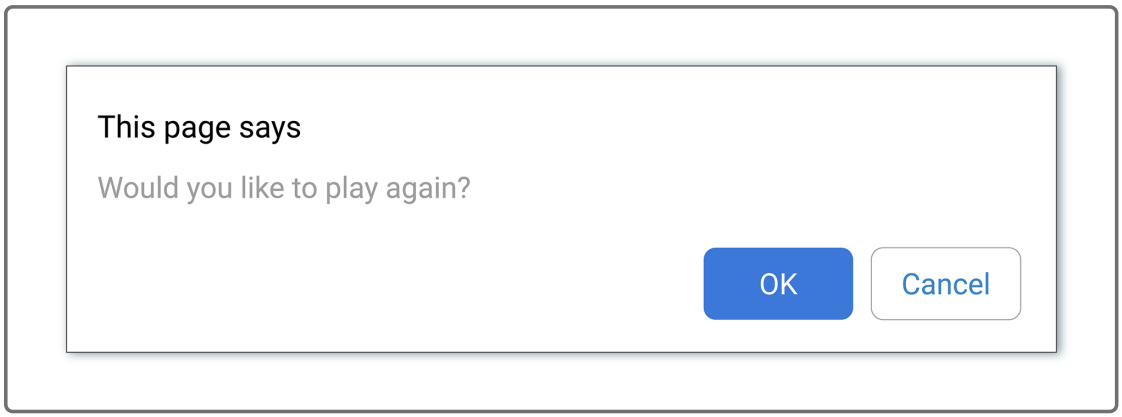
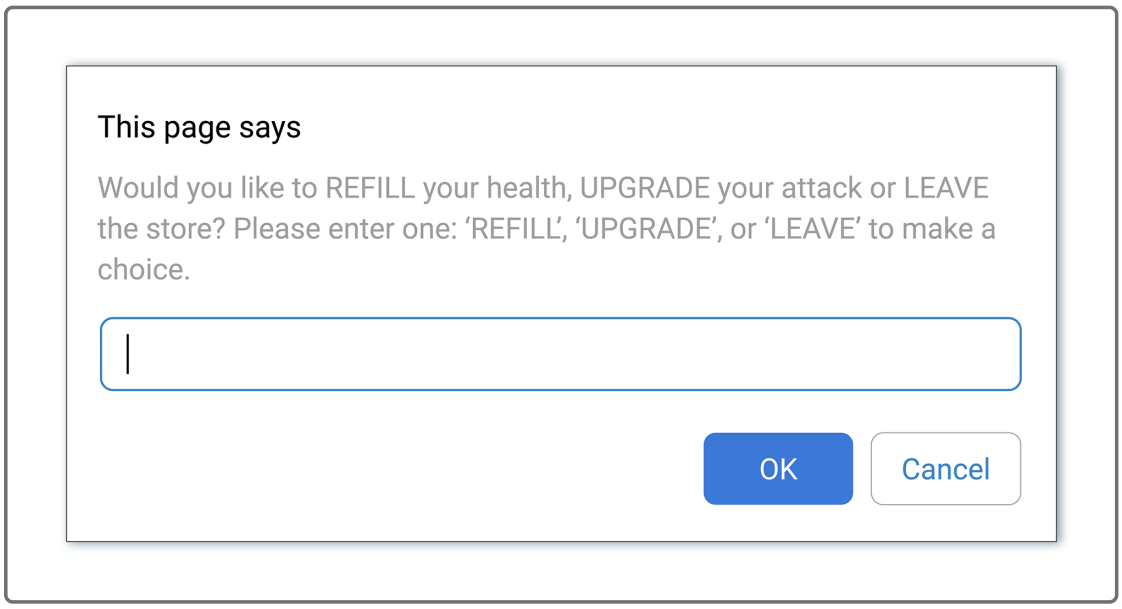
3.3.2 Preview

From the player's perspective, two things are being added to the game.

First, at the end of the game, we'll display the player's score in an alert and then ask the player if they want to play again:



Second, after the player defeats or skips an enemy robot, we'll ask if they want to visit the shop. If so, they'll be presented with a few options:



Take a moment to pseudocode how you might approach each of these features. Remember, pseudocode isn't real code; you simply write the sequence in which you feel things need to happen.

Here's our take on pseudocoding the "play again" and "shop" features:

- Wrap the game logic in a `startGame()` function
- When the player is defeated or there are no more enemies, call an `endGame()` function that:
    - Alerts the player's total stats
    - Asks the player if they want to play again
    - If yes, call `startGame()` to restart the game
- After the player skips or defeats an enemy (and there are still more robots to fight):
    - Ask the player if they want to "shop"
    - If no, continue as normal
    - If yes, call the `shop()` function

- In the `shop()` function, ask player if they want to "refill" health, "upgrade" attack, or "leave" the shop
- If refill, subtract money points from player and increase health
- If upgrade, subtract money points from player and increase attack power
- If leave, alert goodbye and exit the function
- If any other invalid option, call `shop()` again

Notice that we actually outlined three new functions: the `startGame()` function to start/restart the game, the `endGame()` function to handle endgame logic, and the `shop()` function to house the shop. If your instinct was to use fewer or more functions, that's perfectly fine! We chose these three to future-proof cases where we might need to run the same logic again.

This is how we'll proceed:

1. Add the `startGame()` function to define (and reset) the state of the game.
2. Add the `endGame()` function to display stats and prompt the user to play again.
3. Finalize the MVP and switch branches.
4. Add the `shop()` function for all shop-related logic.
5. Save our progress by using Git.