Sprint 8

1) Summary	
Sprint leader(s)	Vivien
Sprint start date	23/04/2020
Sprint end date	28/04/2020

2) Individual key contributions	
Team member	**Key Contribution(s)**
Neumann, Vivien	Design Documentation Finish Process Documentation (Sprint Documents, Meeting Documents, etc.) Adjust project plan and documenting the task distributions
Jiao, Haotian (Hallton)	Game Agent + Documentation
Wang, Mingfeng (Foret)	Game Agent + Documentation
Banes, Hayden J	Update GUI - Property management - Fix issue with ClassCastException - Fix bug in players becoming stuck - Fix NullPointerException when accessing property info - (More to come during sprint) - Simulation monitoring Documentation
Tang, Zhenyu (tang)	Continue working out the previous sprints tasks

3) User stories / task card
Task Card: Game Agent **Priority: 1** **Value: 9**

Sprint 8

In order to play Property Tycoon with less than two human players, a game agent needs to be implemented. An agent that can take the role of 1 or more of the players. This allows for a limited number of human players to enjoy a richer gaming experience but also provides the possibility for a fully autonomous play when all of the players are provided by the program.

For the beginning, the game agent should perform random decision making but could incorporate some simple rules for making in game decisions in a later stage. However, the emphasis is on a game agent that works well and can be used to replace one or more human players.

The game agent should throw the dice, move its token and act accordingly. They should also buy properties and develop these by building houses and hotels as well as participate in auctions. In the following, we defined the features of a game agent:

- Buy property (already implemented in Sprint7)
- Buy house and hotels (already implemented in Sprint7)
- Roll dice (already implemented in Sprint7)
- Free Parking (already implemented in Sprint7)
- Pay rent
- Auctioning
- Pot luck / opportunity cards

A game player agent may not opt to retire from the game. A game player only leaves the game when they are bankrupt.

Task Card 2: Abridged game (→ might be finished already)
Priority: 2
Value: 9

- Need to implement end game:
    - Value of assets
    - End of time

When the user chooses the abridged game mode, they also define a time limit. This has to be integrated in the GUI as well as the players should be able to insert the amount of time they want to play. During the game, players should know how much time is left and therefore, a timer that counts down should be visible at any time.

When the time limit is reached, the system calculates the value of the game assets of each player. The calculation is done by adding up to following assets of a player:

- Cash held
- Value of properties shown on the game card unless the property is mortgaged, in which case the value is half the value shown on the game card
- Value of houses and hotels purchased for each property
- Pot luck and opportunity cards do not have cash value

After calculating the value of the game assets of each player, the players are ranked and the one with the greatest value is declared as the winner. Part of this ranking are not only human players, but also the game agents. The ranking including the value of game assets should be shown to all players and the winner is notified.

Sprint 8

Task Card 3: Improve Code Documentation
Priority: 3
Value: 8

- Completeness of documents covering key sections, describing key aspects of core components, etc.

Task Card 4: Improve GUI
Priority: 1
Value: 10

Property Management

- Groups property cards by colour
- Select property to sell + buy/upgrade houses
- Start a trade using a property?
- Free Parking (how much money is in the centre of the board)
- Mortgage

Task Card 5: Open issues / adjustments?

- Player leaves the game feature
- Developing properties feature: “there may never be a difference of more than 1 house between the properties in that set.”
- Monitoring performance of simulation: worth of each of the players and the property assets that they own should be available for all to see at every time.

4) Requirement analysis

	Functional	**Non-Functional**
TC1: Game Agent	TC1-F1: The game agent shall takes the role of one or more players TC1-F2: The game agent shall act like a human player: - throws the dice - moves its token and acts according to the space they landed on (take a pot luck or opportunity card, goes to jail, buys property, takes place in	- TC1-NF1: The application shall be capable of replacing all six players of the game and play without a human player - TC1-NF2: Human players shall not be able to manipulate the game agent - TC1-NF3: The game agent should be implemented in a

Sprint 8

	property auction, etc.) TC1-F3: The game agent shall stick to the rules(e.g. Go to jail)	way that its performance can be improved throughout the development process
TC2: Abridged Game	TC2-F1: Player shall be able to choose the length of the Property Tycoon game (integrated in GUI) TC2-F2: A timer should be available to show how much game time is left TC2-F3: When the time limit is reached, the value of game assets owned by each player shall be added up to calculate an end result TC2-F4: When calculating the game assets, it shall be ensured that only values owned by the player are added. TC2-F5: A leaderboard with the end results should be shown to all players TC2-F6: The winner should be notified	
GUI	Player can use the GUI to: - View owned properties - Buy houses for property - Sell property - View game statistics	

5) Design

Design of game agent feature is shown in Sprint 7.

6) Test plan and evidence of testing

***TC1-F1 &&* TC1-F2*:* pay rent**

System test1:

Create a agent called “agent1” with token boot

Sprint 8

Add the agent1 to playerlist
Create a player name player2 with token smartphone
Add the agent2 to playerlist
Give agent2 a property1
Let agent1 pay rent of the property1 to agent2`s property
Ensure minus agent1`s money by rent of the property1.
Add agent2`s money by rent of the property1.

Build house

System test2:

Give agent1 a property1
Have enough money
AutobuildHouse()
Ensure the number of houses in property1 is 1.
Ensure minus agent1`s money by cost of the house.

Build hotel

System test3::

Give agent1 a property1
Have enough money
Build 4 house
AutoBuildHotel
Ensure the number of hotels in property1 is 1.
Ensure minus agent1`s money by cost of the hotel.

Buy property

System test4:

Make agent1 land on location 2
AutoBuyProperty

Ensure the number of property1 in property1 is 1.
Ensure minus agent1`s money by cost of the property1.

7) Summary of sprint

All GUI tasks are done:

- Property Management
- Simulation Monitoring
- Documentation for both of them is done
- Fixing bugs and issues
- Still missing: Auctioning and Trading
- For now: we can use our time more wisely in the upcoming sprint rather than upgrade the GUI. It’s more important that all features are working.

Sprint 8

Missing game agent features are implemented (pay rent, auctioning and opportunity cards)

- Proper testing is still missing

Open Issues:

- Developing properties feature: “there may never be a difference of more than 1 house between the properties in that set.” → finished in code

-----

Open issues and bug we have found and should be prioritised in the next sprint:

- Check if players select the same game piece
- Start game with less than two players
- Get the price of upgrading the property (increase number of houses)
- Duplicate methods throughout (See javadoc)
- Current method for reading csv only works in netbeans, getResource needs to be used
- Landing on jail does not send player to jail
- Passing go crashes the game (And might not award player 200)
- Dice method in player class does not work
- Developing properties feature: “there may never be a difference of more than 1 house between the properties in that set.”
- One player can still start the game
- Blank name and same name players can be created
- Repeated token can be used