|  |  |
| --- | --- |
| 1. **Summary Data** | |
| **Team Number** | 15 |
| **Sprint Technical Lead(s)** | Thomas Senyah |
| **Sprint Start Date** | 02/04/20 |
| **Sprint End Date** | 10/04/20 |

|  |  |
| --- | --- |
| 1. **Individual Key Contributions** | |
| **Team Member** | **Key Contribution(s)** |
| Thomas Senyah | The focal point of this sprint was to introduce chance and opportunity knock card stacks into the game. Along with this, reading data in from Excel documents would be implemented and rent and mortgaging functionality would also be added to the property functions developed previously in Sprint 2. Thomas again completed this code himself. |

|  |
| --- |
| 1. **User Stories/Task Cards** |
|  |

|  |
| --- |
| 1. **Requirements Analysis** |
| **Functional Requirements:**   * **FR-10:** As requested by Watson Games, none of the players should be assigned the role of banker and rather the program itself is assigned and handles the role of banker. This way the players can continue on and enjoy the game. The banker should be given the ability to distribute the correct amount of cash to each player throughout the game, via access to the “Bank”. The “Bank” will have a total of £50000 to ben used by the banker during the game. The Bank CANNOT lend out money to be borrowed from players. * **FR-12:** From the very start of the game there will be two stacks of cards, shuffled/randomised and then placed on the game. This would be two packs, with both outcomes visible on the game board for the players to see. For every card selected from either pile, a replacement is given to said pile and placed at the bottom of the corresponding pile. These cards will include instructions as well as property (Refer to FR-18 for more information relating to the property aspect of some of the cards). * **FR-15:** In the event that a player lands on a space with properties; “pot luck” or “opportunity knocks” (please refer to FR-14 for more information), then the player will be required to take a card from the top of the pile and carry out the instruction on the card. No other action should override this process when playing. When this is done the second process of FR-12 should be followed, where after a card has been selected, it is then replaced at the bottom of the corresponding pile. Only then can the next player take the next turn. * **FR-21:** Constraint for other players at play here, whereby, if a player lands on a property space, already owned by another of the 2-6 players, they must pay “Rent” to the owner of the property. Thus having “rent money”, if you will, deducted from the player’s cash amount. * **FR-23:** Any player holding all the of one particular colour (e.g. Player 1 owns all property of colour Red) with no “improvements” made onto them will enable a doubling of rent for the other players to pay when in conjunction with FR-21. However, in the event that improvements have been made the any of the player’s owned property (again, of that colour) then the rent shall remain the same as originally assigned. Please note, houses and hotels may only be purchased for properties where a player owns all of the properties in a particular colour coded group. * **FR-24:** ALL "rent” due from players must be paid, and thus deducted from, their cash amount. In the event that a player does not have enough for the "rent” then whatever game assets are in their possession must be sold to the property owner that is expecting rent. This will be used to cover whatever is owed in cash. If, still after selling ALL assets, the player is still in debt to the property owner then they must be classified as “bankrupt” and thus ejected from the game. In doing so their game token is also removed. * **FR-26: *(a)***Continuing on with property-owning players, if such a player needs to raise money, they should be able to do so by selling one of their properties to the bank for its original purchasing prices. ***(b)*** Another option for a player raising funds would be mortgaging a property with the bank. ***(c)*** The bank will pay the player one half of the value of the property as shown on the game card. **No rent money will be collected** for that property while it is under a mortgage. Taking FR-23 into consideration for this, there must be no additional improvements on the property, I.e. there should not be ANY improvements on the property before the whole process for FR-26 is initiated. * **FR-28:** Following from FR-26-part b, 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. * **FR-32:** 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. * **FR-33:** The Excel sheets provided by Watson games that contain information/data relating to the game board and certain spaces and cards (e.g. 'opportunity knocks' and 'pot luck' cards) are just a convenient format to provide you with the data. You can process, store and manage the data anyway you see fit. We have no view on the best way to do this.   **User Interface Requirements:**   * **UI-05:** Cards should be displayed when they are withdrawn from the pack. From this, if it is an instruction, then they should be clearly visible for the user to read. A close/exit button for the card should also be clearly visible. The cards that display property information should disclose the following information:   + Property Name   + Property image (would be ideal)   + Property price   + Property description   + Colour category (name of category in the colour of its name)   This type of card should also have a close/exit button on the card and must be visible for the user to see and click. |

|  |
| --- |
| 1. **Design** |
|  |

|  |
| --- |
| 1. **Test Plan & Evidence of Testing** |
| **System Level Testing:**   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Test No.** | **Description** | **Input(s)** | **Expected Outcome** | **Actual Outcome** | **Success?** | | 1 | Bank is supplied the correct amount of cash. | Successful game setup. | Value of cash owned by the bank is £50,000. | The bank has infinite money. | Changed from | | 2 | Only the Banker has access to the bank – no players. | Successful game setup. | No player can make ‘direct contact’ with the bank. | The players cannot access the bank and there is no trading. | Partial - the bank is not a player. | | 3 | Player lands on a property already owned by another player. | Player lands on a property owned by another player [board tile]. | The rent value of the property is deducted from the player’s account and added to the property owner’s account. | The rent is automatically deducted from the player and sent to the owners account. |  | | 4 | Player makes ‘improvements’ to their property; doesn’t have enough cash to do so. | Player clicks ‘buy house/hotel’ button [button click].  Insufficient cash amount [integer]. | Error message is displayed; house/hotel is not added to the property. | An error message is shown saying that the player doesn’t have the cash. |  | | 5 | Player makes ‘improvements’ to their property; has enough cash to afford to. | Player clicks ‘buy house/hotel’ button [button click]. | Purchased house/hotel is added to the property; rent value of property is increased as specified.  Cost of transaction is deducted from player’s account and added to banker’s account. | A window is shown and which says the number of houses and the new rent. The money for the house is also subtracted |  | | 6 | Player lands on a property where the completed property set is owned by another player. | Player lands on a property owned by another player [board tile]. | The player pays the property owner double the value of the rent. | The player has rent taken and sent to the owner. | Partial - need to implement a method to check if the owner has the set. | | 7 | Player lands on a property where the completed property set is owned by another player; said property owner has made ‘improvements’ to this set. | Player lands on a property owned by another player [board tile]. | The player pays the property owner the calculated ‘improved’ rent. | The player pays the owner the rent with the new improvement cost |  | | 8 | Player does not have enough cash to pay a property owner upon landing on their property. | Player lands on a property owned by another player [board tile].  Insufficient cash amount [integer]. | The player’s properties in their inventory are sold to the rent receiver. | N/A - No implementation as yet. |  | | 9 | Player does not have enough cash to pay a property owner upon landing on their property; doesn’t have enough asset value to cover the rent. | Player lands on a property owned by another player [board tile].  Insufficient cash amount [integer].  Insufficient asset value in inventory [integer]. | As much cash is added to the rent receiver’s account; paying player is declared bankrupt and ejected from the game; bankrupt player’s game token is removed from board. | N/A - No implementation as yet. |  | | 10 | Player sells a property back to the bank. | Player clicks ‘sell property’ button [button click]. | Property card is transferred from player’s account to the bank; value of the property is added to the player’s account and deducted from the bank. | N/A - No implementation as yet. |  | | 11 | Player mortgages a property. | Player clicks ‘mortgage property’ button [button click]. | Bank pays the player half the value of the property; rent value for said property is reduced to £0. | The player does receive half the property value. And all houses/hotels are removed and rent is set to 0; |  | | 12 | Player attempts to sell or mortgage a property; currently has ‘improvements’ assigned to said property. | Player clicks ‘mortgage property’ or ‘sell property’ button [button click].  ‘improvements’ on property in question [Boolean]. | Error message displayed; the property is not sold or mortgaged. | N/A - No implementation as yet. |  | | 13 | Player buys a house/hotel for a property in their owned set; difference between houses on said property is no greater than the houses on other properties in the same set. | Player clicks ‘buy house/hotel’ button [button click].  Difference between houses in the set is -1 or 0 [integer]. | House/hotel is successfully bought and added to the property; value of house is transferred from player’s account to the bank. | N/A - No implementation as yet. |  | | 14 | Player buys a house/hotel for a property in their owned set; difference between houses on said property is 1 greater than the houses on other properties in the same set. | Player clicks ‘buy house/hotel’ button [button click].  Difference between houses in the set is 1 [integer]. | Error message is displayed; transaction is not completed. | N/A - No implementation as yet. |  | | 15 | A player’s mortgaged property is sold back to the bank. | Player clicks ‘sell property’ button [button click]. | Half of the property value is transferred from the bank to the selling player’s account. | A message shows saying that there is now a mortgage on the property. |  | | 16 | User withdraws a card from stack. | User lands on a card stack tile [board tile]. | Card at top of stack is withdrawn and displayed to all players. | The card is displayed to the user |  | | 17 | User withdraws a card from stack. | User lands on a card stack tile [board tile]. | A duplicate of withdrawn card is placed at bottom of stack. | The card returns to the pile | Partial - random identifier used to select what card is picked. | | 18 | A player picks up a fine. | Player lands on a tile applying a fine [board tile] or picks up a card applying a fine [card draw]. | Value of the fine is deducted from the player’s account and added to the ‘free parking’ total. | The value is taken from the player and added to the free parking |  | | 19 | A player lands on the ‘free parking’ tile. | Player lands on ‘free parking’ tile [board tile]. | Current total value of ‘free parking’ is transferred to the player in question’s account. | The value of free parking is given to the player |  | | 20 | Player is sent to jail; pays £50 fine. | Player picks up a ‘go to jail’ card [card draw].  Pays £50 fine [integer] to ‘free parking’. | Player’s game token is moved to the ‘jail’ tile, but placed on ‘just visiting’; player cannot make any other actions that turn. | The player is moved to the just visiting position. | Partial - the player can manage properties. | | 21 | Player is sent to jail; doesn’t pay £50 fine. | Player picks up a ‘go to jail’ card [card draw]. | Player’s game token is moved to the ‘jail’ tile and can make no action (including receiving rent) for the next 2 rounds. | The players in jail Boolean is set to true. And the player can only roll to get out of jail | Partial - the user can roll a double to get out of jail on the next turn. | | 22 | Player uses ‘get out of jail free’ card. | Player uses ‘get out of jail free card’ [card draw]. | The card is taken from the player’s inventory and placed at the bottom of the appropriate card pile; game token is moved to ‘just visiting’. | The card is taken from the user and the user is moved to just visiting |  |   - |

|  |
| --- |
| 1. **Summary of Sprint** |
|  |