

Polylaunch

Safer, fairer Web3 fundraising

True Stakeholding: supporters receive a functional NFT Venture Bond with unique privileges

Safe: DAO-based capital management prevents rug pulls

Yield Bearing: locked stablecoin funds are invested into yield bearing protocols through a Polyvault

Aligned Incentives: based on transparency and trust



Problems

Rugs cut both ways:

- Scammers launch credible-looking projects, raise capital and then stop working
- Early investors dump liquid tokens on retail investors at launch

Honest projects raise capital, launch, then fail to deliver

Early investors cannot liquidate their vested assets if they want to move on

Solution

Safer, fairer token fundraising, inspired by the DAICO

- Supporters invest a stablecoin (like Dai) to receive a **venture bond**, a functional NFT
- Venture bond holders are streamed purchased project tokens over time
- Developers are streamed raised funds over time
 meanwhile, locked assets can earn yield in a yield bearing protocol via a Polyvault
- Venture bond holders can vote to collectively revoke project funding and reclaim invested Dai
- Or they can sell their venture bond to exit investment position while still vesting, without dumping tokens on market

Roles



Launchers

Honest projects looking to raise capital

Initiate Polylaunch: transfer project tokens to the Polylaunch contract with launch parameters

Supporters invest stablecoins; reaching soft cap means project is funded

Launchers withdraw Dai on a schedule, which they can vote to accelerate: secure, automated treasury management

Supporters

Invest stablecoin to purchase project tokens locked in the contract

If soft cap is reached, Supporters can claim an NFT **venture bond**

Venture bond entitles the holder to a vesting stream of project tokens and governance rights

Venture bond can be traded on secondary market, or bond holders can vote to revoke funds



Venture Bonds

A new DeFi primitive: functional, financial NFTs with governance rights

ERC721 tokens: composable with other NFT protocols, and governance and vesting tokens are transferable with the venture bond

Supporters can keep the Venture Bond NFT as a collectible - evidence of their early support for a project



Status

Anonymous Web3 product studio has built the dapp

Solidity contracts are nearly ready for audit, currently undergoing peer review

Front end is developed, contract connections pending

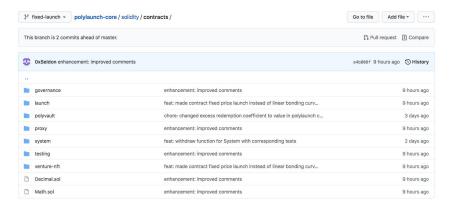
Next: token economics, business development, branding, project and artist onboarding

More details in the Polylaunch Protocol whitepaper: https://hackmd.io/@polylaunch/BJxaI06HO

Smart Contracts

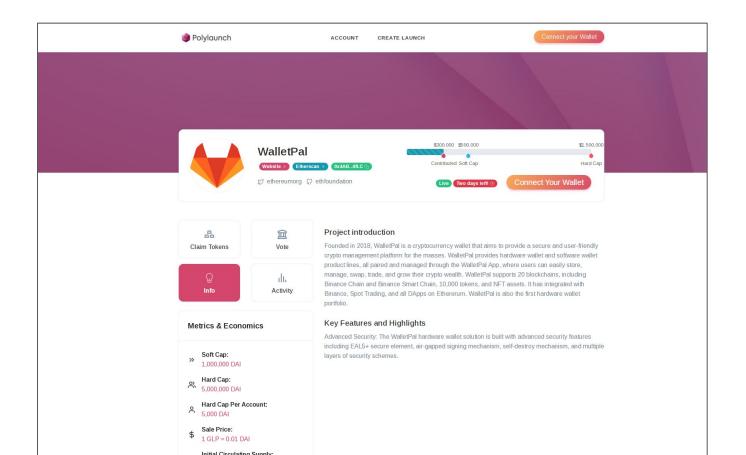


```
22 /**
19 * @title A tradable tap access control token system, with perpetual equity to creators
                                                                                                                      23 * @author PolyLaunch Protocol
20 * @notice This contract provides an interface to mint a venture bond with a market
                                                                                                                      24 * @title Basic launch
21 * owned by the creator, market to be modified.
                                                                                                                      25 * @notice A PolyLaunch DAICO launch contract following an IBCO/Dynamic swap pool mechanism
22 */
                                                                                                                            * @dev should add a sweeper function to withdraw tokens accidentally sent into the contract
23 contract VentureBond is IVentureBond, ERC721Burnable, ReentrancyGuard {
        using Counters for Counters. Counter;
                                                                                                                      28 contract BasicLaunch is PolyVault, ReentrancyGuard {
        using SafeMath for uint256;
                                                                                                                      29
                                                                                                                               using SafeERC20 for IERC20;
26
        using EnumerableSet for EnumerableSet.UintSet;
                                                                                                                               using SafeMath for uint256;
                                                                                                                      30
                                                                                                                               using LaunchUtils for LaunchUtils.Data;
28
        /sk stotokololok
                                                                                                                      32
                                                                                                                               using LaunchRedemption for LaunchUtils.Data;
29
        * Globals
                                                                                                                               using LaunchGovernance for LaunchUtils.Data;
30
        * Hotokskolotok
                                                                                                                      34
                                                                                                                               using LaunchVault for LaunchUtils.Data;
                                                                                                                      35
32
                                                                                                                      36
                                                                                                                               using VentureBondDataRegistry for VentureBondDataRegistry.Register;
33
        // Address for the market
                                                                                                                      37
34
        address public marketContract;
                                                                                                                      38
                                                                                                                               // storage struct for all information pertaining to a particular Launch
35
        // Address for the market
36
        address public launchContract;
                                                                                                                      39
                                                                                                                               LaunchUtils.Data self:
        // Deployment Address
                                                                                                                      40
38
        address public deployer;
                                                                                                                               VentureBondDataRegistry.Register register;
                                                                                                                      42
```



Interfaces







Future

Regulatory compliance: third-party KYC

Venture bond composability in other DeFi / NFT protocols

Choice in sale model: Dutch auction, various bonding curves

Venture options: collectible ERC721 call options on project tokens, with governance rights

Public goods funding / philanthropy - donors receive an NFT collectible with governance rights but no token stream



Get involved



@polylaunchorg



contact@polylaunch.org