Algorithmic Trading System Implementation using

Kite Connect

Akshay Navin

Project Assignment Goal

To develop a real-time algorithmic trading system using Zerodha’s Kite Connect API

Visualize market data, trade logs, and PnL in a live dashboard

Execute trades based on a simple technical analysis strategy **(EMA 5, 15 Crossover)**

Logs trade activity and Profit and Loss

Algo Trade Logs

2025-07-11 15:21:02 - Order placement failed: Intraday orders (MIS) are allowed only till 3.20 PM. Try placing a CNC order.

2025-07-11 15:19:11 - BUY order placed for HB. Order ID: 250711201022903

2025-07-11 15:19:05 - Position squared off for HB via BUY

2025-07-11 15:19:04 - BUY order placed for HB. Order ID: 250711201022463

2025-07-11 15:07:11 - SELL order placed for HB. Order ID: 250711200989806

2025-07-11 15:07:05 - Position squared off for HB via SELL

2025-07-11 15:07:04 - SELL order placed for HB. Order ID: 250711200989575

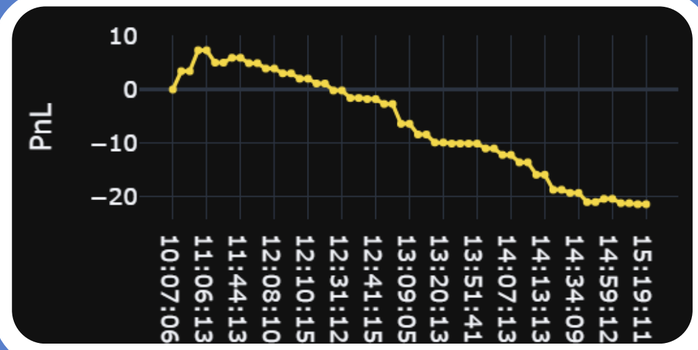
2025-07-11 14:59:12 - BUY order placed for HB. Order ID: 250711200970730

2025-07-11 14:59:06 - Position squared off for HB via BUY

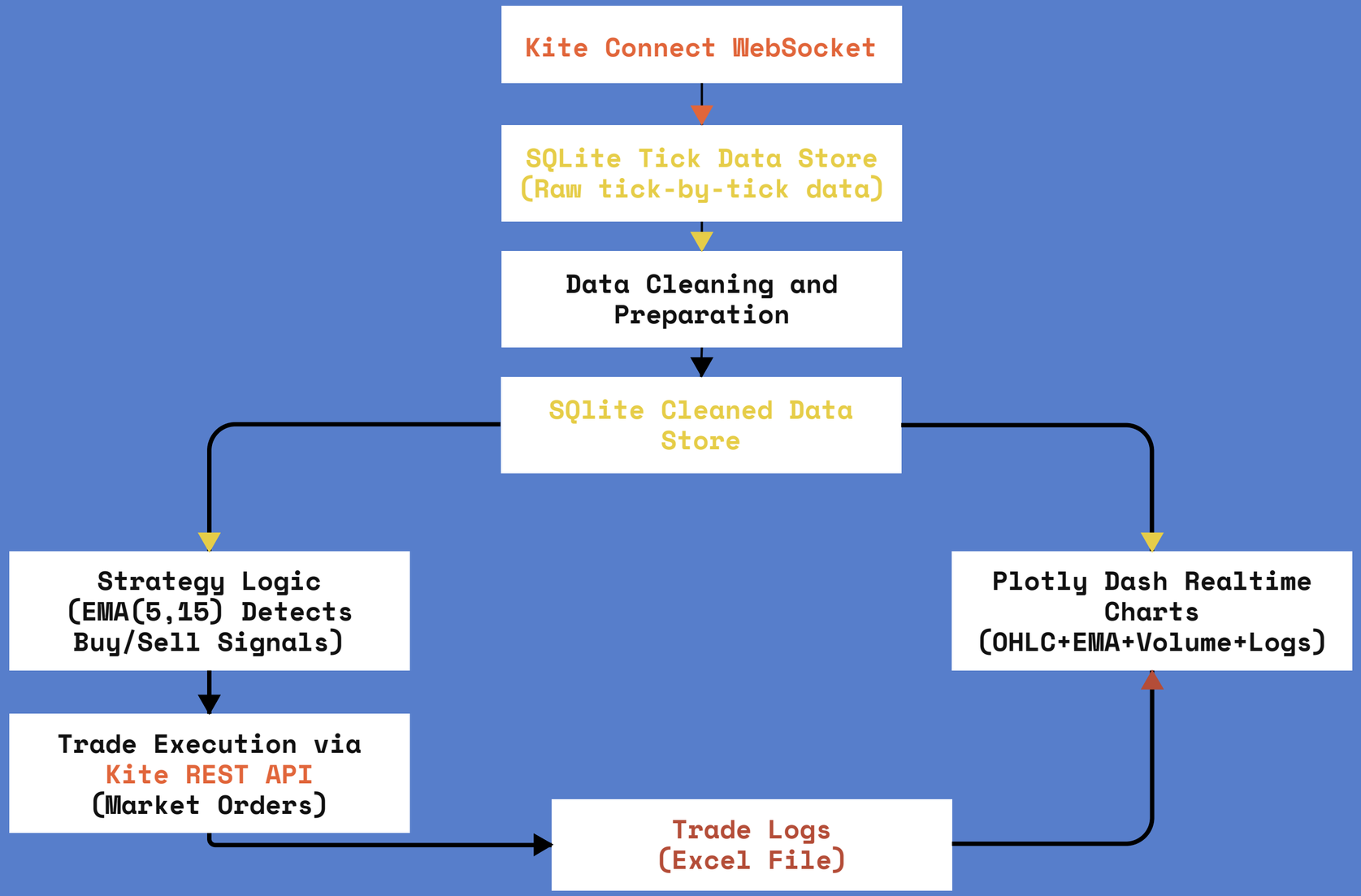
2025-07-11 14:59:05 - BUY order placed for HB. Order ID: 250711200970503

2025-07-11 14:36:11 - SELL order placed for HB. Order ID: 250711200927714

2025-07-11 14:36:05 - Position squared off for HB via SELL



Code Architecture Diagram

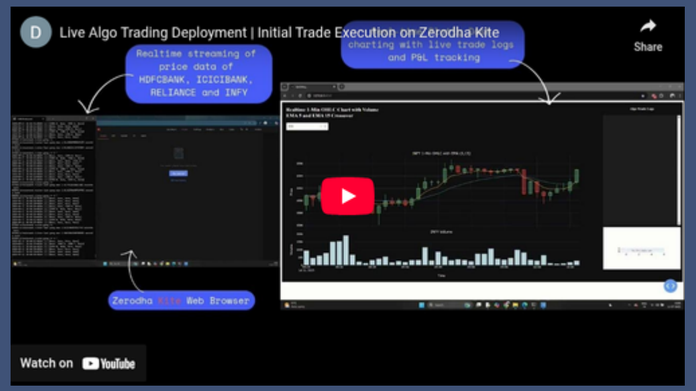


Strategy Logic and Live Trading Demo

Strategy

Simple **EMA (5,15) crossover** strategy was deployed

- EMA 5 crosses above EMA 15 - **Buy**
- EMA 15 crosses above EMA 5 - **Sell**
- When reverse signal appears on open position - **Square-off** and take new position

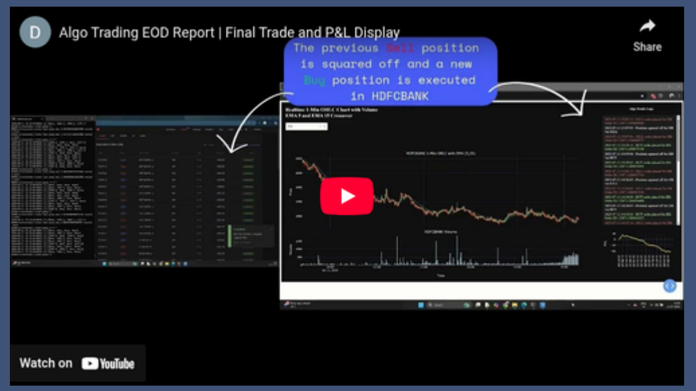


Trade Management

- Strategy was deployed over 4 stocks with code names:
  1. HB for HDFCBANK
  2. IB for ICICIBANK
  3. RS for RELIANCE
  4. ISS for INFY
- One trade per symbol at a time
- Live positions tracked via Excel file
- PnL calculated via executed prices log

Additional Features

- Trades placed and logged via Kite Connect Live
- Dashboard updates every 5 seconds



Key Modules and Code Snippets

Real-time Tick Data Capture via Kite WebSocket

```
kws = KiteTicker(api_key, access_token, user_id)
def on_ticks(ws, ticks):
    for tick in ticks:
        ltp = tick['last_price']
        bid_qty = tick['depth']['buy'][0]['quantity']
        sell_qty = tick['depth']['sell'][0]['quantity']
        volume = tick['volume_traded']
```

Live Charting with Plotly Dash

```
query = f"SELECT datestamp, `{stock}` FROM testTable"
df = pd.read_sql_query(query, conn)

ohlc = df[stock].resample('1T').ohlc()
ema5 = ohlc['close'].ewm(span=5).mean()
ema15 = ohlc['close'].ewm(span=15).mean()

fig.add_trace(go.Candlestick(...))
fig.add_trace(go.Scatter(y=ema5))
fig.add_trace(go.Scatter(y=ema15))
```

Market Order Placement

```
order_id = kite.place_order(
    variety=kite.VARIETY_REGULAR,
    exchange=kite.EXCHANGE_NSE,
    tradingsymbol=mapped_symbol,
    transaction_type=transaction_type,
    quantity=quantity,
    order_type=kite.ORDER_TYPE_MARKET,
    product=product_type)
```

Fetching Positions & Orders

```
positions = kite.positions()
orders = kite.orders()
```

Improvements / Future Scope

Refine Signal Logic:

- Use EMA 5,15 for entry and an aggressive square-off of EMA 5,10 to protect profits and limit large losses
- Place and modify stop loss to trail the price
- Use other Technical Indicators in conjunction for stronger entry/exit signals
- Add Volume breakout or Open Interest (for FnO)
- Apply the strategy to all the stocks available/chosen

Enhance Visualization:

- Show trade markers **(Buy/Sell)** live on charts
- Annotate PnL per trade
- Use a GUI like pyqtgraph instead of plotly dash to improve upon latency

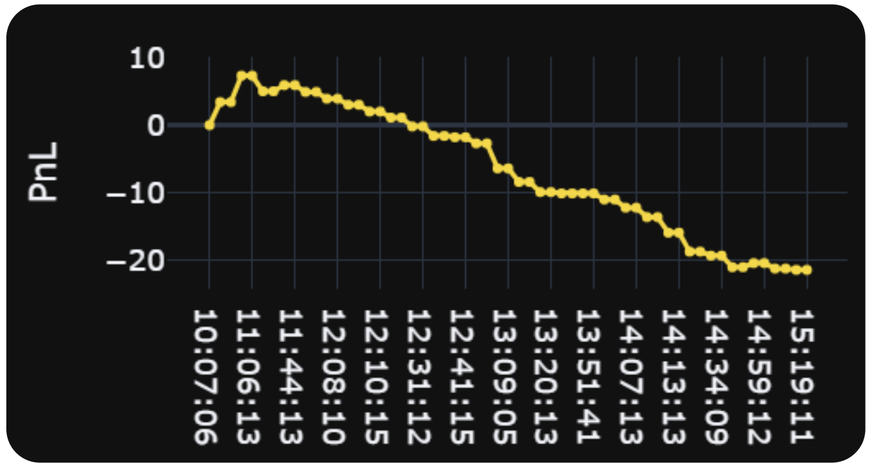
Risk Management:

- Position sizing, stop-loss, max loss per day
- Trade limit per stock

Logging:

- Use SQLite based log storage for better scalability

Final Profit and Loss



Conclusion

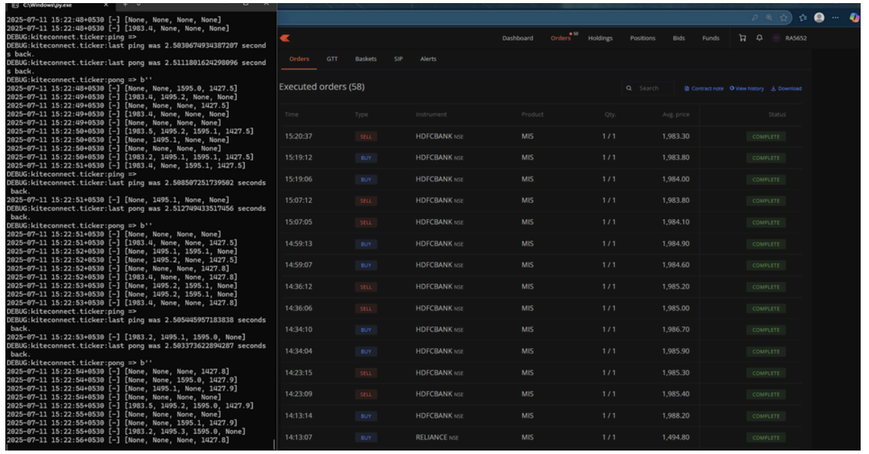
Objective:
Develop a complete fundamental and code implementation workflow using **Kite Connect API**/mock data for an algorithmic trading strategy.

Achieved:

- Live algo trading system developed using Zerodha's **Kite Connect API**, with real-time tick data streaming via **WebSocket** and signal-based market order execution via **REST API**
- Realtime tick-by-tick streaming along with data cleaning and preparation
- Realtime OHLCV charting, **EMA (5,15) crossover** with a dropdown for stock selection
- Live Algo trade logs and cumulative PnL
- Clean, modular and expandable workflow

Real-time OHLCV, EMA Crossover, Live Logs, PnL

Tick-by-Tick Stream and Orders on Kite Web Browser



Thank you!

Akshay Navin

https://dadquant.com/

https://www.youtube.com/@DadQuant

akshaynvn@gmail.com