Workshop 2

Pairs Red-14
Shun Nishijima(2977923)
Mateusz Bartnicki(3006891)

Task 1

Elicit a set of SMART requirements for the P-project:
• Make sure you cover every type of requirements (8 types, as in slide 14) and every possible facet (4 facets, as in slide 17).
• For each requirement, explicitly state its type and the facet it corresponds to.
• You should use the P-project description in Workshop 1 to devise some of the requirements.
• Interact with other groups of students to extract at least two requirements from stakeholders outside your development team! Indicate in the assignment sheet which are the requirements you identified at this step.
• Write down the elicited requirements using Rupp's format. Make your requirements traceable by numbering them.

If the chef has rejected the guest's order, the system should ask the guest whether the guest would like to choose another dish

Functional Requirements

1. Completeness / Usage facet

If a user wants to start a standard game, the system shall allow play on both client and server in conjunction with the reference server and client, respectively.

2. Correctness / Subject facet

Whenever a client disconnects during a game, the server of the system shall inform the other client(s) and end the game, allowing the other player to start a new game.

3. Appropriateness / Usage facet

If the user wants to change the level of AI, the system should ask the user to adjust the AI difficulty via the TUI.

Non-Functional Requirements

4. Performance / IT system facet

In any case, the game should be able to run at minimum 30 frames per second on the user's system.

5. Usability / Usage facet

At the start of the game, the game shall provide the ability for the user to choose who decides on the moves, AI or human via the TUI.

6. Reliability / IT system facet

When playing a game using your client on a server not adhering to the protocol (or vice versa), your product should not crash under any circumstance.

7. Compatibility / IT system facet

When a game is finished, the system shall allow the users be able to play a next game immediately after: client and server should not need to re-establish a connection.

8. Security / Usage facet

If someone wants to access the server, the server of the system shall only be accessible to authorized admins via an encrypted communication network.

9. Maintainability / Development facet

If someone wants to write a new user interface without modifying the rest of the program, the system should be made use of Listener patterns as well as the Model-View-Controller pattern.

10. Portability / Development facet

If the user uses a Macbook, the system may be able to be downloaded as a Mac OS version.

Requirements From the other group (yellow 9)

11. If more players than it is allowed try to enter, the system will move the players into a waiting queue. – Performance/USAGE FACET

12. If a user tries to exploit bugs, the system should have a detection system for exploits and should it detect the player he will be banned. – Reliability/USAGE FACET

Task 2

Locate in the stakeholders' satisfaction coordinate system the requirements in Task 1 (use the associated requirements numbers), and briefly motivate your choice.

Requirements prioritization

1. Satisfier

This function is necessary in the whole system and customers really expect it to be developed.

2. Dissatisfier

The requirement is something that might have gotten overlooked in a different scenario, but once raised it does seem obvious. Also while the user won't fawn over it, their satisfaction could go down very fast if they were stuck in a game with a disconnected player.

3. Delighter

The requirement is an additional part of the system, the user can play if the requirement doesn't exist. But this requirement contributes to user experience. The system becomes more attractive if the user can change the level.

4. Satisfier

The requirement is a crucial part of the system, one of the main things users look at any software is the performance and their satisfaction grows/falls proportionally to how smooth the experience is.

5. Satisfier

The requirement is essential in the system. If it doesn't exist, the user can't select which AI or humans. The user expects that it must be developed.

6. Dissatisfier

The requirement is more of a fringe situation that will not happen a lot and that the user does not think about until it happens. But when it does it can cause a lot of frustration which would lead to satisfaction going down.

7. Dissatisfier

The requirement seems to be necessary, but the user doesn't expect it to exist. If the requirement was not developed, the user must have complaints, because they need to construct a connection for every new game.

8. Satisfier

The more effort and thought is put into the security of the system, the harder it will be to breach and ruin the experience for the user.

9. Delighter

This requirement is for developers and it's an additional requirement. The user will be happy when they want to make a new Interface, because they don’t need to make it perfectly new.

10. Delighter

It is not a must since most user systems are windows but it could make a Mac user very happy when they find out they can also use the product.

11. Delighter

The requirement can make users more comfortable, because the requirement allows them to recognize how many queues for starting a new game. It helps user satisfaction.

12. Dissatisfier

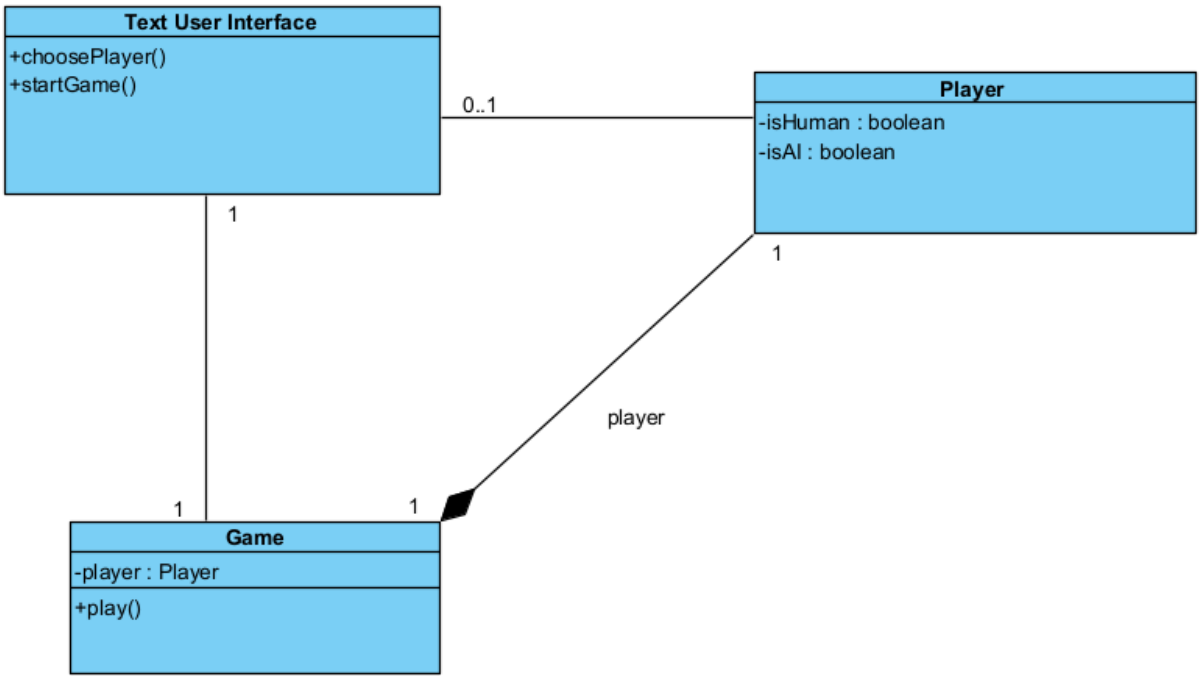
Not something very obvious when designing the system but facing an opponent that has an unfair advantage through exploits can severely detriment user satisfaction.

Task 3

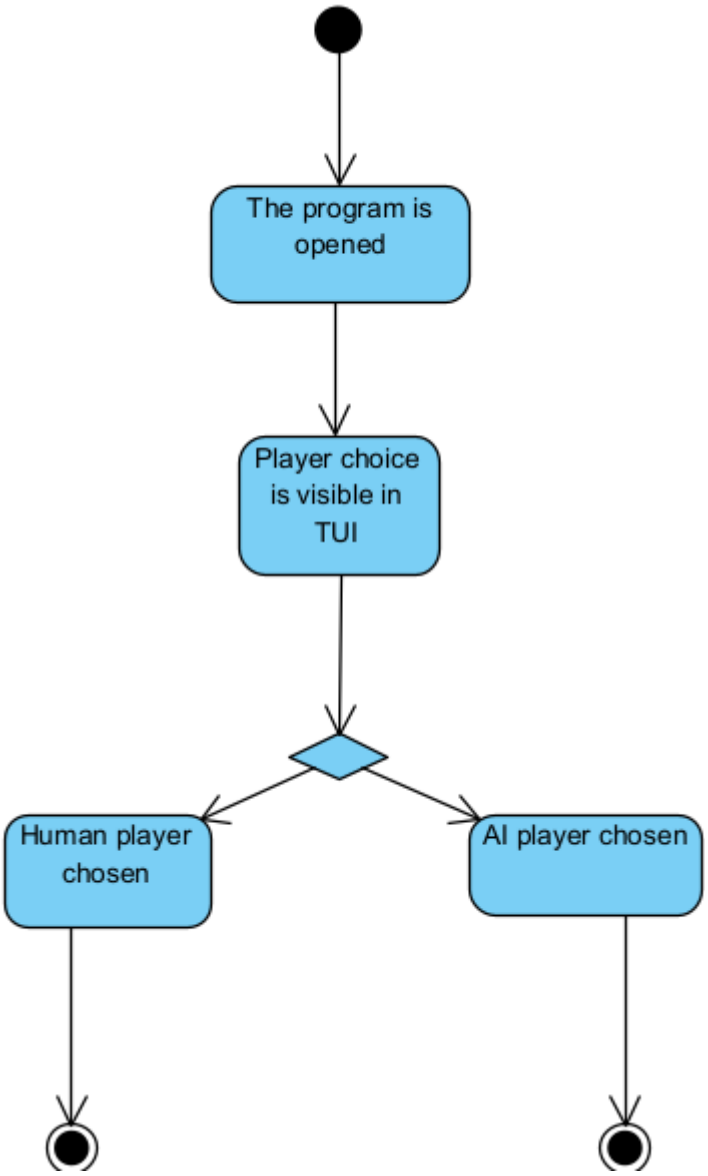
Document one of the requirements in Task 1 from the functional(activity diagram), behavioral(state machine diagram) and data(class diagram) perspective. If documenting is not possible for some of the perspectives, then explain why

5. At the start of the game, the game shall provide the ability for the user to choose who decides on the moves, AI or human via the TUI.(Usability / Usage facet)

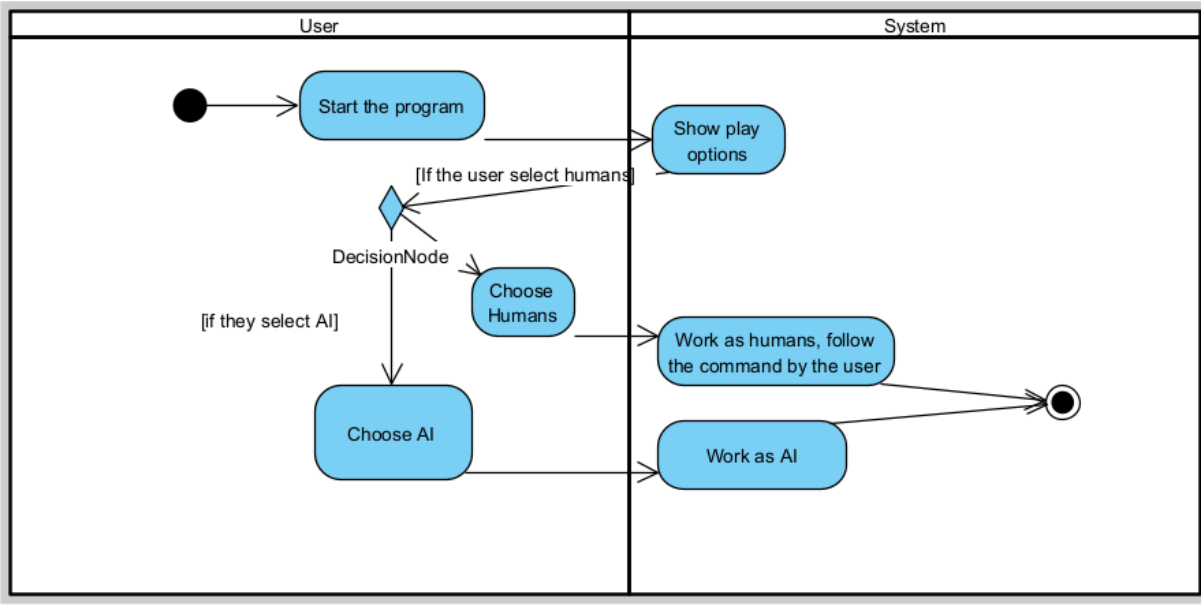
a) Data Perspective



b) Behavioral Perspective



c) Functional Perspective


User
System
Start the program
Show play options
[If the user select humans]
DecisionNode
Choose Humans
[if they select AI]
Work as humans, follow the command by the user
Choose AI
Work as AI