✔ 01. Should deploy with correct name and symbol (75ms)								
✔ 02. Should mint an NFT (69ms)	Mint a Single Research NFT	createResearchNFT()	Mints a single research NFT with metadata URI and doc	Test Implemented				
✔ 03. Should update metadata and document URI (117ms)	Retrieve Metadata URI for an NFT	tokenURI()	Retrieves the metadata URI of a specific NFT.	Test Implemented				
✔ 04. Should start an auction, allow bidding, and end auction (222ms)	Allow Auction-Style Bidding on NFTs	startAuction(), placeBid()	Starts an auction for an NFT and places a bid.	Test Implemented				
✔ 05. Should increment view count (67ms)	Track Readership Count for Research NFTs	incrementViewCount()	Increments the view count for an NFT.	Test Implemented				
✔ 06. Should pause and unpause the contract (175ms)	Enable/Disable Trading Temporarily (Pause)	pause(), unpause()	Temporarily halts trading or resumes it.	Test Implemented				
✔ 07. Should allow NFT purchase (119ms)	Purchase License to Use Research Content	buyNFT()	Allows users to purchase a license to access content.	Test Implemented				
✔ 08. Should check NFT expiration status (46ms)	Check NFT Expiration Status	isExpired()	Checks if an NFT has expired.	Test Implemented				
✔ 09. Should transfer ownership of NFT (67ms)	Retrieve List of NFTs Owned by an Address	balanceOf()	Gets the number of NFTs owned by an address.	Test Implemented				
✔ 10. Should restrict non-owners from updating metadata (60ms)	Restrict Non-Owners from Updating Metadata	updateMetadataURI()	Ensures only the NFT owner can update metadata.	Test Implemented				
✔ 11. Should burn an NFT (69ms)	Permanently Burn NFT from Circulation	burnNFT()	Destroys the NFT, removing it from circulation.	Test Implemented				
✔ 12. Should set royalties correctly (72ms)	Set Royalties for NFTs	setRoyalties()	Sets royalty percentage for future sales of the NFT.	Test Implemented				
✔ 13. Should extend auction on new bid (95ms)	Extend Auction on New Bid	placeBid()	Extends auction duration when a new bid is placed.	Test Implemented				
✔ 14. Should allow NFT purchase (102ms)	Direct Sale of NFT	buyNFT()	Allows direct purchase of an NFT without an auction.	Test Implemented				
✔ 15. Should allow checking if NFT is for sale (64ms)	Check if NFT is for Sale	isForSale()	Checks if a specific NFT is listed for sale.	Test Implemented				
✔ 16. Should not allow non-owners to start auction (55ms)	Restrict Non-Owners from Starting Auction	startAuction()	Restricts starting an auction to the owner of the NFT.	Test Implemented				
✔ 17. Should allow the NFT owner (not contract owner) to start auction (79ms	Allow NFT Owner to Start Auction	startAuction()	Allows the NFT owner to initiate an auction.	Test Implemented				
✔ 18. Should not allow purchase if not for sale (54ms)	Restrict NFT Purchase if Not for Sale	buyNFT()	Prevents the purchase of NFTs not listed for sale.	Test Implemented				
✔ 19. Should not allow bidding on non-auction NFTs (57ms)	Restrict Bidding on Non-Auction NFTs	placeBid()	Prevents bids on NFTs not currently under auction.	Test Implemented				
✔ 20. Should finalize auction correctly when ending (173ms)	Finalize Auction Correctly	endAuction()	Finalizes an auction and transfers ownership to the high	Test Implemented				
✔ 21. Should extend auction duration with a new bid (121ms)	Extend Auction Duration on New Bid	placeBid()	Extends the auction duration when a new bid is placed.	Test Implemented				
✔ 22. Should not allow re-sale of NFT after auction (165ms)	Prevent Re-Sale After Auction	setForSale()	Prevents the re-sale of NFTs after an auction.	Test Implemented				
✔ 23. Should restrict non-owners from setting royalties (84ms)	Restrict Non-Owners from Setting Royalties	setRoyalties()	Ensures that only the NFT owner can set royalties.	Test Implemented				
✔ 24. Should not allow NFT transfer during auction (154ms)	Prevent NFT Transfer During Auction	transferFrom()	Prevents transferring NFTs while an auction is active.	Test Implemented				
✔ 25. Should not allow multiple auctions for the same NFT (78ms)	Prevent Multiple Auctions for the Same NFT	startAuction()	Prevents starting a new auction while one is active.	Test Implemented				
✔ 26. Should pay royalties during NFT sale (351ms)	Transfer NFT and Pay Royalties to Original Creator	transferWithRoyalties()	Transfers the NFT and pays royalties to the original crea	Test Implemented				
✔ 27. Should restrict pausing/unpausing to the owner (113ms)	Set Admin Roles	onlyOwner, onlyResearchOwner	Sets admin roles to manage the contract.	Test Implemented				
✔ 28. Should not allow actions when the contract is paused (243ms)	Enable/Disable Trading Temporarily (Pause)	pause(), unpause()	Allows the addition and verification of research owners.	Test Implemented				
✔ 29. Should add new research owners and emit the event (265ms)	Set Admin Roles	addResearchOwner(address newOwner), isResearchOwner(address owner)		Test Implemented				
✔ 30. Should bundle multiple NFTs into one bundle NFT (315ms)	Bundle Multiple Research Reports into a Single NFT	bundleNFTs(uint256[] memory tokenIds), _createBundleNFT(	Combines multiple research reports into a single bundle	Test Implemented				
✔ 31. Should revoke access by burning an NFT (65ms)	Revoke Access by Burning the NFT	burnNFT()	Burns the NFT to revoke access.	Test Implemented				
✔ 32. Should emit ContentVerified event when verifying authenticity (42ms)	Verify Document Authenticity via Hash	verifyAuthenticity()	Verifies the authenticity of research content.	Test Implemented				
✔ 33. Should set and retrieve NFT translation (66ms)	Set and Retrieve NFT Translations	setTranslation()	Stores and retrieves translations of research content.	Test Implemented				
✔ 34. Should upgrade and downgrade subscription access level (96ms)	Renew Subscription by Updating Expiration Date	renewSubscription()	Renews the subscription period for a research NFT.	Test Implemented				
✔ 35. Should share access via minting a new NFT (71ms)	Share Access by Minting a New NFT	shareAccess()	Mints a new NFT to share access to the research conte	Test Implemented		Share Access by Minting a New NFT		
✔ 36. Should set and retrieve content preview URL	Set and Retrieve Content Preview URL	setPreview()	Sets and retrieves a preview URL for the research conte	Test Implemented				
✔ 37. Should set and get user access level	Set and Retrieve User Access Level	setUserAccessLevel(),getUserAccessLevel()	Manages access levels for specific users.	Test Implemented				
✔ 38. Should transfer contract ownership	Transfer Ownership of the Contract	transferOwnershipOfContract()	Transfers ownership of the smart contract.	Test Implemented				
✔ 39. Should reject bids lower than current highest (98ms)	Reject Bids Lower than Current Highest	placeBid()	Rejects bids lower than the current highest bid.	Test Implemented				
✔ 40. Should prevent non-owner from ending the auction (88ms)	Ensures only the NFT owner can end the auction, even after the auct	endAuction()	Ensures only the owner can end an auction after it has e	Test Implemented				
✔ 41. Should prevent bidding after auction ends (109ms)	Prevents bidding on an auction that has already ended.	placeBid()	Prevents bidding on auctions that have already ended.	Test Implemented				
✔ 42. Should revert royalty info request for non-existent token	Prevent Royalty Info Request for Non-Existent Token	royaltyInfo()	Prevents royalty info retrieval for non-existent tokens.	Test Implemented				
✔ 43. Should prevent buying an NFT not set for sale (48ms)	Prevent Buying NFT Not Set for Sale	buyNFT()	Prevents buying NFTs not explicitly listed for sale.	Test Implemented				
✔ 44. Should retrieve the correct document URI (45ms)	Retrieve Correct Document URI	researchData()	Retrieves the correct document URI for a research NFT.	Test Implemented				
✔ 45. Should increment readership count and emit event when research is vie	Track Readership Count for Research NFTs	viewResearch(), getReadershipCount()	Tracks the number of times a research NFT has been vi	Test Implemented				
✔ 46. Should correctly increment multiple views (102ms)	Track Readership Count for Research NFTs	viewResearch(), getReadershipCount()	Allows users to add feedback and ratings for research c	Test Implemented				
✔ 47. Should revert when viewing nonexistent research	Track Readership Count for Research NFTs (with Nonexistent Token	viewResearch()	Retrieves all feedback provided for a specific research N	Test Implemented				
✔ 48. Should revert when getting readership count for nonexistent research	Track Readership Count for Research NFTs (with Nonexistent Token	getReadershipCount()		Test Implemented				
✔ 49. should allow a user to add feedback (48ms)	Provide Feedback/Rating for Research Content	addFeedback()	Allows users to add comments to research content.	Test Implemented				
✔ 50. should retrieve all feedback for a token (73ms)	Provide Feedback/Rating for Research Content	getFeedback()		Test Implemented				
✔ 51. should calculate the average rating correctly (67ms)	Provide Feedback/Rating for Research Content	getAverageRating()	Calculates the average rating for a research NFT.	Test Implemented				
✔ 52. Should allow users to add comments (125ms)	Comment on NFT Research Content	addComment()	Retrieves all comments for a specific research NFT.	Test Implemented				
✔ 53. Should emit CommentAdded event upon adding a comment (72ms)	Comment on NFT Research Content	addComment()		Test Implemented				
✔ 54. Vote to promote quality research (121ms)	Vote to Promote Quality Research	voteForResearch()	Allows users to vote and provide feedback for research.	Test Implemented				
✔ 55. Award tokens for research engagement	Award Tokens for Research Engagement	awardTokens()	Rewards users with tokens for engaging with research c	Test Implemented				
✔ 56. List NFTs for sale (90ms)	List NFTs for Sale	listNFTForSale()	Lists an NFT for sale with a specified price.	Test Implemented				
✔ 57. Delist NFTs from sale (130ms)	Delist NFTs from Sale	delistNFT()	Removes an NFT from the marketplace.	Test Implemented				
✔ 58. Update sale price of NFT (127ms)	Update Sale Price of NFT	updateNFTSalePrice()	Updates the sale price of a listed NFT.	Test Implemented				
✔ 59. View NFTs on sale (60ms)	View NFTs on Sale	getNFTsOnSale()	Retrieves all NFTs currently listed for sale.	Test Implemented				
✔ 60. Retrieve all active research NFTs (91ms)	Retrieve All Active Research NFTs	getActiveResearchNFTs()	Returns a list of all active research NFTs.	Test Implemented				
✔ 61. Ban a user from interacting with NFTs	Ban a User from Interacting with NFTs	banUser()	Prevents a specific user from interacting with NFTs.	Test Implemented				
✔ 62. Issue certificate after subscription expiration (167ms)	Issue Certificate After Subscription Expiration	issueCertificate()	Issues a certificate to the NFT owner after subscription e	Test Implemented				
✔ 63. Provide update logs/history for NFTs	Provide Update Logs/History for NFTs	addNFTLog(), getNFTLogs()	Stores and retrieves update logs for NFTs.	Test Implemented				

✔ 64. Allow gifting of NFTs to another user (81ms)	Allow Gifting of NFTs to Another User	giftNFT()	Allows NFT owners to gift their NFTs to others.	Test Implemented			
✔ 65. Allow auction-style bidding on NFTs (152ms)	Allow Auction-Style Bidding on NFTs	startAuction(),placeBid(),getHighestBid()		Test Implemented	more edge case	can add	
✔ 66. Integrate oracle for real-time data use	Integrate Oracle for Real-Time Data Use	getOracleData()	Retrieves real-time data from an oracle.	Test Implemented			
✔ 67. Automatically extend subscription upon payment (140ms)	Automatically Extend Subscription Upon Payment	extendSubscription()	Extends the subscription period for an NFT upon payme	Test Implemented			
✔ 68. Verify document authenticity via hash (41ms)	Verify Document Authenticity via Hash	storeDocumentHash(), verifyDocumentHash()	Stores and verifies the authenticity of a research docum	Test Implemented			
✔ 69. Show expiration countdown (87ms)	Show Expiration Countdown	getExpirationCountdown()	Shows the remaining time before an NFT expires.	Test Implemented			
✔ 70. Enable NFT lending or renting (97ms)	Enable NFT Lending or Renting	lendNFT(), getLendingInfo()	Allows NFT owners to lend their NFTs for a fixed duratio	Test Implemented			
✔ 71. Notify user before NFT expires (110ms)	Notify User Before NFT Expires	checkForExpiryNotification()	Notifies the user before their NFT subscription expires.	Test Implemented			
✔ 72. Enable batch transfer of NFTs (151ms)	Enable Batch Transfer of NFTs	batchTransferNFTs()	Allows batch transfer of multiple NFTs.	Test Implemented			
✔ 73. Add tags or categories to NFTs (125ms)	Add Tags or Categories to NFTs	addTag(), getTags()	Allows adding and retrieving tags for NFTs.	Test Implemented			
✔ 74. Export metadata and transaction history as JSON/CSV	Export Metadata and Transaction History as JSON/CSV	exportData()	Exports metadata and transaction history as JSON/CSV	Test Implemented			
✔ 75. Should emit UserEngaged event on engagement tracking (43ms)	Track User Engagement Tokens	trackEngagement	Tracks user engagement with research NFTs.	Test Implemented			
✔ 76. Should retrieve the list of NFTs created by an address (73ms)	Retrieve List of NFTs Created by an Address	getCreatedNFTs()	Retrieves a list of all NFTs created by a specific address	Test Implemented			
✔ 77. Should mint multiple NFTs in a single transaction (135ms)	Mint Multiple Research NFTs in One Transaction	createMultipleResearchNFTs()	Mints multiple NFTs in a single transaction.	Test Implemented			
✔ 78. Should check if caller has access to the research document	Check Access	hasAccess(uint256 tokenId)	Checks if a user has access to a research document ba	Test Implemented			
✔ 79. Should retrieve the access level of the research document	Retrieve Access Level	getAccessLevel(uint256 tokenId)	Retrieves the access level of a research documen	Test Implemented			
✔ 80. Should allow users to stake tokens	Stake Tokens	stakeTokens(uint256 amount)	Allows a user to stake a specified amount of tokens.	Test Implemented			
✔ 81. Should allow users to unstake tokens	Unstake Tokens	unstakeTokens(uint256 amount)	Allows a user to unstake a portion of their staked tokens	Test Implemented			
✔ 82. Should allow users to withdraw staked tokens	Withdraw Staked Tokens	withdrawStakedTokens(uint256 amount)	Allows a user to withdraw a portion of their staked token	Test Implemented			
✔ 83. Should allow users to set a referral	Set Referral	setReferral(address referrer)	Allows a user to set a referrer for referral rewards.	Test Implemented			
✔ 84. Should reward a referral	Reward Referral	rewardReferral(address referrer, uint256 amount)	Rewards a referrer with a specified amount of tokens.	Test Implemented			
✔ 85. Should calculate dynamic price based on views	Calculate Dynamic Price	getDynamicPrice(uint256 tokenId)	Calculates the dynamic price of an NFT based on the nu	Test Implemented			
✔ 86. Should allow users to purchase a license	Purchase License	purchaseLicense(uint256 tokenId)	Allows a user to purchase a license for an NFT.	Test Implemented			
✔ 87. Should remove address from whitelist	Remove from Whitelist	removeFromWhitelist(address account)	Removes an address from the whitelist.	Test Implemented			
✔ 88. Should allow NFT owner to manage subscription	Manage Subscription	manageSubscription(uint256 tokenId, uint256 accessLevel, u	Allows an NFT owner to manage the subscription details	Test Implemented			
✔ 89. Should add address to whitelist	Add to Whitelist	addToWhitelist(address account)	Adds an address to the whitelist.	Test Implemented			