

## User Stories

*Nouns highlighted in orange.*

*Actions highlighted in blue.*

As two Players,  
So we can play a personalised game of Battle,  
We want to Start a fight by entering our Names and seeing them

As Player 1,  
So I can see how close I am to winning  
I want to see Player 2's Hit Points

As Player 1,  
So I can win a game of Battle,  
I want to attack Player 2, and I want to get a confirmation

As Player 1,  
So I can start to win a game of Battle,  
I want my attack to reduce Player 2's HP by 10

As two Players,  
So we can continue our game of Battle,  
We want to switch turns

As Player 1,  
So I can see how close I am to losing,  
I want to see my own hit points

As Player 1,  
So I can lose a game of Battle,  
I want Player 2 to attack me, and I want to get a confirmation

As Player 1,  
So I can start to lose a game of Battle,  
I want Player 2's attack to reduce my HP by 10

As a Player,  
So I can Lose a game of Battle,  
I want to see a 'Lose' message if I reach 0HP first

## Investigating the user stories.

Take the nouns and actions highlighted in the user stories and fill in the three tables below to help us uncover some more details about them and see how they relate to each other.

Nouns	Property or owner of property?
fight	owner of property
Players	Owner of property
Player 1 Hit Points	Property (players)
Player 2 Hit Points	property(players)
Lose message	property

Actions	Owned by?
enter_names_to_start	fight
player_2_score	Players?
attack_player_2	fight
attack_confirmed?	fight
reduce_player_2_score	Players
switch_turns	fight
player_1_score	players
player_2_attack	fight
reduce_player_1_score	players
lost	fight

Actions	Property it reads or changes
enter_names_to_start	@names(in players)
player_2_score	@score(in players)
attack_player_2	(attacked?)(if general attack method read turn/attacked?)
attack_confirmed?	Reads attack_player_2
reduce_player_2_score	@score
switch_turns	@turn(in fight)
player_1_score	@score(in players)
player_2_attack	(attacked?)(if general attack method read turn/attacked?)
reduce_player_1_score	@score
lost	@score(reached 0)

### Class diagrams for these user stories:

*Summarise what we have learned from the three tables in the previous section into class diagrams.*

<b>Class (Owner)</b>	Players
<b>Properties (instance variables)</b>	@name, @score,
<b>Actions (methods)</b>	Player_2_score, player_1_score, reduce_player_1_score, reduce_player_2_score

<b>Class (Owner)</b>	Fight
<b>Properties (instance variables)</b>	@turn, @player_1, @player_2
<b>Actions (methods)</b>	Enter_names_to_start, attack_player_2, attack_confirmed?, switch_turns, player_2_attack, lost?

<b>Class (Owner)</b>	Battle(if multiple rounds)
<b>Properties (instance variables)</b>	@fight
<b>Actions (methods)</b>	begin_fight(fight.start), main_fight(attack, switch_turns), lost?