

Team: Darren White
Kristen Hanslik
Robert Kendl
John Murphy

Title: RPG Game: Goblin Party 1979

Project Summary: We are making a graphics based computer game that allows a user to make a playable character, with a range of customizable attributes. This character will walk around a map and interact with different types of goblins, some good which can be added to his party as allies, and some bad which can be fought for experience. Through playing the game the user will watch as different events have an effect on the player and how the world and goblins interact with him.

Functional Requirements:

ID	Requirement	Topic Area	User	Priority
FR-1	There shall be 4 options to choose from to customize each of the 3 attributes of the character.	Character Creation	User	Critical
FR-2	The users strength shall increase by a set amount of points for each goblin it defeats.	Gameplay	User	High
FR-3	The user shall be able to make friends with a good goblin by offering the goblin a set amount of goblin treats.	Gameplay	User	High
FR-4	When a user acquires 5 goblin friends, they will be	Gameplay	User	Medium

	awarded the title, "Goblin Master," which will provide a boost to their health attribute.			
FR-5	When a user has defeated 10 evil goblins, they will be awarded the title, "Goblin Destroyer," which will provide a boost to their skill attribute.	Gameplay	User	Medium

User Requirements:

ID	Requirement	Topic Area	User	Priority
UR-1	As a user I need to customize three attributes of my character so that I can have a personalized game experience.	Character Creation	User	Critical
UR-2	As a user I need to be able to battle goblins, and keep track of total number of goblins killed so that I can advance in the game.	Gameplay	User	High
UR-3	As a user I need to be able to make friends with nice goblins and have them	Gameplay	User	Medium

	follow me so that I have them to assist me on my journey.			
UR-4	As a user I need to be able to pick up different types of weapons with different attributes, so that I can have control over the fighting style of my player.	Gameplay	User	High
UR-5	As a user I want to move my character around the map, avoiding pit obstacles as I explore, so that I can advance in the game.	Gameplay	User	Medium
UR-6	As a user I need to be able to collect Goblin Coins so that I can use them make friends with nice Goblins.	Gameplay	User	High

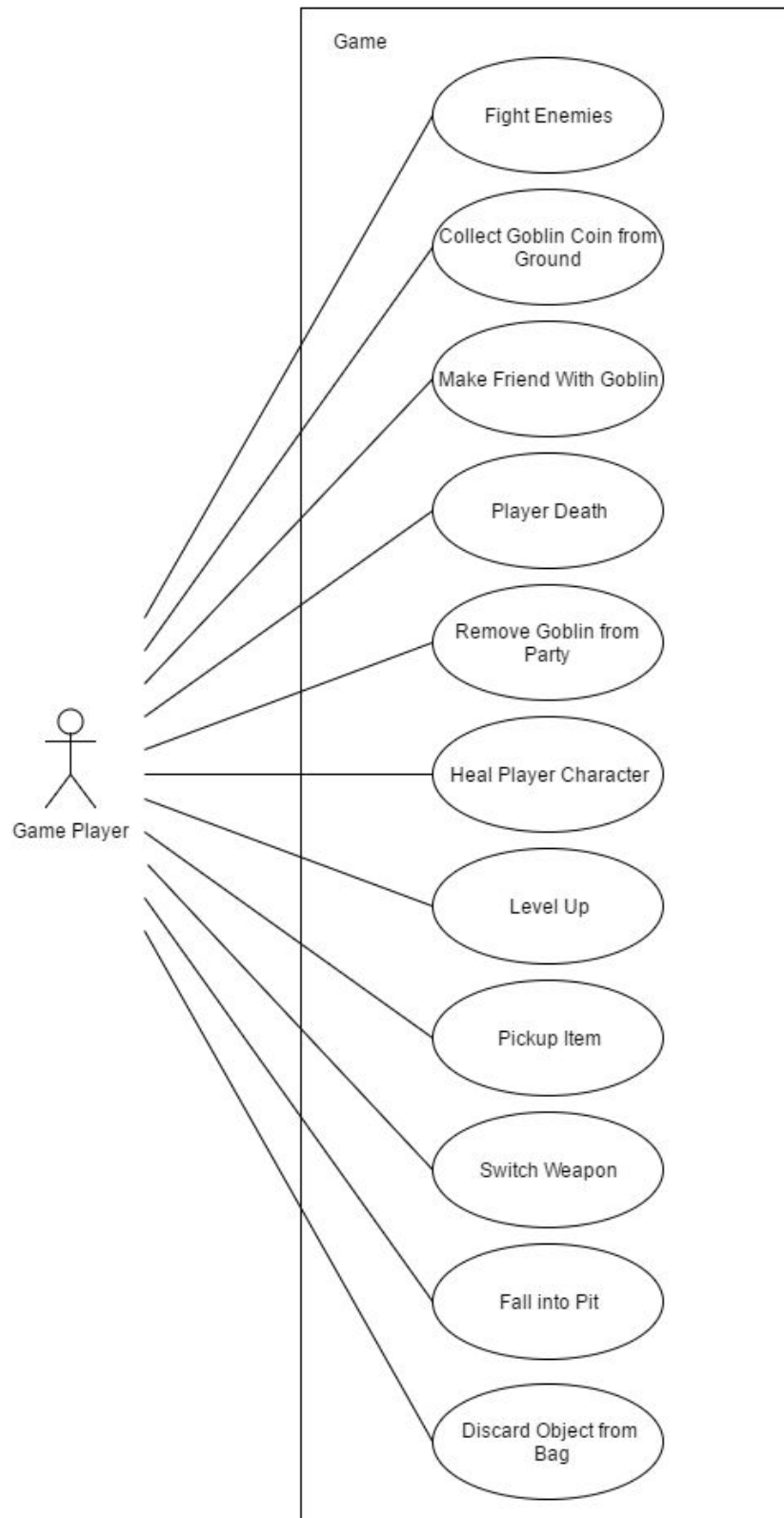
Non-Functional Requirements

ID	Requirement	Topic Area	User	Priority
NFR-1	The user shall be able to start the game using a minimum of 2 commands, once they are in the folder that	User Experience	Admin	High

	contains the game.			
NFR-2	The game should be pleasing to the eye, with bright colors and simple shapes.	User Experience	All	High
NFR-3	The game should have a small file size, no more than 1MB.	Back End	Admin	Medium

There are no Business Requirements

Use Cases:



Use Case ID:	UC - 01									
Use Case Name:	Fight Enemy									
Use Case Description:	The player character must be able to fight the enemies they encounter throughout the world. After successfully defeating an enemy, the player will get an amount of experience points towards their next level.									
Actors:	Player Character									
Pre-Condition:	There must be a player character and an enemy goblin in the world									
Post-Condition:	The enemy goblin will be destroyed and the player will have received experience points									
Frequency of Use:	Every conflict with an enemy (a lot)									
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>1. Player moves up to enemy</td><td></td></tr><tr><td>2. Player attacks enemy</td><td>Enemy attacks back</td></tr><tr><td>3. Player attacks enemy enough to reduce its number of health points to zero</td><td>Enemy is destroyed, experience is given to the player</td></tr></table>		Actor Action	System Response	1. Player moves up to enemy		2. Player attacks enemy	Enemy attacks back	3. Player attacks enemy enough to reduce its number of health points to zero	Enemy is destroyed, experience is given to the player
Actor Action	System Response									
1. Player moves up to enemy										
2. Player attacks enemy	Enemy attacks back									
3. Player attacks enemy enough to reduce its number of health points to zero	Enemy is destroyed, experience is given to the player									
Variations:	More powerful enemies give more experience.									
Exceptions:	N/A									
Developer Notes:	N/A									

Use Case ID:	UC - 02	
Use Case Name:	Collect Goblin Coin from Ground	
Use Case Description:	Main character stands within range of Goblin Coin, presses “c” to	

	initiate the collection of the coin. Coin bag shows the previous coin amount incremented by the value of the coin collected.				
Actors:	Player Character				
Pre-Condition:	There must be a player character and coin of some value in the world.				
Post-Condition:	The player's Goblin Coin amount will reflect an updated amount, updated by the amount of the Goblin Coin, of Goblin Coins of the type picked up if the acquisition was successful. The amount of items in the user's bag will also be increased by the value of the coin.				
Frequency of Use:	Every time the user presses "c".				
Flow of Events:	<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>1. Press "c" to attempt collect a coin</td><td> <ul style="list-style-type: none"> • If player is not within range of coin, end case. • If player does not have enough room in bag, notify user that there is not enough room and end case. • If player is within range of coin and there is enough room in bag, update that type of Goblin Coin's amount in the Bag. </td></tr> </tbody> </table>	Actor Action	System Response	1. Press "c" to attempt collect a coin	<ul style="list-style-type: none"> • If player is not within range of coin, end case. • If player does not have enough room in bag, notify user that there is not enough room and end case. • If player is within range of coin and there is enough room in bag, update that type of Goblin Coin's amount in the Bag.
Actor Action	System Response				
1. Press "c" to attempt collect a coin	<ul style="list-style-type: none"> • If player is not within range of coin, end case. • If player does not have enough room in bag, notify user that there is not enough room and end case. • If player is within range of coin and there is enough room in bag, update that type of Goblin Coin's amount in the Bag. 				
Variations:	N/A				
Exceptions:	The player may not have room in their bag for the amount of Goblin Coins the coin is worth.				
Developer Notes:	N/A				

Use Case ID:	UC - 03
Use Case Name:	Make Friend with Goblin

Use Case Description:	The player character must be able to use the coins that they have collected from defeating the Goblins they face. The user will then be able to use these coins to persuade good Goblins to join their quest.									
Actors:	Player Character									
Pre-Condition:	There must be a player character, the character has to have coins collected, there is a good Goblin in the world									
Post-Condition:	The good Goblin will consume the coin and follow the main character in the world									
Frequency of Use:	Every interaction with good Goblin where User has coins									
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>1. Player Moves up to good Goblin</td><td><ul style="list-style-type: none">• If contents of bag is greater than cost of Goblin, system prompts user to add goblin or not• If contents of bag is less than cost of Goblin, system prompts user they have insufficient coins to add Goblin</td></tr><tr><td>2. User attempts to give Goblin coin</td><td><ul style="list-style-type: none">• If number of Goblins in character's party is smaller than X. Assign good Goblin to user' party• If number of Goblins in character's party is larger than X. Give a prompt to player to remove good Goblin or continue in world</td></tr><tr><td>3. User gains good Goblin</td><td>Goblin follows character in world</td></tr></table>		Actor Action	System Response	1. Player Moves up to good Goblin	<ul style="list-style-type: none">• If contents of bag is greater than cost of Goblin, system prompts user to add goblin or not• If contents of bag is less than cost of Goblin, system prompts user they have insufficient coins to add Goblin	2. User attempts to give Goblin coin	<ul style="list-style-type: none">• If number of Goblins in character's party is smaller than X. Assign good Goblin to user' party• If number of Goblins in character's party is larger than X. Give a prompt to player to remove good Goblin or continue in world	3. User gains good Goblin	Goblin follows character in world
Actor Action	System Response									
1. Player Moves up to good Goblin	<ul style="list-style-type: none">• If contents of bag is greater than cost of Goblin, system prompts user to add goblin or not• If contents of bag is less than cost of Goblin, system prompts user they have insufficient coins to add Goblin									
2. User attempts to give Goblin coin	<ul style="list-style-type: none">• If number of Goblins in character's party is smaller than X. Assign good Goblin to user' party• If number of Goblins in character's party is larger than X. Give a prompt to player to remove good Goblin or continue in world									
3. User gains good Goblin	Goblin follows character in world									

Variations:	Bigger good Goblins cost more coins
Exceptions:	Too many Goblins in party, cannot add new good Goblin
Developer Notes:	N/A

Use Case ID:	UC - 04									
Use Case Name:	Player Death									
Use Case Description:	If the player character takes too much damage without healing, the player character will be killed and be given a “Game Over” screen									
Actors:	Player Character									
Pre-Condition:	The player character must have zero health points									
Post-Condition:	The player will be put on a game over screen and given prompts for the main menu or for a level restart									
Frequency of Use:	Every time the player loses all of their health points									
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>Player takes enough damage to reduce health to zero</td><td>Player is given a death animation and a “Game Over” screen. Player is given prompts for “Restart Level” or “Main Menu”</td></tr><tr><td>User picks “Restart Level”</td><td>Restarts the level that the player had died in</td></tr><tr><td>User picks “Main Menu”</td><td>User is taken back to main menu screen</td></tr></table>		Actor Action	System Response	Player takes enough damage to reduce health to zero	Player is given a death animation and a “Game Over” screen. Player is given prompts for “Restart Level” or “Main Menu”	User picks “Restart Level”	Restarts the level that the player had died in	User picks “Main Menu”	User is taken back to main menu screen
Actor Action	System Response									
Player takes enough damage to reduce health to zero	Player is given a death animation and a “Game Over” screen. Player is given prompts for “Restart Level” or “Main Menu”									
User picks “Restart Level”	Restarts the level that the player had died in									
User picks “Main Menu”	User is taken back to main menu screen									
Variations:	Could vary upon which enemy attack kills the player									
Exceptions:	N/A									
Developer Notes:	N/A									

Use Case ID:	UC - 05									
Use Case Name:	Delete Goblin from Party									
Use Case Description:	The player character must be able drop unwanted good Goblins from player party to be able to add new, more powerful good Goblins.									
Actors:	Player Character									
Pre-Condition:	There must be a player character, the character has the maximum amount of Goblins in party									
Post-Condition:	The chosen Goblin will be eliminated from the party.									
Frequency of Use:	Every interaction with good Goblin where User has a full party and wants to eliminate Goblin to clear space for new one.									
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>1. Walks up to good Goblin, having full party, with sufficient funds for Goblin</td><td>User is prompted to either edit Goblin party, or continue in world</td></tr><tr><td>2. User chooses to edit good Goblin party</td><td>Good Goblins are presented to the player</td></tr><tr><td>3. User chooses Goblin to eliminate</td><td>Goblin disappears in a cloud of smoke. Then the good Goblin is assigned to User's party</td></tr></table>		Actor Action	System Response	1. Walks up to good Goblin, having full party, with sufficient funds for Goblin	User is prompted to either edit Goblin party, or continue in world	2. User chooses to edit good Goblin party	Good Goblins are presented to the player	3. User chooses Goblin to eliminate	Goblin disappears in a cloud of smoke. Then the good Goblin is assigned to User's party
Actor Action	System Response									
1. Walks up to good Goblin, having full party, with sufficient funds for Goblin	User is prompted to either edit Goblin party, or continue in world									
2. User chooses to edit good Goblin party	Good Goblins are presented to the player									
3. User chooses Goblin to eliminate	Goblin disappears in a cloud of smoke. Then the good Goblin is assigned to User's party									
Variations:	Arrangement of Goblins									
Exceptions:	N/A									
Developer Notes:	N/A									

Use Case ID:	UC - 06
Use Case Name:	Heal Player Character

Use Case Description:	Player will be able to use a health potion or player will be healed while walking the map at a rate per minute proportional to number and strength of goblin followers					
Actors:	Player Character, goblin followers					
Pre-Condition:	The player character must have any amount of lost health points, as well as a health potion or goblin followers					
Post-Condition:	The player will be given an amount of health up to the cap of the potion or amount related to the rate determined by goblin followers					
Frequency of Use:	Every time the player wants to heal themselves manually via health potion, as well as every minute the user spends on the main map.					
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>Player uses Good Goblin ability or Health Potion</td><td>Removes health potion from inventory or put Good Goblin ability on cooldown. Player is healed up to the max amount for the source</td></tr></table>		Actor Action	System Response	Player uses Good Goblin ability or Health Potion	Removes health potion from inventory or put Good Goblin ability on cooldown. Player is healed up to the max amount for the source
Actor Action	System Response					
Player uses Good Goblin ability or Health Potion	Removes health potion from inventory or put Good Goblin ability on cooldown. Player is healed up to the max amount for the source					
Variations:	Varies upon strength of healing source					
Exceptions:	Player has no health potions or healing goblins left					
Developer Notes:	N/A					

Use Case ID:	UC - 07
Use Case Name:	Level Up
Use Case Description:	Once the player gains enough experience, they will level up, gaining extra health, strength, increased bag capacity, and a new goblin teammate space (every 5 levels)
Actors:	Player Character
Pre-Condition:	The player character must have the required amount of

	experience to level up				
Post-Condition:	The player will power up and start at zero experience towards the next level				
Frequency of Use:	Every time the player can level up				
Flow of Events:	<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>Player gets enough experience to level up</td><td>Increases health, damage, bag capacity and new goblin teammate (every 5 levels). Removes experience from bar and gives a new bar for the next level</td></tr> </tbody> </table>	Actor Action	System Response	Player gets enough experience to level up	Increases health, damage, bag capacity and new goblin teammate (every 5 levels). Removes experience from bar and gives a new bar for the next level
Actor Action	System Response				
Player gets enough experience to level up	Increases health, damage, bag capacity and new goblin teammate (every 5 levels). Removes experience from bar and gives a new bar for the next level				
Variations:	N/A				
Exceptions:	Once the player hits level 20, they cannot level up anymore				
Developer Notes:	N/A				

Use Case ID:	UC - 08				
Use Case Name:	Pick Up Item				
Use Case Description:	When a player character walks over an item, they will get a prompt to pick up the item. If they press "E" they will attempt to pick up the item, if the size of their bag allows it.				
Actors:	Player Character				
Pre-Condition:	The player must be standing over the item				
Post-Condition:	The item will be gone from the world and present in the player's bag				
Frequency of Use:	Every time the player wants to pick up an item				
Flow of Events:	<table border="1"> <thead> <tr> <th>Actor Action</th><th>System Response</th></tr> </thead> <tbody> <tr> <td>Player stands on item</td><td>Prompts player to press "E" to</td></tr> </tbody> </table>	Actor Action	System Response	Player stands on item	Prompts player to press "E" to
Actor Action	System Response				
Player stands on item	Prompts player to press "E" to				

		pick up item
	Player presses "E"	Item will be put into player's inventory and removed from the game world
Variations:	Different sized items will take a larger space in the bag	
Exceptions:	The player's bag doesn't have enough room to carry the item	
Developer Notes:	N/A	

Use Case ID:	UC - 09							
Use Case Name:	Switch Weapon							
Use Case Description:	When the player has a new weapon that they want to equip, they can switch that weapon with their current in-hand weapon if they have the level required to wield the weapon.							
Actors:	Player Character							
Pre-Condition:	The player must have the weapon in their bag							
Post-Condition:	The new weapon will be in the player's hand while the old one will be stashed away back in the player's bag							
Frequency of Use:	Every time the player wants to pick up an item							
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>Player attempts to switch weapon from inventory screen</td><td>Checks to see if the player has a high enough level to wield the weapon. If they do, the weapon will switch places with the player's current weapon</td></tr><tr><td></td><td>Player will now be seen on screen as wielding the new weapon, and will be able to use it</td></tr></table>		Actor Action	System Response	Player attempts to switch weapon from inventory screen	Checks to see if the player has a high enough level to wield the weapon. If they do, the weapon will switch places with the player's current weapon		Player will now be seen on screen as wielding the new weapon, and will be able to use it
Actor Action	System Response							
Player attempts to switch weapon from inventory screen	Checks to see if the player has a high enough level to wield the weapon. If they do, the weapon will switch places with the player's current weapon							
	Player will now be seen on screen as wielding the new weapon, and will be able to use it							

Variations:	Different weapons give different attributes to the player
Exceptions:	The player is not a high enough level to wield the weapon
Developer Notes:	N/A

Use Case ID:	UC - 10							
Use Case Name:	Fall into pit							
Use Case Description:	Main character while exploring the world falls into a pit on the main screen, injuring both himself and his pack of goblin followers.							
Actors:	Player Character, Goblins							
Pre-Condition:	There must be a player character and an open pit he walks into.							
Post-Condition:	The player's health stat will be decreased by 5 points, as well as the health stat of all goblin followers being decreased by 5 points. If the player or goblin does not have sufficient health that actor will be killed.							
Frequency of Use:	Every time there is a pit on the screen that the player walks over.							
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>1. Player walks over pit on screen</td><td><ul style="list-style-type: none">Animation is played showing the character and any followers falling into the pit, and deletes 5 health from player</td></tr><tr><td>2. Player uses arrows to point which direction of pit to appear out of</td><td><ul style="list-style-type: none">System sets user on outside of pit</td></tr></table>		Actor Action	System Response	1. Player walks over pit on screen	<ul style="list-style-type: none">Animation is played showing the character and any followers falling into the pit, and deletes 5 health from player	2. Player uses arrows to point which direction of pit to appear out of	<ul style="list-style-type: none">System sets user on outside of pit
Actor Action	System Response							
1. Player walks over pit on screen	<ul style="list-style-type: none">Animation is played showing the character and any followers falling into the pit, and deletes 5 health from player							
2. Player uses arrows to point which direction of pit to appear out of	<ul style="list-style-type: none">System sets user on outside of pit							
Variations:	If player does not have more than 5 health, the player is killed, triggering UC-04							
Exceptions:	N/A							

Developer Notes:	N/A
------------------	-----

Use Case ID:	UC - 11					
Use Case Name:	Discard Object From Bag					
Use Case Description:	Player wants to discard an item from their bag and can do so by selecting an item from their bag and hitting the delete button on the keyboard.					
Actors:	Player Character					
Pre-Condition:	The player must have 1 or more items in their bag.					
Post-Condition:	The player will have 1 less object in their bag and the amount of items in the bag is updated.					
Frequency of Use:	Every time a player would like to discard an item from their bag.					
Flow of Events:	<table><tr><th>Actor Action</th><th>System Response</th></tr><tr><td>1. Player selects item to be deleted and hits backspace</td><td><ul style="list-style-type: none">System deletes item from the bag and updates size of</td></tr></table>		Actor Action	System Response	1. Player selects item to be deleted and hits backspace	<ul style="list-style-type: none">System deletes item from the bag and updates size of
Actor Action	System Response					
1. Player selects item to be deleted and hits backspace	<ul style="list-style-type: none">System deletes item from the bag and updates size of					
Variations:	N/A					
Exceptions:	N/A					
Developer Notes:	N/A					

Activity Diagrams

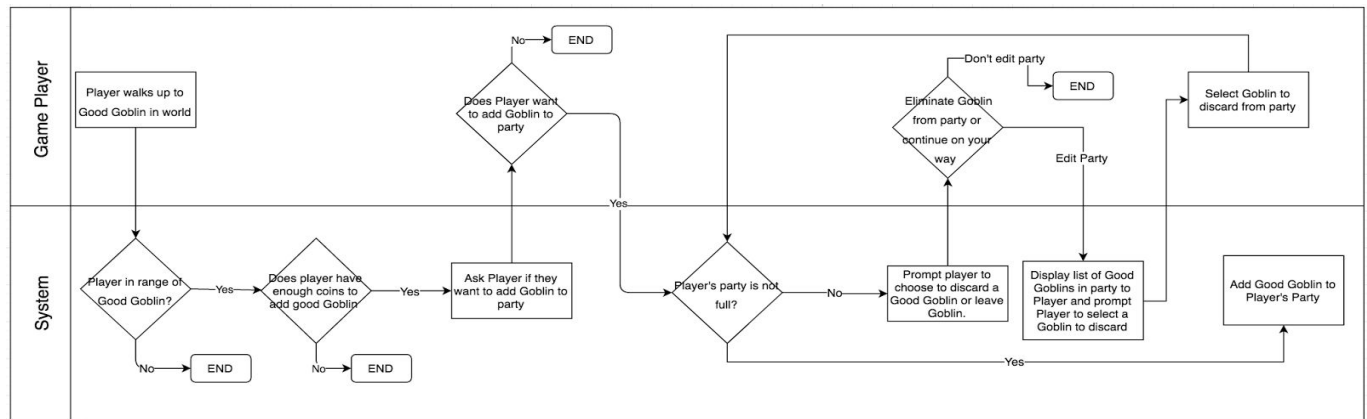
Group Member: John Murphy

Requirement(s) met: UR.3

Use case(s): UC-3 and UC-5

Description: A player will be close to a Goblin. The System will check if the player has enough coins to add the Goblin to their party, if the player can afford Goblin then they are prompted to add it to the party or continue on their way. If the player want to the add the Goblin and their

party is not full, then the Goblin will be added to party. Otherwise, the current party of Goblins will be displayed and the player will choose one to discard. Once they have chosen one to discard, the chosen Goblin will be deleted and the new Goblin will be added.

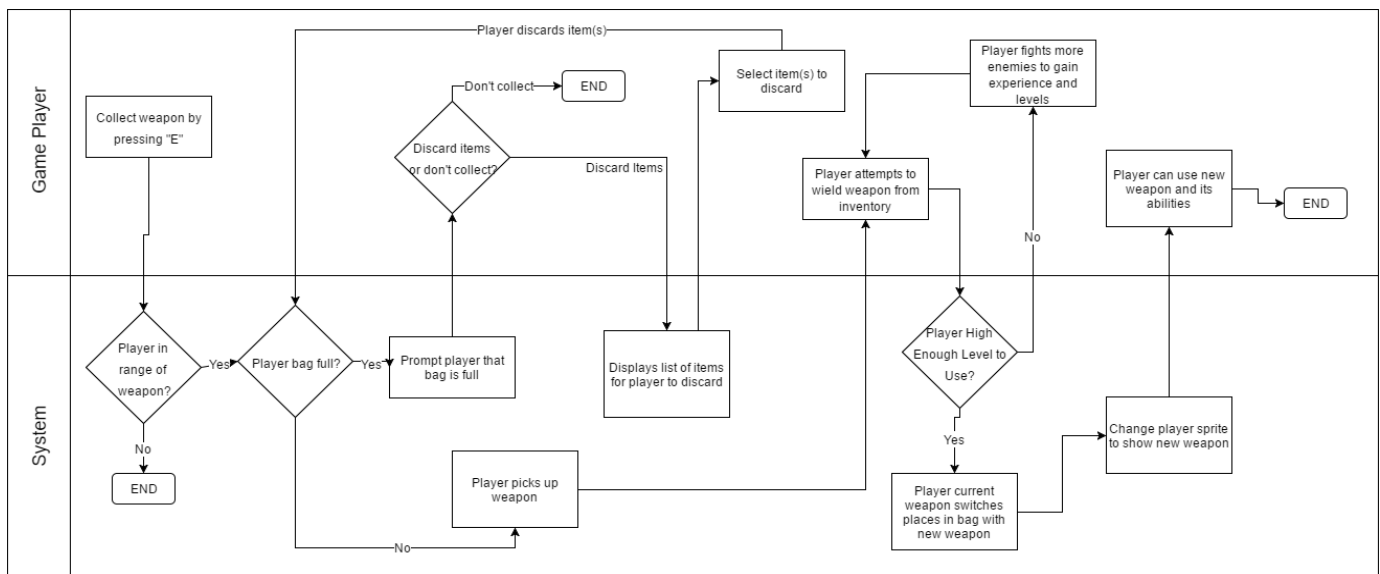


Group Member: Robert Kendl

Requirement(s) met: UR 4

Use cases: UC-8 and UC-9

Description: The player wants to pick up and use a weapon. The system will check if the player is in range to grab the weapon and check to make sure that the player's bag isn't full. Once the player picks up the weapon, the system will check if the player is a high enough level to use the weapon. If the player can use the weapon, they will switch that weapon with their current weapon.

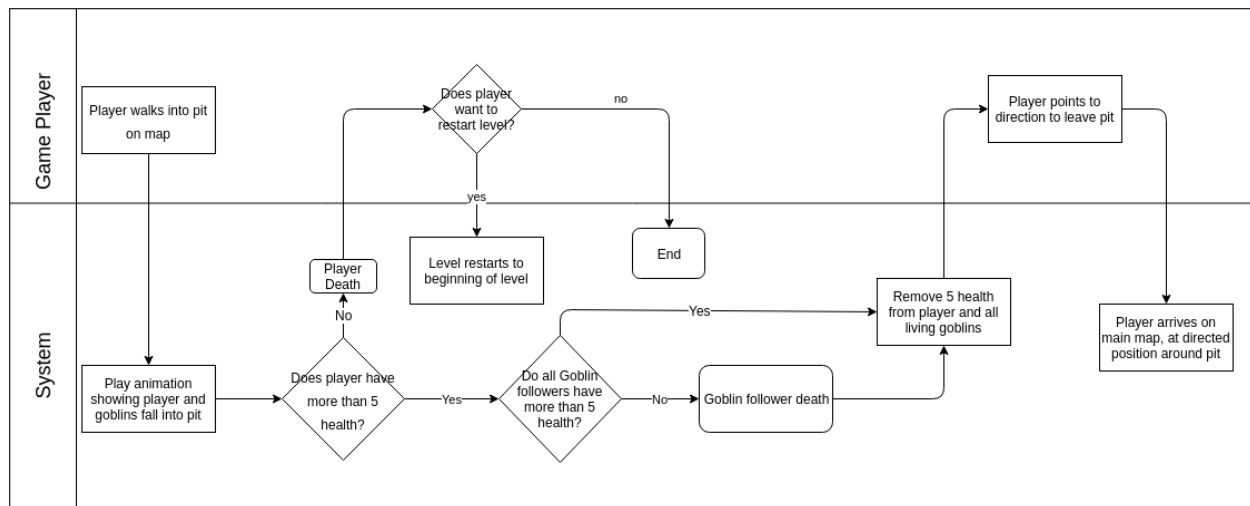


Group Member: Darren White

Requirement(s) met: UR-5

Use cases: UC-10 and UC-4

Description: The player is unable to avoid a pit and walks in. An animation will play show him and any goblin followers falling in. The system will check if the player has more than 5 health, if not it will result in player death, and the player will choose if he wants to end the game or retry this current level. If the player does have more than 5 health, the system will check if any goblin followers have less than 5 health, which will result in that goblin follower's death. The system will then subtract 5 health from the player and any surviving goblins. The player will then point to the direction of the pit he wants to escape from, and the system will place the player on this edge of the pit, to continue the game.



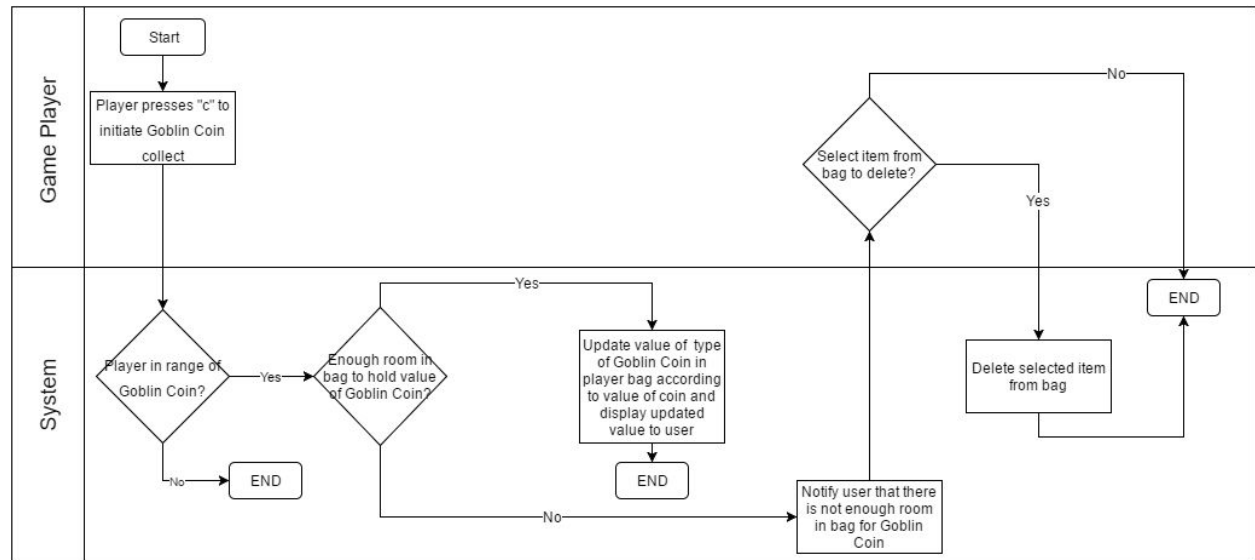
Group Member: Kristen Hanslik

Requirement(s) met: UR-6

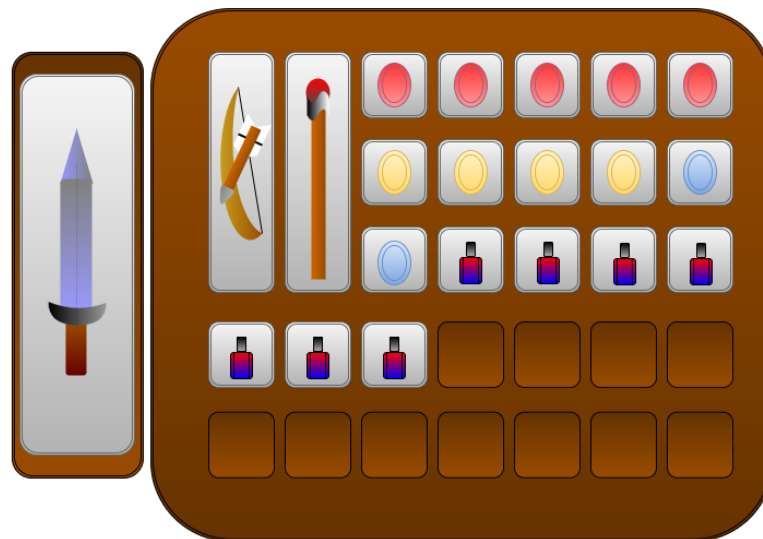
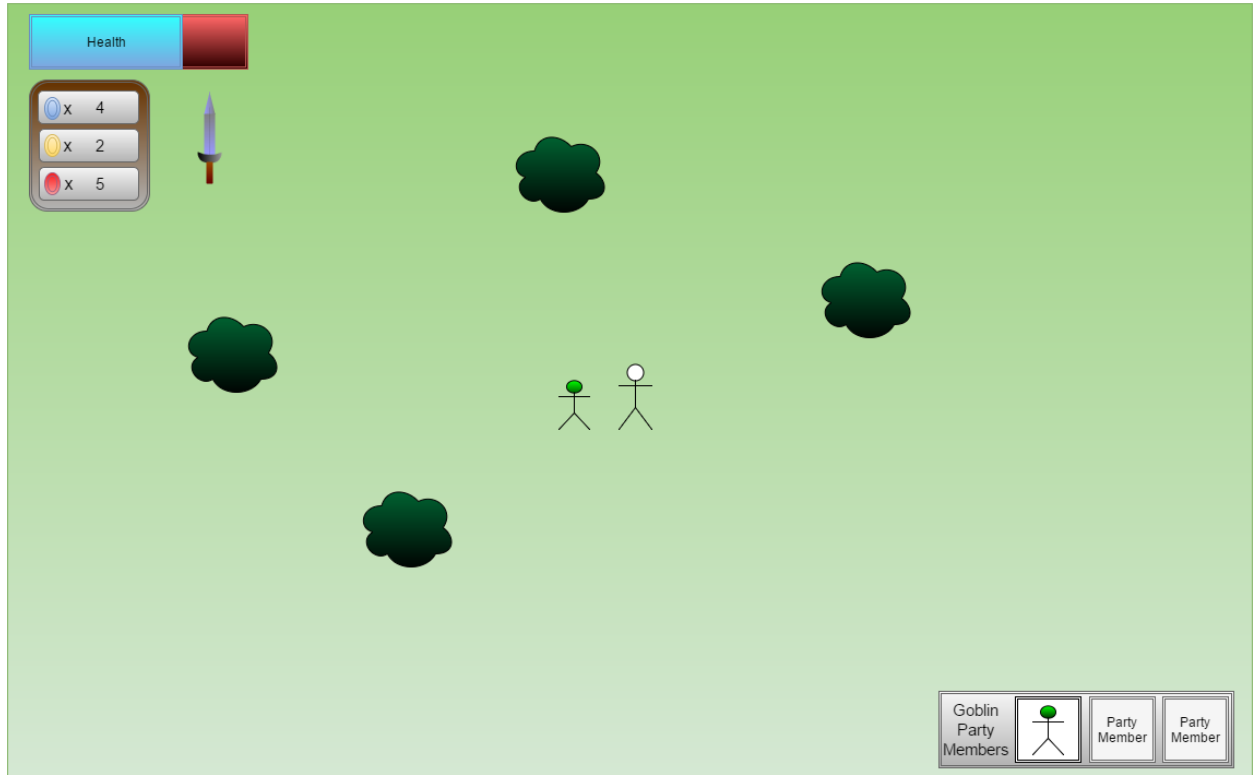
Use cases: UC-2 and UC-11

Description: This combines the two use cases Collect Goblin Coin from Ground and Discard Item From Bag. This diagram shows the acquisition of Goblin Coins by a Player and what happens if there is not room in the bag for the amount the Goblin Coin is worth. If these two use cases were not combined then the activity diagram would simply end after notifying the user that there is not enough room for the Goblin Coin. If an item is deleted in this Activity Diagram, the user can simply press "c" again after exiting this sequence, to restart the sequence with the updated state of the bag.

UseCase: Collect Goblin Coin, Discard Item from Bag



UI Mockup



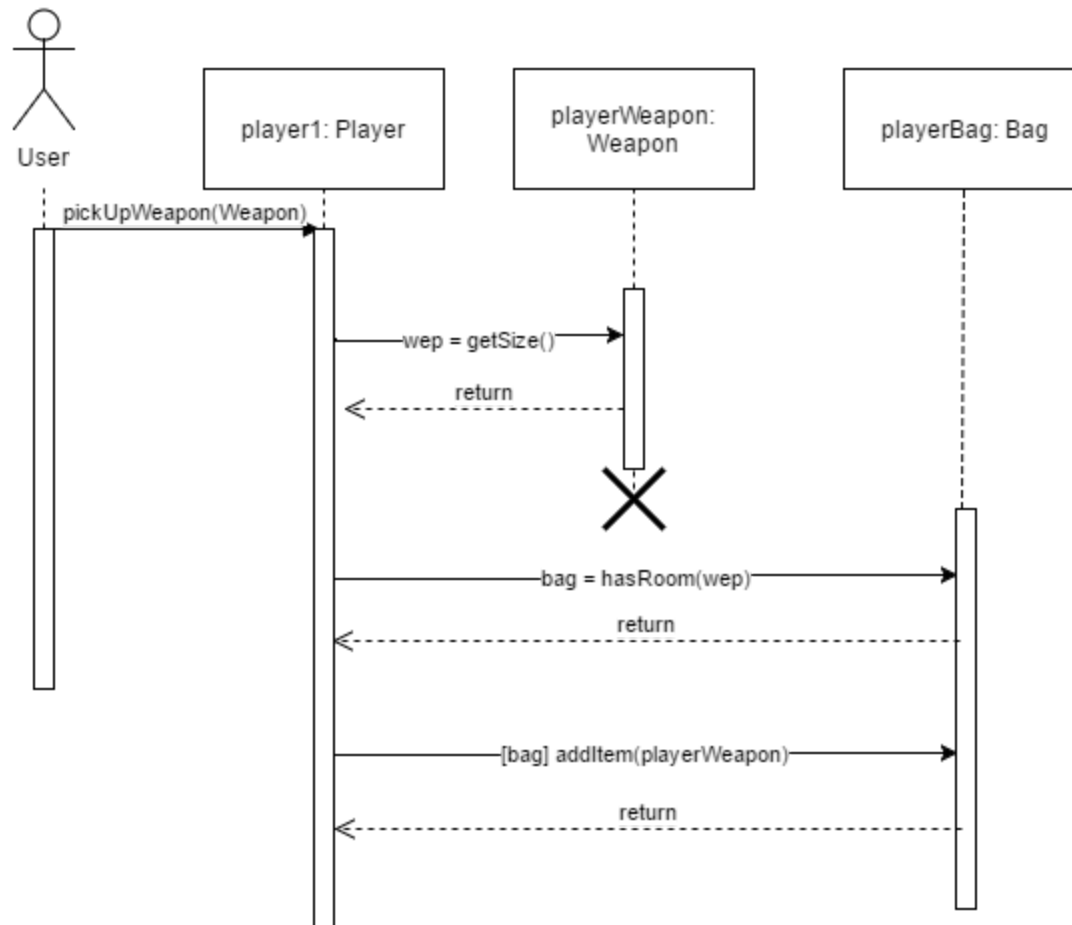
User Interactions

Group Member: Robert Kendl

Requirement(s) met: UR-4

Use case: UC-8

Description: The player wants to pick up a weapon. The program will check if there are open spaces in the bag. If there are, the player will be able to pick up the weapon.

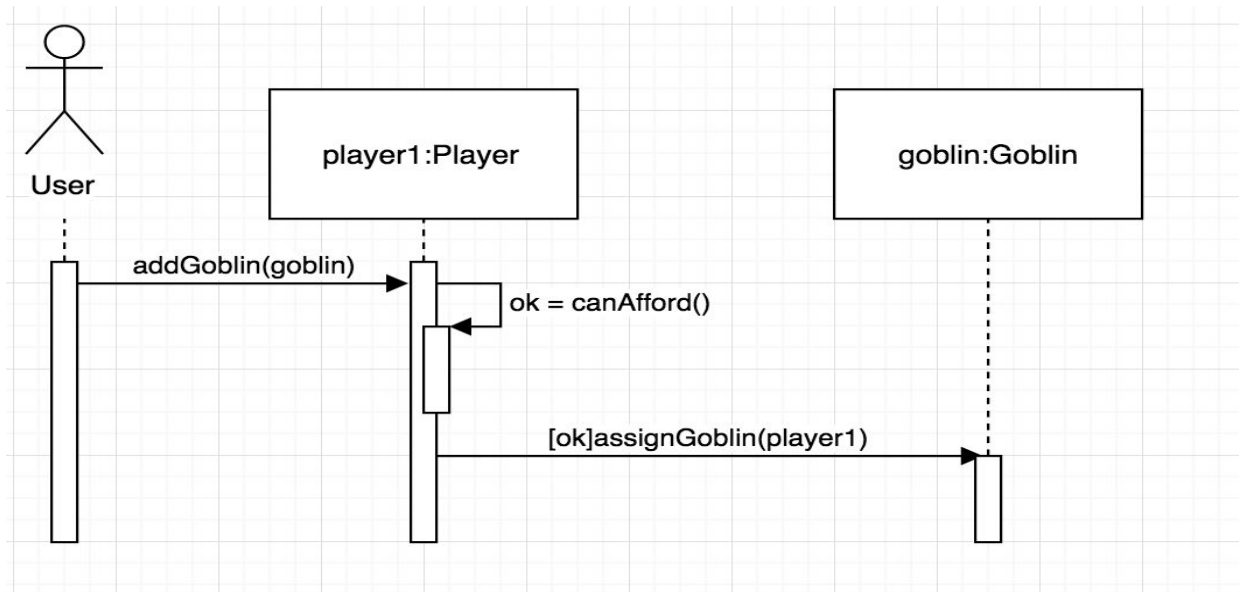


Group Member: John Murphy

Requirement(s) met: UR-3

Use case: UC-3

Description: If a user can afford to add goblin (instance of class Goblin), the Goblin's state will be changes so it is now following the player.

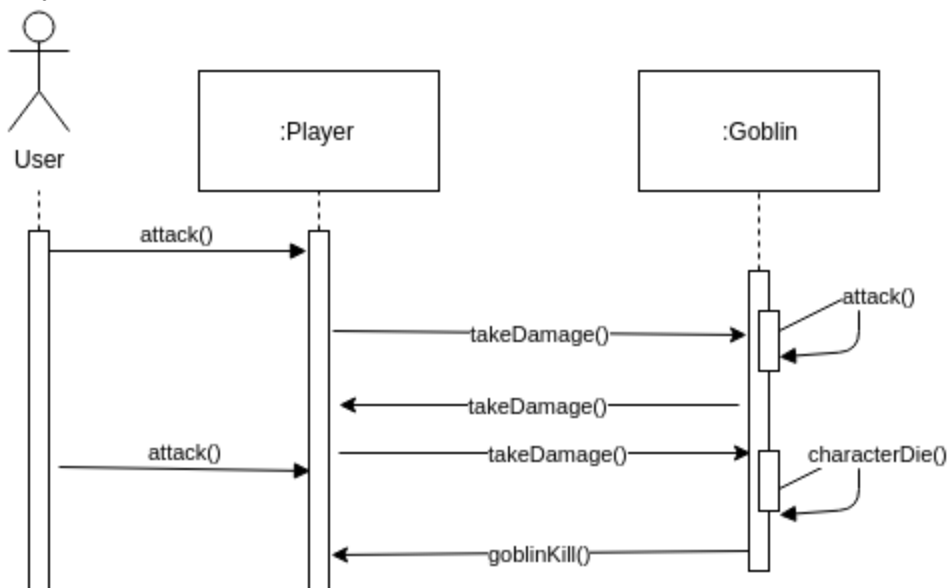


Group Member: Darren White

Requirement(s) met: UR-2

Use case: UC-1

Description: The user commands his player to attack the goblin, which then takes damage. The Goblin attacks the player, which causes the player to take damage. The player attacks back, killing the Goblin, which calls the goblinKill() method to keep track of total goblin kills by this player.

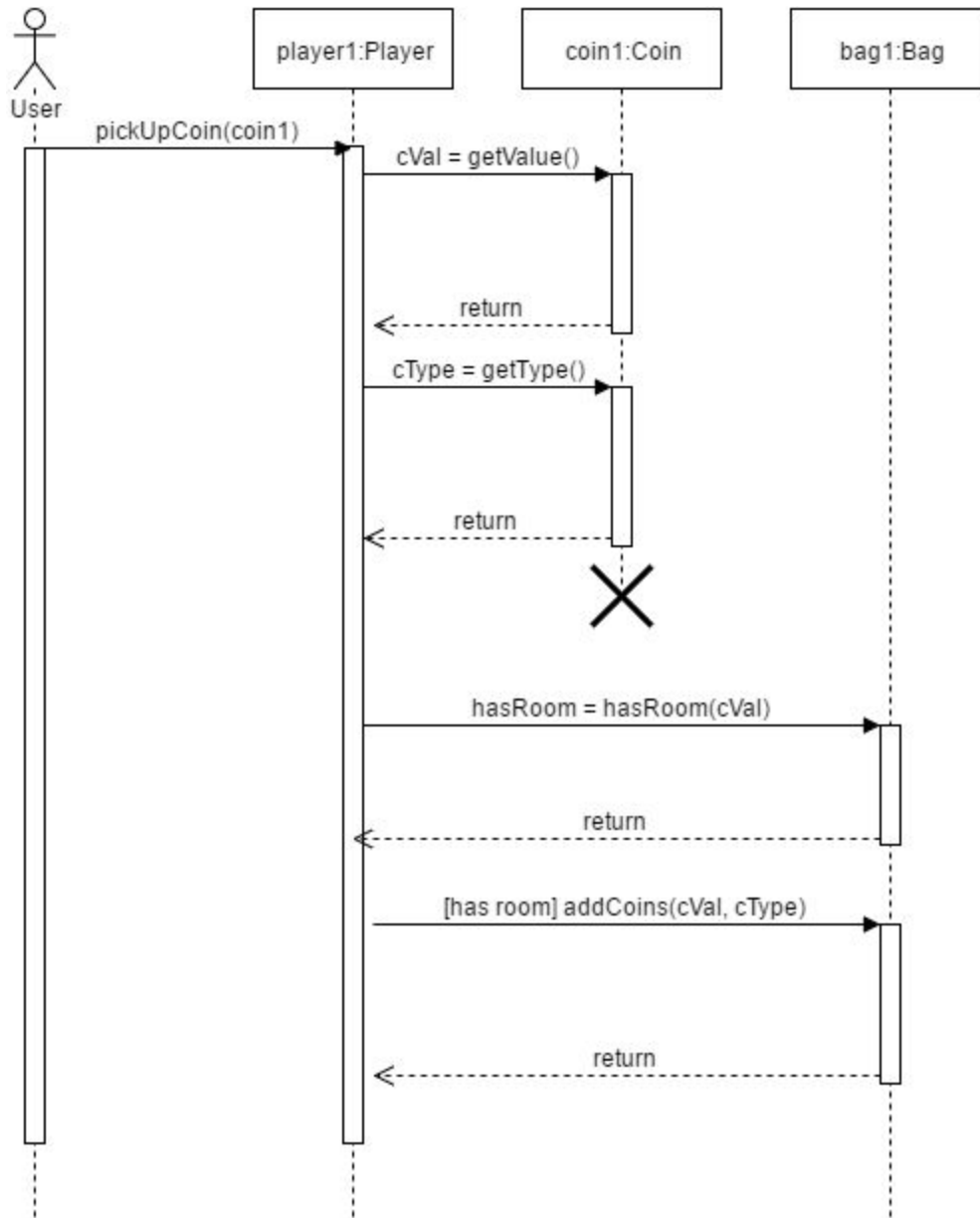


Group Member: Kristen Hanslik

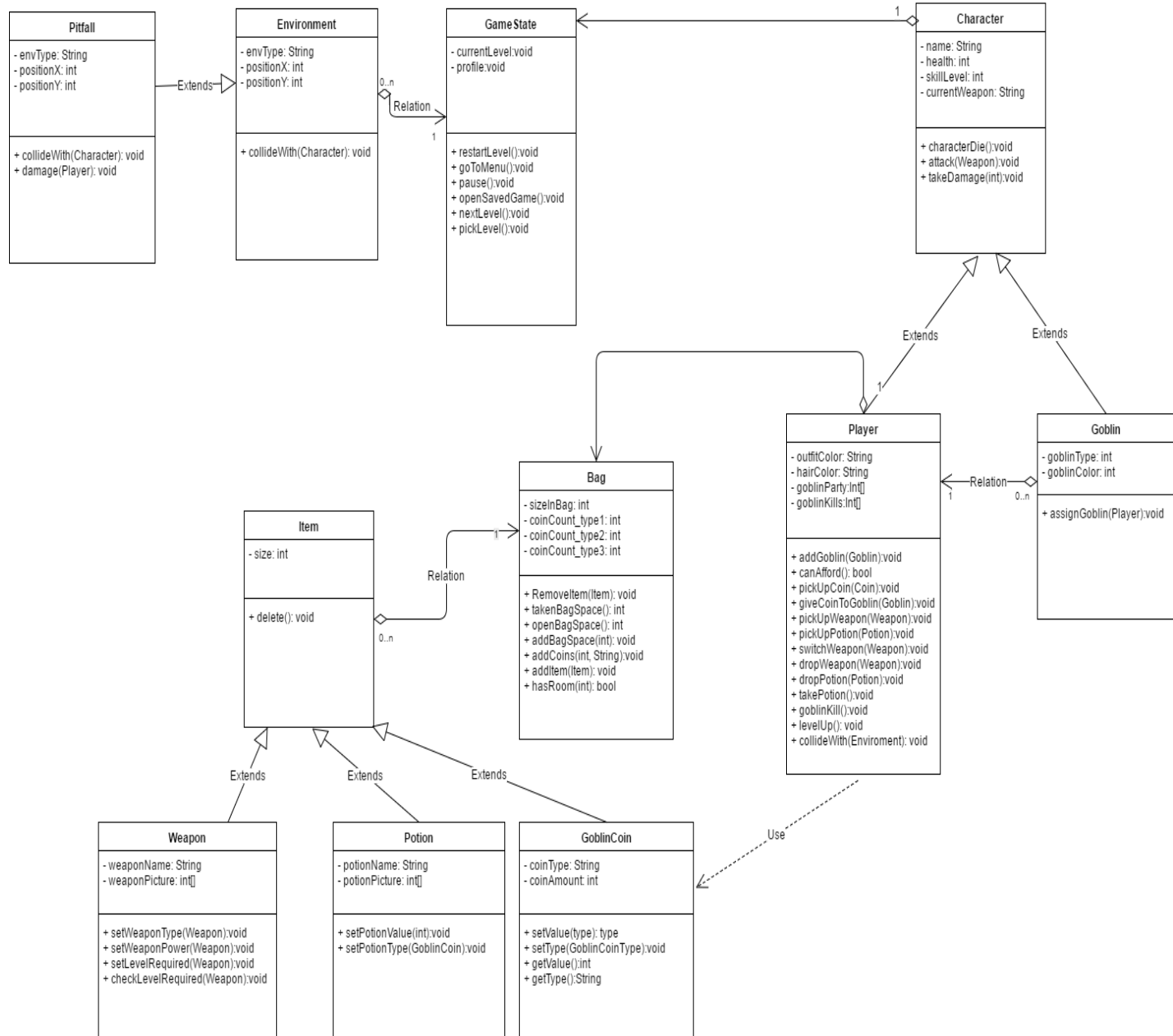
Requirement(s) met: UR-6

Use case: UC-2

Description: The user wants to pick up a coin that they see on the screen. The user must be in a set range of the coin and have room enough in their bag to hold the value of the coin. To initiate the collection of the coin, they must press “c” while in range. The amount of that type of coin in the player’s bag will increase by the value of the coin if the coin can be added.



Class Diagram



State Diagram

