Project Brief: P2P Marketplace on Starknet

1. Project Overview

This document outlines the requirements for a peer-to-peer (P2P) marketplace implemented as a series of smart contracts on the Starknet network. The platform's primary goal is to facilitate secure transactions for physical goods by incorporating a third-party validation system. All transactional logic, escrow services, and user roles will be managed by the smart contracts. The development will be done exclusively in Cairo.

2. User Roles

The platform will have three distinct user roles, each with specific permissions and responsibilities within the smart contract logic:

- **Seller:** A user who can post a product for sale on the platform. The seller must provide details of the product, including its price, description, and physical location.
- **Buyer:** A user who can browse and initiate the purchase of a product posted by a seller.
- **Validator:** A neutral third-party user responsible for physically inspecting the goods. Validators will be compensated for their service.

3. Core Features and Transaction Flow

The smart contract system must manage the following sequence of events:

1. **Product Listing:** A seller posts a product for sale, specifying its price, description, and location. This data is stored on-chain.
2. **Purchase Initiation:** A buyer selects a product and initiates a purchase.
3. **Escrow & Fee Calculation:** The buyer's payment, in the form of STARK tokens, is immediately moved into a secure escrow managed by the smart contract. A validator fee of 2% of the product's price is automatically added to the total amount in the escrow. The final price in escrow is Product Price + (Product Price * 0.02).
4. **Validator Selection:** Upon a successful payment to the escrow, the system will automatically select an available validator. The selection criteria are:
    - The validator must be active (e.g., have an active status flag).
    - The validator must be in proximity to the seller's specified location. The smart contract should use a basic geo-location mechanism or a pre-defined region system to determine proximity.
5. **Validator Task Assignment:** The chosen validator receives a notification or an

on-chain task assignment with the product details and the seller's location information.

6. **Physical Validation:** The validator physically inspects the goods.
7. **Validator Decision:** The validator submits a transaction to the smart contract with a decision:
    - **Approve:** The product is as described.
    - **Disapprove:** The product does not meet the description.
8. **Buyer's Final Decision:** After the validator's decision, the smart contract allows the buyer a final option to proceed. The buyer can:
    - **Continue with Purchase (Validator Approved):** The funds are released from the escrow to the seller, and the validator's 2% fee is released to the validator.
    - **Cancel Purchase (Validator Approved/Disapproved):** The funds (minus a fixed amount of $3) are returned to the buyer. $3 is released to the validator as compensation for their service. This ensures the validator is always paid for their work, regardless of the final outcome.

4. Starknet Wallet Requirements

The system will require users to interact with a Starknet-compatible wallet. The smart contract and any associated front-end logic must support the following wallet functionalities:

- **Fund Management:** The ability for users to send STARK tokens to other Starknet addresses.
- **Token Swapping:** Integration with a protocol like an "Autoswapper" to allow for seamless conversion between STARK and other tokens (e.g., USDT). This could be a separate smart contract interaction.
- **Two-Factor Authentication (2FA):** A mechanism to add an extra layer of security for transactions, requiring a second form of verification. This will be implemented as a feature of the wallet contract itself, allowing users to enable or disable it.

5. Technical Stack

- **Blockchain:** Starknet
- **Smart Contract Language:** Cairo

Ribawheels - Updated Hackathon Brief Addendum

The following updates have been made to the original project brief:

1. Validator Rating System
- After each completed transaction, buyers can rate validators (1–5 stars).
- Ratings are stored on-chain and used to calculate an average score.

2. Rigorous Validator Vetting
- Validators must apply with details (name, location, contact info, experience).
- Status is initially 'Pending'. Platform admin can approve or reject.
- Only 'Approved' and active validators can be assigned to orders.

3. Buyer Choice of Validator
- Buyers select their preferred validator from a list of approved and active validators in the product's location.
- Validator list includes ratings and fee percentages.

4. Dynamic Validator Fees Based on Rating
- Validator fee is determined by rating tier:
• Rating $\geq 4.5 \rightarrow$ 3% fee
• Rating ≥ 3.0 and $< 4.5 \rightarrow$ 2% fee
• Rating $< 3.0 \rightarrow$ 1% fee
- Fee is added to the escrow amount.

5. Payments
- Platform supports STARK-only escrow for all transactions.

This addendum reflects the new functional requirements and supersedes corresponding sections in the original b