TEST PLAN FOR

LISTEN NFT

PCS24-03

Change Log

Version	Change Date	By	Description
version number	Date of Change	Name of the person who made changes	Description of the changes made
001	30/10/23	Nishant Varshney	Initial Draft

Contents

1 Introduction

ListenNFT is a WEB 3.0 platform that makes the creator and user an ad-free platform. Any artist can mint their fabulous art on the platform and showcase it to the world with any intervention of external entities.

The main objective of creating this Dapp is because artists don't get the amount, importance, and fame that they deserve. For example, a huge amount is slashed by big music firms like T-Series, Sony Music, etc.

An NFT is a digital asset representing real-world objects like art, music, in-game items, and videos. They are bought and sold online, frequently with cryptocurrency, and they are generally encoded with the same underlying software as many crypto.

NFTs are also generally one of a kind, or at least one of a minimal run, and have unique identifying codes.

By this, the Creator becomes the owner of their work and any user can buy the music to make it exclusive in exchange for paying to the owner.

1.1 Scope

1.1.1 In Scope

A) Product functions are as follows

- **Add:** one can add their NFT.
- **Upload:** one can upload their song/album on the decentralized chain.
- **Mint:** one can mint the other's songs and albums.
- **Save:** one can save added songs/albums.
- **Basket:** A place where saved albums are stored.
- **Metamask:** wallet used for minting NFTs.
- **NFT:** non-fungible tokens for minting.
- **Check for Updates**: Displays the plugins that can be updated to newer versions
- **About**: Displays the logo of Gephi, which licenses are being used, the product version, and other info

B) Functional Requirements

1) **Home page**: The homepage should provide users with a clear and concise overview of the marketplace, including featured NFTs, a search bar, categories, and any promotions or events.

2) **NFT listings:** The marketplace should allow users to view detailed information about each NFT, such as images, descriptions, creator information, and price.

3) **Search and filter**: The marketplace should provide users with powerful search and filter options, allowing them to find the NFTs that match their criteria quickly.

4) **User profile:** Users should be able to manage their profile information, view their transaction history, and access their wallets.

5) **Transaction management**: The marketplace should provide users with a streamlined and secure process for buying and selling NFTs, including options to make offers, accept bids, and complete transactions.

6) **Notifications:** The marketplace should notify users about any important updates, such as new NFT listings, offers, or transaction confirmations.

7) **Community and social features:** The marketplace should enable users to connect, share feedback and reviews, and participate in community events.

C) Non-Functional Requirements

1) **Scalability:** The marketplace should be able to handle a large volume of transactions without compromising its performance. It should be able to accommodate a large number of users and handle a high volume of traffic.

2) **Speed:** The marketplace should be fast and responsive. Transactions should be processed quickly, and users should not have to wait long periods for their transactions to be confirmed.

3) **Security:** Security is crucial in any blockchain-based system, and an NFT marketplace is no exception. The platform should have robust security measures in place to protect users' assets and prevent unauthorized access.

4) **User-friendly interface**: The marketplace should be easy to use and navigate. The interface should be intuitive and user-friendly, allowing users to quickly find the assets they are looking for and complete transactions.

5) **Low transaction fees**: The transaction fees should be reasonable and competitive compared to other NFT marketplaces. High transaction fees can discourage users from using the platform.

6) **Interoperability**: The marketplace should be compatible with multiple blockchain networks, allowing users to trade assets across different blockchains.

7) **Reliability:** The marketplace should be available and accessible at all times. Downtime or disruptions can result in lost opportunities for users, and damage the reputation of the platform.

1.1.2 Out of Scope

1) **Verification:** Verification is through MetaMask, it is not tested.
2) **File Storage**: File storage is through Pinata Api.

1.2 Quality Objective

1) **Functionality:**

- **NFT Minting**: Verify that users can create NFTs, providing all required information, including name, description, and digital assets.
- Listing and Browsing: Ensure users can list NFTs for sale and browse/search for NFTs.
- **Transaction Handling**: Test buying, selling, and bidding on NFTs.
- **User Authentication**: Verify that user accounts and authentication work correctly.
- **Wallet Integration**: Test integration with various cryptocurrency wallets for payments.
- **Smart Contracts**: Ensure that smart contracts governing NFTs operate correctly.

2) **Usability:**

- User Interface: Evaluate the user interface for simplicity and ease of use.
- Responsiveness: Verify that the website is responsive on various devices and screen sizes.
- Accessibility: Ensure the website complies with accessibility standards.

3) **Performance:**

- Load Testing: Test how the website handles concurrent user loads.
- Response Times: Measure and optimize response times for various actions.
- Scalability: Ensure the website can scale to handle an increased number of NFTs and users.

4) **Security:**

- **Data Encryption:** Verify that user data and transactions are encrypted.
- Smart Contract Audits: Conduct security audits of the smart contracts to avoid vulnerabilities.
- **DDoS Protection:** Implement measures to prevent Distributed Denial of Service (DDoS) attacks.
- **Authentication and Authorization**: Ensure that user authentication and authorization mechanisms are robust.

5) **Compatibility**:

- Browser Compatibility: Test the website on various web browsers.
- Wallet Integration: Ensure compatibility with popular cryptocurrency wallets.

6) **Data Integrity**:

- Ensure that NFT data, including metadata and ownership, remains intact and is not tampered with.

7) **Regulatory Compliance**:

- Verify that the marketplace complies with local and international regulations, especially regarding Know Your Customer (KYC) and Anti-Money Laundering (AML) requirements.

8) Interoperability:

- Test interoperability with other NFT marketplaces and blockchain networks.

9) Documentation:

- Ensure comprehensive and up-to-date documentation for users and developers.

10) Recovery and Backup:

- Implement recovery mechanisms in case of data loss or system failure.

1.3 Roles and Responsibilities

- QA Analyst – Mr. Abhishek Goyal
- Test Manager – Mr. Abhishek Goyal
- Configuration Manager – Avaneesh Singh
- Developers – Nishant Varshney, Harsh Kumar
- Installation Team – Avaneesh Singh, Nishant Varshney

.

2 Test Methodology

2.1 Overview

- **Waterfall Model**

The waterfall model is a software development model used in the context of large, complex projects, typically in the field of information technology. It is characterized by a structured, sequential approach to software development.
The waterfall model is useful in situations where the project requirements are well-defined and the project goals are clear. It is often used for large-scale projects with long timelines, where there is little room for error and the project stakeholders need to have a high level of confidence in the outcome.

Features of the Waterfall Model:

- The waterfall model involves a sequential approach to software development, where each phase of the project is completed before moving on to the next one.
- The waterfall model relies heavily on documentation to ensure that the project is well-defined and the project team is working towards a clear set of goals.
- The waterfall model places a high emphasis on quality control and testing at each phase of the project, to ensure that the final product meets the requirements and expectations of the stakeholders.
- The waterfall model involves a rigorous planning process, where the project scope, timelines, and deliverables are carefully defined and monitored throughout the project lifecycle.

Overall, the waterfall model is used in situations where there is a need for a highly structured and systematic approach to software development. It can be effective in ensuring that large, complex projects are completed on time and within budget, with a high level of quality and customer satisfaction.

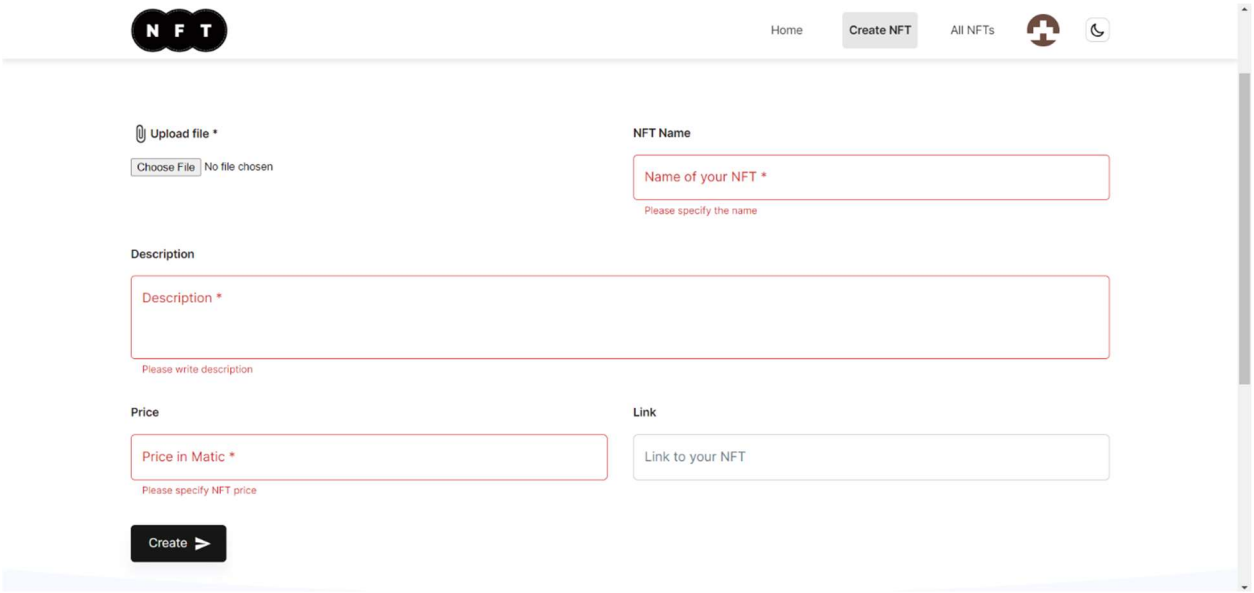
2.2 Test Levels

1) Unit Testing
2) Boundary Value Analysis
3) Equivalence Testing
4) Testing using the Selenium tool

3 Test Deliverables

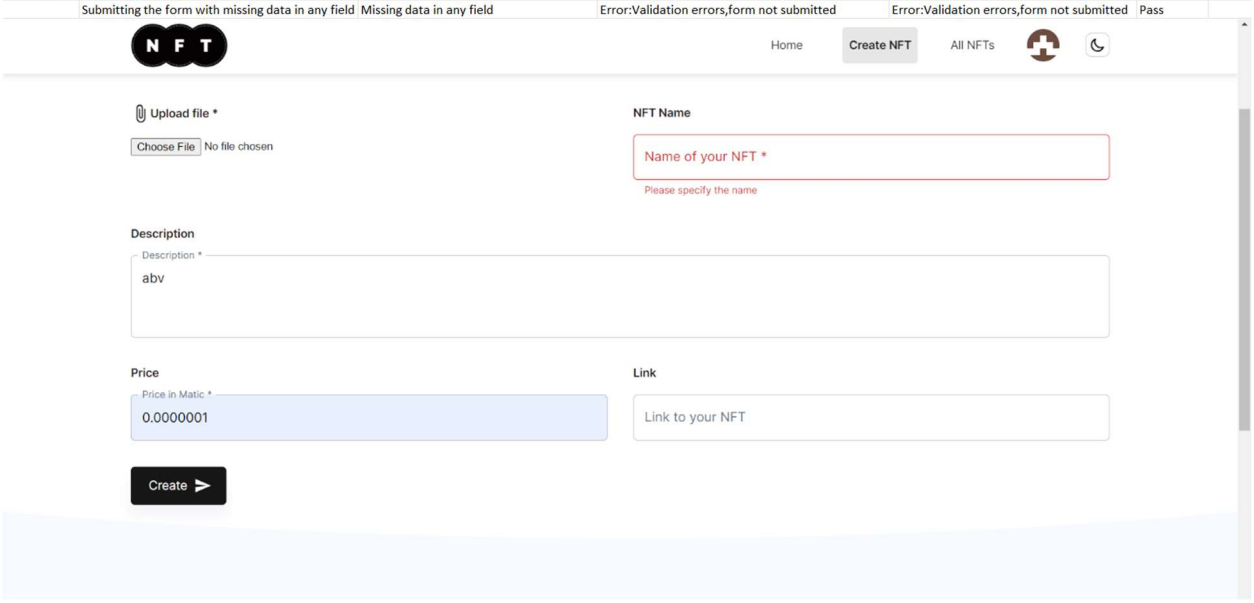
1) Unit Testing

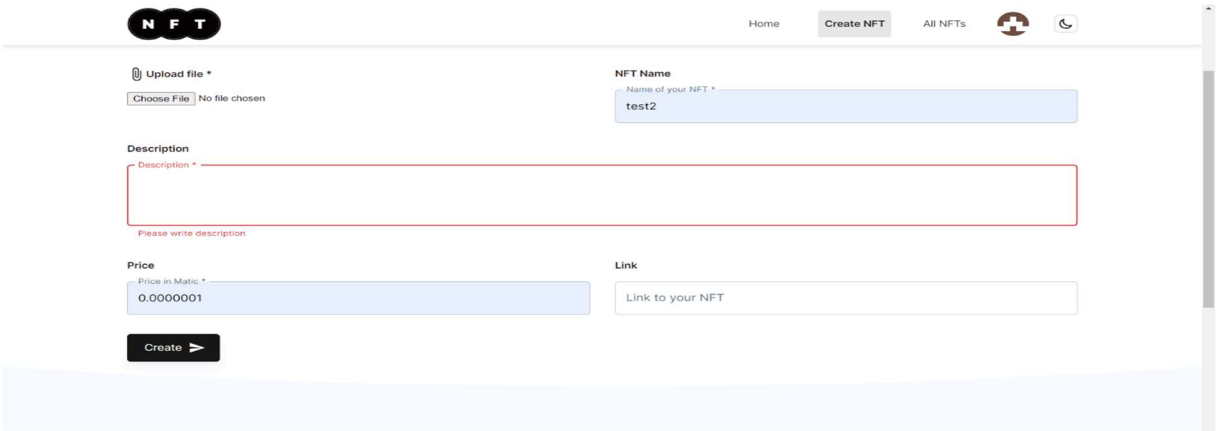
Test Case Description	Input/Action	Expected Outcome	Actual Outcome	Pass/Fail
Test form validation: Valid input	Enter valid data in all fields and submit	Form submission is successful without errors.	Form submission is successful without errors.	Pass
Test form validation: Empty name field	Leave the "Name" field empty and submit	Error is displayed for the "Name"field.	Error is displayed for the "Name"field.	Pass
Test form validation: Name too short	Enter a name shorter than 2 characters	Error is displayed for the "Name" field	Error is displayed for the "Name" field	Pass
Test form validation: Name too long	Enter a name longer than 50 characters	Error is displayed for the "Name" field.	Error is displayed for the "Name" field.	Pass
Test form validation:Empty description field	Leave the "Description" field empty and submit	Error is displayed for the "Description" field.	Error is displayed for the "Description" field.	Pass
Test form validation: Description too long	Enter a description longer than 1000 characters	Error is displayed for the "Description" field.	Error is displayed for the "Description" field.	Pass
Test form validation: Empty price field	Leave the "Price" field empty and submit	Error is displayed for the "Price" field.	Error is displayed for the "Price" field.	Pass
Test form validation: Price is zero	Enter a price of "0" and submit	Form submission is successful without errors.	Form submission is successful without errors.	Pass
Test form validation: Negative price	Enter a negative price and submit	Error is displayed for the "Price" field.	Error is displayed for the "Price" field.	Pass
Test form validation: Non- numeric price	Enter a non-numeric value in the"Price"field	Error is displayed for the "Price" field	Error is displayed for the "Price" field	Pass
Test form validation: Empty address field	Leave the "Link" field empty and submit	Form submission is successful without errors.	Form submission is successful without errors.	Pass
Test form validation: Address too short	Enter an address shorter than 3 characters	Error is displayed for the "Link" field.	Error is displayed for the "Link" field.	Pass
Test form validation: Invalid address format	Enter an invalid URL in the "Link" field	Error is displayed for the "Link" field.	Error is displayed for the "Link" field.	Pass
Test file upload: Valid file uploaded	Upload a valid file	File upload is successful, and a success message is displayed	File upload is successful, and a success message is displayed	Pass
Test file upload: No file uploaded	Submit without uploading a file	Error is displayed for file upload.	Error is displayed for file upload.	Pass
Test NFT creation :All data valid	Submit the form with valid data	NFT creation is successful,and a success message is displayed	NFT creation is successful,and a success message is displayed	Pass
Test NFT creation :Missing data	Submit the form with missing data	Error message is displayed,NFT creation fails.	Error message is displayed,NFT creation fails.	Pass



2) Equivalence Testing

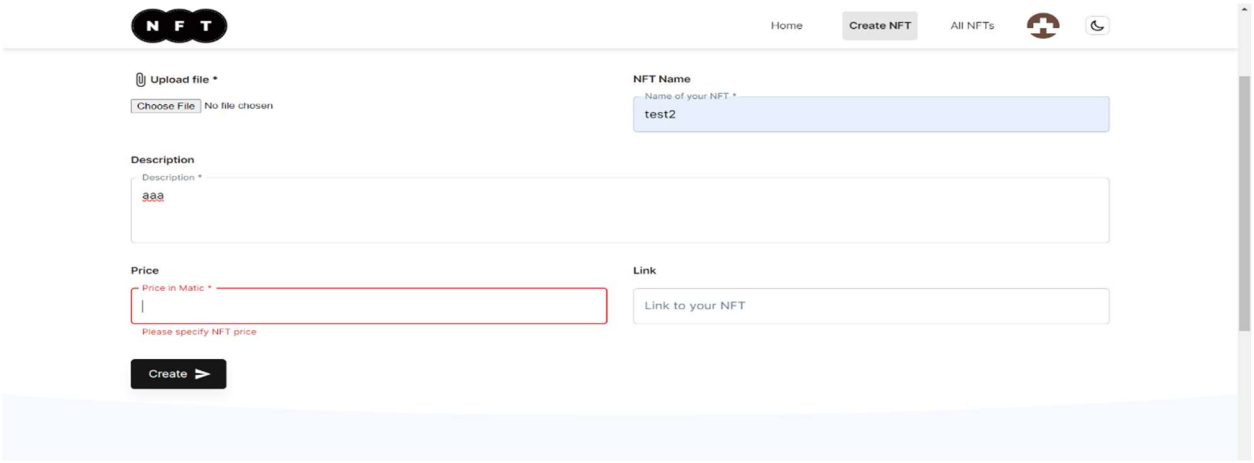
Field	Test Case Description	Input Value	Expected Outcome	Actual Outcome	Pass/Fail
					Pass
Name	Valid Name	'My NFT'	No error, form can be submitted	No error, form can be submitted	Pass
	Name too short	'A'	Error: "Name too short"	Error: "Name too short"	Pass
	Name too long	A name that is exactly 50 chars long'	Error: "Name is too long"	Error: "Name is too long"	Pass
					Pass
Description	Valid Description	'A description for my NFT'	No error, form can be submitted	No error, form can be submitted	Pass
	Description too long	A' + (998 more characters)	Error:"Should be less than 1000 chars"	Error:"Should be less than 1000 chars"	Pass
					Pass
Price	Valid Price	10'	No error, form can be submitted	No error, form can be submitted	Pass
	Price is zero(boundary value)	O'	No error, form can be submitted	No error, form can be submitted	Pass
	Negative Price (Invalid)	-10'	Error. "Price should be minimum O	Error. "Price should be minimum O	Pass
	Non-numeric Price (invalid)	abc'	Error: "Please specify NFTprice	Error: "Please specify NFTprice	Pass
					Pass
Address	Valid URL	https://example.com'	Error. "Price should be minimum 3"	Error. "Price should be minimum 3"	Pass
	URL too short	ab'	No error, form can be submitted	No error, form can be submitted	Pass
	URL with incorrect format(invalid)	example.com'	Error. "Enter correct URL!"	Error. "Enter correct URL!"	Pass
	Empty Address	"	No error, form can be submitted	No error, form can be submitted	Pass
					Pass
File Upload	Valid file upload	Valid file	No error, form can be submitted	No error, form can be submitted	Pass
	Empty file upload(boundary value)	No file uploaded	Error: "Please upload a file!	Error: "Please upload a file!	Pass
					Pass
Form Submit	Submitting the form with valid data	Valid data in all fields	Success: NFT created successfully	Success: NFT created successfully	Pass
	Submitting the form with missing data in any field	Missing data in any field	Error:Validation errors,form not submitted	Error:Validation errors,form not submitted	Pass



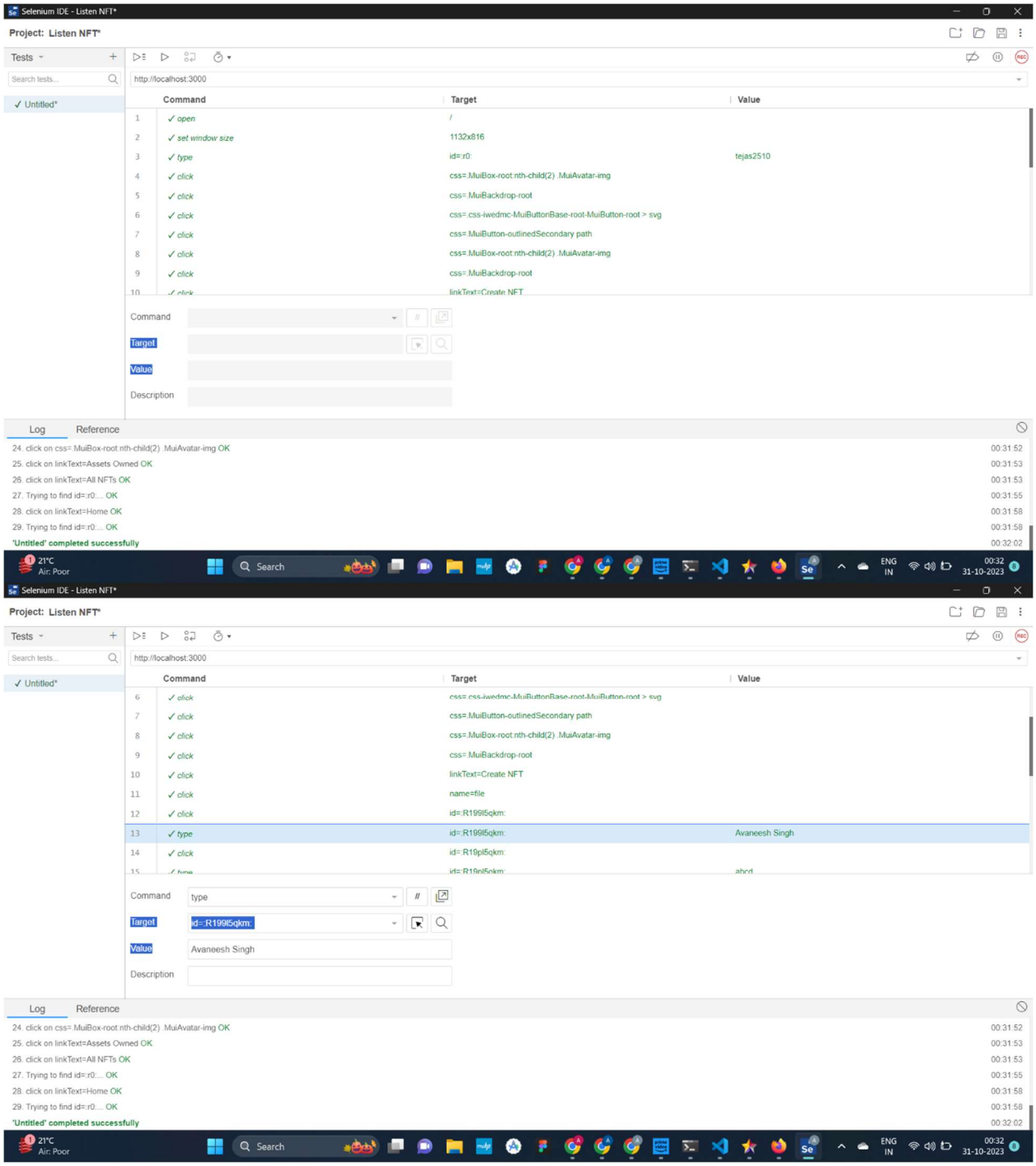


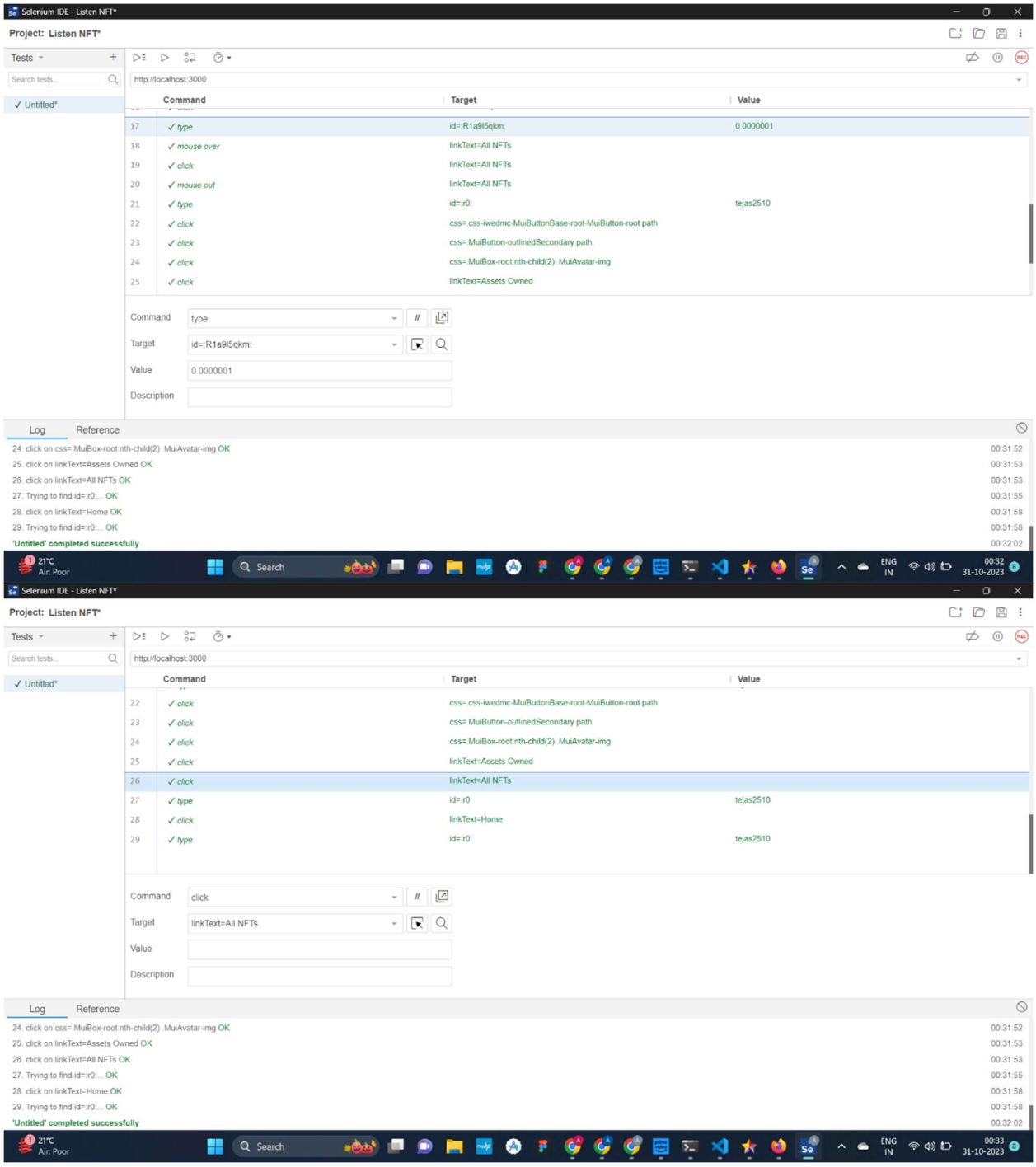
3) Boundary Value Analysis

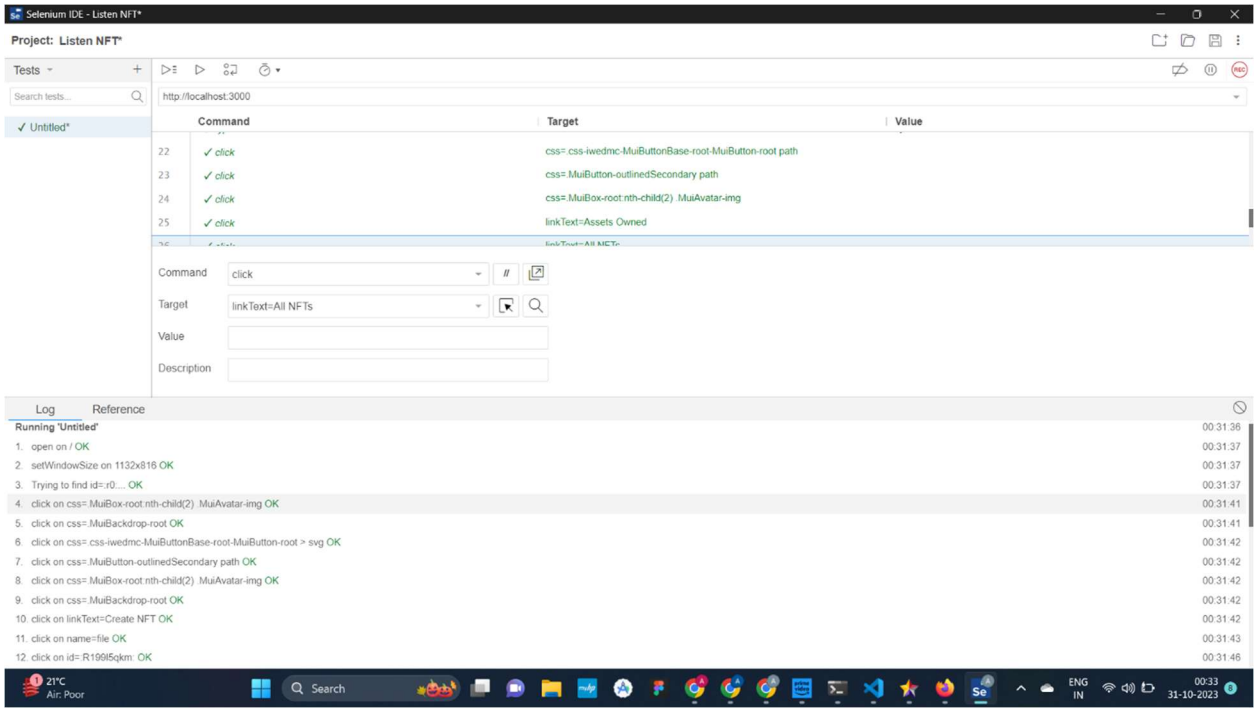
Field	Test Case Description	Input Value	Expected Outcome	Actual Outcome	Pass/Fail
Name	Minimum name length(too short)	'A'	No error, form can be submitted	No error, form can be submitted	Pass
	Maximum name length (valid)	A name that is exactly 50 characters long'	No error, form can be submitted	No error, form can be submitted	Pass
	Empty name	''	Error: "Please specify the name"	Error: "Please specify the name"	Pass
Description	Minimum description length (valid)	'A'	No error, form can be submitted	No error, form can be submitted	Pass
	Maximum description length (too long)	'A' + (997 more characters)	No error, form can be submitted	No error, form can be submitted	Pass
	Maximum description length (valid)	A' + (998 more characters)	Error: "Should be less than 1000 chars"	Error: "Should be less than 1000 chars"	Pass
	Empty description	''	Error: "Please write description"	Error: "Please write description"	Pass
Price	Minimum price (valid)	O'	No error, form can be submitted	No error, form can be submitted	Pass
	Negative Price (Invalid)	-1'	Error. "Price should be minimum O	Error. "Price should be minimum O	Pass
	Non-numeric input (invalid)	abc'	Error: "Please specify NFTprice	Error: "Please specify NFTprice	Pass
Address	Minimum address length (too short)	ab'	No error, form can be submitted	No error, form can be submitted	Pass
	Valid URL format (valid)	https://example.com'	No error, form can be submitted	No error, form can be submitted	Pass
	Invalid URL format (invalid)	example.com'	Error. "Enter correct URL!"	Error. "Enter correct URL!"	Pass
	Empty address	''	No error, form can be submitted	No error, form can be submitted	Pass
File Upload	Valid file upload (e.g.,image or video)	Valid file	No error, form can be submitted	No error, form can be submitted	Pass
	Empty file upload	No file uploaded	Error: "Please upload a file!	Error: "Please upload a file!	Pass
	Uploading an excessively large file	Very Large file	Error: (Depends on IPFS limitations, not tested here)	Error: (Depends on IPFS limitations, not tested here)	Pass
Form Submit	Submitting the form with valid data	Valid data in all fields	Success: NFT created successfully	Success: NFT created successfully	Pass
	Submitting the form with missing data in any field	Missing data in any field	Error:Validation errors,form not submitted	Error:Validation errors,form not submitted	Pass



4) Selenium Tool




Selenium IDE - Listen NFT*
Project: Listen NFT*
Tests
Search tests...
Untitled*
http://localhost:3000
Command
Target
Value
17 type id=:R1a9l5qkm: 0.0000001
18 mouse over linkText=All NFTs
19 click linkText=All NFTs
20 mouse out linkText=All NFTs
21 type id=:r0: tejas2510
22 click css=.css-iwedmc-MuiButtonBase-root-MuiButton-root path
23 click css=.MuiButton-outlinedSecondary path
24 click css=.MuiBox-root:nth-child(2) .MuiAvatar-img
25 click linkText=Assets Owned
Command type
Target id=:R1a9l5qkm:
Value 0.0000001
Description
Log
Reference
24. click on css=.MuiBox-root:nth-child(2) .MuiAvatar-img OK 00:31:52
25. click on linkText=Assets Owned OK 00:31:53
26. click on linkText=All NFTs OK 00:31:53
27. Trying to find id=:r0:... OK 00:31:55
28. click on linkText=Home OK 00:31:58
29. Trying to find id=:r0:... OK 00:31:58
'Untitled' completed successfully 00:32:02
21°C Air: Poor
Search
ENG IN
00:32 31-10-2023
Selenium IDE - Listen NFT*
Project: Listen NFT*
Tests
Search tests...
Untitled*
http://localhost:3000
Command
Target
Value
22 click css=.css-iwedmc-MuiButtonBase-root-MuiButton-root path
23 click css=.MuiButton-outlinedSecondary path
24 click css=.MuiBox-root:nth-child(2) .MuiAvatar-img
25 click linkText=Assets Owned
26 click linkText=All NFTs
27 type id=:r0: tejas2510
28 click linkText=Home
29 type id=:r0: tejas2510
Command click
Target linkText=All NFTs
Value
Description
Log
Reference
24. click on css=.MuiBox-root:nth-child(2) .MuiAvatar-img OK 00:31:52
25. click on linkText=Assets Owned OK 00:31:53
26. click on linkText=All NFTs OK 00:31:53
27. Trying to find id=:r0:... OK 00:31:55
28. click on linkText=Home OK 00:31:58
29. Trying to find id=:r0:... OK 00:31:58
'Untitled' completed successfully 00:32:02
21°C Air: Poor
Search
ENG IN
00:33 31-10-2023


Selenium IDE - Listen NFT*
Project: Listen NFT*
Tests
Search tests...
Untitled*
http://localhost:3000
Command
Target
Value
22 click css=.css-iwedmc-MuiButtonBase-root-MuiButton-root path
23 click css=.MuiButton-outlinedSecondary path
24 click css=.MuiBox-root:nth-child(2) .MuiAvatar-img
25 click linkText=Assets Owned
Command click
Target linkText=All NFTs
Value
Description
Log
Reference
Running 'Untitled'
1. open on / OK
2. setWindowSize on 1132x816 OK
3. Trying to find id=:r0:... OK
4. click on css=.MuiBox-root:nth-child(2) .MuiAvatar-img OK
5. click on css=.MuiBackdrop-root OK
6. click on css=.css-iwedmc-MuiButtonBase-root-MuiButton-root > svg OK
7. click on css=.MuiButton-outlinedSecondary path OK
8. click on css=.MuiBox-root:nth-child(2) .MuiAvatar-img OK
9. click on css=.MuiBackdrop-root OK
10. click on linkText=Create NFT OK
11. click on name=file OK
12. click on id=:R199l5qkm: OK
Search
ENG
IN
00:33
31-10-2023

4 Resource & Environment Needs

4.1 Testing Tools

1) Selenium

4.2 Test Environment

The following **software is** required in addition to client-specific software.

- Windows 8 and above
- Office 2013 and above
- MS Exchange, etc.

5 Terms/Acronyms

Make a mention of any terms or acronyms used in the project:

TERM/ACRONYM	DEFINITION
API	Application Program Interface
AUT	Application Under Test

Mr. Abhishek Goyal
(Testing Lab Faculty)

Dr. Harsh Khatter
(Project Guide)

Submitted by:
Avaneesh Singh – 2000290120047
Harsh Kumar – 2000290120092
Nishant Varshney – 2000290120102

Submitted To:
Prof. Neha Shukla
Project Coordinator
Deptt. of Computer Science