

Project Brief: Cookie Jar App - UI MVP

Project Overview

The Cookie Jar app is a decentralized digital petty cash system allowing users to create, fund, and manage shared financial pools ("cookie jars") with controlled access and withdrawal limits.

Platform Support

- Primary focus on desktop experience
- Responsive design for mobile devices (leveraging Next.js capabilities)
- Modern browser compatibility

Design Direction

- Clean, simple interface without being visually boring
- No specific color scheme requirements
- Prioritize usability and intuitive interactions

Core Functionality

- Create and fund cookie jars with ERC20 tokens or native blockchain assets
- Configure access controls through address whitelisting
- Set withdrawal limits and cooldown periods
- Search for accessible or owned cookie jars
- Withdraw funds from jars when eligible
- Manage jar settings as an owner

UI Structure

1. Landing Page

A simple welcome page featuring:

- App name/logo
- Three prominent buttons:
 - "Search for Cookie Jars"
 - "Create a Cookie Jar"
 - "Manage My Jars"
- Optional wallet connection status indicator in header

2. Jar Search Page

Search interface featuring:

- Search field for EVM address input
- Search button
- Wallet connection button (prominent if wallet not connected)
- Search results area (initially empty)
- When search is performed, the results area will display:
 - Cards grid/list showing eligible jars:
 - Jar title
 - Token type
 - Current balance
 - Maximum withdrawal amount
 - Cooldown period
 - Total claims made
 - Contract address (shortened)
 - Network
 - "View Details" button

- Empty state message when no jars are found: "Looks like the provided address doesn't have access to any cookie jars 😞"
- "Search for a new address" option
- "Create a Jar" button when empty
- Back button to return to landing page

3. Jar Creation Page

Form with the following input fields:

- Jar owner address field (EVM compatible)
- Jar title field
- Cooldown period input (with time unit selector)
- Cookie token contract address field
- Network selection dropdown (options: Mainnet, Optimism, Base, Arbitrum, Cello, Gnosis, Unichain)
- Maximum withdrawal amount field (basic implementation for MVP)
- Whitelisted addresses input area (with validation):
 - Text area for manual input
 - CSV upload option
- Initial funding amount field (using the token specified in cookie token contract address)
- Donation options:
 - Checkbox to donate to the project (pre-selected)
 - Donation percentage field (default 5% of initial funding amount)
 - Option to adjust percentage or opt out
 - Donation will use the same token as the cookie jar token
 - Total transaction amount indicator (e.g., "Total: 105 USDC = 100 USDC to jar + 5 USDC donation")
- "Create Jar" button (prominent)

- Success animation area (for cookie confetti)
- Back button

5. Cookie Jar Detail Page

Expanded view shown when view details button is pressed from jar search page featuring:

- Jar title
- Token type (cookie token contract address)
- Current balance display
- Maximum withdrawal amount
- Cooldown period
- Network
- Contract address with copy and block explorer link (clickable to open in relevant block explorer)
- Total claims made
- Claim history section (list of past withdrawals)
- Withdrawal section (if eligible and cooldown period has passed):
 - Amount input field with slider
 - Claim reason text field (minimum 20 characters required)
 - "Withdraw" button
- "Add Funds to Jar" section:
 - Amount input field (must be same token as jar already contains)
 - Donation options:
 - Checkbox "Donate funds to the cookie jar team to maintain this project" (pre-selected)
 - Donation percentage field (default 5% of added funds amount)
 - Option to adjust percentage or opt out
 - Donation will use the same token as the jar token

- Total transaction amount indicator (e.g., "Total: 105 USDC = 100 USDC to jar + 5 USDC donation")

- "Add Funds" button

- Back button

6. Manage All My Cookie Jars Page

Owner management interface featuring:

- Wallet connection status

- Search field for address input

- Cards grid/list showing owned jars:

- Jar title

- Token type

- Current balance

- Maximum withdrawal amount

- Cooldown period

- Total claims made

- Contract address (shortened)

- Network

- "Manage Jar" button

- Empty state message when no owned jars are found

- "Create New Jar" button

- Back button

7. Manage My Individual Jars Page

Administrator view shown when "manage my jar" button is pressed featuring:

- Jar title

- Token type (cookie token contract address)
- Current balance display
- Maximum withdrawal amount (editable)
- Cooldown period
- Network
- Contract address with copy and block explorer link
- Total claims made
- Claim history section (list of past withdrawals)
- Whitelist management interface:
 - Current whitelist display
 - Add address field
 - Remove address option
 - Bulk upload option
- Action buttons:
 - "Add Funds" (opens a modal with):
 - Amount input field (must be same token as jar already contains)
 - Donation options:
 - Checkbox "Donate funds to the cookie jar team to maintain this project" (pre-selected)
 - Donation percentage field (default 5% of added funds amount)
 - Option to adjust percentage or opt out
 - Donation will use the same token as the jar token
 - Total transaction amount indicator (e.g., "Total: 105 USDC = 100 USDC to jar + 5 USDC donation")
 - "Confirm" button
 - "Add Owners"
 - "Remove All Funds"

- "Save Changes"
- Transaction status indicators
- Back button

Key UI Components

- EVM address input fields (with validation)
- Token amount inputs
- Card components for jar display
- Modal confirmations for transactions
- Transaction status indicators
- Network-aware block explorer links
- Wallet connection interface
- Form validation messaging
- Character-counting input field for claim reasons
- Donation calculators showing total transaction amounts

User Flows

1. **Search Flow**: Landing → Search → View Results → View Details
2. **Creation Flow**: Landing → Create → Fill Form → Confirm Transaction → Success
3. **Management Flow**: Landing → Manage → Select Jar → Modify Settings → Save
4. **Withdrawal Flow**: View Details → Enter Amount → Provide Claim Reason → Withdraw
5. **Fund Addition Flow**: View Details/Manage → Enter Amount → Confirm Donation Option → Add Funds

Technical Considerations

- Wallet connection integration

- EVM address validation
- Transaction confirmation and status handling
- CSV parsing for whitelist uploads
- Character count validation for claim reasons
- Real-time calculation of donation amounts based on percentages

Known Implementation Challenges

- ****Token Decimal Handling****: The maximum withdrawal amount field needs to account for different token decimals based on the selected token contract. This is acknowledged as an unsolved problem and will be addressed in a future implementation phase.

Component Organization

The UI should be organized as a single-page application with different views rather than separate pages where possible. The navigation between "pages" should be smooth and intuitive, maintaining context throughout the user journey.

This brief outlines the key UI elements needed for the Cookie Jar MVP. The actual implementation should prioritize simplicity and usability while maintaining all core functionality.