Test Plan

A. Identification and classification:

Test Case 1

System: Save a game

Make sure the game saves in the first place

Severity: 1

B. Instructions:

1. Login and play for a few minutes

2. Save your game in an empty slot

3. Exit and try to load

C. Expected result:

1. Success message should appear, save state of your game should be consistent with what you had before

A. Identification and classification:

Test Case 2

System: Overwriting saves

Save a game, and then save a new one in the same place

Severity: 2

B. Instructions:

1. Login and play for a few minutes

2. Save your game in an empty slot

3. Start another new game and save over the old slot

4. Load the game that you saved over and make sure saved state is consistent

C. Expected result:

1. Success message should appear, saved state should be the most recently saved slot.

A. Identification and classification:

Test Case 3

System: Handle multiple saves

Save a game in all ten slots and make sure they all match up

Severity: 2

B. Instructions:

1. Login and play the game for a few minutes, and then save.

2. Start a new game and continue until you have all 10 spots taken up

C. Expected result:

1. Success message should appear for each game save

2. Upon loading, save state should be consistent with the corresponding game

A. Identification and classification:
Test Case 4
System: Saving
Game Saving Performance
Severity: 2
B. Instructions:
Sign in and play a game and save it.
C. Expected result:
Game state should save in at most 2 seconds.

A. Identification and classification:
Test Case 5
System: Loading
Try to load an empty spot
Severity: 2
B. Instructions:
1. Load an empty save spot
C. Expected result:
1. Error should be thrown

A. Identification and classification:
Test Case 6
System: Loading
Trying to load locally changed save file
Severity: 2
B. Instructions:
Save a game, modify the file locally and reload the game.
C. Expected result:
1. It should throw an exception

A. Identification and classification:
Test Case 7
System: Loading
Map Loading Performance
Severity: 2
B. Instructions:
Sign in and load a map to start playing.
C. Expected result:
Map should load in at most 2 seconds.

A. Identification and classification:

Test Case 8

System: Registration

Create an account with unique username

Severity: 1

B. Instructions:

1. Create a valid username and password

C. Expected result:

User successfully creates account and is able to see list of maps on server.

A. Identification and classification:

Test Case 9

System: Registration

Create an account with invalid username

Severity: 2

B. Instructions:

1. Type a username already being used and a password

C. Expected result:

System lets user know username already exists and cannot proceed.

A. Identification and classification:

Test Case 10

System: Registration

Create an account with invalid or null password in second password field

Severity: 2

B. Instructions:

1. Type mismatching passwords in create account window

C. Expected result:

System lets user know passwords do not match and cannot proceed.

A. Identification and classification:

Test Case 11

System: Registration

Create an account with null username or password

Severity: 2

B. Instructions:

1. Leave username or password fields blanks

C. Expected result:

System lets user know no username or password typed and cannot proceed.

A. Identification and classification:
Test Case 12
System: Login
Successful login
Severity: 1
B. Instructions:
1. Start the game and enter a correct username and password
C. Expected result:
System lets user login and brings the user to the next screen

A. Identification and classification:
Test Case 13
System: Login
Unsuccessful login with wrong username
Severity: 2
B. Instructions:
1. Start the game and enter a correct username and an incorrect password
C. Expected result:
System informs user that the login was unsuccessful and doesn't go to the next screen

A. Identification and classification:
Test Case 14
System: Login
Unsuccessful login with wrong password
Severity: 2
B. Instructions:
1. Start the game and enter an incorrect username and a correct password
C. Expected result:
System informs user that the login was unsuccessful and doesn't go to the next screen

A. Identification and classification:
Test Case 15
System: Login
Unsuccessful login with null username and correct password
Severity: 2
B. Instructions:
1. Start the game and enter a correct password and leave username blank
C. Expected result:
System informs user that the login was unsuccessful and doesn't go to the next screen

A. Identification and classification:
Test Case 16
System: Login
Unsuccessful login with a username and a null password
Severity: 2
B. Instructions:
1. Start the game and enter a correct username and leave password blank
C. Expected result:
System informs user that the login was unsuccessful and doesn't go to the next screen

A. Identification and classification:
Test Case 17
System: Login
Unsuccessful login with a null username and a null password
Severity: 2
B. Instructions:
1. Start the game and leave both fields for username and password blank
C. Expected result:
System informs user that the login was unsuccessful and doesn't go to the next screen

A. Identification and classification:
Test Case 18
System: Login
Sql injection
Severity:2
B. Instructions:
1. Start the game enter a username and enter name' Or 'a'='a'
C. Expected result:
System informs user that the login was unsuccessful and doesn't go to the next screen

A. Identification and classification:
Test Case 19
System: Maps
Correctly download a map
Severity:1
B. Instructions:
1. Start the game enter a correct username and password then select a valid map to download
C. Expected result:
System successfully downloads the map and starts the game

A. Identification and classification:

Test Case 20
System: Maps
Download a null map
Severity:2
B. Instructions:
1. Start the game enter a correct username and password. then try to download a map without selecting one
C. Expected result:
System informs user no map is selected and fails to start the game

A. Identification and classification:
Test Case 21
System: Maps
Equivalence test for downloading maps
Severity: 2
B. Instructions:
1. Start the game enter a correct username and password. then try to download the last map on the list
C. Expected result:
System successfully downloads the map and starts the game

A. Identification and classification:
Test Case 22
System: Maps
Equivalence test for downloading maps
Severity: 2
B. Instructions:
1. Start the game enter a correct username and password. then try to download the first map on the list
C. Expected result:
System successfully downloads the map and starts the game

A. Identification and classification:
Test Case 23
System: Upload Map
Upload a map
Severity: 1
B. Instructions:
1. Select upload map from the options bar
2. Select a map to upload
C. Expected result:

1. The map is uploaded to the server and a dialog upon successful completion is displayed.

A. Identification and classification:
Test Case 24
System: Upload Map
Map list is updated
Severity: 2
B. Instructions:
1. Upload a map.
2. Select new game from the options bar.
C. Expected result:
1. The list of maps should be updated with the newly uploaded map.
2. The new map should be shown in the list of maps.

A. Identification and classification:
Test Case 25
System: Upload Map
Map validation (map is a valid file)
Severity: 1
B. Instructions:
1. Select upload map from option bar.
2. Select a file.
C. Expected result:
1. The map will upload correctly

A. Identification and classification:
Test Case 26
System: Upload Map
Map validation (not valid file type)
Severity: 1
B. Instructions:
1. Select upload map from the option bar.
2. Select a file to upload.
C. Expected result:
1. Dialog that says the file is not a valid map file type.

A. Identification and classification:
Test Case 27
System: Upload Map
Select no file
Severity: 2

B. Instructions:
1. Select upload map from the option bar.
2. Don't select a map to upload and hit upload
C. Expected result:
1. Dialog that says the file is not a valid map file.

A. Identification and classification:
Test Case 28
System: Containers
Open a container
Severity: 1
B. Instructions:
1. Be near a container
2. Type the command open container
C. Expected result:
 The container should be opened and the program should indicate it

A. Identification and classification:
Test Case 29
System: Containers
Try to open a container that doesn't exist
Severity: 2
B. Instructions:
1. Be nowhere near any containers
2. Type the command open container
C. Expected result:
 The program should indicate that there are no containers nearby and fail to open it

A. Identification and classification:
Test Case 30
System: Containers
Put item in container
Severity: 1
B. Instructions:
1. Pick up an item
2. Be near a container
3. Type the command open container
4. Type the command put item in container
C. Expected result:
 The item should go into the container and out of the users hands and the program should indicate it

A. Identification and classification:

Test Case 31

System: Containers

Try to put nothing into the container

Severity: 2

B. Instructions:

1. Keep inventory empty

2. Be near a container

3. Type the command open container

4. Type the command put item in container

C. Expected result:

 The program should indicate that there is no item to put in the container and should fail to put anything into the container

A. Identification and classification:

Test Case 32

System: Containers

Try to put an item without being near a container

Severity: 2

B. Instructions:

1. Pick up an item

2. Be nowhere near a container

3. Type the command put item in container

C. Expected result:

 The program should indicate that there is no container to put the item in and it should fail to put the item in the nonexistent container

A. Identification and classification:

Test Case 33

System: Containers

Try to put an item while being near a container but without opening the container

Severity: 3

B. Instructions:

1. Pick up an item

2. Be near a container

3. Type the command put item in container

C. Expected result:

 The program should indicate that the container is not open or that there is no container and fail to put the item in the container

A. Identification and classification:

Test Case 34

System: Attack

Attack non existent creature with item

Severity: 2

B. Instructions:

1. Pick up an item

2. Be nowhere near a creature

3. Type the command attack creature with item

C. Expected result:

The program should indicate that there is no creature to attack and the item should not be used on the non existent creature

A. Identification and classification:

Test Case 35

System: Attack

Attack creature with item

Severity: 1

B. Instructions:

1. Pick up an item

2. Be near a creature

3. Type the command attack creature with item

C. Expected result:

The creature should be attacked with the item and affected with the items abilities and the program should indicate it

A. Identification and classification:

Test Case 36

System: Attack

Attack creature with no item

Severity: 2

B. Instructions:

1. Keep inventory empty

2. Be near a creature

3. Type the command attack creature with item

C. Expected result:

The program should indicate that there is no item to use on the creature and it should fail to attack the creature

A. Identification and classification:

Test Case 37

System: Exit

Exit

Severity: 1

B. Instructions:

1. Be near the exit

2. Type the command open exit

C. Expected result:

The user should exit and the program should indicate it

A. Identification and classification:

Test Case 38

System: Exit

Exit without being near exit

Severity: 2

B. Instructions:

1. Be nowhere near the exit

2. Type the command open exit

C. Expected result:

The user should fail to exit and the program should indicate it

A. Identification and classification:

Test Case 39

System: Login

Login with correct username and password.

Severity: 1

B. Instructions:

1. At the login window, type a valid username and the correct password

C. Expected result:

User is successful logged in and can being selecting a game.

A. Identification and classification:

Test Case 40

System: Login

Login with invalid username or password.

Severity: 2

B. Instructions:

1. At the login window, type an invalid username or password

C. Expected result:

System lets user know of invalid login and does not let user continue

A. Identification and classification:

Test Case 41

System: Login

Login with no username or password entered.

Severity: 2

B. Instructions:

1. At the login window, leave username or password blank and hit login.

C. Expected result:

System lets user know username or password not entered.

A. Identification and classification:

Test Case 42

System: New Game

Start a new game

Severity: 1

B. Instructions:

Upon successful login, select "Start a new game"

C. Expected result:

A new game is launched and ready to play.

A. Identification and classification:

Test Case 43

System: Continue Game

Continue a current game

Severity: 1

B. Instructions:

Upon successful login, select "Continue an old saved game"

C. Expected result:

A list of old records for saved games will be shown.

A. Identification and classification:

Test Case 44

System: Continue Game

Select a current came to continue

Severity: 1

B. Instructions:

Select a game from the list to play.

C. Expected result:

User can resume playing the game.

A. Identification and classification:

Test Case 45
System: Continue Game
Boundary test select from list of old records
Severity: 1
B. Instructions:
Select the first game from the list to play.
C. Expected result:
User can resume playing the game.

A. Identification and classification:
Test Case 46
System: Continue Game
Boundary test - select from list of old records
Severity: 1
B. Instructions:
Select the last game from the list to play.
C. Expected result:
User can resume playing the game.

A. Identification and classification:
Test Case 47
System: Continue Game
Equivalence test - select from list of old records
Severity: 1
B. Instructions:
Select a game from middle of the list to play.
C. Expected result:
User can resume playing the game.

A. Identification and classification:
Test Case 48
System: Gameplay
Functional test: Moving player north
Severity: 2
B. Instructions:
Assuming map is loaded successfully. Start a game and begin playing. During game, type 'n'
C. Expected result:
Player moves to north by 1 unit.

A. Identification and classification:
Test Case 49

System: Gameplay

Boundary test: Moving player north results in player going to an invalid point of map

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'n'

C. Expected result:

Moving north is not possible, the application will print the description of the room, and prompt user "Can't go that way"

A. Identification and classification:

Test Case 50

System: Gameplay

Functional test: Moving player west

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'w'

C. Expected result:

Player moves to west by 1 unit.

A. Identification and classification:

Test Case 51

System: Gameplay

Boundary test: Moving player west results in player going to an invalid point of map

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'w'

C. Expected result:

Moving west is not possible, the application will print the description of the room, and prompt user "Can't go that way"

A. Identification and classification:

Test Case 52

System: Gameplay

Functional test: Moving player east

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'e'

C. Expected result:

Player moves to east by 1 unit.

A. Identification and classification:

Test Case 53

System: Gameplay

Boundary test: Moving player east results in player going to an invalid point of map

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'e'

C. Expected result:

Moving east is not possible, the application will print the description of the room, and prompt user "Can't go that way"

A. Identification and classification:

Test Case 54

System: Gameplay

Functional test: Moving player south

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 's'

C. Expected result:

Player moves to south by 1 unit.

A. Identification and classification:

Test Case 55

System: Gameplay

Boundary test: Moving player south results in player going to an invalid point of map

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 's'

C. Expected result:

Moving south is not possible, the application will print the description of the room, and prompt user "Can't go that way"

A. Identification and classification:

Test Case 56

System: Gameplay

Boundary test: Listing empty inventory

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, drop all items in inventory, then type 'i'

C. Expected result:

Program prompts "Inventory: empty".

A. Identification and classification:

Test Case 57
System: Gameplay
Functional test: Listing items in inventory
Severity: 2
B. Instructions:
Assuming map is loaded successfully. Start a game and begin playing. During game, drop all items in inventory, take 2 items and put into inventory, then type 'i'
C. Expected result:
Program will list the 2 items added initially.

A. Identification and classification:
Test Case 58
System: Gameplay
Functional test: Taking an item
Severity: 2
B. Instructions:
Assuming map is loaded successfully. Start a game and begin playing. During game, type 'take <item>'
C. Expected result:
Program prompts "Item <item> added to inventory". Make sure the item exists in inventory.

A. Identification and classification:
Test Case 59
System: Gameplay
Functional test: Taking a non-existing item
Severity: 3
B. Instructions:
Assuming map is loaded successfully. Start a game and begin playing. During game, type 'take <non-existing item>'
C. Expected result:
Program prompts that item does not exist.

A. Identification and classification:
Test Case 60
System: Gameplay
Functional test: Taking an item with full inventory
Severity: 2
B. Instructions:
Assuming map is loaded successfully. Start a game and begin playing. During game,with a full inventory, type 'take <item>'
C. Expected result:
Program should inform user that their inventory is full.

A. Identification and classification:

Test Case 61

System: Gameplay

Boundary test: Taking an invalid item

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'take '

C. Expected result:

Program should inform user that <item> to be taken cannot be null.

A. Identification and classification:

Test Case 62

System: Gameplay

Functional test: Dropping an existing item in inventory

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'drop <item>'

C. Expected result:

Program should prompt "<item> dropped". Make sure item disappears from inventory.

A. Identification and classification:

Test Case 63

System: Gameplay

Boundary test: Dropping a non-existing item in inventory

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, empty items in inventory, then type 'drop <item>'

C. Expected result:

Program should inform user that <item> is not found in inventory.

A. Identification and classification:

Test Case 64

System: Gameplay

Boundary test: Dropping an invalid item in inventory

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'drop '

C. Expected result:

Program should inform user that dropped <item> cannot be null.

A. Identification and classification:

Test Case 65

System: Gameplay

Functional test: Reading an existing item in inventory with description

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'read <item>'

C. Expected result:

Program prints the description of the item.

A. Identification and classification:

Test Case 66

System: Gameplay

Functional test: Reading a valid non existing item in inventory

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, empty the inventory, then type 'read <item>'

C. Expected result:

Program prints <item> not found.

A. Identification and classification:

Test Case 67

System: Gameplay

Boundary test: Reading an invalid item in inventory

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'read '

C. Expected result:

Program inform that <item> cannot be null.

A. Identification and classification:

Test Case 68

System: Gameplay

Functional test: Activating a valid item in inventory

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'turn on <item>'

C. Expected result:

Program prompts "You activate the <item>". Program executes the effect of the item

A. Identification and classification:

Test Case 69

System: Gameplay

Equivalence test: Activating a valid item in inventory with different effects

Severity: 2

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'turn on <item>'

C. Expected result:

Program prompts "You activate the <item>". Program executes the effect of the item. Make sure the different effect is experienced

A. Identification and classification:

Test Case 70

System: Gameplay

Boundary test: Activating a non existing item in inventory

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, empty items in the inventory, then type 'turn on <item>'

C. Expected result:

Program should inform user that item does not exist in inventory.

A. Identification and classification:

Test Case 71

System: Gameplay

Boundary test: Activating an invalid item in inventory

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'turn on '

C. Expected result:

Program should inform user that <item> cannot be null.

A. Identification and classification:

Test Case 72

System: Gameplay

Equivalence test: Executing an invalid command

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. During game, type 'invalidcommand'

C. Expected result:

Program should inform user that command does not exist.

A. Identification and classification:

Test Case 73

System: Gameplay

Equivalence test: Interacting with items (drop, read, activate) in the first, middle, and last slot of inventory

Severity: 3

B. Instructions:

Assuming map is loaded successfully. Start a game and begin playing. List the items in inventory. Try interacting (drop, read, activate) item in the first, middle, and last slot of inventory.

C. Expected result:

Program should be able to execute the command without errors.