Requirements document

**System Requirements**

**Section 3 & 4 : The rules of property Tycoon and other**

Functional requirements

|  |  |  |
| --- | --- | --- |
| Requirement | Rational | Source Quote |
| The game shall have a minimum of two players. | See quote | “The game is for 2-6 players.” |
| The game shall have a maximum of six players. | See quotes | “The game is for 2-6 players.”  “There is no particular reason why it can't be more than 6, but in practice we found that with more than 6, the chances of any player actually winning was too low and made the gameplay boring. So 6 will remain the maximum number of players.” |
| The game shall require each player to have a token representation in game. | See quote | “Each player is assigned one of the game tokens.” |
| The game shall allow the player to choose their token or have it randomly assigned. | We decided to not allow players to pick their token, but give them a random one. | “Q) Are tokens allocated to players randomly or can players choose? A) It would be nice to be able to choose (you can in real life). But equally a random option is useful when players don’t care.” |
| The game shall have the following tokens: boot, smartphone, goblet, hat stand, cat, spoon. | See quote | “The tokens are: boot, smartphone, goblet, hatstand, cat and spoon.” |
| The game shall have an infinite amount of money available to the bank. | See quote | “The bank is always able to pay the players. In the board game version, the bank can issue IOUs or generate new notes to ensure that game play can continue.” |
| The game shall not allow the players to borrow money from the bank. | See quote | “Players may not borrow additional money from the bank” |
| The game shall allow the players to sell properties to the bank. | See quotes | “they (referring to players) can trade game items with the bank”  “If a player is unable to pay the rent for a property they have landed on, they must sell game assets to make good on the rent.” |
| The game shall shuffle the two packs of cards at the start of the game. | See quote | “At the outset of the game, the two packs of cards labelled “pot luck” or “opportunity knocks” are shuffled and placed on the board.” |
| The game shall put used cards onto the bottom of the appropriate deck. | See quote | “When cards are taken, they must be replaced at the bottom of the corresponding pile.” |
| The game shall have the turn player roll two dice on their turn to show the distance they will move. | See quotes | “Each player takes a turn by rolling two dice to determine how they move around the board.”  “For each turn, the player rolls the two dice. They move the number of spaces shown on the dice and arrive at a board space.” |
| The game shall move the player clockwise a number of squares equal to their dice roll. | See quotes | “all players start on the board space labelled Go and move clockwise around the board.”  “Players move clockwise around the board.” |
| The game shall give the player an additional turn if they roll a double on their turn. | See quote | “If a player throws a double, then they take another turn.” |
| The game shall send a player to jail if they roll three consecutive doubles. | See quote | “If a player throws another double at the third turn, then they “go to jail”” |
| The game shall be able to send a player to jail: the player's token is moved to the "in jail" square and does not pass go. | See quote | “When a player goes to jail, they go directly and do not pass Go” |
| |  | | --- | | The game board shall consist of spaces which can be: properties, pot luck, opportunity knocks, | |  |   free parking, jail/just visiting, instructions. | See quote | “Board spaces may consist of properties, a “pot luck” space, an “opportunity knocks” space, “free parking”, the jail/just visiting space or a space with specific instruction” |
| The game shall present the player with the top card of the pot luck or opportunity knocks deck when the player lands on the space associated with it and then carries out the action listed on the card. | See quote | “If a player lands on a “pot luck” or “opportunity knocks” space, they take a card for the top of the corresponding pile and carry out the instructions on the card” |
| The game shall carry out the instructions listed on an instruction space when the player lands on it. | See quote | “a space with specific instructions that must be followed by the player.” |
| The game shall ensure that no player may purchase a property until they have completed a complete circuit of the board. | See quote | “Players may not purchase property until they have completed one complete circuit of the board by passing the Go space.” |
| The game shall ensure that when a player passes the 'Go' space, they receive £200 from the bank. | See quote | “When a player passes Go, they receive £200 from the bank.” |
| The game shall ensure that no player owns any of the properties at the start of the game. | See quote | “All properties are initially the property of the bank.” |
| The game shall transfer money from a player to the bank when they purchase an un-owned property and that property is transferred from the bank to the player. | See quote | “When a player purchases a property, the card is transferred from the bank to that player and the amount shown on the card is paid to the bank.” |
| The game shall ensure that if a player lands on a space that contains a property that is not yet purchased, they have the option to by that property. | See quote | “Once a player has made their move, if they land on a property that has not yet been purchased, they have the opportunity to buy that property.” |
| The game shall auction a property if it is not yet purchased and the player who landed on the corresponding space did not buy that property. | See quote | “If they decide not to buy that property then the property is auctioned by the bank.” |
| When a property is auctioned, each player has the ability to make a bid. | See quote | “…then the property is auctioned by the bank. Each player makes a bid to the bank.” |
| The bank must sell an auctioned property to the highest bidder. | See quote | “The bank sells the property to the highest bidder.” |
| The game shall leave auctioned properties unsold if there are no bidders. | See quote | “If there are no bids, then the property remains unsold.” |
| The game shall transfer money from the highest bidding player to the bank equal to their bid upon winning an auction. | This is inferred by the bank selling to the highest bidder | “The bank sells the property to the highest bidder.” |
| The game shall only allow players to use their currently held money at the time m | See quote | “A player cannot sell assets outside of their turn. Bids have to be made using cash assets in their possession at the time the bid is made.” |
| The game shall run on both Windows and Mac, if not possible, Windows as priority. | See quote | “The electronic version should be for desktop machines, and ideally should be playable on both Mac and PCs. If this is difficult, then PC development should be preferred. “ |
| The game shall charge a player rent equal to the value shown on a property card for any property a player lands on which another player owns | See quotes | “If a player lands on a property owned by another player, they must pay the player who owns the property the value of the rent shown on the card.”  “If a property is improved with houses or hotels, then the rent to be paid is as shown on the card.” |
| The game shall double the rent charged for a property if the owner owns all other properties in its group and they are unimproved | See quote | “If a player owns all of the properties in a colour coded group, but the properties are otherwise not developed further with houses and hotels, then the rent due is double” |
| The game shall make players sell game assets to pay rent if they are unable to pay. If the rent total is still not reached, they are bankrupt and removed from the board. | See quote | “All rents must be paid for in cash. If a player is unable to pay the rent for a property they have landed on, they must sell game assets to make good on the rent.” |
| The game shall declare a player bankrupt and remove their token from the board if they cannot afford to pay rent after the sale of all assets | See quote | “If they are unable to pay the rent after selling all of their game assets, then they are bankrupt and must leave the game. Their game token is then removed from the board.” |
| The game shall not allow money borrowing/lending between players, and between a player and the bank | See quote | “Players may not borrow or lend money from each other, and may not borrow money from the bank.” |
| The game shall only allow improving properties when a player have finished moving their token, and completed any property purchase. | See quote | “When a player has finished moving their token, and has completed any property purchase activity, they have the option to buy houses and hotels to improve their properties. Players are not permitted to improve their properties at any other time.” |
| The game shall only allow house/hotel purchases on a property where the player owns the whole colour group | See quote | “Houses and hotels may only be purchased for properties where a player owns all of the properties in a particular colour coded group.” |
| The game shall not allow a player to sell a property whilst it still has houses/hotel on it. | See quote | “A property can only be sold when there are no houses or hotels on the property.” |
| The game shall only allow purchases of houses as long as there is not a difference more than 1 house between all the properties in the set. | See quote | “there may never be a difference of more than 1 house between the properties in that set” |
| The game shall treat a hotel as the equivalent of 5 houses in terms of cost and improvement limitations. | See quote | “the maximum difference between improvements on one colour group is one house. So you can have a hotel on one, and four houses on the others. … To buy a hotel costs effectively the price of 5 houses” |
| The game shall only allow one hotel on any given property. | See quote | “The maximum development permitted on any one property is one hotel.” |
| The game shall allow the raising of funds for a player by mortgaging their properties, the bank provides half of the original cost. | See quote | “If a player needs to raise funds, they may mortgage a property with the bank. The bank will pay the player one half of the value of the property as shown on the game card” |
| The game shall not allow any rent to be collected while a property is under mortgage. | See quote | “No rents may be collected for that property whilst it is under mortgage.” |
| The game shall allow the player to sell a mortgaged property to the bank, sold for half the original property price shown on the card. | See quote | “If a mortgaged property is then sold back to the bank, it is sold for one half of the property price as shown on the card.” |
| The game shall put any fines paid into the free parking space in the centre of the board. | See quote | “Where fines are to be paid, the proceeds accumulate on the free parking space in the centre of the board.” |
| The game shall reward a given player with all money in free parking when they land on it. | See quote | “When a player lands on free parking, they collect all of the funds currently on the free parking space.” |
| The game shall allow a player to exit jail by paying £50. This £50 is added to free parking. This can be done in the same turn as the player is sent to jail. | See quotes | “If a player is sent to the jail, they may pay £50 to be released from jail. The £50 is added to the free parking fines.”  ”The £50 may be paid immediately.” Clarifications 11 March |
| The game shall move a player who has exited jail to the 'just visiting' space and the player turn ends. | See quote | “The player token is then moved to “just visiting” and the players turn ends. The player takes a normal turn in the next round” |
| The game shall allow a player to stay in jail, giving up their turn for the next 2 rounds. | See quote | “If a player opts to stay in jail, they give up their turn for the next 2 rounds.” |
| The game shall not allow rent to be collected while a player is in jail. | See quote | “Whilst in jail, a player may not collect any rents from other players.” |
| The game shall move a player to 'just visiting' after two rounds have passed since they opted to stay in jail. | See quote | “At the end of the next 2 rounds, the player token is moved to “just visiting” and the players turn ends. The player takes a normal turn in the next round.” |
| The game shall allow the player to exit jail and move to 'just visiting' if they have a 'get out of jail free' card. The player turn ends, and the card is then placed at the bottom of the appropriate pile. | See quote | “If a player has a “get out of jail free” card, then they place the card at the bottom of the “pot luck” or “opportunity knocks” pile as appropriate, the player token is moved to “just visiting” and the players turn ends. The player takes a normal turn in the next round.” |
| The game shall allow a player exiting jail to take their turn as normal the next round. | See quotes | The player takes a normal turn in the next round.” x3 |
| The game shall begin with each player having £1,500. | See quote | “At the outset of the game, each player has £1,500 in cash.” |

Non-Functional requirements

|  |  |  |
| --- | --- | --- |
| The game shall be 'fun to play', and a 'colourful and intuitive' UI, reflecting that of the original game. | See quote | The game should be fun to play and have a colourful and intuitive interface that reflects the spirit and character of the original board game. |

**Section 5: Core elements of the software system**

Functional Requirements

*Requirement Rational Source Quote*

|  |  |  |
| --- | --- | --- |
| The software shall be able to have one or more players in the game controlled by the autonomous agent. | This will allow games to be played with a more optimal amount of players when there are not enough people available to play and so enhance the gaming experience. | “An agent that can take the role of 1 or more of the players.” |
| The software shall be able to have all players controlled by the autonomous agent. | This will provide a means of testing our software. | “However, it also provides the possibility for fully autonomous play when all of the players are provided by the program.” |
| The software shall be able to import initial data for the pot luck and opportunity knocks cards. | This allows the game to be more customizable and allow Watson games to change it as they see fit. | “.....a means of initialising it with data on the board layout, the “pot luck” and “opportunity knocks” cards and the details about the various properties (the data that is currently on the cards in the physical version of the game.” |
| The software shall be able to import initial data for properties. | This allows the game to be more customizable and allow Watson games to change it as they see fit. | “As this data will be loaded on start-up from external files….” |
| The software shall be able to display current worth of each player at any time. | A vital component of the game. | “…..current worth of each of the players and the property assets that they own. This should be available for all to see as it is the current board game version.” |

Non - Functional Requirements

*Requirement Rational Source Quote*

|  |  |  |
| --- | --- | --- |
| The software should be able to run at high speed when all players are autonomous. | Ensures the game is enjoyable. | “Such simulations could be performed at high speed.” |
| The software shall be easily customised. | Will give users more freedom to enhance the gaming experience. | “….this means that the game is easily customised….” |

**Section 6: The game player agent**

Functional Requirements

*Requirement Rational Source Quote*

|  |  |  |
| --- | --- | --- |
| The autonomous agent shall be capable of making any choices that the user is able to make. | The purpose of the agent is to be an effective substitute for an actual player. | “The game player agent should be able to play the game to the same extent that a human player would.” |
| The autonomous agent shall not be able to stop playing before it is bankrupt. | Has the potential of ruining the game. | “A game player agent may not opt to retire from the game.” |

**Section 7: Integrity of the game**

Functional Requirements

*Requirement Rational Source Quote*

|  |  |  |
| --- | --- | --- |
| The software shall ensure that the bank has an infinite amount of money. | The bank must always be able to operate otherwise the game would end early. | “The bank is always able to pay the players. In the board game version, the bank can issue IOUs or generate new notes to ensure that game play can continue.” |
| Players shall not be able borrow or lend money to each other. | Could decrease the competitiveness of the game. | “Player may not borrow or lend money to one another.” |
| The software shall ensure that properties are never allocated to a player if their assets aren't decremented accordingly. | A vital component of the game. | “All assets procured from the bank must be paid for in cash. The bank does not provide credit.” |
| The software shall ensure that the probability of the dice producing a number between 1 and 6 is the same. | A vital component of the game. | “The dice used in the game must be fair with each dice have an equal probability of landing on one of its six sides.” |

Non - Functional Requirements

*Requirement Rational Source Quote*

|  |  |  |
| --- | --- | --- |
| The software shall ensure that the user interface displays all information about properties owned by a player or available. | Allows players to track their progress relative to other players and plan their next move. | “The range of properties available for sale by the bank, and owned by players, is a matter of public record and that information must be available to all players at all times.” |

Player API

* Shall contain a variable of type Integer balance
* Shall contain a variable of type Integer position
* Shall contain a variable of type character player\_character
* Shall contain a variable of type Properties Array property
* Shall contain a variable of type Boolean jailed
* Shall contain a variable of type Cards Array player\_cards
* Shall contain a variable of type Boolean isAuto
* Shall contain a variable of type Boolean passedGo

|  |  |  |
| --- | --- | --- |
| Variable | Description | Get/Set |
| Integer balance | The current balance of the player | Both |
| Integer position | The current position in relation to the board tiles of the player | Both |
| Character player\_character | The player token of the player, e.g. boot | Both |
| Properties Array property | An array of properties owned by the player | Both |
| Boolean passedGo | Return whether the player has passed go or not | Both |
| Boolean jailed | A Boolean to return whether the player is in jail or not | Both |
| Cards Array player\_cards | An array od cards currently owned by the player | Both |
| Boolean isAuto | Return whether the player is an AI agent or not | Both |

The player API should contain, at minimum the following functions:

|  |  |
| --- | --- |
| Function | Description |
| Move(int i) | Increase the players position by i if they are able to move |
| Roll() | Roll two dice and return the value rolled |
| increaseBalance(int i) | Increase the players balance by i |
| decreaseBalance(int i) | Decrease the players balance by i |
| getBalance() | Return the players current balance |
| addProperty() | Add a property to the players Array of properties |
| getProperty(Properties p) | Return whether the player owns passed property |

Game API

* Shall contain an array of type Player players
* Shall contain an array of type Space space
* Shall contain a variable of type Dice a
* Shall contain an array of type Properties properties
* Shall contain a variable of type Integer parking

|  |  |
| --- | --- |
| Variable | Description |
| Player Array players | An array to store the current players in the game |
| Space Array space | An array to store every ‘space’ on the board, the information about it |
| Dice a | Shall store an instance of the Dice class used to return dice values |
| Properties Array properties | An array to store all the properties in the game |
| Integer parking | An integer which holds the current value of the free parking fines |
| Integer turn | The index of the player in the players array who’s turn it currently is. |

At minimum, the following functions should be present in the Game API

|  |  |
| --- | --- |
| Function | Description |
| New\_turn(Player player) | Sets the turn variable to be that of the passed Player |
| Create\_space() | A function to create all the spaces on the board. Will read from excel files to do so. |

Cards API

Testing Strategy

We will perform both unit testing and systems level testing:

1. Unit testing – We will use the requirements document to test our methods with suitable inputs to make sure the output matches what the corresponding requirement specifies.

2. Systems testing – Against the requirements document, we will be testing the system as a whole using the UI to make sure everything is working as intended. Specified inputs will be passed to the UI multiple times to make sure the expected output is given every time.