HALALEND PROTOCOL WRAP-UP

Halalend lending protocol mimics all the features of the currently existing lending protocols on Cardano except the interest paid by borrowers on loan repayment. Loans are repaid at zero interest rate. Initially, the protocol will be the sole lender. Loans will be issued out in stablecoins from the protocol's treasury. This treasury will be seeded/bootstrapped by the community through a token sale. This means that the protocol will maintain a finite amount of liquidity and limited loan threshold. Ultimately, the protocol is aiming to be the de facto hub for quick, short term loans.

HALALEND TRANSACTION FLOW

This section outlines all transactional aspects of the product that involves money flow in and out of the protocol.

1. Halalend Native Token Sale:

The protocol will have a native utility token which will use a token sale. This sale/fundraiser will be used to fund development of the product, as well as seed the stable coin treasury (The stablecoin treasury is a pool where loans are issued from). The native token will have the following utility in the protocol.

A. Serve as part collateral asset.
B. Bearer of protocol fee distribution.
C. Governance.

Part Collateral Asset: The protocol native token will be a required asset for taking loan from the protocol. A certain percentage of the token will be required as collateral for creating a loan position.

Bearer of protocol fee distribution: A percentage of fees generated from the protocol will be distributed to stakers of the protocol’s native token.

Governance: The tokens will be used as a means of achieving consensus within the protocol. Consensus is required to change various protocol parameters like collateral ratio for loans, liquidation threshold, etc.

2. Creating A Loan Position:

This will require a user to provide the following.

a. Intending Loan Amount
b. Collateral Asset A (In ADA)
c. Collateral Asset B (In our native token)
d. Specify Loan Term (Duration).

When all the above information is provided, the user will sign a transaction using his wallet. The transaction will lock the collateral assets in a secure smart contract, deduct protocol usage fee, and credit the user's wallet with the specified loan amount in USD. The user pays blockchain transaction fee and protocol usage fee.

NOTE: This action generates income for the protocol in the form of service fee (Protocol usage fee).

3. Repaying A Loan:

This will require a user to provide the exact amount owed to the protocol without any increment. When the amount is provided and authenticated, the user will sign a transaction with their wallet. The transaction will deduct the amount specified to be repaid, unlock collateral and deposit collateral into the user wallet, deduct protocol usage fee from the user wallet.The user pays blockchain transaction fee and protocol usage fee.

NOTE: This action generates income for the protocol in the form of service fee (Protocol usage fee).

4. Liquidation:

Liquidation is a safety measure that is employed to keep the protocol solvent at all times. In any case where the user's collateral falls below the protocol's liquidation threshold or the loan term expires, the protocol will sell all the user's locked collateral, repay back the loan pool the exact borrowed amount and deposit any remaining balance to the user's wallet. This action will attract a protocol usage fee.

The protocol will adopt an instant liquidation mechanism. In the event where the liquidation threshold is breached or a loan term matures, the protocol's smart contracts will automatically initiate the immediate liquidation process. The process involves the protocol's smart contracts to place market orders to sell the borrower's collateral at the prevailing market price as soon as liquidation is triggered. So basically, each active loan (debt) contract will be programmed to handle liquidation. This process is further streamlined below:

- ➢ ***If Collateral hits liquidation threshold or below, Smart contract should perform the following liquidation actions:***
    - Unlock collateral, and place a sell order of collateral assets at market prices.
    - Repay the exact borrowed amount into the lending pool.
    - Deduct liquidation fee.
    - Refund the borrower any remnant/change after all debts and fees are settled.
- ➢ ***If Loan exceeds it's term of maturity, Smart contract should perform the following liquidation actions:***
    - Unlock collateral, and place a sell order of collateral assets at market prices.
    - Repay the exact borrowed amount into the lending pool.

- Deduct liquidation fee.
- Refund the borrower any remnant/change after all debts and fees are settled.

NOTE: This action generates income for the protocol in the form of service fee (Protocol usage fee).

The above actions are the major transactions that make up the lending protocol. The fees generated from the protocol will be split (in percentages) among the following entities.

a. Certain percentage of the fee goes to our native token stakers.
b. Certain percentage of the fee goes for protocol maintenance.
c. Certain percentage goes back to the lending pool (To increase loan liquidity).

PROTOCOL PARAMETERS

This refers to the configurable variables and settings that govern the protocol's behavior, including collateralization ratios, liquidation thresholds, loan terms, reserve ratios, etc. These parameters are essential for defining the protocol's rules and risk management. The following parameters are essential for our model:

- **Collateralization Ratios:** Thresholds that determine the minimum value of collateral required for borrowing. Our protocol will require both a certain percentage of our native token and ADA as collateral, and must sum up to a certain amount to be able to take loan.
- **Liquidation Threshold:** The collateralization ratio at which a borrower's position becomes eligible for liquidation.
- **Minimum Loan Amounts:** The minimum loan size that can be borrowed on the platform.
- **Loan Terms:** This specifies the loan duration i.e the maximum time duration for loans. This will be fixed across all loans in our platform. However, long-term, we should implement an algorithm that rebalances loan terms based on available protocol owned liquidity which means, if the protocol has enough loan-able balance in its reserve, then relatively long-term loans can be issued and if the protocol has thin liquidity, the system should only issue short term loans (We need to define long-term and short-term loans in relation to time-limits).
- **Liquidation Penalty:** A fee or penalty imposed on borrowers when their positions are liquidated, acting as a deterrent against under-collateralization (We need to examine this liquidation penalty closely and conclude if it is permissible (halal) or not).
- **Reserve Threshold:** Levels at which reserve pools (protocol owned liquidity pool) trigger actions, such as loan terms, based on protocol's financial health.
- **Reserve Ratios:** Parameters related to allocation of assets into reserve pools, which can be used for system stability, yield farming, or other purposes.
- **Governance Token Parameters:** TBD (includes staking requirements, voting power, and protocol fee distribution).
- **Maximum Loan-To-Value (LTV) Ratios:** The highest ratio at which borrowers can borrow relative to the value of their collateral.

This is basically the first product that will precede many components to come like yield generating fixed deposits, profit sharing partnerships, etc.