Project Cryptela

(MVP)

**SUMMARY**

Objective:

1. Create a browser plugin that displays the following output:
   1. A user’s portfolio of crypto currency
2. Create the backend functions to support the output

Structure of project (in order):

1. Minimum Viable Product (MVP)
2. Remaining database
3. Remaining outputs
4. Additional items

High level details

1. MVP Data flow:

Firefox plugin application

Basic retrieve and compute function

Price database from UniSwap

1. Remaining databases - sourced from following crypto exchanges (where possible):
   1. Top 10 from the following website: <https://coinmarketcap.com/rankings/exchanges/>
   2. All database prices to be simple averaged to produce a single price per unit time – this average price is the one used by the program
2. Remaining outputs – output features will be identical:
   1. Plugins for Chrome, IE, Safari, Edge
   2. Website for Desktop and Mobile
   3. Mobile app for Android and Apple
3. Additional items – in-house development
   1. Remaining exchanges from the following website: <https://coinmarketcap.com/rankings/exchanges/>
   2. To be scanned for transactions – All other websites that perform the same function as Etherscan.io for all Coins (list not comprehensive)
      1. E.g. Blockchain Explorer for Bitcoin
      2. E.g. https://chainz.cryptoid.info/

**PROJECT DETAILS**

Details

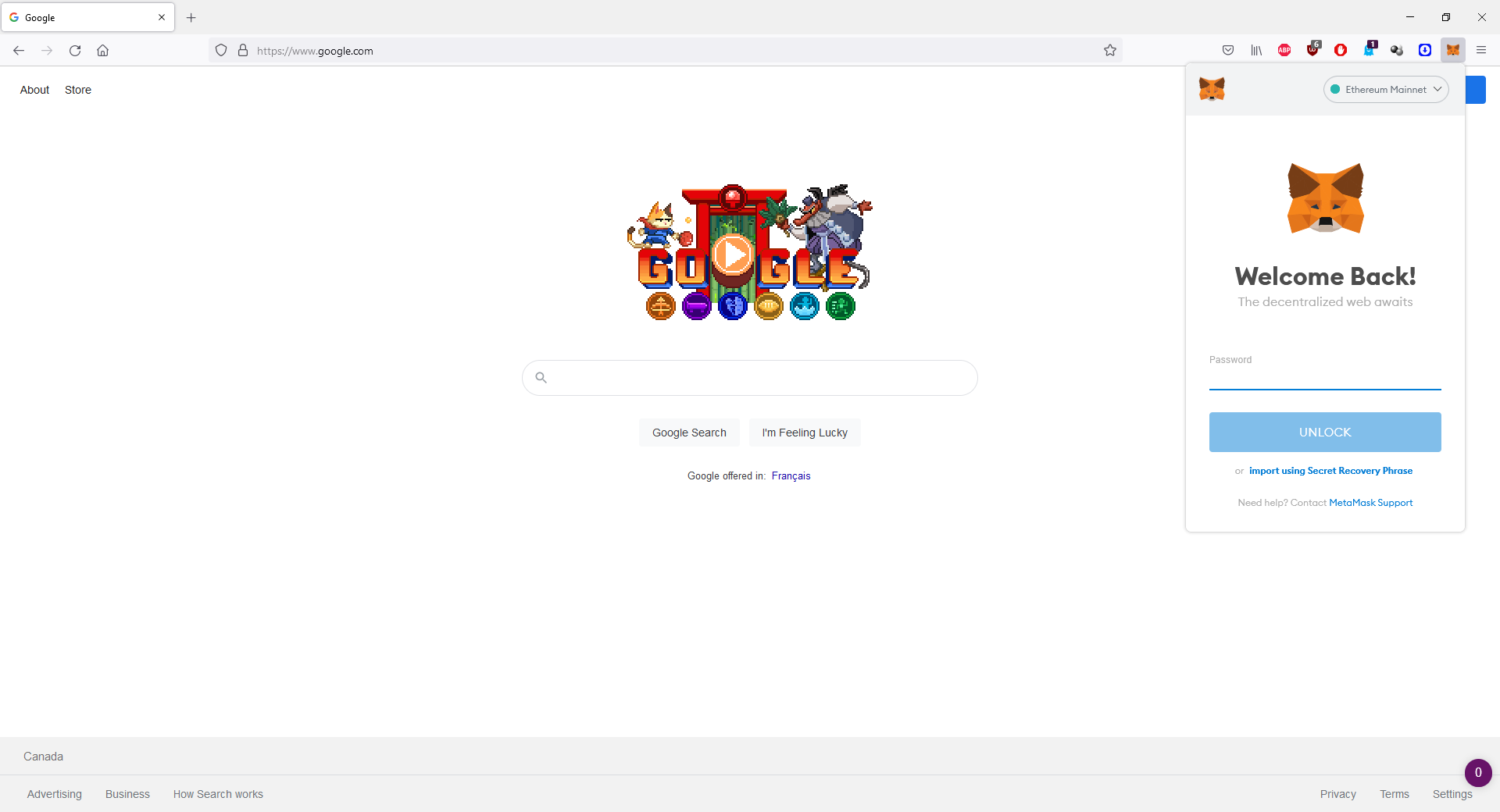
1. MPV Data flow:

Firefox plugin application

Basic retrieve and compute function

Price database from UniSwap

* Price database from Uniswap:
  1. API (if one exists/can be created) from Uniswap database
  2. Download historical prices to AWS (or some other data storage environment)
     1. Only spot prices
        + All prices are vs. USDC
     2. Coin tickers
     3. Timestamps for spot prices
        + All timestamps to be converted to GMT +0 hrs
* Basic retrieve and compute function (read “Firefox plugin application” section first)
  1. When Complete button is pressed, program will take user’s wallet address(s) and scan Etherscan.io for transactions that have utilized the wallet
  2. Program will download following transaction details into server-side temporary storage
     + Coin ticker
     + Timestamp of transaction
     + Direction of transaction (into wallet or out of wallet)
     + Volume of coins in transaction
  3. Program will cross reference Timestamp of transaction against Timestamps for spot prices
  4. Direction of transaction will determine signage (+/-) of transaction (into wallet = positive, out of wallet = negative)
  5. Perform simple addition of inflow vs outflow of wallet by Coin ticker
  6. Push information into plugin
  7. Repeat for each Coin ticker
  8. Repeat for each user wallet
* Firefox plugin application
  1. Approximate size of target plugin display – see Metamask window (resize horizontally such that there is no horizontal scroll after all development is complete)



* 1. 1st screen: Requesting user’s public key
     + [“Wallet 1”] [Input textbox] [Add button]
     + “Add” button to the right of input area
       - Add button triggers 2nd input area to appear below 1st input area
       - Repeat
       - “Complete” button as a footer, center justified
  2. 2nd screen:
     + 1st row: icons for “Refresh” and icon for “Export to PDF” in order
       - Refresh: refresh data from server
       - Export to PDF: Opens a new PDF document. Table in PDF sorted according to current display. Format of PDF –



* + - 2nd row (vertical scroll section) first tab (Tab name: “Summary”)
      * Displays information from retrieve and compute function
      * All transactions to be sorted from latest to earliest by default
      * Field headers are as follows in order from left to right:
        + Wallet name (starting with Wallet 1 from above. Display name: “Wallet”. Only show 1st 4 char “…” and last 4 char)
        + Coin ticker (Display name: “Ticker”)
        + Profit & Loss (Display name: “Profit/Loss”)
      * Each field header can be sortable from A-Z or Highest to Lowest depending on datafield
      * Total: $X,XXX,XXX.XX USDC displayed as footer, center justified
    - 2nd row (vertical scroll section) first tab (Tab name: “Details”)
      * Displays information from retrieve and compute function
      * Contains breaks called “Subtotal:” as additional lines inserted automatically
      * All transactions to be sorted from latest to earliest by default
      * Field headers are as follows in order from left to right:
        + Wallet name (starting with Wallet 1 from above. Display name: “Wallet”. Only show 1st 4 char “…” and last 4 char)

Subtotal by Wallet

* + - * + Coin ticker (Display name: “Ticker”)

Subtotal by Coin ticker

* + - * + Time (Display name: “Time (GMT+0)”)

Data display format: dd mmm yy, [hh]:[mm]:[ss]

Subtotal by Year only

* + - * + Profit & Loss (Display name: “Profit/Loss”)

No Subtotal

* + - * Each field header can be sortable from A-Z, Latest to Earliest, or Highest to Lowest depending on datafield
      * Total: $X,XXX,XXX.XX USDC displayed as footer, center justified