

★ “স্টক ডেটা কোথা থেকে পাবো?”

ডকুমেন্টে ক্লায়েন্ট তিনি API সোর্স উল্লেখ করেছে:

1) Zoya API → Halal / Haram Screening

- এটা মূলত শরিয়াহ স্ক্রিনিং অ্যাপ।
- Zoya একটি Islamic investment অ্যাপ, যেটা paid API দেয়।
- **Free** নয়, 100% paid।
- তাদের API ডকুমেন্টেশন পাবলিক নয়।
→ তাদের টিমের সাথে contact করতে হয়।
- ডেটা:
 - Halal/Haram status
 - Compliance score
 - Business sector filtering

চ্যালেঞ্জ:

ক্লায়েন্ট যদি Zoya-এর "legal user" না হয় → তোমরা ইউজ করতে পারবা না।
ক্লায়েন্টকেই access নিতে হবে।

2) Musaffa API → Halal Screening + Analyst Ratings

- Musaffa Islamic Finance Platform।
- Full API দেয়:
 - Halal/Haram screening
 - Analyst Ratings (Buy/Sell/Hold)
 - Dividend info
 - ESG score
 - Price targets

২ টাই সমস্যা:

- এটা paid (free নয়)
- Access পেতে কোম্পানির subscription লাগে।

ক্লায়েন্ট নিজে বলেছে:

"Zoya/Musaffa যেটার legal user ওবা—ক্লায়েন্টকেই access নিতে হবে!"

Access client দিতে হবে।

3) TradingView / Seeking Alpha API → Stock Price Data

এগুলোতে free API নেই।

- TradingView API → paid
- Seeking Alpha → paid
- AlphaVantage → free কিন্তু খুব limited
- Finnhub → free tier আছে + paid

👉 **Recommendation:**

স্টক প্রাইস, চার্ট, real-time data এর জন্য **Finnhub + AlphaVantage** দুইটাই ধরো।
ক্লায়েন্ট যদি TradingView কিনে দেয় → বোনাস।

★ কোড আর্কিটেকচার:

🔥 Step 1: Full System Architecture

Frontend → React.js

- React.js
- Major modules:
 - Stock search
 - Halal screening result page
 - Analyst Ratings
 - Watchlist
 - Subscription page
 - VIP Chat UI
 - Finance dashboard
 - Loan calculator
 - Wealth panel

Backend → Node.js (Express)

Backend হবে মাস্টার ব্রেইন।

Backend কাজ:

- External APIs call করা (Zoya / Musaffa / TradingView)
- Database এ স্টক cache করে রাখা
- Authentication
- Subscription management
- AI based explanation
- Admin panel APIs
- VIP chat logic
- Notifications
- Schedule job (cron job) for price updates

Database → PostgreSQL

Tables:

- users
- subscriptions
- stocks
- halal_screening_results
- analyst_ratings

- `vip_chats`
 - `finance_inputs`
 - `wealth_assets`
 - `articles`
 - `notifications`
 - `admin signals (Entry, SL, Targets)`
-

★ Step 2: “Stock building”

সব ডেটা external API থেকে আসবে।

Flow:

(1) User searches → “AAPL”

- Backend request পাঠাবে Finnhub / TradingView API-তে
- Real-time data ফেচ করবে
- Local database এ 5 min cache করে রাখবে
- Frontend-এ পাঠাবে

(2) User clicks “Halal Screening”

- Backend request পাঠাবে Zoya/Musaffa API-তে
- Result DB-তে save
- AI দিয়ে explanation generate
- Frontend-এ show

(3) User checks “Analyst Ratings”

- Backend Musaffa/Seeking Alpha → analyst ratings
- “Buy / Hold / Sell”
- “Price targets”
- “Consensus rating”
- Frontend show

(4) Watchlist

- Local DB
- Periodically stock price update via Cron job

প্রতিটা জিনিস API driven।

তোমাদের কিছু build করতে হবে না, শুধু API integrate করতে হবে।

★ Step 3: API Integration Breakdown

◆ A) Halal Screening API (Zoya / Musaffa)

Backend Process:

GET /api/halal-screening?symbol=AAPL
→ Zoya/Musaffa API GET call
→ Save result to DB
→ Generate AI explanation (ChatGPT API)
→ Return response

Returned Data:

- halal_status: halal / haram / questionable
- compliance_score
- business_score
- financial_screen
- explanation_ai

Complexity: Medium

Access: Paid

Barrier: Client must provide API key

◆ B) Analyst Ratings

Musaffa + TradingView + Seeking Alpha → যেটা client দেয়।

যদি Musaffa দেয়:

If not → fallback:

- Finnhub → কিছু analyst rating দেয়
- Yahoo Finance → unofficial APIs আছে
- SeekingAlpha API → paid

Backend route:

GET /api/analyst-ratings?symbol=AAPL

Data:

- Strong Buy %
- Buy %
- Hold %
- Sell %
- Price target high/low/median
- Analyst count

Complexity: Medium

Free? No

Client must purchase

◆ **C) Stock Price & Chart Data**

Best free + reliable sources:

- **Finnhub** (free tier + premium)
- **AlphaVantage** (free but slow)
- **Yahoo Finance Unofficial API** (কম সেফ)

Recommended → Finnhub

Backend:

GET /api/stocks/price?symbol=AAPL

Data:

- Current price
- Open
- High
- Low
- Volume

Complexity: Low

Access: Free tier available

★ Step 4: Personal Recommendation System

(Admin নিজে Entry/SL/Targets দিবে)

Backend DB table:

```
admin_signals {  
    id,  
    symbol,  
    entry,  
    stop_loss,  
    targets,  
    notes,  
    status,  
    created_at  
}
```

Frontend:

- Subscriber ছাড়া কেউ দেখতে পারবে না
- Real-time status update

AI can analyze these signals later:

- Risk level
- Success probability
- Explanation

★ Step 5: VIP Chat

Real-time chat:

- WebSocket / Socket.io
- Users ↔ Admin
- Save messages DB

Complexity: Medium

Step 6: AI Financial Tools

Everything will be **custom built in backend**:

1) Income–Expense Analysis

User input → save → route → AI prompt → financial tips generate.

2) Loan Calculator

Pure math formula
No API needed.

3) Wealth Management Panel

User assets → DB → calculate net worth → pie chart → future growth → AI advice
Simple.

Step 7: Notification System

Backend cron jobs:

- Every 24 hours → new articles
 - Every 5 min → stock price update
 - Subscription expiry reminder
-

⭐ Conclusion

সব ডেটা external APIs দ্বারা:

Feature	API Source	Free?
Halal/Haram	Zoya/Musaffa	Paid
Analyst Ratings	Musaffa / TradingView / Seeking Alpha	Paid
Stock Price	Finnhub / AlphaVantage	Free + Paid
Charts	Finnhub	Free
Subscription	Stripe	Paid
AI	ChatGPT	Paid