

**Name:** User Attack

**Scope:** user goal

**Primary Actor:** User/Player

**Stakeholders and Interests:**

- Player: Wants to enter into a fight with an enemy quickly, and have a smooth battle that is a good amount of difficulty.

**Preconditions:**

- Player is attacked or attacks an enemy.

**Postconditions:**

- Battle ends with the player gaining experience or being defeated.
- Player is returned to the map

**Main Success Scenario:**

- 1) The user attacks or is attacked by an enemy.
- 2) The system starts the battle, loading the menu and screens.
- 3) The system displays the enemies and their relevant information, and displays the battle menu consisting of Fight, Flee, Abilities.
- 4) The system then loads the battle functions.

**Extensions:**

**Special Requirements:**

- Screen display is colorful
- Menu is easy to read
- Enemy designs are decent enough to tell what they are
- Enemy info displayed legibly

**Technology and Tech Variations List:**

- Keyboard to control player, select menu items

**Frequency of Occurrence:**

- Fairly frequent, though the character can run from a battle

**Open Issues:**

**Name:** User Battle

**Scope:** user goal

**Primary Actor:** Player

**Stakeholders and Interests:**

-Player: Would like a quick battle, easy to navigate menus that display simple information effectively.

**Preconditions:**

-Battle is initiated, system properly loads the system

**Postconditions:**

-Player exits the battle in victory or defeat, depending on which one a screen is shown.

**Main Success Scenario:**

- 1) The user selects an option from the battle menu.
- 2) If Fight, the system asks the user which enemy to attack.
- 3) The user selects the enemy they wish to attack.
- 4) The system then calculates damage based on the attack of the user minus the defense of the enemy.
- 5) The system deducts from the enemy health accordingly, and updates the screen.
- 6) This ends the user's turn.
- 7) After the user's turn, the enemy system selects the user to be attacked, and the system calculates damage.
- 8) The system then updates the screen.
- 9) This repeats until either the user flees, the enemy is defeated, or the user is defeated.
- 10) The system then exits the battle, and returns the player to the map screen.

**Extensions:**

- a) If Flee is chosen, the system exits the battle and returns the user to the map.
- b) If Abilities, the system updates the menu displayed and shows the user's abilities on the menu.
  - 1) The user then selects which ability they wish to use.
  - 2) If applicable, the system asks for a target to attack, and the user selects which enemy to attack with that ability.
  - 3) The system performs the ability and if necessary, changes necessary data, then updates the screen.
  - 4) The user's turn is now over.
- c) If the user loses the battle, the system exits the battle and displays a game over screen

**Special Requirements:**

-Screen is readable  
-Battle is fast, no hiccups in graphics

**Technology and Tech Variations List:**

-Keyboard for input of commands by player

**Frequency of Occurrence:**

-Frequent, as the player will battle often for experience.

## **Open Issues:**

**Name:** Map

**Scope:** Functionality

**Primary Actor:** Player

**Stakeholders and Interests:**

-Player: Wants a colorful map with functionality, easy to recognize Baylor buildings

**Preconditions:**

-Game is loaded.

**Postconditions:**

-Map is available for player to traverse

**Main Success Scenario:**

- 1) The user uses the directional arrows to move their character across the map.
- 2) The user can encounter enemies randomly, loading the battle.
- 3) The user can also enter buildings.
- 4) If the user enters a building, the system updates the map to display the inside of the building.

**Extensions:**

- a) Player wants to quit game. They load the menu and quit through that or close the program directly.
- b) Player wants to save. Player opens the menu and selects save.
- c) Player wants to enter something that cannot be entered. The system tells player that the action is not allowed

**Special Requirements:**

**Technology and Tech Variations List:**

**Frequency of Occurrence:**

-Very frequent.

**Open Issues:**