**1. Hiring Cost Issue – עלות גיוס עובדים**

בגיוס רחב היקף, במיוחד לחברות גדולות, העלויות יכולות להיות עצומות.  
חברה שמגייסת אלפי עובדים צריכה לפרסם מודעות דרושים בעשרות לוחות דרושים ואתרים שונים.  
המשמעות היא הוצאה חודשית גבוהה מאוד על פרסום משרות.  
לדוגמה: אמזון שמגייסת עובדים למחסנים שלה בכל רחבי ארה״ב – מוציאה מאות אלפי דולרים בחודש רק על פרסום מודעות.  
מעבר להוצאה הגבוהה, נוצר מצב שבו העלות הממוצעת למועמד בודד אינה יציבה ולעיתים גבוהה מדי.  
כאשר אין ניהול ואופטימיזציה של הקמפיינים – כסף רב מתבזבז ללא החזר השקעה ברור.

**2. Our Solution – הפתרון שלנו (HireHeroes SaaS)**

הפתרון שלנו הוא פלטפורמת SaaS חכמה לניהול פרסום מודעות דרושים.  
אנחנו עושים שימוש בבינה מלאכותית כדי לנהל ולייעל את תקציבי הקמפיינים.  
המטרה: להבטיח שהתקציב ינוצל במלואו, ובצורה אופטימלית.  
כך עלות הגיוס למועמד בודד (Cost Per Applicant) יורדת בצורה משמעותית.  
לדוגמה – עבור כל מועמד למחסני אמזון, החברה תשלם פחות מהסכומים שהיא רגילה לשלם כיום.  
זה חוסך לחברות גדולות הוצאות של מיליוני דולרים בשנה.  
המערכת מתאימה לארגונים גדולים וגם לחברות קטנות יותר, בזכות הסקלביליות שלה.

**3. Use Case – Adzuna**

אדזונה היא לוח דרושים בינלאומי, שבו מתפרסמות מיליוני משרות.  
חברות מכל העולם מעלות לשם מודעות במטרה למשוך מועמדים.  
המערכת שלנו משתלבת ישירות עם ה-API של אדזונה.  
באמצעותה, אנחנו מזהים ומנהלים את המודעות והקמפיינים שלנו.  
המטרה היא לא רק לפרסם – אלא גם לאופטם את התקציב כדי להשיג יותר מועמדים בפחות כסף.  
חשוב לציין: אדזונה היא רק דוגמה – אנחנו עובדים עם לוחות נוספים, והמערכת יודעת להשתלב עם מגוון פלטפורמות.

**4. Adzuna API**

האינטגרציה עם ה-API של אדזונה מאפשרת לנו לקבל בזמן אמת את המשרות שמתפרסמות שם.  
המידע שאנחנו שולפים כולל: כותרת המשרה, טווחי שכר, תיאור תפקיד, סוג החוזה (מלא/חלקי), שם החברה, תחום המשרה, ועוד.  
בנוסף, מתקבל מידע על מיקום – ארץ, מדינה, עיר – וגם נתונים גיאוגרפיים כמו קואורדינטות.  
המידע הזה מאפשר לנו לבנות בסיס נתונים מובנה של קמפיינים ומודעות, שעליהם אפשר להריץ אנליטיקות מתקדמות.

**5. Tech Stack – טכנולוגיות**

כדי ליישם את הפתרון, השתמשנו בסט טכנולוגי מגוון:

* **Adzuna API** – מקור הנתונים הראשי של מודעות הדרושים.
* **Python** – לשאיבת נתונים ועיבוד ראשוני.
* **Amazon S3** – לאחסון גולמי ומעובד של כל הנתונים.
* **Nominatim API** – להשלמת מידע חסר על מיקומים גיאוגרפיים.
* **PySpark** – לעיבוד נתונים בקנה מידה גדול.
* **Amazon Glue** – לניהול סכמות והכנת נתונים ל-ETL.
* **Airflow** – לתזמון וניהול של תהליכי ETL מורכבים.
* **Amazon Redshift** – מחסן נתונים לאנליטיקות.
* **Metabase** – ליצירת דוחות ודשבורדים לניטור ביצועי הקמפיינים.

**6. Data Flow – תרשים זרימה**

בתרשים הזרימה אנו רואים את תהליך הנתונים מקצה לקצה:  
נתונים נשאבים מה-API של אדזונה.  
הם עוברים סדרת שלבים ב-ETL – החל מ-S3, דרך Glue ו-PySpark, כולל העשרה עם נתוני מיקום מנומינטין.  
לאחר מכן נשמרים ב-S3 בפורמט מובנה.  
משם הנתונים נטענים ל-Redshift, שם אנחנו מגדירים Views לניתוחים.  
לבסוף, Metabase מתחברת ל-Redshift ומאפשרת לנו להציג אנליטיקות בזמן אמת.

**7. ETL Stages – שלבי ה-ETL**

המערכת בנויה על תהליך רב-שלבי:

1. **שלב 0** – שאיבת מודעות גולמיות ושמירתן ב-S3.
2. **שלב 1** – ניקוי המידע, שינוי שמות עמודות, והגדרת טיפוסי נתונים.
3. **שלב 2** – השלמת נתוני מיקום חסרים באמצעות Nominatim.
4. **שלב 3** – עיבודים נוספים והנדסת פיצ’רים.
5. **שלב 4** – הוספת הצעת מחיר ראשונית לכל מודעה.
6. **שלב 5** – יצירת ישויות: קמפיין, משרה, קליק.
7. **שלב 6** – שמירת הישויות המובנות ב-S3.

**8. Orchestration: Glue**

בתחילת הדרך, השתמשנו ב-AWS Glue כדי להריץ את תהליכי ה-ETL.  
Glue היא פלטפורמה **Serverless**, כלומר אין צורך לנהל שרתים בעצמנו.  
באמצעותה יכולנו להגדיר ולתזמן משימות יומיות של עיבוד ושאיבת נתונים.  
היתרון המרכזי – מהירות בהקמה ועלות נמוכה יחסית.  
היא איפשרה לנו לבנות את צנרת הנתונים הראשונית בצורה פשוטה ויעילה.

**9. Orchestration: Airflow**

בהמשך, עברנו לנהל את התהליכים באמצעות **Apache Airflow**.  
Airflow מאפשר גמישות רבה יותר – הוא מותאם לתזמון של תהליכים מורכבים, עם תלות הדדית בין משימות.  
בנוסף, יש שליטה טובה יותר על נושאים כמו **ריטריים, ניטור ושקיפות** של ריצות.  
המערכת מתאימה יותר לסביבת פרודקשן אמיתית, בקנה מידה גדול.  
כיום, הפרויקט מוכן לרוץ על Airflow בסביבה חיה ואמינה (גם אם כרגע בפועל הוא כבוי).

**10. Glue Tables**

במהלך התהליך בנינו טבלאות Glue שהגדירו את הסכמות של הנתונים.  
הטבלאות הללו אפשרו לנו לשאול נתונים ישירות מ-S3 בצורה נוחה.  
בנוסף, הן שימשו כשכבת ביניים בין ה-Data Lake (S3) לבין ה-Data Warehouse (Redshift).  
כך יכולנו להבטיח שהנתונים יהיו עקביים, בעלי שמות אחידים וסוגי נתונים מוגדרים מראש.  
בסופו של דבר, ה-Views ב-Redshift נבנו על בסיס הטבלאות האלו, מה שאפשר אנליטיקות אמינות.

**11. Presentation & Analytics (Metabase)**

בשלב הסופי, חיברנו את Metabase ל-Redshift.  
באמצעות זאת יצרנו דשבורדים וניתוחים שונים על בסיס נתוני הקמפיינים.  
בין היתר – ניתוח תנועה, כמות קליקים, עלות פר קליק, ועלות פר מועמד.  
המערכת מאפשרת לנו לעקוב אחרי ניצול התקציב של כל קמפיין, ולזהות חריגות.  
בזכות הדוחות, ניתן לבצע התאמות בזמן אמת כדי להגיע ליעדי הקמפיין.  
היכולת להגיב במהירות מביאה לחיסכון משמעותי בעלויות הפרסום.