Data

POC requirements FREEMIUM VERSION

- Odds data collection through API and scraping
- Data cleaning and storage(server v/s on-chain)
- Arbitrage identifier logic
- Data visualisation + highlighting

Post MVP

- Bet tracker
- Pinnacle arb calculator clone

Additional POC requirements PRO VERSION A

Hedge

- Authenticate user to Polymarket CLOB API
  - L1: Private key authentication(Polygon)
  - L2: API key authentication
- Order execution
- Retrieve historical trade data
- Visualize hedge data on frontend

Additional POC requirements PRO VERSION B

Vault

- Allow user to deposit USDC tokens
- Allow user to withdraw USDC tokens

Hedge

- Bridge user funds between Solana and Polygon(deBridge widget)
- Polymarket deposit/withdrawal
- Facilitate betting through wScout proxy wallet / API key

★ **Choose between A or B(Post MVP)**

Arbitrage logic

Arb highlighting

Odds data Storage

Data visualisation

Data visualisation

Data format

1. Source ID
2. Event ID
3. Event Time
4. Sport / Market type
5. Side A
   1. Name
   2. Odds
6. Side B
   1. Name
   2. Odds
7. (Side C)
   1. (Name)
   2. (Odds)

Site A

Site B

Site C

Scraper

Raw Json data

Data cleaning

Polymarket

GET {clob-endpoint}/book

Data format

```
market:        string
asset_id:      string
hash:          string
timestamp:     string
bids           Ordersummary[]
asks           Ordersummary[]
```

Frontend

1. Alert for arb opportunity

2. Input hedge on frontend

User account

Polygon private key

Polymarket API key

Backend logic

3. Signal execution script

6. Send hedge details

Execution logic

self call

7. Create order object

4. Initiate API connection

8. Post order

9. Polymarket returns trade data

Verification

5. Verification
-If yes, send hedge details
-If no, alert user

Polymarket

POST {clob-endpoint}/order

Order object

```
salt:           integer
maker:          string
signer:         string
taker:          string
tokenId:        string
amount:         string
nonce:          string
side:           string
signatureType:  integer
signature:      string
```

L1: Private Key Authentication

```
POLY_ADDRESS

POLY_SIGNATURE

POLY_TIMESTAMP

POLY_NONCE
```

L2: API Key Authentication

```
POLY_ADDRESS

POLY_SIGNATURE

POLY_TIMESTAMP

POLY_API_KEY

POLY_PASSPHRASE
```