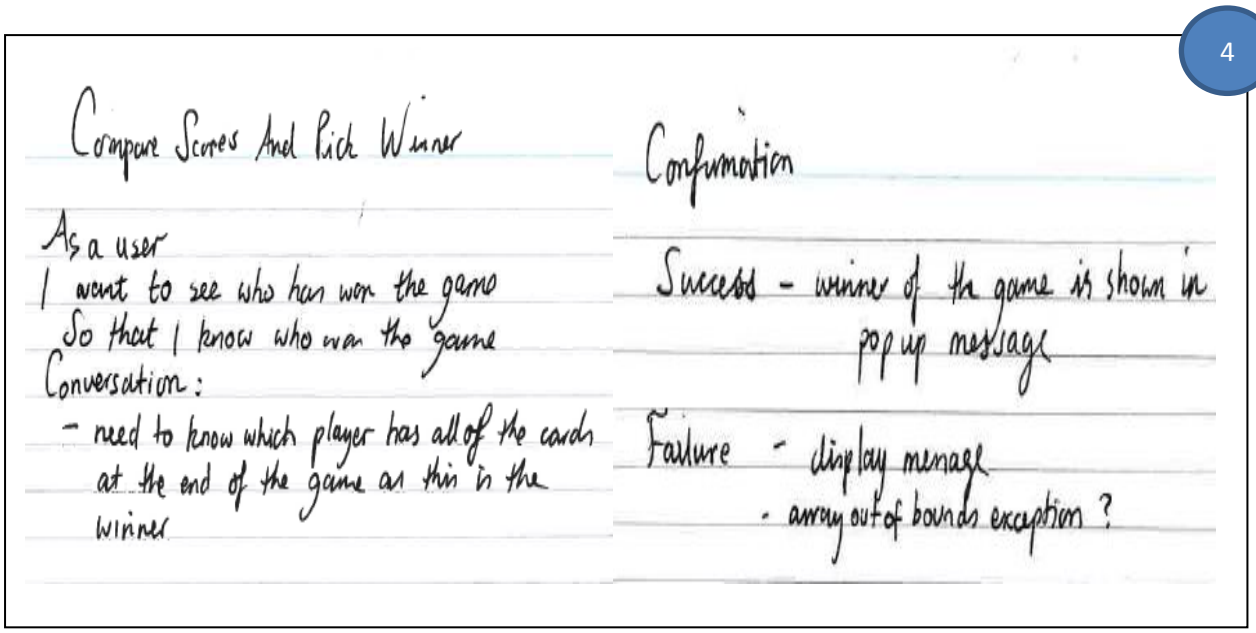



Figure 8: User Story 1.6

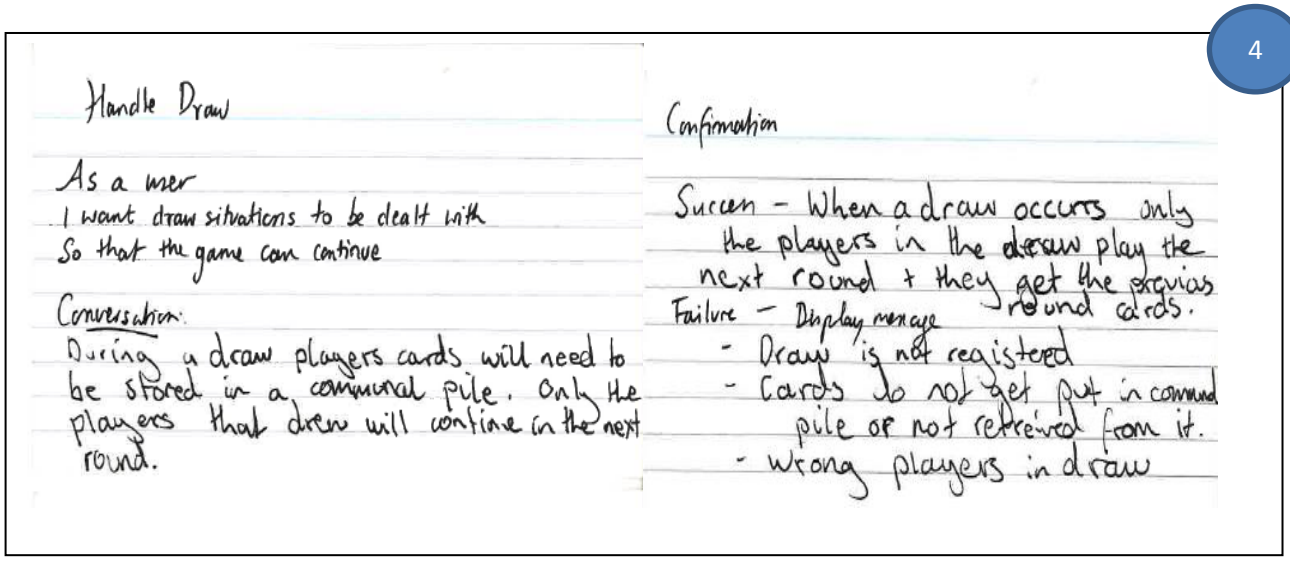



Figure 9: User Story 1.7

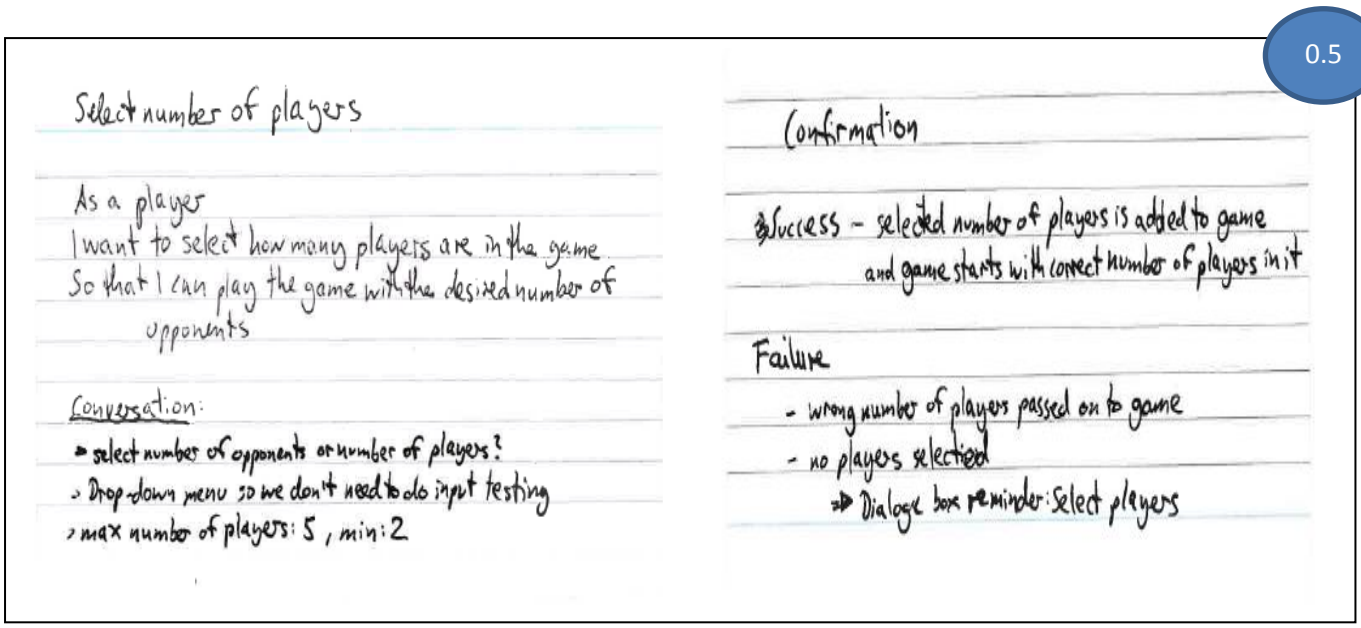



Figure 10: User story 2

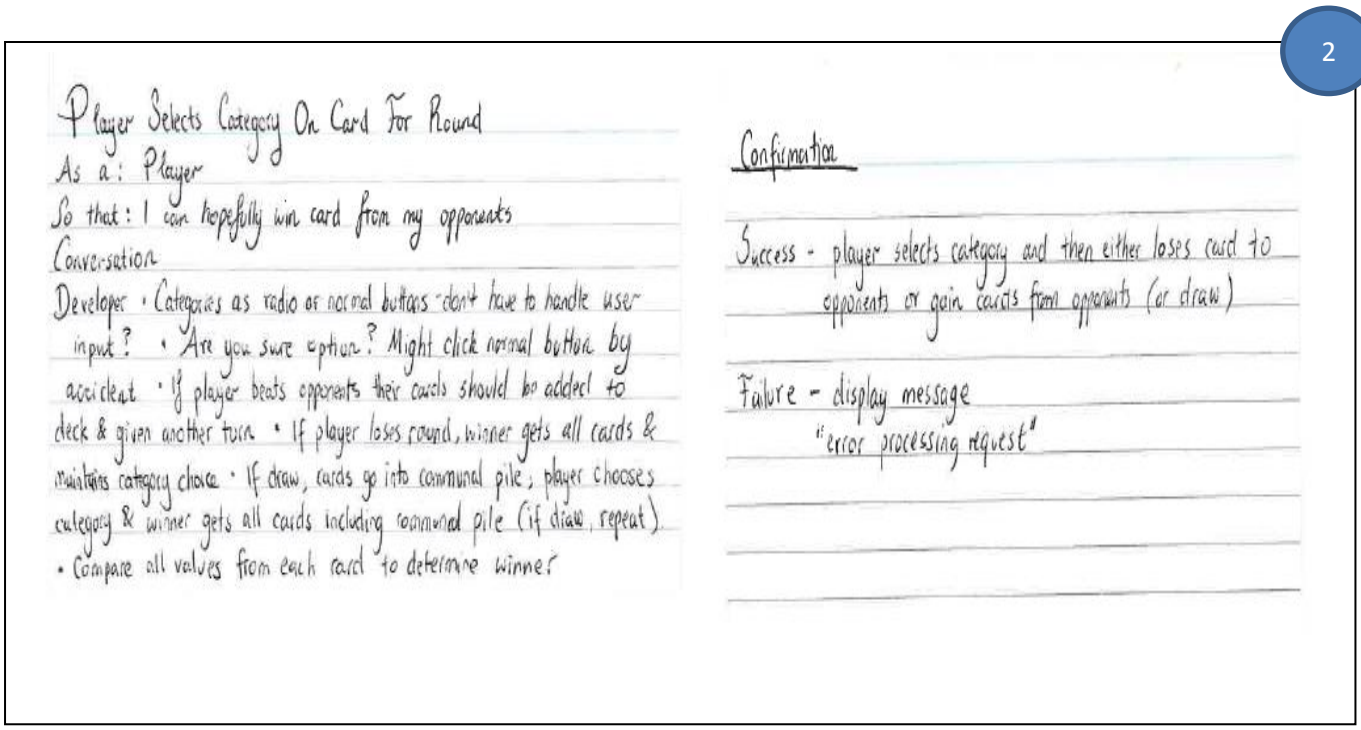



Figure 11: User Story 3

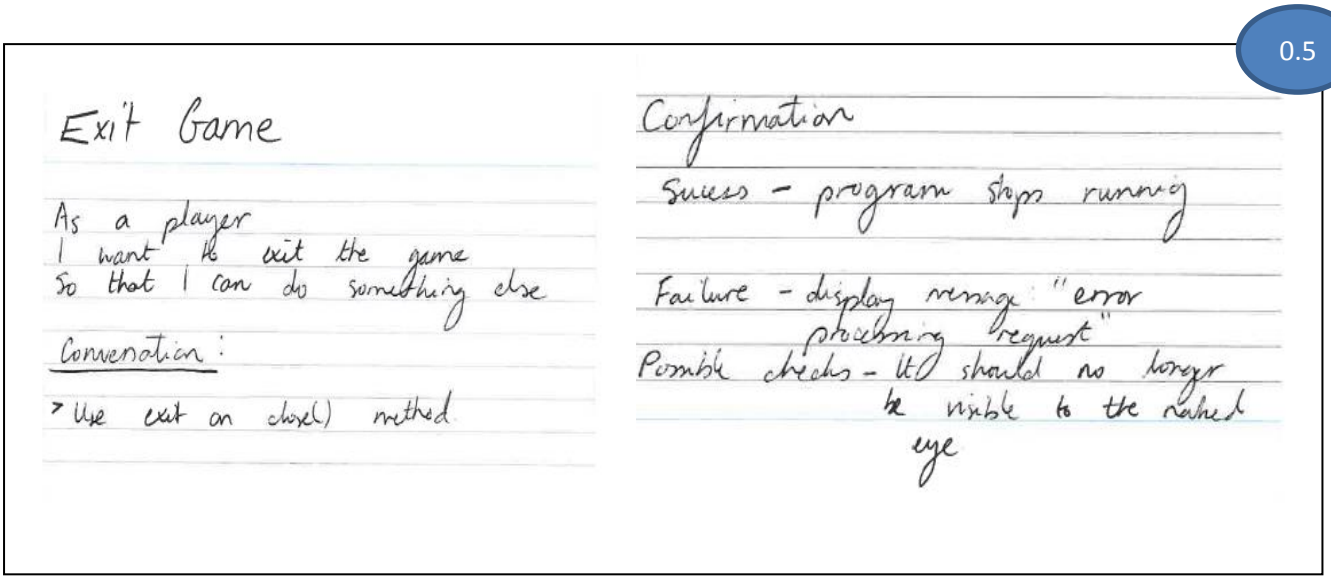



Figure 12: User Story 4

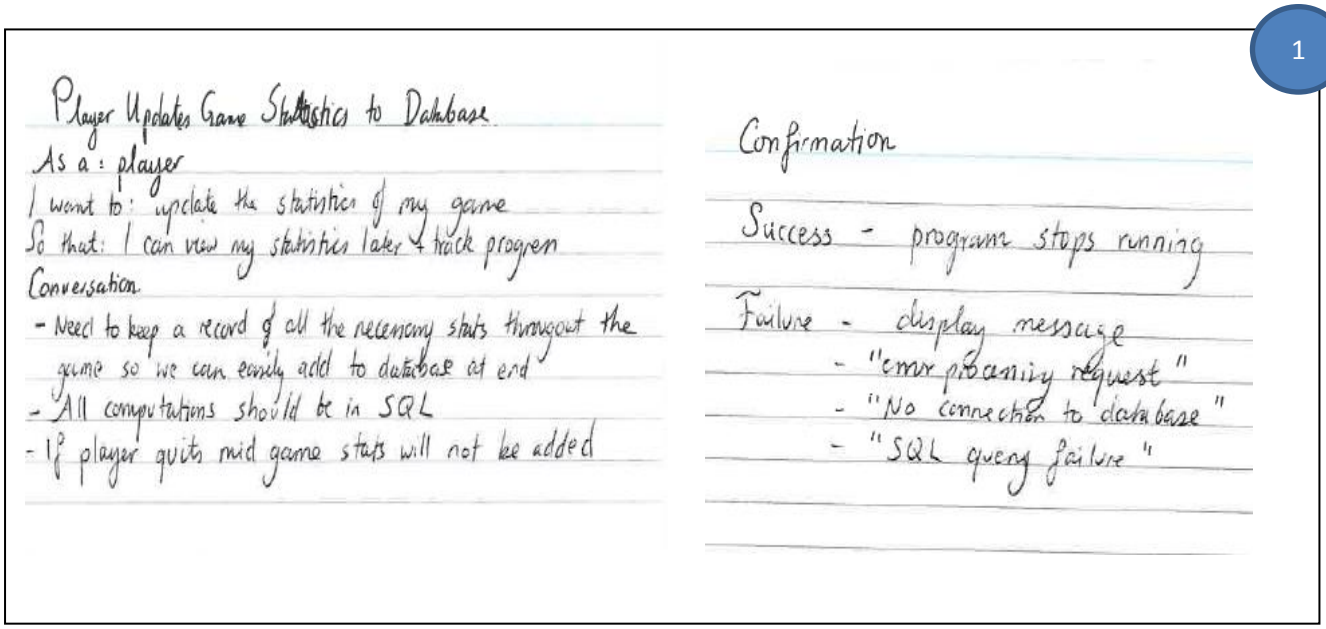



Figure 13: User Story 5

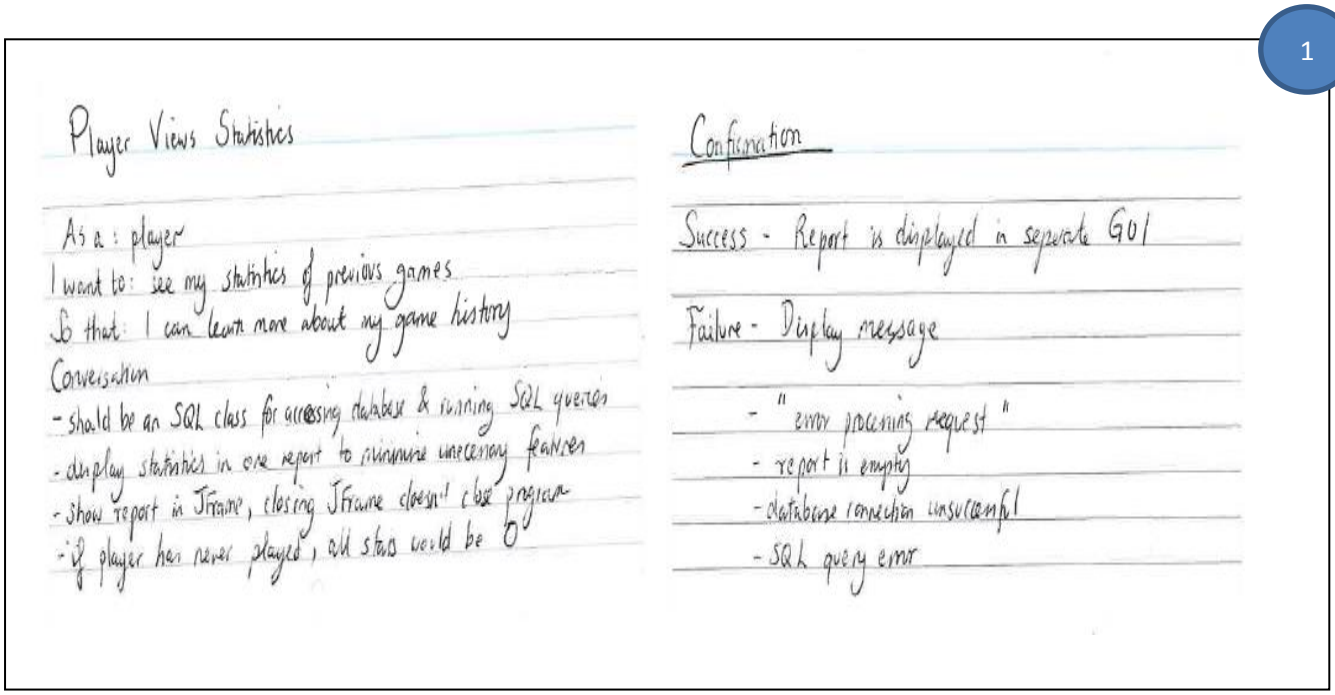



Figure 14: User Story 6

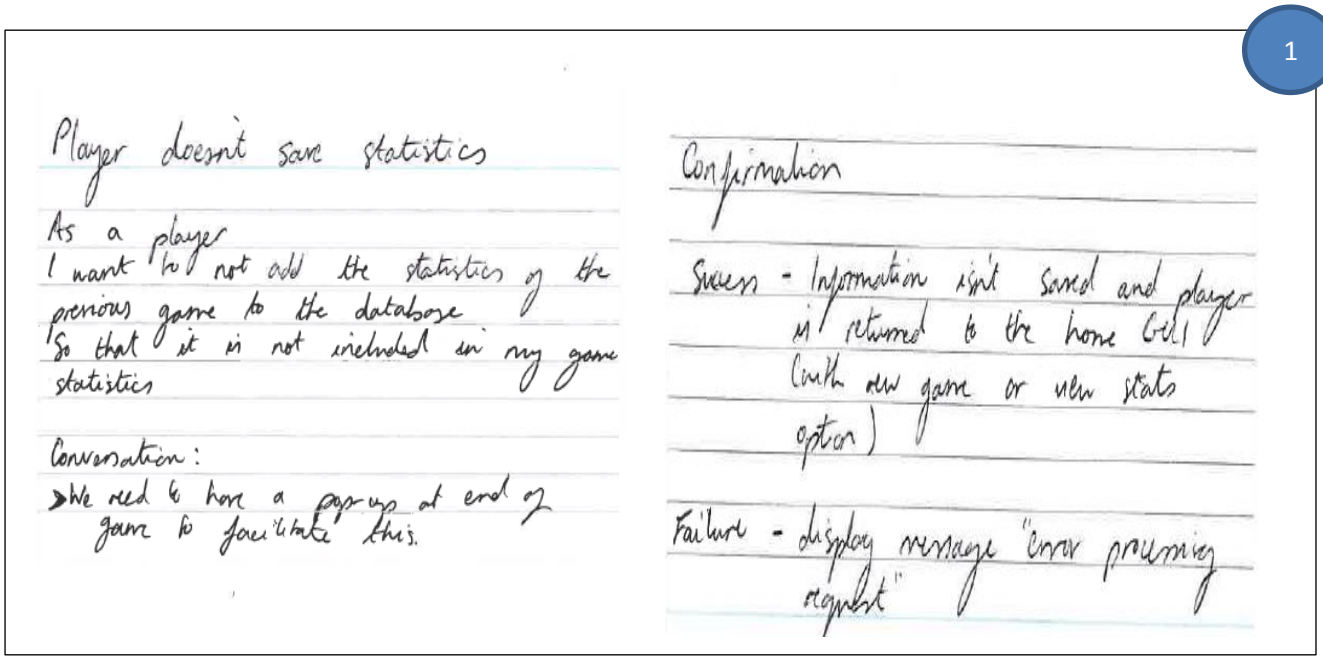



Figure 15: User Story 7

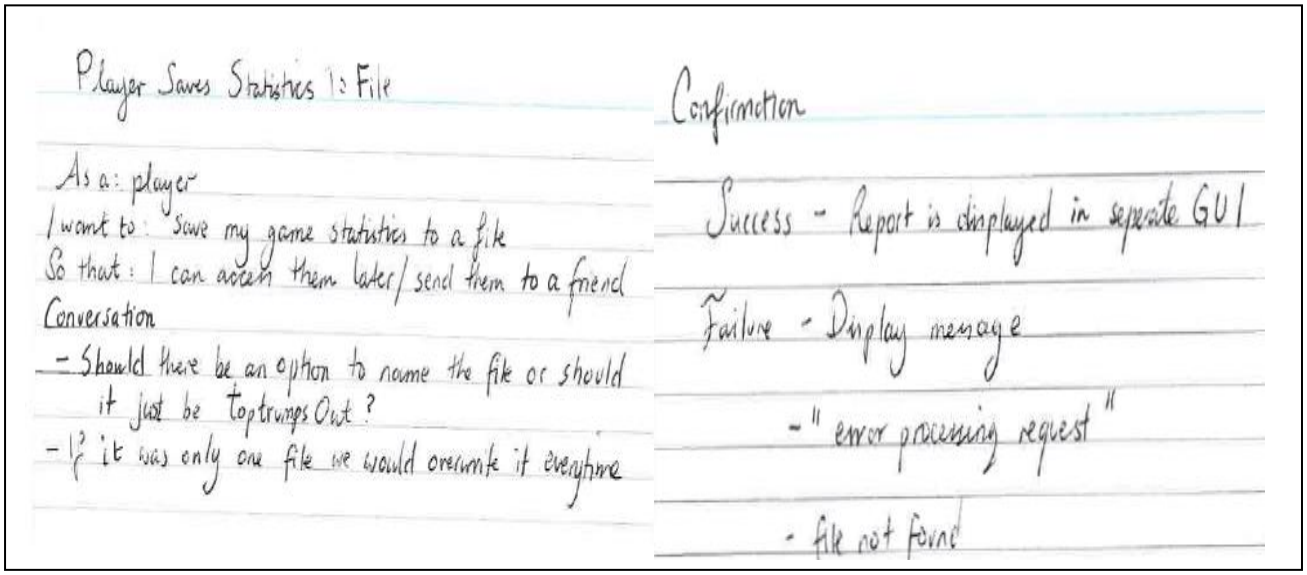
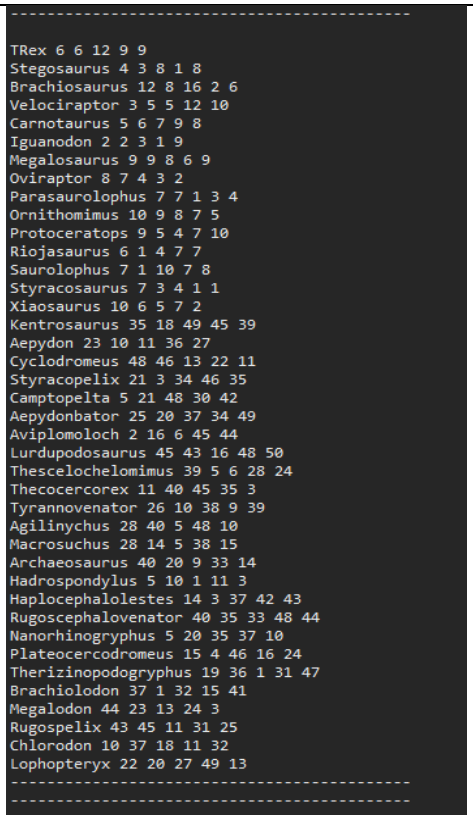
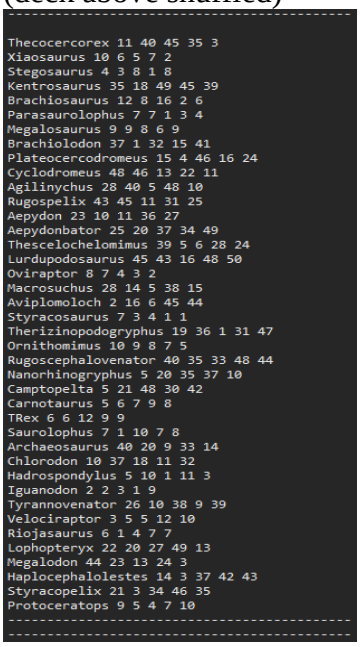



Figure 16: User Story 8

# 7 Appendix B: Testing Screenshots

| Test Number | Image |
|---|---|
| 1 |  |
| 2 | (deck above shuffled)  |

3

2 Players




3 Players




4 Players

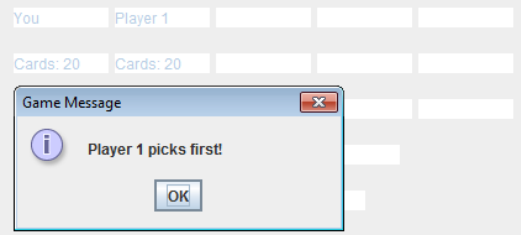



5 Players

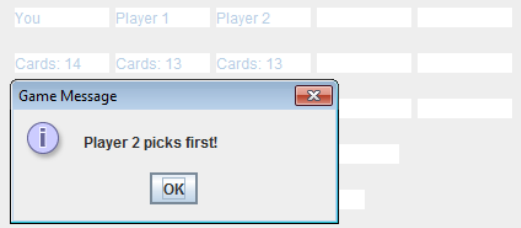



4

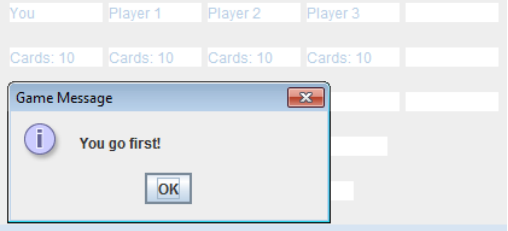
2 Players




3 Players



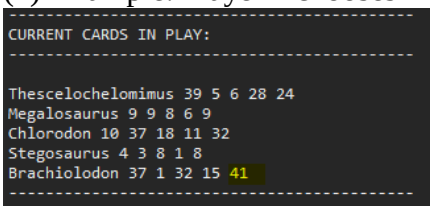

4 Players

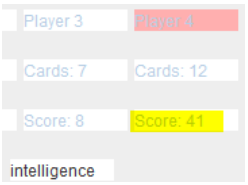


5 Players

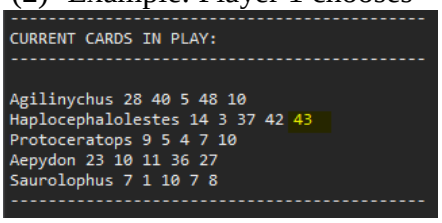

5

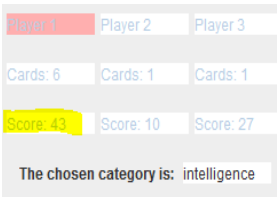
(1) Example: Player 2 chooses



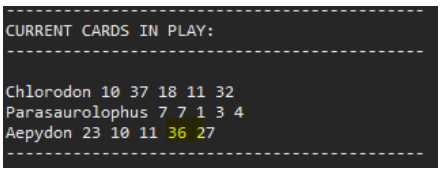



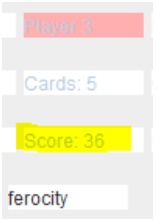

(2) Example: Player 1 chooses



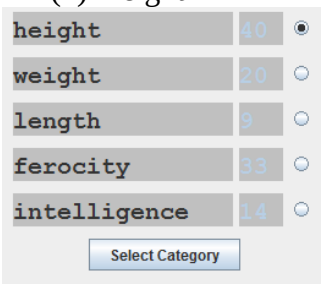

(3) Example: Player 3 chooses

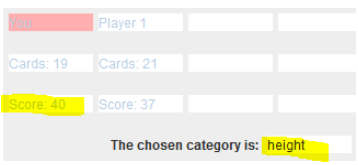


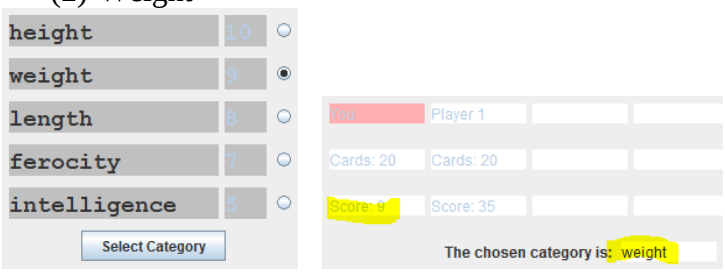

6

(1) Height

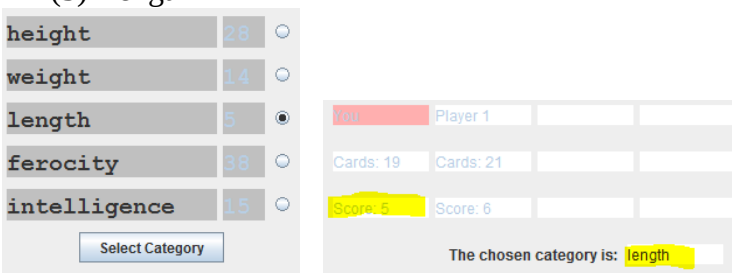


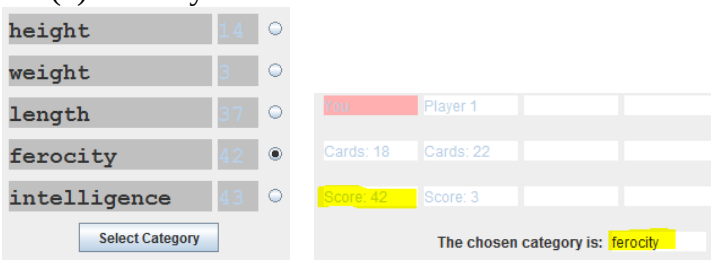
(2) Weight



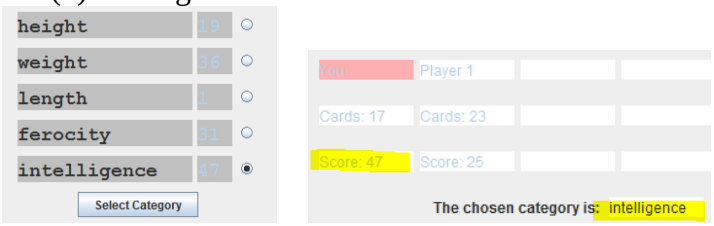

(3) Length




(4) Ferocity




(5) Intelligence




7 Example 1:

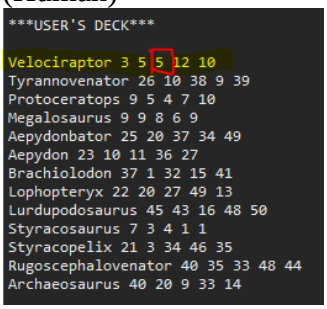

Example 2:


Example 3:

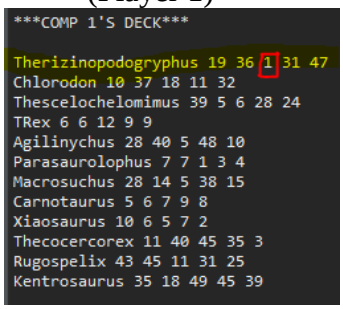

8

Example 1:

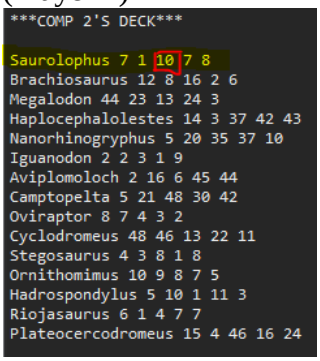
(Human)



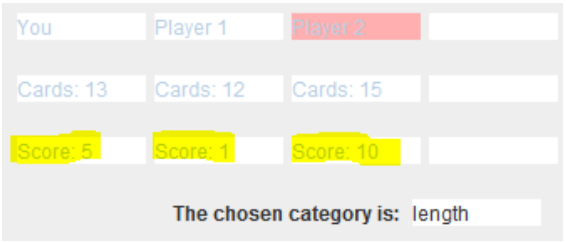

(Player 1)




(Player 2)

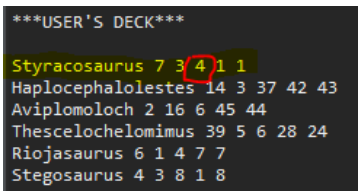



(compare scores)

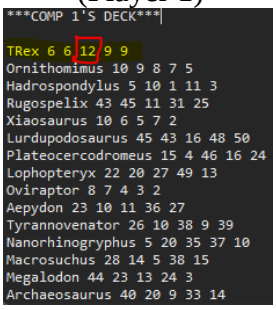



Example 2:

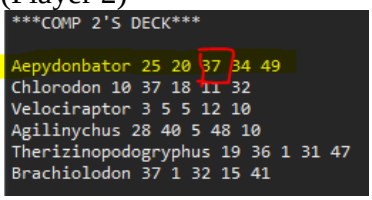
(Human)



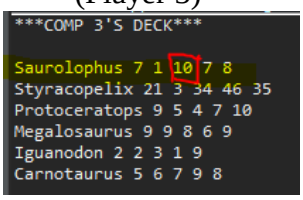

(Player 1)



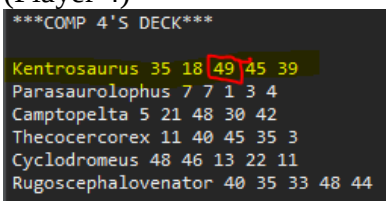

(Player 2)



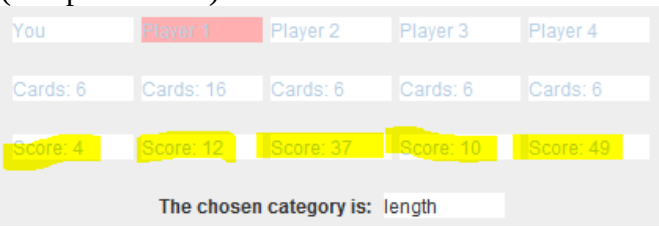

(Player 3)



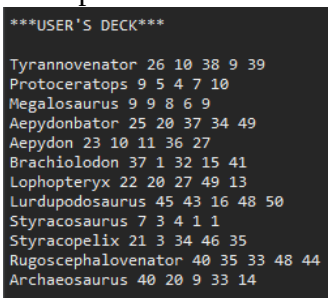

(Player 4)



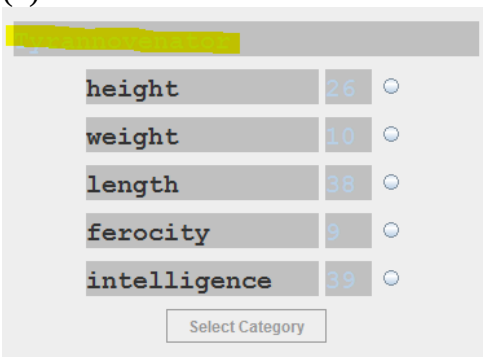

(compare scores)



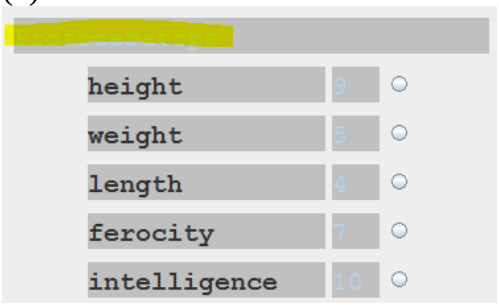

9 Example 1:



Expect GUI to display cards in order above (sample of 2)
(1)




(2)

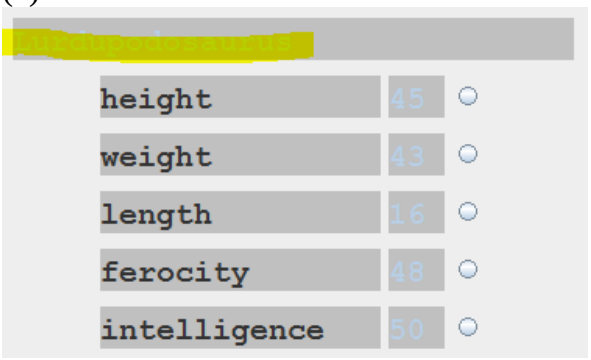



Example 2:

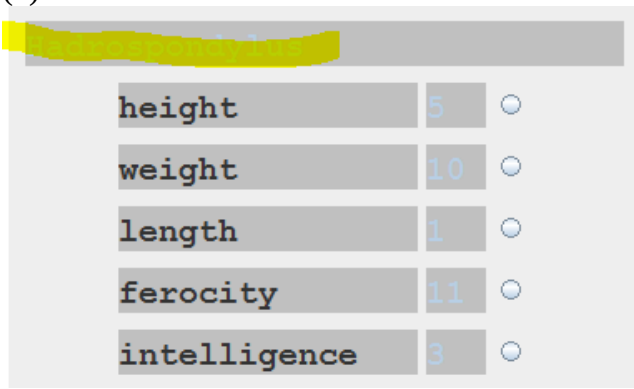
```
***USER'S DECK***

Lurdupodosaurus 45 43 16 48 50
Hadrospondylus 5 10 1 11 3
Iguanodon 2 2 3 1 9
Therizinopodogryphus 19 36 1 31 47
Chlorodon 10 37 18 11 32
Rugospelix 43 45 11 31 25
Protoceratops 9 5 4 7 10
Tyrannovenator 26 10 38 9 39
```

Expect GUI to display cards in order above (sample of 2)

(1)

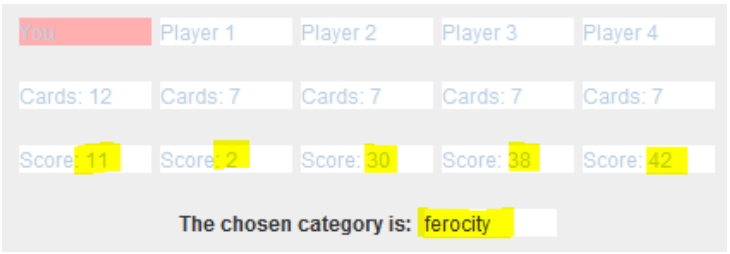


(2)




10 Example 1:

```
-------------------------------------------
CURRENT CARDS IN PLAY:
-------------------------------------------

Hadrospondylus 5 10 1 11 3
Brachiosaurus 12 8 16 2 6
Camptopelta 5 21 48 30 42
Macrosuchus 28 14 5 38 15
Haplocephalolestes 14 3 37 42 43
-------------------------------------------
```

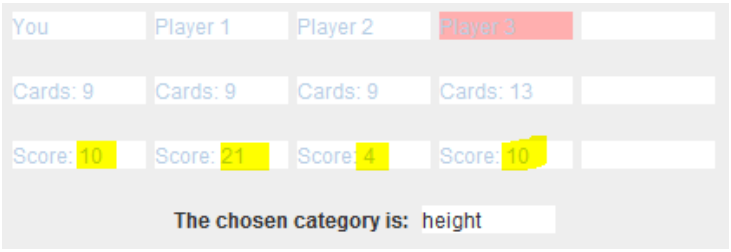



Example 2:

```
-------------------------------------------
CURRENT CARDS IN PLAY:
-------------------------------------------

Chlorodon 10 37 18 11 32
Thescelochelomimus 39 5 6 28 24
Megalosaurus 9 9 8 6 9
Cyclodromeus 48 46 13 22 11
Kentrosaurus 35 18 49 45 39
-------------------------------------------
```

Example 3:

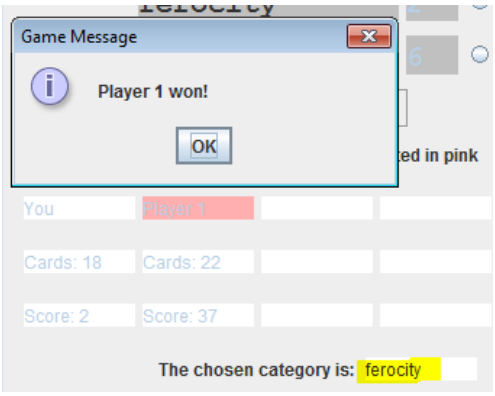
```
-----------------------------------------
CURRENT CARDS IN PLAY:
-----------------------------------------

Chlorodon 10 37 18 11 32
Styracopelix 21 3 34 46 35
Stegosaurus 4 3 8 1 8
Xiaosaurus 10 6 5 7 2
-----------------------------------------
```



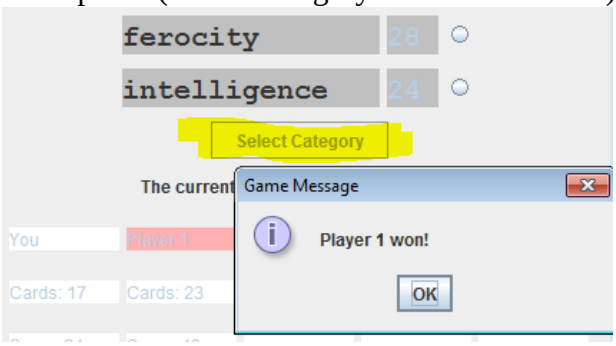

11 Example 1:


Example 2:

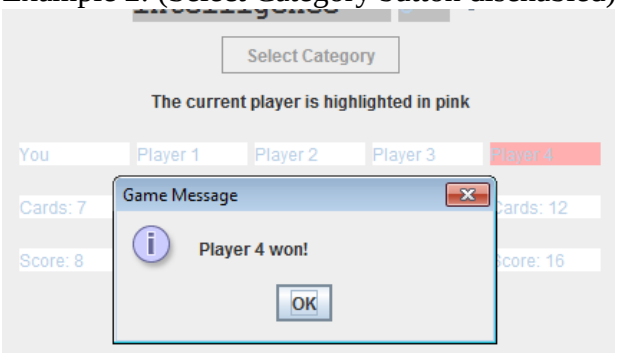
Example 3:




12 Example 1: (Select Category button disenabled)




Example 2: (Select Category button disenabled)

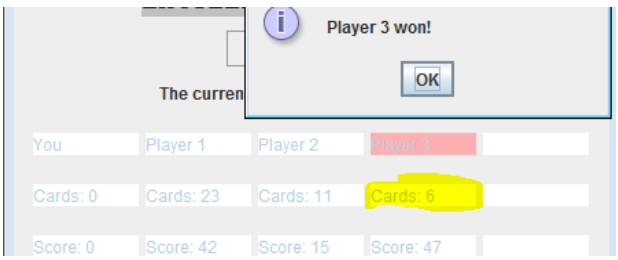



Example 3: (Select Category button disenabled)

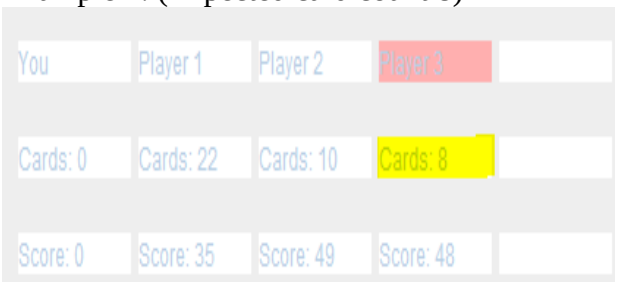
13 Example 1: (top row buttons disenabled)


Example 2: (top row buttons disenabled)

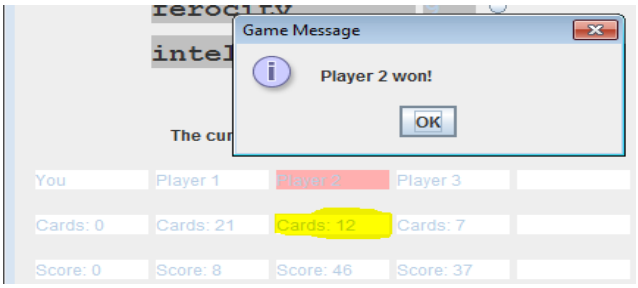

14 Example 1: (Card count 6)



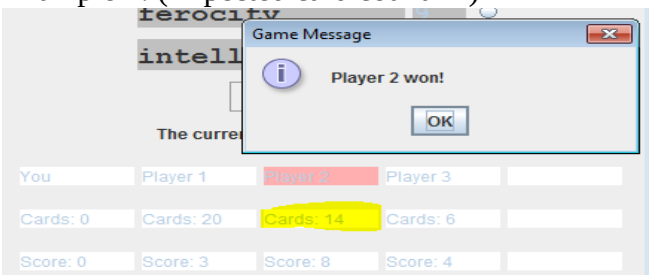
Example 1: (Expected card count 8)




Example 2: (Card count 12)

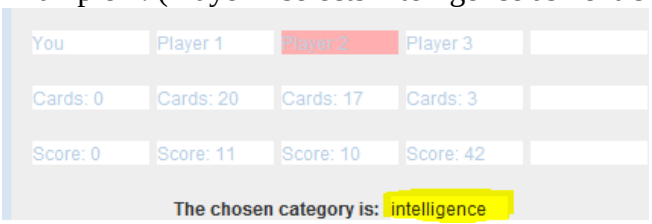



Example 2: (Expected card count 14)

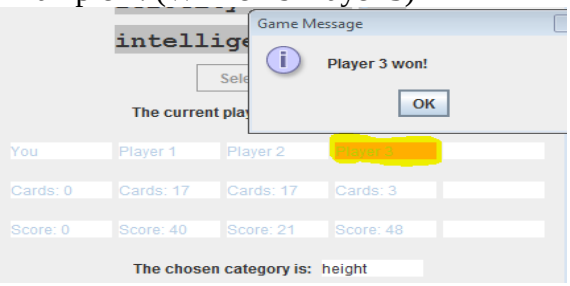



15 Example 1: (Winner is Player 2)

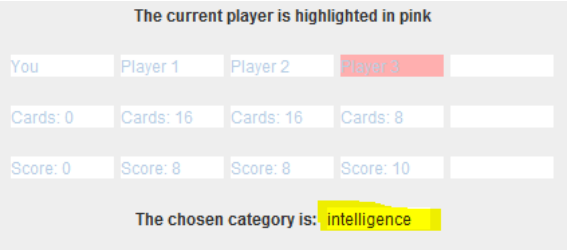

Example 1: (Player 2 selects intelligence as next choice)



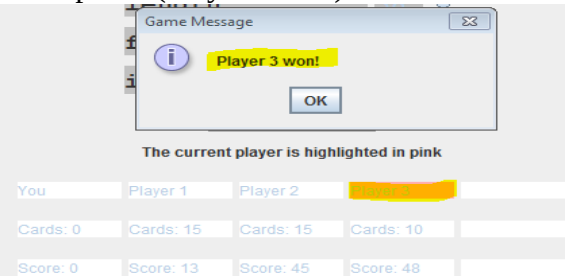
Example 2: (Winner is Player 3)



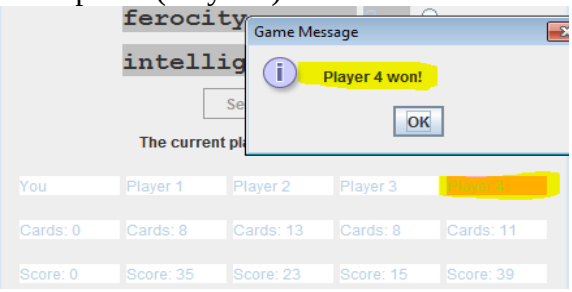

Example 3: (Player 3 selects intelligence as next choice)



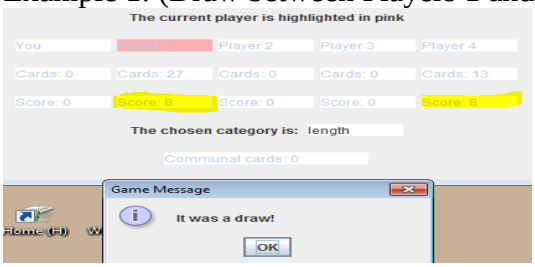

16 Example 1: (Player 3 won)



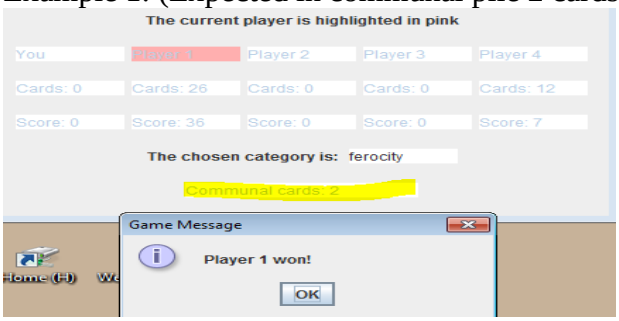

Example 2: (Player 4)




17 Example 1: (Draw between Players 1 and 4)

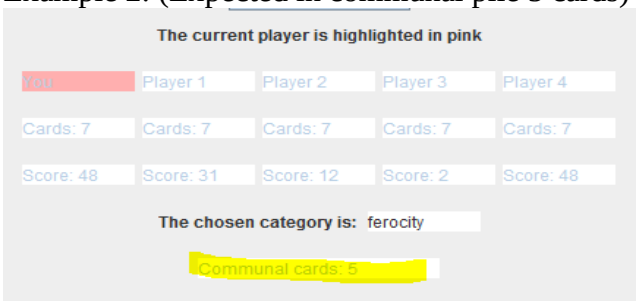


Example 1: (Expected in communal pile 2 cards)

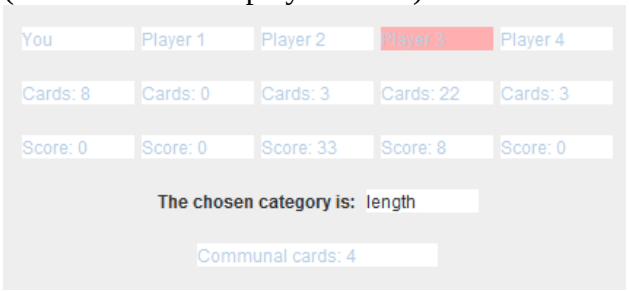



Example 2: (Draw between User and Player 4)

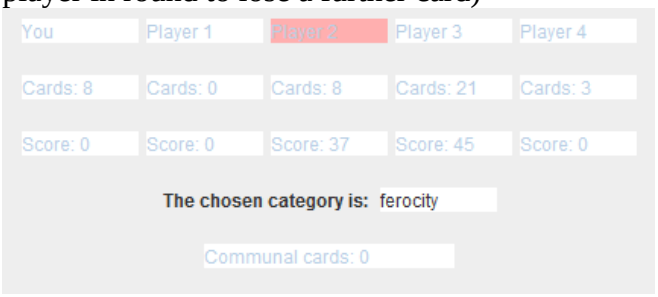

Example 2: (Expected in communal pile 5 cards)



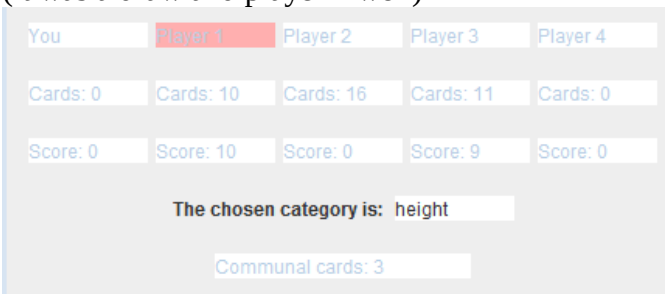

18 Example 1:
(it was a draw and player 2 won)



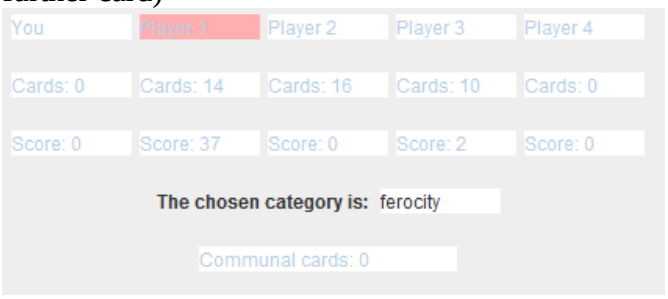
(expect communal pile to clear, player 2's deck to increase by 5 cards, and other player in round to lose a further card)




Example 2:
(it was a draw and player 1 won)

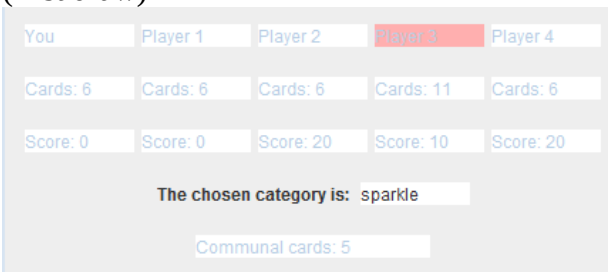



(expect communal cards to clear, player 1 to gain 4 cards, another player lose a further card)

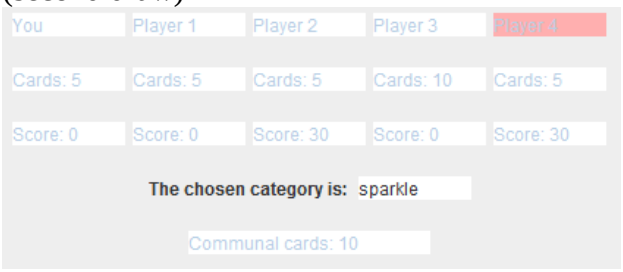



19 (multiple consecutive draws)

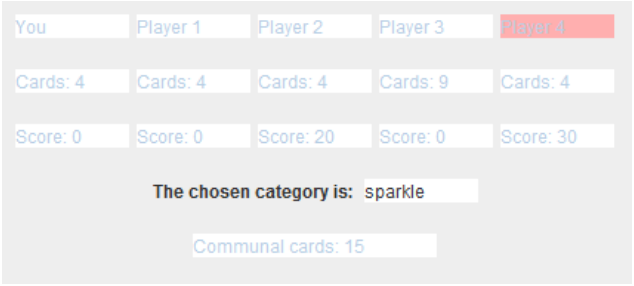
(first draw)



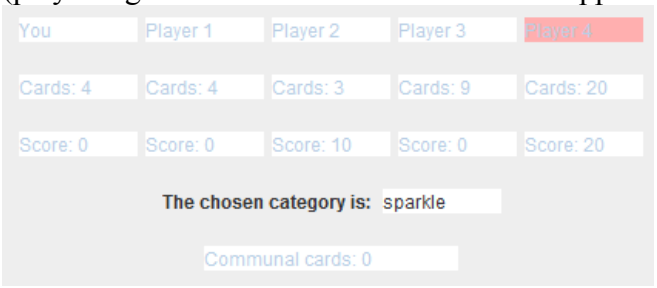
(second draw)



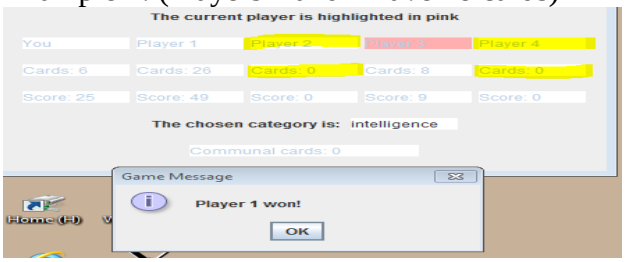

(player 4 wins)



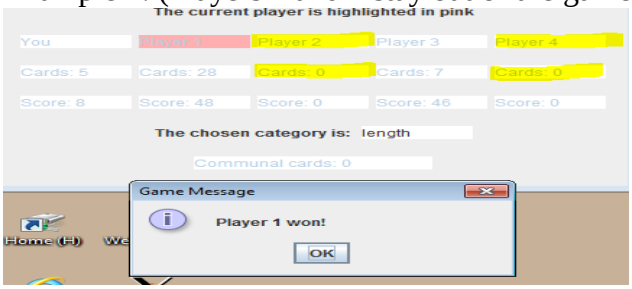

(player 4 gets all the communal cards + the opponent's card)



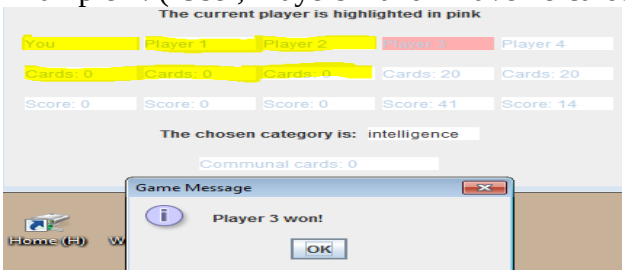

20 Example 1: (Players 1 and 4 have no cards)



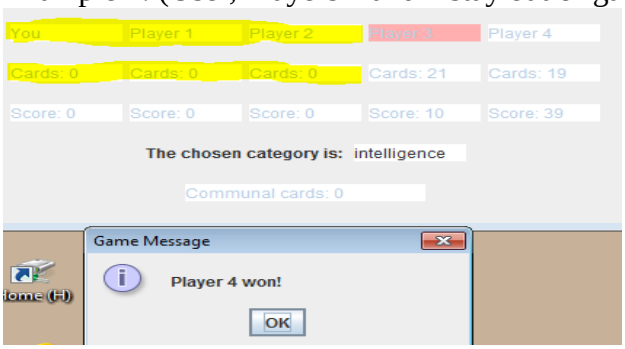

Example 1: (Players 1 and 4 stay out of the game)



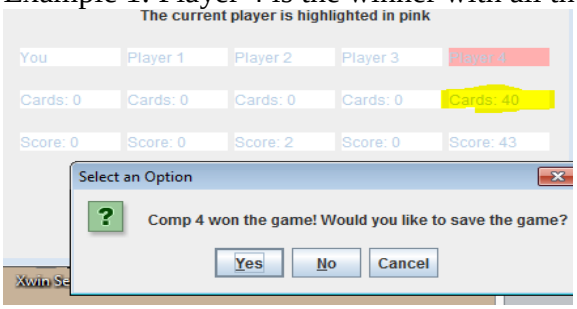
Example 2: ( User, Players 1 and 4 have no cards)



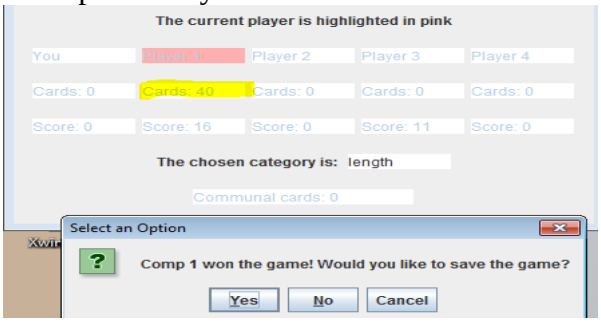

Example 2: (User, Players 1 and 4 stay out of game)



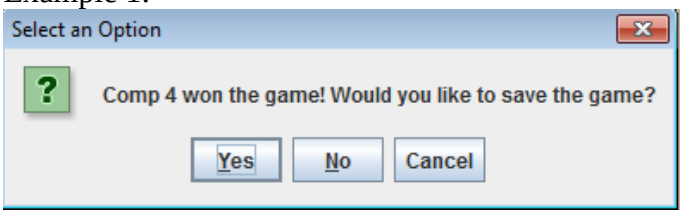

21 Example 1: Player 4 is the winner with all the cards



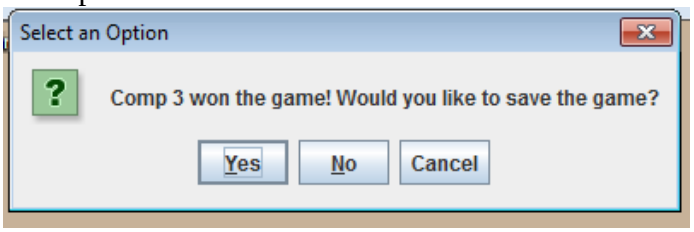

Example 2: Player 1 is the winner with all the cards



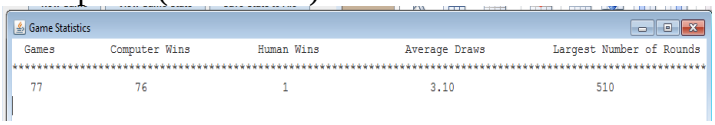
22 Example 1:



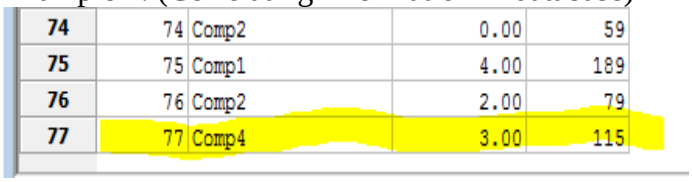

Example 2:




23 Example 1: (Game stats)

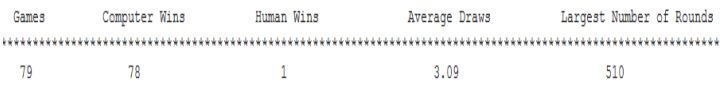



Example 1: (Correlating information in database)

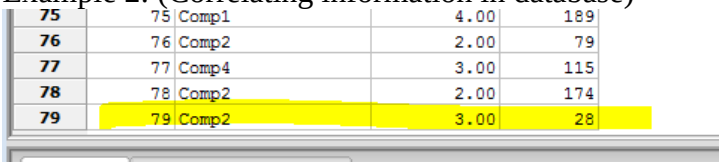


| | | | | |
|---|---|---|---|---|
| 74 | 74 | Comp2 | 0.00 | 59 |
| 75 | 75 | Comp1 | 4.00 | 189 |
| 76 | 76 | Comp2 | 2.00 | 79 |
| 77 | 77 | Comp4 | 3.00 | 115 |

Example 2: (Game stats)

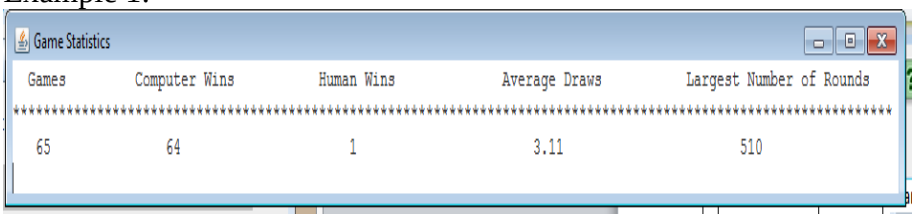



Example 2: (Correlating information in database)

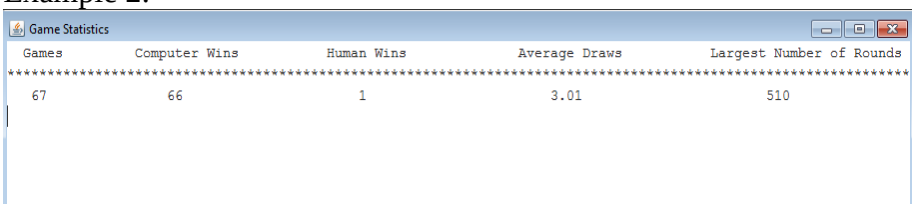


| | | | | |
|---|---|---|---|---|
| 75 | 75 | Comp1 | 4.00 | 189 |
| 76 | 76 | Comp2 | 2.00 | 79 |
| 77 | 77 | Comp4 | 3.00 | 115 |
| 78 | 78 | Comp2 | 2.00 | 174 |
| 79 | 79 | Comp2 | 3.00 | 28 |

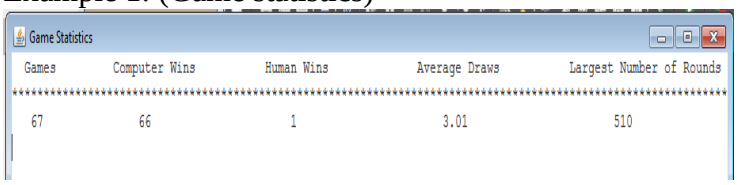
24 Example 1:



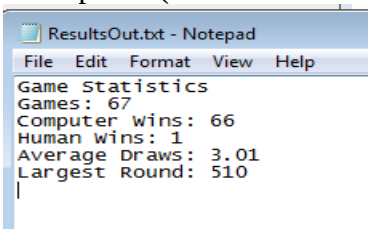
Example 2:



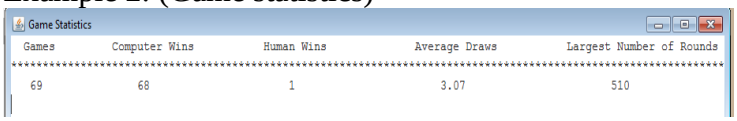

25 Example 1: (Game statistics)



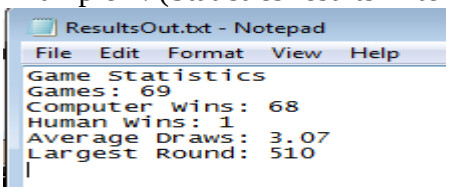

Example 1: (Statistics results in text file)




Example 2: (Game statistics)

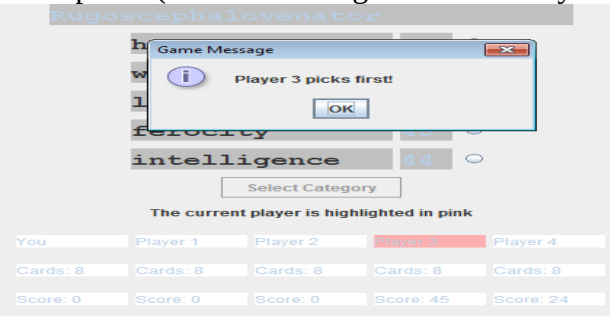



Example 2: (Statistics results in text file)




26 Example 1: (End of game)

Example 1: (Start of a new game where Player 3 goes first)




Example 2: (End of game)


Example 2: (Start of a new game where user goes first)


27

Example 1: (Game in play – cards in players hand)

```
***COMP 1'S DECK***

Styracosaurus 7 3 4 1 1

***COMP 2'S DECK***

Thescelochelomimus 39 5 6 28 24
Agilinychus 28 40 5 48 10
Stegosaurus 4 3 8 1 8
Saurolophus 7 1 10 7 8
TRex 6 6 12 9 9
Brachiolodon 37 1 32 15 41
Archaeosaurus 40 20 9 33 14
Carnotaurus 5 6 7 9 8
```

Example 1: (End of game)

```
----------------------------------------
THE WINNER OF THE GAME IS: PLAYER 2
Number of games: 82
Number of games: 82
Number of games: 83
Number of games: 83
Avg from Database: 3.04
Avg rounded: 3.04
Number of games: 83
Avg from Database: 3.04
Avg rounded: 3.04
```

Example 2: (Cards in players hand)

```
***COMP 3'S DECK***

Plateocercodromeus 15 4 46 16 24
Aepydonbator 25 20 37 34 49
Tyrannovenator 26 10 38 9 39
Parasaurolophus 7 7 1 3 4
Nanorhinogryphus 5 20 35 37 10
Agilinychus 28 40 5 48 10
Chlorodon 10 37 18 11 32
Therizinopodogryphus 19 36 1 31 47
Oviraptor 8 7 4 3 2
```

Example 2: (End of game)

```
----------------------------------------
THE WINNER OF THE GAME IS: PLAYER 3
Number of games: 83
Number of games: 83
Number of games: 84
Number of games: 84
Avg from Database: 3.08
Avg rounded: 3.08
```

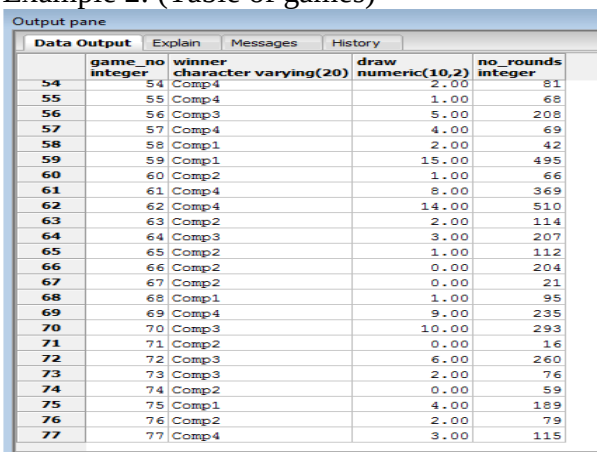
28 Example 1: (Table of games)

Output pane

Data Output | Explain | Messages | History

| | game_no integer | winner character varying(20) | draw numeric(10,2) | no_rounds integer |
|---|---|---|---|---|
| 1 | 1 | Human | 2.00 | 152 |
| 2 | 2 | Comp1 | 1.00 | 84 |
| 3 | 3 | Comp1 | 0.00 | 54 |
| 4 | 4 | Comp4 | 1.00 | 45 |
| 5 | 5 | Comp2 | 1.00 | 66 |
| 6 | 6 | Comp1 | 11.00 | 342 |
| 7 | 7 | Comp3 | 0.00 | 86 |
| 8 | 8 | Comp1 | 1.00 | 179 |
| 9 | 9 | Comp2 | 1.00 | 89 |
| 10 | 10 | Comp2 | 5.00 | 123 |
| 11 | 11 | Comp2 | 2.00 | 108 |
| 12 | 12 | Comp2 | 2.00 | 123 |
| 13 | 13 | Comp1 | 7.00 | 347 |
| 14 | 14 | Comp1 | 5.00 | 106 |

Example 2: (Table of games)

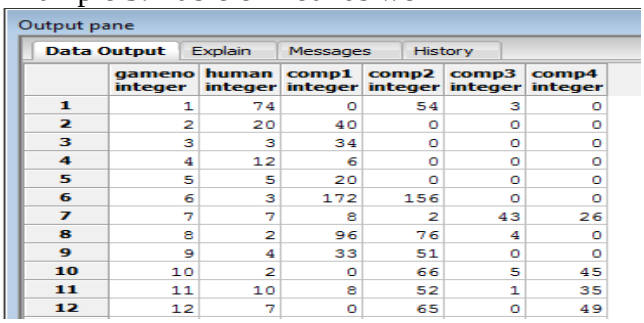


Output pane

Data Output | Explain | Messages | History

| | game_no integer | winner character varying(20) | draw numeric(10,2) | no_rounds integer |
|---|---|---|---|---|
| 54 | 54 | Comp4 | 2.00 | 81 |
| 55 | 55 | Comp4 | 1.00 | 68 |
| 56 | 56 | Comp3 | 5.00 | 208 |
| 57 | 57 | Comp4 | 4.00 | 69 |
| 58 | 58 | Comp1 | 2.00 | 42 |
| 59 | 59 | Comp1 | 15.00 | 495 |
| 60 | 60 | Comp2 | 1.00 | 66 |
| 61 | 61 | Comp4 | 8.00 | 369 |
| 62 | 62 | Comp4 | 14.00 | 510 |
| 63 | 63 | Comp2 | 2.00 | 114 |
| 64 | 64 | Comp3 | 3.00 | 207 |
| 65 | 65 | Comp2 | 1.00 | 112 |
| 66 | 66 | Comp2 | 0.00 | 204 |
| 67 | 67 | Comp2 | 0.00 | 21 |
| 68 | 68 | Comp1 | 1.00 | 95 |
| 69 | 69 | Comp4 | 9.00 | 235 |
| 70 | 70 | Comp3 | 10.00 | 293 |
| 71 | 71 | Comp2 | 0.00 | 16 |
| 72 | 72 | Comp3 | 6.00 | 260 |
| 73 | 73 | Comp3 | 2.00 | 76 |
| 74 | 74 | Comp2 | 0.00 | 59 |
| 75 | 75 | Comp1 | 4.00 | 189 |
| 76 | 76 | Comp2 | 2.00 | 79 |
| 77 | 77 | Comp4 | 3.00 | 115 |

Example 3: Table of Rounds won

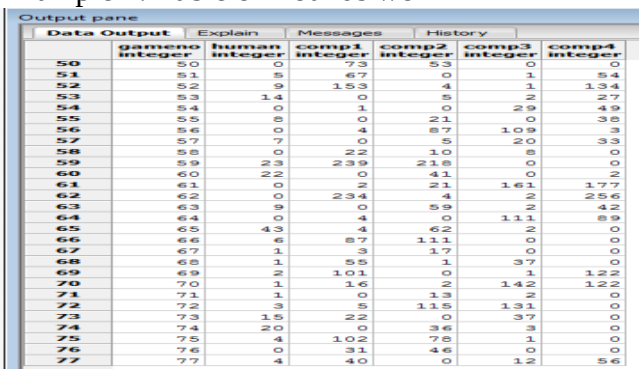


Output pane

Data Output | Explain | Messages | History

| | gameno integer | human integer | comp1 integer | comp2 integer | comp3 integer | comp4 integer |
|---|---|---|---|---|---|---|
| 1 | 1 | 74 | 0 | 54 | 3 | 0 |
| 2 | 2 | 20 | 40 | 0 | 0 | 0 |
| 3 | 3 | 3 | 34 | 0 | 0 | 0 |
| 4 | 4 | 12 | 6 | 0 | 0 | 0 |
| 5 | 5 | 5 | 20 | 0 | 0 | 0 |
| 6 | 6 | 3 | 172 | 156 | 0 | 0 |
| 7 | 7 | 7 | 8 | 2 | 43 | 26 |
| 8 | 8 | 2 | 96 | 76 | 4 | 0 |
| 9 | 9 | 4 | 33 | 51 | 0 | 0 |
| 10 | 10 | 2 | 0 | 66 | 5 | 45 |
| 11 | 11 | 10 | 8 | 52 | 1 | 35 |
| 12 | 12 | 7 | 0 | 65 | 0 | 49 |

Example 4: Table of Rounds won



Output pane

Data Output | Explain | Messages | History

| | gameno integer | human integer | comp1 integer | comp2 integer | comp3 integer | comp4 integer |
|---|---|---|---|---|---|---|
| 50 | 50 | 0 | 73 | 53 | 0 | 0 |
| 51 | 51 | 5 | 67 | 0 | 1 | 54 |
| 52 | 52 | 9 | 153 | 4 | 1 | 134 |
| 53 | 53 | 14 | 0 | 5 | 2 | 27 |
| 54 | 54 | 0 | 1 | 0 | 29 | 49 |
| 55 | 55 | 8 | 0 | 21 | 0 | 38 |
| 56 | 56 | 0 | 4 | 87 | 109 | 3 |
| 57 | 57 | 7 | 0 | 5 | 20 | 33 |
| 58 | 58 | 0 | 22 | 10 | 8 | 0 |
| 59 | 59 | 23 | 239 | 218 | 0 | 0 |
| 60 | 60 | 22 | 0 | 41 | 0 | 2 |
| 61 | 61 | 0 | 2 | 21 | 161 | 177 |
| 62 | 62 | 0 | 234 | 4 | 2 | 256 |
| 63 | 63 | 9 | 0 | 59 | 2 | 42 |
| 64 | 64 | 0 | 4 | 0 | 111 | 89 |
| 65 | 65 | 43 | 4 | 62 | 2 | 0 |
| 66 | 66 | 6 | 87 | 111 | 0 | 0 |
| 67 | 67 | 1 | 3 | 17 | 0 | 0 |
| 68 | 68 | 1 | 55 | 1 | 37 | 0 |
| 69 | 69 | 2 | 101 | 0 | 1 | 122 |
| 70 | 70 | 1 | 16 | 2 | 142 | 122 |
| 71 | 71 | 1 | 0 | 13 | 2 | 0 |
| 72 | 72 | 3 | 5 | 115 | 131 | 0 |
| 73 | 73 | 15 | 22 | 0 | 37 | 0 |
| 74 | 74 | 20 | 0 | 36 | 3 | 0 |
| 75 | 75 | 4 | 102 | 78 | 1 | 0 |
| 76 | 76 | 0 | 31 | 46 | 0 | 0 |
| 77 | 77 | 4 | 40 | 0 | 12 | 56 |

* The draw test pack has cards with lots of equal values to ensure frequent draws occur.