**Low-Level Design (LLD) Document**

**Data Handling:**

* Two CSV files with features like price, volume, market cap, and % changes
* Filled missing values with median (numeric columns only)

**Feature Engineering Steps and Importance:**

1. log\_volume: Log-transformed 24h volume
   * **Why:** Cryptocurrency trading volumes are heavily skewed. Log transformation reduces outlier impact and normalizes the scale.
2. log\_liquidity: Log(volume / market cap)
   * **Why:** Represents the core liquidity proxy. Log scaling makes it more interpretable and comparable across coins.
3. log\_mkt\_cap: Log-transformed market capitalization
   * **Why:** Market cap scales vary massively between coins. Log helps handle this variance and stabilize feature impact.
4. log\_price: Log-transformed coin price
   * **Why:** Like volume and market cap, price has a wide range. Log brings consistency to scale and reduces skew.
5. momentum\_3d: Sum of 1h, 24h, and 7d % changes
   * **Why:** Captures the short-to-mid-term price movement. Helps identify bullish or bearish trend behavior.
6. volume\_per\_price: 24h volume divided by price
   * **Why:** Indicates trading intensity per unit price. A high value means more trades for every dollar of coin value.
7. volatility\_24h: Absolute value of 24h price change
   * **Why:** Represents short-term instability. Useful for detecting unstable or less liquid assets.

**Target Creation:**

* liquidity\_class: Binary class (1 = high, 0 = low) from median split of liquidity ratio

**Model:**

* Algorithm: RandomForestClassifier
* Train-test split: 80-20
* Metrics: Accuracy, Cross-validation, Confusion Matrix, ROC-AUC

**Model Evaluation:**

* Train Accuracy: 100%
* Test Accuracy: 99.5%
* Cross-Validation Score: 99.75%
* ROC-AUC: High, curve plotted

**Streamlit App:**

* Inputs: User provides 7 features (log\_volume, log\_liquidity, etc.)
* Output: Displays prediction (High/Low Liquidity)
* Uses joblib to load .pkl model