Smart Contract
Security Audit
V1

theNFtreasury Marketplace Smart Contract

5/2/2023

https://saferico.com/

business@saferico.com
https://t.me/SFI_ANN

–

Table of Contents

Background

The purpose of the audit was to achieve the following:

- Ensure that the smart contract functions as intended.
- Identify potential security issues with the smart contract.

The information in this report should be used to understand the risk exposure of the smart contract, and as a guide to improve the security posture of the smart contract by remediating the issues that were identified.

Project Information

- **Platform**: Polygon smart chain
- **Contract Address**: 0x1ff69d7e2c238fff08a5f5988f0c807f89c32a02
- **Code:**

https://github.com/kevinjanada/nftreasury-contracts

Smart contract Information

- Name: theNFtreasury MarketPlace for theNFtreasury collection.

Test net addresses:

https://mumbai.polygonscan.com/address/0xaE983F165dC5aaD40B5ee4B0311Ae455e139f43c

https://mumbai.polygonscan.com/address/0xf0d4f72fb649dd2d7a76743f82ab4365b07f8305

Contracts address deployed to test net (polygon)

theNFtreasury Marketplace smart contract on polygon test net to test functions by the auditor.

https://thirdweb.com/mumbai/0x1ff69D7E2C238fff08a5f5988f0c807f89C32A02/

https://mumbai.polygonscan.com/address/0x1ff69d7e2c238fff08a5f5988f0c807f89c32a02

https://thirdweb.com/mumbai/0x0b3a653513eba266a511B59eb88D592Fb7677fa9/

https://mumbai.polygonscan.com/address/0x0b3a653513eba266a511b59eb88d592fb7677fa9

Executive Summary

According to our assessment, the customer`s solidity smart contract is **“WELL SECURED”**.

Well Secured	✓
Secured	
Poor Secured	
Insecure	

Automated checks are with remix IDE. All issues were performed by the team, which included the analysis of code functionality, manual audit found during automated analysis were manually reviewed and applicable vulnerabilities are presented in the audit overview section. The general overview is presented in the Project Information section and all issues found are located in the audit overview section.

Team found 0 critical, 0 high, 0 medium, 2 low, 0 very low-level issues and 0 note in all solidity files of the contract

P.S: the smart contract is import thirdweb libraries https://thirdweb.com/ which is one of most secure libraries And you can check all audit reports of thirdweb smart contracts here:
https://thirdweb.com/explore

The files:

theNFtreasuryMarketplace.sol

File and Function Level Report

File in Scope:

Contract Name	SHA 256 hash	Contract Address
NFTreasuryMarketplace.sol	**355f56b691b96e6488390e1acad38f27876bd37b452c738f4f2c80c3da6a2711**	0x1ff69d7e2c238fff08a5f5988f0c807f89c32a02

- Contract: NFTreasuryMarketplace
- Inherit:Initializable,INFTreasuryMarketplace,ReentrancyGuardUpgradeable,ERC2771ContextUpgradeable,MulticallUpgradeable,AccessControlEnumerableUpgradeable,IERC721ReceiverUpgradeable, IERC1155ReceiverUpgradeable
- Observation: All passed including security check
- Test Report: passed
- Score: passed
- Conclusion: passed

Function	Test Result	Type / Return Type	Score
AUCTION_ENABLED	✓	Read / public	**Passed**
bidBufferBps	✓	Read / public	**Passed**
contractType	✓	Read / public	**Passed**
contractURI	✓	Read / public	**Passed**
contractVersion	✓	Read / public	**Passed**
DEFAULT_ADMIN_ROLE	✓	Read / public	**Passed**
getPlatformFeeInfo	✓	Read / public	**Passed**
getRoleAdmin	✓	Read / public	**Passed**
getRoleMember	✓	Read / public	**Passed**
getRoleMemberCount	✓	Read / public	**Passed**
hasRole	✓	Read / public	**Passed**

isTrustedForwarder	✓	Read / public	**Passed**
LIST_PRICE_BPS_INCREASE	✓	Read / public	**Passed**
Listing	✓	Read / public	**Passed**
mainNFT	✓	Read / public	**Passed**
MAX_BPS	✓	Read / public	**Passed**
offers	✓	Read / public	**Passed**
onERC721Received	✓	Read / public	**Passed**
OUTSIDE_LISTING_ALLOWED	✓	Read / public	**Passed**
supportsInterface	✓	Read / public	**Passed**
timeBuffer	✓	Read / public	**Passed**
totalListings	✓	Read / public	**Passed**
winningBid	✓	Read / public	**Passed**
acceptOffer	✓	Write / public	**Passed**
buy	✓	Write / payable	**Passed**
cancelDirectListing	✓	Write / public	**Passed**
closeAuction	✓	Write / public	**Passed**
createListing	✓	Write / public	**Passed**
grantRole	✓	Write / public	**Passed**
multiCall	✓	Write / public	**Passed**
initialize	✓	Write / public	**Passed**
onERC1155BatchReceived	✓	Write / public	**Passed**
offer	✓	Write / payable	**Passed**
onERC1155Received	✓	Write / public	**Passed**
renounceRole	✓	Write / public	**Passed**
revokeRole	✓	Write / public	**Passed**
setAuctionBuffers	✓	Write / public	**Passed**

setAuctionEnabled	✓	Write / public	**Passed**
setConractURI	✓	Write / public	**Passed**
setListPriceBpsIncrease	✓	Write / public	**Passed**
setMainNFT	✓	Write / public	**Passed**
setOutsideListingAllowed	✓	Write / public	**Passed**
setPlatformFeeInfo	✓	Write / public	**Passed**
updateListing	✓	Write / public	**Passed**

Issues Checking Status

No.	Issue Description	Checking Status
1	Compiler warnings.	**Passed**
2	Race conditions and Reentrancy. Cross-function race conditions.	**Passed**
3	Possible delays in data delivery.	**Passed**
4	Oracle calls.	**Passed**
5	Design Logic.	**Passed**
6	Timestamp dependence.	**Passed**
7	Integer Overflow and Underflow.	**Passed**
8	DoS with Revert.	**Passed**
9	DoS with block gas limit.	**Passed with Notes**
10	Methods execution permissions.	**Passed**
11	Economy model. If application logic is based on an incorrect economic model, the application would not function correctly and participants would incur financial losses. This type of issue is most often found in bonus rewards systems, Staking and Farming contracts, Vault and Vesting contracts, etc.	**Passed**
12	The impact of the exchange rate on the logic.	**Passed**
13	Private user data leaks.	**Passed**
14	Malicious Event log.	**Passed**
15	Scoping and Declarations.	**Passed**
16	Uninitialized storage pointers.	**Passed**
17	Arithmetic accuracy.	**Passed**

Severity Definitions

Risk Level	Description
Critical	Critical vulnerabilities are usually straightforward to exploit and can lead to tokens loss etc.
High	High-level vulnerabilities are difficult to exploit; however, they also have significant impact on smart contract execution, e.g. public access to crucial functions
Medium	Medium-level vulnerabilities are important to fix; however, they can’t lead to tokens lose
Low	Low-level vulnerabilities are mostly related to outdated, unused etc. code snippets, that can’t have significant impact on execution
Note	Lowest-level vulnerabilities, code style violations and info statements can’t affect smart contract execution and can be ignored.

Audit Findings

Critical:

No Critical severity vulnerabilities were found.

High:

No High severity vulnerabilities were found.

Medium:

No Medium severity vulnerabilities were found

Low:

#Use of block.timestamp for comparisons

Description

The value of block.timestamp can be manipulated by the miner.
And conditions with strict equality is difficult to achieve - block.timestamp

Remediation

Avoid use of block.timestamp

Status: Acknowledged

#Pragam version not fixed

Description

It is a good practice to lock the solidity version for a live deployment (use 0.8.18 instead of ^0.8.11). contracts should be deployed with the same compiler version and flags that they have been tested the most with. Locking the pragma helps ensure that contracts do not accidentally get deployed using, for example, the latest compiler which may have higher risks of undiscovered bugs. Contracts may also be deployed by others and the pragma indicates the compiler version intended by the original authors.

Remediation

Remove the ^ sign to lock the pragma version.

Status: Acknowledged.

Very Low:

No Very Low severity vulnerabilities were found.

Notes:

No Notes were found.

Automatic Testing

1- Check for security

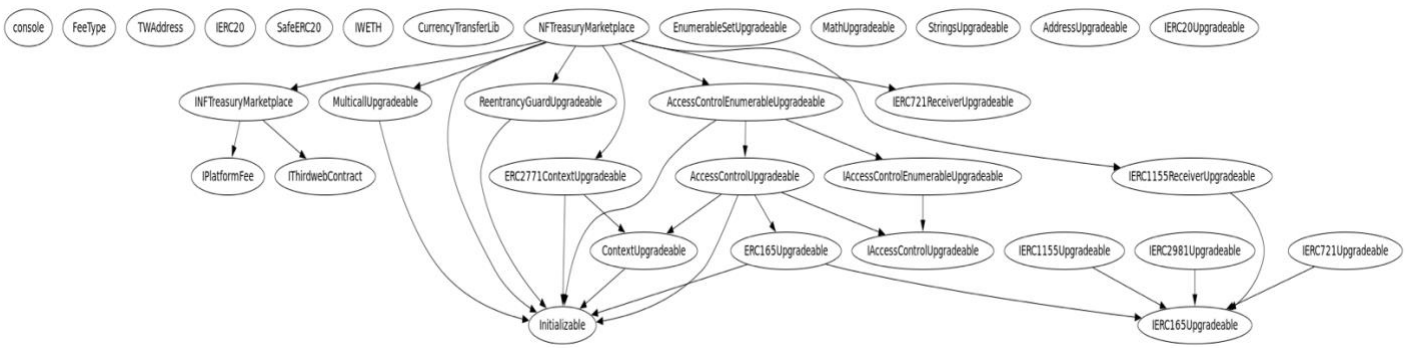
355f56b691b96e6488390e1acad38f27876bd37b452c738f4f2c80c3da6a2711

File: NFTrea... | Language: solidity | Size: 38727 bytes | Date: 2023-02-05T12:36:50.584Z

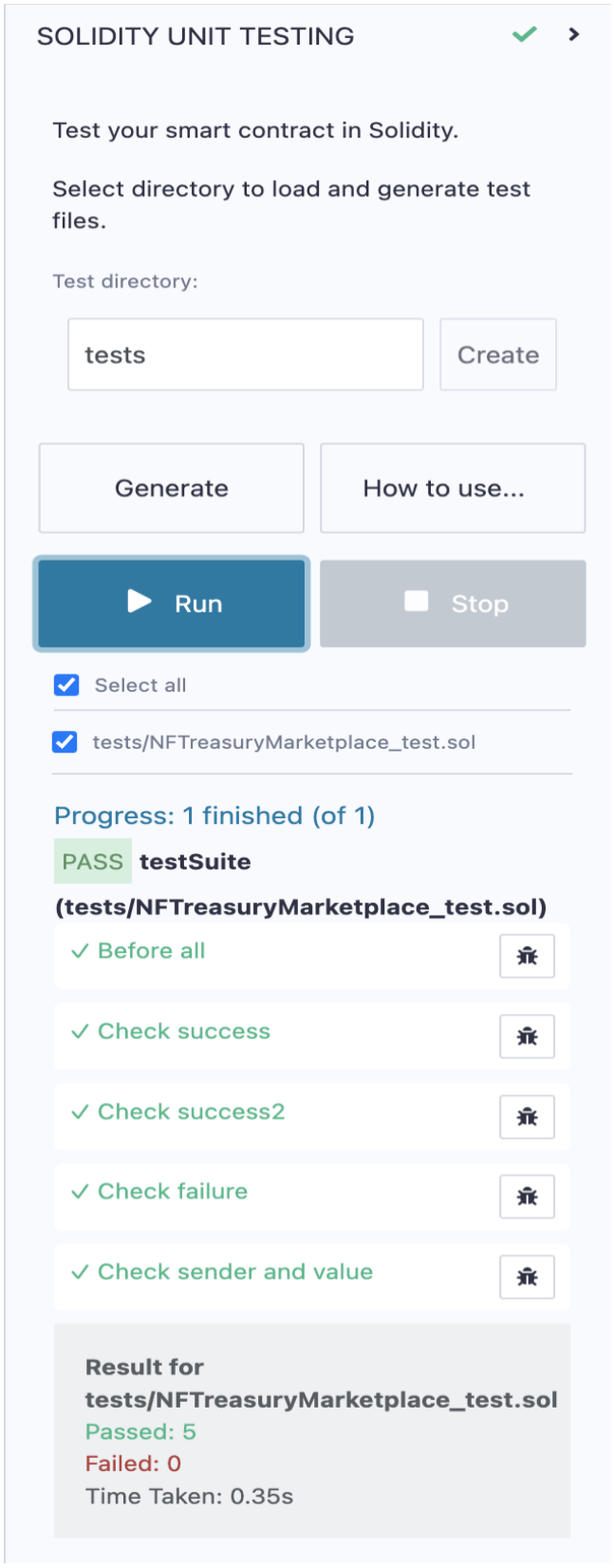
Critical	High	Medium	Low	Note
0	0	0	0	0

2- SOLIDITY STATIC ANALYSIS

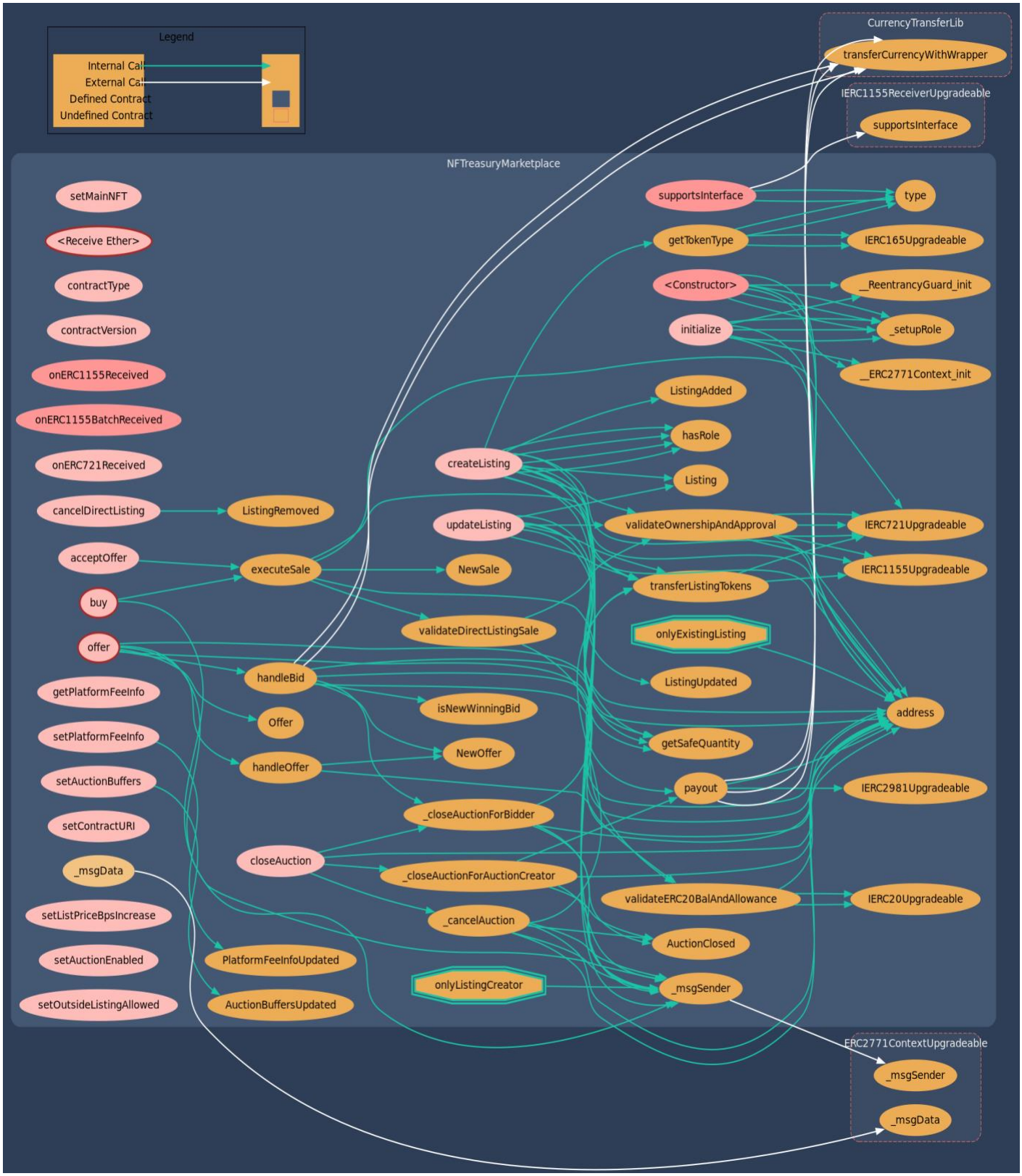
3- Inheritance graph



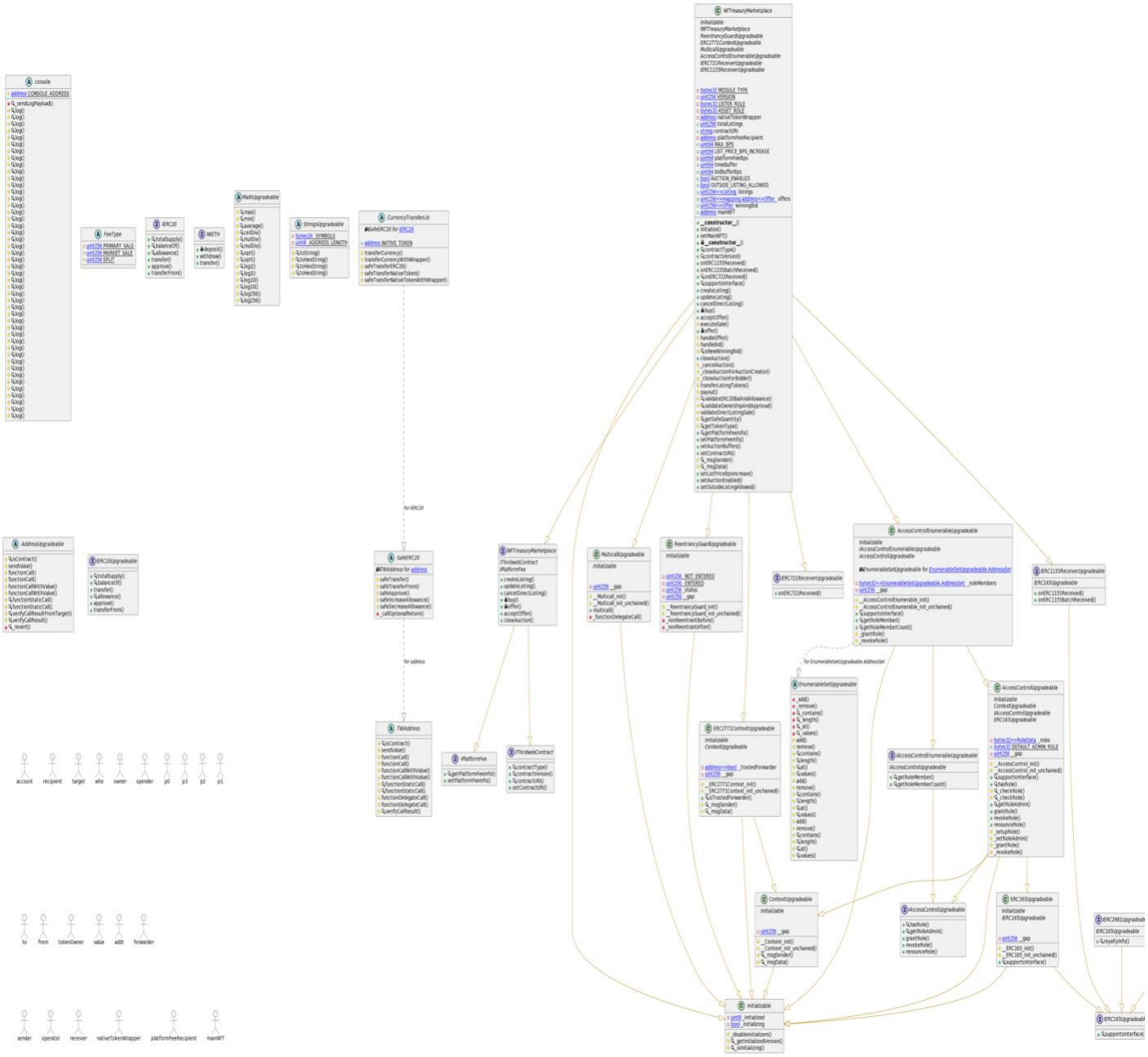
4- SOLIDITY UNIT TESTING



5- Call graph


Legend
Internal Call
External Call
Defined Contract
Undefined Contract
CurrencyTransferLib
transferCurrencyWithWrapper
IERC1155ReceiverUpgradeable
supportsInterface
NFTreasuryMarketplace
setMainNFT
<Receive Ether>
contractType
contractVersion
onERC1155Received
onERC1155BatchReceived
onERC721Received
cancelDirectListing
ListingRemoved
acceptOffer
executeSale
NewSale
buy
offer
validateDirectListingSale
handleBid
getPlatformFeeInfo
Offer
isNewWinningBid
setPlatformFeeInfo
handleOffer
NewOffer
setAuctionBuffers
setContractURI
_closeAuctionForBidder
_msgData
closeAuction
_closeAuctionForAuctionCreator
setListPriceBpsIncrease
_cancelAuction
setAuctionEnabled
PlatformFeeInfoUpdated
onlyListingCreator
setOutsideListingAllowed
AuctionBuffersUpdated
supportsInterface
type
getTokenType
IERC165Upgradeable
<Constructor>
__ReentrancyGuard_init
initialize
_setupRole
__ERC2771Context_init
ListingAdded
hasRole
createListing
Listing
updateListing
validateOwnershipAndApproval
IERC721Upgradeable
IERC1155Upgradeable
transferListingTokens
onlyExistingListing
ListingUpdated
address
getSafeQuantity
payout
IERC2981Upgradeable
validateERC20BalAndAllowance
IERC20Upgradeable
AuctionClosed
_msgSender
ERC2771ContextUpgradeable
_msgSender
_msgData

Unified Modeling Language (UML)



Functions signature

```
Sighash   |   Function Signature
========================
16279055  =>  isContract(address)
63183678  =>  log(address,string,uint256,address)
87033622  =>  setAuctionEnabled(bool)
47ee4fe3  =>  _sendLogPayload(bytes)
0ef7e050  =>  log(address,string,uint256,bool)
159f8927  =>  log(address,string,string,uint256)
5d02c50b  =>  log(address,string,string,string)
35a5071f  =>  log(address,string,string,bool)
a04e2f87  =>  log(address,string,string,address)
515e38b6  =>  log(address,string,bool,uint256)
bc0b61fe  =>  log(address,string,bool,string)
5f1d5c9f  =>  log(address,string,bool,bool)
205871c2  =>  log(address,string,bool,address)
457fe3cf  =>  log(address,string,address,uint256)
f7e36245  =>  log(address,string,address,string)
0df12b76  =>  log(address,string,address,bool)
0d36fa20  =>  log(address,string,address,address)
386ff5f4  =>  log(address,bool,uint256,uint256)
0aa6cfad  =>  log(address,bool,uint256,string)
c4643e20  =>  log(address,bool,uint256,bool)
ccf790a1  =>  log(address,bool,uint256,address)
80e6a20b  =>  log(address,bool,string,uint256)
475c5c33  =>  log(address,bool,string,string)
50ad461d  =>  log(address,bool,string,bool)
19fd4956  =>  log(address,bool,string,address)
8c4e5de6  =>  log(address,bool,bool,uint256)
dfc4a2e8  =>  log(address,bool,bool,string)
cac43479  =>  log(address,bool,bool,bool)
cf394485  =>  log(address,bool,bool,address)
a75c59de  =>  log(address,bool,address,uint256)
2dd778e6  =>  log(address,bool,address,string)
a6f50b0f  =>  log(address,bool,address,bool)
660375dd  =>  log(address,bool,address,address)
be553481  =>  log(address,address,uint256,uint256)
fdb4f990  =>  log(address,address,uint256,string)
9b4254e2  =>  log(address,address,uint256,bool)
8da6def5  =>  log(address,address,uint256,address)
ef1cefe7  =>  log(address,address,string,uint256)
21bdaf25  =>  log(address,address,string,string)
6f1a594e  =>  log(address,address,string,bool)
8f736d16  =>  log(address,address,string,address)
3971e78c  =>  log(address,address,bool,uint256)
aa6540c8  =>  log(address,address,bool,string)
2cd4134a  =>  log(address,address,bool,bool)
9f1bc36e  =>  log(address,address,bool,address)
94250d77  =>  log(address,address,address,uint256)
f808da20  =>  log(address,address,address,string)
0e378994  =>  log(address,address,address,bool)
665bf134  =>  log(address,address,address,address)
24a084df  =>  sendValue(address,uint256)
a0b5ffb0  =>  functionCall(address,bytes)
```

```
241b5886  =>  functionCall(address,bytes,string)
2a011594  =>  functionCallWithValue(address,bytes,uint256)
d525ab8a  =>  functionCallWithValue(address,bytes,uint256,string)
c21d36f3  =>  functionStaticCall(address,bytes)
dbc40fb9  =>  functionStaticCall(address,bytes,string)
ee33b7e2  =>  functionDelegateCall(address,bytes)
57387df0  =>  functionDelegateCall(address,bytes,string)
946b5793  =>  verifyCallResult(bool,bytes,string)
18160ddd  =>  totalSupply()
70a08231  =>  balanceOf(address)
dd62ed3e  =>  allowance(address,address)
a9059cbb  =>  transfer(address,uint256)
095ea7b3  =>  approve(address,uint256)
23b872dd  =>  transferFrom(address,address,uint256)
d0c407e1  =>  safeTransfer(IERC20,address,uint256)
5beae096  =>  safeTransferFrom(IERC20,address,address,uint256)
d6dcec8d  =>  safeApprove(IERC20,address,uint256)
390cc046  =>  safeIncreaseAllowance(IERC20,address,uint256)
5164ffed  =>  safeDecreaseAllowance(IERC20,address,uint256)
becc5a20  =>  _callOptionalReturn(IERC20,bytes)
d0e30db0  =>  deposit()
2e1a7d4d  =>  withdraw(uint256)
31c13bd8  =>  transferCurrency(address,address,address,uint256)
02b63f89  =>  transferCurrencyWithWrapper(address,address,address,uint256,address)
557b00f3  =>  safeTransferERC20(address,address,address,uint256)
3e167aaf  =>  safeTransferNativeToken(address,uint256)
f4f0ca3e  =>  safeTransferNativeTokenWithWrapper(address,uint256,address)
d45573f6  =>  getPlatformFeeInfo()
1e7ac488  =>  setPlatformFeeInfo(address,uint256)
cb2ef6f7  =>  contractType()
a0a8e460  =>  contractVersion()
e8a3d485  =>  contractURI()
938e3d7b  =>  setContractURI(string)
23ca32c3  =>  createListing(ListingParameters,address)
c4b5b15f  =>
updateListing(uint256,uint256,uint256,uint256,address,uint256,uint256)
7506c84a  =>  cancelDirectListing(uint256)
7687ab02  =>  buy(uint256,address,uint256,address,uint256)
5fef45e7  =>  offer(uint256,uint256,address,uint256,uint256)
b13c0e63  =>  acceptOffer(uint256,address,address,uint256)
6bab66ae  =>  closeAuction(uint256,address)
616848ba  =>  _add(Set,bytes32)
ccc22744  =>  _remove(Set,bytes32)
9ce9d722  =>  _contains(Set,bytes32)
9cbb0a1c  =>  _length(Set)
c203e0ff  =>  _at(Set,uint256)
96b91c48  =>  _values(Set)
5581e150  =>  add(Bytes32Set,bytes32)
74ea0f04  =>  remove(Bytes32Set,bytes32)
b933a783  =>  contains(Bytes32Set,bytes32)
26c1be23  =>  length(Bytes32Set)
b06d4168  =>  at(Bytes32Set,uint256)
0180d2d9  =>  values(Bytes32Set)
49ca7600  =>  add(AddressSet,address)
4e257bd0  =>  remove(AddressSet,address)
8bd27f2f  =>  contains(AddressSet,address)
```

```
ccdae21e  =>  length(AddressSet)
fa3592c6  =>  at(AddressSet,uint256)
2b8b1529  =>  values(AddressSet)
a58c3260  =>  add(UintSet,uint256)
81319dc3  =>  remove(UintSet,uint256)
47cce690  =>  contains(UintSet,uint256)
7ac93b84  =>  length(UintSet)
3cba95ef  =>  at(UintSet,uint256)
e4469c14  =>  values(UintSet)
6d5433e6  =>  max(uint256,uint256)
7ae2b5c7  =>  min(uint256,uint256)
2b7423ab  =>  average(uint256,uint256)
9cb35327  =>  ceilDiv(uint256,uint256)
aa9a0912  =>  mulDiv(uint256,uint256,uint256)
1db78456  =>  mulDiv(uint256,uint256,uint256,Rounding)
677342ce  =>  sqrt(uint256)
a902bc5e  =>  sqrt(uint256,Rounding)
5456bf13  =>  log2(uint256)
2ee6af53  =>  log2(uint256,Rounding)
ebdae5f9  =>  log10(uint256)
f86799ff  =>  log10(uint256,Rounding)
36cb4c48  =>  log256(uint256)
2910b3a1  =>  log256(uint256,Rounding)
6900a3ae  =>  toString(uint256)
8fba8d5c  =>  toHexString(uint256)
63e1cbea  =>  toHexString(uint256,uint256)
1bb0c665  =>  toHexString(address)
1daa78c1  =>  verifyCallResultFromTarget(address,bool,bytes,string)
6cadf5e1  =>  _revert(bytes,string)
8129fc1c  =>  initialize()
5cd8a76b  =>  initializeV2()
4caf63ac  =>  _disableInitializers()
69dc0693  =>  _getInitializedVersion()
8f44d3b0  =>  _isInitializing()
d38c3bf3  =>  __Multicall_init()
dfe1d074  =>  __Multicall_init_unchained()
ac9650d8  =>  multicall(bytes[])
378f61a0  =>  _functionDelegateCall(address,bytes)
97cee86a  =>  __ReentrancyGuard_init()
547602e1  =>  __ReentrancyGuard_init_unchained()
62898eb8  =>  _nonReentrantBefore()
c7443eb2  =>  _nonReentrantAfter()
f08d647e  =>  __Context_init()
ab96f671  =>  __Context_init_unchained()
119df25f  =>  _msgSender()
8b49d47e  =>  _msgData()
29748fc7  =>  __ERC2771Context_init(address[])
d946e18b  =>  __ERC2771Context_init_unchained(address[])
572b6c05  =>  isTrustedForwarder(address)
91d14854  =>  hasRole(bytes32,address)
248a9ca3  =>  getRoleAdmin(bytes32)
2f2ff15d  =>  grantRole(bytes32,address)
d547741f  =>  revokeRole(bytes32,address)
36568abe  =>  renounceRole(bytes32,address)
9010d07c  =>  getRoleMember(bytes32,uint256)
ca15c873  =>  getRoleMemberCount(bytes32)
```

```
150b7a02  =>  onERC721Received(address,address,uint256,bytes)
01ffc9a7  =>  supportsInterface(bytes4)
02aecff4  =>  __ERC165_init()
15a8d1d0  =>  __ERC165_init_unchained()
c2985578  =>  foo()
2fec1469  =>  __AccessControl_init()
7a156eb1  =>  __AccessControl_init_unchained()
6bb50616  =>  _checkRole(bytes32)
5b7b2c38  =>  _checkRole(bytes32,address)
4fa943a6  =>  _setupRole(bytes32,address)
7612997d  =>  _setRoleAdmin(bytes32,bytes32)
ce2cc1d0  =>  _grantRole(bytes32,address)
2c95bd23  =>  _revokeRole(bytes32,address)
8413eac3  =>  __AccessControlEnumerable_init()
0c05793b  =>  __AccessControlEnumerable_init_unchained()
2a55205a  =>  royaltyInfo(uint256,uint256)
f23a6e61  =>  onERC1155Received(address,address,uint256,uint256,bytes)
bc197c81  =>  onERC1155BatchReceived(address,address,uint256[],uint256[],bytes)
6352211e  =>  ownerOf(uint256)
b88d4fde  =>  safeTransferFrom(address,address,uint256,bytes)
42842e0e  =>  safeTransferFrom(address,address,uint256)
a22cb465  =>  setApprovalForAll(address,bool)
081812fc  =>  getApproved(uint256)
e985e9c5  =>  isApprovedForAll(address,address)
00fdd58e  =>  balanceOf(address,uint256)
4e1273f4  =>  balanceOfBatch(address[],uint256[])
f242432a  =>  safeTransferFrom(address,address,uint256,uint256,bytes)
2eb2c2d6  =>  safeBatchTransferFrom(address,address,uint256[],uint256[],bytes)
a0bfdd31  =>  initialize(address,string,address[],address,uint256,uint64)
15b12494  =>  setMainNFT(address)
14c88a42  =>  executeSale(Listing,address,address,address,uint256,uint256)
5321a55b  =>  handleOffer(Listing,Offer)
f86b1a5f  =>  handleBid(Listing,Offer)
2e79239c  =>  isNewWinningBid(uint256,uint256,uint256)
45d28ef4  =>  _cancelAuction(Listing)
98ed9917  =>  _closeAuctionForAuctionCreator(Listing,Offer)
753fc451  =>  _closeAuctionForBidder(Listing,Offer)
275aa857  =>  transferListingTokens(address,address,uint256,Listing)
7ed5052a  =>  payout(address,address,address,uint256,Listing)
cf5a7277  =>  validateERC20BalAndAllowance(address,address,uint256)
cbb4ec0c  =>
validateOwnershipAndApproval(address,address,uint256,uint256,TokenType)
fbf8d49b  =>  validateDirectListingSale(Listing,address,uint256,address,uint256)
03335cf6  =>  getSafeQuantity(TokenType,uint256)
93272baf  =>  getTokenType(address)
ea0e0241  =>  setAuctionBuffers(uint256,uint256)
4f6194a9  =>  setListPriceBpsIncrease(uint64)
f8c84539  =>  setOutsideListingAllowed(bool)
```

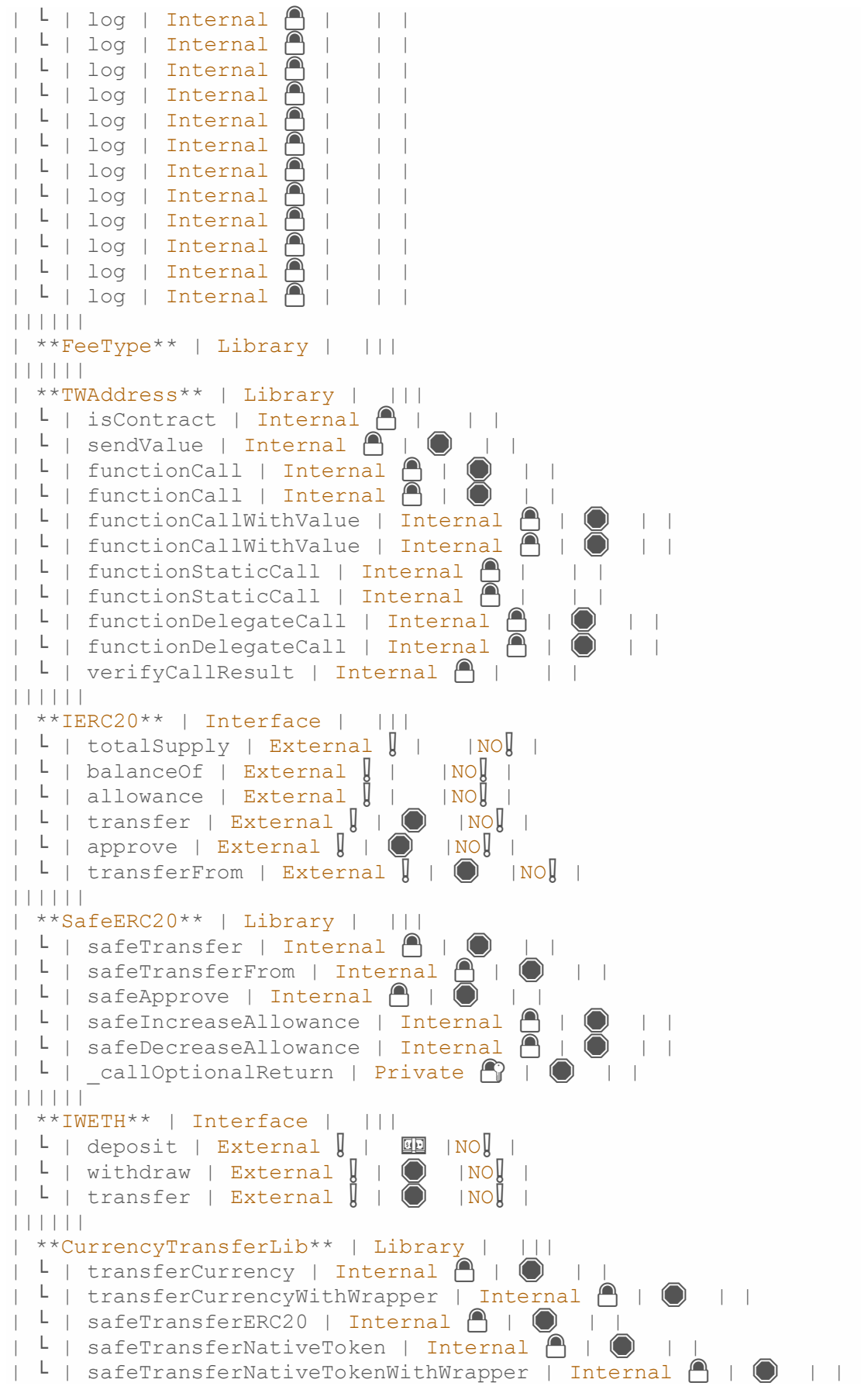
Automatic general report

```
Files Description Table

|  File Name  |  SHA-1 Hash  |
|-------------|--------------|
| /Users/macbook/Desktop/smart contracts/NFTreasuryMarketplace.sol | a7d54eefc331d0058be29119718cbeea46b5df55 |

 Contracts Description Table

|  Contract  |         Type        |       Bases      |                  |                 |
|:----------:|:-------------------:|:----------------:|:----------------:|:---------------:|
|     └      |  **Function Name**  |  **Visibility**  |  **Mutability**  |  **Modifiers**  |
||||||
| **console** | Library |  |||
| └ | _sendLogPayload | Private 🔐 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
| └ | log | Internal 🔒 |   | |
```


└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
└	log	Internal 🔒		
FeeType	Library			
TWAddress	Library			
└	isContract	Internal 🔒		
└	sendValue	Internal 🔒	🛑	
└	functionCall	Internal 🔒	🛑	
└	functionCall	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	functionStaticCall	Internal 🔒		
└	functionStaticCall	Internal 🔒		
└	functionDelegateCall	Internal 🔒	🛑	
└	functionDelegateCall	Internal 🔒	🛑	
└	verifyCallResult	Internal 🔒		
IERC20	Interface			
└	totalSupply	External ❗		NO❗
└	balanceOf	External ❗		NO❗
└	allowance	External ❗		NO❗
└	transfer	External ❗	🛑	NO❗
└	approve	External ❗	🛑	NO❗
└	transferFrom	External ❗	🛑	NO❗
SafeERC20	Library			
└	safeTransfer	Internal 🔒	🛑	
└	safeTransferFrom	Internal 🔒	🛑	
└	safeApprove	Internal 🔒	🛑	
└	safeIncreaseAllowance	Internal 🔒	🛑	
└	safeDecreaseAllowance	Internal 🔒	🛑	
└	_callOptionalReturn	Private 🔐	🛑	
IWETH	Interface			
└	deposit	External ❗	💵	NO❗
└	withdraw	External ❗	🛑	NO❗
└	transfer	External ❗	🛑	NO❗
CurrencyTransferLib	Library			
└	transferCurrency	Internal 🔒	🛑	
└	transferCurrencyWithWrapper	Internal 🔒	🛑	
└	safeTransferERC20	Internal 🔒	🛑	
└	safeTransferNativeToken	Internal 🔒	🛑	
└	safeTransferNativeTokenWithWrapper	Internal 🔒	🛑	

IPlatformFee	Interface			
└	getPlatformFeeInfo	External ❗️		NO❗️
└	setPlatformFeeInfo	External ❗️	🛑	NO❗️
IThirdwebContract	Interface			
└	contractType	External ❗️		NO❗️
└	contractVersion	External ❗️		NO❗️
└	contractURI	External ❗️		NO❗️
└	setContractURI	External ❗️	🛑	NO❗️
INFTreasuryMarketplace	Interface	IThirdwebContract, IPlatformFee		
└	createListing	External ❗️	🛑	NO❗️
└	updateListing	External ❗️	🛑	NO❗️
└	cancelDirectListing	External ❗️	🛑	NO❗️
└	buy	External ❗️	💵	NO❗️
└	offer	External ❗️	💵	NO❗️
└	acceptOffer	External ❗️	🛑	NO❗️
└	closeAuction	External ❗️	🛑	NO❗️
EnumerableSetUpgradeable	Library			
└	_add	Private 🔐	🛑	
└	_remove	Private 🔐	🛑	
└	_contains	Private 🔐		
└	_length	Private 🔐		
└	_at	Private 🔐		
└	_values	Private 🔐		
└	add	Internal 🔒	🛑	
└	remove	Internal 🔒	🛑	
└	contains	Internal 🔒		
└	length	Internal 🔒		
└	at	Internal 🔒		
└	values	Internal 🔒		
└	add	Internal 🔒	🛑	
└	remove	Internal 🔒	🛑	
└	contains	Internal 🔒		
└	length	Internal 🔒		
└	at	Internal 🔒		
└	values	Internal 🔒		
└	add	Internal 🔒	🛑	
└	remove	Internal 🔒	🛑	
└	contains	Internal 🔒		
└	length	Internal 🔒		
└	at	Internal 🔒		
└	values	Internal 🔒		
MathUpgradeable	Library			
└	max	Internal 🔒		
└	min	Internal 🔒		
└	average	Internal 🔒		
└	ceilDiv	Internal 🔒		
└	mulDiv	Internal 🔒		
└	mulDiv	Internal 🔒		
└	sqrt	Internal 🔒		
└	sqrt	Internal 🔒		


└	log2	Internal 🔒		
└	log2	Internal 🔒		
└	log10	Internal 🔒		
└	log10	Internal 🔒		
└	log256	Internal 🔒		
└	log256	Internal 🔒		
StringsUpgradeable	Library			
└	toString	Internal 🔒		
└	toHexString	Internal 🔒		
└	toHexString	Internal 🔒		
└	toHexString	Internal 🔒		
AddressUpgradeable	Library			
└	isContract	Internal 🔒		
└	sendValue	Internal 🔒	🛑	
└	functionCall	Internal 🔒	🛑	
└	functionCall	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	functionCallWithValue	Internal 🔒	🛑	
└	functionStaticCall	Internal 🔒		
└	functionStaticCall	Internal 🔒		
└	verifyCallResultFromTarget	Internal 🔒		
└	verifyCallResult	Internal 🔒		
└	_revert	Private 🔐		
Initializable	Implementation			
└	_disableInitializers	Internal 🔒	🛑	
└	_getInitializedVersion	Internal 🔒		
└	_isInitializing	Internal 🔒		
MulticallUpgradeable	Implementation	Initializable		
└	__Multicall_init	Internal 🔒	🛑	onlyInitializing
└	__Multicall_init_unchained	Internal 🔒	🛑	onlyInitializing
└	multicall	External ❗	🛑	NO❗
└	_functionDelegateCall	Private 🔐	🛑	
ReentrancyGuardUpgradeable	Implementation	Initializable		
└	__ReentrancyGuard_init	Internal 🔒	🛑	onlyInitializing
└	__ReentrancyGuard_init_unchained	Internal 🔒	🛑	onlyInitializing
└	_nonReentrantBefore	Private 🔐	🛑	
└	_nonReentrantAfter	Private 🔐	🛑	
ContextUpgradeable	Implementation	Initializable		
└	__Context_init	Internal 🔒	🛑	onlyInitializing
└	__Context_init_unchained	Internal 🔒	🛑	onlyInitializing
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
ERC2771ContextUpgradeable	Implementation	Initializable, ContextUpgradeable		
└	__ERC2771Context_init	Internal 🔒	🛑	onlyInitializing
└	__ERC2771Context_init_unchained	Internal 🔒	🛑	onlyInitializing
└	isTrustedForwarder	Public ❗		NO❗
└	_msgSender	Internal 🔒		

└	_msgData	Internal 🔒		
IAccessControlUpgradeable	Interface				
└	hasRole	External ❗		|NO❗
└	getRoleAdmin	External ❗		|NO❗
└	grantRole	External ❗	🛑	|NO❗
└	revokeRole	External ❗	🛑	|NO❗
└	renounceRole	External ❗	🛑	|NO❗
IAccessControlEnumerableUpgradeable	Interface	IAccessControlUpgradeable		
└	getRoleMember	External ❗		|NO❗
└	getRoleMemberCount	External ❗		|NO❗
IERC721ReceiverUpgradeable	Interface				
└	onERC721Received	External ❗	🛑	|NO❗
IERC165Upgradeable	Interface				
└	supportsInterface	External ❗		|NO❗
ERC165Upgradeable	Implementation	Initializable, IERC165Upgradeable		
└	__ERC165_init	Internal 🔒	🛑	onlyInitializing
└	__ERC165_init_unchained	Internal 🔒	🛑	onlyInitializing
└	supportsInterface	Public ❗		|NO❗
AccessControlUpgradeable	Implementation	Initializable, ContextUpgradeable, IAccessControlUpgradeable, ERC165Upgradeable		
└	__AccessControl_init	Internal 🔒	🛑	onlyInitializing
└	__AccessControl_init_unchained	Internal 🔒	🛑	onlyInitializing
└	supportsInterface	Public ❗		|NO❗
└	hasRole	Public ❗		|NO❗
└	_checkRole	Internal 🔒			
└	_checkRole	Internal 🔒			
└	getRoleAdmin	Public ❗		|NO❗
└	grantRole	Public ❗	🛑	onlyRole
└	revokeRole	Public ❗	🛑	onlyRole
└	renounceRole	Public ❗	🛑	|NO❗
└	_setupRole	Internal 🔒	🛑		
└	_setRoleAdmin	Internal 🔒	🛑		
└	_grantRole	Internal 🔒	🛑		
└	_revokeRole	Internal 🔒	🛑		
AccessControlEnumerableUpgradeable	Implementation	Initializable, IAccessControlEnumerableUpgradeable, AccessControlUpgradeable		
└	__AccessControlEnumerable_init	Internal 🔒	🛑	onlyInitializing
└	__AccessControlEnumerable_init_unchained	Internal 🔒	🛑	onlyInitializing
└	supportsInterface	Public ❗		|NO❗
└	getRoleMember	Public ❗		|NO❗
└	getRoleMemberCount	Public ❗		|NO❗
└	_grantRole	Internal 🔒	🛑		
└	_revokeRole	Internal 🔒	🛑		
IERC2981Upgradeable	Interface	IERC165Upgradeable		
└	royaltyInfo	External ❗		|NO❗

IERC1155ReceiverUpgradeable	Interface	IERC165Upgradeable		
└	onERC1155Received	External ❗	🔴 |NO❗
└	onERC1155BatchReceived	External ❗	🔴 |NO❗
IERC721Upgradeable	Interface	IERC165Upgradeable		
└	balanceOf	External ❗	|NO❗
└	ownerOf	External ❗	|NO❗
└	safeTransferFrom	External ❗	🔴 |NO❗
└	safeTransferFrom	External ❗	🔴 |NO❗
└	transferFrom	External ❗	🔴 |NO❗
└	approve	External ❗	🔴 |NO❗
└	setApprovalForAll	External ❗	🔴 |NO❗
└	getApproved	External ❗	|NO❗
└	isApprovedForAll	External ❗	|NO❗
IERC1155Upgradeable	Interface	IERC165Upgradeable		
└	balanceOf	External ❗	|NO❗
└	balanceOfBatch	External ❗	|NO❗
└	setApprovalForAll	External ❗	🔴 |NO❗
└	isApprovedForAll	External ❗	|NO❗
└	safeTransferFrom	External ❗	🔴 |NO❗
└	safeBatchTransferFrom	External ❗	🔴 |NO❗
IERC20Upgradeable	Interface			
└	totalSupply	External ❗	|NO❗
└	balanceOf	External ❗	|NO❗
└	transfer	External ❗	🔴 |NO❗
└	allowance	External ❗	|NO❗
└	approve	External ❗	🔴 |NO❗
└	transferFrom	External ❗	🔴 |NO❗
NFTreasuryMarketplace	Implementation	Initializable, INFTreasuryMarketplace, ReentrancyGuardUpgradeable, ERC2771ContextUpgradeable, MulticallUpgradeable, AccessControlEnumerableUpgradeable, IERC721ReceiverUpgradeable, IERC1155ReceiverUpgradeable		
└	<Constructor>	Public ❗	🔴	initializer
└	initialize	External ❗	🔴	initializer
└	setMainNFT	External ❗	🔴	onlyRole
└	<Receive Ether>	External ❗	💸 |NO❗
└	contractType	External ❗	|NO❗
└	contractVersion	External ❗	|NO❗
└	onERC1155Received	Public ❗	🔴 |NO❗
└	onERC1155BatchReceived	Public ❗	🔴 |NO❗
└	onERC721Received	External ❗	|NO❗
└	supportsInterface	Public ❗	|NO❗
└	createListing	External ❗	🔴 |NO❗
└	updateListing	External ❗	🔴	onlyRole
└	cancelDirectListing	External ❗	🔴	onlyRole
└	buy	External ❗	💸	nonReentrant onlyExistingListing
└	acceptOffer	External ❗	🔴	nonReentrant onlyListingCreator onlyExistingListing
└	executeSale	Internal 🔒	🔴	
└	offer	External ❗	💸	nonReentrant onlyExistingListing
└	handleOffer	Internal 🔒	🔴	

└	handleBid	Internal 🔒	🛑	
└	isNewWinningBid	Internal 🔒		
└	closeAuction	External ❗	🛑	nonReentrant onlyExistingListing
└	_cancelAuction	Internal 🔒	🛑	
└	_closeAuctionForAuctionCreator	Internal 🔒	🛑	
└	_closeAuctionForBidder	Internal 🔒	🛑	
└	transferListingTokens	Internal 🔒	🛑	
└	payout	Internal 🔒	🛑	
└	validateERC20BalAndAllowance	Internal 🔒		
└	validateOwnershipAndApproval	Internal 🔒		
└	validateDirectListingSale	Internal 🔒	🛑	
└	getSafeQuantity	Internal 🔒		
└	getTokenType	Internal 🔒		
└	getPlatformFeeInfo	External ❗		NO❗
└	setPlatformFeeInfo	External ❗	🛑	onlyRole
└	setAuctionBuffers	External ❗	🛑	onlyRole
└	setContractURI	External ❗	🛑	onlyRole
└	_msgSender	Internal 🔒		
└	_msgData	Internal 🔒		
└	setListPriceBpsIncrease	External ❗	🛑	onlyRole
└	setAuctionEnabled	External ❗	🛑	onlyRole
└	setOutsideListingAllowed	External ❗	🛑	onlyRole

Legend

Symbol	Meaning
🛑	Function can modify state
💵	Function is payable

Conclusion

The contracts are written systematically. Team found no critical issues. So, it is good to go for production.

Since possible test cases can be unlimited and developer level documentation (code flow diagram with function level description) not provided, for such an extensive smart contract protocol, we provide no such guarantee of future outcomes. We have used all the latest static tools and manual observations to cover maximum possible test cases to scan Everything.

Security state of the reviewed contract is “ Well Secured”.

✓ No volatile code.

✓ No high severity issues were found.

Disclaimer

This is a limited report on our findings based on our analysis, in accordance with good industry practice as of the date of this report, in relation to cybersecurity vulnerabilities and issues in the framework and algorithms based on smart contracts, the details of which are set out in this report. In order to get a full view of our analysis, it is crucial for you to read the full report. While we have done our best in conducting our analysis and producing this report, it is important to note that you should not rely on this report and cannot claim against the team on the basis of what it says or doesn't say, or how team produced it, and it is important for you to conduct your own independent investigations before making any decisions. team go into more detail on this in the below disclaimer below – please make sure to read it in full.

By reading this report or any part of it, you agree to the terms of this disclaimer. If you do not agree to the terms, then please immediately cease reading this report, and delete and destroy any and all copies of this report downloaded and/or printed by you. This report is provided for information purposes only and on a non-reliance basis, and does not constitute investment advice. No one shall have any right to rely on the report or its contents, and Saferico and its affiliates (including holding companies, shareholders, subsidiaries, employees, directors, officers and other representatives) (Saferico s) owe no duty of care towards you or any other person, nor does Saferico make any warranty or representation to any person on the accuracy or completeness of the report. The report is provided "as is", without any conditions, warranties or other terms of any kind except as set out in this disclaimer, and Saferico hereby excludes all representations, warranties, conditions and other terms (including, without limitation, the warranties implied by law of satisfactory quality, fitness for purpose and the use of reasonable care and skill) which, but for this clause, might have effect in relation to the report. Except and only to the extent that it is prohibited by law, Saferico hereby excludes all liability and responsibility, and neither you nor any other person shall have any claim against Saferico, for any amount or kind of loss or damage that may result to you or any other person (including without limitation, any direct, indirect, special, punitive, consequential or pure economic loss or damages, or any loss of income, profits, goodwill, data, contracts, use of money, or business interruption, and whether in delict, tort (including without limitation negligence), contract, breach of statutory duty, misrepresentation (whether innocent or negligent) or otherwise under any claim of any nature whatsoever in any jurisdiction) in any way arising from or connected with this report and the use, inability to use or the results of use of this report, and any reliance on this report. The analysis of the security is purely based on the smart contracts alone. No applications or operations were reviewed for security. No product code has been reviewed.