The Happy Cow Game

Report Name	Requirements Analysis
Author (User Id)	Simeon Smith (sis17)
Supervisor (User Id)	Amanda Clare (afc)
Module	CS39440
Degree Scheme	G401 (Computer Science (Inc Integrated Industrial And Professional Training))
Date	February 4, 2015
Revision	0.1
Status	Draft

Contents

1 Introduction

This document is an attempt to understand what tasks are needed in order to successfully turn the Happy Cow board game into a web-application. The web developer gathering the requirements is Simeon Smith (sis17@aber.ac.uk), and the game owner, the client, who will check these requirements and clarify concepts is Dr Gabriel de la Fuente Oliver (gfuente@prodan.udl.cat).

Any requirement marked '**Enhancement**' is an extra feature that can be supported if there is time. Additionaly a section of enhancements is provided, these will be implimented after all other requirements.

2 Scope and Purpose

The purpose of a web-application version of the Happy Cow Game is to make the game more accessible by providing a platform from which the game can be more easily played. Making the game accessible from the web gives several advantages. People do not have to create a physical version of the game to play it, but provided they have access to the Internet can play the game on any device. This in turn will lead to more users, who can provide feedback and suggest improvements, thereby improving the game. Finally a static version which anyone can play will help to solidify the rules of the current prototype board game.

The Happy Cow Game is intended for educational purposes. It's scope of use therefore extends from personal use to classroom or lecture theatre use. This affects the environment the game will be played in: screen size, device power, Internet connection and web browser type will differ substantially. One example is Internet Explorer. While being the least favourite browser for web application development, it continues to be used widely in classroom environments, so should be supported as much as possible.

There are two ways the Happy Cow Game application will be used: by players gathered around a single screen, or by players at their own machines interacting across a network. The game design needs to take into account both these scenarios.

3 Requirements

3.1 Users

This section covers registration and authentication of users. This section does not cover game play.

3.1.1 Players must register to have an account

- Players can register before they begin playing a game. This is full registration. Players must provide a name, to be used in game play, and email address, for account management, and a password.

- If players are being included in a group game (where they play around a single machine), they do not need to register. They must provide a name only. They can then play a single game with only a user name.

3.1.2 Players must login to authenticate themselves

- Players must be able to login to use their account and play any saved games. They will provide their email address and password.

- Players can request to have a new password emailed to them if they have forgotten their password.

3.1.3 Players can manage their details and statistics

- Players must be able to change their name and password.
- **Enhancement:** Players must be able to update their email address. This will require validating their new email address by sending a link to verify it is a real email address.
- Players can choose their primary and secondary colours for game play. These can be updated.
- **Enhancement:** Players have an experience level, determined by the number of games they have played, and the number of games they have won. This can be seen by other players.
- **Enhancement:** Players can see the number of games that they have played, and their score at the end of the game.

3.1.4 Player communication

- Players can send messages to other players within the context of a game only.
- Messages contain a sender, recipient, subject and content.
- Players can read messages and delete messages.

3.2 Game Management

This section covers creating and saving games. This section does not cover game play.

3.2.1 Group Games

- Anybody can create a group game, whether they have an account or not.
- Existing players can be searched for and added to the game. A login must be provided for the player to authenticate themselves if they have not already.
- Alternitavely a temporary player can be added, by providing a name and choosing a colour for that player.

3.2.2 Persistent Games

- Only a registered player can create a persistent game. The game is then in setup phase.
- The creator can search for other players and invite them to join the game.
- **Enhancement:** Players can choose to automatically ignore invitations from a certain other player.
- Other players can accept invitations, and then can see the game setup details, but cannot change them.
- Players can delete invitations, and leave a game at any point.
- Players can only join a game during setup.

- A game must have 2 players, and cannot have more than 5.
- The creator can choose the number of rounds the game will last, or select a range of rounds (for example 8-10).

3.2.3 Game Information

- Each game must have a record of the players who are part of the game, and their score.
- Each must store information about game play: cow welfare, cow body condition, cow PH levels, a muck marker, and an oligos marker.
- A game also records what state it is in: the round number, the phase of the turn, and which player is active.
- **Enhancement:** The game creator can decide which events and action cards will be used for that game. So each game could use a different selection of the possible decks of cards. This could change the difficulty of the game.

3.2.4 Game Initialisation

- The creator can choose to begin the game. From this point on they have no special privilidges.
- The game is changed from the setup phase to the play phase. Players and their colours are now set for the duration of the game.
- The cow information markers are set at their starting position.
- The starting order in which players take turns is randomly decided.

3.2.5 Games Board

- Each player can view a games board, showing all the games they are in, or have been involved in.
- Each game will show the players and score of the game.
- From here a player can select a game, and if it is in progress they are taken to the game play screen.
- **Enhancement:** If a game is finished, players can view statistics of the game. And a game history.

3.3 Rounds, Phases and Turns

This section covers the structure of game play, the order and consequences of actions.

3.3.1 Rounds

- The game begins in round 1.
- A game has 4 phases: Event Phase, Cards Phase, Movement Phase, Review Phase.

3.3.2 Phases

- The Cards Phase and Movement Phase each cycle through the users, each user takes a turn being active.
- A turn marker decides who goes first in the two phases. The turn marker is updated every round by passing to the last player from the round before.
- The Review Phase does not have active players, but each player must register that they want to move on to the next round before the phase ends.
- The Event Phase requires no action from the users, an event is selected at random.

3.3.3 Event Phase

- An event is generated, and the consequences of the event are carried out.
- Players are notified what the event is, and it's changes, when they next make and action in the game. Most likely this notification will be during the Cards Phase.
- If the number of rounds is set, and it is the last round, the event will be the slaughter house.
- If the number of rounds is within the ending range, but not at the end of the range, then the event generator will select the slaughter house, or another event, by chance.

3.3.4 Cards Phase

- The first active player is the one marked by the turn marker.
- The active player receives 2 new cards, selected at random from the deck of ingredients and actions. This information is presented to the player.
- During this phase, ingredient cards will have two possible actions: 'add to a ration' or 'discard'. Action cards will have two possible actions: 'perform action' (if appropriate during this phase) or 'discard'.
- The active player must be able to view his hand of cards. This will display characteristics of the cards, and their possible actions.
- The active player can build 1 ration per turn. This is done by selecting 1 to 4 ingredients from his hand. Once satisfied, the player can create the ration.
- When a player creates a ration they can no longer change it's contents. The ration is placed at the end of the queue of rations, on the board.
- If the player has less than 9 cards, they can end their turn. The next player in order then becomes active.
- If the player has more than 9 cards, they must first discard cards until they have 9.

3.3.5 Movement Phase

- The Movement Phase is initiated when the last active player of the Cards Phase ends their turn.
- The active player can carry out actions from cards in their hand by selecting the action in their deck. They cannot, however, build rations or discard cards.
- The active player must select one package from a list. They can view the position of each package, and then confirm their choice.
- The active player can move the game board and zoom in and out to get a good view of their packages and those of other players.
- When a player selects a package, the game board is moved and zoomed to provide a good view of the package.
- The active player is then presented with a result of 2 dice. They can select either result and see possible moves highlighted on the board. The player must then confirm their choice of dice.
- If the result of the dice is a double, these are combined into 1 choice.
- The active player must then move their selected package the selected amount of moves. The player does this by clicking on board squares adjacent to the package. The package follows their clicks until the amount of movements are used up.
- If the ration ends on the Milk or Meat squares, or exits the cow intestines players are notified of the consequences, and given a number of points. This must update the players' scores, and the cow information straight away.
- Once a player performs the last movement of their selected ration their turn is finished and they are notified it is the next player's turn.
- When the last active player finishes their turn, the turn marker passes the opposite way to play (to the last player), and it is the Review Phase.

3.3.6 Review Phase

- Because there may be some time between actions of players, and the movements of players are not seen as clearly as when everyone is gathered around a board, the Review Phase gives everyone a chance to see the actions of other players.
- Cards are not revealed unless they are used (for example, ingredients made into rations, or actions performed), otherwise they are kept secret.
- The order of events and consequences to the health of the cow are layed out in the order they happened. (for example: 'Player A moved a ration of 2 energy 3 spaces through the rumen. The PH of the rumen decreased by 2.')
- Once every player has confirmed that they have finished the review, the next turn begins.

3.4 Game Artifacts

This section covers special requirements of parts of the game that do not fit into game play order.

3.4.1 Cow Information

- The cow information will be visible on a side bar throughout game play.
- It will be updated automatically as players make actions.
- **Enhancement:** a small animation of (+1) or (-1) will float up from one of the cow information scales upon changes, to make them more noticable.
- Points awarded for rations turning into milk, meat or muck change, so must be shown here as a table, and updated.

3.4.2 The Feeding Trough

- Rations are added to the end of this when they are created.
- If there is no space in the trough (it has 6 spaces) no rations can be created.
- Only the front 2 rations can be moved out of the trough.
- When a ration is moved from the trough, the other rations move towards the front straight away. They can then be put into play by players who take their turn afterwards.

3.4.3 The Rumen

- There is no restriction on movement inside the Rumen.
- When a ration reaches the exit square of the rumen, it is automatically moved to the start of the intestines.
- There is one motile peice in the rumen. It is 4 squares long.
- The position of motile peice is updated at the end of a round. It can move between 0-3 spaces horizontally and/or vertically. So if a ration is within 3 spaces of it, it could be crushed. This means it will loose a random ingredient.
- The walls of the rumen are dangerous. If a ration is bumbped into the wall it will loose a random ingredient.

3.4.4 The Eusophegous and Intestine

- Rations can only move sideways, or forward in these 2 areas, they cannot move backwards. On bends, where indicated by a thick line, the rations can only move forwards.
- If a ration ends on the Milk squre the ration gets absorbed and the owner gets points. 6 for energy, 3 for protein, 2 for fibre, 1 for water. Oligos begins at 10 points, but reduces to 2 points and then 0.
- If a ration ends on the Meat squre the ration gets absorbed and the owner gets points. 2 for energy, 4 for protein, 2 for fibre, 1 for water. Oligos begins at 10 points, but reduces to 2 points and then 0.
- If a ration exits the intestine the ration is destroyed as muck and the owner gets 1 points for each ingredient but water. The muck marker is increased

3.4.5 Special Ingredients

- If a ration contains at least 1 water, when moved the player gets to chose from 3 dice. If all three dice show the same number, the amount is added, and they get no choice.

- If there are more water than energy in the rumen at the end of a turn, the ph in the rumen increases by 1. If there are more energy than water it decreases by 1.

- If a ration has fiber it can block other rations with less fibre. This means they cannot pass it. When moving the ration can also push one other ration with less fibre. This means it moves the ration in the opposite direction it came from. A ration can only push 1 other ration a turn. The other ration moves the difference of fibre between it and the attacking ration. For example, a ration with 3 fibre can push a ration with 1 fibre for 2 spaces.

3.4.6 Alerts

- When an action succeeds a green alert will notify the user. This includes creating a ration, performing an event, having a ration absorbed, pushing a ration.

- When an event is blocked a red alert will notify the user, with information describing why they can't perform that event. This includes moving over a blocking ration.

- When the game want to give warning advice to the user an orange alert will be used. This includes notifying that it is the final turn, or that the cow is dangerously unhealty, or having one of your rations pushed.

3.4.7 Messages

- Players can send a message at any time. They can also read and delete messages at any time.

3.5 Testing

This section concerns requirements of how the game will be tested.

3.5.1 Code Tests

- Unit tests will be performed on all server-side model objects. So if Ruby on Rails is used, Unit tests will cover the ORM.

- Unit tests will be performed on the client-side javascript controllers. Using a client side MV* Framework will make this possible.

- Functional tests will also be performed on server and client side. This is end-to-end testing. On the server side it will cover the possible uses of the RESTful API. On the client side it will cover possible and expected user actions.

3.5.2 Usability Tests

- Usability tests will be performed by developers playing the game.

- A bug report system will be set up to manage feedback about problems with the application.

- People will be asked to test the game and fill in a form so that developers can get feedback. This will ask questions about the overall ease of usability, and ask them to identify weak or confusing areas.
- Usability tests will consider the web-application, but also improvements to the game itself.

3.5.3 Browser Tests

- The web-application must operate as expected in both Firefox and Chrome.
- The web-application must be playable in Safari and the newest Internet Explorer. However, certain work arounds may be necessary to get the detailed functionallity to work. IE is important as it is widely used in teaching envrironments.
- For old version of IE, if the web-application does not work in them, the system must gratiously say so and suggest using a different browser.

3.6 Enhancement Features

These requirements will be added if those above are completed. They will make the game more playable, and dynamic, but are not core features.

3.6.1 Change Cards

- Players will be able to create cards and select which cards to play a game with. The cards will be the same for each player.
- Certain cards have quite complex results (eg: constipation). These must have hard coded actions. Other cards have a description and only effect one or two recognisable objects, such as increase welfare. It is this second type that users will be able to create and add themselves.
- A card making page will be accessible by users with accounts. The use will be able to combine pre-configuered events with results, and quantities. For example: stealing a card, at the cost of discarding 2 cards. Or moving the PH marker to 0 and gaining a point.
- When satisfied with the card, the player can create it. It is then available for anyone to use, but will not be in the default card deck.
- Players can group cards that they like into decks, and select the probability of occuring. This can be done when creating a game, but decks can also be saved, so that they can be re-used. A page for organising card decks will be needed.
- Players can also rank a card from 1 to 10 to indicate how good it is for game play.

3.6.2 A Tutorial

- This would be a demo game, with many more instructions as alerts to notify the new player of possibilities such as viewing their cards, sending messages, and how ration movment works.
- The player would have to take the tutorial alone, so a computer player is needed. The artificial player does not need to make decisions, as the cards they have and the actions of the artificial player will be pre-determined. So every tutorial will be similar, changes will only come from the new player's actions.

- The tutorial would take the new player through 3 rounds. In the first round the location of game objects are pointed out, and the necessary actions are explained. The second round teaches the player about the point of the game, continuing to give help as to the location of game objects. The third round teaches about some of the strategy of the game, and different options, such as blocking and pushing rations.
- As well as a demo-tutorial, help pages will be written about diffetent subjects. Links can be provided during game play to these help pages for players to find out more information.
- A player can choose to turn off help links and alerts in their profile.

3.6.3 A Feedback Form

- This page will not show on the game play screen, but a link will be available from other pages, and when a game is finished. This allows a player to give feedback when they log in or out, or review their profile or the games they are involved in.
- The player will be able to make general comments in a text area, or a specific comment by choosing an area of the game form a list.
- The comments, suggestions, or bug reports, will be stored on a back end where developers maintaining the site can see them, and delete them. The email of the sender will be stored so developers can ask for more information.