MSc IT+ Masters Team Project

Group Report

12th Feb 2018

Team *Inspired*

Lyle Christine 0912407c

Muhammad Ikhwani 2346771i

Candra Kupang 2311218k

Martina Pironkova 2068174p

Cassie Spalding 2140148s

Table of Contents

Introduction

Program Functionality:

Our program has all the functionality as set out in the requirements in the coursework specification. The online version and the Database and is configured to run on Lab machine with database name, username and password is:
String = "m_17_2140148s, m_17_2140148s, 2140148s"

All the functionality is working:

Command Line Mode:

- This mode is selected `java -jar TopTrumps.jar -c`
- The log file records the statistics of the command line game
- Test Log: logfile.txt if this file already exits the program overwrites the file and prints out the required details as in the spec. `java -jar TopTrumps.jar -c -t`

(see Testing * appendix screen shot 1, 2, 3 & 4)

Online Mode:

- This mode is selected `java -jar TopTrumps.jar -o`
- The game is displayed on several web pages
- Multiple players can play the game concurrently in different browser tabs

(see Testing * appendix screen shot 1, 2, 3 & 4)

In both Modes:

- The user can play against one or more AI opponents, maximum 4 AI players.
- The cards are shuffled randomly and dealt to the players, the first player is selected by random.
- The game loads a deck of cards when the program starts.
- When there is a draw, the same player has another turn, the winner wins the communal pile of cards.
- The results of past games are stored in a database as well as displayed to the user when requested.
- The results statistics can be cleared to reset.
- The jar-file when compiled on the desktop will run a shortcut.

(see Testing * appendix screen shot 1, 2, 3 & 4)

User Stories

- **21 functional story cards (2 of which are out of scope of the requirements)**
- **3 non-functional story cards.**

The following cards are meant to be used by the developers of the software as guidance and reminders. (The appendix contains the initial personas and user stories, mostly these are out of scope, hence why they are in the appendix.)

Table 1 below contains the guidelines for the priority points allocated to each story card. The effort story points allocated for each user story follow the widely used Fibonacci numbers (1,2,3,5,8,13, etc)

Table 1: Priority guidelines

Value	Guide
1-3	Nice to have. Low priority.
4-6	Moderately important. Moderate priority.
7-9	Important. High priority
10	Critical. Must be done. Highest priority

Command line user stories

Story 1

Front of card	**Back of card**
User Story: User wants to be able to choose whether he will view the statistics of past games or start a new game. **As a:** user **I want:** to be able to choose between seeing the statistics and playing a new game upon launching the command line version of the game. **So that:** I have options. **And I know I am done when:** Upon launching I am able to choose between the two.	**Priority:** 8 **Estimate:** 8 story points **Tests:** Try to launch the command line version of the game and choose to view statistics of past games Try to launch the command line version of the game and choose to play a new game

Conversation:
Can the user view the statistics of past games?
Are the statistics clearly stated and readable?
Can the user start a new game?
Is the new game running properly?
Can the user do that on both the command line version of the game and the online one?

Happy day – Upon launching the command line version of the game the user is prompted to choose between viewing the statistics or playing a new game. When the user chooses one of the two, it runs without errors.
Rainy day – Upon launching the game the user is not given a choice and a new game starts.
Rainy day – Upon launching the game the user is given a choice between viewing the statistics or playing a new game but when the user selects one of the options an error occurs.

Story 2

Front of card	Back of card
User Story: After viewing the statistics of past games, the user wants to play a new game. **As a:** user **I want:** to start a new game after viewing the statistics of past games **So that:** I can play the game **And I know I am done when:** I can play a new game after viewing the statistics	**Priority:** 10 **Estimate:** 5 story points **Tests:** Try to view the statistics and choose to play a new game. Try to view the statistics and choose to exit the game. Try to view the statistics and then refresh and view them again.

Conversation:
Does the game offer a choice of a course of action after the user has viewed the statistics?
Can the user start a new game after the statistics have been displayed?
Can the user do that on both the command line version of the game and the online one?
Happy day – The user can start a new game after he has seen the statistics of past games.
Rainy day – The user cannot start a new game after the statistics have been displayed.

Story 3

Front of card	Back of card
User Story: After each round the user wants to be able to see the round number, the name of the active player and the card drawn. **As a:** user **I want:** to be able to see the round number, the name of the active player and the drawn card **So that:** I am sure there haven't been any mistakes **And I know I am done when:** I am able to see the round number, the name of the active player and the drawn card after each round.	**Priority:** 7 **Estimate:** 3 story points **Tests:** Try to view the round number after each round. Try to view the name of the active player after each round. Try to view the drawn card after each round.

Conversation:
Can the user see the round number?
Can the user see the active player's name?
Can the user see the card that he/she has drawn?
Happy day – The user is able to see the round number, the active player's name and the drawn card after each round is finished.
Rainy day – The user cannot see the round number, the active player's name and the drawn card after each round is finished.

Story 4

Front of card	**Back of card**
User Story: If the user is the active player, the game should ask the player to select a category. **As a:** user **I want:** to be able to select a category if I am the active player **So that:** I can play the game **And I know I am done when:** I am able to select a category when I am the active player.	**Priority:** 10 **Estimate:** 8 story points **Tests:** Try to select a category when the user is the active player Try to select a category when the user is the inactive player

Conversation:
Can the user select a category when he/she is the active player?
Can the user select a category when he/she is the inactive player?
Happy day – The user can select a category when he/she are the active player.
Rainy day – The user cannot select a category when he/she is the active player.
Rainy day – The user can select a category when he/she is the inactive player.

Story 5

Front of card	**Back of card**
User Story: If the user is the inactive player, the AI player should select a category **As a:** user **I want:** the AI player to select a category if I am the inactive player **So that:** I can play the game **And I know I am done when:** The AI player has selected a category when it is not my turn.	**Priority:** 10 **Estimate:** 8 story points **Tests:** Try to select a category when the user is the inactive player Try to wait for the AI player to select a category when the user is the inactive player

Conversation:
Can the user select a category when he/she is the inactive player?
Does the AI select a category when the user is the inactive player?
Happy day – The AI player selects a category when the human is the inactive player.
Rainy day – The human player is able to select a category when the AI is the active player.
Rainy day – The AI doesn't select a category when it is the active player

Story 6

Front of card	**Back of card**
User Story: The user wants the game to inform him who won the round or if it was a draw. **As a:** user **I want:** to know who won the round or if it was a draw **So that:** I am aware of the progress of the game **And I know I am done when:** after each round a message informs me of the winner of that round	**Priority:** 9 **Estimate:** 5 story points **Tests:** Try it with a round that results in a draw Try it with a round that has been won by the human player Try it with a round that has been won by an AI player

Conversation:
Does the game print out the name of the winner of each round?
Does the game print out an informative message?
Does the game inform the user if the round has been a draw?
Happy day – The game prints out the result of each round after it has ended.
Rainy day – The game does not inform the user who the winner of the round was.

Story 7

Front of card	**Back of card**
User Story: The user wants the game to inform him what was the winning card and the category it won in after each round. **As a:** user **I want:** to know what the winning card and the category it won in were after each round **So that:** I am informed about the process of the game. **And I know I am done when:** after each round a message informs me what the winning card was and the category it won in.	**Priority:** 8 **Estimate:** 5 story points **Tests:** Try it with different categories Try it with different cards Try it with a round that ends in a draw

Conversation:
Does the game inform the user about the winning card's name?
Does the game inform the user about the category that was selected in the current round?
Happy day – The game informs the user about the winning card of each round and the category the card won in.
Rainy day – The game doesn't display any information about the winning card or the category it won in.

Story 8

Front of card	**Back of card**
User Story: The user wants to know if a player has been eliminated (has no more cards left). **As a:** user **I want:** to know when a player is eliminated from the game. **So that:** I am aware of the game's progress. **And I know I am done when:** The game has informed me whenever a player has been eliminated.	**Priority:** 7 **Estimate:** 5 story points **Tests:** Try it when an AI player has been eliminated Try it when the human player has been eliminated

Conversation:
Does the game inform the user when an AI player has been eliminated?
Does the game inform the user when he/she has been eliminated?
Happy day – The game informs the user clearly about any player that have no more cards left and have therefore been eliminated.
Rainy day – The game does not provide any information about the eliminated players.

Story 9

Front of card	**Back of card**
User Story: If the user has been eliminated, the remaining rounds will be completed automatically, without user input. **As a:** user **I want:** the game to be automatically completed if I am eliminated and the name of the winner to be displayed **So that:** I know who won **And I know I am done when:** The game is completed even when I am eliminated and the name of the winner is displayed.	**Priority:** 8 **Estimate:** 8 story points **Tests:** Try it when the user is eliminated Try it with different number of players

Conversation:
Does the game automatically finish after the use has been eliminated?

Does the game provide the name of the winning AI player when the user has been eliminated?
Happy day – If the user has been eliminated, the game is automatically completed without his/her input and the name of the winner is displayed.
Rainy day – The game ends when the user is eliminated and is not completed and there is no winner.

Story 10

Front of card	**Back of card**
User Story: After the game has finished, the user would like to be given a choice between viewing the statistics or playing another game. **As a:** user **I want:** to be able to choose whether I would like to view the stats or play another game after my current game has finished **So that:** I have options **And I know I am done when:** after I have completed a game I am given a choice between starting a new game or viewing the stats.	**Priority:** 8 **Estimate:** 8 story points **Tests:** Try it when the user has won Try it when the user has lost

Conversation:
Does the program provide the user with a choice to start a new game or view stats after the current game has ended?
Happy day – After a game is finished the program offers the user a choice between starting a new game or viewing the statistics from past games.
Rainy day –The program does not give any options after a game is finished.

Story 11

Front of card	**Back of card**
User Story: During game play, the GUI should display the contents of the user's top card in every round. **As a:** user **I want:** to be able to see the contents of my top card **So that:** I know what card I have **And I know I am done when:** the game displays the content of my top card during each round.	**Priority:** 7 **Estimate:** 3 story points **Tests:** Try it with different cards Try it with different rounds

Conversation:

Does the game display the contents of the top card of the user?
How clearly does the game display this?
Happy day – The game displays the contents of the top card of the user during every round.
Rainy day –The user is unaware of the contents of his top card.

Story 12

Front of card	Back of card
User Story: The user wants to play against 2 AI players **As a:** user **I want:** to be able to play the game against 2 AI players **So that:** I can compete against 2 other players **And I know I am done when :** The game has dealt the 40 cards, with 2 players receiving 13 cards and one player receiving 14 cards.	**Priority:** 6 **Estimate:** 5 story points **Tests:** Try it with different decks Try it a number of times to see if the player with 14 cards is chosen at random

Conversation:
Is the user with 14 cards chosen at random?
Happy day – The user successfully plays the game against 2 AI players and the player with more cards is always chosen at random
Rainy day – The user encounters an error and cannot play the game with 2 AI players
Rainy day – The player with 14 cards is always the same.

Story 13

Front of card	Back of card
User Story: The user would like the statistics to contain the number of games played overall, how many times the computer has won, how many times the human has won, the average number of draws, the largest number of rounds played in a single game. **As a:** user **I want:** to the statistics to include the aforementioned **So that:** I have information about past games **And I know I am done when:** The statistics include the number of games played overall, how many times the computer has won, how many times the human has won, the average number of draws, the largest number of rounds played in a single game.	**Priority:** 7 **Estimate:** 13 story points **Tests:** Try it with different stats Try it different users

Conversation:

Do the stats contain this information?
How are these numbers calculated?
Happy day – The statistics of the game calculate and display the number of games played overall, how many times the computer has won, how many times the human has won, the average number of draws, the largest number of rounds played in a single game.
Rainy day – The statistics do not calculate or display the aforementioned.

Online user stories

Story 14

Front of card	**Back of card**
User Story: The user wants the GUI of the online version of the game to be simple, clear and easy to use **As a:** user **I want:** to be able to use the game's interface without difficulty **So that:** I can easily play the game **And I know I am done when:** The game's interface is easy, clear and simple to use.	**Priority:** 3 **Estimate:** 2 story points **Tests:** Try it with a user with little IT skills Try it with a user with advanced IT skills

Conversation:
Is the game's GUI simple and easy to use?
Can any user navigate through the GUI without any issues?
Is prior knowledge of the software necessary before playing the game?
Happy day – The GUI is easy to understand, and any user can navigate through it
Rainy day – The GUI is overly complicated and is difficult to understand by a user without prior knowledge of the software

Story 15

Front of card	**Back of card**
User Story: Once the category has been selected for a round and the values compared, the GUI should display the values of the category for each player and highlight who won the round. **As a:** user **I want:** to be able to see who won each round and why **So that:** I know how the game is progressing **And I know I am done when:** the GUI has displayed the values of the category for each player and highlighted who won the round.	**Priority:** 8 **Estimate:** 8 story points **Tests:** Try it with different cards Try it with different rounds Try it with different categories

Conversation:
Does the GUI display who won each round?
Does the GUI display the values of the category for each player?
Happy day – The GUI displays the winner and the values of the category for each player.
Rainy day –The GUI does not display the winner and the values of the category for each player.

Story 16

Front of card	**Back of card**
User Story: The GUI should contain an indication of how many cards are in the communal pile. **As a:** user **I want:** to know how many cards are in the communal pile **So that:** I know how the game is progressing **And I know I am done when:** the GUI has an indication of how many cards are in the communal pile.	**Priority:** 7 **Estimate:** 3 story points **Tests:** Try it with different cards Try it with different number of cards

Conversation:
Does the GUI display the number of cards left in the communal pile?
Does the GUI have an indication when the cards are nearly over?
Happy day – The GUI has an indication of the number of cards in the communal pile
Rainy day – The GUI does not have an indication of the number of cards in the communal pile

Story 17

Front of card	**Back of card**
User Story: The GUI should contain an indication of how many cards are left in the user's deck and in the computer's decks **As a:** user **I want:** to know how many cards are in mine and the computer's decks **So that:** I know how the game is progressing **And I know I am done when:** the GUI has an indication of how many cards are in the user's and computer's decks	**Priority:** 7 **Estimate:** 3 story points **Tests:** Try it with different cards Try it with different number of cards

Conversation:
Does the GUI display the number of cards left in the user's and computer's decks?
Does the GUI have an indication when the cards are nearly over?
Happy day – The GUI has an indication of the number of cards in the user's and computer's decks

Rainy day – The GUI does not have an indication of the number of cards in the user's and computer's decks

Story 18

Front of card	Back of card
User Story: The software developer wants to update the game **As a:** software developer **I want:** to be able to update the game **So that:** the game is compatible with new systems and platforms and is secure **And I know I am done when:** The game is running successfully on all devices regardless of their release date	**Priority:** 10 **Estimate:** 13 story points **Tests:** Try it with different hardware Try it with different versions of Windows/Mac OS Try to subject the software to a treat

Conversation:
Can the game run on different versions of Windows/Mac OS ?
Can the game be updated?
Is the software safe from any cyber treats that might occur?
Happy day – The software developer updates the game and it is safe and running on all devices.
Rainy day – The software cannot be updated and is subjected to malware and malfunction.

Story 19

Front of card	Back of card
User Story: The software developer wants to be able to maintain the software **As a:** software developer **I want:** to be able to maintain the software **So that:** the software's performance is optimal **And I know I am done when:** The game is always the best version of itself and there are no faults	**Priority:** 10 **Estimate:** 13 story points **Tests:** Try it with correcting faults that might occur Try to improve performance Try to adapt the software

Conversation:
Can the game run smoothly and without any glitches?
Can the code be corrected?
Happy day – The software developer is able to maintain the software and perfect its performance.
Rainy day – The software cannot be maintained properly, and issues cannot be resolved.

Story 20

Front of card	Back of card
User Story: User wants to log in **As a :** user **I want :** to be able to log in and log out of the Top Trumps Game **So that:** I can play the game **And I know I am done when :** The game is accessible when I input my username and password.	**Priority:** 1 **Estimate:** 4 story points **Tests:** Try it with unregistered users Try it with register users Try case sensitive log in Try it with cap lock ON Try it with lower case only Try weak password Try strong password

Conversation:
Is a log in essential?
Yes because the game needs to keep record of the results.
Can the user change his password?
Can the user change his email?
Do we need the user to register before he has a log in access?
Happy day – it works
Rainy day – an issue (case sensitive), it does NOT work

Story 21

Front of card	Back of card
User Story: User wants to learn how to play **As a:** user **I want:** to learn how to play **So that:** I can play the game **And I know I am done when:** There are instructions about how to play the game.	**Priority:** 1 **Estimate:** 4 story points **Tests:** Try it with new users Try it with register users Try it upon registration

Conversation:
Does the game provide instructions about how to play the game to new users?
Is there a tutorial that teaches new users to play the game?
Is there a button to bring up the instructions of the game?
Happy day – instructions are available to new users and to already registered users if they need them.
Rainy day – there are no instructions, neither when registering, nor following that.

Non-Functional Requirements

Constraint Card 1:

Front of card	Back of card
Story: The system must have a short response time. Acceptance Criteria: The system reacts to the user's commands within a maximum response time of 10 seconds	Conversations: This non-functional requirement might be needed because it will allow for the system to be used with ease and swiftness

Constraint Card 2:

Front of card	Back of card
Story: The system must be able to save the information from the games played. Acceptance Criteria: The system stores the information and its capacity is suitable for its use.	Conversations: This non-functional requirement might be needed because it will allow all the information required to be stored without reaching the limit of the system.

Constraint Card 3:

Front of card	Back of card
Story: The system must create a back-up for all the information that it has to store and operate with. Acceptance Criteria: The system creates a back-up for all the data.	Conversations: This non-functional requirement might be needed because it will allow the users to acquire access to this information even in the case of system break down.

Top Trumps Game: Release Plan

Total number of story points = **126**

Iteration velocity = **63** points/ sprint

Estimated project length = total number of story points / iteration velocity (126/63)

Estimated total project length = 2 sprints each sprint is 2 weeks, 1 additional week for backlog burn. Total project time is 5 weeks.

Release Goals:

1. First release after user command line game is running
2. Second release after on line version is running
3. Third release after database is running with statistics display and test log file
4. Final release after all stories are done and the game has been fully tested.

First Sprint:

Command Line Stories 1, 2, 3, 4, 5, 6, 7, 8, 9, 10

Estimation: 63

Estimation/Velocity = 63/63 = 1 sprint

Second Sprint:

Continue with command line stories

Stories: 11, 12, 13, develop the online version 14, 15, 16, 17, 18, 19

Estimation: 63 story points

Estimation/Velocity = 63/63 = 1 sprint

Table 1 - User Stories Priority & Effort Chart

Sprint 1 Story Number	**Effort Story Points**	**Priority**
1	**8**	**H**
2	**5**	**H**
3	**3**	**M**
4	**8**	**H**
5	**8**	**H**
6	**5**	**H**
7	**5**	**H**
8	**5**	**M**
9	**8**	**H**
10	**8**	**H**
Sprint 2 Story Number	**Effort Story Points**	**Priority**
11	**7**	**L**
12	**5**	**M**
13	**13**	**H**
14 on line version	**2**	**L**

15	**8**	**H**
16	**3**	**H**
17	**3**	**H**
18	**13**	**H**
19	**13**	**H**
20	**4**	**Out of scope**
21	**4**	**Out of scope**
Backlog Burn		**Priority**
Incomplete or added stories. Fix bugs. Report Writing.		**H**

Using the priority chart, it was easy to see which stories were the highest priority. The code development followed a logical sequential order. Write the classes to code the game function, i.e. Top Trump Card, Top Trump Player, Top Trump Game- the game logic. Then deal with the command line user input/ output in display and output to log file. Then develop the online version, adding the database and statistical requirements.

Table 2 - Functionality Grid for Top Trump Game

Stories are organised into functionality and MHSHCH

Functionality 1 MH for the game to work on the command line		**Functionality 2 SH Design for online version**		**Functionality 3 CH Features for a real life web game**		**Functionality 4 MH Non-functional requirements**	
1	**8**	**14**		**20**		**3 constraint cards**	
2	**9**	**15**		**21**			
3	**10**	**16**					
4	**11**	**17**					
5	**12**	**18**					
6	**13**	**19**					
7							

Initial Planning

Form the Group. Team meet with introduction, Team Roles, Introduction to Scrum, Team Organisation Document to be submitted.
Team agree the following:

- loose scrum rules
- roles and responsibilities
- regular meeting times
- channels of communication
- set up Group Shared Files OneDrive at Git repository
- Definition of 'Done'
- plan of action
- who is working on what

Sprint 1

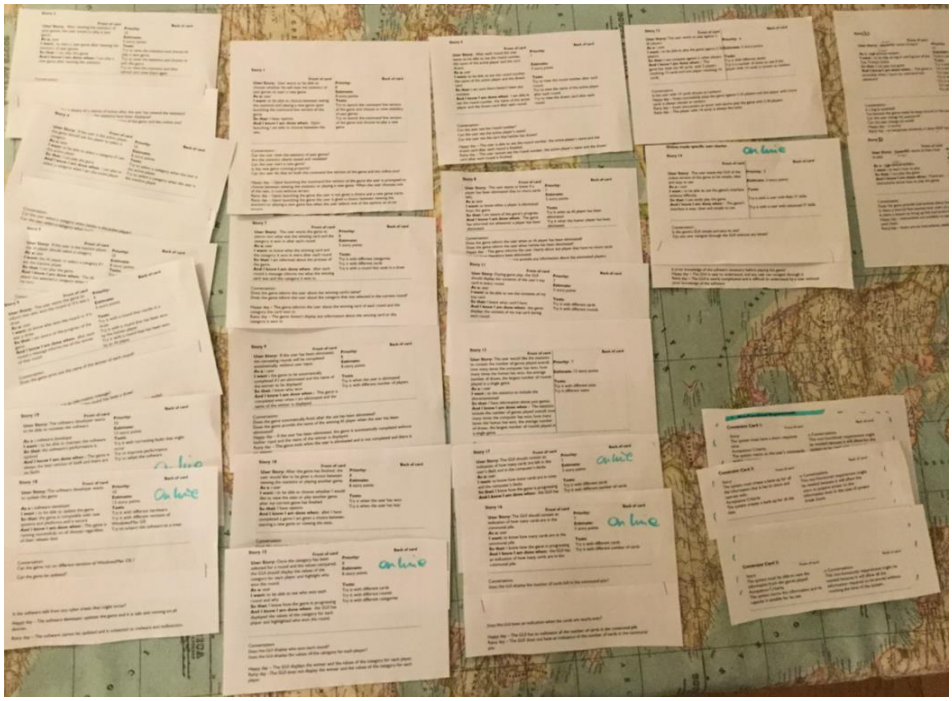
Research: The team research and discuss the Scrum process, Scrum values, Scrum goals: lead by Cassie. Lyle researched and started to learn the new technologies involved with the API : CSS, HTML and Muhammad researched the database requirements. The team all downloaded and took time to become familiar with the template code.

Gathering requirements: the coursework specification was very clear about the requirements.

Clarifying game functionality: The team engaged in real life playing the game with real cards, so everyone understood how the cards are used, how draws work, who's turn is next. In the meeting functionality was discussed; how cards are moved physically, who's turn is next when active player wins, what happens after a draw. Random shuffling of cards, random selection of who is player one etc.

Personas & User Stories - Martina and Cassie developed the initial stories, these were reviewed by the team and rewritten to better reflect the requirements. The team discussed and deciding story priorities and which stories are in the scope of the project, which were not. Stories 20 and 21 are out of scope. Estimating times of the stories points was developed by team discussion.

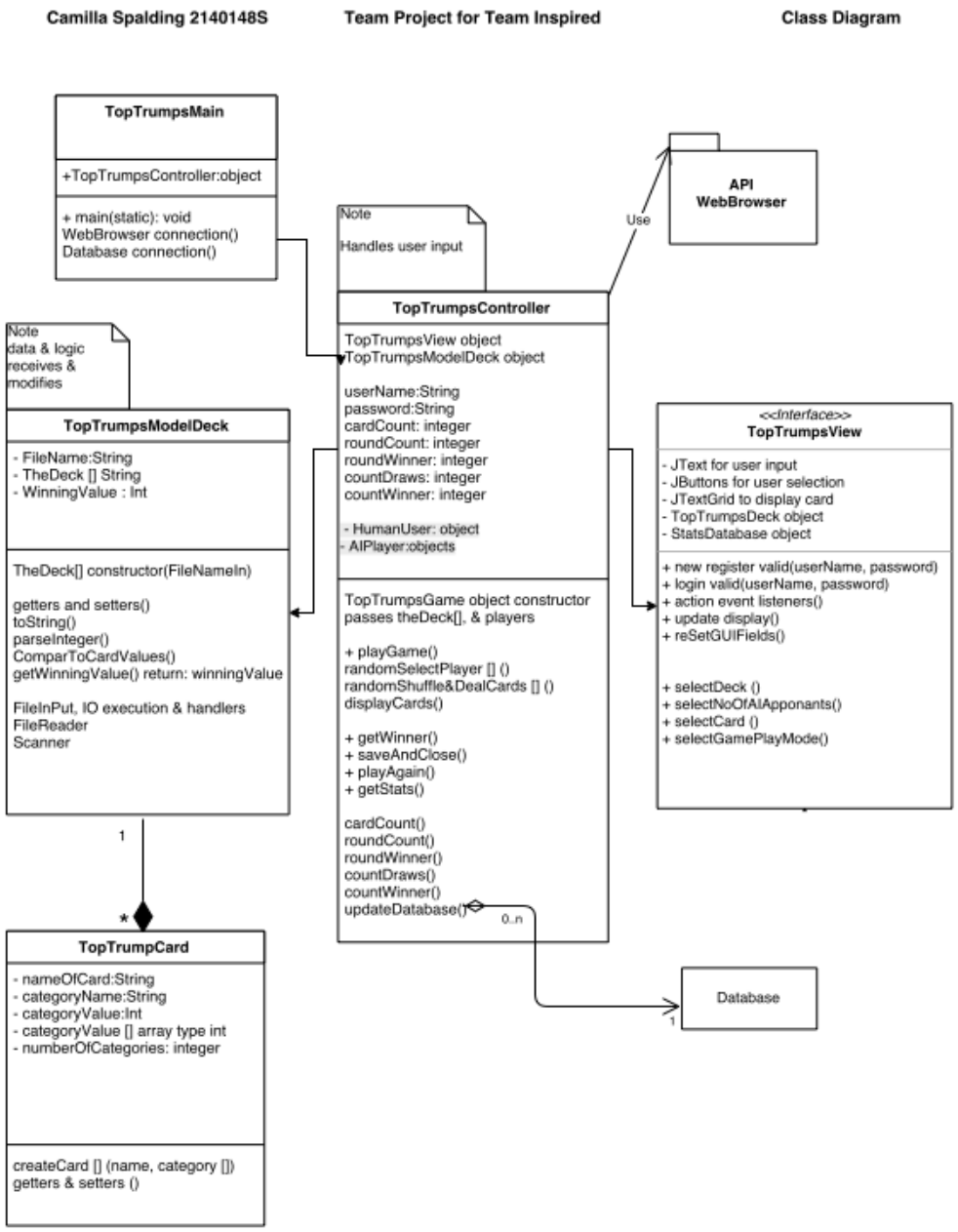
Figure 1 - Story Map Top Trumps Game



Martina led the decision making on which stories to do first – organising sprints.

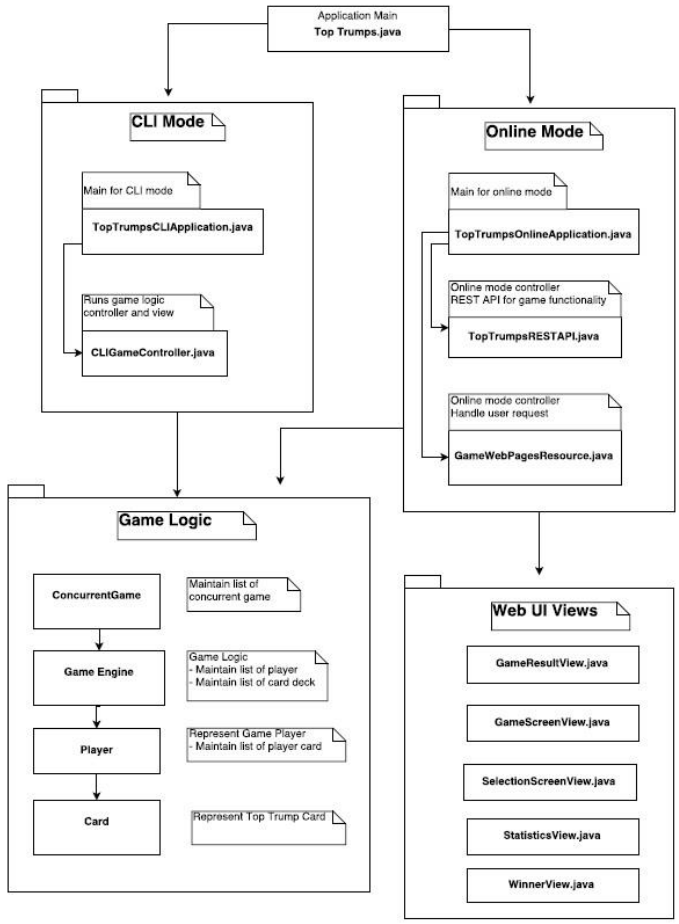
Producing a class diagram prior to analysing the coursework template was helpful in the process of thinking though what code was needed to set up the initial game functionality, it helped work out was methods and instances were needed in a MVC design pattern.

Figure 2 – Class diagram



On reflection this diagram is missing a main class and a player class. The actual template code for the coursework was complex, with layered cohesion using packages and folders to separate all the different levels of applications for the web pages, and database.

Figure 3 – Final Top Trumps program class diagram



The team all began developing code from the user stories, it was very clear Candra was way ahead and more advanced with the game functionality using ArrayLists to maintain the Players and the Cards in the Game Engine. E.g. select the first card from each player in their ArrayList to an new card-in-play communal array list, the active player decides the attribute, the winner is the highest attribute, these winning cards are added to the winners array list. After each round the top card index 0 is swapped to the back so the index 1 becomes index 0 etc. The team all agreed to use Candra's version follow his lead. Lyle set up a Git Hub repository for the team to use. Cassie set up a shared folder on the cloud OneDrive.

Sprint 1 Review

The command line version was ahead of schedule by the end of this sprint. See screen shots in the Testing section. The team reviewed the working code, reflecting on the coursework requirements asking if any game functionality was missing, and what needed to be done next. The team discussed the display and use of language and if this was confusing for a user, and how to offer the user options. In this meeting 3 options were proposed and agreed, (1) play again (2) view statics or (3) quit game (Story 1 &10). The team tested the log out file. Actual velocity was higher than estimated as 14 stories were 'Done', the command line version and output to a Log file was all running (Story 13). The code needed some

refactoring and a little more time on Story 2, 3, 6, 7, 11,13 to make the display and language clearer for the user.

Estimated Story Points = 63

Actual Story Points 'done'= 84

Actual Velocity: 84/63 = 1.3

Screen Shots in Testing show Command Line Mode Functionality: (See Testing section)

Sprint 2

Developing the online mode of the Top Trumps game

Lyle and Candra worked on the development of the online mode of the game, with Candra developing the code itself and Lyle working on the styling of the game and Cassie and Martina supporting with general coding and input from user stories, reviewing requirements and testing. The team communicated via Facebook and shared documents and code via OneDrive. Screenshots were shared via OneDrive and the process of styling of the online version of the game can be observed in the Testing section below under figures 2, 3 and 4 which are screenshots of the different stages of the online mode of the game.

The team met regularly through the period of the development of the second sprint and hence the online mode of the game and kept each other up to date with both the process of coding and following the requirements. In each meeting Martina and Cassie refined the user stories and did regular checks of the code and the code comments and specifically the use of language, e.g. "first player" was changed to "active player", when a player, whether an AI or the human player were out, they were "eliminated", the draw pile was named the 'the communal pile' which contains the cards from the draw, which are added to the winner of the round's pile. Cassie refined the class diagram and the domain model and configured code on the lab PCs with Candra. The updated versions of the code were tested on multiple occasions by all the members of the team, then agreed the stories were done. Several reviews of the requirements over the final few days of sprint 2 to check we had fulfilled all the criteria and to make tiny changes to the code comments, to the wording on the display.

Developing the Database for the online mode of the Top Trumps game

Time was spent on designing Game Creation Script there are several versions showing the development in the appendix. (See Figures....), the statistics are styled differently.

Lyle and Martina created the ER Diagram to organise the database. (See ER diagram below) Muhammad developed the code and SQL queries for the database, but encountered a major problem when he lost his code, so he spent a stressful couple of days trying to recover it. (See appendix: SQL query txt and ER diagram). Since Candra has the most experience with software development in the team, he helped out with any issues that Muhammad had with the database code.

Sprint 2: Review

Screen Shots in Testing show Online Mode functionality: (see Testing section)

Lyle put together a Testing document with screen shots.

Candra wrote another file with football.txt to test another deck of cards, and this ran successfully. (see Screen shot in Appendix).

Sprint 1 had a higher velocity than expected. But in Sprint 2, developing the online section took more time than expected as unforeseeable technical issues arose. The time gained in Sprint 1 was lost, ending up with a much lower velocity in sprint 2. The higher than expected velocity of Sprint 1 meant the team still managed to finish the project on time.

Estimated Story Points = 63

Actual Story Points 'done'= 50

Actual Velocity: 42/63 = 0.6

Backlog

Configuration time was lengthy, developing programs to release in industry would require much more planning and time.

New Stories Added

We added a Fast Forward Game button to the online version so when the user had been eliminated they did not have to click through the rounds. We added a non functional restraint Game Reset to enable us to test the program more efficiently. Once tested the team agreed these features were 'done'.

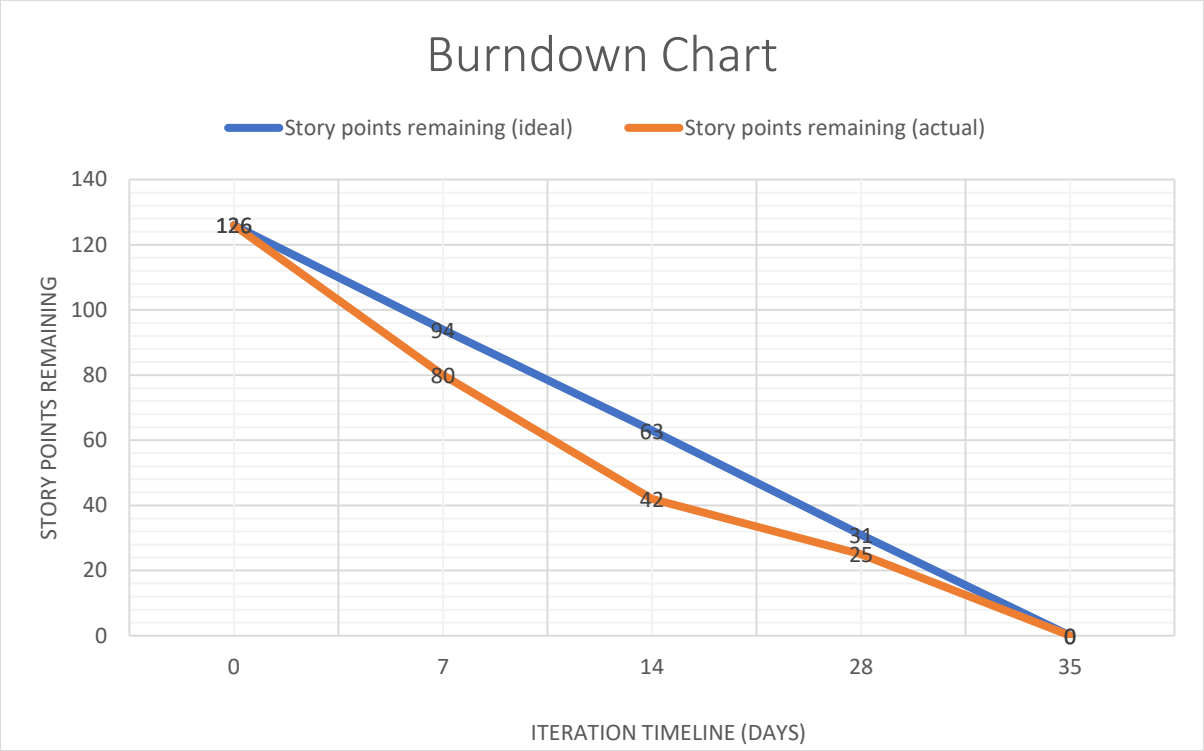
The team realised a deficiency in a final review, then added a class ConcurrentGame.java to enable the program to run concurrently in sperate browser tabs. Hash maps were used to keep each version of the game separate but running concurrently.

The deployment was fairly simple, but still technical problems with the Maven Dependencies and the database was more time consuming than expected. Also putting together all the documentation to write up the group report was a lengthy process. Our team gathered all the drafts reports for a report meeting to finalise all the report writing.

Only when the whole Team agreed all information was correct & completed, could work be signed off or coursework submitted.

Figure 4 – Burndown chart – a chart capturing the process of developing the user stories in a timeline encompassing the timeline of the project (35 days), the first sprint ends at the 14th day of the timeline and the second sprint ends at the last day.

Burndown Chart



Configuring all the separate parts together for the online version was most time consuming, and there was no user story to reflect the work.

Assumptions

The program assumes;

(1) Any users know how to play the game Top Trumps as we are not required to provide an instruction set for this coursework. To make our code more maintainable, to allow an instruction set to be added on in the future, we initially included a hook 'Help' button. We took this out of our code as we did not want to be marked down for gold plating the requirements.

(2) The human player is always the same user. If developing this further we would add an opening window to allow users to login or register, and to choose a username, this would extend program to allow the database to keep track of different human users scores as additional statistics.

(3) The template is set up and run securely as we have not added non-functional security requirements.

(4) When the last round is a draw and there are no more cards for each player, the active player who won the last round is the winner of the game.

(5) In order for our online program to run with the database we assume the user is running a lab machine connected to the database PostgreSQL on the yacata server, as the database password is in the jar.file. Finally for the short cut to run we assume the directory is H:\ as this is the directory for the Desktop for the Lab machines.

(6) The user types **`http://localhost:7777/toptrumps`** in lowercase as uppercase does not connect.

(7) The user will play until the current game is completed. If the user closes the browser prior to the game ending, the statistics may result in incorrect calculations. A possible solution might be to save the current game statistics into a temporary file and only update the database once the game is completed.

(8) If a user wants to change how many AI opponents to play against they need to change the configuration in the program, by changing the number of players value (max 4) on the TopTrumps.json "`numAIPlayer:_`".

Limitations

The user cannot choose other decks of cards, this could be a extension feature to develop later the files with accompanying image folders would have to be written into the program if the user wants to choose another deck they have to configure the Top Trumps.json and input a next deck txt file e.g. "SoccerTopTrumpDeck.txt" .

The program is limited to running on Lab machines and would need to connected to yacata to run on any other laptop or computer.

The game has not been tested the program on Linux /macOS. It does not have a phone or tablet application.

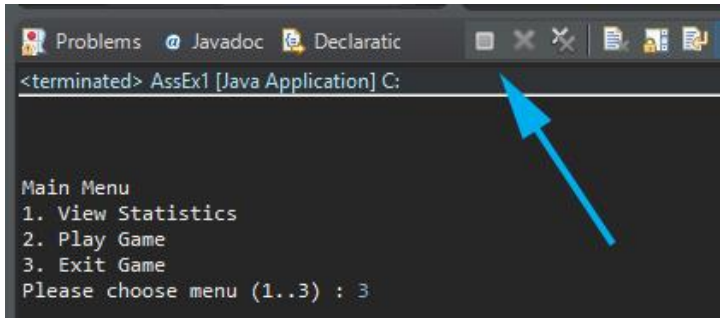
The program only keeps statistic of a generic human player, who is nameless. There is no way to store individual user's names, or to allow different users to compete with other users scores in the game. The user cannot choose how many AI opponents in the GUI. The game has no instruction set.

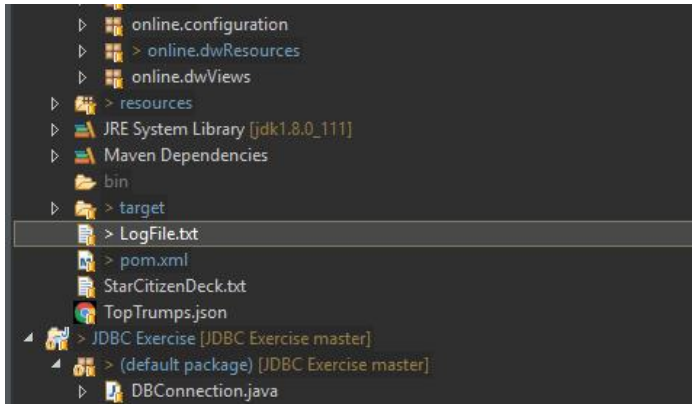
Deficiencies – none the team are aware of.

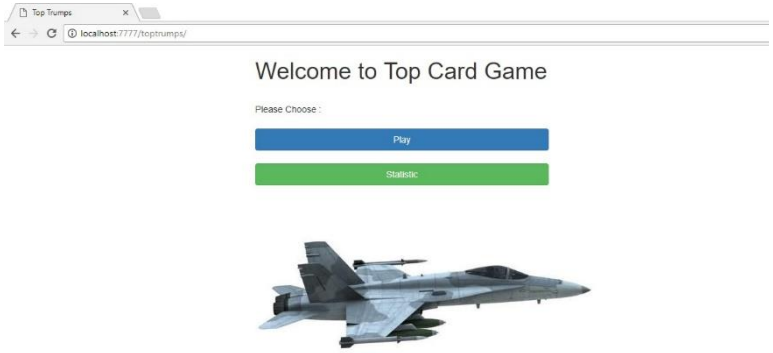
Top Trumps: Testing

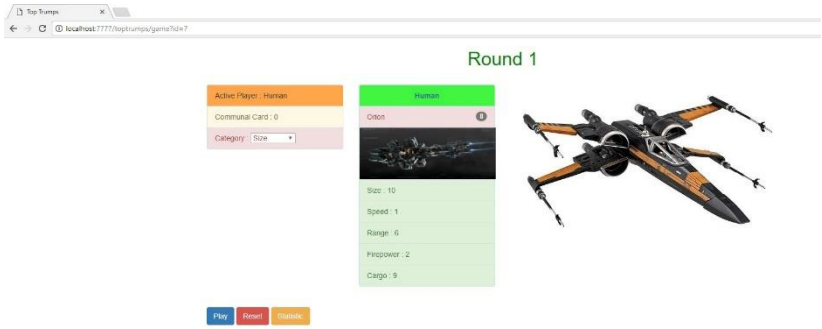
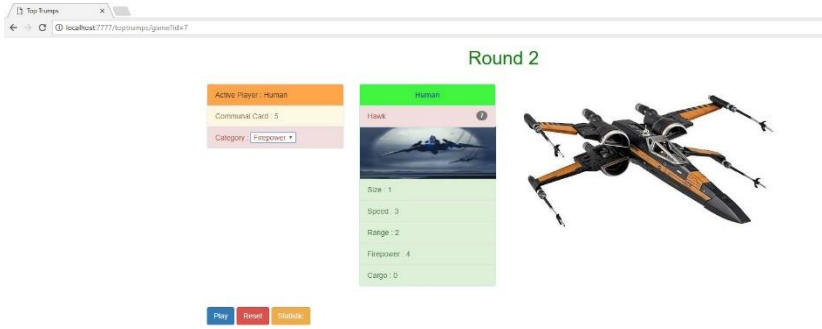
Top Trumps: Console Version		
Test	**Expected Result**	**Actual Result**
Test: Launching the console version of the game **Description:** Running the main TopTrumps.java file in Eclipse using the '-c' switch	The game should start-up and run in the console window, prompting the user to choose from the main menu.	The test performs as expected: Main Menu 1. View Statistics 2. Play Game 3. Exit Game Please choose menu (1..3) :
Test: Viewing statistics of previous games **Description:** Choosing main menu option "1. View Statistics"	The console should display statistical information on previous games such as: number of games, record of wins and record of rounds.	The test performs as expected: Please choose menu (1..3) : 1 ---------- All Game Summary ---------- Total Game : 1 Human Win : 0 AI Player Win : 1 Total Draw : 9 Largest Round : 86 ---------- Previous Game Stats ---------- Game Winner : AI-3 Total Draw : 9 Total Round : 86 ---------- Previous Round Winner ---------- Human : 31 AI-3 : 55

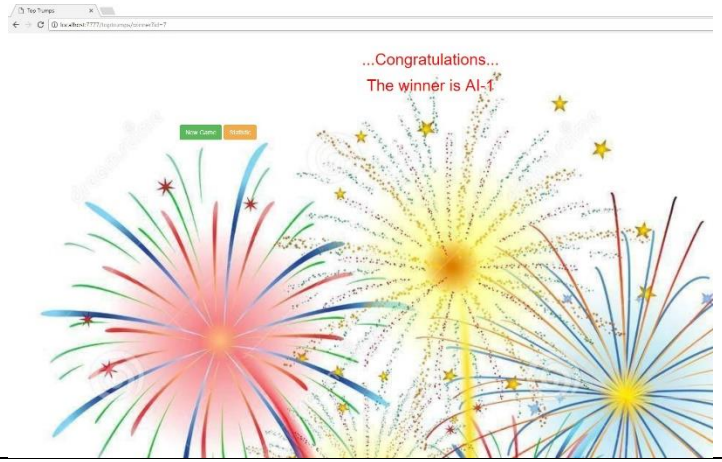
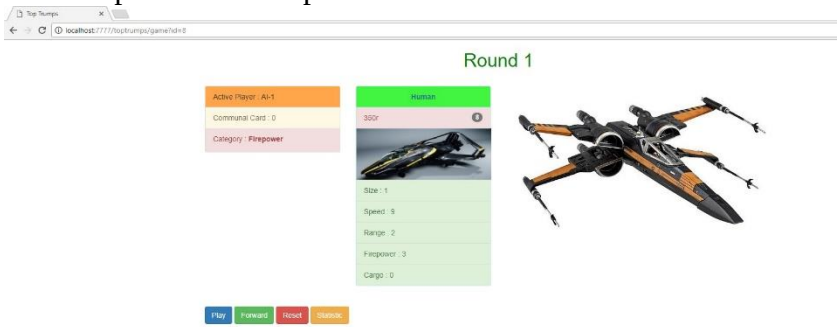
Test: Playing the game **Description:** Choosing main menu option “2. Play Game”	A new game of Top Trumps should be initialised beginning with round 1; if the human player not the starting player, rounds should automatically progress until the human becomes the active player.	The test performs as expected: Please choose menu (1..3) : 2 >> ROUND : 1 << First Player : AI-3 ---------- First Player Card ---------- Description Size Speed Range Firepower Cargo Idris 8 2 7 10 6 Choosen Characteristic Firepower Round is draw Eliminated player = >> ROUND : 2 << First Player : AI-3 ---------- First Player Card ---------- Description Size Speed Range Firepower Cargo Avenger 2 5 4 3 2 Choosen Characteristic Speed Round Winner is : Human ---------- Winning Card ---------- Description Size Speed Range Firepower Cargo 350r 1 9 2 3 0 Eliminated player = >> ROUND : 3 << First Player : Human ---------- First Player Card ---------- Description Size Speed Range Firepower Cargo Hawk 1 3 2 4 0

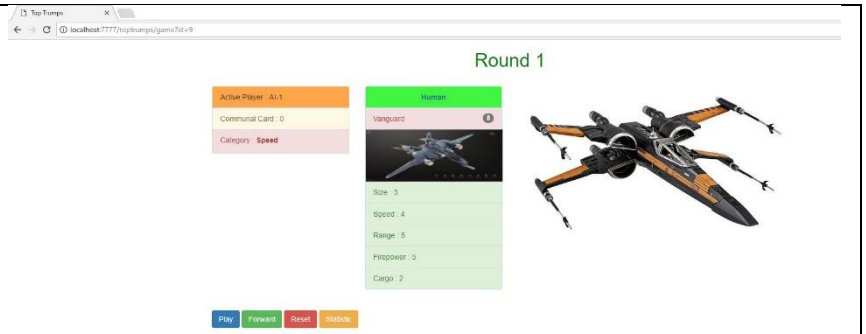
		Category Option : 1.Size 2.Speed 3.Range 4.Firepower 5.Cargo Please input option category index (1..5) :
Test: Choosing a category **Description:** Selecting one of the 5 options size, speed, range, firepower or cargo	Depending on the choice of the human player, the winner of the round should be calculated and the subsequent rounds should progress	The test performs as expected. In this example, the rounds progressed until the human player was eliminated in round 21 and the game finished after 43 rounds where AI-2 player won: […] Choosen Characteristic Speed Round is draw Eliminated player = AI-3 AI-4 Human AI-1 Game Winner is : AI-2
Test: Exiting the game **Description:** Choosing main menu option "3. Exit Game"	The game should terminate	The test performs as expected. After option 3 is selected, the terminate button in Eclipse greys out demonstrating that the application is no longer running: 

Test: Printing a test log **Description:** Running the main TopTrumps.java file in Eclipse using the '-c' and '-t' switches	After exiting the game, a test log text file should be created in the game directory	The test performs as expected. After exiting and refreshing the game folder, a new LogFile.txt is created which has a record of the round results:  ********** Game - 1 ********** Original Deck : 350r~Avenger~Carrack~Constellation~Hawk~Hornet~Hurricane~Idris~m50~Merchantman~Orion~Sabre~Vanguard~350r~Avenger~Carrack~Constella Suffled Deck : Hornet~Hornet~Carrack~Merchantman~Constellation~Orion~m50~Sabre~Avenger~m50~Sabre~350r~Hurricane~Idris~Avenger~Vanguard~Constellat ---------- Round - 1 ---------- First Player : AI-1 All Player Card : Player AI-1 : Hornet~Orion~Sabre~Vanguard~Idris~350r~Hurricane~Hawk~ Player AI-2 : Hornet~m50~350r~Constellation~Vanguard~Idris~m50~Hurricane~ Player AI-3 : Carrack~Sabre~Hurricane~Hawk~Carrack~Carrack~Merchantman~Orion~ Player AI-4 : Merchantman~Avenger~Idris~Vanguard~Hawk~Avenger~Sabre~Hornet~ Player Human : Constellation~m50~Avenger~350r~Orion~Merchantman~Constellation~Avenger~ All Player Top Card : Player AI-1 : Hornet -> Size:2 Speed:5 Range:3 Firepower:4 Cargo:1 Player AI-2 : Hornet -> Size:2 Speed:5 Range:3 Firepower:4 Cargo:1 Player AI-3 : Carrack -> Size:6 Speed:2 Range:10 Firepower:4 Cargo:6 Player AI-4 : Merchantman -> Size:7 Speed:3 Range:5 Firepower:6 Cargo:8 Player Human : Constellation -> Size:4 Speed:5 Range:7 Firepower:3 Cargo:4 Choosen Characteristic : Speed, Characteristic Value : 5 Game is draw Cummonal Card : Hornet~Hornet~Carrack~Merchantman~Constellation~ ---------- Round - 2 ---------- First Player : AI-1 All Player Card : Player AI-1 : Orion~Sabre~Vanguard~Idris~350r~Hurricane~Hawk~ Player AI-2 : m50~350r~Constellation~Vanguard~Idris~m50~Hurricane~ Player AI-3 : Sabre~Hurricane~Hawk~Carrack~Carrack~Merchantman~Orion~ Player AI-4 : Avenger~Idris~Vanguard~Hawk~Avenger~Sabre~Hornet~ Player Human : m50~Avenger~350r~Orion~Merchantman~Constellation~Avenger~ All Player Top Card :

Top Trumps: Online Version		
Test	**Expected Result**	**Actual Result**
Test: Launching the online version of the game **Description:** Running the main TopTrumps.java file in Eclipse using the '-o' switch	After the launch, the game should be playable in a browser at address http://localhost:7777/toptrumps/	The test performs as expected. Eclipse's console window demonstrates a successful connection to the localhost address and the database. The online menu appears at the address: 

Test: Playing the online version of the game **Description:** Clicking the ‘Play’ button	The first round should begin. The rounds screen should highlight the active player, the category chosen by the active player, and the human player’s card.	The test performs as expected: 
Test: Selecting a category when the human player becomes active **Description:** Choosing an option from the drop-down categories box and clicking the ‘play’ button	Depending on the human player’s choice, the winner of the round should be calculated and all the player’s cards for the round should be revealed.	The test performs as expected: 
Test: Forwarding the rounds when the human is not the active player **Description:** Clicking the ‘forward’ button to speed up gameplay, either to the human player’s turn or end of the game	Depending on the results, clicking the ‘forward’ button should instantly play through the successive rounds, either until the human player’s next turn or end of the game	The test performs as expected.

Test: Playing through the game until the end **Description:** Participating in the rounds until a winner is determined by clicking ‘play’ and ‘forward’ buttons	When one player is left holding all the cards, the winner should be declared and the round progression should finish. The ‘New Game’ button should appear.	The test performs as expected: 
Test: Starting a new game after a winner is declared **Description:** Following a complete game in which a winner is declared, the ‘New Game’ button is clicked	A new game should be instantiated after ‘New Game’ is clicked, beginning at round 1.	The test performs as expected: 
Test: Resetting the game in the middle of a round **Description:** Clicking the ‘Reset’ button in the middle of playing the game	When the ‘Reset’ button is clicked, a new game should be instantiated beginning at round 1.	The test performs as expected:

		
Test: Viewing statistical information **Description:** Clicking the 'Statistics' button	A summary of statistics relating to previous games should open when the 'Statistics' button is clicked (relaying information from the database connection).	The test performs as expected: 

Appendix

- Testing with Screen shots
- Initial Ideas for Personas & User Stories - out of scope
- SQL query.txt
- Game CreateScript.sql
- ER diagram
- Bibilography

Initial Personas

User profile: Software Developer
Frequency of use: Regular
Domain expertise: Advanced
Software expertise: Advanced
Goals: Upkeep of the software (update/reset/debug)

User profile: Gillian Smith, age – 35y, occupation - Sales
Frequency of use: Casual/ occasional
Domain expertise: General
Software expertise: Basic
Goals: Play Top Trumps with her children

User profile: Imam Amir, age – 25y, occupation – IT student
Frequency of use: Regular
Domain expertise: Advanced
Software expertise: Intermediate
Goals: Play Top Trumps competitively

User profile: Alexander Smith, age –12y, occupation – high school student
Frequency of use: Regular
Domain expertise: General
Software expertise: Basic
Goals: Play Top Trumps for fun

Initial User Stories **(many of these out of scope)**

Story 1

Front of card	Back of card
User Story: Alexander wants to log in **As a :** high school student **I want :** to be able to log in and log out of the Top Trumps Game **So that:** I can play the game **And I know I am done when :** The game is accessible when I input my username and password.	**Priority:** 10 **Estimate:** 4 story points **Tests:** Try it with unregistered users Try it with register users Try case sensitive log in Try it with cap lock ON Try it with lower case only Try weak password Try strong password

Conversation :

Is a log in essential?
Yes because the game needs to keep record of the results.

Can the user change his password?

Can the user change his email?

Happy day – it works

Rainy day – an issue(case sensitive), it does NOT work

*Do we need the user to register before he has a log in access?

Story 2

Front of card	Back of card
User Story: Alexander wants to learn how to play **As a :** high school student **I want :** to learn how to play **So that:** I can play the game **And I know I am done when :** There are instructions about how to play the game.	**Priority:** 10 **Estimate:** 4 story points **Tests:** Try it with new users Try it with register users Try it upon registration

Conversation :

Does the game provide instructions about how to play the game to new users?

Is there a tutorial that teaches new users to play the game?

Is there a button to bring up the instructions of the game?

Happy day – instructions are available to new users and to already registered users if they need them.

Rainy day – there are no instructions, neither when registering, nor following that.

Story 3

Front of card	**Back of card**
User Story: Gillian wants the interface of the game to be easy and simple to use **As a :** working mom **I want :** to be able to use the game's interface without difficulty **So that:** I can easily play the game **And I know I am done when :** The game's interface is easy, clear and simple to use.	**Priority:** 5 **Estimate:** 2 story points **Tests:** Try it with a user with little IT skills Try it with a user with advanced IT skills

Conversation :

Is the game's GUI simple and easy to use?
Can any user navigate through the GUI without any issues?

Is prior knowledge of the software necessary before playing the game?

Happy day – The GUI is easy to understand and any user can navigate through it

Rainy day – The GUI is overly complicated and is difficult to understand by an user without prior knowledge of the software

Story 4

Front of card	**Back of card**
User Story: Gillian wants to play against an AI player **As a :** working mom **I want :** to be able to choose an opponent who is an AI player **So that:** I can learn to play the game so that I can play with my son	**Priority:** 10 **Estimate:** 8 story points **Tests:** Try it with more than one player. Try it when Gillian is player one.

And I know I am done when : I have successfully played the game against an AI player.	Try it with different number of players(max 5 + the user) Try it with random player selection

Conversation:

Can the user choose only one opponent?

Can that opponent be AI?

Are the players selected randomly?

The user should not ALWAYS be player 1 or they would have an unfair advantage when cards are not dealt equally.

It would be great if she could play the game against her son but this is beyond the scope of this program.

Happy day – Gillian succeeds in playing the game against the AI player

Rainy day – Gillian is always player one so she has an unfair advantage when the cards are dealt.

Rainy day – Gillian is unable to play the game against a single AI player.

Story 5

Front of card	**Back of card**
User Story: Imam wants to play against 2 AI players **As a :** student **I want :** to be able to play the game against 2 AI players **So that:** I can compete against 2 other players **And I know I am done when :** The game has dealt the 40 cards, with 2 players receiving 13 cards and one player receiving 14 cards.	**Priority:** 7 **Estimate:** 4 story points **Tests:** Try it with different decks Try it a number of times to see if the player with 14 cards is chosen at random

Conversation:

Is the user with 14 cards chosen at random?

Happy day – Imam successfully plays the game against 2 AI players and the player with more cards is always chosen at random

Rainy day – Imam encounters an error and cannot play the game with 2 AI players

Rainy day – The player with 14 cards is always the same.

Story 6

Front of card	**Back of card**
User Story: Alexander wants to select an attribute **As a :** high school student **I want :** to be able to select the attribute which will be evaluated this round **So that:** the winner of the round can be established **And I know I am done when :** The attribute is selected and the other players' cards are revealed and there is a winner of the round.	**Priority:** 10 **Estimate:** 4 story points **Tests:** Try it with different attributes Try it with no attribute Try it with more than one attribute Try to change the attribute before the round ends

Conversation:

Can the user choose an attribute?

Can the user select more than one attribute?

Can the user change an already selected attribute?

How many attributes are there for each category of deck?

Happy day – Alexander successfully selects an attribute and the game continues.

Rainy day – The game refuses to take Alexander's input into account and

Rainy day – Imam encounters an error message and the game does not recognize the deck he attempted to load.

Story 7

Front of card	**Back of card**
User Story: Gillian wants to view the ranking of all past games **As a :** working mom **I want :** to be able to be able to view the results of all past games **So that:** I can see the progress my son is making in the game	**Priority:** 6 **Estimate:** 4 story points **Tests:** Try it with different users Try it with the results from the last month Try to view the results of a particular player

And I know I am done when : I am able to access and see all the past games' results	Try to view the dates and times the game was played from a particular desktop Try to view the highest score of each player

Conversation:

Can the user choose the time period for which the stats will be displayed?

Can the user view just the results of a particular player?

Can the user view the log list – dates and times the game was played?

Happy day – Gillian successfully opens and views the stats for a particular time/day/user.

Rainy day – Gillian is unable to access the stats for the games played.

Story 8

Front of card	**Back of card**
User Story: The software developer wants to update the game **As a :** software developer **I want :** to be able to update the game **So that:** the game is compatible with new systems and platforms and is secure **And I know I am done when :** The game is running successfully on all devices regardless of their release date	**Priority:** 10 **Estimate:** 8 story points **Tests:** Try it with different hardware Try it with different versions of Windows/Mac OS Try to subject the software to a threat

Conversation:

Can the game run on different versions of Windows/Mac OS ?

Can the game be updated?

Is the software safe from any cyber threats that might occur?

Happy day – The software developer updates the game and it is safe and running on all devices.

Rainy day – The software cannot be updated and is subjected to malware and malfunction.

Story 9

Front of card	Back of card
User Story: The software developer wants to be able to maintain the software **As a :** software developer **I want :** to be able to maintain the software **So that:** the software's performance is optimal **And I know I am done when :** The game is always the best version of itself and there are no faults	**Priority:** 10 **Estimate:** 4 story points **Tests:** Try it with correcting faults that might occur Try to improve performance Try to adapt the software

Conversation:

Can the game run smoothly and without any glitches?

Can the code be corrected?

Happy day – The software developer is able to maintain the software and perfect its performance.

Rainy day – The software cannot be maintained properly and issues cannot be resolved.

SQL Query.txt

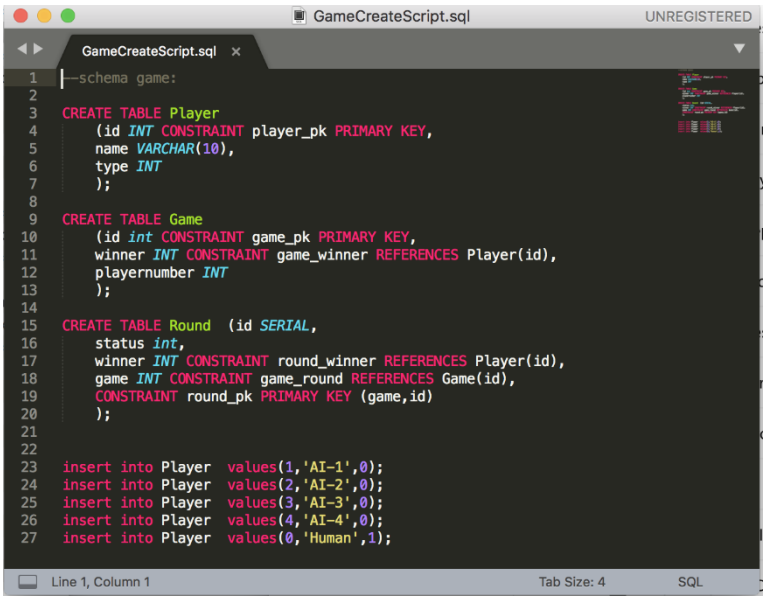
```
• How many draws were there?
• Who won the game?
• How many rounds were played in the game?
• How many rounds did each player win?

INSERT INTO game (gameid, winner, numdraws, numroundsplayed, playerroundswon,"
                + " cpu1roundswon, cpu2roundswon, cpu3roundswon, cpu4roundswon) VALUES ('" + primaryKey+ "','" + winner
                + "', '" + numDraws + "', '" + numRounds + "', '" + p1RoundsWon + "', '" + cpu1RoundsWon + "', '" +
                cpu2RoundsWon + "', '" + cpu3RoundsWon + "', '" + cpu4RoundsWon + "');";


• Number of games played overall  ==> SELECT COUNT(GameID) AS gameCount FROM Game;
• How many times the computer has won  ==>  SELECT COUNT(Winner) AS cpuWinCount
                                                                        FROM Game WHERE Game.Winner = 'CPU 1'
                                                                        OR Game.Winner = 'CPU 2' OR Game.Winner = 'CPU 3'
                                                                        OR Game.Winner = 'CPU 4';
• How many times the human has won  ==> SELECT COUNT(Winner) AS playerWinCount
                                                                    FROM Game WHERE Game.Winner = 'Player 1';
• The average number of draws  ==> SELECT AVG(numDraws) AS avgDraws FROM Game;
• The largest number of rounds played in a single game  ==> SELECT MAX(numRoundsPlayed) AS largestNumRounds FROM Game;

CREATE TABLE Game (GameID INTEGER PRIMARY KEY, Winner VARCHAR(20),
                                    NumDraws INTEGER, NumRoundsPlayed INTEGER,
                                    PlayerRoundsWon INTEGER, CPU1RoundsWon INTEGER,
                                    CPU2RoundsWon INTEGER, CPU3RoundsWon INTEGER, CPU4RoundsWon INTEGER);

sample of insert into;
INSERT INTO Game (GameID,Winner,NumDraws,NumRoundsPlayed,PlayerRoundsWon,
                                    CPU1RoundsWon,CPU2RoundsWon,CPU3RoundsWon,CPU4RoundsWon)
                                    VALUES (1,'CPU3',31,16,21,18,6,13,17),(2,'CPU2',2,95,41,37,15,6,16),
                                    (3,'Player1',3,7,24,17,8,12,25),(4,'CPU2',13,45,3,2,13,16,10),
                                    (5,'CPU4',4,21,5,44,11,2,21),(6,'CPU1',29,59,10,26,17,1,1),
                                    (7,'CPU2',35,17,2,6,20,20,1),(8,'Player1',31,69,20,17,12,9,12),
                                    (9,'Player1',30,41,38,40,3,7,11),(10,'CPU1',43,100,19,32,19,20,22),
                                    (11,'CPU4',42,45,1,50,9,19,18),(12,'Player1',20,67,15,38,8,2,11),
                                    (13,'CPU1',23,37,50,20,24,18,17),(14,'Player1',50,73,44,11,18,16,17),
                                    (15,'CPU2',30,54,15,41,22,10,24),(16,'CPU4',15,39,19,17,29,19,20),
                                    (17,'CPU4',41,80,10,1,12,3,4),(18,'CPU3',4,35,44,41,12,19,14),
                                    (19,'CPU4',36,64,44,38,16,12,8),(20,'CPU1',19,20,18,22,24,14,6),
                                    (21,'CPU2',40,98,32,23,18,4,2),(22,'CPU2',12,82,36,24,19,2,6),
                                    (23,'CPU4',14,42,3,22,2,18,14),(24,'CPU2',43,93,29,43,13,18,14),
                                    (25,'CPU2',36,87,33,17,28,12,21),(26,'CPU4',19,91,21,31,30,3,23),
                                    (27,'CPU2',5,49,37,41,25,6,18),(28,'CPU2',18,28,37,32,27,12,20),
                                    (29,'CPU1',42,95,26,10,18,14,7),(30,'CPU2',18,4,38,18,18,6,9),
                                    (31,'CPU1',39,33,13,9,22,4,6),(32,'CPU2',22,60,9,12,6,10,2),
                                    (33,'Player1',10,35,37,36,11,15,6),(34,'CPU4',38,63,35,20,5,16,21),
                                    (35,'CPU4',5,100,41,24,16,17,1),(36,'CPU3',47,57,4,39,28,1,16),
                                    (37,'Player1',49,84,43,41,16,19,14),(38,'CPU4',45,62,24,10,8,15,6),
                                    (39,'CPU3',14,96,23,12,15,19,22),(40,'CPU1',42,29,34,19,3,17,5),
                                    (41,'CPU1',2,26,11,31,7,20,12),(42,'CPU3',2,82,42,20,30,3,15),
                                    (43,'Player1',21,16,3,37,27,17,24),(44,'CPU4',7,72,4,39,29,14,11),
                                    (45,'CPU3',38,12,12,12,29,1,25),(46,'CPU2',19,89,5,5,18,6,10),
                                    (47,'CPU3',20,65,35,4,5,20,8),(48,'CPU2',19,27,35,47,7,20,3),
                                    (49,'CPU3',49,85,43,29,21,12,22),(50,'Player1',25,2,1,36,2,8,18),
                                    (51,'CPU3',28,85,21,18,1,20,11),(52,'Player1',18,49,6,34,29,3,11),
                                    (53,'CPU2',27,48,45,47,6,8,6),(54,'CPU4',8,86,2,23,20,11,3),
                                    (55,'CPU3',23,58,29,32,19,16,13),(56,'CPU1',42,77,40,27,29,16,10),
                                    (57,'CPU3',30,16,48,1,13,9,25),(58,'CPU3',31,50,43,5,27,6,2),
                                    (59,'CPU3',45,58,45,43,14,20,2),(60,'CPU3',19,58,13,28,3,16,14),
                                    (61,'CPU1',10,5,8,36,2,18,12),(62,'CPU4',39,77,45,9,18,6,22),
                                    (63,'CPU1',4,77,16,14,18,9,23),(64,'CPU1',12,24,8,38,7,4,22),
                                    (65,'CPU3',40,34,19,26,21,13,14),(66,'CPU1',6,37,25,49,1,19,10),
                                    (67,'CPU2',50,91,12,26,21,9,7),(68,'CPU3',35,14,43,17,18,16,3),
                                    (69,'CPU1',8,38,30,10,12,12,23),(70,'CPU1',27,28,10,5,30,7,22),
                                    (71,'CPU4',22,67,12,35,11,15,15),(72,'CPU1',32,53,7,5,28,16,2),
                                    (73,'CPU3',9,89,15,34,14,13,6),(74,'CPU4',21,23,21,44,27,13,6),
                                    (75,'CPU4',25,78,42,13,27,9,15),(76,'Player1',33,98,43,20,12,5,8),
                                    (77,'CPU1',38,78,30,48,20,12,6),(78,'Player1',25,6,50,25,1,4,12),
                                    (79,'CPU2',40,97,10,4,23,16,19),(80,'CPU4',42,30,31,5,27,6,12),
                                    (81,'CPU3',33,92,44,42,7,12,14),(82,'CPU3',3,99,29,42,18,12,25),
                                    (83,'CPU4',1,1,48,19,8,16,23),(84,'CPU3',22,68,11,9,30,3,3),
                                    (85,'CPU4',8,91,22,50,14,17,18),(86,'CPU3',49,45,29,27,18,18,8),
                                    (87,'Player1',33,3,28,7,14,15,23),(88,'CPU2',13,61,17,34,25,7,24),
                                    (89,'CPU3',13,29,31,38,5,20,2),(90,'CPU2',30,77,21,2,5,1,21),
                                    (91,'CPU4',19,9,34,43,11,4,3),(92,'CPU1',14,54,11,37,24,13,20),
                                    (93,'Player1',29,64,13,19,8,3,7),(94,'CPU1',12,8,10,18,20,6,7),
                                    (95,'CPU1',10,18,38,21,21,13,21),(96,'CPU1',50,61,37,36,18,2,1),
                                    (97,'CPU4',5,8,50,41,5,19,20),(98,'Player1',1,50,31,10,29,18,6),
                                    (99,'CPU2',28,54,34,5,2,20,5),(100,'CPU1',28,90,44,35,12,20,14);
```



ER Diagram

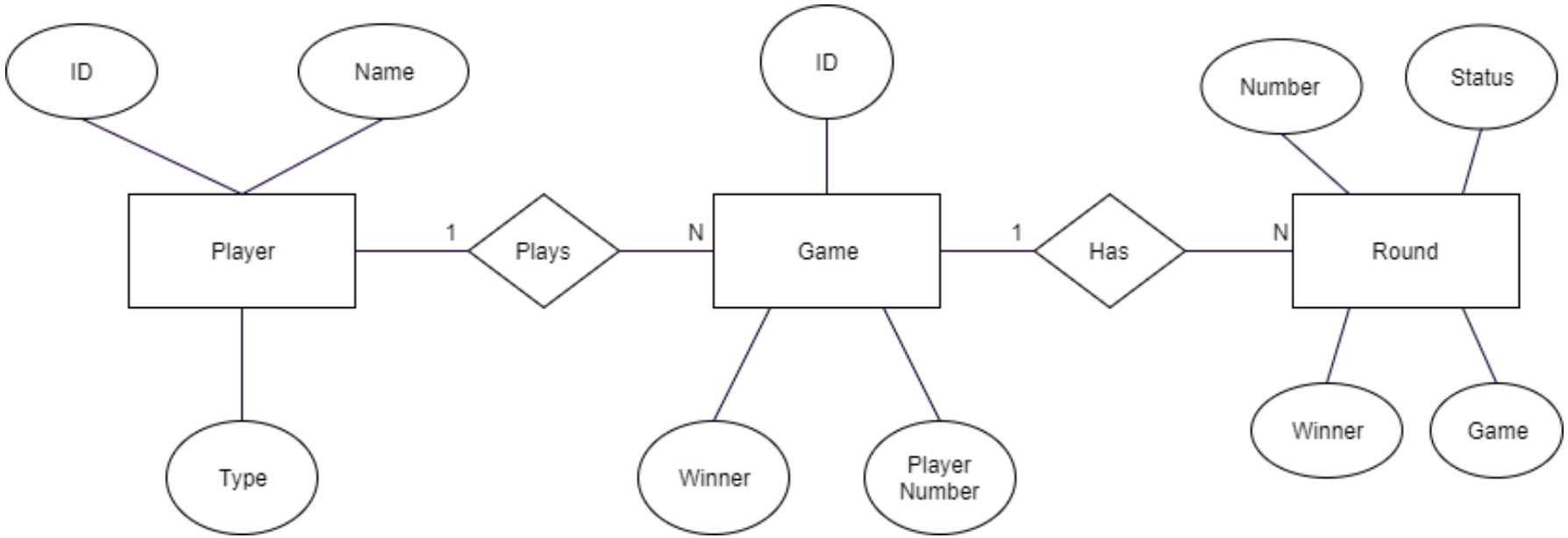


Figure 1 – ER diagram for the database of the Top Trumps game.

Figure 2 – Pre-CSS version of the online mode of the Top Trumps game.

Figure 3 – Pre-Bootstrap version of the online mode of the Top Trumps game

Styling example: result page

Figure 4 – Final Bootstrap version of the online mode of the Top Trumps game

Styling example: result page

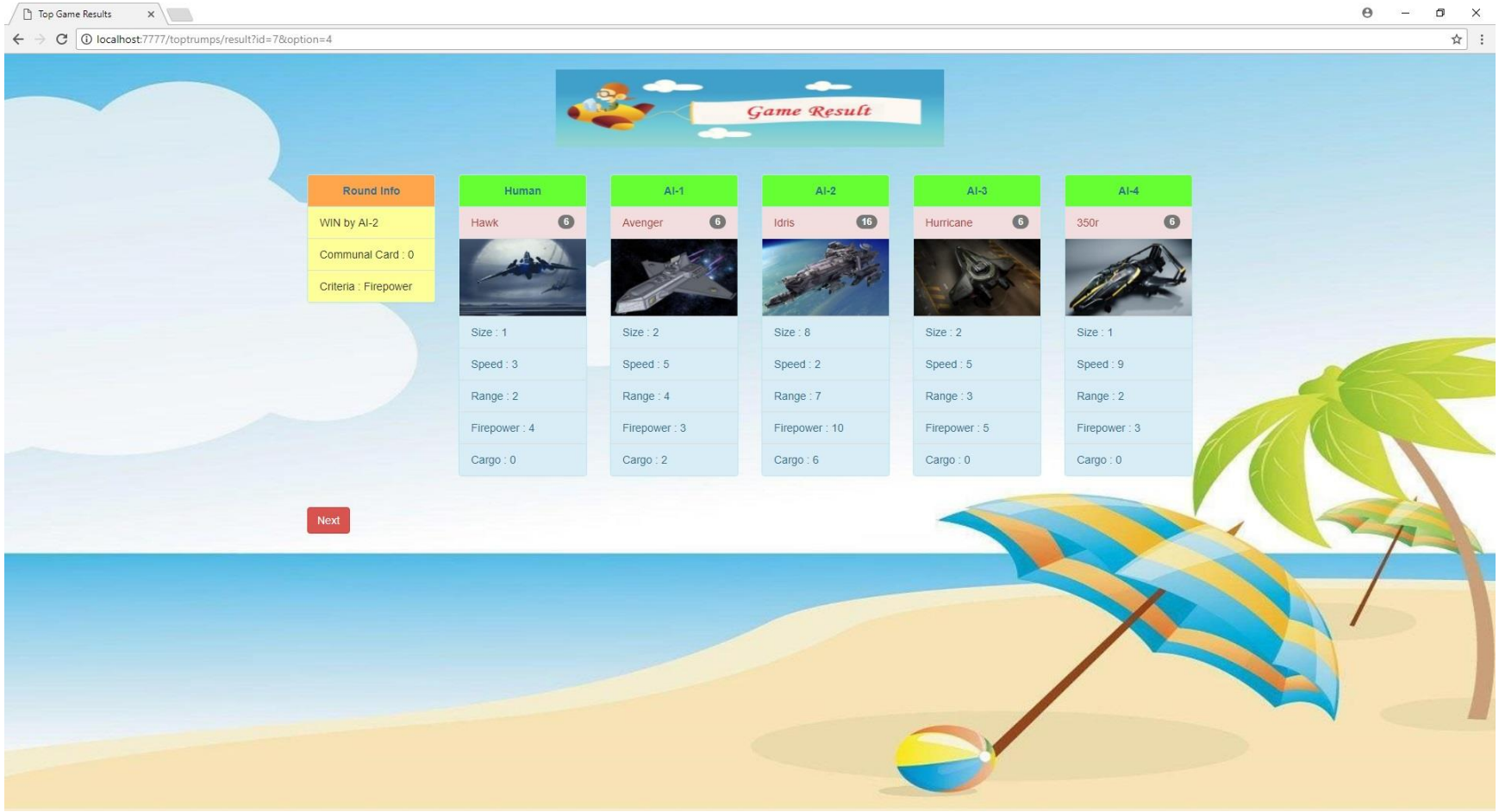
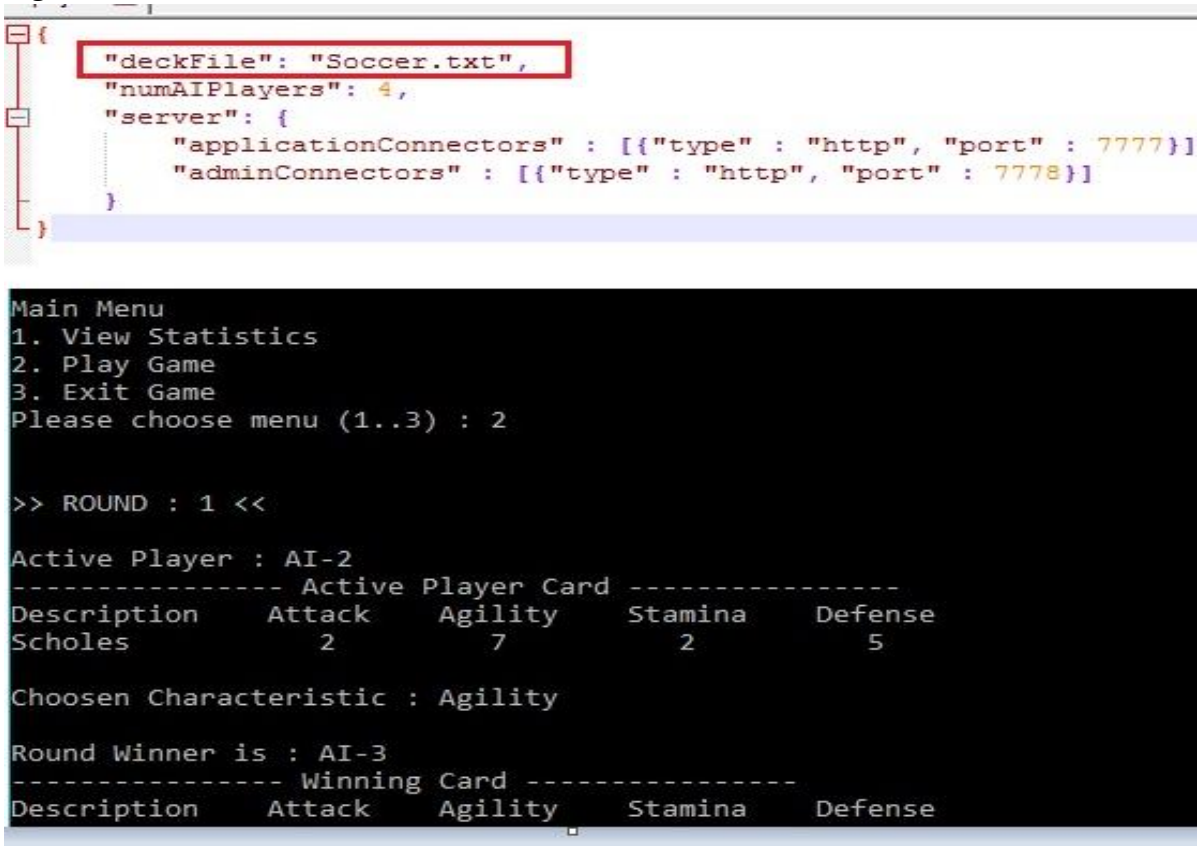


Figure 5 – Alternative deck – soccer.txt test



Bibliography

1. User Stories/Scrum/Agile Software Development/What is Agile Project Management.

   https://www.mountaingoatsoftware.com/agile/user-stories

   "As a < type of user >, I want < some goal > so that < some reason >."

   Mike Cohn

2. "It's often best to think of the written part as a pointer to the real requirement. User stories could point to a diagram depicting a workflow, a spreadsheet showing how to perform a calculation, or any other artefact the product owner or team desires."

   Ibid [1] Mike Cohn

3. User Stories Applied: For Agile Software Development Mike Cohn (Addison-Wesley Signature) 2004 (Page 7 Acceptance Tests, Page 11 Story Points)
4. To plan a release, the customer team starts by prioritizing the stories. While prioritizing they will want to consider:

- The desirability of the feature to a broad base of users or customers
- The desirability of the feature to a small number of important users or customers
- The cohesiveness of the story in relation to other stories. *Ibid[3] Mike Cohn Page 10*

5. Chris Nodder Consulting LLC, an agile user experience company
6. User Story Mapping https://www.scrumalliance.org/community/articles/2013/august/creating-an-agile-roadmap-using-story-mapping
7. Release Planning in an Agile Project by Mark C.Layton

http://www.dummies.com/careers/project-management/release-planning-agile-project/

TIP “It’s a good idea to achieve releases with about 80 percent of the user stories, using the final 20 percent to add robust features that will meet the release goal while adding to the product’s “wow” factor.” Mark C. Layton

8. Acceptance Tests http://www.fitnesse.org/FitNesse.UserGuide.AcceptanceTests
9. Scrum Guide http://www.scrumguides.org/scrum-guide.html
10. Scrum Master https://www.mountaingoatsoftware.com/agile/scrum/roles/scrummaster
11. Scrum Institute http://www.scrum-institute.org/Scrum_Roles_The_Scrum_Team