DESSERT FINANCE

DCHESS

BEP-20 Audit

Performed at block **11514019**

PERFORMED BY DESSERT FINANCE

FOR CONTRACT ADDRESS: 0x461485e4589D38E649AC830BF21B46fDe03FF256

FOR CONTRACT ADDRESS: 0xcf2D2CE89AeD0073540C497fcF894Ea22d37C7aF

FOR CONTRACT ADDRESS: 0x8585A95B08d754b39fD9495C7a10D0f931109dbd

INITIAL DISCLAIMER

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

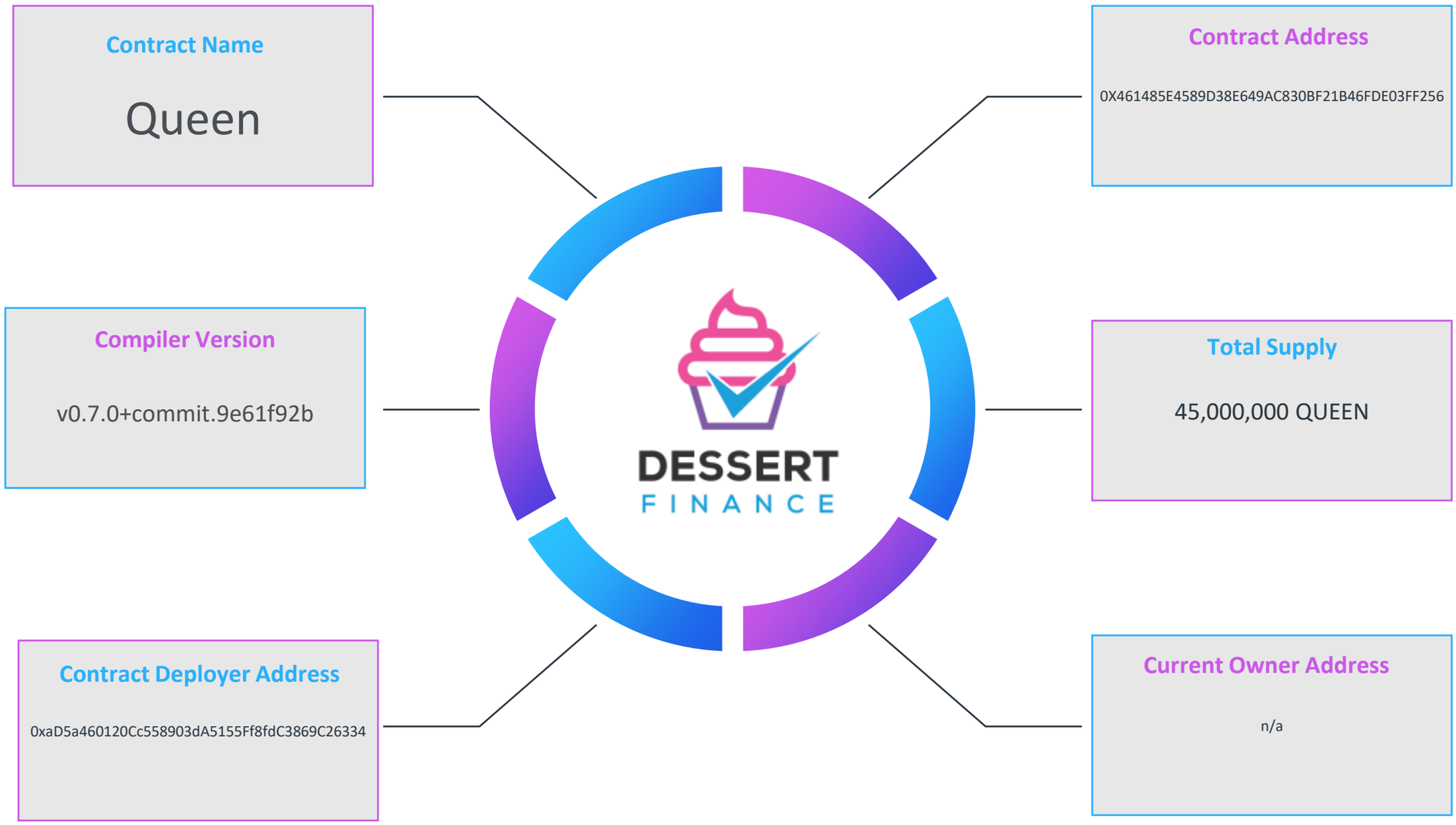
Dessert Finance does the legwork and provides public information about the project in an easy-to-understand format for the common person.

Agreeing to a project audit can be seen as a sign of confidence and is generally the first sign of trust for a project, but in no way guarantees that a team will not remove ***all*** liquidity (“Rug Pull”), sell off tokens, 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.

Table of Contents

Contract Code Audit – Token Overview
QUEEN – Official Governance Token



BEP-20 Contract Code Audit – Overview

Dessert Finance was commissioned to perform an audit on Queen (QUEEN)

```
pragma solidity ^0.7.0;

import "@openzeppelin/contracts/access/AccessControl.sol";
import "@openzeppelin/contracts/token/ERC20/ERC20.sol";
import "@openzeppelin/contracts/utils/Context.sol";

contract Queen is Context, AccessControl, ERC20{

  using SafeMath for uint256;

  bytes32 public constant MINTER_ROLE = keccak256("MINTER_ROLE");

  uint256 private maxSupply = 100000000 * 10 ** 18;

  //REAL ADDRESS
  address public constant TEAM_ADDRESS = 0xf6FB62c2F1C8cFA1b546b81DfG3A96B5FAa60F1e;
  address public constant INVESTOR_ADDRESS = 0x0aDD67cb0bBB6d8FA026d0ee2954e4CB45937954;
  address public constant ADVISORS_ADDRESS = 0x48b9EEF5241EE20CE4708D1Cc24319d756c516A3;
  address public constant ECOSYSTEM_ADDRESS = 0x923E20d45CAedcA7235c85a869870a1812D3447C;
  address public constant IDO_ADDRESS = 0x4b0dBcbAc44636d3a46e93A0Af12aCe7B2F1d25b;
  address public constant AIRDROP_ADDRESS = 0x1bf2E7b695706FA542c21F8CF30dc42B23790744;


  constructor(
    string memory name,
    string memory symbol
  ) public ERC20(name,symbol) {

    //ROLE DEFINITION
    _setupRole(DEFAULT_ADMIN_ROLE, _msgSender());
    _setupRole(MINTER_ROLE, _msgSender());

    mint(TEAM_ADDRESS, maxSupply.div(100).mul(10));
    mint(INVESTOR_ADDRESS, maxSupply.div(100).mul(5));
    mint(ADVISORS_ADDRESS, maxSupply.div(100).mul(10));
    mint(ECOSYSTEM_ADDRESS, maxSupply.div(100).mul(10));
    mint(IDO_ADDRESS, maxSupply.div(100).mul(7));
    mint(AIRDROP_ADDRESS, maxSupply.div(100).mul(3));
  }

  function mint(address to, uint256 amount) public virtual {
    require(
      hasRole(MINTER_ROLE, _msgSender()),
      "THIS USER DOES NOT HAVE MINTER_ROLE");

    require(
      totalSupply().add(amount) <= maxSupply,
      "MAXSUPPLY EXCEEDED");

    _mint(to, amount);
  }

  function burn(uint256 amount) public virtual {
    _burn(_msgSender(), amount);
  }
```

Contract Address
0x461485e4589D38E649AC830BF21B46fDe03FF256

TokenTracker
Queen (QUEEN)

Contract Creator
0xad5a460120cc558903da5155ff8fdc3869c26334

Source Code
Contract Source Code Verified

Contract Name
Queen

Other Settings
default evmVersion

Compiler Version
v0.7.0+commit.9e61f92b

Optimization Enabled
No 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 BSCScan.

BEP-20 Contract Code Audit – Vulnerabilities Checked

Vulnerability Tested	AI Scan	Human Review	Result
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	Complete	Complete	✓ Low / No Risk
Deployer Can Access User Funds	Complete	Complete	✓ Low / No Risk

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

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

Queen.sol

BEP-20 Contract Code Audit – Overview

Dessert Finance was commissioned to perform an audit on King (KING)

Contract Address
0xcf2D2CE89AeD0073540C497fcF894Ea22d37C7aF

TokenTracker
King (KING)

Contract Creator
0xad5a460120cc558903da5155ff8fdc3869c26334

Source Code
Contract Source Code Verified

Contract Name
King

Other Settings
default evmVersion

Compiler Version
v0.7.0+commit.9e61f92b

Optimization Enabled
No 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 BSCScan.

BEP-20 Contract Code Audit – Vulnerabilities Checked

Vulnerability Tested	AI Scan	Human Review	Result
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	Complete	Complete	✓ Low / No Risk
Deployer Can Access User Funds	Complete	Complete	✓ Low / No Risk

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

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

King.sol

BEP-20 Contract Code Audit – Overview

Dessert Finance was commissioned to perform an audit on DChess Boards (BOARD)

Contract Address
0x8585A95B08d754b39fD9495C7a10D0f931109dbd

TokenTracker
DChess Boards (BOARD)

Contract Creator
0xad5a460120cc558903da5155ff8fdc3869c26334

Source Code
Contract Source Code Verified

Contract Name
Board

Other Settings
default evmVersion

Compiler Version
v0.7.0+commit.9e61f92b

Optimization Enabled
No 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 BSCScan.

BEP-20 Contract Code Audit – Vulnerabilities Checked

Vulnerability Tested	AI Scan	Human Review	Result
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	Complete	Complete	✓ Low / No Risk
Deployer Can Access User Funds	Complete	Complete	✓ Low / No Risk

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

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

Board.sol

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.

Contract Code Audit – Mint Functions

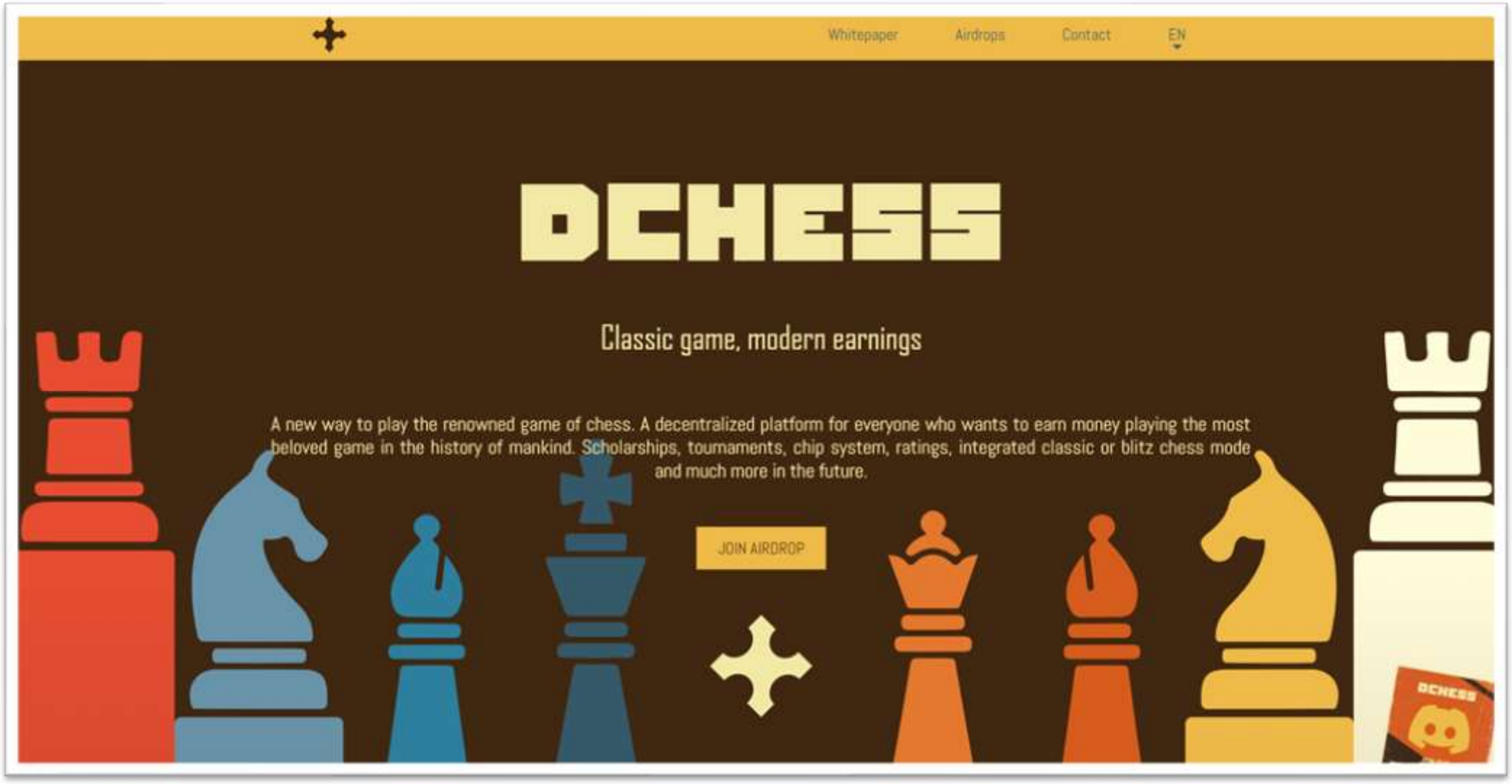
This Contract Cannot Mint New Queen Tokens.

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

A mint function was not found in the contract code.

Website Part 1 – Overview

www.dchess.net

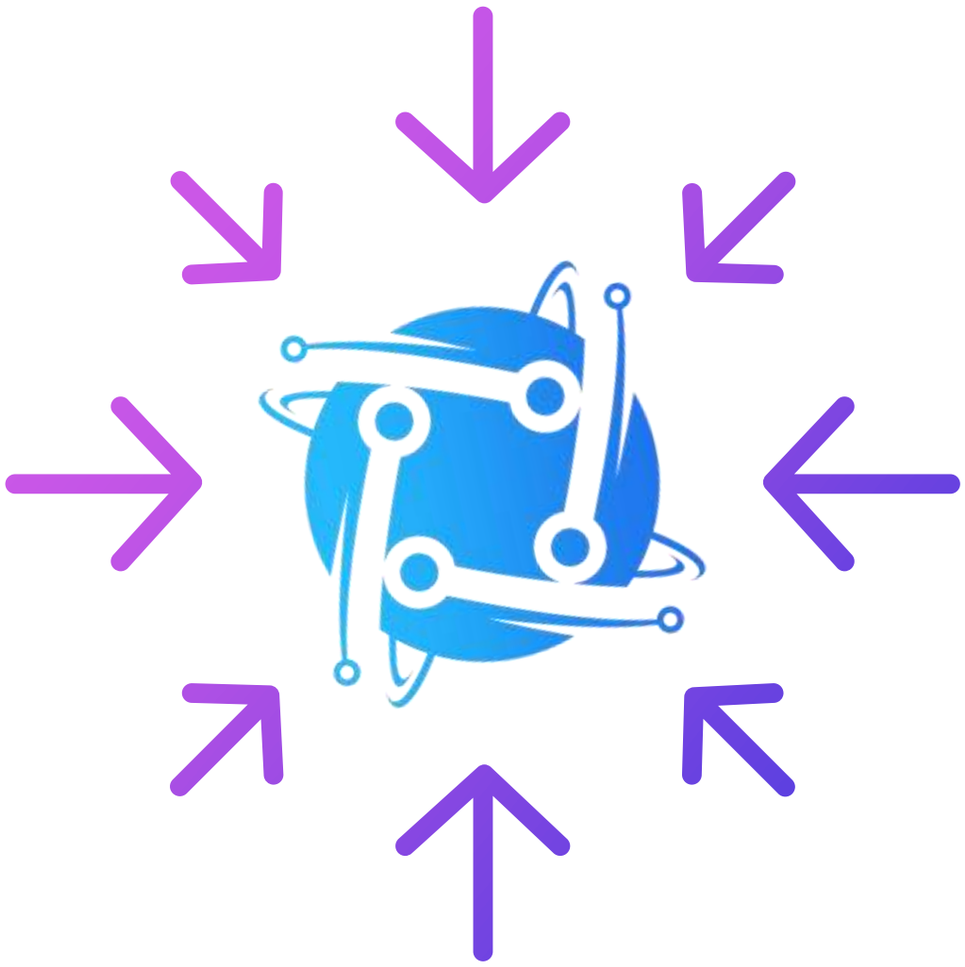


Above images are actual snapshots of the current live website of the project.

Website was registered on 09/03/2021, registration expires 09/03/2023.

✗ 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

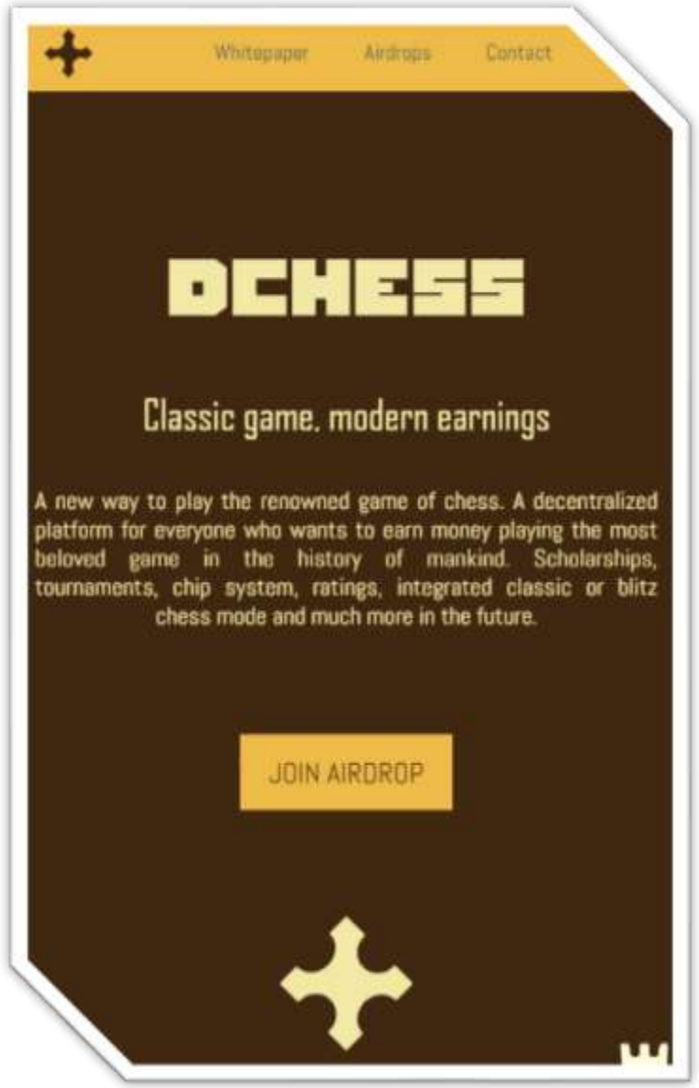
The website contained no JavaScript errors. No typos, or grammatical errors were present, and we found a valid SSL certificate allowing for access via https.

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: dchess.net

Issued by: Cloudflare Inc.

Valid Until: 09/16/2022

✓

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.

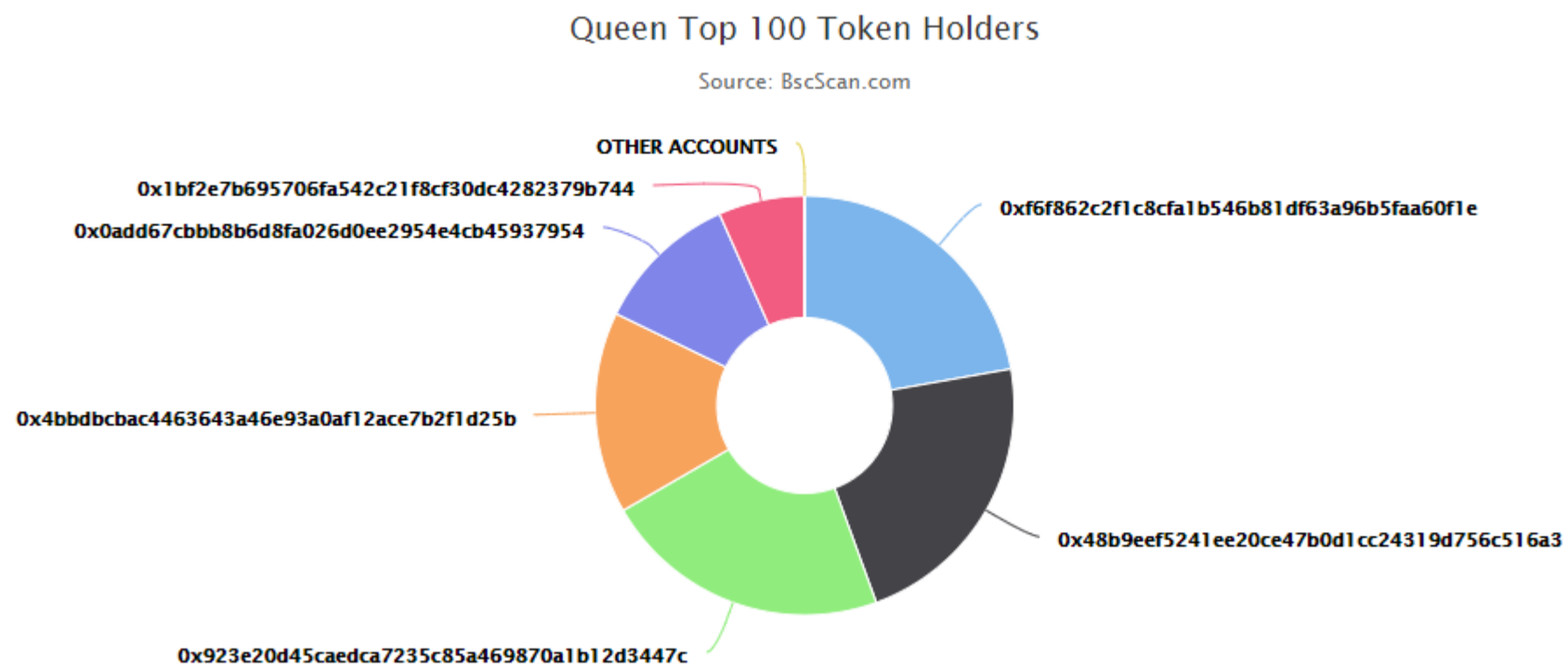
Twitter Telegram Discord Medium

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

Top Token Holders

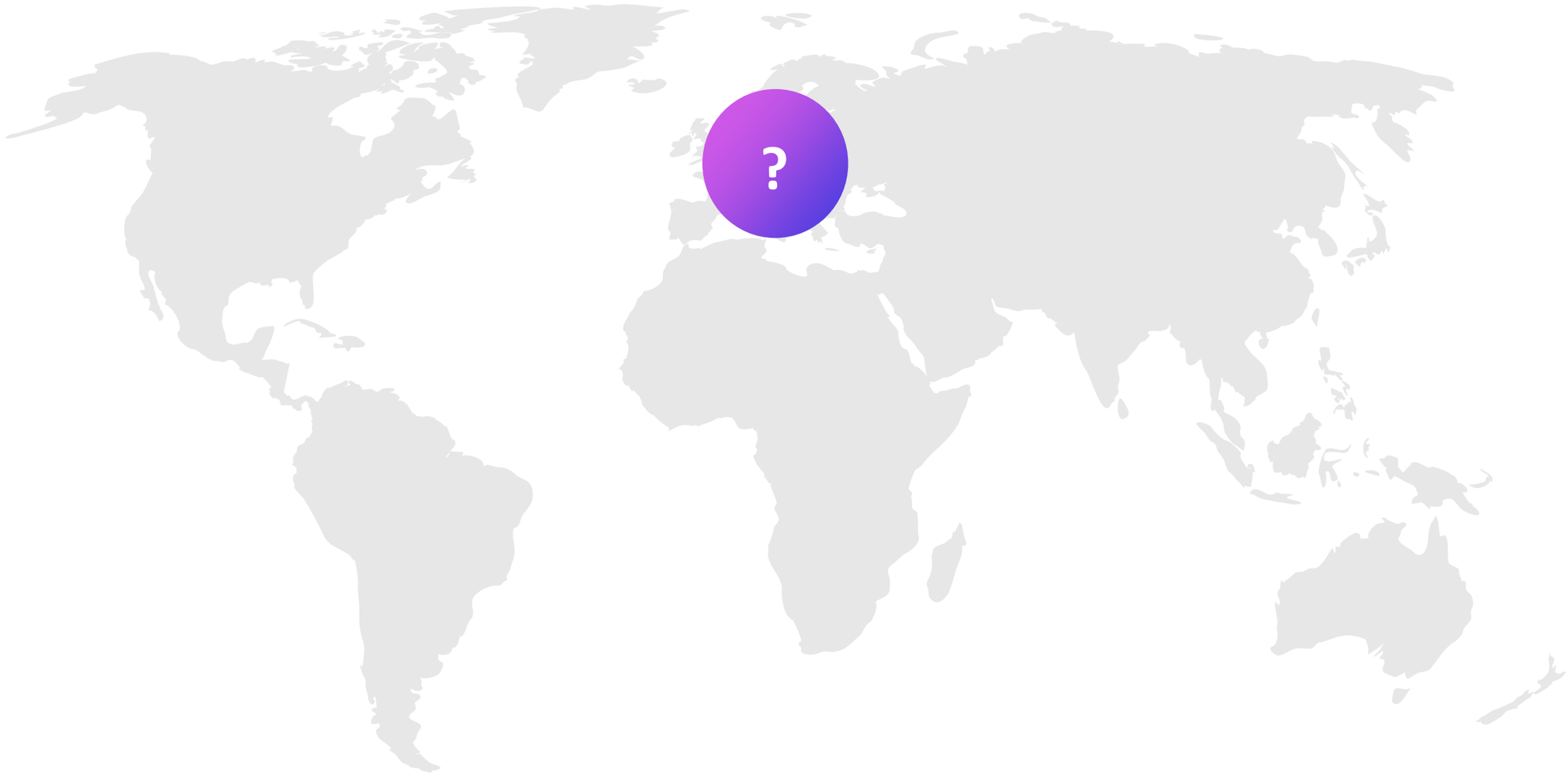
The entire supply was in a few wallet at the time of audit. We expect this to change as the project goes through initial distribution phases. Please use the link below to view the most up-to-date holder information.

Click here to view the most up-to-date list of holders



Location Audit

We were unable to identify a primary location for the project at this time or a location has not been declared.



Team Overview

Team information was found on the website and has been shown below.
Social media profiles are linked on the project's website.

Roadmap

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

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 BSC 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 DCHESS 1 DSRT HAS BEEN SENT TO AUDITED PROJECT'S CONTRACT ADDRESS FOR VERIFICATION OF THIS AUDIT AT BLOCK NUMBER: **11514019**

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