syst 17796 Deliverable 2

design document template

Group 1 – The Bronze Medalists

# Overview

## Project Background and Description

As discussed in Deliverable 1, the aim of the project is to develop an offline virtual simulation of the card game Coup, by Indie Board Games. As per the rules of the game and the design decisions determined for this project, our application will perform the following.

At initiation, the game will provide a brief overview of the rules, after which it will register a user-specified number of players between 3 and 6. The game will then request a username from each player and register them for the game. Once all players are registered, a round will begin. If no prior rounds have been played, a random player will be selected as the starting player, otherwise the winner of the previous round will start. Turn order will proceed from the starting player in the order that players registered (i.e. the player who registered after the starting player will go next and so on). Each player is given two cards (with their associated effects) in their hand corresponding to two influence points, and two coins.

At the start of each turn, the active player’s name is shown in addition to a small guide, each player’s public information (coins and influence) and a list of discarded cards. The active player is then prompted to select an action to declare. If the effect is valid and the player has enough coins to perform the action, the action is now declared (otherwise the player is prompted to re-enter input). If an action targets another player, the active player must also enter the name of the player to target. At this stage, if the action is refutable, each player is asked if they would like to challenge the action. If any player challenges the action, the active player reveals if they were bluffing, and if so, is required to lose one influence point and discard a card from their hand. Additionally, the declared effect is cancelled (including any expenditure of coins) and the player’s turn immediately ends. If the active player was not bluffing, the challenging player loses one influence point and a card from their hand, and the declared effect continues.

If no player challenges, or a challenge is unsuccessful, an unblockable action is executed and its effect is processed. If the action is blockable, the target player is asked if they would like to attempt a block (in the case where the active player declares Foreign Aid, any player may choose to block). Should they choose to do so – and since, like actions, blocking actions have specific character associations – the active player may issue a challenge which will proceed in a similar fashion to the process described above. If the challenge is not successful, in addition to losing influence, the active player’s declared effect is blocked (requiring them to pay any coins needed for the action) and the turn ends immediately.

If the target player does not block, or a block is successfully challenged, the declared action is executed, and its effect is processed. At the conclusion of each turn, any player whose influence fell to 0 (or below) is removed from set of living players and will no longer receive a turn for the remainder of the round. The next living player is then set to the active player and proceeds with their turn.

If at the end of a turn, there is only one player remaining in the set of living players, that player is declared the winner of the round and their score is incremented by one. If the user decides to play another round, all registered players are returned to the living players set and the round initiation process is executed. If the user decides not to continue at the end of a round, a list of the players ranked by their score is shown and the program exits. A full summary of the features can be found in the Use Case diagram shown in Figure 1.

## Design Considerations

Describe the Class Diagram you delivered above (it should be descried as Figure 1 or Figure x if you have more than one Figure), explaining the associations and multiplicities depicted.

Comment on each of the following as it pertains to the class groupings you have decided upon and if you have included methods, modifiers and return types, comment on those here as well. You may wish to describe any data structures you wish to use (i.e an enumeration) when you are explaining your design choices. Be specific for full credit.

* Encapsulation
* Delegation
* Cohesion
* Coupling
* Inheritance
* Aggregation
* Composition
* Flexibility/Maintainability