

# Trading Bot – Development Brief

## Project Overview:

Build a sophisticated trading bot system with integrated technical analysis, alerting, and execution capabilities. This includes:

- Multi-timeframe signal analysis
  - Buy/sell automation logic
  - Telegram/Discord integration
  - Binance Futures API execution
  - A mobile-friendly dashboard (PWA)
- 

## Modules & Deliverables

### 1. Core Signal Engine

- **Multi-Timeframe Analysis:** 4H, 1H, 15min, 1min
- **Buy Signals:**
  - Bollinger Band (20, 40, 60) breakouts
  - Trendline breaks (preferably using ZigZag)
  - EMA crossovers and touch logic
  - BB Width reduction alerts (early + confirmed)
  - Volume spikes detection (price × volume)
- **Sell Logic:**
  - Partial exits: 50/30/20%

- Trailing SL and custom stop conditions
  - 15-min candle wick detection for early exits
  - **Re-buy Logic:**
    - Auto re-entry if price breaks upper BB again after SL
- 

## 2. Trendline & Strategy Engine

- ZigZag-based trendline detection (primary method)
  - Regression trendlines (backup)
  - Optional: Parallel channel detection
  - Optional: ATR-based breakout logic
- 

## 3. Volume Spike & Coin Scanner

- Real-time scan of all exchange-listed coins
  - Spike detection via trading value vs MA
  - Configurable thresholds (timeframe-based)
  - Ranking system of coins based on spike %
- 

## 4. Alert System

- Alerts via Telegram/Discord with templates
- Early alerts (e.g., BB narrowing)
- Volume spike top coin alerts

- Tap/drag interface for alert setting (TV Pro or Lightweight Charts)
  - Alerts for trendlines, breakouts, spike %, BB width % etc.
- 

## 5. PWA UI/UX Panel

- **Settings Panel:**
    - Spike thresholds
    - Entry logic (EMA, BB, Volume %)
    - Timeframes
    - Auto/manual mode
    - Alert configs
  - **Dashboard:**
    - Live signals
    - Top spike coins
    - Open trades and triggered alerts
  - **Inspiration Sources:** Upbit UI, Figma, Screenshots, YT, Gemini AI
- 

## 6. Integration Layer

- **Binance Futures API:**
  - Order placement (Buy/Sell)
  - Leverage control
  - Position tracking

- **Messaging Bot:**
  - Telegram or Discord bot for real-time alerts
- **Charting Libraries:**
  - TradingView Pro (preferred) or Lightweight Charts
- **Optional:** Upbit-style alert interactions




---

## Tech Stack

Component	Technology
Backend Logic	Python
Charting	TradingView Pro / Lightweight Charts
Alert System	Pine Script (w/ Python Hybrid)
Exchange API	ccxt (Binance Futures)
Notification Bot	Telegram / Discord Bot
Frontend (PWA)	HTML/CSS/JS + PWA Framework
AI Integration	Gemini API, Cursor, Replit, v0
Hosting/Storage	Client-provided with local session persistence

---

## Feature Checklist

Feature	Status	Details
Buy Logic		BB breakouts (4H, 1H, 1min), Volume > 150% MA
Stop Loss		1-min price drops below mid-BB
Candle Tail Exit		15-min wick exit/re-entry logic

Sell Logic	✓	50/30/20% split, trailing SL, EMA-based exits
Volume Spike Detector	✓	Coin scan, spike %, custom MA windows
Visual Alerts	✓	Chart-based drag/drop, mobile editable

---



## Timeline & Milestones

Phase	Duration
Requirements Finalization	2–3 days
Planning + Architecture	2–3 days
Core Logic Implementation	8–10 days
UI/UX (PWA + Charts)	5–7 days
API Integrations	4–5 days
Alerts & Notification Layer	3–4 days
Final Testing & Revisions	5–7 days

**Total Estimated Timeline:** 27–30 Days

**First Working Demo:** Within 8–14 days

**Verification:** Through video demonstration

**Delivery:** Full source code after testing and approval

---



## Notes for Dev Team

- TradingView Pro license may be required.
- Use modular codebase for scalability.
- All thresholds must be configurable via the PWA.
- Ensure mobile compatibility and responsiveness.

- Reuse any AI-based recommendation system if provided.
- Set up local storage for user-defined persistent settings.