Diagram Key

- Method calls or assignment operation
- Scanning the player's input
- Conditional statements
- Print out messages to the player
- Comments

All inputs for letters and strings are handled ignored case which means can be whether capital or small letters.

The start of the startGame() method which will start the game in a continuous loop until the user exits

Start

The beginning of the loop

Repeat

the clearScreen() method may throw IOException and InterruptedException exceptions

clearScreen() - clear the screen
displayLegend() - display the game's legend
displayWorld() - display the game world
displayInventory() - display the player's inventory

INPUT player's action

Can be either capital or small letters

input = 'w' || 'up'
'd' || 'right'
's' || 'down'
'a' || 'left'

Yes → unlockMode = true
No → movementCommandEntered = true
Yes → movementCommandEntered = true
movePlayer(input)

Here is the solution for the secret key, you need first to enable the unlock mode first. Then, you need to mine some blocks which means you need to do some movements in which will enable both the movement and mining commands. Lastly, you need to craft something.

unlockMode = true
Yes → input = 'c' → Yes → craftingCommandEntered = true
No → input = 'm' → Yes → miningCommandEntered = true
No → input = 'open' → Yes → openCommandEntered = true
No

Mining a block and save it to the inventory.

input = 'm'
Yes → unlockMode = true
No → miningCommandEntered = true
Yes → miningCommandEntered = true
mineBlock()

No

Place a block in the current player's position

input = 'p'
yes → display the inventory so that the player would know what he has in the inventory
displayInventory() → INPUT blockType → placeBlock(blockType)

No

Craft items from items in the inventory

input = 'c'
Yes → display the crafting recipes so that the user would know what he needs to craft an item
displayCraftingRecipes() → INPUT recipe → craftItem(recipe)

No

Interact with the current player's position which will tell the player from where he can get the current block type, i.e. wood, leaves, etc.

input = 'i'
Yes → interactWithWorld()

No

save the current game to a file in the current working dictionary including the inventory, the crafted items and unlock mode

input = 'save'
Yes → INPUT fileName → saveGame(fileName)

This method throws an IOException exception if any error occured during the process

No

load a game state from a file - you can write the full path of file or just the name of the file only if it exist in the current working dictionary (along with the extension)

input = 'load'
Yes → INPUT fileName → loadGame(fileName)

This method throws an IOException exception if any error occured during the process

No

Open the secret door when all requirements are available, otherwise, reset all the conditions.

input = 'open'
Yes → unlockMode = true
craftingCommandEntered = true
miningCommandEntered = true
movementCommandEntered = true

No → Invalid passkey. Try again!
waitForEnter()
unlockMode = false
craftingCommandEntered = false
miningCommandEntered = false
movementCommandEntered = false
openCommandEntered = false

Yes → secretDoorUnlocked = true
reset the world by setting the player position in the middle and generate a new empty world with the Duch flag
resetWorld()
Secret Door unlocked!
waitForEnter()

secretDoorUnlocked = true
Yes → clearScreen()
the clearScreen() method may throw IOException and InterruptedException exceptions
You have entered the secret area!
You are now presented a game board with a flag!
inSecretArea = true
resetWorld()
secretDoorUnlocked = false
Disable the secred door mode once it is open.
fillInventory()
Fill the player's inventory with 100 block of each type
waitForEnter()

No

Repeat

No

Display the surrounding blocks that surrounds the player from all directions

input = 'look'
Yes → lookAround()

No

Enable the unlock mode which will allow the player to start working on the passkey

input = 'unlock'
Yes → unlockMode = true

No

Get a country and a quote from that country and display them to the player

input = 'getflag'
Yes → getCountryAndQuoteFromServer() → waitForEnter()

No

Display the surrounding blocks that surrounds the player from all directions

input = 'exit'
Yes → Exiting the game. Goodbye!

No

Show a message to the user that he entered an invalid action

Invalid input. Please try again.

End