

פרויקט Spearhead: מעבר לארכיטקטורת ענן

תכנית ארכיטקטורה ופריסה

8 בפברואר 2026

1 מבוא וסקירה כללית

פרויקט Spearhead הוא כלי לעיבוד נתונים וקבלת החלטות המיועד לדרג הפיקודי ביחידות שדה. המערכת ניזונה מדיווחי תפקוד שבועיים הממולאים ב-Google Forms, מעבדת אותם באמצעות מנוע ניתוח מבוסס Python ומציגה תובנות לגבי כשירות, שחיקת כוח אדם ושלמות נתונים. המעבר לענן נועד לפתור את המגבלה של הרצה מקומית, לאפשר גישה מאובטחת מכל מקום, ולהפוך את תהליך ייבוא הנתונים לאוטומטי לחלוטין.

2 ארכיטקטורת היעד (Cloud Architecture)

הארכיטקטורה הנבחרת מבוססת על מודל Serverless, המבטיח זמינות גבוהה לצד עלות אפסית בנפחי הפעילות הנוכחיים.

2.1 מרכיבי המערכת

- **שכבת הנתונים:** שימוש ב-Google Sheets API כמקור הנתונים החי. המערכת תתחבר ישירות לגיליון התגובות של הטופס.
- **עיבוד (Backend):** הרצת שרת FastAPI על גבי Google Cloud Run. זהו שירות המריץ קונטיינרים (Docker) ומשלם רק על זמן העיבוד בפועל.
- **בסיס נתונים:** מעבר מ-SQLite ל-PostgreSQL המנוהל ע"י Supabase. פתרון זה מאפשר התמדת נתונים (פרסיסטנטיות) בענן ללא עלות.
- **ממשק משתמש (Frontend):** אפליקציית ה-React תוגש דרך Firebase Hosting, שירות אופטימלי להגשת קבצים סטטיים בצורה מהירה ומאובטחת.

3 תוכנית הריפקטור (Step-by-Step Refactor)

3.1 שלב 1: הזנת נתונים אוטומטית (The Data Pipe)

כיום המערכת קוראת קבצי CSV מתיקיית data/input. הריפקטור יכלול:

1. הקמת Service Account ב-Google Cloud Console.
2. מימוש SheetsAdapter בתוך src/spearhead/data/adapters שישתמש בספריית gspread.
3. ה-Adapter ימיר את הנתונים מהגיליון ישירות ל-Pandas DataFrame, כך שכל לוגיקת ה-Analyzers הקיימת תישאר ללא שינוי.

3.2 שלב 2: התאמת בסיס הנתונים

המערכת משתמשת כרגע ב-SQLAlchemy מול קובץ מקומי.

1. שינוי ה-Database Engine לשימוש ב-PostgreSQL.
2. העברת הגדרות החיבור למשתנה סביבה (Environment Variable) בשם DATABASE_URL.
3. יצירת סקריפט Migration להקמת הטבלאות ב-Supabase.

3.3 שלב 3: אריזה (Dockerization)

יצירת Dockerfile המשלב את הקוד הקיים עם שרת Uvicorn. השלב הזה קריטי ליצירת סביבה זהה בין המחשב האישי לענן.

4 ניתוח עלויות (FinOps)

עבור נפח של כ-16 תגובות בשבוע ושימוש של מספר משתמשים בודדים, העלות הצפויה היא:

רכיב	שירות	עלות חודשית
Compute	Google Cloud Run	\$0.00 (תחת ה-Free Tier)
Database	Supabase	\$0.00 (עד 500MB נתונים)
Hosting	Firebase Hosting	\$0.00 (עד 10GB תעבורה)
Storage	Artifact Registry	\$0.15 ~ (אחסון גרסת הקוד)
סה"כ		~ 1 ש"ח

5 לוחות זמנים משוערים

- **שבוע 1:** הגדרת סביבת ענן (GCP, Supabase), חיבור ה-FastAPI ל-Google Sheets.
- **שבוע 2:** בניית Docker Image, פריסה ל-Cloud Run וחיבור ה-React ל-API החדש.
- **שבוע 3:** הטמעת Google OAuth לאבטחת כניסה, בדיקות קבלה ותיקוני באגים.

6 שיקולים נוספים ואפשרויות מגירה

1. **אבטחת מידע:** מכיוון שהנתונים רגישים, יש להגדיר Identity-Aware Proxy או להשתמש ב-Google OAuth כדי לוודא שרק בעלי הרשאות יוכלו לגשת לדשבורד.
2. **אפשרות מגירה - Stateless:** אם תרצה לפשט את המערכת עוד יותר, ניתן לוותר על ה-Database ולבצע עיבוד מלא מהגיליון בכל טעינה. זה יחסוך את הצורך ב-Supabase אך יאט מעט את זמן הטעינה.
3. **Cold Start:** הבקשה הראשונה ביום עלולה לקחת כ-10 שניות בזמן שהשרת "מתעורר". ניתן לפתור זאת ע"י השארת מופע אחד דולק תמיד בעלות של כ-5 דולר לחודש, אך זה אינו הכרחי.