**Game Initialization**

Use Case: Players Select Illuminati

Actors: Player, Deck

Pre-Condition: Player pressed “Play Game” button

Post-Condition: Each player has a Illuminati card in play that is unique to each player and has an Illuminati-specific win condition

Events:

1. Deck shuffles the eight Illuminati cards
2. Repeat for each player: Player draws an Illuminati card from the deck
3. Deck removes top card
4. Player receives the top card

Extensions:

2b: Player receives “The UFOs”

2b.1: GamePlay prompts player to select a win condition

2b.2: Player selects win condition, GamePlay records this win condition

2b.3: End Use Case

Test Case:

Success: Post-Condition fulfilled: Each player has a Illuminati card in play that is unique to each player and has an Illuminati-specific win condition

Failure: Deck did not remove the top card from the deck, so more than one player has the same Illuminati

Use Case: Game Initialization - Draw Four Uncontrolled Groups

Actors: Deck

Pre-Condition: Players have picked their Illuminati card and victory condition

Post-Condition: There are four uncontrolled groups in play

Events

1. Deck shuffles group and special cards
2. Deck draws a card
3. Card drawn is a group card
4. Add card to uncontrolled groups
5. Repeat steps 2 through 4 until there are four uncontrolled groups

Extensions:

3b. Card drawn is a special card

3b.1. Return card to a random spot in the deck.

3b.2. Rejoin step 2

Test Case:

Success: Post-condition met: There are four uncontrolled groups in play

Failure:

1. Deck draws a special card, and fails to return it to the deck (it vanishes)
2. When deck draws a group and it’s placed into the uncontrolled groups, the card is not removed from the deck
3. There are fewer or more than four uncontrolled groups in play
4. A special card is in the uncontrolled groups section
5. Deck draws a group card, but fails to put it in the uncontrolled groups section

**Attacking**

**Use Case**: Selecting the type of attack (control, neutralize, or destroy) and the attacking group

Actors: Player, GroupCard or IlluminatiCard

Pre-Condition: Player decides to Attack and has at least one Action remaining

Post-Condition: The type of attack and the attacking group have been selected. These selections have been validated.

Events:

1. Attacking player is prompted for the attack type
2. Attacking player selects the attack type
3. Attacking player is prompted to select the attacking GroupCard
4. Attacking player selects the attacking GroupCard

Extensions:

4a: Attacking player selects a group with no power

4a.1: Prompt that the attacking group is invalid because it has no power

4a.2: Rejoin step 3

4b: Attack type is control, attacking GroupCard has no open control arrows

4b.1: Prompt that the attacking group is invalid because it has no open control arrows

4b.2: Rejoin step 3

**Test Case**:

Success: Post-Condition fulfilled: The type of attack and the attacking group have been selected. These selections have been validated.

Failure:

1. The type of attack and attacking group has not been established
2. Validation fails
   1. Attacking group has no power
   2. In an attack to control, attacking group has no open control arrows

**Use Case**: Select the group to attack (defending group) for an attack to control

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: The type of attack is control, and attacking group has been established

Post-Condition: The defending group has been selected. This selection has been validated.

Events:

1. Attacking player is prompted for the group to attack
2. Attacking player selects the group to attack
3. For each identical alignment, +4 to attack modifier
4. For each opposite alignment, -4 to attack modifier
5. Add (attacker power - defender resistance) to attack modifier

Extensions:

2a: Attacking player selects the attacking group as the defending group

2a.1: Prompt that the group is invalid because a group cannot attack itself

2a.2: Rejoin step 1

2b: Attacking player selects a group that the attacking player already controls

2b.1: Prompt that the defending group is invalid because the player already controls it

2b.2: Rejoin step 1

2c: Defending group is owned by a player, defending player is *The Discordian Society* and the attacking group is Government or Straight

2c.1: Prompt that the group is invalid because *The Discordian Society* is immune to attacks from the attacking group

2c.2: Rejoin step 1

5a: Defending group is *Gun Lobby*

5a.1: If the attacking group is Communist, Liberal, or Weird, add (attacker power - 10) to attack modifier. Otherwise, add (attacker power - 3) to attack modifier.

5a.2: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: The defending group has been selected. This selection has been validated.

Failure:

1. The defending group has not been established
2. Validation fails
3. Attacking group is the same as the defending group
4. The defending group is a group the attacking player controls
5. *Gun Lobby* doesn’t have the correct resistance

**Use Case**: Select the group to attack (defending group) for an attack to neutralize

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: The type of attack is neutralize, and attacking group has been established

Post-Condition: The defending group has been selected. This selection has been validated.

Events:

1. Attacking player is prompted for the group to attack
2. Attacking player selects the group to attack
3. Add +6 to attack modifier
4. For each identical alignment, +4 to attack modifier
5. For each opposite alignment, -4 to attack modifier
6. Add (attacker power - defender resistance) to attack modifier

Extensions:

2a: Attacking player selects the attacking group as the defending group

2a.1: Prompt that the group is invalid because a group cannot attack itself

2a.2: Rejoin step 1

2b: Defending group is owned by a player, defending player is *The Discordian Society* and the attacking group is Government or Straight

2b.1: Prompt that the group is invalid because *The Discordian Society* is immune to attacks from the attacking group

2b.2: Rejoin step 1

3a: If attacking player is *The Society of Assassins*

3a.1: +10 to attack modifier

3a.2: Rejoin step 4

5a: Defending group is *Gun Lobby*

5a.1: If the attacking group is Communist, Liberal, or Weird, add (attacker power - 10) to attack modifier. Otherwise, add (attacker power - 3) to attack modifier.

5a.2: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: The defending group has been selected. This selection has been validated.

Failure:

1. The defending group has not been established
2. Validation fails: Attacking group is the same as the defending group
3. *Gun Lobby* doesn’t have the correct resistance

**Use Case**: Select the group to attack (defending group) for an attack to destroy

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: The type of attack is destroy, and attacking group has been established

Post-Condition: The defending group has been selected. This selection has been validated.

Events:

1. Attacking player is prompted for the group to attack
2. Attacking player selects the group to attack
3. For each opposite alignment, +4 to attack modifier
4. For each identical alignment, -4 to attack modifier
5. Add (attacker power - defender power) to attack modifier

Extensions:

2a: Attacking player selects the attacking group as the defending group

2a.1: Prompt that the group is invalid because a group cannot attack itself

2a.2: Rejoin step 1

2b: Attacking player selects a group to attack that has 0 power

2b.1: Prompt that the group is invalid because a group with 0 power cannot be destroyed

2b.2: Rejoin step 1

5a: If attacking player is *The Servants of Cthulu*

5a.1: Add ((attacker power + 2) - defender power) to attack modifier

5a.2: End Use Case

2c - 5c: Attacking player played *Whispering Campaign*, attacking player selects a group to attack that has 0 power

2c.1: For each opposite alignment, +4 to attack modifier

2c.2: For each identical alignment, -4 to attack modifier

2c.3: Add (attacker power - defender resistance) to attack modifier

2c.4: End Use Case

2c.3a: If attacking player is *The Servants of Cthulu*

2c.3a.1: Add ((attacker power + 2) - defender power) to attack modifier

2c.3a.2: End Use Case

2d: Defending group is owned by a player, defending player is *The Discordian Society* and the attacking group is Government or Straight

2d.1: Prompt that the group is invalid because *The Discordian Society* is immune to attacks from the attacking group

2d.2: Rejoin step 1

**Test Case**:

Success: Post-Condition fulfilled: The defending group has been selected. This selection has been validated.

Failure:

1. The defending group has not been established
2. Validation fails
   1. Attacking group is the same as the defending group
   2. Defending group has zero power and *Whispering Campaign* has not been played
3. Attack modifier calculations incorrect (e.g. calculation uses defending resistance on a standard attack to destroy, or on an attack with *Whispering Campaign* the calculation attempts to use the defending group’s non-existent power)

**Use Case**: Calculate attack based on power structure position

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: The defending group has been established, and it is owned by a player

Post-Condition: The attack modifier is adjusted based on the distance of the group from the Illuminati

Events:

1. Calculate the distance between the defending group and its Illuminati
2. If the defending group is one card away from the Illuminati, -10 to attack modifier
3. If the defending group is two cards away from the Illuminati, -5 to attack modifier
4. If the defending group is three cards away from the Illuminati, -2 to attack modifier
5. If the defending group is four or more cards away from the Illuminati, do nothing

**Test Case**:

Success: Post-Condition fulfilled: The attack modifier is adjusted based on the distance of the group from the Illuminati

Failure:

1. The attack modifier is not changed properly
   1. The attack modifier is changed if defending group four or more cards away from the Illuminati
   2. The attack modifier is not changed, or is changed by the wrong number, when a card is fewer than four cards away from the Illuminati

**Use Case**: Calculate “any attempt” abilities

Actors: Player, GroupCard

Pre-Condition: The defending group has been selected and validated

Post-Condition: All “any attempt” abilities on the group cards owned by the attacking player have been added to the attack modifier.

Events:

1. For each group owned by the attacking player, repeat steps 2-5.
2. Check if the group card has an “any attempt” ability
3. If it does, check if the “any attempt” ability applies to the selected attack type
4. If true, check if the “any attempt” ability applies to the defending group (by name or by alignment)
5. If true, add the numerical value of the ability to the attack modifier

**Test Case**:

Success: Post-Condition fulfilled: All “any attempt” abilities on the group cards owned by the attacking player have been added to the attack modifier.

Failure:

1. A group’s “any attempt” ability has not been included in the attack modifier when the attack type and defending group are listed in the ability.
2. A group’s “any attempt” ability has been included in the attack modifier when the attack type and defending group are not listed in the ability.
3. A group’s “any attempt” ability has been included more than once.

**Use Case**: Calculate “direct control” abilities

Actors: Player, GroupCard

Pre-Condition: The defending group has been selected and validated

Post-Condition: The “direct control” ability of the attacking group has been calculated into the attack modifier

Events:

1. Attacking group has “direct control” ability
2. Check if the attack type is control
3. If true, check if the ability applies to defending group (by name or by alignment)
4. If true, add the numerical value of the ability to the attack modifier

Extensions:

1a: Attacking group does not have “direct control” ability

1a.1: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: The “direct control” ability of the attacking group has been calculated into the attack modifier

Failure:

1. Attacking group’s “direct control” ability has not been included in the attack modifier when the attack type and defending group are listed in the ability.
2. Attacking group’s “direct control” ability has been included in the attack modifier when the attack type and defending group are not listed in the ability.
3. Attacking group’s “direct control” ability has been included more than once.

**Use Case**: Calculate *Chinese Campaign Donors* Ability

Actors: Player, GroupCard

Pre-Condition: The defending group has been selected and validated, attacking card is *Chinese Campaign Donors*

Post-Condition: The attack modifier is adjusted if *Chinese Campaign Donors* is attacking a card with the Government alignment

Events:

1. Check if the attack type is control
2. If true, check if the defending card has the government alignment
3. If true, +4 to attack modifier (as *Chinese Campaign Donors* is treated as a Government group

Extensions:

3a: Defending player is *The Discordian Society*

3a.1: Prompt that the group is invalid because *The Discordian Society* is immune to attacks from the attacking group

3a.2: Return to Use Case *“Select the group to attack (defending group) for an attack to X”*, where X is the attack type.

**Test Case**:

Success: Post-Condition fulfilled: The attack modifier is adjusted if *Chinese Campaign Donors* is attacking a card with the Government alignment

Failure:

1. Attack modifier is adjusted when the defending group is not Government.
2. The attack modifier is adjusted by an amount not equal to 4.
3. Player is not prompted to select a new group to attack if the defending player is *The Discordian Society*.

**Use Case**: Calculate *Survivalists* Ability

Actors: Player, GroupCard

Pre-Condition: The defending group has been selected and validated

Post-Condition: The attack modifier is adjusted if *Survivalists* is controlled by the defending player

Events:

1. Repeat steps 2-3 for each group controlled by defending player
2. Check if group is *Survivalists*.
3. If true, -2 to attack modifier, end use case.

**Test Case**:

Success: Post-Condition fulfilled: The attack modifier is adjusted if *Survivalists* is controlled by the defending player

Failure:

1. Attack modifier is adjusted when the defending player does not own *Survivalists*.
2. The attack modifier is adjusted by an amount not equal to 2.

**Use Case**: Add groups to aid attack

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: The defending group has been selected and validated

Post-Condition: The attacking player selects the groups that will aid in the attack. The aiding groups have been validated (have not attacked this turn, have transferable power)

Events:

1. Optional. Repeat. Attacking player selects group to aid attack
2. Add group to attackers, add transferable power to attack modifier.
3. Optional. Repeat. Attacking player removes group to aid attack
4. Remove group from attackers, remove transferable power from attack modifier.

Extensions:

1a: Aiding group has already attacked this turn (attack counter is 0)

1a.1: Prompt that the group is invalid because the group has already attacked

1a.2: Rejoin step 1.

1b: Aiding group has no transferable power

1b.1: Prompt that the group is invalid because the group has no transferable power

1b.2: Rejoin step 1.

**Test Case**:

Success: Post-Condition fulfilled: The attacking player selects the groups that will aid in the attack. The aiding groups have been validated (have not attacked this turn, have transferable power)

Failure:

1. A group that has already attacked this turn is aiding in the attack.
2. A group that has no transferable power is aiding in the attack.
3. An aiding group’s transferable power is not added to the attack modifier.
4. The aiding group’s power is added to the attack modifier instead of the transferable power.

**Use Case**: The Bavarian Illuminati special ability (privileged attack)

Actors: Player, IlluminatiCard

Pre-Condition: The attack has been declared, attacking player is *The Bavarian Illuminati*, Illuminati treasury has at least 5 MB, the player has not used the special ability this turn, attack has never been privileged.

Post-Condition: The attacking player chooses if the special ability is used for this attack. The Illuminati treasury spends 5 MB. The attack is set to privileged.

Events:

1. Prompt attacking player if the special ability is to be used for this attack.
2. Player selects yes
3. Attack is set to privileged
4. 5 MB is removed from Illuminati treasury
5. 5 MB is placed in the bank

Extensions:

2a: Player selects no

2a.1: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: The attacking player chooses if the special ability is used for this attack. The Illuminati treasury spends 5 MB. The attack is set to privileged.

Failure:

1. User selects yes, and attack is not set to privileged
2. User selects yes, and 5 MB are not removed from the Illuminati treasury
3. User selects no, and attack is set to privileged
4. User selects yes, and 5 MB are removed from the Illuminati treasury

**Use Case**: Discard special for privilege

Actors: Player, IlluminatiCard, SpecialCard

Pre-Condition: The attack has been declared, attacking player has at least one special card, attack has never been privileged.

Post-Condition: The attacking player chooses if the attack is privileged. If it is, the selected special card is discarded and the attack is set to privileged.

Events:

1. Prompt attacking player if he or she wants to discard a special card to make an attack privileged
2. Player selects yes
3. Prompt attacking player which card to discard
4. Player selects a card
5. Place selected special card in the discard pile
6. Attack is set to privileged

Extensions:

2a: Player selects no

2a.1: End Use Case

4a: Player selects cancel

4a.1: Rejoin step 1

**Test Case**:

Success: Post-Condition fulfilled: The attacking player chooses if the attack is privileged. If it is, the selected special card is discarded and the attack is set to privileged.

Failure:

1. User selects yes, and attack is not set to privileged
2. User selects yes, and the special card is not discarded
3. User selects no, and attack is set to privileged

**Use Case**: Abolish privilege using *Deep Agent*

Actors: Player, SpecialCard

Pre-Condition: The attack is set to privileged, player holds *Deep Agent*

Post-Condition: If the player chooses to play *Deep Agent*, privilege is abolished, and *Deep Agent* is discarded

Events:

1. Prompt player holding *Deep Agent* if he or she wishes to play *Deep Agent*
2. Player selects yes
3. Discard *Deep Agent*
4. Attack is set to abolished privilege.

Extensions:

2a: Player selects no

2a.1: End Use Case

4a: Opposing player plays *Secrets Man Was Not Meant to Know*

4a.1: Discard *Secrets Man Was Not Meant to Know*

4a.2: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: If the player chooses to play *Deep Agent*, privilege is abolished, and *Deep Agent* is discarded

Failure:

1. User selects yes, and privilege is not abolished
2. User selects yes, and *Deep Agent* is not discarded
3. User selects no, and privilege is abolished
4. User selects no, and *Deep Agent* is discarded

**Use Case**: Abolish privilege by discarding two specials

Actors: Player, SpecialCard

Pre-Condition: The attack is set to privileged, player holds at least two special cards

Post-Condition: If the player chooses to discard two special cards, privilege is abolished, and the two specials are discarded

Events:

1. Optional. Repeat steps 1-2 until discardCounter is 2. Prompt player to select special to discard
2. Player selects special card that is not in list of cards to discard
3. Special card added to list of cards to discard
4. discardCounter increased by 1
5. Discard selected special cards
6. Attack is set to abolished privilege.

Extensions:

2a: Player selects cancel

2a.1: End Use Case

2b - 4b: Player selects special card that has already been selected

2b.1: Special card removed from list of cards to discard

2b.2: discardCounter decreased by 1.

5a: Opposing player plays *Secrets Man Was Not Meant to Know*

5a.1: Discard *Secrets Man Was Not Meant to Know*

5a.2: discardCounter decreased by 1.

5a.3: Rejoin step 1.

**Test Case**:

Success: Post-Condition fulfilled: If the player chooses to discard two special cards, privilege is abolished, and the two specials are discarded

Failure:

1. User selects yes, and privilege is not abolished
2. User selects yes, and two special cards are not discarded
3. User selects no, and privilege is abolished
4. User selects no, and two special cards are discarded

**Use Case**: Cancel an attack

Actors: Player

Pre-Condition: No money has been spent on the attack, the attacking player has not rolled the dice

Post-Condition: The attack has been canceled, player is allowed to choose another action

Events:

1. Attacking player selects the “Cancel Attack” button
2. Prompt all players that the attack has been canceled
3. Attacking player returns to action selection

**Test Case**:

Success: Post-Condition fulfilled: The attack has been canceled, player is allowed to choose another action

Failure:

1. The attack was not canceled
2. The attacking player is unable to take another action
3. The attack was canceled when the “Cancel attack” button was not pressed

**Use Case**: Attacker spends MB on attack

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: Attack is declared, privilege or lack thereof has been established

Post-Condition: The attack modifier is adjusted appropriately for the MB spent by the attacking group and the attacking player’s Illuminati (+1 for each MB spent)

Events:

1. Prompt attacking player to spend money on the attack
2. Optional. Attacking player selects amount of MB to spend from Illuminati treasury
3. Remove designated MB from Illuminati treasury
4. Place MB into bank
5. +1 to attack modifier for each MB spent
6. Optional. Attacking player selects amount of MB to spend from attacking group’s treasury
7. Remove designated MB from attacking group’s treasury
8. Place MB into bank
9. +1 to attack modifier for each MB spent

Extensions:

2a: Player selects no

2a.1: Rejoin step 6

6a: Player selects no

6a.1: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: The attack modifier is adjusted appropriately for the MB spent by the attacking group and the attacking player’s Illuminati (+1 for each MB spent)

Failure:

1. Player spends MB and the attack modifier isn’t changed.
2. The player decides to spend MB but the MB aren’t removed from the treasuries.
3. The player decides not to spend MB but MB are removed from the treasuries or the attack modifier is changed.

**Use Case**: Defender spends MB on attack

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: Attack is declared, privilege or lack thereof has been established, defending group is controlled by a player

Post-Condition: The attack modifier is adjusted appropriately for the MB spent by the defending group and the defending player’s Illuminati (-1 for each MB from Illuminati treasury, -2 for each MB from defending group’s treasury)

Events:

1. Prompt defending player to spend money on the attack
2. Optional. Defending player selects amount of MB to spend from Illuminati treasury
3. Remove designated MB from Illuminati treasury
4. Place MB into bank
5. -1 to attack modifier for each MB spent
6. Optional. Defending player selects amount of MB to spend from defending group’s treasury
7. Remove designated MB from defending group’s treasury
8. Place MB into bank
9. -2 to attack modifier for each MB spent

Extensions:

2a: Player selects no

2a.1: Rejoin step 6

6a: Player selects no

6a.1: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: The attack modifier is adjusted appropriately for the MB spent by the defending group and the defending player’s Illuminati (-1 for each MB from Illuminati treasury, -2 for each MB from defending group’s treasury)

Failure:

1. Player spends MB and the attack modifier isn’t changed.
2. The player decides to spend MB but the MB aren’t removed from the treasuries.
3. The player decides not to spend MB but MB are removed from the treasuries or the attack modifier is changed.

**Use Case**: Players Interfere with Attack

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: Attack is declared, privilege or lack thereof has been established

Post-Condition: The attack modifier is adjusted appropriately for the MB spent by the player’s Illuminati treasury (+1 per MB if assisting the attacker, -1 per MB if opposing the attacker)

Events:

1. Check if attack is privileged
2. Attack is not privileged
3. Prompt player to spend money interfering in the attack
4. Optional. Player selects amount of MB to spend from Illuminati treasury
5. Player selects to assist the attacker
6. Remove designated MB from Illuminati treasury
7. Place MB into bank
8. +1 to attack modifier for each MB spent
9. Repeat steps 3-8 for each player not involved in the attack

Extensions:

2a - 3a: Attack is privileged

2a.1: No player plays *Interference*

2a.2: End Use Case

2b - 3b: Attack is privileged

2b.1: One or more players plays *Interference*

2b.2: Repeat steps 3-8 for each player who played *Interference*

2b.3: End Use Case

4a: Player selects no

4a.1: Rejoin step 7

5a - 8a: Player selects to assist the attacker

5a.1: Remove designated MB from Illuminati treasury

5a.2: Place MB into bank

5a.3: -1 to attack modifier for each MB spent

5a.4: Rejoin step 9

**Test Case**:

Success: Post-Condition fulfilled: The attack modifier is adjusted appropriately for the MB spent by the player’s Illuminati treasury (+1 per MB if assisting the attacker, -1 per MB if opposing the attacker)

Failure:

1. Player spends MB and the attack modifier isn’t changed.
2. The player decides to spend MB but the MB aren’t removed from the treasuries.
3. The player decides not to spend MB but MB are removed from the treasuries or the attack modifier is changed.
4. Attack is privileged but a player interfered without playing *Interference*
5. Attack is privileged and a player could not interfere after playing *Interference*

**Use Case**: Spending phase of an attack

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: Attack is declared, privilege or lack thereof has been established, defending group is controlled by a player

Post-Condition: Each player who is allowed to spend MB on an attack has been given the opportunity to, and the attack modifier has been adjusted appropriately. The spending phase is complete when all players spent no MB in one loop.

Events:

1. Use case: *“Attacker spends MB on attack”*
2. Use case: *“Defender spends MB on attack”*
3. Use case: *“Players Interfere with Attack”*
4. Repeat steps 1-3

Extensions:

4a: No MB were spent in the current iteration of steps 1-3

4a.1: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: Each player who is allowed to spend MB on an attack has been given the opportunity to, and the attack modifier has been adjusted appropriately. The spending phase is complete when all players spent no MB in one loop.

Failure:

1. A player spends MB in the current iteration of steps 1-3, but is not allowed another chance to spend MB
2. No player spends MB in the current iteration of steps 1-3, but the use case returns to step 1.

**Use Case**: Dice roll to determine success of attack

Actors: Player, Die

Pre-Condition: Use case *“Spending phase of an attack”* is complete

Post-Condition: The dice have been rolled, and it is determined whether the attack is successful or not

Events:

1. Die generates a random integer between 2 and 12, inclusively
2. Roll is less than attack modifier and less than 11
3. Attack is successful

Extensions:

1a: Player plays *Assassination*

1a.1: Roll is set to 2

1a.2: Rejoin step 2

1b: Player plays *Murphy’s Law*

1b.1: Roll is set to 12

1b.2: Rejoin step 2

2a - 3a: Number is greater than attack modifier or greater than 10

2a.1: Attack is unsuccessful

2a.2: End Use Case

**Test Case**:

Success: Post-Condition fulfilled: The dice have been rolled, and it is determined whether the attack is successful or not

|  |  |  |
| --- | --- | --- |
| **Test ID** | **Test Description** | **Expected Results** |
| 1 | Check if the die is generating random numbers from 2 to 12, inclusively   1. Run function Die.roll() to get a large sample size of values (e.g. 100) | Check if all of the results are all integers between 2 and 12, inclusively |
| 2 | Check if a roll less than the attack modifier and less than 11 is a success   1. Instead of using Die.roll(), give a specific input for the roll and attack modifier | A roll less than the attack modifier and less than 11 should return a success   1. For example, set roll to 7 and attack modifier to 8, check that the attack is a success. 2. Set roll to 11 and attack modifier to 8, check that attack is a failure |

**Use Case**: Successful attack aftermath, control

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: Attack was successful, attack type was control

Post-Condition: The defending group and its children are under the control of the attacking player or uncontrolled because it could not fit in the power structure. The MB of the new group and its children is reduced by half. MB is transferred from attacking group to new group, if desired. The attack counter of the attacking group and all aiding groups is reduced by one.

Events:

1. Prompt attacking player to place the newly acquired defending group
2. Player selects an outward control arrow on attacking group
3. Place new group’s inward control arrow adjacent to attacking group’s selected outward control arrow
4. Optional. Place the new group’s children on the same control arrow prior to the attack
5. Optional. Repeat step 4 for the children's children, and all groups further down the hierarchy
6. Repeat for new group and all children. Reduce MB of group by half (rounded down). Place MB in the bank
7. Prompt player to enter MB to transfer from attacking group to new group
8. Player enters MB to transfer
9. Remove the amount of MB entered from the attacking group
10. Add the amount of MB entered to the new group
11. Reduce attack counter of attacking group by 1
12. Repeat for all aiding groups. Reduce attack counter by 1

Extensions:

4a: Child cannot fit in hierarchy

4a.1: Prompt player to move child to a different control arrow

4a.2: Player selects an outward control arrow on parent group

4a.3: Place child’s inward control arrow adjacent to parent group’s outward control arrow

4a.4: Rejoin step 4.

4b: Child cannot fit in hierarchy

4b.1: Child does not fit on any of parent’s control arrows

4b.2: Place child and all of its children in the uncontrolled groups section

4b.3: Remove all MB from these groups

4b.4: Place the MB into the bank

4b.5: Rejoin step 4.

**Test Case**:

Success: Post-Condition fulfilled: The defending group and its children are under the control of the attacking player or uncontrolled because it could not fit in the power structure. The MB of the new group and its children is reduced by half. MB is transferred from attacking group to new group, if desired. The attack counter of the attacking group and all aiding groups is reduced by one.

Failure:

1. Player does not gain control of the newly acquired group
2. Children of the new group are not under the player’s control (unless the children do not fit)
3. Cards overlap in the new power structure
4. A child fits in the structure on its current control arrow, but the player is allowed to change the control arrow
5. The attack counter is not decremented for each attacking or aiding group
6. The MB of the new group and its children is not cut in half
7. MB is not transferred properly between the attacking group and the new group

**Use Case**: Successful attack aftermath, neutralize

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: Attack was successful, attack type was neutralize

Post-Condition: The defending group and its children are placed in the uncontrolled area. The MB of the neutralized group and its children is reduced to zero. The attack counter of the attacking group and all aiding groups is reduced by one.

Events:

1. Remove all MB from neutralized group and its children
2. Place neutralized group and its children in the uncontrolled area
3. Reduce attack counter of attacking group by 1
4. Repeat for all aiding groups. Reduce attack counter by 1

Extensions:

**Test Case**:

Success: Post-Condition fulfilled: The defending group and its children are placed in the uncontrolled area. The MB of the neutralized group and its children is reduced to zero. The attack counter of the attacking group and all aiding groups is reduced by one.

Failure:

1. Neutralized group and its children are not placed in the uncontrolled area.
2. Neutralized group and its children still have MB in its treasury.
3. The attack counter is not decremented for each attacking or aiding group.

**Use Case**: Successful attack aftermath, destroy

Actors: Player, GroupCard, IlluminatiCard

Pre-Condition: Attack was successful, attack type was destroy

Post-Condition: The defending group is placed in the “Dead” pile. Its children is placed in the uncontrolled area. The MB of the destroyed group and its children is reduced to zero. The attack counter of the attacking group and all aiding groups is reduced by one.

Events:

1. Remove all MB from destroyed group and its children
2. Place destroyed group in the “Dead” pile
3. Place its children in the uncontrolled area
4. Reduce attack counter of attacking group by 1
5. Repeat for all aiding groups. Reduce attack counter by 1
6. End Use Case

Extensions:

6a: Attacking player is *The Servants of Cthulu*

6a.1: Increase destroyedGroups counter by 1

**Test Case**:

Success: Post-Condition fulfilled: The defending group is placed in the “Dead” pile. Its children is placed in the uncontrolled area. The MB of the destroyed group and its children is reduced to zero. The attack counter of the attacking group and all aiding groups is reduced by one.

Failure:

1. Destroyed group is not placed in the “Dead” pile
2. Destroyed group’s children are not placed in the uncontrolled area
3. MB is not removed from the destroyed group and its children
4. The attack counter is not decremented for each attacking or aiding group.
5. If *The Servants of Cthulu* destroyed the group and the destroyedGroups counter is not incremented by 1

Use Case: Move Money

Actors: Player

Pre-Condition: The player’s Illuminati has money.

Post-Condition: The specified amount of money has been transferred to another player.

Events:

1. The Use Case begins when a player decides to give away money to another player and selects Move Money.
2. The player enters the amount of money and selects another player and then selects Move.
3. The application asks the player for a confirmation and the player selects Yes.
4. The application transfers the money from his or her Illuminati to the selected player’s Illuminati.
5. The Use Case ends.

Use Case: Use a Specialty

Actors: Player

Pre-Condition: The player has at least one special card.

Post-Condition: The used special card is discarded.

Events

1. The Use Case begins when a player selects Use a Specialty during his or her turn.
2. The player selects a special card and selects Use.
3. The application displays the instructions on the card and asks the player for a confirmation. The player selects Yes.
4. The application executes the instructions on the card and discards it. See the corresponding use case.
5. The Use Case ends.

Use Case: Drop a Group

Actors: Player

Pre-Condition: The player has at least one Group.

Post-Condition: The selected Group has been returned to the uncontrolled area.

Events:

1. The Use Case begins when a player selects Drop a Group during his or her turn.
2. The player selects a Group to drop and selects Drop.
3. The application asks the player for a confirmation and the player selects Yes.
4. The application removes the Group from the player’s Power Structure and returns it to the uncontrolled area.
5. The Use Case ends.

Use Case: Transfer Money

Actors: Player

Pre-Condition: The player has at least two Groups.

Post-Condition: The specified amount of money has been transferred from a Group to another Group.

Events

1. The Use Case begins when a player selects Transfer Money during his or her turn.
2. The player selects the following and selects Transfer:

· A Group to transfer money from

· A Group to transfer money to (must be an adjacent Group)

· The amount of money

1. The application asks the player for a confirmation and the player selects Yes.
2. The application transfers money from the first Group to the second Group.
3. The Use Case ends.

Use Case: Give a Group

Actors: Player

Pre-Condition: The player has at least one Group.

Post-Condition: The selected Group has been given to another player.

Events:

1. The Use Case begins when a player decides to give away a Group during his or her turn and selects Give a Group.
2. The player selects a Group and another player and then selects Give.
3. The application asks the player for a confirmation and the player selects Yes.
4. The application notifies the selected player that the player wants to give him or her a Group.

5. The selected player selects Allow, the application transfers the Group and its puppet (if any), along with all treasuries to the selected player and notifies the player.

1. The Use Case ends.

Extensions:

5A: The selected player select Reject.

5A.1: The application notifies the player that the selected player doesn’t accept the Group.

5A.2: The Use Case ends.

Use Case: Give a Card

Actors: Player

Pre-Condition: The player has at least one special card.

Post-Condition: The selected special card has been transferred to another player.

Events

1. The Use Case begins when a player decides to give away a special card to another player and selects Give a Card.
2. The player selects a special card and another player and then selects Give.
3. The application asks the player for a confirmation and the player selects Yes.
4. The application transfers the selected card to the selected player.
5. The Use Case ends.