Development plan:

Oscar:

1. Creation of the Board class
2. Creation of the Player class
3. Joint Creation of the Game class

The board class is vital for the operating of all other classes, so it is important to be created as early as possible in the development process. The player class is also important for the game, but other classes rely less on it so should be completed afterwards. As both classes are large in scope it is sufficient work to be completed by one person.

Sam:

1. Creation of the Dice class
2. Creation of the Animal class
3. Joint creation of the Game class

The Dice class is relatively small and is important for some other classes so should be completed early in the development process. The animal class is large and important for the functionality of the game so should have a lot of time dedicated to it and be completed after the dice class. As this is one large class and two smaller classes this is a sufficient amount of work to be completed by one person.

Isaac:

1. Design and planning of the GUI
2. Creation of the GUI
3. Creation of the Cards class
4. Implementing the GUI with the events of the game

A graphical user interface is highly important for the accessibility of the game and conveying information in an engaging and understandable way but as it is a large task which involves creating a graphical design for each feature it is a sufficient task to be completed by one person. The Cards class is important to meet success criteria and improve the game, but few other classes rely on it so should be completed towards the end of the development process.

Once all individual classes are made Oscar and Sam will collaborate to create the Game class which controls the events of the game and allows the classes to work together. Additionally, Issac’s GUI made using swift will be made to work with the game to update the screen when the game changes.

Post development test plan:

|  |  |  |  |
| --- | --- | --- | --- |
| Test description | Test Type | Success criteria | Pass/Fail |
| Roll two dice and check the number generated. | Normal | Two dice successfully rolled and the result is between 2-12 |  |
| See if the total number of spaces to move after the dice roll is displayed to the player. | Normal | The total number of spaces to move will be displayed |  |
| Check if there is a double roll then a special event will occur e.g. a card being drawn. | Normal | A special event occurs after a double roll |  |
| See if there is a board with 26 spaces and that it is displayed onto the screen. | Normal | A 26 space board will be displayed to the users |  |
| Roll the dice a few times and examine the players’ positions. | Normal | The players’ positions will be stored in the board and updated as the game goes on |  |
| Roll the dice a few times and check if the players’ positions on the board are visible. | Normal | All of the players’ positions are visible on the board at all times |  |
| Land on a space and view all information given to you about the current space. | Normal | All relevant rules and actions which can be taken on the current space will be presented on the screen. |  |
| Roll the dice enough times to pass start. | Normal | £500 will be given to the player who passes start. |  |
| Roll the dice to exactly land on start. | Normal | £1000 will be given to the player who lands on start. |  |
| Roll the dice to land on the miss a turn space | Normal | The player who lands on the miss a turn space will miss a turn. |  |
| Select unique playing pieces for each player. | Normal | Each player will be allowed to select a unique playing piece to be used during the game. |  |
| Input a player name and play the game as that player. | Normal | The player’s name will be stored and displayed at the start of each of their turns. |  |
| Play the game as a player and view the amount of money owned throughout the game. | Normal | The player’s money amount will be initialised and updated due to various events throughout the game |  |
| Play the game and land on an animal space which is owned by another person. | Normal | The player’s money is decreased and given to another player. |  |
| Play the game and attempt to purchase an animal which cannot be afforded. | Erroneous | This should not be allowed as this would cause the player to become bankrupt intentionally. |  |
| Play the game and be forced into bankruptcy. | Normal | The player should be notified that they are bankrupt and removed from the game. |  |
| Play the game and become the last remaining player with money. | Normal | The player should be announced as the winner and the game should end. |  |
| View the amount of animals on the board. | Normal | There should be 24 unique animals across the board with different attributes/interactions. |  |
| Check the amount of information about each animal. | Normal | There should be information about name/species, level, cost to land on, cost to buy, owner, etc stored about each animal. |  |
| Land on the space of an unowned animal you can afford and attempt to purchase it. | Normal | You should be able to purchase animals on different spaces on the board at their set cost. |  |
| Attempt to upgrade the level of an animal you currently own. | Normal | You should be able to upgrade an animal (which increases the cost to land on it) for the same price as the animal originally cost. |  |
| Check the cost to land on an animal space owned by another player. | Normal | The players should be able to check the cost of landing on another player’s animals during their turn. |  |
| Check that there are 20 cards generated to begin the game. | Normal | There should be 20 cards generated, each detailing a random scenario which has some effect on the game likely money gain/loss |  |
| Roll 2 dice until you get a double. | Normal | A random card from the deck should be selected and its effect used on the player. |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |