**Use Cases**

**for**

**Build-The-Bot**

**Version 1.0**

**Prepared by**

**Julia Waggoner, Karen Hernandez, Miranda Castle, Truc Le**

**DePauw University**

**I. Use Case: Go**

1. Player enters “go ‘direction’”, with the directions being either ‘front’ to go forward, ‘back’ to go backward, ‘left’ to go to the left, and ‘right’ to go to the right.

2. Game executes the command to alter the current location and player’s placement accordingly.

3. Game notifies player about the command’s result (the command’s effect on the game environment).

4. Player continues game.

**Variation #1**

1.1 Player enters movement command that is not appropriate to player’s current location (such as going through a wall or a locked door).

1.2 Game notifies player that the player cannot go that way.

1.3 Player enters another command.

**Variation #2**

2.1 Player enters “go” without a specified direction, or player enters a direction without entering “go” first.

2.2 Game notifies player that the command is invalid.

2.3 Player enters another command.

**II. Use Case: Look**

1. Player enters “look” to learn about non-hidden items that are in the current location.

2. Game prints the description of the current location, as well as the names and descriptions of the items in the environment.

3. Player continues game.

**Variation #1**

1.1 Player enters “look” but there are no items in the current location.

1.2 Game prints the description of the current location but no items are printed.

1.3 Player continues game.

**III. Use Case: Examine Item**

1. Player enters “examine ‘item’” to examine the specified item.

2. Game prints the name and the description of the item.

3. Player continues game

**Variation #1**

1.1 Player enters “examine ‘item’” to an item that does not exist in current location.

1.2 Game notifies player that the specified item does not exist.

1.3. Player may re-enter a command or continue game.

**Variation #2**

2.1 Player enters “examine” without specifying an item to examine.

2.2 Game prints a message stating that player entered an unknown command.

2.3 Player may re-enter a command or continue game.

**Use Case IV: Take Item from Current Location and Store it in Inventory**

1. Player enters “take ‘item’” to take the specified item from the current location.

2. Game saves that item into player’s inventory, which contains all previously taken items.

3. Player enters “inventory” to check player’s inventory.

4. Game prints out the names and descriptions of all items that are in player’s inventory.

5. Player continues game.

**Variation #1**

1.1 Player chooses not to take the items that are in the current location.

1.2 The items stay in the location and nothing gets added to player’s inventory.

1.3 Player continues game.

**Variation #2**

2.1 Player enters “take ‘item’” to an item that is not possible to take.

2.2 Game prints a message stating that the item cannot be taken.

2.3 Player continues game.

**Variation #3**

3.1 Player enters “take ‘item’” to an item that is not in the current location or does not exist.

3.2 Game notifies the player that the item cannot be found.

3.3 Player continues game.

**Variation #4**

4.1 Player enters “inventory” but does not take any items first.

4.2 Game prints “Inventory:” with an empty list.

4.3 Player continues game.

**V. Use Case: Take Item from Container**

1. Player enters “take ‘item’ from ‘container’” to take the specified item from the specified container.

2. Game takes the specified item from the specified container and puts it in player’s inventory.

3. Player continues game.

**Variation #1**

1.1 Player enters “take ‘item’” that is within a container without specifying “from ‘container.’”

1.2 Game notifies player that the item is not found.

1.3 Player may re-enter a command .

**Variation #2**

2.1 Player enters “take ‘item’ from ‘container’” of an item that does not exist in the specified container.

2.2 Game notifies player that the item is not found.

2.3 Player may re-enter a command.

**Variation #3**

3.1 Player enters “take ‘item’ from ‘container’” of a container that does not exist in the current location.

3.2 Game notifies player that the container is not found.

3.3 Player may re-enter a command.

**VI. Use Case: Put Item in Current Location**

1. Player enters “put ‘item,’” to take the specified item from player’s inventory and place it into the current location.

2. Game takes the specified item from player’s inventory and puts it in the current location.

3. Game notifies player if the given command leads to an altered environment or event.

4. Player continues game.

**Variation #1**

1.1 Player enters “put” followed by either a non-existent item or an item that is not in player's inventory.

2.2 Game notifies the player that the item is not found.

2.3 Player re-enters a command.

**Variation #2**

2.1 Player enters “put” without specifying what item to put.

2.2 Game prints an error message stating that it is an invalid command.

2.3 Player re-enters a command.

**VII. Use Case: Put Item in a Container**

1. Player enters “put ‘item’ in ‘container,’” to take the specified item from player’s inventory and put it in the specified container.

2. Game takes the specified item from player’s inventory and puts it in the specified container.

3. Game notifies player if the given command leads to an altered environment or event.

4. Player continues game.

**Variation #1**

1.1 Player enters a non-existent item or an item that is not in player's inventory.

1.2 Game notifies the player that the item is not found.

1.3 Player re-enters a command.

**Variation #2**

2.1 Player enters “put ‘item’ in ‘container’” to a non-existent container or a container that cannot hold anything.

2.2 Game notifies the player that the container (receiver) is not found.

2.3 Player re-enters a command.

**VIII. Give Item to a Character**

1. Player enters “give ‘item’ to ‘character,’” to take the specified item from player's inventory and give it to the specified character.

2. Game takes the specified item from player's inventory and gives to the specified character.

3. Game notifies player if the given command leads to an altered environment or event.

4. Player continues game.

**Variation #1**

1.1 Player enters an non-existent item or an item not in player's inventory.

1.2 Game notifies the player that the specified item is not found.

1.3 Player re-enters a command.

**Variation #2**

2.1 Player enters a character that does not exist or is unable to receive the item.

2.2 Game notifies the player that the character (receiver) is not found.

2.3 Player re-enters a command.

**Variation #3**

3.1 Player enters “give ‘item’” without specifying a character to give the item to.

3.2 Game notifies the player that player entered an unknown command.

3.3 Player re-enters a command.

**IX. Use Case: Save, Load, and Quit the Game**

1. Player enters “save.”

2. Game saves the current location, inventory, and progress, and prints a message stating that the game is saved along with a description of the time and date that the game was saved.

3. Player enters “quit.”

4. Game prints a message stating the game is quitting, stops the game, and player may no longer enter commands.

5. Player runs the game again.

6. Player enters “load.”

7. Game prints a message stating that the game loaded from the time and date that the game was saved last. The game then sets the environment, inventory, and progress from the last saved moment.

8. Player continues game.

**Variation #1**

1.1 Player enters “quit” without entering “save” first.

1.2 Game stops the game.

1.3 Player re-runs the game.

1.4 The game starts from the beginning of the game.

1.5 Player continues game.

**Variation #2**

2.1 Player enters “quit” without entering “save” first.

2.2 Game stops the game.

2.3 Player re-runs the game.

2.4 Player enters “load.”

2.5 Game starts from the last saved state if it exists, otherwise game starts from the beginning of the game.

2.6 Player continues game.

**Variation #3**

3.1 Players enters “save,” and then continues the game.

3.2 Player later enters “quit,” reruns, and enters “load.”

3.3 Game will load the player’s progress up to the moment the player entered “save.” All progress after the the player entered “save” will be lost.

3.4 Player continues game.

**Variation #4**

4.1 Player enters “quit” without re-running the game.

4.2 Game notifies the player that the game is quitting.

4.3 Player may no longer enter commands. Game does not continue.

**X. Use Case: Help**

1. Player enters “help.”

2. Game prints a list of valid commands and a description of each.

3. Player continues game.