

אפיון עסקי מלא – LLM מבוססת ETF אפליקציית חיזוי

מטרת המערכת

תוך שילוב בין מקורות, (ETF) לאפשר למשקיע פרטי לנתח, לשפר ולחזות את ביצועי תיק ההשקעות שלו בקרנות סל מידע עדכניים (כתבות, מדדים, ציורים), נתונים היסטוריים, כלים אלגוריתמיים, תחזיות סנטימנט, וסוכן המלצות מבוסס context7. רלוונטיים מתוך MCP חיבור עתידי לכלי LLM.

תשתית טכנולוגית

Node.js או Python FastAPI Backend TailwindCSS + Next.js או React Frontend רכיב טכנולוגיה מוצעת DB Supabase (PostgreSQL) LLM API OpenAI GPT-4o (Vision enabled) Deployment Docker Data Sources Yahoo Finance, Finnhub, Alpha Vantage, TradingView, SEC, MacroMicro, CoinGecko, Twitter/X, Google Finance Celery / Cron jobs משימות רקע PDF export, תצוגה גרפית, טבלאות, כרטיסים

(רלוונטי לכל פיצ'ר MCP כולל חיבור) רשימת פיצ'רים *

Feature 1 – SentimentSnap (MCP-NLP-Context7-Sentiment)

PDF הורדת העמוד הראשי של אתרים פיננסים מובילים כקובצי PDF • ניתוח מסמכים כלכליים בפורמט חישוב מדדי סנטימנט: פחד, אופטימיות, בועה, פוטנציאל • (OCR אין) GPT-4o Vision עיבוד באמצעות • (וכו) WSJ המלצה • MarketSignals צמיחה • ניתוח ציורים ופוסטים קצרים ממובילי דעת קהל, כלכלנים, מנהיגים וכו' • שילוב עם כמותית לסנטימנט לפי סקטור (טכנולוגיה, בריאות וכו')

Feature 2 – PatternPulse (MCP-Timeseries-Context7)

ניתוח מגמות, עונתיות, והשפעות הדדיות • זיהוי מגמות קצרות וארוכות טווח • עונתיות ברמות חודש, רבעון, שנה • ETF בדיקת השפעות הדדיות בין • RSI, MACD, ממוצעים נעים • Prophet, ARIMA, Fourier, LSTM: מודלים ריביות, אג"ח, Bitcoin, Ethereum, VIX: שילוב מדדים מאקרו

Feature 3 – WhatIfWealth (MCP-Backtesting-Context7)

סימולציית השקעה רטרואקטיבית • הכנסת תמהיל השקעה ותאריך התחלה • חישוב רווח/הפסד לפי נתוני שוק היסטוריים של התוצאות PDF ים • ייצוא-Benchmark-תמיכה בקריפטו ו drawdown, • תצוגת צמיחה, סטיית תקן, שארפ

Feature 4 – BacktestOptimizer (MCP-PortfolioOptimizer-Context7)

אופטימיזציית תיק על סמך נתוני עבר • ניתוח תיק בטווחים של 3 חודשים עד 10 שנים • סימולציית תיקים לפי תשואה, Benchmarks סיכון, שארפ • בניית תיק אופטימלי לפי קלט היסטורי • השוואה מול התיק הקיים ומול

Feature 5 – RiskBalancer (MCP-RiskAnalysis-Context7)

Beta, להקטנת סיכון • ניתוח ETF למציאת LLM משלימים • קלט: תיק קיים • ניתוח ETF איזון סיכון בתיק באמצעות Volatility, Drawdown, Diversification • Fear & Greed Index שילוב מדדים כמו • 3–2 ETF פלט: היציבות + הסבר פשוט

Feature 6 – PortfolioAgent (MCP-AgentFramework-Context7)

סוכן המלצות חכם המשלב את כל הפלטים • שימוש אקטיבי בכל שאר הפיצ'רים • מחשב תיק מומלץ בשלוש רמות שינוי חדשים ETF סיכונים ותועלות צפויות • סוכן שבועי לבדיקת המלצות חדשות • כולל המלצות ל-2, Before/After, גרף אם רלוונטי

Feature 7 – ExitAdvisor (MCP-TaxEventAnalyzer-Context7)

המלצת מכירה חכמה תוך שקלול מס וזמן • מזהה מקרים למכירה רווחית • ניתוח מגמות, סנטימנט, ממוצעים נעים • מייל המלצה אוטומטי למכירה עם תחזית והשוואה

Feature 8 – FundamentaLens (MCP-Fundamentals-Context7)

P/E, נתונים: הכנסות, רווח נקי • ETF נתונים חשבונאיים וניתוח מאזני חברות מובילות • ניתוח חברות עיקריות בתיק לפי ליציבות וכדאיות • תצוגה השוואתית בין חברות GPT תחזיות • ניתוח, ROE, חוב, E,

Feature 9 – MarketSignals (MCP-MarketFeedHandler-Context7)

זהב ונפט • שילוב, VIX, Fear & Greed Index, Yield Curve, BTC/ETH צריכת מידע חיצוני מהשוק • חיבור ל-SentimentSnap, ExitAdvisor, לסנטימנט ותזמון • משולב עם LLM של אנליסטים • ניתוח X ו-RSS עם PortfolioAgent

Feature 10 – SmartAnalytics Engine (MCP-ModelManager-Context7)

Breakouts איתור • Prophet, XGBoost, LSTM, Moving Average: מנוע אלגוריתמי מתקדם • מודלים
Walk-forward validation • Cross-validation • Hyperparameter Tuning: Optuna, GridSearch • Breakdowns •

לכל פיצ'ר ודוחות תקופתיים PDF export • והמלצות שבועיות ExitAdvisor יצוא ודיווח • מיילים אוטומטיים מה-

פרופיל משקיע (גיל, טווח, סיכון, נזילות) • אזור "הזדמנויות מיוחדות" • (Business Layer) תוספות עסקיות • ייעוץ אנושי עם טופס פנייה • ניתוח תקופתי קבוע ומיילים • התראות מותאמות אישית

מבנה פרויקט מוצע

```
project/ ├── frontend/ # React / Next.js ├── backend/ | |── api/ # API לכל פיצ'ר | |── data/ # וכו' analysis/ # PatternPulse, Backtest | |── agents/ # PortfolioAgent, ExitAdvisor | |── mail/ # llm/ # OpenAI + MCP future integration | |── db/ # Supabase schemas + ORM | |── tasks/ # Cron / Celery | |── reports/ # PDF Generator | |── config/
```

.שבו ייכתבו ויתוחזקו כל הגדרות הפיצ'רים ברמה הפרודקטית בצורה מפורטת specs.md בנוסף - תתחזק קובץ בשם

שבו תפרט את כל המשימות הפתוחות: מספר משימה, תיאור מלא של המשימה, תאריך tasks.md בנוסף - תתחזק קובץ או פיתוח. הכל באנגלית. הקובץ הזה צריך להתעדכן בכל איטרציה ui-ux ושעת ביצוע (במידה ובוצעה), האם היא משימת לניהול המשימות taskmaster.ai של פיתוח על ידי כל אחד. אפשר להשתמש ב

לפי Cursor או GitHub Agent תומכים כמו agents בנוסף - הפרויקט יתנהל בגיט ויתעדכן אוטומטית באמצעות התאמה.

אפיון טכני מלא – LLM מבוססת ETF אפליקציית חיזוי

אפיון מלא – LLM מבוססת ETF אפליקציית חיזוי

מטרת המערכת

תוך שילוב בין מקורות, (ETF) לאפשר למשקיע פרטי לנתח, לשפר ולחזות את ביצועי תיק ההשקעות שלו בקרנות סל מידע עדכניים (כתבות, מדדים, ציורים), נתונים היסטוריים, כלים אלגוריתמיים, תחזיות סנטימנט, וסוכן המלצות מבוסס context7 רלוונטיים מתוך MCP חיבור עתידי לכלי LLM.

תשתית טכנולוגית

טכנולוגיה מוצעת רכיב

Frontend React או Next.js + TailwindCSS

Backend FastAPI (Python)

DB Supabase (PostgreSQL)

LLM API OpenAI GPT-4o (Vision enabled)

Deployment Docker

Data Sources Yahoo Finance, Finnhub, Alpha Vantage, TradingView, SEC, MacroMicro, CoinGecko, Twitter/X, Google Finance

משימות רקע Celery / Cron jobs

תצוגה גרפים, טבלאות, כרטיסים PDF export

עם LLM מבוססת ETF אפיון טכני מלא – מערכת חיזוי Supabase ומCP

תשתית כללית

- **Backend:** Python 3.11 (FastAPI) שפת
- **LLM אינטגרציה:** GPT-4o (OpenAI API) או Claude 3 Opus (Anthropic API), תלוי בזמינות, מקומית
- **Supabase (PostgreSQL):** ספק בסיס נתונים
- **שירותים צד שלישי:** Bloomberg, WSJ (Scraping via Playwright)
- **סביבת הרצה:** Docker + Scheduler (Celery או Temporal רצוי) עם Redis
- **MCP Integration:** לפי מודולים מוגדרים, context7 עתידי, דרך
- **Supabase Storage:** פורמט אחסון קבצים

1 פיצ'ר: SentimentSnap

תיאור

שומר אותם, (WSJ, Bloomberg) פיצ'ר זה אוסף תוכן מהעמודים הראשיים של אתרי חדשות פיננסיים מהימנים sentiment score ומבצע עליהם ניתוח סנטימנט איכותי הכולל תקציר ו-PDF.

תזמון

- **ריצה יומית** (כל יום ב-08:00 לפי שעון ישראל)
- Temporal.io (למשל) פנימי Scheduler מופעל על-ידי קרון או

שילובי ביצוע

שלב 1: אחזור עמודים ראשיים

- Headless Chromium להרצת **Playwright** (Node.js) שימוש ב-
- טעינת העמודים:
 - <https://www.wsj.com>
 - <https://www.bloomberg.com>
- (network idle, DOM loaded) "עד שהעמוד יציב JavaScript הפעלה"
- (A4, Portrait) באיכות גבוהה PDF שמירה כ-
- /sentiment_snap/YYYY-MM-DD/ לתיקיית Supabase Storage העלאה ל-
{source}.pdf

LLM שלב 2: ניתוח באמצעות

- (Supabase Storage signed URL באמצעות) PDF שאיבת ה-
- (GPT-4o / Claude 3 Opus) PDF ingestion התומך ב- LLM קריאה דרך
- פרומפט ראשי:

You are a financial sentiment analyst.
Analyze the content of this front-page newspaper PDF and return:

- A 3-paragraph summary of the main financial trends
- A sentiment score from -1 (bearish) to 1 (bullish)
- Confidence score from 0 to 1
- Mention any market-moving events or tone of headlines
 - (בהתאם למודל, API באמצעות) PDF קובץ LLM: קלט ל-

שלב 3: שמירה למסד הנתונים

- טבלה: daily_sentiment_snapshots
- עמודות:
 - id: UUID
 - source: TEXT (bloomberg, wsj)
 - pdf_url: TEXT
 - summary: TEXT
 - sentiment_score: FLOAT
 - confidence_score: FLOAT

- `created_at`: `TIMESTAMP WITH TIMEZONE`
- אינדקסים: `created_at`, `source`

עתידי MCP שלב 4: חיבור

- MCP-NLP-Context7-Sentiment: בעתיד OpenAI יישמש להחלפת הקריאה ל-
- API דרישות:
 - Input: Raw PDF or extracted text
 - Output: Summary, sentiment score, confidence
 - Auth: Token-based

תלותיות ספרייה (Dependencies)

- `playwright-core`, `playwright-chromium`
- `supabase-py` / `supabase-js`
- `openai`, `anthropic`
- `python-dotenv`
- `croniter`, `apscheduler` או `Temporal`

אבטחה

- Supabase Storage – בלבד (גישה זמנית) `signed URL` שימוש ב-
- (LLM רק קריאה עבור) מוגבלות לפי רול – API הרשאות

בדיקות (Testing)

- (5 timeout שניות או <) בדיקת זמני אחזור לכל אתר
- (לא ריק, כולל טקסט) PDF בדיקת תקינות
- (בין 1-1 ל-1 sentiment כולל) LLM בדיקת תקפות הפלט מה-
- Supabase בדיקת תקינות הרשומות ב-

PatternPulse: פיצ'ר 2

תיאור

ומשווה אותן, (Recurring Patterns) מאתר תבניות חוזרות, ETF פיצ'ר זה מבצע ניתוח רצפים היסטוריים של מדדי למצב השוק הנוכחי כדי להעריך אם השוק עוקב אחרי דפוס דומה – לצורך איתות "חיזוי" הסתברותי קצר-טווח.

תזמון

- ריצה שבועית (ראשון ב-09:00)
- או מופעל על-פי דרישה Scheduler מבוסס

שילבי ביצוע

שלב 1: משיכת נתוני היסטוריה

- מקור נתונים: Yahoo Finance API (או Alpaca/Polygon.io)
- מדדים נבחרים (לדוג'): QQQ, SPY, IWM, TLT
- יומי granularity, עומק נתונים: 5 שנים אחרונות

שלב 2: איתור תבניות חוזרות

- (sliding window) של 10-30 יום windows חלוקה ל-
- לכל חלון באמצעות embedding הפקת:
 - Time2Vec + FeedForward מודל
 - AutoEncoder (MCP-Forecast-Context7-Encoding) או
- similarity חישוב:
 - מול החלון האחרון Cosine distance
 - "אחוז דמיון $< 85\% \rightarrow$ " התאמה
- תבניות הכי קרובות (כולל תאריך התחלה וסיום) top 5 תיעוד

שלב 3: הפקת תחזית הסתברותית

- עבור כל התאמה:
 - ?ימים X האם לאחריה התרחש ראלי/ירידה תוך
 - (similarity על פי) הפקת הסתברות מותאמת משקל
- יצירת תחזית:

```
{  
  "ticker": "QQQ",  
  "match_count": 4,  
}
```



```

    "predicted_direction": "up",
    "confidence": 0.78,
    "matched_windows": [
        {"start": "2021-02-01", "end": "2021-02-15", "outcome":
"up", "similarity": 0.89},
        ...
    ]
}

```

שלב 4: שמירה למסד הנתונים

- טבלה: etf_pattern_matches
- עמודות:
 - id: UUID
 - ticker: TEXT
 - run_date: TIMESTAMP
 - matched_windows: JSONB
 - confidence: FLOAT
 - predicted_direction: TEXT

עתידי MCP שלב 5: שילוב

- MCP-Forecast-Context7-PatternEncoder:
 - קלט: סדרות זמן
 - או תחזית Embedding vector: פלט
 - שימוש עתידי להחלפת האלגוריתם הלוקאלי

תלותיות

- yfinance, pandas, numpy, scikit-learn
- torch, tslearn, umap-learn
- supabase-py

אבטחה

- IP/token תהיה מבוקרת לפי MCP הגישה ל-
- audit לצורך Supabase תיעוד מלא של כל התחזיות ב-

בדיקות

- (שונות, נורמליזציה) embeddings בדיקת איכות
- (backtesting) בדיקת שגיאת חיזוי מול דאטה אמיתי
- false positive בדיקת התפלגות תוצאות

WhatIfWealth: פיצ'ר

תיאור

LLM. חישוב רווח/הפסד תיאורטי אילו המשתמש היה בוחר תמהיל השקעות שונה בפרק זמן נתון. אין שימוש ב־ הפונקציונליות היא מתמטית בלבד.

קלטים

- (portfolio_actual, טבלה, DB־מה) רשימת עסקאות בפועל
 - symbol, buy_date, sell_date, amount_usd
- תמהיל חלופי מוצע (משקל לכל נייר ערך)
 - {symbol: weight} 1 = כאשר סכום המשקלים
- פרק הזמן לניתוח (ברירת מחדל: מועד ההשקעה המקורי של המשתמש)

עיבוד

1. חישוב ערך תיק מקורי (לפי תשואות בפועל של כל נייר)
2. חישוב ערך תיק לפי תמהיל חלופי, בהנחה של אותה השקעה כוללת
3. השוואה בין השניים והצגת:
 - רווח/הפסד אבסולוטי ודולרי
 - רווח/הפסד באחוזים

טכנולוגיה

- pandas, numpy שימוש ב־
- בעתיד MCP-PriceFeed או yfinance, polygon: ספריית שווקים
- Supabase כל הנתונים ההיסטוריים נשמרים ב־ (etf_prices_daily טבלה)

טבלאות מסד נתונים

- `portfolio_actual`: עסקאות בפועל
- `etf_prices_daily`: מחיר פתיחה/סגירה יומי לכל נייר
- `whatif_results` (תוצאה לכל הפעלה):
 - `id, user_id, baseline_value, alternative_value, delta_usd, delta_pct, created_at`

אבטחה

- ביצוע פעולות מותר רק למשתמש המחובר לבעל הרשאות מתאימות
- זיהוי תמהיל חלופי נשמר זמנית לזמן הפעלת התרחיש בלבד

בדיקות

- בדיקה של סכום משקלים == 1
- השוואה מול נתונים ידועים מראש לתיק בדיקה
- בדיקות קצה: תמהיל 100% באותו נייר, תמהיל מפוזר

ForecastForge: פיצ'ר

תיאור

ספציפיים, על סמך למידת תבניות היסטוריות, מדדי סנטימנט, ETF יצירת תחזיות לטווח קצר (5–90 יום) על מחירי LLM ותנודתיות. המודל מתעדכן מדי לילה ומתבסס על שילוב של ממוצעים נעים פשוטים + תחזיות

קלטים

- `QQQ` (למשל) ETF סימבול
- פרק זמן תחזיתי: 5 / 10 / 15 / 30 / 60 / 90 ימים קדימה
- בחירת מודל: Simple (SMA only) / Hybrid (SMA + LLM)

שלבי עיבוד

1. `etf_prices_daily` שאיבת מחירי עבר (365 ימים אחרונים) מתוך
2. `SMA` ו-`EMA` (5, 20, 50) חישוב
3. Hybrid אם נבחר מודל:
 - להעברת התכנים הבאים (`context7`) `MCP-ForecastGen` גישה ל-
 - `Description DB` או `LLM` מתוך `ETF` הסבר על

- SentimentSnap סיכום סנטימנט עדכני מ-
 - PatternPulse מגמות טכניות מ-
 - (של ערכים + תיאור מילולי vector) מחזיר תחזית LLM ה-
4. Supabase (etf_forecasts) שמירה ב-
 5. (במידת האפשר Backtest) ניתוח אמינות בדיעבד

טכנולוגיה

- pandas, numpy, statsmodels
- MCP-ForecastGen (עתידי) context7 מבוסס
- Anthropic או OpenAI מבוצעת ב- LLM תחזית

טבלאות מסד נתונים

- etf_prices_daily
- etf_forecasts:
 - id, symbol, method, forecast_days, forecast_values (JSON), forecast_text, created_at

לוגיקת תוצאה

- לפי ימים (forecast_values) כוקטור של ערכים עתידיים forecast שמירה של
- טקסט נלווה: הסבר, מגמה כללית, סיכון אפשרי
- חיזוי על טווח תחזית (רוחב אמון אם יש)

אבטחה ובדיקות

- אימות סימבול חוקי
- מגבלות שימוש: תדירות תחזית למשתמש
- עם תנודתיות גבוהה מאוד או נמוכה מאוד ETF: בדיקות גבול
- (RMSE / MAE) בדיקת דיוק בדיעבד

PortfolioAgent: פיצ'ר

תיאור

הסוכן המרכזי שמרכז את כל הפלטים מהפיצ'רים האחרים ומייצר עבור המשתמש תובנות חכמות על תיק ההשקעות שלו – כולל זיהוי סיכונים, הצעות שיפור תמהיל, וסיכומי תחזיות. פועל לפי תזמון יומי או קריאה יזומה.

קלטים

- מזהה משתמש
- `user_portfolios` (מתוך) בתיק הפעיל ETF רשימת
- מצב סיכון מועדף (למשל: "שמרן", "מאוזן", "אגרסיבי")

שלבי עיבוד

1. Supabase (`user_portfolios`, `etf_prices_daily`) אחזור פרטי התיק מתוך
2. ETF לכל אחד מה- `etf_forecasts` אחזור תחזיות עדכניות מ-
3. (לפי תאריך עדכני) SentimentSnap שילוב נתוני סנטימנט מ-
4. (בתיק ETF למשל אם תבנית מסוימת מזוהה ב-) PatternPulse שילוב זיהוי דפוסים מ-
5. ניתוח פער בין ביצועי התיק לבין תחזיות לטווח קצר
6. אם מצב סיכון מוגדר:
 - התאמת מדדים של סטיית תקן / שארפ למצב הרצוי
 - המלצות לשינוי תמהיל בהתבסס על תחזית וסיכון
7. user: יצירת דוח מפורט ל-
 - סיכום מצב שוק
 - בתיק ETF חיזוי אישי לכל
 - אזהרות סיכון חריגות
 - ("Y ב-X למשל: "שקול להחליף) הצעות שיפור תמהיל

טכנולוגיה

- pandas, numpy, scipy
- פנייה פנימית למודולים קיימים
- לפי דרישה API מתוזמן או Python קוד
- רק בפסקאות המלצה מילולית (טקסט מותאם אישית) LLM שילוב

טבלאות במסד הנתונים

- user_portfolios: user_id, symbol, allocation, created_at
- etf_forecasts, pattern_insights, sentiment_snapshots
- portfolio_reports: id, user_id, date, report_data (JSON), llm_summary, risk_alerts

דוגמה לפלט

```
{
  "user_id": "abc123",
  "summary": "QQQ. שוק תנודתי, שקול להקטין חשיפה ל-",
  "warnings": ["מראה סימני תבנית שלילית ETF IWM"],
  "suggested_changes": [
    {"from": "ARKK", "to": "SPY", "reason": "פחות תנודתיות"}
  ]
}
```

אבטחה ובדיקות

- Supabase Auth אימות זהות משתמש מול
- הגבלת תדירות דוחות לפי זמן (1 ביום)
- ללא תחזית, מצב סיכון חריג ETF, בדיקות קצה: תיקים קטנים/גדולים
- שמירת לוג פעילות למעקב במידת הצורך

Feature 7 – ExitAdvisor** ( MCP-TaxEventAnalyzer-Context7)** 

****תיאור:**** מנגנון המלצת מכירה חכמה שמטרתו לאתר עיתוי אופטימלי למכירת ניירות ערך, תוך שקלול מס, מגמות שוק, תחזיות, וסנטימנט.

****תכלית MCP-TaxEventAnalyzer:**** חיבור עתידי ל-MCP שמוזהה ארועי מס (Tax Events) על פי החזקה של המשתמש לאורך זמן, כולל כפל מיסוי, תקופות החזקה, קיזוזי הפסדים, ואפשרות למכירה לצורך הטבת מס.

****תהליך מלא:****

1. ****קלט:****

* נתוני היסטוריית השקעות מהתיק.

* נתוני שוק מה־MarketSignals (כולל ממוצעים נעים, תחזיות LLM, מגמות).

* סנטימנט כללי מה־SentimentSnap.

* מגמות השוק במקרו: VIX, Yield Curve.

* נתוני מיסוי: תקופת החזקה, רווח ריאלי, הטבות מס (ע"פ MCP).

2. **עיבוד:**

* זיהוי ניירות שהגיעו לרווח/הפסד קריטי.

* הערכת האם מכירה כעת תניב יתרון מיסוי (לפי MCP).

* חישוב תחזית קדימה (Trend+LLM).

* שילוב ממוצעים נעים: MA50, MA200.

* הערכת רמת ביטחון ההמלצה.

3. **פלט:**

* דירוג מניות לפי פוטנציאל מכירה אופטימלית.

* סיבה מוצגת למשתמש (בגיבוי נתונים).

* PDF אישי + שליחת המלצה בדוא"ל.

* תיעוד היסטוריית המלצות והיענות.

Feature 8 – FundamentaLens** ( MCP-Fundamentals-Context7)** 

תיאור: רכיב המנתח את הביצועים הפיננסיים של חברות הנמצאות בתיק המשתמש דרך ETF. מבוסס על נתוני MCP (מאזן, רווח, P/E וכו') ומשלב ניתוח GPT ליציבות עסקית.

****תכלית MCP-Fundamentals****

MCP משמש כספק מידע לחשבונות כספיים (Income Statement, Balance Sheet, Cash Flow) תוך עדכון שוטף מה־EDGAR או Bloomberg. מאפשר שאילתה מבוססת סימבול או לפי חשיפת ETF.

****תהליך מלא****

1. **קלט**

* ETF בתיק המשתמש.

* ניירות ערך הכלולים בכל ETF.

*

משקיע (לניתוח סיכונים רלוונטיים).

2. **עיבוד**

* שאילת MCP לפי סימבולים.

* ניתוח נתונים (הכנסות, רווח נקי, P/E, חוב, ROE).

* ניתוח GPT של איתנות פיננסית והשוואה מול מתחרים.

* הצגת תובנות בלשון אנושית.

3. **פלט**

* טבלת השוואה עם חברות עיקריות בתיק.

* המלצה על חשיפה-יתר או תת-חשיפה.

* ייצוא PDF עם ממצאים.

****תיאור:**** ניתוח ממוקד של אינדיקטורים שוקיים (מאקרו ומיקרו) לצורך הבנה כללית של מגמת שוק, נתונים מוזרמים דרך MCP ומנותחים סנטימנטלית וסטיסטית.

****תכלית:MCP-MarketFeedHandler**:**

מנוע צריכת פידים של נתוני שוק (VIX, FGI, עקום תשואה, מתכות, BTC/ETH), כולל קונקטור ל-RSS ו-X (טוויטר) של אנליסטים.

****תהליך מלא:****

1. ****קלט:****

* RSS ו-X של אנליסטים.

* מדדי מאקרו כלכלה (VIX, FGI, Yield Curve).

* נתוני קריפטו (BTC, ETH), Commodities (Oil, Gold).

2. ****עיבוד:****

* ניתוח סנטימנט בטקסטים (LLM-based sentiment scoring).

* השוואה מול התנהגות שוק בפועל.

* סימון אירועים קריטיים (נפילות, עליות חדות).

3. ****פלט:****

* סקירה יומית/שבועית.

* חיווי גרפי (Dashboard).

* יצוא כ-JSON לצרכים פנימיים.

* שיתוף אוטומטי עם ExitAdvisor, SentimentSnap.

Feature 10 – SmartAnalytics Engine** ( MCP-ModelManager-Context7)** 

תיאור: מנוע תחזיות מתקדם עם מספר מודלים סטטיסטיים ו-ML. מאפשר למשתמש לקבל תחזית ביצוע מניה/ETF בטווח קצר-בינוני.

תכלית MCP-ModelManager: ניהול, הרצה, ותזמון של מודלים. כולל ממשק CLI / API להרצת מודלים מוגדרים מראש עם קלטים מה-DB.

מודלים נתמכים:

* Prophet (ע"י Facebook) – תחזיות סדרות זמן עם עונתיות.

* XGBoost – מודל גרדיאנט בוסטינג.

* LSTM – רשת נוירונים חוזרת.

* Simple Moving Average, Bollinger Bands

עיבוד:

* Hyperparameter Tuning עם Optuna.

* Cross-validation לפי Walk-forward

* שילוב נתוני עבר מתוך Supabase.


* Benchmark מול ביצועים בפועל.

פלט:

* קובץ תחזית לכל מניה.

* גרף ביצועים חזוי מול אמיתי.


* קובץ Log + תובנות איכותיות.

 **יצוא ודיווח**

* מיילים אוטומטיים למשתמש על בסיס ExitAdvisor (מכירה), WhatIfWealth (הזדמנות), ForecastForge (תחזיות שבועיות).

* PDF export עם טבלאות, גרפים ופרשנות (LLM).

* מערכת CRON שמריצה דו"חות תקופתיים (שבועי/חודשי) לכל תיק אישי.


 **Business Layer – שכבת ביזנס**

1. **פרופיל משקיע:** גיל, טווח השקעה, סיכון, נזילות (משפיע על המלצות).

2. **אזור "הזדמנויות מיוחדות":** ניתוח מניות חריגות לפי סף תנודתיות או תחזית יוצאת דופן.

3. **ייעוץ אנושי:** טופס פנייה מהאפליקציה לצוות אנושי.

4. **מיילים והתראות:** סיכום שבועי מותאם אישית.

 **מבנה פרויקט (מפורט):**

```

/project
  frontend/      # Next.js + Tailwind UI — |
  /pages          # מסכים ראשיים (Dashboard, Portfolio, Reports) — |
  /components     # רכיבים (Card, Table, Chart) — |
  /backend        — |
  api/            # RESTful API endpoints — |
  agents/         # ExitAdvisor, PortfolioAgent ועוד — |
  /analysis       # קוד חיזוי, Backtest, סטטיסטיקה — |
  /data           # חיבורים ל-RSS, X, Bloomberg — |
  /llm            # אינטגרציה ל-OpenAI ול-MCP בעתיד — |
  # SMTP          # mail/ + תבניות מיילים — |
  /db             — |
  schema.sql      # סכמות Supabase — |
  # Prisma ORM    orm/ — |
  # Cron jobs, Celery tasks/ (תזמון) — |
  # PDF Generator reports/ — |
  # קונפיגורציה לפי ENV /config — |
  ...
  ---

```

****specs.md**** 

* תיעוד מפורט לכל פיצ'ר כולל מטרות, קלט, פלט, מודולים, חיבורים ל-MCP, UI ו-LLM.

****tasks.md**** 

* רישום כל משימה לפי מבנה אחיד:

* מספר משימה

* תיאור מלא


* סטטוס

* תאריך ביצוע

* אחראי

* קטגוריה (UI/Backend)

* עדכון לפי iterations דרך agents כמו taskmaster.ai

 **ניהול קוד**

* כל ריפו בגיט.

* עדכון קוד באמצעות GitHub Agent או Cursor.

* Pull Requests מנותחים אוטומטית.

* כל merge מעדכן tasks.md / specs.md דרך agents.

/project

frontend/ # React / Next.js + TailwindCSS — |

מסכים /pages — |

רכיבי UI (טבלאות, גרפים, כרטיסים) /components — L |

/backend — |

api/ לכל פיצ'ר # REST API endpoints — |

agents/ ועוד # PortfolioAgent, ExitAdvisor — |

backtests # אלגוריתמים, חישובים, /analysis — |

# חיבורים למקורות חיצוניים (Bloomberg, X, RSS)	/data	—	
# אינטגרציה ל-OpenAI + MCP (Context7)	/llm	—	
# שירות דיזור (SMTP, תבניות)	/mail	—	
# יצירת PDF ודוחות	/export	—	⌋
	/db	—	
# סכמות Supabase	schema.sql	—	
# Prisma / Hasura / אחר / orm/		—	⌋
# Cron jobs, Celery workers, tasks/ תזמון		—	
# קבצי PDF שנוצרו	/reports	—	
# קונפיגורציות (ENV, API Keys)	/config	—	
# תיעוד מפורט של הפיצ'רים (עדכונים שוטפים)	specs.md	—	
# רשימת משימות (מספר, תיאור, סטטוס, סוג, אחראי)	tasks.md	—	
	README.md	—	⌋
		—	---

Deployment & Compute: RunPod + Docker Setup

Why RunPod?

RunPod offers serverless GPU instances, ideal for running model inference or data pipelines without maintaining infrastructure. It allows reproducible, container-based deployments at a predictable cost structure.

Docker Architecture

All compute workloads will be packaged as Docker containers. Each agent/module (e.g. LLM analysis, sentiment snapshot, PDF parsing, model inference) will run in an isolated container. All containers adhere to a base spec for observability and logging.

Container Types & Resources

Container	Function	Trigger	GPU?	Runtime
sentiment-pipeline	Download, snapshot, analyze market news	Scheduled / Manual	Optional	Python + PDF + OpenAI LLM

exit-advisor	Run tax-aware exit strategies	Triggered by portfolio events	No	Python
analytics-engine	Forecasting models + tuning	Manual / Daily	Yes (LSTM)	Python + XGBoost + Optuna
fundamental	Parse company financials	On stock refresh	No	Python +
market-	Fetch VIX, F&G, RSS	Hourly	No	Python

Deployment Plan

Each container is deployed to a RunPod endpoint with its own API token. Secure secrets and credentials are injected at runtime via `env` vars from RunPod's Vault system.

DevOps Tasks

•

Inter-Service Comms

Services communicate via internal API Gateway on `backend/api/`. Messages between agents can be queued (via Redis or Supabase Realtime Channels).