

# Espento Whitepaper v1

## The ESPENTO Team

December 19, 2022

### Abstract

The number of supporters in the financial services sector is rising for the daily use of distributed ledger technology (DLT). Why not, then? After all, blockchain and DLT have both demonstrated enormous potential. Blockchain promises to significantly reduce infrastructure, transaction, and administrative expenses for financial institutions (FIs).

By removing the need for central counter parties, it can disintermediate digital financial asset transfers. Also, it can contribute to the financial ecosystem's robustness, accuracy, and trustworthiness. Blockchain can save the infrastructure costs incurred by banks for securities trading, international payments, and regulatory compliance by around US\$20 billion annually by 2022, according to a Santander analysis published in 2015.

By using affiliate-based Reward distribution and Defi implementations, some decentralized networks concentrate on developing a speedier medium for the decentralized exchange ecosystem. Yet, none have achieved adequate success to date.

By a cutting-edge technology, Espento aims to fix the ecosystem's current problems and deliver the appropriate outcome in accordance with the trading systems.

Decentralized markets provide non-custodial dealing that is trustless, increased openness, and the ability to share order books. Despite these benefits, decentralized platforms currently account for a negligible portion of total blockchain transaction traffic. High entry hurdles in terms of technology, speed, availability, usability, and a dearth of rewards for collaboration are all reasons for low adoption.

### Introduction

Spent is a digital token that symbolizes the website [www.espento.com](http://www.espento.com) as well as the Espento Ecosystem. The Espento decentralized exchange eliminates the need for buyers and vendors to go through a centralized third party by enabling them to trade value directly with other market players.

The Espento brand actively markets the 9 leading features.

1. DEX Algorithmic Liquidity Reserves
2. Bridge Contract to Promote the Binance Smart Chain Environment
3. Yield Farming
4. Staking for validating transactions on Binance Smart Chain
5. Smart Lending-borrowing protocol
6. DAO protocol
7. Defi Gaming platform
8. Web3 Wallet

The protocol is backed by a native digital token SPENT Token - in order to accomplish its decentralized objectives. This token works on the BNB Chain and can be transferred from wallet to wallet. Users are encouraged to keep their SPENT tokens for the long term as the Espento community matures.

Users will be able to make a yield in stable token and SPENT Token by holding and staking the tokens. As a result, it discourages users from engaging in day trading, which has the unintended consequence of increasing instability and causing wild price fluctuations.

More significantly, a significant portion of the admin fees and revenue made on the total Espento ecosystem transactions will be distributed to the ecosystem promoters. This simply encourages their efforts to support the ecosystem, comparable to traditional dividend payouts.

### Ecosystem Application

SPENT is a token created for the Espento Ecosystem on the BNB Smart Chain that provides multiple solutions and services to the DeFi Ecosystem, such as Yield Farming, DEX with Order Books, Bridge Development, Staking, Defi games, Affiliate program, and the development of Application Native Blockchain. Our entrance into the metaverse will allow us to provide

entertainment-based earning via the creation of NFT games based on the 'Play to Earn' paradigm, thereby building a complete world for the DeFi-centered community members. This will also promote the formation of a DAO for the community's full autonomy and transparency.

### Decentralized Exchange

DEXs enable crypto buyers to trade while holding their keys by utilising liquidity solutions ranging from order books to liquidity pools. The first generation of decentralised exchanges, like traditional controlled exchanges, employ order books. These order books keep track of all active purchase and sell transactions for a specific asset. The spread between these values decides the order book depth and the current market price. This information is frequently kept on-chain during trades on DEXs with order books, while your funds stay off-chain in your wallet.

Users can exchange tokens at the desired price with Espento starting with a limit order. As in CEX (Centralized Exchange), the success of a particular transaction is dependent on the pending orders of both the customer and vendor, which can save users from staring at the market.

The order book for DEX is a technological innovation that combines the order book and the AMM liquidity pool while fully incorporating user suggestions. This will address the issue of transaction failures created by the market's varying order book depth, providing users with an improved trading experience.

Espento creatively integrates the various benefits of various fundamental public chains, resulting in a well-rounded and composite DEX environment with high performance. The "dual mining" mechanism of liquidity mining and transaction mining maximizes returns for users, while the transaction charge "repurchase and burn" mechanism creates a closed and self-driven value capture cycle.

As the only DeFi project on BINANCE, Espento has chosen to debut the order book feature, which is significant for the entire DEX field. Previously, most DEX projects that use the order book feature have a limited market share, few assets, and a small number of users. Although dYdx utilizes the order book, it is primarily used for contract trades and has a low spot transaction volume. Espento's emergence is an indication of change, as it is the first mature DEX initiative to use the order book feature. So, can purchase books truly compensate for AMM's shortcomings?

Although we will have to wait until the next quarter of 2023 to fully test Espento's new function, trading experiences in CES appear to indicate that all of the above-mentioned defects plaguing DEX can be successfully addressed. The order book mechanism's most fundamental purpose is pending orders. It enables users to pend purchases at a specified price based on their insights and preventative measures. At the same time, the set price implies that the usual AMM issues of slippage and front-running will be significantly reduced.

DEX initiatives on ETHEREUM are the primary casualties of excessive transaction costs. Despite the fact that Espento is implemented on BINANCE, a new public chain known for its cheap transaction costs and AMM-based protocol structures.

As a result, Espento is free of the problem of exorbitant transaction costs. The introduction of the order book feature means that liquidity providers (LPs), another important position in the DEX ecosystem, now have another brand-new market-making route.

Professional LPs use the order book method to create trades in a more flexible and frequent manner, increasing their market-making profits. Espento will not, of course, abandon the market's existing original AMM offerings, and ordinary LPs can keep earning from the AMM pool.

Espento is dedicated to building a composite DeFi ecosystem that combines DEX, IMO, and DAO, providing one-stop liquidity services for more high-quality assets, and offering users a more safe, reliable, diverse, and cost-effective trading experience. Espento is definitely the superior, faster, better, and cheaper option for cryptocurrency dealing!

Note: Please keep in mind that this area will increase substantially as work progresses. The team is constantly working diligently to provide you with the finest product details possible.

### The Ultimate Transaction Experience

Launch liquidity optimization agreements to increase capital usage efficiency significantly.

Espento will improve the current AMM system by incorporating new features such as liquidity aggregation, range orders, and limit orders. Because of price changes, few users are currently willing to use inactive tokens for liquidity mining. However, in the future, losses from short-term fluctuations will be prevented to the maximum extent feasible, allowing a DEX to perform essential functions such as liquidity aggregation and range and limit orders. DEX miners are currently forced to endure losses caused by short-term volatility, which is why many token holders are hesitant to place their assets in liquidity pools. As a DEX project, Espento has been consistently engaged in technological and business model innovation, strengthening the business model after the technology is stable and introducing technological innovations after the model is clear, creating a virtuous cycle to provide users with the best DEX experience possible.

Experience with trading interfaces and order books

To create a working experience similar to that of a centralized exchange, Espento will provide a market display page and order book trading features that are in accordance with user behaviors, as well as a more efficient and better pending order strategy.

### Core Functions of Espento's Router :

#### quote

```
function quote(uint amountA, uint reserveA, uint reserveB) internal pure returns (uint amountB);
```

Given some asset amount and reserves, returns an amount of the other asset representing equivalent value.

Useful for calculating optimal token amounts before calling mint.

#### getAmountOut

```
function getAmountOut(uint amountIn, uint reserveIn, uint reserveOut) internal pure returns (uint amountOut);
```

Given an input asset amount, returns the maximum output amount of the other asset (accounting for fees) given reserves.

#### getAmountIn

```
function getAmountIn(uint amountOut, uint reserveIn, uint reserveOut) internal pure returns (uint amountIn);
```

Returns the minimum input asset amount required to buy the given output asset amount (accounting for fees) given reserves.

#### getAmountsOut

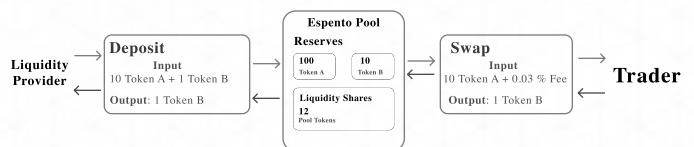
```
function quote(uint amountA, uint reserveA, uint reserveB) internal pure returns (uint amountB);
```

Given an input asset amount and an array of token addresses, calculates all subsequent maximum output token amounts by calling getReserves for each pair of token addresses in the path in turn, and using these to call getAmountOut.

Useful for calculating optimal token amounts before calling swap.

#### AddLiquidity

```
function addLiquidity(
    address tokenA,
    address tokenB,
    uint amountADesired,
    uint amountBDesired,
    uint amountAMin,
    uint amountBMin,
    address to,
    uint deadline
) external returns (uint amountA, uint amountB, uint liquidity);
```



**For Example :** if the User adds \$50001 liquidity into the Pool, how many LP tokens he receives with airdrop LP tokens are described below.

#### Considerations:

- Total Pool size: 322552.32 LP
- Ratio: 10:1
- SPENT's current Price is \$0.10
- Users are required to buy and add \$25000.50 worth of SPENT and \$25000.50 worth of eUSD

The user will receive LP tokens as per the below-given formula

$$\text{liquidity} = \frac{\text{Math.min}(\text{amount0.mul}(\_totalSupply) / \_reserve0, \text{amount1.mul}(\_totalSupply) / \_reserve1)}{}$$

$$\text{LP Token} = \frac{250005 * 322552.32}{1020000} = \frac{25000.50 * 322552.32}{102000}$$

User total LP = **Received LP**

User total LP = **79058.52**

$$\text{Value of 1 LP Token} = \frac{\text{Total Value of Liquidity Pool}}{\text{Circulating Supply of LP Tokens}}$$

$$\text{LP Token Price} = \frac{254001}{401610.84} = 0.6324$$

### Add ERC-20 liquidity to an ERC-20 pool.

To cover all possible scenarios, **msg.sender** should have already given the router an allowance of at least amountADesired/amountBDesired on tokenA/tokenB.

Always adds assets at the ideal ratio, according to the price when the transaction is executed.

If a pool for the passed tokens does not exist, one is created automatically, and exactly amountADesired/amountBDesired tokens are added.

#### removeLiquidity

```
function removeLiquidity(
    address tokenA,
    address tokenB,
    uint liquidity,
    uint amountAMin,
    uint amountBMin,
    address to,
    uint deadline
) external returns (uint amountA, uint amountB);
```

*Removes liquidity from an ERC-20 ERC-20 pool.*

**msg.sender** should have already given the router an allowance of at least liquidity on the pool.

#### swapExactTokensForTokens

```
function swapExactTokensForTokens(
    uint amountIn,
    uint amountOutMin,
    address[] calldata path,
    address to,
    uint deadline
) external returns (uint[] memory amounts);
```

*Swaps an exact amount of input tokens for as many output tokens as possible, along the route determined by the path. The first element of path is the input token, the last is the output token, and any intermediate elements represent intermediate pairs to trade through (if, for example, a direct pair does not exist).*

**msg.sender** should have already given the router an allowance of at least amountIn on the input token.

#### swapTokensForExactTokens

```
function swapTokensForExactTokens( uint amountOut, uint amountInMax,
    address[] calldata path, address to, uint deadline ) external returns (uint[]
    memory amounts);
```

*Receive an exact amount of output tokens for as few input tokens as possible, along the route determined by the path. The first element of path is the input token, the last is the output token, and any intermediate elements represent intermediate tokens to trade through (if, for example, a direct pair does not exist).*

**msg.sender** should have already given the router an allowance of at least amountInMax on the input token.

### Yield Farming

Our yield farming strategy entails lending tokens through the BNB smart chain network. When banks make advances with fiat money, the sum loaned is repaid with interest. The idea is similar to yield farming, with the only variation being the cryptocurrency used in the process. A token that would otherwise be lying in an exchange or a wallet is loaned out via DeFi protocols (or, in Binance terminology, locked into smart contracts) in order to earn a return.

Our yield farming strategy entails lending tokens through the BNB smart chain network. When banks make advances with fiat money, the sum loaned is repaid with interest. The idea is similar to yield farming, with the only variation being the cryptocurrency used in the process. A token that would otherwise be lying in an exchange or a wallet is loaned out via DeFi protocols (or, in Binance terminology, locked into smart contracts) in order to earn a return.

### Core Functions For CropYard :

```
function Farm(uint256 _pId, uint256 amount) external returns (bool);
```

*Used to start farm*

```
function Harvest(uint256 _pId) external returns (bool);
```

*Used to harvest the farming rewards*

```
function unStakeFarm(uint256 _pId) external returns (bool);
```

*Used to harvest the farming rewards*

### Staking

Staking cryptocurrency is the process of pledging your crypto assets to support and confirm transactions on a blockchain network. It is compatible with cryptocurrencies that process payments using the proof-of-stake mechanism.

Staking is a way of earning rewards on cryptocurrency by depositing the stake accepted tokens for a fixed period. Staking works in a similar way to interest accounts with traditional banks.

### Core Functions for Upright ( Staking ) :

*For staking :*

```
function stakeStableTokens(uint256 _pId, uint256 amount) external returns (bool);
function stakeSwapLpTokens(uint256 _pId, uint256 amount) external returns (bool);
function stakeSwapTokens(uint256 _pId, uint256 amount) external returns (bool);
function stakeBorrowStakeTokens(uint256 _pId, uint256 amount) external returns
(bool);
```

*For Claiming stake rewards :*

```
function ClaimStableToken(uint256 _pId) external returns (bool);
function ClaimSwapLpToken(uint256 _pId) external returns (bool);
function ClaimSwapToken(uint256 _pId) external returns (bool);
function ClaimBorrowStakeToken(uint256 _pId) external returns (bool);
```

*For unstaking/stopping stake :*

```
function unStakeStableToken(uint256 pId) external returns (bool);
function unStakeSwapLpToken(uint256 pId) external returns (bool);
function unStakeSwapToken(uint256 pId) external returns (bool);
function unStakeBorrowStakeToken(uint256 pId) external returns (bool);
```

### DAO

A decentralized organization that governs the network, which enables Espento holders to vote for key protocol changes.

Espento allows DAO members holding the required SPENT tokens to easily participate, profit, and vote in various protocol settings without long voting periods.

### Tokenomics

#### AIRDROP (1%)

Espento promotes airdrops for its branding. SPENT Token is used from time to time to promote knowledge of the ecosystem and community through various Social Media initiatives and CPA programs.

#### MARKETING & REWARDS(22%)

A portion of the token supply will be allocated to online and offline promotions, content-driven marketing, social media initiatives, and affiliate/referral marketing activities in general.

#### DEVELOPMENT(10%)

The technological staff plays an important role in the ecosystem, and as the platform's primary movers, they are inextricably linked to the original benefactors.

#### GAME DEVELOPMENT (5%)

The game platform is an essential part of the ecosystem revenue, and a portion of the tokenomic's share has been given for larger and more comprehensive growth.

**POOL LIQUIDITY (2%)****Initial Liquidity for Swapping**

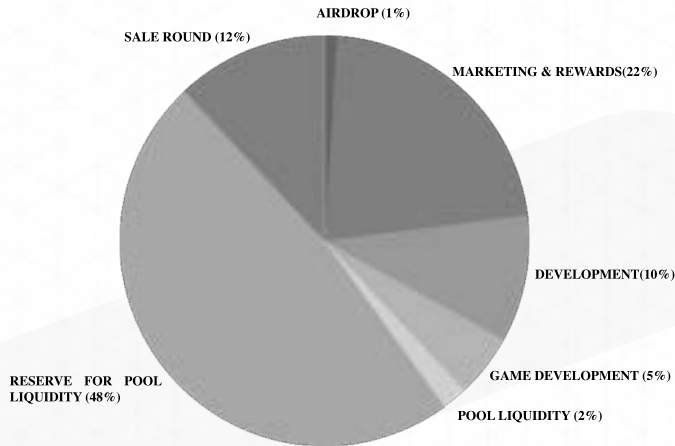
Espento has added 1.02M(2%) SPENT tokens as Liquidity for Swap for pair SPENT/eUSD.

**RESERVE FOR POOL LIQUIDITY (48%)**

Espento has set aside 48% of the entire quantity for Dex Liquidity. When the SPENT supply is short in the liquidity pool, the Dex contract calls the Vault contract and requests the shorted supply.

**SALE ROUND (12%)**

Espento has reserved 12% of total supply for its community members during the Private sale round. By collecting votes from DAO members, the private sale rounds with percentages of the total available supply from the SALE ROUND allocation will be published on schedule.

**Tokenomics Contract Balance Checking Functions :**

```
function getTotalBalance() external returns (uint256);
function getFunds(bytes32 id) external returns (uint256);
function getSaleRoundFunds() external returns (uint256);
function getReserveForPoolLiquidityFunds() external returns (uint256);
function getMarketingAndRewardsFunds() external returns (uint256);
function getDevelopmentFunds() external returns (uint256);
function getAirdropFunds() external returns (uint256);
function getGamingFunds() external returns (uint256);
function setFunds(bytes32 id, uint256 amount) external returns (bool);
function transferFunds(bytes32 id, uint256 amount, address to) external returns (bool);
```

**Upcoming developments****Wallet Development****Features**

Our yield farming strategy entails lending tokens through the BNB smart chain network. When banks make advances with fiat money, the sum loaned is repaid with interest. The idea is similar to yield farming, with the only variation being the cryptocurrency used in the process. A token that would otherwise be lying in an exchange or a wallet is loaned out via DeFi protocols (or, in Binance terminology, locked into smart contracts) in order to earn a return.

Our yield farming strategy entails lending tokens through the BNB smart chain network. When banks make advances with fiat money, the sum loaned is repaid with interest. The idea is similar to yield farming, with the only variation being the cryptocurrency used in the process. A token that would otherwise be lying in an exchange or a wallet is loaned out via DeFi protocols (or, in Binance terminology, locked into smart contracts) in order to earn a return.

- Multi-currency support for storing cryptocurrencies of different chains : BSC Tokens/ ERC 20 /TRC 20/AVAX (ERC - 20).
- Security protocols – Biometric, Seed Phrase, Private Key, User Passkey.
- Portfolio Page – List of crypto with value.
- Use QR Scan Code.
- Create, Restore and Import Wallet.
- Send and receive crypto currencies.
- Token Swapping.
- Backup and Reset Wallet.
- Address Book.
- Real-time notifications and regular updates.
- Ability to change Features – Language, Base Currency.
- Referral (Invite Friends).
- Support Link.

**Espento Wallet**

Espento is a multichain wallet that allows you to securely store, swap, and send Crypto and NFTs across 17 blockchains.

**One wallet for all your crypto**

Swap, send and store more than 10,000 assets on Bitcoin, Ethereum, Solana, THORChain, Avalanche, Fantom, Arbitrum, Polygon, Bitcoin, BNB Smart Chain, Doge, and more.

**Swap 10,000+ assets within the wallet**

Unlimited cross chain swaps and bridging for 10,000+ assets – no sign-up required.

**Industry-leading security**

Additional security options on all of your devices provide more ways to keep your crypto safe and secure.

**Control your crypto**

Wallet is a self-custody wallet, giving you complete control of your crypto.

**Flash transactions**

Flash transactions, available in the Ethereum network, have a very low probability of getting to the public memory pool before execution.

**WalletConnect flexibility**

The WalletConnect functionality can be used in the background, while several WalletConnect options can be used simultaneously.

**Instant transaction signing**

Regardless of their size, transactions can be instantly approved and signed including via WalletConnect.

**Blockchain Development**

- The Espento native chain will be a framework for open, programmable smart contracts for decentralized apps.
- It is our goal to create a fast and scalable blockchain with a robust Validation protocol and a large number of Validator nodes using minimal infrastructure, guaranteeing equitable agreement and optimum chain security.
- The chain will be quick (high TPS count) and scalable enough to serve the Espento Metaverse's social media and Games initiatives.
- ESPENTO will be a scalable Blockchain. Its adaptable architecture allows it to expand indefinitely without sacrificing functionality.
- ESPENTO has a number of distinguishing features that set it apart from other blockchains.