Note that:

\*All of server replied message is relevant in meaning, not exact message as it might change along development.

A. Server 001, Connection Command, Severity 1

B. Telnet server IP address

C. If connection failed, telnet will display relevant error message; if connection successed, nothing will show but telnet will be ready to receive advanced "server command"

A. Server 002, Login Command, Severity 1

B. While telneting server, send login command with exist/not exist username and correct/incorrect password

C. Exist username & correct password should receive "success" message; exist username & incorrect password should receive "fail, wrong combination" message; not exist username & any password should receive "fail, user not exist"

A. Server 003, User Info Upload Command, Severity 2

B. While telneting server & successfully logged in, send new saving data command with in-game process

C. If success, "success" received; if fail, "fail, <reason of failing>" (possible reasons, "wrong input format", "input data size out of database field limit", "input value not expected")

A. Sever 004, User Info Download Command, Severity 2

B. While telneting server & successfully logged in, send exist saving data command

C. If success, "<data id>, <in-game process>" 10 times received; should not have fail case

A. Client 001, Upload file to S3, Severity 1

B. Create a new S3 uploader and upload a file

C. The file successfully uploads and I can see it in the S3 bucket

A. Client 002, Retrieve file list from S3, Severity 1

B. Call the MapList method

C. The method returns a list of all the existing maps

A. Client 003, Download file from S3, Severity 1

B. Call the download method with the file name

C. A new file is created in the local dir with the correct content

A. Client 004, Retrieve user saved data, Severity 1

B. Call the server to get the list saved states

C. The server will return the correct list of the saved states if username and password are correct

A. Client 005, Validate XML file, Severity 2

B. Pass an xml file to the validator

C. The validator should return true if the file is valid and false otherwise

A. Client 006, Create user, Severity 1

B. Call the server with a new username and password

C. The server returns true upon successfully create of a user

A. Client UI 001, Login Panel, Severity 1

B. Click login from initial screen, user name and password request pop up, enter and continue with OK button.

C. If username, password are correctly entered and matches, user will continue to game play screen, otherwise Login will pop up again.

A. Client UI 002, Save Game, Severity 2

B. Pull down the tool bar from upper left corner, click ‘Save’ to store the current game status.

C. If ‘Save’ is clicked, all current status in the game can be saved and generate archives successfully, otherwise error message about why will be showed up.

A. Client UI 003, Load Game, Severity 2

B. Click the ‘Load’ from the pulled down tool bar on left upper corner, choose the stored file you wanted from correct directory confirm by ’OK’ and it will import the saved archive about previous game status.

C. Upon ‘Load’->’OK’, a successful load will restore the saving point previously saved and continue game otherwise show the error message.

A. Client UI 004, Import Map, Severity 3

B. Click Import Map from the tool bar on upper left corner and import the map you made or downloaded for this game through out correct directory.

C. Upon finish, imported map will be deployed into game scene otherwise error message will be delivered.

A. Client Gameplay 001, move north, Severity 1.

B. When a player is in a room, type ‘n’ to move to the room, which is bordered in the north.

C. If a room is bordered in the north, the description of the new room is to be printed to the screen. If not, print “Can’t go that way”.

A. Client Gameplay 002, move south, Severity 1

B. When a player is in a room, type ‘s’ to move to the room, which is bordered in the south.

C. If a room is bordered in the south, the description of the new room is to be printed to the screen. If not, print “Can’t go that way”.

A. Client Gameplay 003, move west, Severity 1

B. When a player is in a room, type ‘w’ to move to the room, which is bordered in the west.

C. If a room is bordered in the west, the description of the new room is to be printed to the screen. If not, print “Can’t go that way”.

A. Client Gameplay 004, move east, Severity 1

B. When a player is in a room, type ‘e’ to move to the room, which is bordered in the east.

C. If a room is bordered in the east, the description of the new room is to be printed to the screen. If not, print “Can’t go that way”.

A. Client Gameplay 005, inventory command, Severity 1

B. Type inventory command (i) in order to list all items in player’s inventory.

C. If there is any item in player’s inventory, all items will be listed by commas. If there is no item, print “Inventory: empty”.

A. Client Gameplay 006, take item command, Severity 1

B. Type “take item” command to change item ownership from container to player’s inventory.

C. If successful, print “Item (item) added to inventory”. Otherwise, print “Can’t take item (item)”.

A. Client Gameplay 007, open container command, Severity 1

B. Use “open (container)” to check the items in the container.

C. If the container is not empty, prints contents of container in format "(container) contains (item), (item)..." and makes those items available to pick up. If empty, print "(container) is empty."

A. Client Gameplay 008, open exit command, Severity 1

B. If a player reaches the end of the game, use “open exit” command to end the game.

C. If the room is type of exit room, prints “Game over” and gracefully ends the game. Otherwise, prints “Not an exit type room”.

A. Client Gameplay 009, read item command, Severity 1

B. Type “read (item)” in order to read the information on a object.

C. Prints writing on object if any available, else prints “Nothing written.” if command is executed on an existing item in the player’s inventory that doesn’t contain writing.

A. Client Gameplay 010, drop item command, Severity 1

B. When a player wants to change item ownership from inventory to present room, use “drop (item)”.

C. Prints “(item) dropped”, if item is available in the player’s inventory. Otherwise, prints “Drop item fails.”

A. Client Gameplay 011, put item command, Severity 1

B. When players want to change item ownership from inventory to declared container, use “put (item) in (container)”.

C. If container is open and item is available in the player’s present inventory, prints “Item (item) added to (container)”. Otherwise “put item fails.”

A. Client Gameplay 012, turn on item command, Severity 1

B. If players want to activate item, which is available in inventory, type “turn on (item)”.

C. Activates item if it is in inventory, prints “You activate the (item).” and executes commands in “turn-on” element. Otherwise, prints “turn-on item fails.”

A. Client Gameplay 013, attack with command, Severity 1

B. When players want to attack creatures with some item in inventory, use “attack (creature) with (item).”

C. If item matches creature’s “vulnerability” and existing conditions are met, prints “You assault the (creature) with the (item).” Otherwise, prints “attack fails.”