

DESSERT
FINANCE



Special Agent Squid (SAS)

ERC-721 Audit

Performed at block 14275964

PERFORMED BY DESSERT FINANCE
FOR CONTRACT ADDRESS: 0xdf45A29bFc064399E78dfA06d10d648Af2b9B639

INITIAL DISCLAIMER

Dessert Finance provides due-diligence project audits for various projects. Dessert Finance in no way guarantees that a project will not remove liquidity, sell off team supply, or otherwise exit scam.

Dessert Finance does the legwork and provides public information about the project in an easy-to-understand format for the common person.

Agreeing to an audit in no way guarantees that a team will not remove *all* liquidity (“Rug Pull”), remove liquidity slowly, sell off tokens, quit the project, or completely exit scam. There is also no way to prevent private sale holders from selling off their tokens. It is ultimately your responsibility to read through all documentation, social media posts, and contract code of each individual project to draw your own conclusions and set your own risk tolerance.

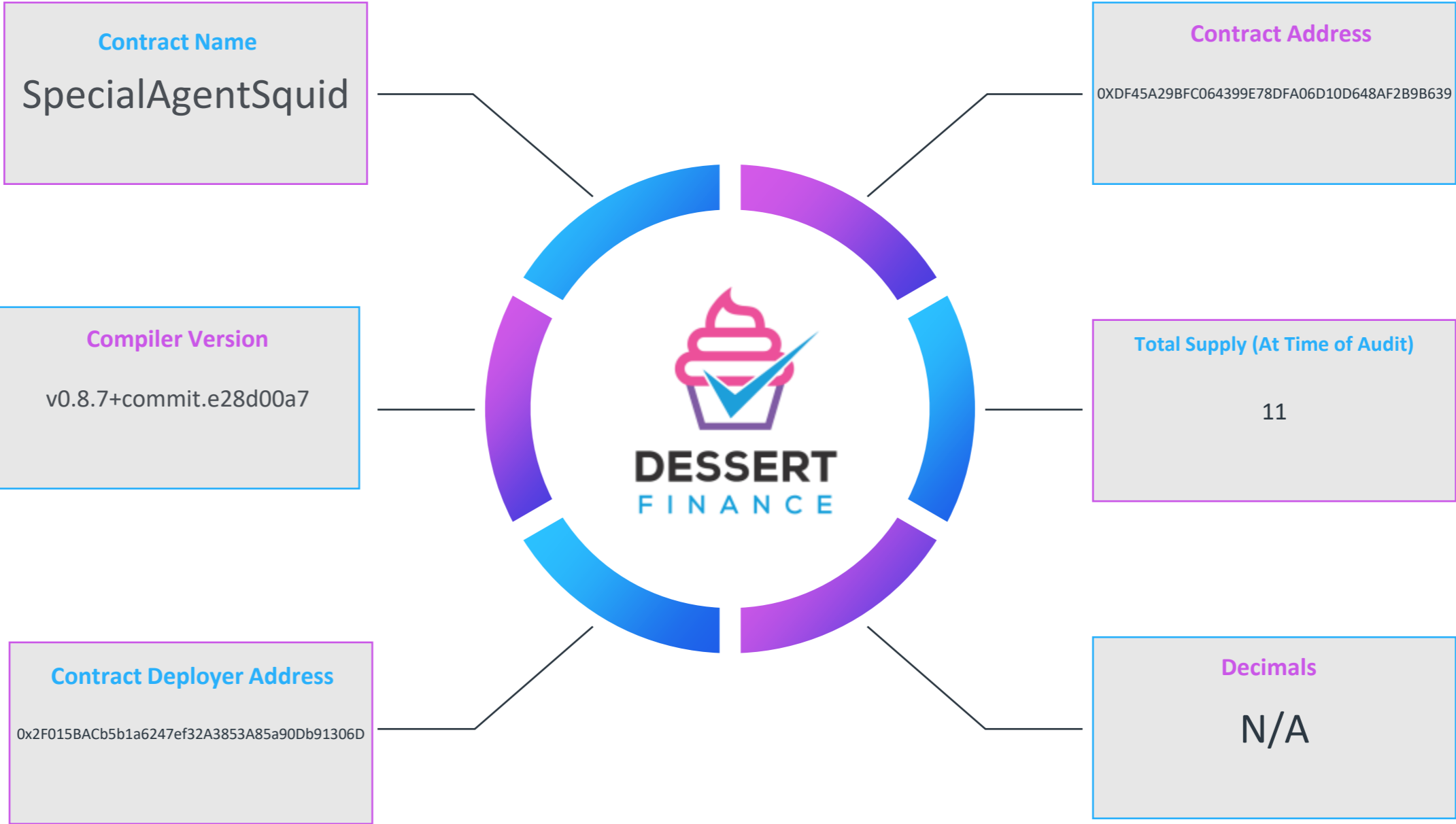
Dessert Finance in no way takes responsibility for any losses, nor does Dessert Finance encourage any speculative investments. The information provided in this audit is for information purposes only and should not be considered investment advice. Dessert Finance does not endorse, recommend, support, or suggest any projects that have been audited. An audit is an informational report based on our findings, We recommend you do your own research, we will never endorse any project to invest in.

Table of Contents



1. Contract Code Audit – Token Overview
2. ERC-721 Contract Code Audit – Overview
3. ERC-721 Contract Code Audit – Vulnerabilities Checked
4. Contract Code Audit – Contract Ownership
5. Contract Code Audit – Owner Accessible Functions
6. Liquidity Ownership – Locked / Unlocked
7. Contract Code Audit – Mint Functions
8. Contract Transaction Fees
9. Website Overview
10. Social Media
11. Top Token Holders/Wallets
12. Location Audit
13. Review of Team
14. Roadmap
15. Disclaimers

Contract Code Audit – Token Overview



ERC-721 (NFT) Contract Code Audit – Overview

Dessert Finance was commissioned to perform an audit on Special Agent Squid (SAS)

```
*Submitted for verification at Etherscan.io on 2021-12-22
**
// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

/**
 * @dev Interface of the ERC165 standard, as defined in the
 * https://eips.ethereum.org/EIPS/eip-165[ERC165].
 * Implementers can declare support of contract interfaces, which can then be
 * queried by others ({ERC165Checker}).
 * For an implementation, see {ERC165}.
 */
interface ERC165 {
    /**
     * @dev Returns true if this contract implements the interface defined by
     * `interfaceId`. See the corresponding
     * https://eips.ethereum.org/EIPS/eip-165#how-interfaces-are-identified[ERC section]
     * to learn more about how these IDs are created.
     */
    function supportsInterface(bytes4 interfaceId) external view returns (bool);
}

// File: @openzeppelin/contracts/token/ERC721/ERC721.sol
pragma solidity ^0.8.0;

/**
 * @dev Standard interface of an ERC721 compliant contract.
 */
interface ERC721 {
    /**
     * @dev Emitted when `tokenId` token is transferred from `from` to `to`.
     */
    event Transfer(address indexed from, address indexed to, uint256 indexed tokenId);

    /**
     * @dev Emitted when `owner` enables `approved` to manage the `tokenId` token.
     */
    event Approval(address indexed owner, address indexed approved, uint256 indexed tokenId);

    /**
     * @dev Emitted when `owner` enables or disables (`approved`) `operator` to manage all of its assets.
     */
    event ApprovalForAll(address indexed owner, address indexed operator, bool approved);

    /**
     * @dev Returns the number of tokens in `owner`'s account.
     */
    function balanceOf(address owner) external view returns (uint256 balance);

    /**
     * @dev Returns the owner of the `tokenId` token.
     */
}

```

Contract Address

0xdf45A29bFc064399E78dfA06d10d648Af2b9B639

TokenTracker

Special Agent Squid (SAS)

Contract Creator

0x2f015bacb5b1a6247ef32a3853a85a90db91306d

Source Code

Contract Source Code Verified

Contract Name

SpecialAgentSquid

Other Settings

default evmVersion, MIT

Compiler Version

v0.8.7+commit.e28d00a7

Optimization Enabled

Yes with 200 runs

Code is truncated to fit the constraints of this document.

[The code in its entirety can be viewed here.](#)

The contract code is **verified** on Etherscan.

ERC-721 Contract Code Audit – Vulnerabilities Checked

Vulnerability Tested	AI Scan	Human Review	Result
Compiler Errors	Complete	Complete	✓ Low / No Risk
Outdated Compiler Version	Complete	Complete	✓ Low / No Risk
Integer Overflow	Complete	Complete	✓ Low / No Risk
Integer Underflow	Complete	Complete	✓ Low / No Risk
Correct Token Standards Implementation	Complete	Complete	✓ Low / No Risk
Timestamp Dependency for Crucial Functions	Complete	Complete	✓ Low / No Risk
Exposed _Transfer Function	Complete	Complete	✓ Low / No Risk
Transaction-Ordering Dependency	Complete	Complete	✓ Low / No Risk
Unchecked Call Return Variable	Complete	Complete	✓ Low / No Risk
Use of Deprecated Functions	Complete	Complete	✓ Low / No Risk
Unprotected SELFDESTRUCT Instruction	Complete	Complete	✓ Low / No Risk
State Variable Default Visibility (x1)	Complete	Complete	✓ Low Risk
Deployer Can Access User Funds	Complete	Complete	✓ Low / No Risk

The contract code is **verified** on EtherScan.

The vulnerabilities listed above were not found in the token's Smart Contract.

Contract Code Audit – Contract Ownership

Contract Ownership has not been renounced at the time of Audit



The contract ownership is not currently renounced.

We have placed the contract owner address below for your viewing:

[0x2f015bacb5b1a6247ef32a3853a85a90db91306d](https://www.etherscan.io/address/0x2f015bacb5b1a6247ef32a3853a85a90db91306d)

The address above has authority over the ownable functions within the contract.

This allows the owner to call certain functions within the contract. Any compromise to the owner wallet may allow these privileges to be exploited.

We recommend:

- Establishing a Time-Lock with reasonable latency
- Assignment of privileged roles to multi-signature wallets

Contract Code Audit – Owner Accessible Functions

Function Name	Parameters	Visibility	Audit Notes
reveal		public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setCost	uint256 _newCost	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setMaxMintAmount	uint256 _newmaxMintAmount	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setNotRevealedURI	string memory _notRevealedURI	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setBaseURI	string memory _newBaseURI	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setBaseExtension	string memory _newBaseExtension	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
pause	bool _state	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
withdraw		public payable	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
renounceOwnership		public virtual	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
transferOwnership	address newOwner	public virtual	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.

The functions listed above can be called by the contract owner. Notes may or may not be available for

If contract ownership has been renounced there is no way for the above listed functions to be called.

Liquidity Ownership – Locked / Unlocked

No locked liquidity information has been found.



This page will contain links to locked liquidity for the project if we are able to locate that information. Locked liquidity information was not found on the project's website.

Contract Code Audit – Mint Functions

This Contract Can Mint New SAS Tokens (NFTs).



We do understand that sometimes mint functions are essential to the functionality of the project.

A mint function was found in the contract code.

Website Part 1 – Overview

www.specialagentsquid.com



Above images are actual snapshots of the current live website of the project.

Website was registered on 12/06/2021, registration expires 12/06/2022.

X This does not meet the 3 year minimum we like to see on new projects.



Website Part 2 – Checklist



- ✓ Mobile Friendly
- ✓ No JavaScript Errors
- ✓ Spell Check
- ✓ SSL Certificate

The website contained no JavaScript errors. No typos, or grammatical errors were present, and we found a valid SSL certificate allowing for access via https.

No additional issues were found on the website.

Website Part 3 – Responsive HTML5 & CSS3

No issues were found on the Mobile Friendly check for the website. All elements loaded properly and browser resize was not an issue. The team has put a considerable amount of thought and effort into making sure their website looks great on all screens.

No severe JavaScript errors were found. No issues with loading elements, code, or stylesheets.



Website Part 4 (GWS) – General Web Security



SSL CERTIFICATE

A valid SSL certificate was found. Details are as follows:

Offered to: specialagentsquid.com

Issued by: Sectigo RSA

Valid Until: 12/09/2022



CONTACT EMAIL

A valid contact email was found on the official website. Contact email is listed as shown below:

Contact

N/A



SPAM / MALWARE / POPUPS

No malware found

No injected spam found

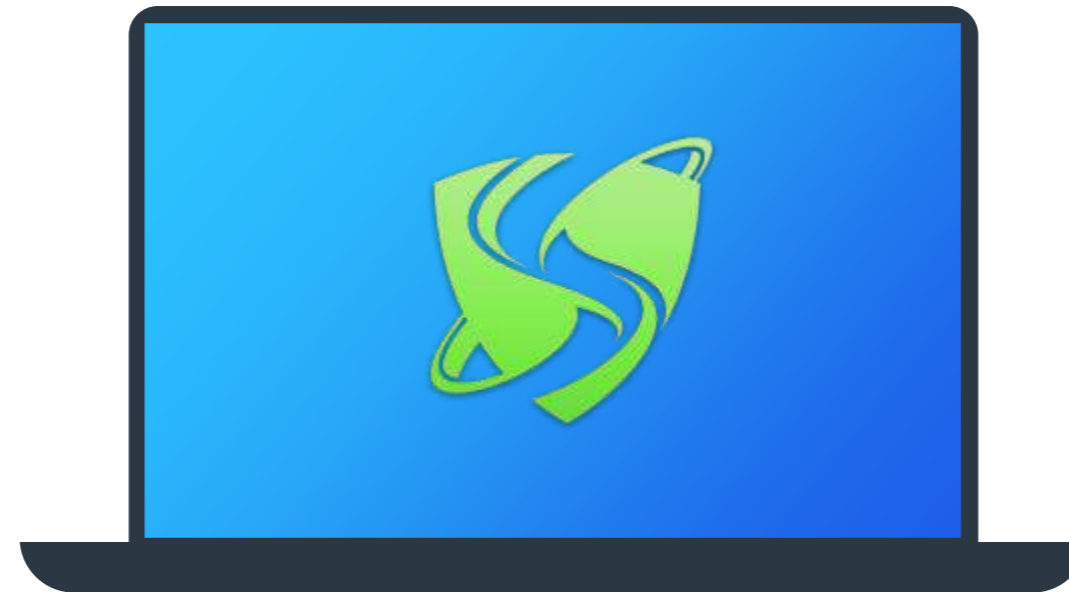
No internal server errors

No popups found

Domain is marked clean by Google, McAfee, Sucuri Labs, & ESET



Social Media



We were able to locate a variety of Social Media networks for the project.

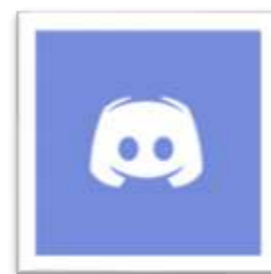
All links have been conveniently placed below.



[Twitter](#)



[Reddit](#)



[Discord](#)



[Facebook](#)



[Instagram](#)

✓ **At least 3 social media networks were found.**

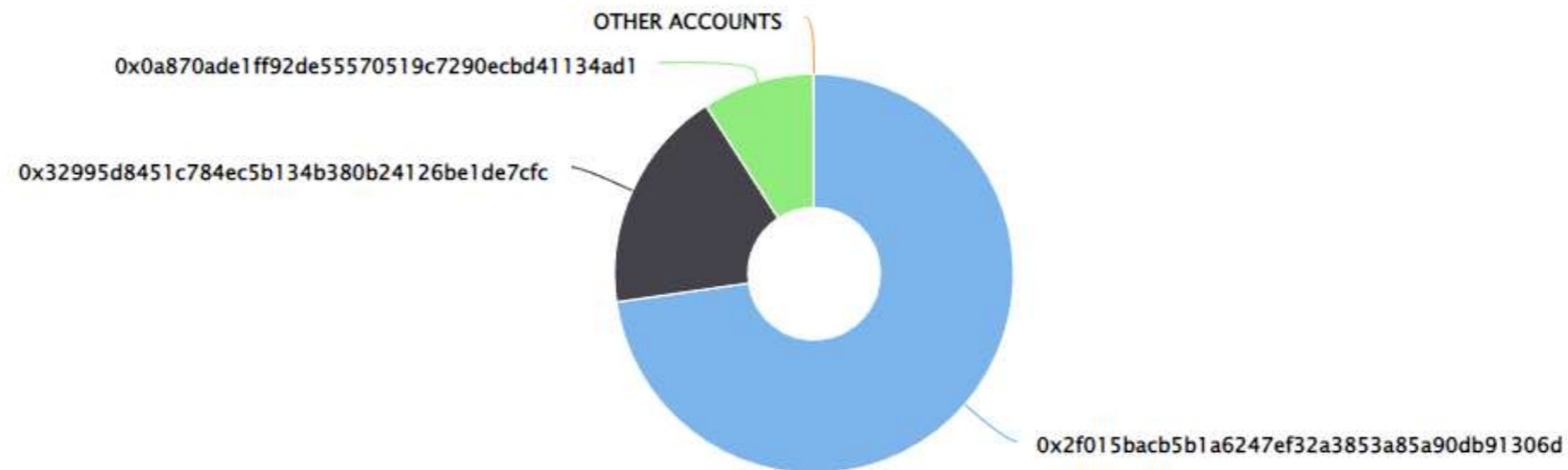
Top Token Holders

The entire supply was in three wallets at the time of audit. We expect this to change as the project goes through initial distribution phases. Please use the link below to view the most up-to-date holder information.

[Click here to view the most up-to-date list of holders](#)

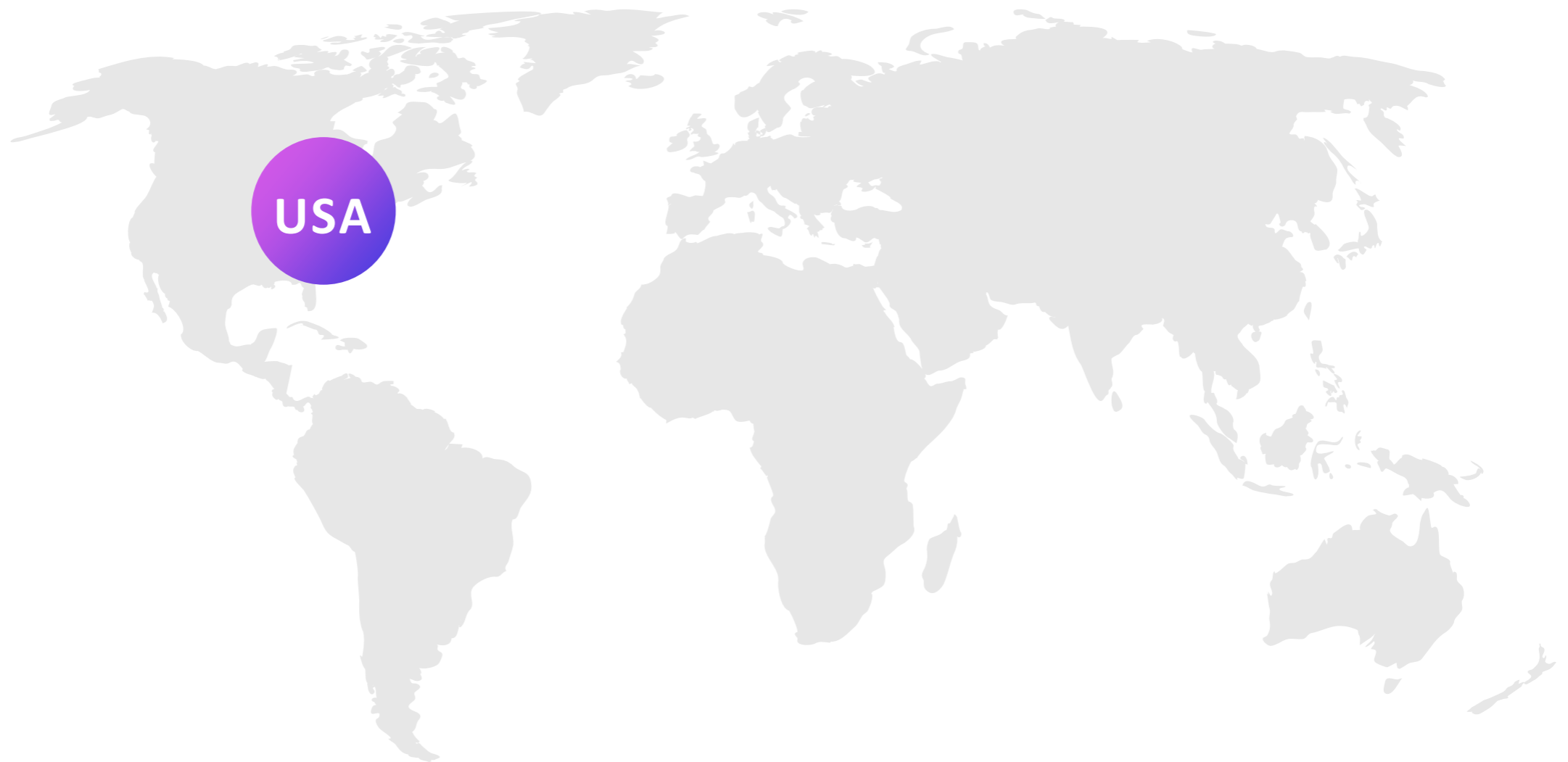
Special Agent Squid Top 100 Token Holders

Source: Etherscan.io



Location Audit

We were able to identify the primary location of the project as Chapin, South Carolina, USA.



Team Overview

Team information was found on the projects website and is shown below.

***Special Agent Squids proudly brought to
you by Midlands Blockchain
Technologies LLC.***

We are a group of Healthcare professionals working every day in the hospital taking care of patients. We all have various leadership and business experiences that drove us to starting Midlands Blockchain Technologies. Our goal was to start with validator nodes on several blockchains, running our own stake pools, and mining for certain cryptocurrencies. We hoped to raise enough revenue to start our work towards making dApps for Healthcare on the blockchain, specifically tackling some of the world's most prevalent problems when it comes to medical record access. NFTs are the future for digital ownership and access and Special Agent Squids was something we thought would be fun and support our future endeavors as a Healthcare blockchain company. We have a lot of cool things planned with this NFT project and it's sequel. We are real people with real aspirations and are fully doxxed. We're driven to help others in everything we do and would be grateful for your support!






Nick (Doc) Ryan (Nurse) Wes (MRI Tech) Mark (Hospital IT) Bryan (Healthcare IT Analyst)

Roadmap

A roadmap was found on the official website, we have conveniently placed it on this page for your viewing.

Rewards Roadmap

We've set up some reward thresholds and we are awarding over \$250,000 in prizes. Once we reach a target sales goal, we will begin to reward our community. Each sales target is set to reward squid owners and our charitable cause.

- 100 squids sold** – The first 100 squid owners will receive a limited run Special Agent Squid 3D challenge coin. 
- 500 squids sold** – Drawing for 10 lucky squid owners to win a custom plasma cut metal poster of their unique Special Agent Squid. The S.A.S. Headquarters will be unlocked.
- 1000 squids sold** – Drawing for 2 lucky squid owners to receive 1 ETH each.  Agents gain access to an exclusive members only water cooler where they can collect important mission information.
- 2500 squids sold** – 1 lucky squid owner will win a fully loaded Alienware R7 gaming PC. 
- 5000 squids sold** – Midlands Blockchain Technologies will write a \$100,000 check to the Coastal Studies Institute. We will also unlock our online store and award 1000 squid owners with a Special Agent Squid t-shirt. 
- 7500 squids sold** – Drawing for two lucky squid owners to win 2.5 ETH each.  The S.A.S. Headquarters will be expanded into a metaverse location.
- 10,000 squids sold** – Midlands Blockchain Technology will write a second check for \$100,000 to the Coastal Studies Institute for a total charitable contribution of \$200,000.

GRAND PRIZE

Upon reaching our 10,000 squid goal, one super lucky squid owner will win a **Tesla Model 3** dual motor AWD with full driving autonomy with tag, taxes, title and delivery fee paid. \$65,000 approximate value.

*Grand prize only available to winner in the contiguous lower 48 U.S. states. Cash option available for single \$65,000 payout in ETH.

Disclaimer



The opinions expressed in this document are for general informational purposes only and are **not intended to provide specific advice or recommendations for any individual or on any specific investment**. It is only intended to provide education and public knowledge regarding projects. This audit is only applied to the type of auditing specified in this report and the scope of given in the results. Other unknown security vulnerabilities are beyond responsibility. Dessert Finance only issues this report based on the attacks or vulnerabilities that already existed or occurred before the issuance of this report. For the emergence of new attacks or vulnerabilities that exist or occur in the future, Dessert Finance lacks the capability to judge its possible impact on the security status of smart contracts, thus taking no responsibility for them. The smart contract analysis and other contents of this report are based solely on the documents and materials that the contract provider has provided to Dessert Finance or was publicly available before the issuance of this report (issuance of report recorded via block number on cover page), if the documents and materials provided by the contract provider are missing, tampered, deleted, concealed or reflected in a situation that is inconsistent with the actual situation, or if the documents and materials provided are changed after the issuance of this report, Dessert Finance assumes no responsibility for the resulting loss or adverse effects. Due to the technical limitations of any organization, this report conducted by Dessert Finance still has the possibility that the entire risk cannot be completely detected. Dessert Finance disclaims any liability for the resulting losses.

Dessert Finance provides no guarantees against the sale of team tokens or the removal of liquidity by the project audited in this document. Even projects with a low risk score have been known to pull liquidity, sell all team tokens, or exit-scam. Please exercise caution when dealing with any cryptocurrency related platforms.

The final interpretation of this statement belongs to Dessert Finance.

Dessert Finance highly advises against using cryptocurrencies as speculative investments and they should be used solely for the utility they aim to provide.



Thank You

DESSERT FINANCE PROJECT AUDIT HAS BEEN COMPLETED FOR SPECIAL AGENT SQUID (SAS) AT BSC BLOCK NUMBER: **14275964**
THIS AUDIT IS ONLY VALID IF VIEWED ON [HTTPS://WWW.DSSERTSWAP.FINANCE](https://www.dessertswap.finance)

www.dessertswap.finance
<https://t.me/dessertswap>