Kaiju Worlds (Kaiju)

BEP-20 Audit

Performed at block **10428290**

PERFORMED BY DESSERT FINANCE
FOR CONTRACT ADDRESS: 0x6fB9D47EA4379CcF00A7dcb17E0a2C6C755a9b4b

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

Website Part 1 – Overview

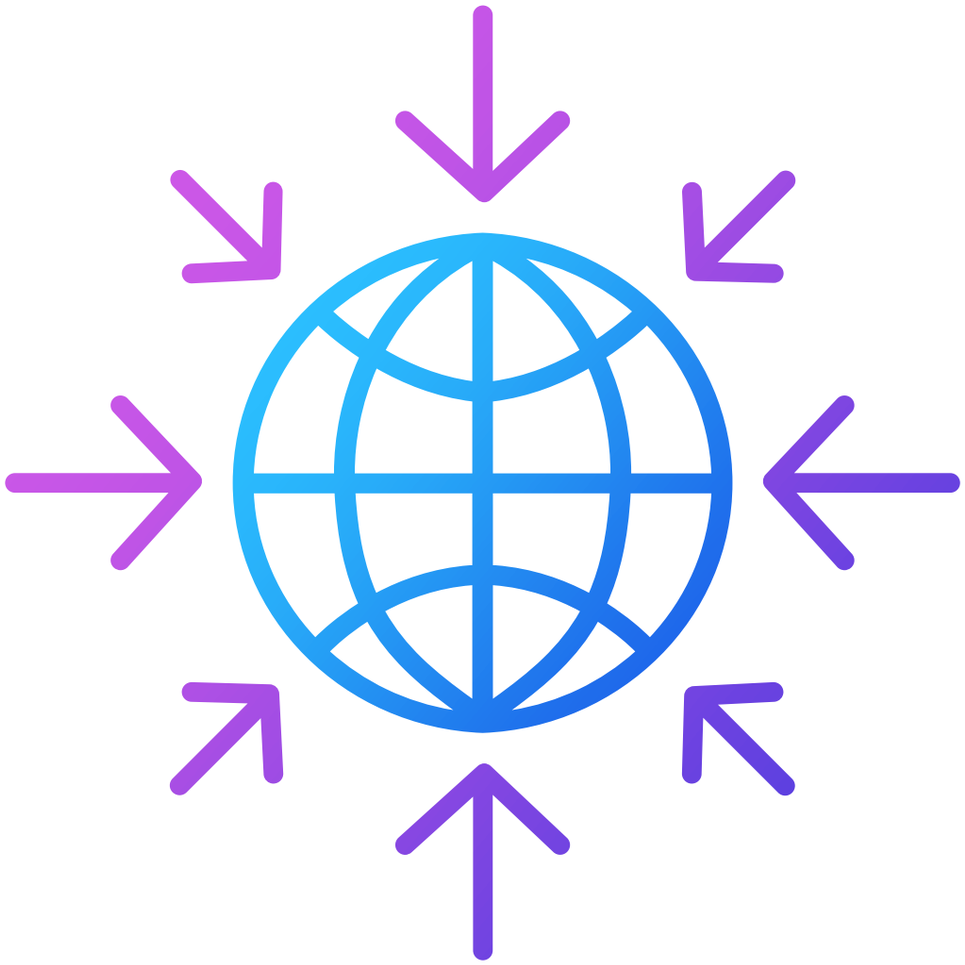
www.kaijuworlds.io

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

Website was registered on 06/08/2021, registration expires 06/08/2024, previously 06/08/2022.

✓ This meets the 3 year minimum we like to see on new projects.

Website Part 2 – Checklist



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

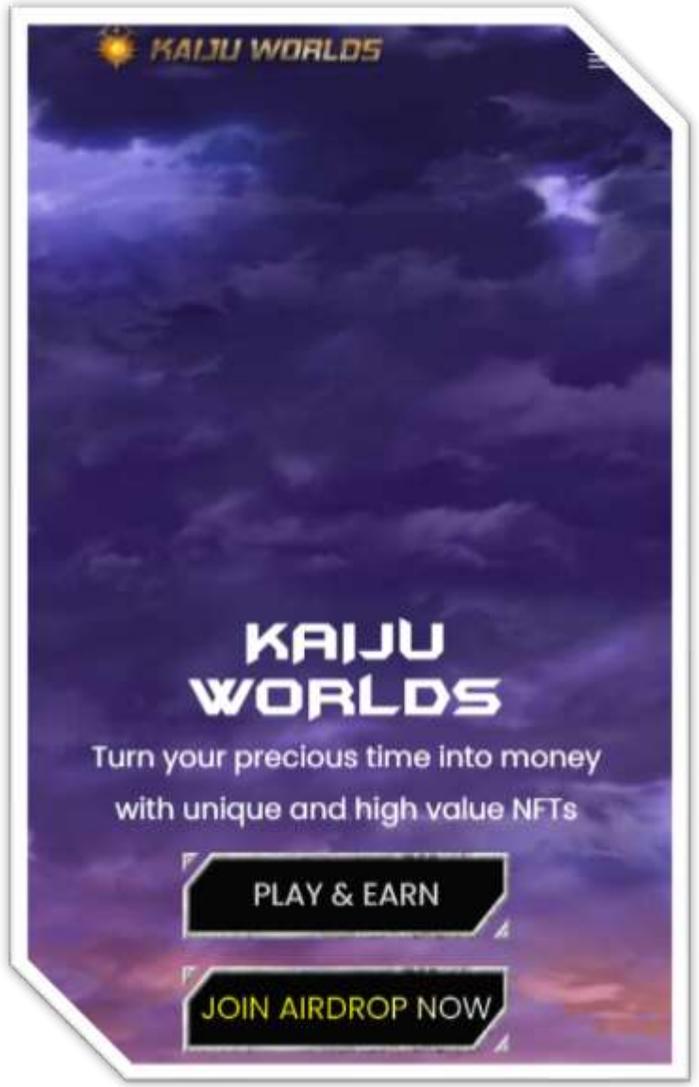
The website contained no JavaScript errors. No typos, or grammatical errors were present, and we found a valid SSL certificate allowing for access via https.

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 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: kaijuworlds.io

Issued by: Cloudflare, Inc

Valid Until: 08/06/2022

✓

CONTACT EMAIL

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

Contact

support@kaijuworlds.io

✓

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

✓

BEP-20 Contract Audit – Common Vulnerabilities - KaijuToken.sol

```
import "@openzeppelin/contracts/token/ERC20/IERC20.sol";
import "@openzeppelin/contracts/access/Ownable.sol";
import "@openzeppelin/contracts/utils/math/SafeMath.sol";

contract KaijuToken is Context, IERC20, Ownable {
  using SafeMath for uint256;

  mapping(address => uint256) private _balances;

  mapping(address => mapping(address => uint256)) private _allowances;

  uint256 private _totalSupply;
  uint8 public _decimals;
  string public _symbol;
  string public _name;

  constructor() public {
      _name = "Kaiju Worlds";
      _symbol = "Kaiju";
      _decimals = 18;
      _totalSupply = 0;
      _balances[msg.sender] = _totalSupply;

      emit Transfer(address(0), msg.sender, _totalSupply);
  }

  /**
   * @dev Returns the bep token owner.
   */
  function getOwner() external view returns (address) {
      return owner();
  }

  /**
   * @dev Returns the token decimals.
   */
  function decimals() external view returns (uint8) {
      return _decimals;
  }

  /**
   * @dev Returns the token symbol.
   */
  function symbol() external view returns (string memory) {
      return _symbol;
  }

  /**
   * @dev Returns the token name.
   */
  function name() external view returns (string memory) {
      return _name;
  }

  /**
   * @dev See {BEP20-totalSupply}.
   */
  function totalSupply() external view override returns (uint256) {
      return _totalSupply;
  }

  /**
   * @dev See {BEP20-balanceOf}.
   */
```

- ✓ **Integer Underflow**
- ✓ **Integer Overflow**
- ✓ **Correct Token Standards Implementation**
- ✓ **Timestamp Dependency for Randomness**
- ✓ **Unexposed Private Transfer Function**
- ✓ **Transaction-Ordering Dependency**

Code is truncated to fit the constraints of this document.

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

Common vulnerabilities were not found in the token's Smart Contract as shown above.

BEP-20 Contract Audit – Common Vulnerabilities - Marketplace.sol

Code is truncated to fit the constraints of this document.

- ✓ **Integer Underflow**
- ✓ **Integer Overflow**
- ✓ **Correct Token Standards Implementation**
- ✓ **Timestamp Dependency for Randomness**
- ✓ **Unexposed Private Transfer Function**
- ✓ **Transaction-Ordering Dependency**

Common vulnerabilities were not found in the token’s Smart Contract as shown above.

BEP-20 Contract Audit – Common Vulnerabilities – PackSeller.sol

```
import "../libs/KaijuLibrary.sol";
import "../libs/RandomHelper.sol";
import "../libs/RoleDefine.sol";
import "../libs/TypeDefine.sol";
import "../interfaces/IKaijuRouter.sol";
import "../interfaces/INFT.sol";
import "../interfaces/IKaijuNFT.sol";
import "../interfaces/IFactory.sol";

import "../nft/KaijuNFT.sol";
import "@openzeppelin/contracts/access/Ownable.sol";

import "@openzeppelin/contracts/token/ERC1155/IERC1155.sol";
import "@openzeppelin/contracts/token/ERC1155/IERC1155Receiver.sol";
import "@openzeppelin/contracts/token/ERC721/IERC721.sol";
import "@openzeppelin/contracts/token/ERC721/ERC721.sol";
import "@openzeppelin/contracts/token/ERC721/IERC721Receiver.sol";
import "@openzeppelin/contracts/utils/math/SafeMath.sol";

import "@openzeppelin/contracts-upgradeable/proxy/utils/Initializable.sol";
import "@openzeppelin/contracts-upgradeable/access/AccessControlUpgradeable.sol";

contract PackSeller is
    Initializable,
    AccessControlUpgradeable,
    IERC1155Receiver,
    IERC721Receiver
{
    using SafeMath for uint256;
    struct Land {
        uint256 typeId;
        uint256 id;
    }

    uint256 public startTime;
    address public packAddress;
    address public sellerAddress;
    uint256[] public prices;
    uint256[] public amounts;
    mapping(uint256 => uint256) public soldAmounts;

    IKaijuRouter router;
    IFactory factory;

    bool private locked;

    //Nonce for random
    uint256 private plantNonce;
    uint256 private eggNonce;
    uint256 private avatarNonce;
    uint256 private avatarFrameNonce;
    uint256 private landNonce;

    //Max for item value
    uint256 public maxBasicEggId;
    uint256 public maxAvatarId;
    uint256 public maxAvatarFrameId;
    address[] public addressList;
    mapping(address => bool) public whitelist;
    mapping(address => bool) public claim;

    uint256 private promoPackNonce;
    bool public canClaim;

    address[] public addressTop;
    mapping(address => uint256) public toplist;
    mapping(address => bool) public rewarded;

    bool public canRewardTop;
```

Code is truncated to fit the constraints of this document.

- ✓ **Integer Underflow**
- ✓ **Integer Overflow**
- ✓ **Correct Token Standards Implementation**
- ✓ **Timestamp Dependency for Randomness**
- ✓ **Unexposed Private Transfer Function**
- ✓ **Transaction-Ordering Dependency**

Common vulnerabilities were not found in the token’s Smart Contract as shown above.

Social Media

We were able to locate a variety of Social Media networks for the project including Twitter, Telegram and Youtube. All links have been conveniently placed below.

Twitter Telegram YouTube

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

Social and Web – Final Thoughts & Suggestions for Improvement

We were able to track down multiple social media channels for the project. Social Media channels included Twitter, Telegram and Youtube. All social media channels had a healthy amount of activity.

It is important to note that there is no tax on transactions including buys and sells. Please use minimum slippage when buying to avoid frontrunning.

This project is being audited before launch. We expect many things to change post-launch. Please review the links on all the pages to make sure that the most up-to-date information is being reviewed. Dessert Finance conducts the audit at or prior to the block number written on the cover of this report.

Suggestions for Improvement

1. Update domain registration to 3+ years ***✓Fixed***
2. Fix Javascript errors on website ***✓Fixed***

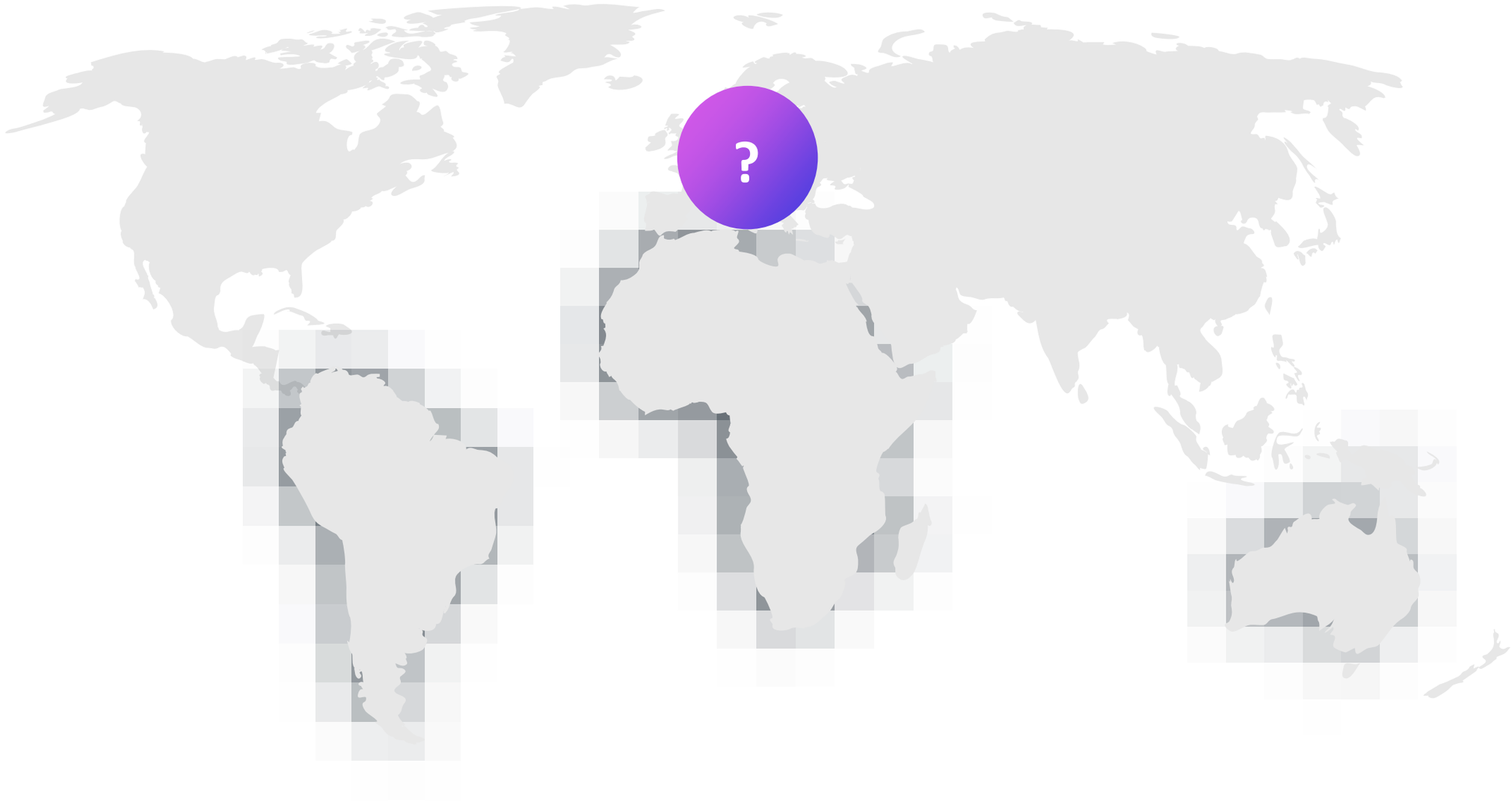
Top Token Holders

The tokens have not been minted yet 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.

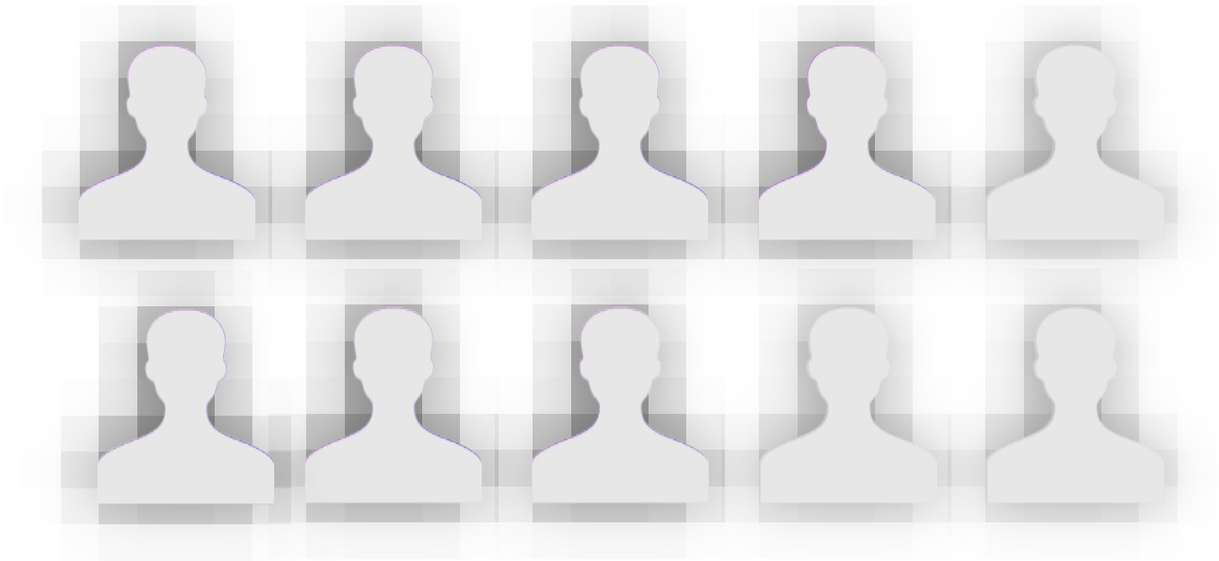
Click here to view the most up-to-date list of holders

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



We are unable to find any information about the team on the website at this time. Projects may choose to stay anonymous for a myriad of reasons.

Potential Signs of Risk

1

1 YEAR DOMAIN REGISTRATION

✓*Fixed*

2

JAVASCRIPT ERRORS FOUND ON WEBSITE

✓*Fixed*

The above listed are the top 2 risk indicators of the project. These are by no means assigning the project as a risky project. Every project will have the top risk indicators posted. It is crucial to note that some may be more important than others.

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 KAIJU WORLDS (KAIJU). 1 DSRT HAS BEEN SENT TO AUDITED PROJECT'S CONTRACT ADDRESS FOR VERIFICATION OF THIS AUDIT AT BLOCK NUMBER: **10428290**

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