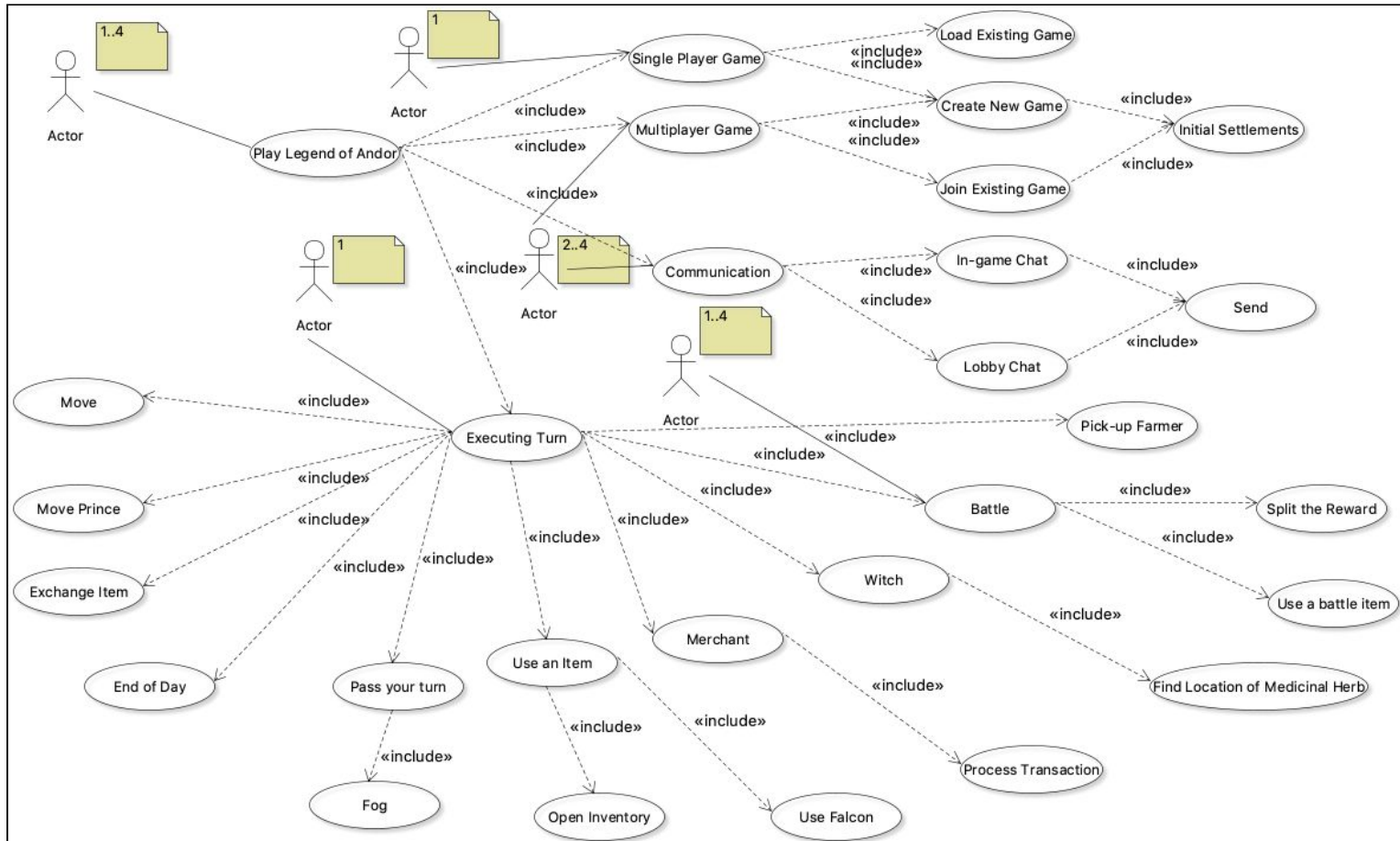


COMP 361D1 - Software Engineering Project

Project Milestone 2

Legend of Andor - Use Case Model
October 21, 2019



Use Case: Play Legend of Andor

Scope: Andor

Level: Summary Level

Intention in Context: The intention of the player is to play a game of Andor

Multiplicity: Many players can play the game concurrently. But a player cannot play multiple games.

Primary Actor: *Player*

Secondary Actor: *Other players, Google Play Games Services*

Main Success Scenario:

1. The *player* chooses to play a single Game (offline) or a multiplayer game (online).
2. *Player* log into the *system*. Players have to enter their credentials to communicate using the *Google Play Games Services*.
3. *Player* enters the lobby. It can choose to play an existing game or create a new game.
Step 3 is executed once enough players have joined the game.

4. *Players* choose their characters.
- Step 4 ends when all the players agree to start the game.*
5. *System* starts the initial settlement of game or the *system* load a previous game if one is saved.
6. *Players* execute turns.
7. The *system* informs if the players have won the legend or not.

Extensions:

3a. If players didn't succeed in play an existing game, use case continues at step 3.

Use case: Single Player Game

Scope: Andor

Level: User Goal

Intention in Context: The intention of the player is to play a game of Andor offline.

Multiplicity: All the players play on the same device. But no players can play a different game.

Primary Actor: *Player*

Secondary Actor: *Other players on the same device*

Main Success Scenario:

1. *Player* chooses an existing game or create a new game.
2. *Player* has to choose how many players are going to play.
3. *Player* select the character he wants.

System determines that enough players have joined.

Extensions:

1a. If there is no existing game, continue with new game creation.

1b. If there already exist a game go to step 5.

Use Case: Multiple Player Game

Scope: Andor

Level: User Goal

Intention in Context: The intention of the player is to play a game of Andor online.

Multiplicity: All the players play on a different devices. But no players can play a different game.

Primary Actor: *Player*

Secondary Actor: *Other players*

Main Success Scenario:

1. *Player* have to sign into the server.
2. *Player* enters into the lobby.
3. System displays the existing online team and the players online.
4. *Player* choose between creating a new game or joining an existing team.

System determines that enough players have joined.

Use Case: Join Existing Game

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the player is to join another team to play.

Primary Actor: *Player*

Secondary Actor: *Other players*

Main Success Scenario:

1. In the lobby, *system* displays all the available online team.
2. Player choose which team we want to join.
3. *System* assign the player to the team he chose and wait for other players.
4. Player can choose their character and inform the *system* when they are ready to start.

System determines that enough players have joined.

5. *System* start initial settlement.

Extensions:

4a. The *player* that create the game have to press ready to start the game.

4b. The game will only start if we have between 2 and 4 ready players.

Use Case: Create New Game

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the player is to create a new game so that other players can join.

Primary Actor: *Player*

Secondary Actor: *Other players*

Main Success Scenario:

1. *Player* choose which character he wants to be.
2. *System* wait for *other players* to join his existing team.
3. *Player* can invite *other online players* to join his team.
4. When player is ready and enough players have joined, the game can start.

System determines that enough players have joined.

5. *System* starts initial settlement.

Extensions:

(3-4) a. *Player* informs *system* that he wishes to cancel game creation. Use case ends in failure.

Use Case: Load Existing Game

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the Player is to load a previously saved game. Only available on Single player mode (offline).

Primary Actor: *Player*

Main Success Scenario:

1. *System* display game saved in the database.
2. *Player* informs *system* that he wishes to load a specific game.
3. *System* presents game board to *player*.
4. *Player* informs *system* that he is ready to begin game.

System determines that enough players have joined.

Extensions:

1a. If no game saved then go to create a new game use case.

Use Case: Initial Settlements

Scope: Andor

Level: Subsubfunction

Intention in Context: The game is set up and the group has to decide how the wineskins and gold are shared.

Multiplicity: Team of players

Primary Actor: *Player*

Secondary Actor: *Other players*

Main Success Scenario:

1. *System* displays the map of Andor.
2. *System* narrates the legend and gives 2 strength points to every player.
3. *System* gives 5 gold and 2 wineskins.
4. *Players* have to decide who gets what with their communication.
5. *Players* with lower rank execute turns first.

Use Case: Executing Turn

Scope: Andor

Level: User Goal

Intention in Context: Intention of the *player* is to perform their turn within a game of Andor.

Multiplicity: One *player* may execute their turn at a time. *Players* will execute a turn multiple times in *Game*

Primary Actor: *Player*

Secondary Actor: *Player* (other than *current player*)

Main Success Scenario:

1. The *system* informs the *player* that it is their turn.
2. The *system* informs the *player* of their hours remaining.
3. The *player* spends their available hours (step 3 may be repeated until the *player* runs out of hours).
4. The *system* informs the *player* that their turn has ended.
5. *System* processes the *Game State* and displays the result to all *Players*.
6. End of Day.

Extensions:

3a. *Player* performs a Move.

3b. *Player* performs a Battle.

3c. *Player* performs a Merchant.

3d. *Player* performs a Pick-up Farmer.

3e. *Player* performs a Witch.

3f. *Player* performs a Use an Item.

3g. *Player* performs an Exchange Item.

3h. *Player* performs a Move Prince.

3i. *Player* performs a Move.

3j. *Player* performs an End of Day.

3k. *Player* performs a Pass Your Turn.

3l. The *player* has chosen to move with no regular hours available, but has overtime hours available.

3l1. The *system* signals to the *player*, he or she must spend (2) willpower to move.

3l2. The *player* confirms this and the *system* will update the game board accordingly.

Player has insufficient willpower, return to Executing Turn, step 3

3l3. The case returns to Executing Turn, step 3.

Use Case: Move

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *player* is to *Move* to a specified tile.

Multiplicity: One *player* may *Move* one time. A *player* may *Move* more than once per turn.

Primary Actor: *Player*

Secondary Actor: *Player* (other than *current player*)

Main Success Scenario:

1. The *player* informs the *system* of their desire to *Move* to a chosen location on the game board.
2. The *system* validates that the *player* has enough hours to *Move* to the chosen location.
3. The *player* moves to chosen location, *system* updates game board for all *players*.
4. The case returns to Executing Turn, step 3.

Extensions:

2a. The *player* has chosen a tile outside his or her movement range.

2a1. The *system* signals to the *player* the *Move* was invalid.

2a2. Use case ends in failure

2a3. The case returns to Executing Turn, step 3.

2b. The *player* has chosen to move, with no more hours available.

2c1. The *system* signals to the *player* that the *Move* was invalid.

2c2. Use case ends in failure

2c3. The case returns to Executing Turn, step 3.

3a. The *player* moves into a tile with an item.

3a1. The *system* removes the item from game tile.

3a2. The *system* updates the *player's* inventory with said item.

3a3. The case returns to Executing Turn, step 3.

Use Case: Move Prince

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *player* is to *Move* the *prince* to a specified tile.

Multiplicity: One *player* may *Move* the prince at one time. The prince may move more than once per turn.

Primary Actor: *Player*

Secondary Actor: *Player* (other than *current player*)

Main Success Scenario:

1. The *player* informs the *system* of their desire to *Move* the *prince* to a chosen location on the game board.
2. The *system* validates that the *player* has enough hours to *Move* the *prince* to the chosen location.
3. The *prince* is moved to chosen location, *system* updates game board for all *players*.
4. The case returns to Executing Turn, step 3.

Extensions:

- 2a. The *player* has chosen a tile outside the prince's movement range.
 - 2a1. The *system* signals to the *player* the *Move* was invalid.
 - 2a2. The case returns to Executing Turn, step 3.
- 2b. The *player* has chosen to move the prince, with no more hours available.
 - 2c1. The *system* signals to the *player* that the move was invalid.
 - 2c2. The case returns to Executing Turn, step 3.
- 3a. The *player* informs the *system* the prince performs a Battle.

Use Case: Fog

Scope: Andor

Level: Subsubfunction

Intention in Context: The intention of the *Player* is to activate the fog tile.

Multiplicity: One *player* may activate a fog tile at a time. Fog tiles can only be activated once.

Primary Actor: *Player*

Secondary Actor: *Player* (other than *current player*)

Main Success Scenario:

1. The *player* begins their new day in a fog tile, the *system* removes the fog.
2. The fog event is triggered for the *player*.
3. *System* updates the game board.
4. The case returns to Executing Turn, step 3.

Extensions:

- 1a. The *player* is adjacent to fog and performs Use Item.
 - 1a1. The *system* validates the item used is a telescope.
 - 1a2. The *system* removes all fog tiles adjacent to the *player's* current tile.
 - 1a3. The case returns to Executing Turn, step 3.
- 2a. The fog contains strength, willpower points, wineskin, or gold for the *player*.
 - 2a1. The *system* updates the *player's* status.
 - 2a2. The case returns to Executing Turn, step 3.
- 2b. The *Fog* contains a *Gor*
 - 2b1. The *system* updates the board to include a gor on the now removed fog tile.
 - 2b2. The case returns to Executing Turn, step 3.
- 2c. The fog contains an Event Card.
 - 2c1. The *player* informs the *system* they wish to Use Item.
 - 2c2. The *system* validates item used was a shield.
 - 2c3. The Event Card activation is undone, the *system* updates board.
 - 2d2. The case returns to Executing Turn, step 3.

Use Case: Use Item

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *player* is to use an item in his/her inventory/bag.

Multiplicity: Only a single *player* can use an item, because every *player* takes turn to complete their moves. A *player* can use more than one item per turn.

Primary Actor: *Player*

Main Success Scenario:

1. *Player* opens inventory.
2. *Player* chooses an item he/she would like to use and presses it, requesting the *system* to use the item.
3. *System* executes the chosen item's effect on the *player*'s character.
4. *System* displays the *player* an inventory with the used-up item removed.

Extensions:

2a. It is not the beginning of the *player*'s turn anymore. The *player* is not allowed to use an item unless it is at the beginning of his/her turn (before moving/battling).

2b. Use case ends in failure.

3a. The selected item is an unusable item. Use case continues at step 2.

3b. If the hero that is the first to start the day with his or her turn and the hero has a shield, he or she can activate the shield to undo the effects of the event card, done by the *system*, that happened at sunrise in the End of Day use case.

3c. The selected item is a falcon, Use Falcon.

Use Case: Open Inventory

Scope: Andor

Level: Subsubfunction

Intention in Context: The intention of the *player* is to open his/her inventory to view the items in the *player*'s character's bag.

Multiplicity: Several *players* can open a character's inventory at the same time, given that the game is in a multiplayer mode currently (i.e. they would be opening the inventories in their own device). A given *player* can only open one inventory at a time.

Primary Actor: *Player*

Main Success Scenario:

1. The *player* presses the inventory/bag button for a particular character, requesting the *system* to display the items stored in the *player*'s character's bag/inventory.
2. The *system* displays the character's inventory on the screen for the *player*.

Use Case: Use Falcon

Scope: Andor

Level: Subsubfunction

Intention in Context: The intention of the *player* is to use the falcon item to exchange items.

Multiplicity: Only one *player* can use the falcon at a time because the game is turn-based. A *player* can only use one falcon at a time, as he/she can exchange as many small articles at a time as they like anyways.

Primary Actor: *Player*

Main Success Scenario:

1. The *system* highlights the other *players* for the current *player* to choose from.
2. The *player* informs the *system* which player he/she would like to exchange items with by selecting on screen.
3. The *players* exchange item.

Use Case: Exchange Item

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *player* is to exchange an item (including gold or gemstones) with another character in the same tile as the *player*.

Multiplicity: Only a single *player* can exchange an item with only one other *player* at a time.

Primary Actor: *Player*

Main Success Scenario:

1. The *player* informs the *system* with which *player* (on the same tile) he/she wants to exchange an item with by tapping on that player.
2. The *system* displays the *player*'s inventory, allowing the *player* to choose which item he/she would like to exchange.
3. The *player* selects an item from his/her inventory to exchange, requesting the *system* to complete the exchange.
4. The *system* informs the *player* that he/she has received a request to exchange an item from another *player*, by displaying the information and two buttons ('Accept'/'Decline').
5. The *player* either accepts or declines the exchange.
6. The *system* displays the result of the exchange to the *player*.

Extensions:

- 1a. The *player* has used the falcon, and thus would've already selected the other *player* to exchange an item with. Use case continues at step 2.
- 1b. The other *player*, that the current *player* selected to exchange an item with, is either not on tile with the *player* or the *player* has not used the falcon. Use case ends in failure.
- 3a. The selected item is a "large article" (shield or bow). Use case continues at step 3.
- 5a. The other *Player* declines the exchange request. Use case continues at step 3.
- 5b. The other *Player* accepts the exchange request. Use case continues at step 6.

Use Case: Merchant

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *Player* is to interact with the merchant.

Multiplicity: Since the game is turn-based, only one *player* can interact with the merchant at a time. A *player* can only interact with one merchant at a given time because the *player* must be on the merchant tile, but there are three merchant tiles far from each other.

Primary Actor: *Player*

Main Success Scenario:

1. The *player* (on a merchant tile) requests the *system* that he/she would like to interact with the merchant by pressing the merchant button.
2. The *system* displays the *player* with the merchant screen with necessary information (purchasable items, the *player*'s golds, etc.).
3. The *player* informs the *system* of the desired item by tapping on the item.
4. The *system* processes transaction.

Extensions:

1a. The *player* is not on a merchant tile. The use case ends in failure.

Use Case: Process Transaction

Scope: Andor

Level: Subsubfunction

Intention in Context: The intention of the *system* is to process the transaction of the *player* and the merchant.

Multiplicity: Since the game is turn-based, there can only be one instance of *ProcessTransaction* at a time.

Primary Actor: *System*

Main Success Scenario:

1. The *system* presents the *player* with his/her updated inventory with the purchased item added and the price of the item deducted from the *player*'s golds.

Extensions:

1a. The *player* does not have enough gold in his/her inventory to purchase the item. The use case ends in failure.

Use Case: End of Day

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *system* is to declare the end of day and start a new day (reset the game board) as all *players* end their day.

Multiplicity: The *system* declares the End of Day once per day (letter space on the legend track).

Primary Actor: *System, narrator*

Secondary Actor: *All Players*

Main Success Scenario:

1. The system is informed to declare the End of Day as all players end their day (all of their time markers are placed in the sunrise box).
2. The system informs all players of the next instructions if any.
3. The system moves all gors. The system refreshes all wells.
4. The system advances the narrator one step on the legend track.

Extensions:

- 1a. Not all *players* has ended their day
 - 1a1. Case returns to Executing Turn of the next *player*.
 - 1a2. Use case ends in failure
- 3a. There are other creatures on the game board other than gors.
 - 3a1. The system moves them according to the same rules applied to the gors.
- 3b. There is already a creature occupying a given space.
 - 3b1. Since no two creatures are allowed on one given space, the system moves the creature (which it was moving) to the next space the arrow is pointing to.
- 4a. If there is a player standing on a space with the well, the well will not be refreshed. In this case, the token is left alone.
- 5a. If the narrator reaches a letter space with a star sign (there are letters with no stars), the system makes the corresponding announcements to all players (shuffled from one of the Golden Event Cards).

Use Case: Pass Your Turn

Scope: Andor

Level: Subfunction

Intention in Context: The intention of *player* is to pass their turn without performing any moves.

Multiplicity: One *player* may pass at one time. It is impossible for a *player* to pass more than once per turn.

Primary Actor: *Player*

Secondary Actor: *System*

Main Success Scenario:

1. The *player* informs system that they wish to skip their turn without performing anymore moves.
2. The *system* validates the request.
3. The *system* informs all *players* of the result of current player's decision.
4. The *system* the time marker of the *player* by one hour.
5. The case returns to Executing Turn of the next *player*.

Extensions:

- 1a. It is the start of day and *player* gets to start on the same space with a gor (or another creature).
 - 1a1. The *system* rejects the *player*'s request to rest.
 - 1a2. The *system* initiates Battle with the creature.
- 1b. The *player* Passes their turn in a Fog.

Use Case: Pick Up Farmer

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *player* is to pick up a farmer and carry the farmer to the castle.

Multiplicity: All *players* may pick up farmer(s) at their turn.

Primary Actor: *Player*

Secondary Actor: *System, creatures*

Main Success Scenario:

1. *Player* moves to one space, encounters a farmer.
2. *Player* informs the system that they want to pick up the farmer.
3. *System* validates the request. The timeline of the *player* is advanced by an hour.
4. The case returns to Executing Turn of the same *player*.

Extensions:

1a. *Player* is already carrying a farmer.

1a1. *Player* is allowed to carry more than one farmer with them.

4a. *Player* may choose to drop the farmer on any space at any time.

4b. *Player* enters a space with a *creature*.

4b1. *System* removes farmer from the game (farmer is killed).

4c. A *creature* enters a space where *player* and farmer(s) stand.

4c1. *System* removes farmer from the game (farmer is killed).

4d. *Player* enters the castle with farmer(s).

4d1. *System* changes the farmer(s) into golden shields and the shields are placed inside the castle. The shields help defend the castle when a creature enters the castle.

Use Case: Witch

Scope: Andor

Level: Subfunction

Intention in Context: *Player* finds witch finding in a fog.

Multiplicity: Once the location of the witch is found, all *players* may approach the witch.

Primary Actor: *Player*, *Witch*

Secondary Actor: *System*

Main Success Scenario:

1. *Player* finds the witch hiding in one of the “fogs”.
2. The *system* rewards the *player* with one *witch*’s brew.
3. The *system* informs the location of the *witch* to all players.
4. The *witch* tells player the location of the medicinal herb. The player rolls the dice to find out the location of the medicinal herb.
5. System informs the location of the medicinal herb to all players.
6. The case returns to Find Location of Medicinal Herb.

Extensions:

3a. Another *player* moves to the space where witch stands.

3a1. This *player* may buy the witch’s brew with its gold (price varies depending on the character of player).

Use Case: Find Location of Medicinal Herb

Scope: Andor

Level: Subsubfunction

Intention in Context: The intention of *player* (whoever is first to find the witch) decides where the medicinal herb is located by rolling dice.

Multiplicity: Once the location of the herb is revealed, all *players* may approach it.

Primary Actor: *Player*

Secondary Actor: *System*

Main Success Scenario:

1. *Player* finds the witch hiding in one of the “fogs”.
2. *Player* is to roll a dice to determine the position of the medicinal herb.
 - a. roll 1 or 2: medicinal herb on space 37.
 - b. roll 3 or 4: medicinal herb on space 67.
 - c. roll 5 or 6: medicinal herb on space 61.
3. The *system* locates the medicinal herb on the determined location.
4. The *system* locates a gor on the same determined location.

Use Case: Battle**Scope: Andor****Level: Subfunction**

Intention in Context: The intention of the *player* is to engage in a fight against a creature when the *player*'s hero has landed on a position containing a creature or is on an adjacent space from a creature (the hero is an archer or the hero is carrying a bow). The player decides voluntarily to engage battle, at the beginning of his or her turn, alone or in pairs if another *player*'s hero is located nearby on an adjacent space or on the same space.

Primary Actor: *Player* or pair (consists of 2 players or more - can also be a group of players) of *players* (warrior, archer, wizard, dwarf)

Secondary Actor: *Creature, witch*

Main Success Scenario:

1. *Player* informs the *system* to initiate a solo battle or one in pairs of two with a second *player* against a *creature*.
 - a. In a group fight, the hero whose turn it is invites the other heroes to battle. The time markers of all the heroes participating in the collective battle are advanced 1 hour.
2. Each *player* can decide, by informing the *system*, to use a battle item that he or she has in their backpack when it is their respective turn at the beginning of their turn to which the *system* updates the respective *player*'s statistics or damage value upon calculation.
3. For every *player*'s turn, their set of dice is rolled depending on the type of their hero and the *system* informs the player of their role.
 - a. For *players* with particular heroes, that role die differently, one dice is displayed at a time with the option to skip the dice until the last is rolled.
 - i. The *player* informs the *system* that he or she confirms the dice roll.
4. The total damage value (either solo or in a group) is calculated by the *system* for the solo *player* or the pair of *players* based on the dice available for the *player*.
 - a. The calculation considers any items that the hero is wearing or that he or she has purchased.
 - b. The hero adds his or her current strength points to the highest roll. This yields the damage value.
 - c. The collective damage value is calculated by first adding up the strength points of all the participating heroes. Then the hero who invited the others rolls the dice, followed by all the other participating heroes in turn, and each

highest die value is noted. These highest values are added to the collective strength points, yielding the collective damage value for all the heroes.

5. The *creature*'s damage value is calculated by the *system* based on the type of monster
 - a. Gors, skrals, and trolls roll with red dice; wardraks roll black dice.
 - b. The highest die value is what counts for the creature as well. For creatures, though, identical dice always have their values added together if it yields a higher total.
 - c. Then, the creature's strength points are added to its highest die value. This yields its battle value.
6. The damage on each side is assessed and the *system* informs the *players* of the result.
 - a. The difference between the two damage values is deducted from the willpower points of the defeated side. The archer can lose willpower points even if he attacks from an adjacent space.
 - b. In a group fight, if the collective damage value is lower than the creature's damage value, each of the heroes loses the difference in willpower points.
7. Each character's statistics are updated accordingly by the *system* and displayed.
 - a. The *system* informs the *players* of the state of the battle.
 - i. Win
 - ii. Lost
 - iii. Draw
 - iv. Round number
 1. In each battle round, the hero's time marker is advanced one space on the time track.
8. The battle is continued from step 2 onwards until:
 - a. The solo *player* or the pair of *players* win the battle and the *creature* dies.
 - i. The *creature* is removed from the map.
 - ii. The *player* or *players* are prompted by the *system* with the merchant to split the reward to which everyone returns to the map.
 - iii. The *system* advances the narrator by one space.
 - b. The *creature* wins the battle and the *players* in battle die.
 - c. There is a draw.
 - i. Nothing happens in a draw.
 - ii. The battle is continued (continue from step 2).
9. On return to the map, the *players*' timeline, statistics and backpack (if applicable) are updated by the *system* and displayed, as well as, the *creature* depending on the end state of the battle.
 - a. The *creature* either stays alive on the same position or gets removed from the map due to losing the battle.
 - b. If the solo *player* or the pair of *players* won the battle, their backpacks are updated based on the prize they chose to repartition.
 - c. The *players*' statistics are updated according to the *players*' end state of the battle.
 - i. Winning carries through with the existent statistics.
 - ii. Losing engages revival.
 - iii. Backing out can make the creature's statistics reset.

10. A hero is allowed to transfer an article to a fellow *player* at the end of a battle round (if both are on the same space or one of the two is using the falcon - Exchange use case).

Extensions:

3a. The *player* is using the *wizard*'s power on his dice roll.

3a1. If the *wizard* wants to use his special ability to turn the die of another participating hero onto its opposite side rather than turning his own die, he has to make that decision immediately after the roll.

3a2. The *witch* must inform the system of this action before all the heroes have rolled.

3a3. The *system* must calculate the damage value accordingly.

3a4. Use case continues at step 4.

8a. At any point in time during the battle, any *player* is allowed to back out of the battle, thus, informing the *system* of his or her intention to go back to the map and exit battle.

8a1. The *player* might back out because he or she doesn't have enough willpower or time track and can't return the battle.

8a1.1. That *player*'s statistics remain the same as the ones in battle before backing out (*system* updates).

8a1.1.1 A hero who leaves the battle because he has 0 willpower points loses 1 strength point and gains 3 willpower points.

8a1.1. The *creature*'s statistics are reset if it's left with no more opponent to fight (*system* updates).

8b. Upon death of the *player*'s hero, the hero is revived on the map, exiting the battle immediately.

8b1. The hero loses 1 strength point, assuming he or she still has more than 1.

8b2. He or she also receives 3 willpower points.

8b3. The turn goes to the next *player* in the clockwise direction of the *player* who originally invited the others to battle if it's a group fight (Executing Turn use case).

Use Case: Use a Battle Item

Scope: Andor

Level: Subsubfunction

Intention in Context: It is the *player*'s turn, as the *system* informs by highlighting the hero on the battle screen, and he or she presses on their hero to open their backpack. He or she then selects an item to use during battle to help them defeat the opponent at the beginning of his or her turn before rolling any dice.

Primary Actor: *Player*

Main Success Scenario:

1. The *system* informs the *player* that it is his or her turn by highlighting their hero icon on the screen.
2. The respective *player* decides to use an item that they purchased and informs the *system* that they want to open their backpack by clicking on their hero's icon.
3. A floating window is displayed by the *system* to show the available items that the *player* can choose from (similar to the hero board).
4. The *player* informs the *system* which of the items he or she wishes to activate and confirms.
5. The *system* updates the battle screen based on the effect of the activated item.

6. The battle is resumed.

Extensions:

2a. The *player* can choose to not use an item in the end to which he or she informs the *system* that they want to exit the backpack window and return to the battle to resume it.

4a. Items to use are anything in the backpack, the *witch*'s brew or a medicinal herb token.

Use Case: Split the Reward

Scope: Andor

Level: Subsubfunction

Intention in Context: The battle has been won by the solo *player* or the pair of *players* and the merchant (the *system*) provides the winners with a prize to split amongst themselves. After agreeing on how the reward will be split, the *players* confirm their choice to the *system*. The *system* updates each *players*' backpack or statistics accordingly.

Primary Actor: *Player*

Main Success Scenario:

1. The *player* or the group has defeated the creature.
2. The *system* prompts the *players* with the merchant to split their prize (gold or willpower points, the size of the reward is indicated on the *creature*'s characteristics) amongst themselves for the *players* who survived till the final battle round (merchant screen).
3. After informing the *system* on how the prize will be partitioned, the *players* must confirm their selection.
4. Upon confirmation, the *system* updates the *players*' backpack and statistics and the battle screen is exited to the map (Execution Turn use case).

Use Case: Communicate

Scope: Andor

Level: User Goal

Intention in Context: The intention of the *player* is to communicate with other players.

Primary Actor: *Player*

Main Success Scenario:

1. *Player* confirms Lobby Chat.
2. *Player* confirms In Game Multiplayer Chat.

Use Case: Lobby Chat

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *Player* is to communicate with other players in the lobby before/after game.

Primary Actor: *Player*

Main Success Scenario:

1. *Player* requests *system* that he/she would like to chat by pressing chat button in the lobby.
2. *System* presents the *player* with a keyboard.
3. *Player* informs *system* what he/she wants to say by typing on the keyboard.
4. *Player* confirms send.

Use Case: In Game Multiplayer Chat

Scope: Andor

Level: Subfunction

Intention in Context: The intention of the *player* is to communicate with other players in-game.

Primary Actor: *Player*

Main Success Scenario:

1. *Player* requests *system* that he/she would like to chat by pressing chat button in-game.
2. *System* presents the *player* with a keyboard.
3. *Player* informs *system* what he/she wants to send a message by typing on the keyboard.
4. *Player* confirms send.

Use Case: Send

Scope: Andor

Level: Subsubfunction

Intention in Context: The intention of the *player* is to send a message that he/she wants to say.

Primary Actor: *Player*

Main Success Scenario:

1. *Player* informs *system* with the message that he/she typed.
2. *System* displays the player's message on the chat display for other *players*.
3. *System* notifies *players* of the new message.