

```

<!DOCTYPE html><html lang="en"><head> <meta charset="UTF-8"> <meta name="viewport"
content="width=device-width, initial-scale=1.0"> <title>Advanced Trading Bot</title> <link
rel="stylesheet" href="styles.css"></head><body> <div id="app"> <h1>Advanced Trading
Bot</h1> <div class="input-section"> <label for="pin">Enter PIN to Activate:</label>
<input type="password" id="pin" placeholder="Enter PIN"> <button
id="activateBot">Activate</button> </div> <div id="status"></div> <div
id="tradingSignals"> <h2>Trading Signals</h2> <canvas id="chart" width="400"
height="200"></canvas> <div id="signals"></div> </div> </div> <script
src="https://cdn.jsdelivr.net/npm/chart.js"></script> <script
src="script.js"></script></body></html>body { font-family: Arial, sans-serif; margin: 0; padding:
0; background-color: #f0f0f0;}#app { max-width: 800px; margin: 50px auto; background: #fff;
padding: 20px; border-radius: 10px; box-shadow: 0 0 15px rgba(0, 0, 0, 0.2);}h1 { text-align:
center; color: #333;}input-section { text-align: center; margin-bottom: 20px;}label { display:
block; margin-bottom: 5px;}input { width: 70%; padding: 10px; margin-bottom: 10px; border:
1px solid #ccc; border-radius: 5px;}button { padding: 10px 20px; background-color: #007bff;
color: white; border: none; border-radius: 5px; cursor: pointer;}button:hover { background-color:
#0056b3;}#status { margin: 20px 0; font-weight: bold;}#tradingSignals { margin-top:
20px;}canvas { display: block; margin: 0 auto;}#signals { margin-top: 20px; text-align:
center;}const CONFIG = { PIN: "20060823", SYMBOLS: ["US30", "EURUSD", "GBPUSD"],
NEWS_API_KEY: "your_newsapi_key", // Replace with your actual NewsAPI key
NEWS_ENDPOINT: "https://newsapi.org/v2/everything",};// Elementsconst statusDiv =
document.getElementById("status");const signalsDiv = document.getElementById("signals");const
chartCanvas = document.getElementById("chart");// Activate the
botdocument.getElementById("activateBot").addEventListener("click", () => { const pinInput =
document.getElementById("pin").value; if (pinInput !== CONFIG.PIN) { statusDiv.textContent
= "Invalid PIN. Access Denied."; statusDiv.style.color = "red"; return; } statusDiv.textContent
= "Bot Activated! Fetching News and Analyzing..."; statusDiv.style.color = "green";
runTradingBot();});// Fetch news articlesasync function fetchNews() { try { const params = new
URLSearchParams({ q: "US economy OR Dow Jones", language: "en", sortBy:
"publishedAt", apiKey: CONFIG.NEWS_API_KEY, }); const response = await fetch(`${
CONFIG.NEWS_ENDPOINT}?${params}`); const data = await response.json(); return
data.articles.slice(0, 5); // Top 5 articles } catch (error) { console.error("Error fetching news:",
error); return []; };// Analyze news sentimentfunction analyzeSentiment(articles) { return
articles.map((article) => { const description = article.description || ""; const score =
description.includes("rise") ? 0.8 : -0.5; // Mock sentiment return { title: article.title,
sentiment: score > 0 ? "positive" : "negative", }; });};// Generate trading signalsfunction
generateSignal(sentiment) { return sentiment === "positive" ? "buy" : sentiment === "negative" ?
"sell" : "hold";};// Update chartfunction updateChart(data) { new Chart(chartCanvas, { type:
"bar", data: { labels: data.map((d) => d.title), datasets: [ { label: "Sentiment
Score", data: data.map((d) => (d.sentiment === "positive" ? 1 : -1)), backgroundColor:
data.map((d) => d.sentiment === "positive" ? "rgba(75, 192, 192, 0.6)" : "rgba(255, 99, 132,
0.6)" }, ], options: { responsive: true, scales: { y: {
beginAtZero: true, }, }, });};// Main bot logicasync function runTradingBot()
{ const news = await fetchNews(); const analyzedNews = analyzeSentiment(news);
signalsDiv.innerHTML = ""; // Clear previous signals const results = analyzedNews.map((article)
=> { const action = generateSignal(article.sentiment); const signal =
document.createElement("div"); signal.textContent = `News: ${article.title} - Sentiment: $
{article.sentiment} - Action: ${action}`; signalsDiv.appendChild(signal); return article; });
updateChart(results);}npm init -ynpm install express axios
dotenvNEWS_API_KEY=your_newsapi_keyMT5_API_URL=https://your_broker_api_endpointMT5_API_KEY=your_broker_api_keyconst express = require("express");const axios =
require("axios");require("dotenv").config();const app = express();const PORT =

```

```

5000;app.use(express.json());// Endpoint to fetch news securelyapp.get("/api/news", async (req, res)
=> { try { const response = await axios.get("https://newsapi.org/v2/everything", { params: {
q: "US economy OR Dow Jones", language: "en", sortBy: "publishedAt", apiKey:
process.env.NEWS_API_KEY, }, }); res.json(response.data.articles.slice(0, 5)); } catch
(error) { console.error("Error fetching news:", error.message); res.status(500).json({ error:
"Failed to fetch news" }); });// Endpoint to place a tradeapp.post("/api/trade", async (req, res) => {
const { action, symbol, lotSize } = req.body; try { const response = await
axios.post( process.env.MT5_API_URL, { action, symbol, lotSize, }, {
headers: { "Authorization": `Bearer ${process.env.MT5_API_KEY}`, }, });
res.json({ message: "Trade executed successfully", result: response.data }); } catch (error)
{ console.error("Error placing trade:", error.message); res.status(500).json({ error: "Failed to
place trade" }); });app.listen(PORT, () => { console.log(`Server running on http://localhost:${
PORT}`);});const API_BASE_URL = "http://localhost:5000/api";// Fetch news from the
backendasync function fetchNews() { try { const response = await
fetch(`${API_BASE_URL}/news`); const news = await response.json(); return news; } catch
(error) { console.error("Error fetching news:", error); return []; }}// Place a trade through the
backendasync function placeTrade(action, symbol, lotSize) { try { const response = await
fetch(`${API_BASE_URL}/trade`, { method: "POST", headers: { "Content-Type":
"application/json" }, body: JSON.stringify({ action, symbol, lotSize }, ); const result =
await response.json(); console.log("Trade result:", result); alert("Trade executed successfully!");
} catch (error) { console.error("Error placing trade:", error); alert("Failed to place trade."); }}//
Main bot logicasync function runTradingBot() { const news = await fetchNews(); const
analyzedNews = analyzeSentiment(news); signalsDiv.innerHTML = ""; // Clear previous signals
analyzedNews.forEach(async (article) => { const action = generateSignal(article.sentiment);
const signal = document.createElement("div"); signal.textContent = `News: ${article.title} -
Sentiment: ${article.sentiment} - Action: ${action}`; signalsDiv.appendChild(signal); if
(action !== "hold") { await placeTrade(action, "EURUSD", 0.01); // Example: fixed lot
size } }); updateChart(analyzedNews);}

```