

DESSERT  
FINANCE



**MaticVerse (Mverse)**

**Polygon Audit**

Performed at block **11167122**

PERFORMED BY DESSERT FINANCE  
FOR CONTRACT ADDRESS: **0xFEB090FcD433dE479396E82DB8B83a470dbAD3c9**

## INITIAL DISCLAIMER

Dessert Finance provides due-diligence project audits for various Polygon 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.

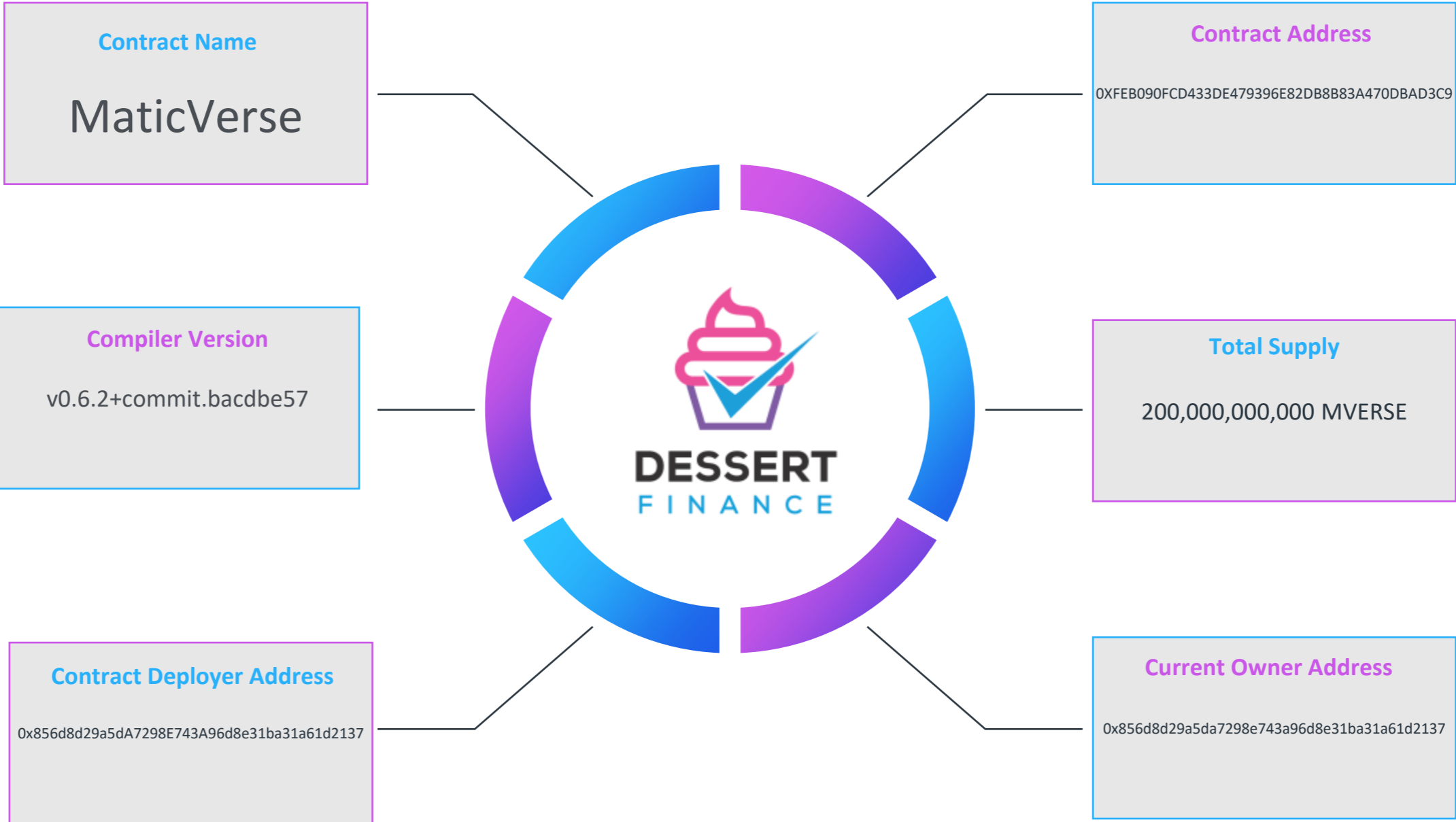
Dessert Finance in no way takes responsibility for any losses, nor does Dessert Finance encourage any speculative investments. The information provided in this audit is for information purposes only and should not be considered investment advice.

# Table of Contents



1. Contract Code Audit – Token Overview
2. Polygon Contract Code Audit – Overview
3. Polygon Contract Code Audit – Vulnerabilities Checked
4. Contract Code Audit – Contract Ownership
5. Contract Code Audit – Mint Functions
6. Contract Transaction Fees
7. Website Overview
8. Social Media
9. Top Token Holders/Wallets
10. Location Audit
11. Review of Team
12. Roadmap
13. Disclaimers

# Contract Code Audit – Token Overview





# Polygon Contract Code Audit – Overview

Dessert Finance was commissioned to perform an audit on MaticVerse (Mverse)

```

Submitted for verification at polygonscan.com on 2021-09-22
*/
pragma solidity ^0.6.2;

/**
 * @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 memory) {
        /**/ // silence state mutability warning without generating bytecode - see https://github.com/ethereum/solidity/issues/2691
        return msg.data;
    }
}

pragma solidity ^0.6.2;

contract Ownable is Context {
    address private _owner;

    event OwnershipTransferred(address indexed previousOwner, address indexed newOwner);

    /**
     * @dev Initializes the contract setting the deployer as the initial owner.
     */
    constructor () public {
        address msgSender = msg.sender();
        _owner = msgSender;
        emit OwnershipTransferred(address(0), msgSender);
    }

    /**
     * @dev Returns the address of the current owner.
     */
    function owner() public view returns (address) {
        return _owner;
    }

    /**
     * @dev Throws if called by any account other than the owner.
     */
    modifier onlyOwner() {
        require(_owner == msgSender(), "Ownable: caller is not the owner");
        _;
    }

    /**
     * @dev Leaves the contract without owner. It will not be possible to call
     * 'onlyOwner' functions anymore. Can only be called by the current owner.
     *
     * NOTE: Renouncing ownership will leave the contract without an owner,
     * thereby removing any functionality that is only available to the owner.
     */
    function renounceOwnership() public virtual onlyOwner {
        emit OwnershipTransferred(_owner, address(0));
        _owner = address(0);
    }
}

```

## Contract Address

0xFEB090FcD433dE479396E82DB8B83a470dbAD3c9

## TokenTracker

MaticVerse (Mverse)

## Contract Creator

0x856d8d29a5da7298e743a96d8e31ba31a61d2137

## Source Code

Contract Source Code Verified (Exact Match)

## Contract Name

MaticVerse

## Other Settings

default evmVersion, MIT

## Compiler Version

0.6.2+commit.bacdbe57

## 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 Polygonscan.

# Polygon 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 Polygonscan.

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:

Owner Address: [0x856d8d29a5da7298e743a96d8e31ba31a61d2137](https://etherscan.io/address/0x856d8d29a5da7298e743a96d8e31ba31a61d2137)

# Contract Code Audit – Owner Access Control

The following functions are accessible by the Owner.



renounceOwnership	setSell_MarketingFee
transferOwnership	setBuy_MATICRewardsFee
updateDividendTracker	setBuy_LiquidityFee
updateUniswapV2Router	setBuy_MarketingFee
excludeFromFees	UpdatePriceImpact
excludeMultipleAccountsFromFees	TurnOn_Trading
setAutomatedMarketMakerPair	TurnOff_Trading
updateGasForProcessing	TurnOn_BuyTax
updateClaimWait	TurnOff_BuyTax
setMarketingWallet	TurnOn_SellTax
setDevWallet	TurnOff_SellTax
TurnOn_PauseTax	excludeFromDividends
TurnOff_PauseTax	updateClaimWait
setSell_MATICRewardsFee	setBalance
setSell_LiquidityFee	processAccount



# Contract Code Audit – Mint Functions

This Contract Cannot Mint New Mverse Tokens.



We do understand that sometimes mint functions are essential to the functionality of the project. If a mint function is ever listed we will have the team clarify its use case for transparency purposes.

**An active 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.

## Buy Tax

4%

TO HOLDERS

2%

LIQUIDITY  
POOL

4%

MARKETING

## Sell Tax

5%

TO HOLDERS

3%

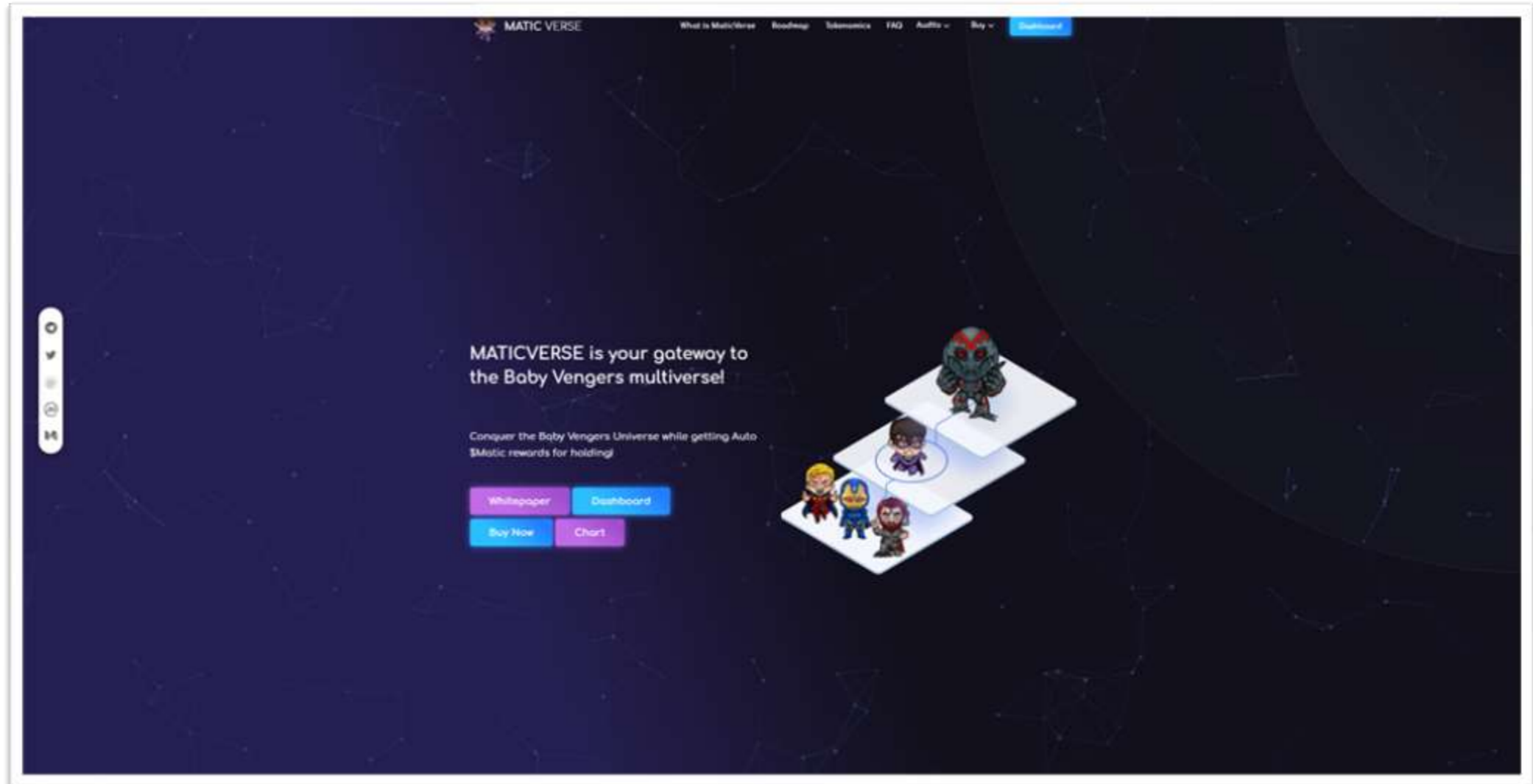
LIQUIDITY  
POOL

7%

MARKETING

# Website Part 1 – Overview

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



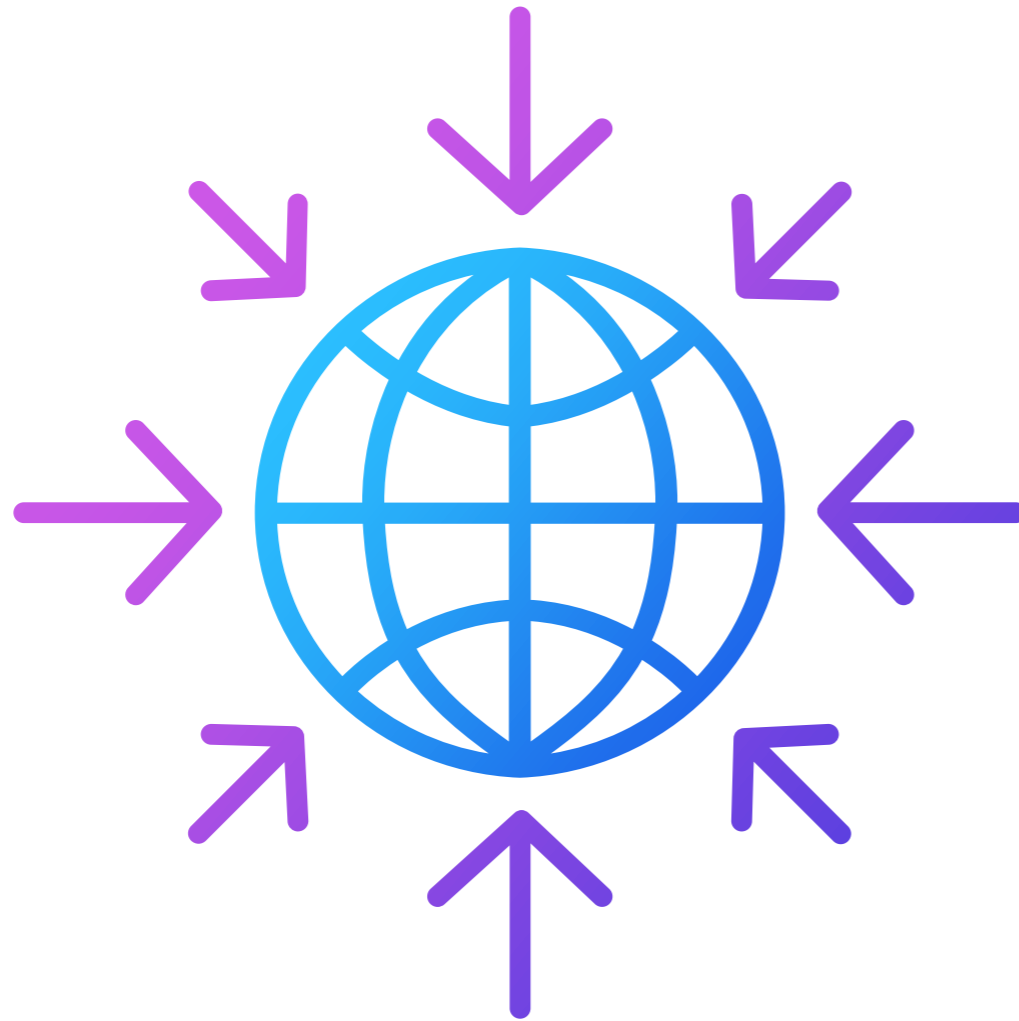
Above images are actual snapshots of the current live website of the project.

Website was registered on 09/05/2021, registration expires 09/05/2024.

✓ 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

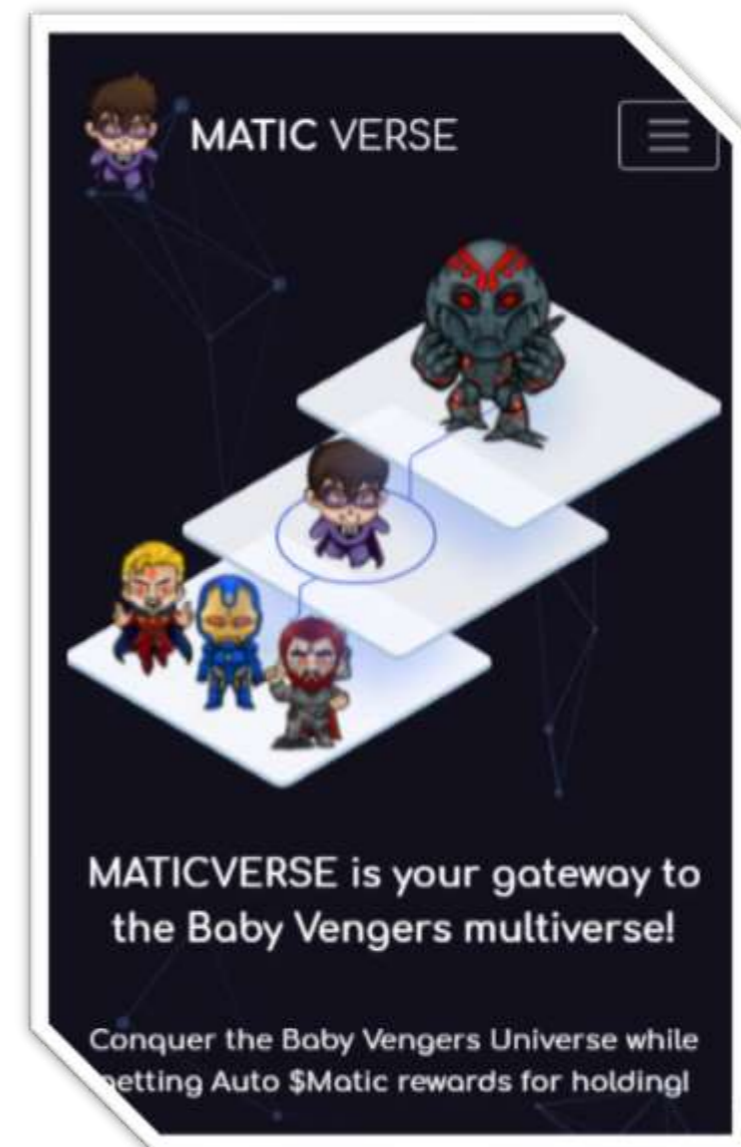
The website contained no JavaScript errors. No typos, or grammatical errors were present, and we found a valid SSL certificate allowing for access via https.

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 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: \*.maticverse.io

Issued by: R3

Valid Until: 12/09/2021



## CONTACT EMAIL

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

Contact

**team@maticverse.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



# 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](#)



[Medium](#)

✓ At least 3 social media networks were found.

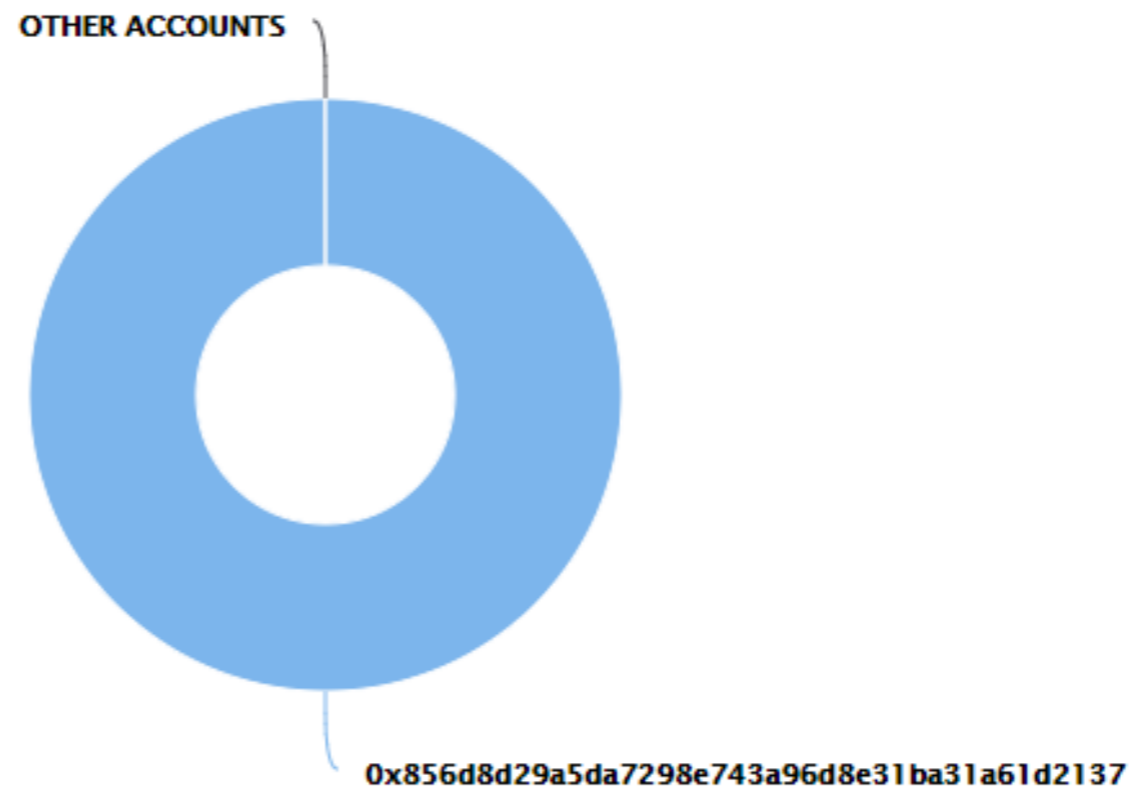
# Top Token Holders

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](#)

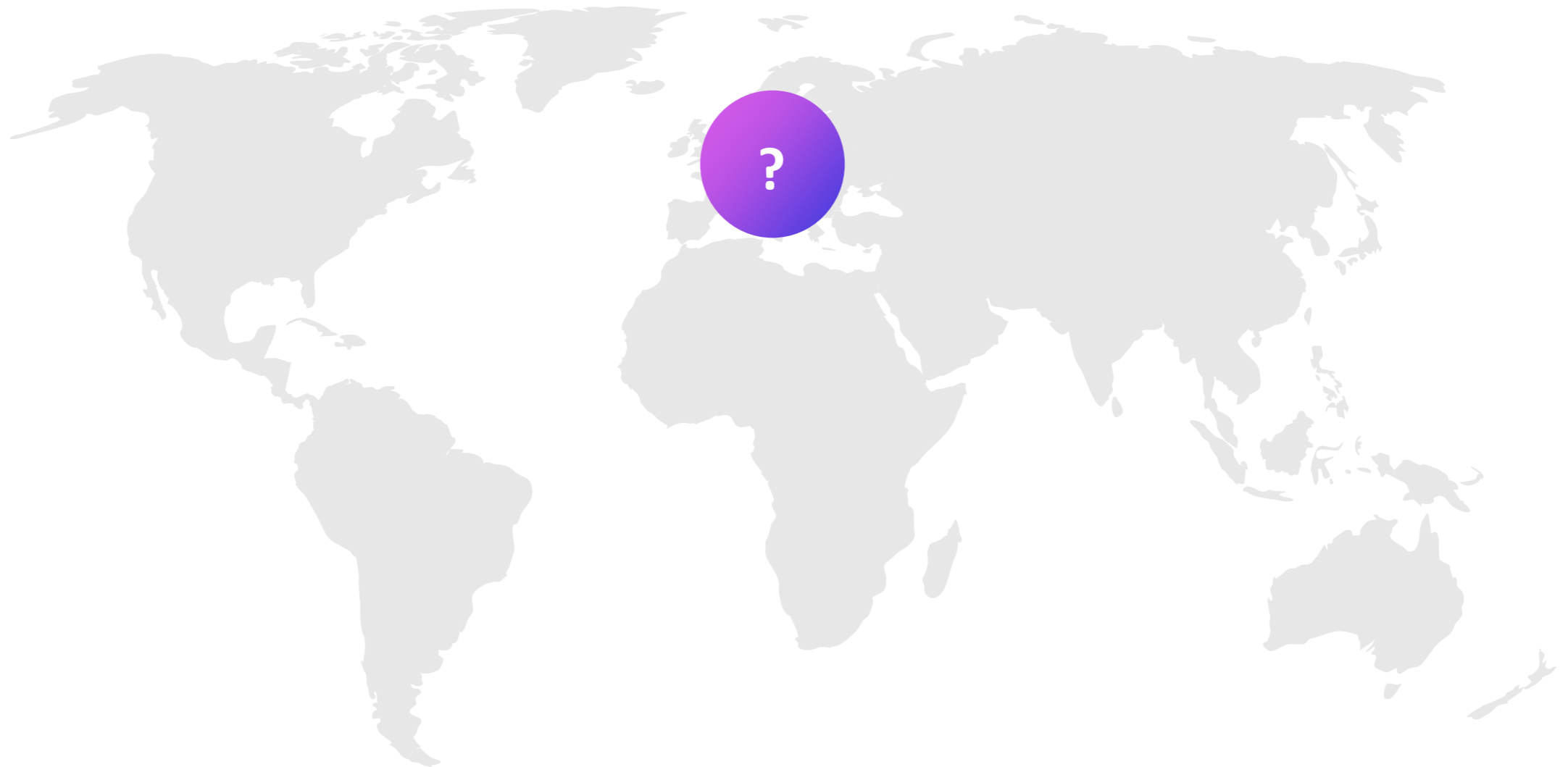
## MaticVerse Top 100 Token Holders

Source: [polygonscan.com](https://polygonscan.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

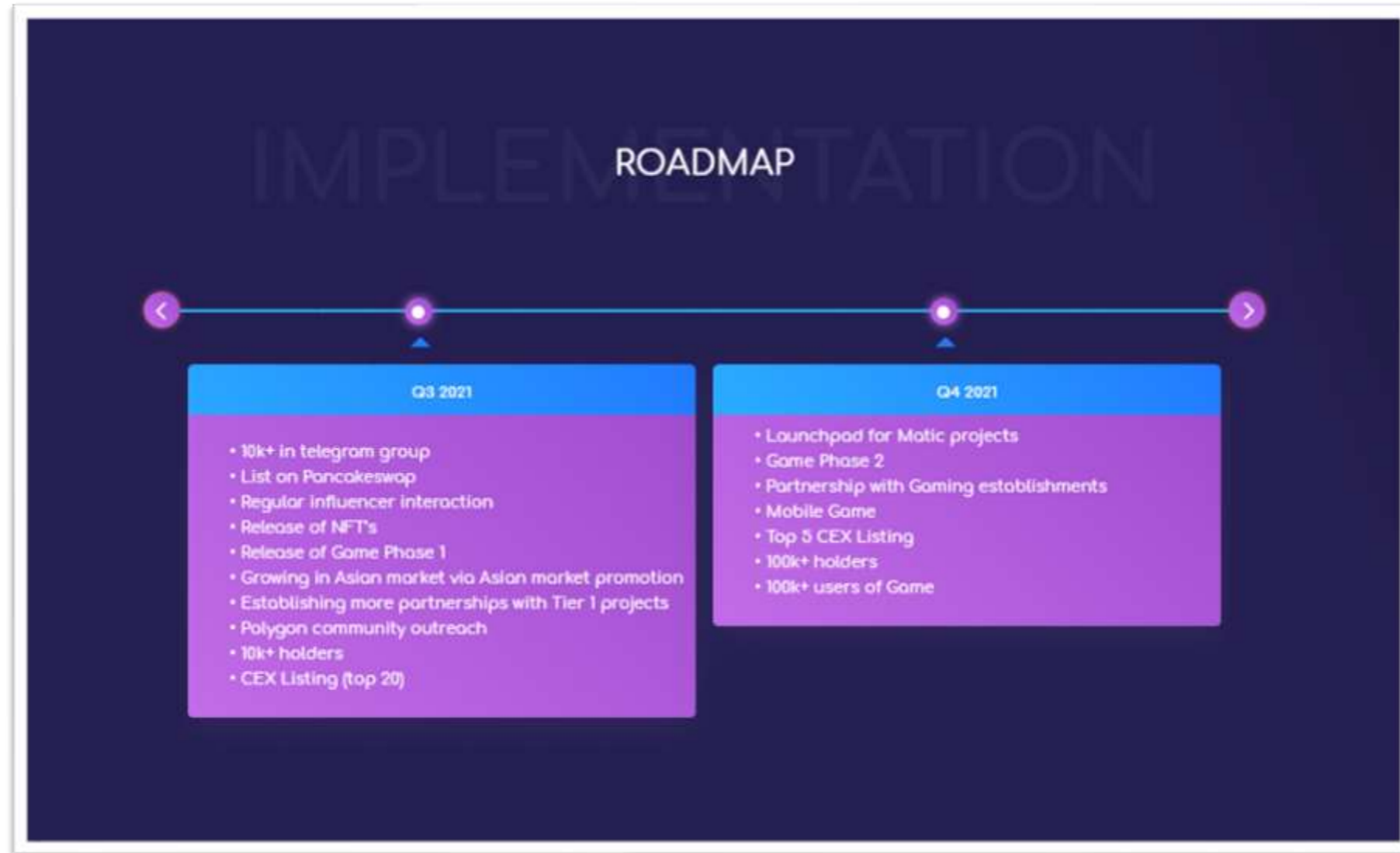
CREATIVE TEAM

 Asad Siddiqui CEO	 Farhan Ghauri CSO	 Mahad CFO
 Rokr CDO	 Vishal CMO	 Zee Advisor
 Cryptoboy Advisor	 Zoro Head of NFTs	 Kauser Uddin Social Media Lead
 Jules Head of Art	 Ameen Farooqui Marketing Executive	 Vivek Advisor
 Ridwan Hussain Marketing Executive	 Abnormal Lead Developer	 Abhi PR Executive
 Xa Advisor		



# 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 polygon 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 MATICVERSE (MVERSE) 1 DSRT HAS BEEN SENT TO AUDITED PROJECT'S CORRESPONDING CONTRACT ADDRESS ON BSC FOR VERIFICATION OF THIS AUDIT AT BLOCK NUMBER: **11167122**

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