

DESSERT  
FINANCE



**MicroZillas (MIZL)**

**BEP-20 Audit**

Performed at block **14121017**

PERFORMED BY DESSERT FINANCE  
FOR CONTRACT ADDRESS: **0x3c130A3E73962Ebd40136E8a6Ac27e837a3f2bED**

## 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.

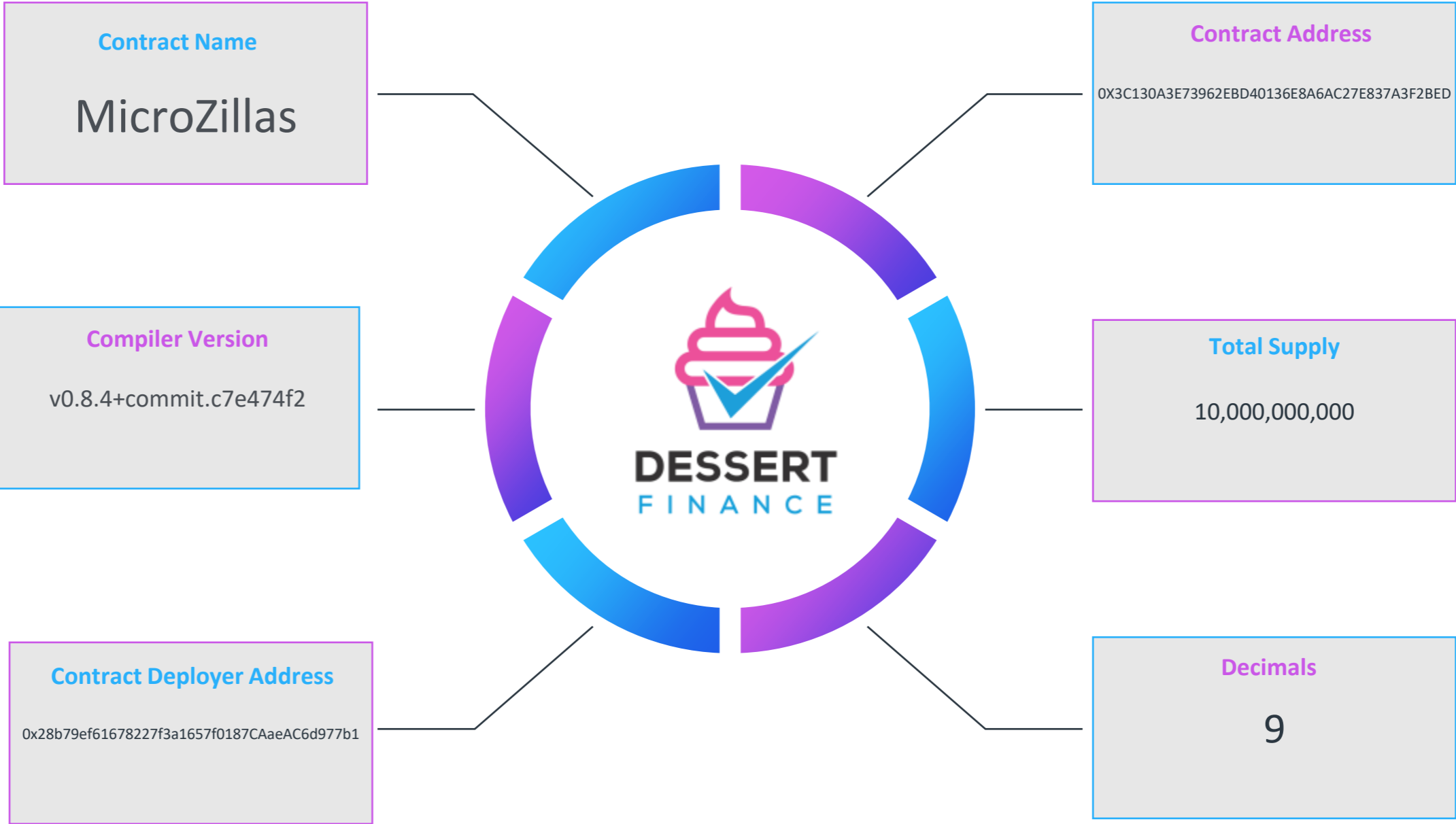
Dessert Finance in no way takes responsibility for any losses, nor does Dessert Finance encourage any speculative investments. The information provided in this audit is for information purposes only and should not be considered investment advice. Dessert Finance does not endorse, recommend, support, 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. BEP-20 Contract Code Audit – Overview
3. BEP-20 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





# BEP-20 Contract Code Audit – Overview

Dessert Finance was commissioned to perform an audit on MicroZillas (MIZL)

```
pragma solidity ^0.8.4;

/**
 * @dev Provides information about the current execution context, including the
 * sender of the transaction and its data. While these are generally available
 * via msg.sender and msg.data, they should not be accessed in such a direct
 * manner, since when dealing with meta-transactions the account sending and
 * paying for execution may not be the actual sender (as far as an application
 * is concerned).
 *
 * This contract is only required for intermediate, library-like contracts.
 */
abstract contract Context {
    function _msgSender() internal view virtual returns (address) {
        return msg.sender;
    }

    function _msgData() internal view virtual returns (bytes calldata) {
        this; // silence state mutability warning without generating bytecode - see https://github.com/ethereum/solidity/issues/2691
        return msg.data;
    }
}

/**
 * @dev Implementation of the [IERC20] interface.
 *
 * This implementation is agnostic to the way tokens are created. This means
 * that a supply mechanism has to be added in a derived contract using {_mint}.
 * For a generic mechanism see [ERC20PermitHolder].
 *
 * TIP: for a detailed writing see our guide
 * https://openzeppelin.io/solutions/#how-to-implement-erc20-supply-mechanisms/2691#how-to-implement-supply-mechanisms.
 *
 * We have followed general OpenZeppelin guidelines: functions revert instead
 * of returning `false` on failure. This behavior is nonetheless conventional
 * and does not conflict with the expectations of ERC20 applications.
 *
 * Additionally, an [Approval] event is emitted on calls to [transferFrom].
 * This allows applications to reconstruct the allowances for all accounts just
 * by listening to said events. Other implementations of the TIP may not emit
 * these events, as it isn't required by the specification.
 *
 * Finally, the non-standard [decreaseAllowance] and [increaseAllowance]
 * functions have been added to mitigate the well-known issues around setting
 * allowances. See [IERC20-approve].
 */
```

## Contract Address

0x3c130A3E73962Ebd40136E8a6Ac27e837a3f2bED

## TokenTracker

MicroZillas (MIZL)

## Contract Creator

0x28b79ef61678227f3a1657f0187caaeac6d977b1

## Source Code

Contract Source Code Verified

## Contract Name

MicroZillas

## Other Settings

byzantium EvmVersion

## Compiler Version

v0.8.4+commit.c7e474f2

## 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.](#)

# BEP-20 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	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.

# 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:

[0x28b79ef61678227f3a1657f0187caaeac6d977b1](https://www.etherbase.net/etherbase-token/contract-ownership)

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
updateUniswapV2Router	address newAddress	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
excludeFromFees	address account, bool excluded	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setBuybackWallet	address payable wallet	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setLiquiditFee	uint256 value	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced. This fee does not have an upper limit and can be set to 100.
setBuybackFee	uint256 value	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced. This fee does not have an upper limit and can be set to 100.
manualSwapnLiq		public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
changeSnipe	bool _snipe	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
Sweep		external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
SetSwapSeconds	uint256 newSeconds	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setEnabledAntiBot	bool _enable	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setLastBlockNumber	uint256 _number	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setLastPairBalance		public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setminimumTokensBeforeSwap	uint256 _new	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
changeDivgas	uint256 _new	public	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.



# Contract Code Audit – Owner Accessible Functions

Function Name	Parameters	Visibility	Audit Notes
setBlockChunk	uint256 _chunk	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setCurrentLiqPair	address _pair	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
botTimer	uint256 _timer	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setMaxTxAmount	uint256 maxTxAmount	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
setcheckUptrendActive	bool _enabled	public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
removeSniper	address account	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
changeMaxWallet	uint256 maxWallet	external	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
launch		public	onlyOwner modifier is detected. Owner can call this function if the contract is not renounced.
addLiquidityHolder	address holder	external	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. This function has not been called
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 Cannot Mint New MIZL 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.**

# Contract Transaction Fees

At the time of Audit the transaction fees (“tax”) listed below are the fees associated with trading. These fees are taken from every buy and sell transaction unless otherwise stated.





# Website Part 1 – Overview

## [www.microzilla.io](http://www.microzilla.io)



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

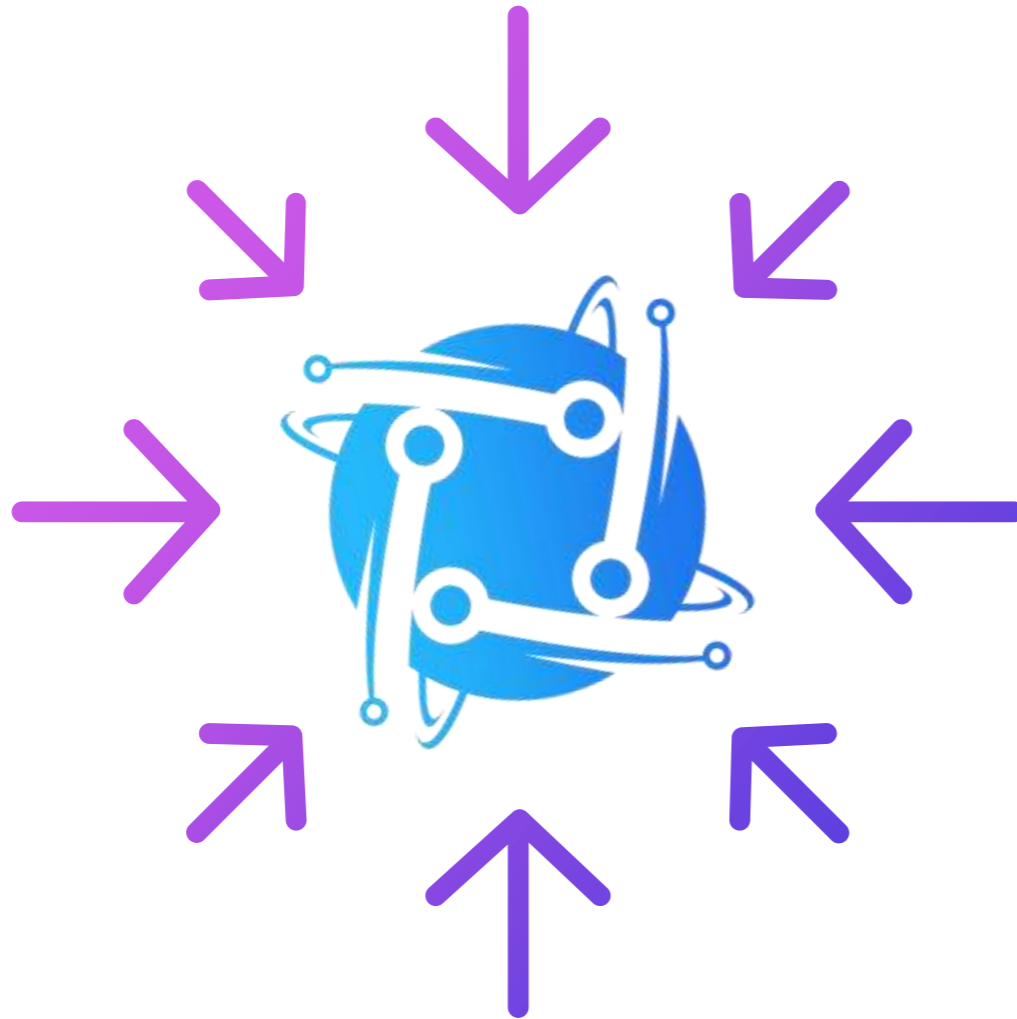
Website was registered on 11/21/2021, registration expires 11/21/2022.

**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: microzilla.io

Issued by: R3

Valid Until: 02/19/2022



## 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

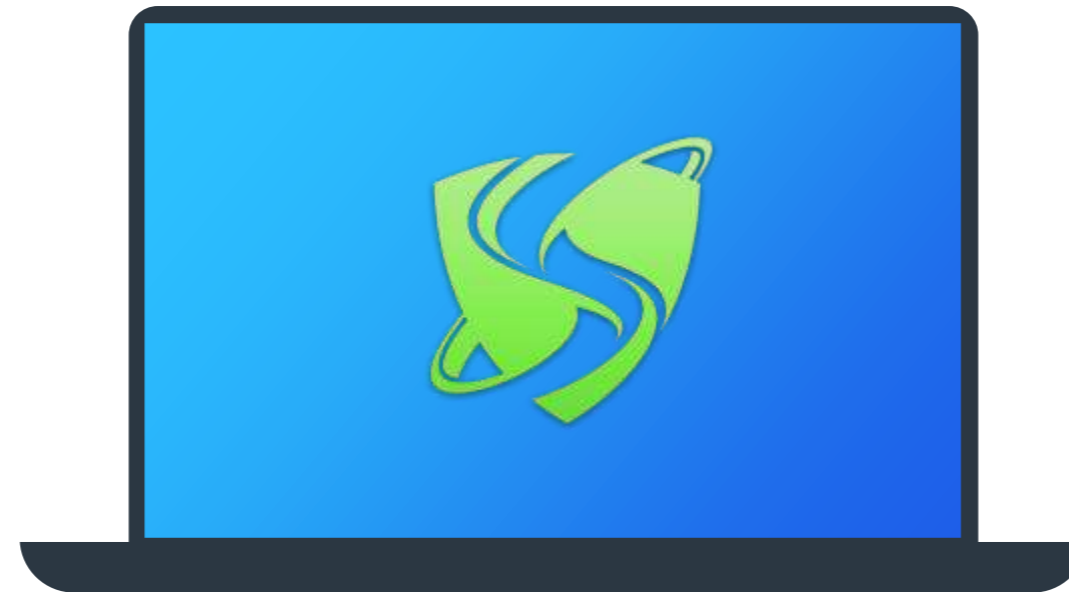
No internal server errors

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](#)



[Facebook](#)



[Instagram](#)

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

# Top Token Holders

The entire supply was in one 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](#)

## MicroZillas Top 100 Token Holders

Source: BscScan.com





# 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

The team has provided us with links to the founder and co-founders LinkedIn profiles.  
We have placed these links here for you to view.



[Bry Shen](#)  
[Founder](#)



[Sam Lee](#)  
[Co-Founder](#)

# Roadmap

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

**ROADMAPS**

**Q4/2021**

- Mystery Eggs & NFT's Zillas
- Farming + Staking (Take care of your Zillas)
- NFT Marketplace
- Dapp Testing
- Contract Audit & KYC
- Community Growth & Engagement (Initial marketing campaign with competitions)

**Q1/2022**

- Presale On Pinksale (Audit & KYC)
- Listing on PancakeSwap
- Listing on CoinGecko & CoinMarketCap
- Order Certik Audit
- Release Additional NFT's Zillas
- P2E Game
- Continued Marketing Push
- Cross Chain Market ( ETH,SOLANA )
- Worldwide Huge Marketing

# 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 MICROZILLAS (MIZL) 1 DSRT HAS BEEN SENT TO AUDITED PROJECT'S CONTRACT ADDRESS FOR VERIFICATION OF THIS AUDIT AT BLOCK NUMBER: **14121017**

[www.dessertswap.finance](http://www.dessertswap.finance)  
<https://t.me/dessertswap>