**Smart Contracts for IDOs**

What is an IDO in cryptocurrency?

An Initial DEX Offering, or IDO for short, is a new crowdfunding technique that enables cryptocurrency projects to introduce their native token or coin through decentralized exchanges (DEXs).

When launching new tokens, they usually:

* manage token sale on several IDO platforms
* collect funds from IDO platforms
* add liquidity to DEX
* investors(who deposited funds on IDO contracts) can claim the tokens on IDO platform after DEX liquidity is added (from a certain time).

Reference(Similar platforms):

* <https://bscpad.com/>
* [https://polkastarter.com](https://polkastarter.com/)
* Please check any crypto IDO/ICO websites to have deep understands of how it works.

Project Specs

* Our team(“Play” team) already has a token named “PLAY” (normal ERC20 token).
* We have “PLAY-BUSD” pair on PancakeSwap.
* Based on the token amount that users owned, we will use “tier” functionality.

|  |  |  |
| --- | --- | --- |
| Tier name | Minimum Point | Multiplier |
| Popular | 100 | 1 |
| Star | 500 | 5 |
| SuperStar | 1500 | 15 |
| MegaStar | 2500 | 25 |

Point of User = User’s “PLAY” balance \* 0.8 + User’s “PLAY-BUSD” balance \* 1.5

NOTE! The above formula can be changed.

* Develop “Point” contract and “Tier” contract separately.

**On “Point” contract, we should be able to change token addresses(PLAY and PLAY-BUSD in the above formula) and multipliers of tokens(0.8 and 1.5 in the above formula) at any time (owner only).**

**Tier contract should be linked with Point contract and it will return “tier index and multiplier” of a certain user.**

* And develop IDO and IDOFactory contracts.

IDOFactory contract is to deploy IDO contracts using admin panel.

IDOFactory will have “feeRecipient” and “feePercent” variable.

IDOFactory owner is able to change the info above.

IDOFactory owner can add several operators (roles) and operators can handle creating new IDOs.

After raising the funds on IDO contract, once it’s finalized, “**feePercent**” of “**totalRaised**” will be sent to “**feeRecipient**” address, and the rest will be sent to the project owner.

* IDO Contract specs.

Only IDOFactory contract can **create** IDO contracts.

If it raises at least 51% of target fund, the IDO is considered as “**SUCCESS**”

Otherwise, it is considered as “**FAILURE**” and investors should be able to refund their fund token.

**IDO** contract will have the following variables **initially**:

o **fundToken**: erc20 token address or zero address. (if it’s zero, investors should buy using BNB, otherwise erc20 token such as BUSD or USDT).

o **fundToken Amount**

o **saleToken** : erc20 token address

o **saleToken** Amount

o **startTime** : timestamp that the sale will start

o **endTime** : timestamp that the sale will end

o **claimTime** : timestamp that users can start claiming the saleToken

o **tge**, **cliffTime**, **duration**, **periodicity**: factors for vesting saleToken.

**tge**: 20%, **cliffTime**: 2022-09-01, **duration**: 2 weeks, **periodicity**: 1week => investors can claim 20% at the time of claimTime, and the rest are vested. They can claim 40% after 2022-09-08 and the last 40% after 2022-09-15.

o **baseAmount** : tiers can fund up to “baseAmount \* multiplier” during Tier Fund Round (see below)

o **maxAmountPerUser**: investors can fund up to this value during FCFS round (see below)

o **whitelistedAmounts**: investors can fund up to “whitelistedAmounts[user]” during whitelist round (see below)

o you can define other variables or functions you need, but validate parameters as much as possible.

**Operators**(operator role on IDOFactory) can use operatorOnly functions to update IDO factors. (startTime, endTime, claimTime, tge, cliffTime, duration, periodicity, fundToken Amount, sale Token amount)

Operators can’t update the time after it’s passed. E.g: startTime is 2022-07-01, current time is 2022-07-02. In that case, operators can’t update startTime. (as IDO started already). Same rule applies to “endTime”, “claimTime”, “tge”, “cliffTime”)

o Define **setSaleInfo** function, and we can change fundToken and saleToken amounts here, but can’t change after IDO is started.

o Define **finalize**(address addr) function. IDOFactory owner can call this function. If total funded amount is more than 51% of “fund token amount”, some of fundToken on the contract is sent to feeRecipient, and the rest is sent to “addr”(from the parameter).

Otherwise, users can call refund function to get back their funded token.

“**finalize**” function can be called only after “endTime”. Also, before calling this function, project owner must send enough “sale token” to this IDO contract.

o Define “**setWhitelistAmount**(address funder, uint256 amount)” and “setWhitelistAmounts(address[] calldata funders, uint256[] calldata amounts)” to set whitelistedAmounts

o Define “**fund**”, “**claim**” and “**refund**” functions.

o Define “**emergencyRefund**” function. Only IDOFactory can call this function and once this function is called, IDO is cancelled and investors can refund their token.

About the IDO process,

IDO starts at “StartTime” and ends at “EndTime”.

Total duration (endTime – startTime) is divided into 3 rounds with the same duration: **tier** round, **whitelist** round and **fcfs** round.

Tier round: only tiers can fund

Whitelist round: only whitelisted investors can fund

FCFS round: any users can fund

“Investors’ saleToken amount” = “investor’s funded amount” \* “saleToken amount” / “fundToken amount”

* Task

Develop Point, Tier, IDO, IDOFactory smart contracts.

Write unit-test for all contracts(many test-cases as possible, even for a small parameter validator)

Example:

Let’s assume there are 3 users: Alice, Bob and Carol

**fundToken**: BUSD (it should be BUSD token address)

**fundAmount**: 1000 Ether (**1000** BUSD)

**saleToken**: “SEG” (18 decimals)

**saleToken Amount**: 5000 Ether(5000 SEG)

It means, “SEG” project owner wants to sell 5000 SEG with 1000 BUSD. ( so SEG token price is 0.02 BUSD per SEG)

startTime: 2022-09-01

endTime: 2022-09-10

claimTime: 2022-09-12

cliffTime: 2022-09-15

tge: 20%, duration: 2 weeks, periodicity: 1 week

**baseAmount**: 100 BUSD

**maxAmountPerUser**: 50 BUSD

whitelistedAmounts[**Alice**] = 0 BUSD, whitelistedAmounts[**Bob**] = 200 BUSD,

whitelistedAmounts[**Carol**] = 500 BUSD,

Alice’s tier is Star, Bob’s tier is Popular. And Carol’s tier is N/A.

IDO starts at 2022-09-01 00:00 AM.

From 00:00AM-08:00AM, only tiers can fund.

Alice can fund up to 100\*1 BUSD, Bob can fund up to 100\*5 BUSD, and Carol can fund nothing.

Alice funds 100 BUSD and Bob funds 400 BUSD.

Alice’s total funded amount is 100 BUSD, Bob’s total funded amount is 400 BUSD and Carol’s total funded amount is 0.

So, total funded amount of this IDO is 500 BUSD.

From 08:00AM-16:00AM, only whitelisted users can fund.

**Alice** can fund up to 0, Bob can fund up to 200 BUSD, and **Carol** can fund up to 500 BUSD.

**Bob** funds 100 BUSD.

Bob’s total funded amount is 500 BUSD, and total funded amount of this IDO is 600 BUSD. So whitelisted amount of Carol is 500 BUSD, but Carol can fund up to 400 BUSD now, as total target fund amount of this IDO is 1000 BUSD. (remaining = 1000 BUSD – 600 BUSD = 400 BUSD)

Carol funds 300 BUSD.

Alice’s total funded amount is 100 BUSD, Bob’s total funded amount is 500 BUSD and Carol’s total funded amount is 300.

So, total funded amount of this IDO is 900 BUSD. Only 100 BUSD is remaining.

From 16:00AM-00:00AM, any users can fund up to 50 BUSD.

Alice funds 50 BUSD, Bob funds 50 BUSD.

This IDO is 100% funded, and Carol can’t fund.

Finally, Alice’s total funded amount is 150 BUSD, Bob’s total funded amount is 550 BUSD and Carol’s total funded amount is 300.

Alice’s SEG token amount is 150 \* 5 = 750 SEG, Bob’s 550 \* 5 = 2750 SEG and Carol’s 300 \* 5 = 1500 SEG.

It’s 2022-09-11 and project owner sends 5000 SEG to this IDO contract.

IDOFactory owner calls “finalize” function.

It’s 2022-09-12, and users can start claim.

Alice’s claimable amount is 750 \* 0.2 = 150 SEG, Bob’s 2750 \* 0.2 = 550 SEG, Carol’s 300 SEG.

Alice and Bob claim their SEG token. After that, Alice and Bob’s claimable SEG are 0, and Carol’s 300.

It’s 2022-09-15, and cliff starts.

Alice’s claimable amount is 0, Bob’s 0, Carol’s 300.

It’s 2022-09-23

Alice’s claimable amount is 0, Bob’s 0, Carol’s 300.

It’s 2022-09-24

Alice’s claimable amount is 0 + 750 \* 0.4=300 SEG, Bob’s 0 + 2750 \* 0.4 = 1100 SEG, Carol’s 300 + 1500 \* 0.4 = 900 SEG.

It’s 2022-09-26

Alice’s claimable amount is 0 + 750 \* 0.4=300 SEG, Bob’s 0 + 2750 \* 0.4 = 1100 SEG, Carol’s 300 + 1500 \* 0.4 = 900 SEG.

Bob claims.

After that, Alice’s claimable amount is 0 + 750 \* 0.4=300 SEG, Bob’s 0, Carol’s 300 + 1500 \* 0.4 = 900 SEG.

It’s 2022-10-01

Alice’s claimable amount is 300 + 750 \* 0.4=600 SEG, Bob’s 0 + 2750 \* 0.4 = 1100 SEG, Carol’s 900 + 1500 \* 0.4 = 1500 SEG.

Alice, Bob and Carol claims.