

Testing Plan for Controller

1.	Constr	uctor Testing:	
		Condition: Testing if the constructor initializes all required fields properly.	
		 Example: Initialize the controller with a non-null Readable and 	
		Appendable object, along with a valid World instance.	
		 Expected: Verify that the fields are initialized correctly and that the 	
		controller is ready to use.	
 Condition: Testing if an IllegalArgumentException is thrown when any par 		Condition: Testing if an IllegalArgumentException is thrown when any parameter	
		is null.	
		 Example: Initialize the controller with a null Readable object and a valid 	
		Appendable and World instance.	
		 Expected: Verify that the appropriate exception is thrown. 	
2. playGame() Method Testing:			
		Condition: Testing the method with various inputs, including valid and invalid	
		World instances.	
		 Example: Initialize the method with both valid and null World instances. 	
		 Expected: Verify that an IllegalArgumentException is thrown for the null 	
		World instance.	
		Condition: Testing the behavior when the maximum number of turns is 0.	
		☐ Example: Set the maximum number of turns to 0 and initialize the game.	
		 Expected: Ensure that the game does not start and an appropriate 	
		message is displayed.	
		Condition: Testing the behavior when each option is selected.	
		 Example: Simulate the selection of each option in the game menu. 	
		 Expected: Verify that the corresponding action is performed, and the 	
2	Edea 6	outputs are displayed accurately.	
3.	_	Case Testing:	
		Condition: Testing the behavior when the maximum number of turns is 1.	
		□ Example: Set the maximum number of turns to 1 and initialize the game.	
		☐ Expected: Ensure that the game ends after the first turn.	
		Condition: Testing the behavior when the maximum number of turns is negative.	
		□ Example: Set the maximum number of turns to -1 and initialize the game.	
		 Expected: Verify that an IllegalArgumentException is thrown. 	
т.		Dien fen DienleuMan DienleuDienenhafe DienleuDeenhafe	

Testing Plan for DisplayMap, DisplayPlayerInfo, DisplayRoomInfo, DisplayTargetInfo Command

1.	execute(Method	Testing:

□ Condition: Testing the method with a valid **World** instance.

	 Example: Initialize the command with a valid World instance and execute it. Expected: Verify that the method of the World class is executed without errors.
	 Condition: Testing the method with a null World instance. Example: Initialize the command with a null World instance and execute it. Expected: Verify that the method gracefully handles the null input and does not cause any unexpected errors.
	Testing Plan for PlayNextRound Command
1. execu	 Condition: Testing the method with a valid World instance and the game not in the final round. Example: Initialize the command with a valid World instance and execute it. Expected: Verify that the playNextRound() method of the World class is executed without errors. Condition: Testing the method with a valid World instance and the game in the final round. Example: Simulate a scenario where the game is in the final round. Expected: Verify that the playNextRound() method of the World class is executed, and the winner is appropriately displayed in the Appendable output.
Testing P	lan for AddComputerPlayer and AddHumanPlayer Command
1. execu	te() Method Testing:
	Condition: Testing the method with a valid World instance and valid user input. Example: Provide valid input to the command during execution. Expected: Verify that the method of the World class is executed without errors, and the appropriate confirmation message is printed.
	Condition: Testing the method with an invalid or null World instance. Example: Provide a null World instance during execution. Expected: Ensure that the command handles the null World instance appropriately, avoiding any potential runtime errors.