WORKFLOW

CHARITY DONOR TRACKING SECURITY SYSTEM

This project appears to be a **Charity Donation Platform** where donors can connect with receivers in need, contribute Ethereum (ETH), and help fulfill financial requests. Below is a **step-by-step workflow** of the application:

1. User Types and Roles

a. Donor

- Connects their MetaMask wallet.
- Views donation requests from receivers.
- Selects a request and donates ETH to a receiver.
- Donation details are recorded in the database.

b. Receiver

- Registers their request by providing personal details, wallet address, and a reason for requesting funds.
- Monitors the status of their request (pending/approved).

2. Key Features

a. MetaMask Integration

- **Donor Dashboard**:
    - Donors connect their MetaMask wallet via `window.ethereum.request`.
    - Wallet address is stored for tracking purposes.
- Used for authenticating and facilitating donations.

3. Project Workflow

A. General Navigation

1. **Home Page (`index.php`)**
    - Presents two options:
        - **Donor**: Redirects to `donor.php`.

Receiver: Redirects to `receiver.php`.

B. Donor Workflow

1. **Donor Dashboard (`donor.php`)**
   - o Connects MetaMask to authenticate the donor.
   - o Displays all pending requests retrieved from the `requests` table.
2. **Donation Process**
   - o Donor selects a request and submits the donation using the form in `donor.php`.
   - o The donation form submits data (`receiver_id`, `wallet_address`, and `amount_needed`) to `donate.php`.
3. **Transaction History (`amthistory.php`)**
   - o Displays all recorded donations from the `transaction_history` table.
   - o Allows donors to view details of their contributions.

C. Receiver Workflow

1. **Receiver Request Form (`receiver.php`)**
   - o The receiver submits a request including:
     - ▪ Name, wallet address, reason for funds, and required amount.
     - ▪ Aadhar card (ID proof).
   - o The request is stored in the `requests` table with a **Pending** status.
2. **Request Status Check (`receiveramt.php`)**
   - o Displays the status of the request (pending/approved).
   - o If **Approved**, the receiver can withdraw funds.

D. Backend Workflow

1. **Database Connection (`db_connection.php`)**
   - o Establishes a connection to the MySQL database.
   - o Creates the database (`charity_connect`) and necessary tables (`requests` and `transaction_history`) if they don't exist.
2. **Request Tables**
   - o **`requests` Table**:
     - ▪ Stores receiver requests.
     - ▪ Tracks status (Pending/Approved).
   - o **`transaction_history` Table**:
     - ▪ Logs donations, including receiver address, amount, and transaction status.

4.Workflow Diagram

```
   Home Page
   |
+---> Donor Dashboard (`donor.php`)
|     |
|     +-- View Requests
|     |   |
|     |   +-- Select Request -> Donate
|     |
|     +-- View History -> `amthistory.php`
|
+---> Receiver Dashboard (`receiver.php`)
      |
      +-- Submit Request
      |
      +-- View Status -> `receiveramt.php`
```

5. Frontend Components

- **HTML & CSS**: Used for structuring and styling the UI.
- **JavaScript**:
  - Handles MetaMask integration and front-end validation.
  - Implements user interactivity such as opening forms and toggling elements.

6. Blockchain Integration

- **MetaMask**: Provides seamless connection to Ethereum wallets for secure transactions.
- **ETH Donations**:
  - The `donateButton` and MetaMask ensure that donations are handled securely.

7. Key Functionalities

1. **Donors**:
   - View and donate to requests.
   - Track donation history.
2. **Receivers**:
   - Submit requests for financial help.
   - Monitor approval status.
   - Withdraw approved donations.
3. **Admin/Database**:
   - Automates request and transaction logging.
   - Ensures data integrity and user actions are properly tracked.

This system combines **web technologies**, **blockchain (MetaMask)**, and **database management** to create a platform that facilitates seamless, transparent charitable donations.