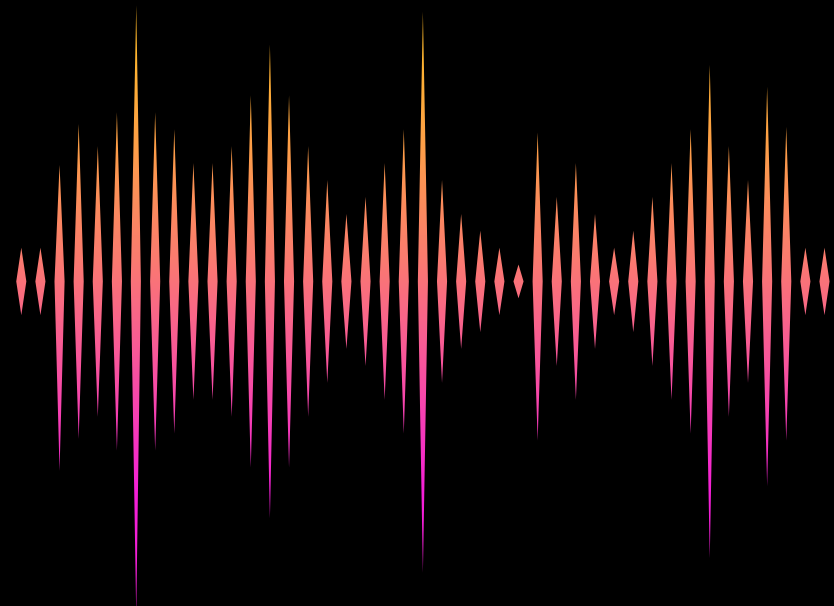


מערכת לניתוח פודקאסטים

רע בורלא



● מה כולל הפרויקט?

- מערכת אנליטית מקצה לקצה לניתוח נתוני פודקאסטים
- אוטומציה מלאה של איסוף וניתוח נתונים

מטרות מרכזיות:

- ניטור ביצועי תוכן הפודקסטים של Mamramic
- הבנת ההתנהגות של המשתמשים
- אופטימיזציה של תוכן
- זיהוי מגמות





תשתית

תשתית AWS עם שלוש שכבות:

- נתונים גולמיים (Raw - **Bronze**)
- נתונים בשלב הכנה או מעובדים חלקית (Staging - Silver)
- נתונים נקיים "מזוקקים" עבור ניתוחים אנליטיים (**Gold** - Curated)



שלבי התהליך

1. אחסון של כלל הנתונים הגולמיים (קובץ הפיד והלוגים) ב-S3 Bronze
2. ניקוי ועיבוד של הנתונים וקבצי הלוגים ואחסונם ב-S3 Silver
3. הקמה של מחסן נתונים מהנתונים המעובדים המשולבים ב-S3 Gold
4. יצירה של קטלוג בעזרת Crawlers
5. ניתוח של הנתונים

Ingestion → Raw → Transformation → Warehouse

כלים ●

תהליכי העיבוד מנוהלים ע"י Apache Airflow ורצים בתוך קונטיינר Docker, תוך שימוש בספריות Python לעיבוד הנתונים, והמרתם לפורמט Parquet. בשל גודל הנתונים היחסית קטן, אנו נמנעים משימוש בכלים כמו Apache Spark, ומאיצים את העיבוד באמצעות עיבוד במקביל, תוך ניצול מיטבי של ליבות המעבד לביצועים משופרים ותהליכי עבודה יעילים.



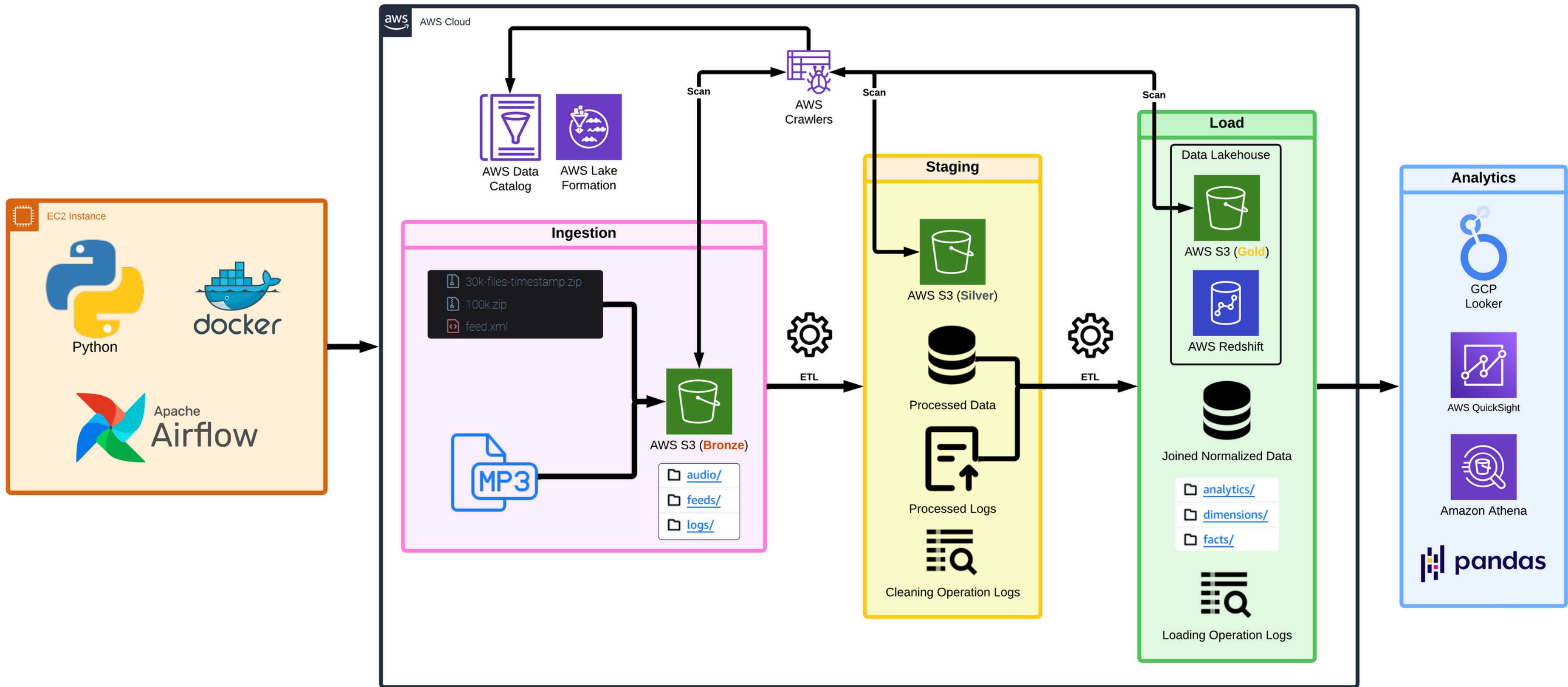
מספק סביבה מבודדת ואחידה להרצת התהליכים **Docker:**



מאפשר ניהול אוטומטי של התהליכים **Apache Airflow:**



Python: pandas, boto3, pyarrow, ffmpeg, nltk



מיפוי מקור-יעד

Source: raw-data-bronze/feeds /[year]/[month]/feed.xml
Target: staging-data-silver/feeds/data/episodes/[year]/[month]/feed.parquet, curated-data-gold/dimensions/

אילוצים	סוג נתונים	טרנספורמציה	טבלת יעד	שדה יעד	שדה מקור
לא ריק	מחרוזת	ניקוי רווחים, חילוץ מבקר	dim_episode	title	title
לא ריק	חותמת זמן	המרה לפורמט ISO	dim_date	pubdate	pubDate
0-3600	מספר שלם	המרה לשניות, הגבלה ל-3600	dim_episode	duration_seconds	duration
גדול מ-0	מספר שלם	תיקון לפי כותרת אם יש אי התאמה	dim_episode	episode	episode
לא ריק	מחרוזת	ניקוי רווחים	dim_author	author	author
לא ריק	מחרוזת	המרה לאותיות קטנות	dim_episode	episodetype	episodeType
גדול מ-0	מספר שלם	ברירת מחדל 1 אם ריק	dim_episode	season	season
יכול להיות ריק	מחרוזת	ניקוי HTML ורווחים	dim_episode	description	description

Source: raw-data-bronze/logs/archives/[year]/[month]/[type]/*.zip
Target: staging-data-silver/logs/data/[type]/[year]/[month]/[type]_logs.parquet

אילוצים	סוג נתונים	כלל טרנספורמציה	טבלת יעד	שדה יעד	שדה מקור
לא ריק	חותמת זמן	המרה לפורמט ISO	fact_engagement	timestamp	timestamp
לא ריק	מחרוזת	העתקה ישירה	fact_engagement	user_id	unique_id
לא ריק	מספר שלם	קיבוץ לספירות	fact_engagement	event_type	event
לא ריק	מספר שלם	מפתח לחיבור עם dim_episode	fact_engagement	episode_id	episode_number

- search_count - ספירת אירועים מסוג 'search' לפי episode_id, date_id
- listen_count - ספירת אירועים מסוג 'listen' לפי episode_id, date_id
- like_count - ספירת אירועים מסוג 'like' לפי episode_id, date_id

ניצור סביבת עבודה






נריץ את קובץ ה-Shell

 build_and_initialize-airflow.sh

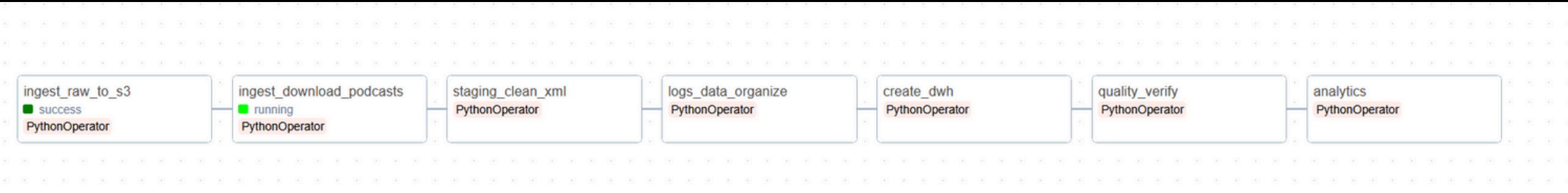
ניצור את ה-Container בעזרת Docker

```
6
7 # Build
8 docker build -t airflow-etl .
9
10 # Run
11 docker run -d -p 8080:8080 --name airflow-etl airflow-etl airflow standalone
12 docker exec -it airflow-etl airflow users create --role Admin --username user --email admin --firstname admin --lastname admin --password 1234
13
14 #UI: http://localhost:8080
15
```

נכנס ל-UI של Airflow ונריץ את ה-DAG בכדי להתחיל את התהליך

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
 run_operation-spotify	user		0 1 1 * *	2025-01-25, 03:09:52	2025-01-01, 03:00:00		 	...

נמתין לסיום שרשרת התהליך



Main-C9 - /home/ec2-user/environment

data

30k

30k-files-timestamp.zip

100k

100k.zip

feed.xml

src_scripts

analytics.py

clean_XML.py

create_DWH.py

download_podcasts.py

logs_data_organize.py

quality_verify.py

raw_to_s3.py

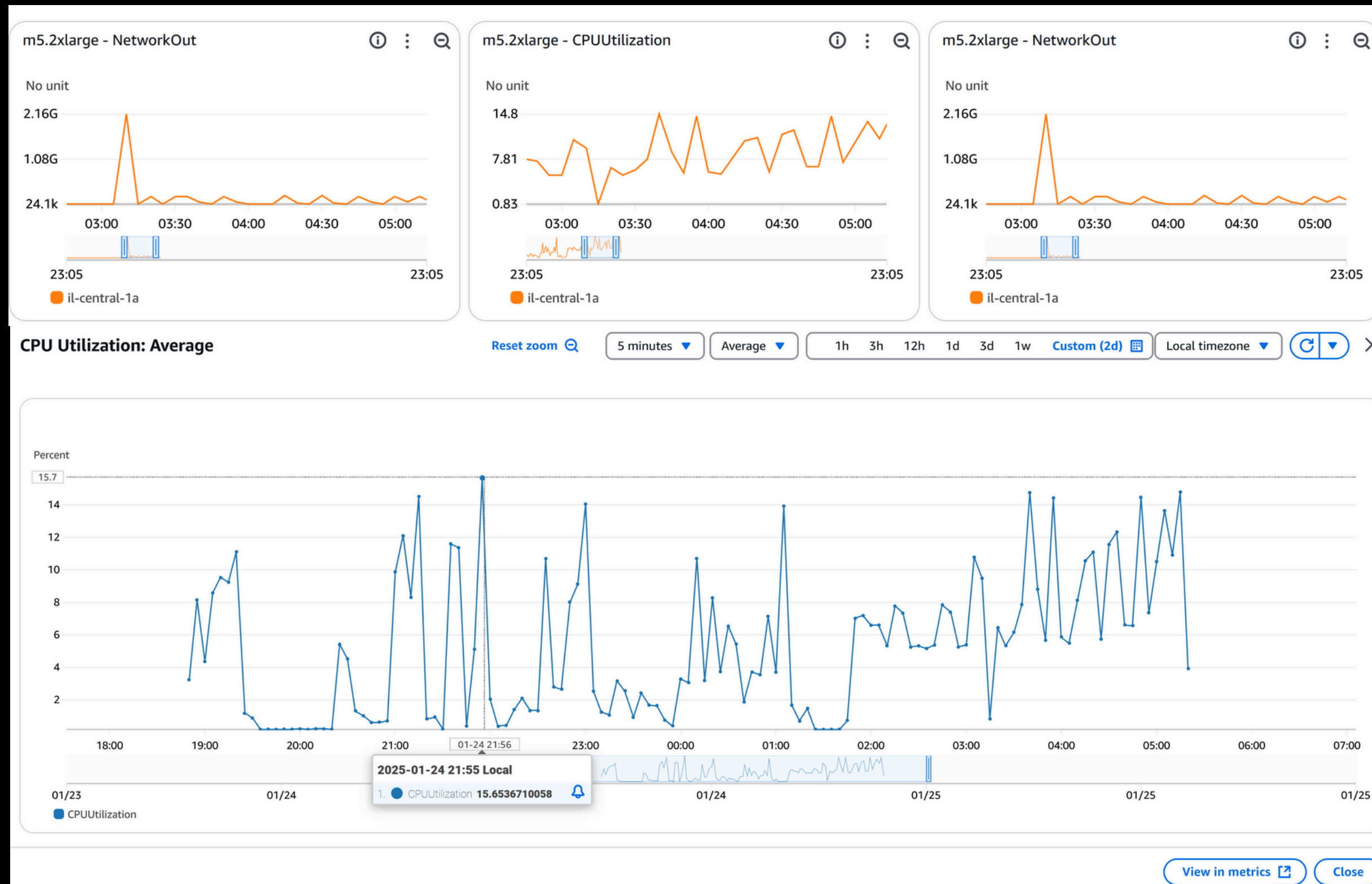
build_and_initialize-airflow.sh

Dockerfile

orchestration-airflow.py

requirements.txt

נעקוב אחר פעילות המשאבים שבשימוש עם AWS CloudWatch (דוגמה להמחשה)





שלב קליטת נתונים גולמיים

● טעינה של נתוני ה-XML ל-S3 Bronze

● טעינה של קבצי הלוג ל-S3 Bronze

● יצירה של דוח עבור האופרציה

● הורדת קבצי אודיו ובדקית האורך שלהם כולל תיקון בעזרת ffmpeg



עיבוד והמרת נתונים

אנו אוספים מידע מפודקאסטים (כותרות, תיאורים, מידע על אורחים וכו') מקבצי XML, לצד לוגים של משתמשים הכוללים האזנות, חיפושים ולייקים. נתונים אלו מספקים תובנה מקיפה על ביצועי פרקים, התנהגות מאזינים ודפוסי מעורבות לאורך זמן.

● סטנדרטיזציה של חותמות זמן

● חילוץ ותיקון מספרי פרקים

● חילוץ ותיקון מספרי פרקים

● ניקוי שדות נתונים

● המרת משך זמן לשניות

● חילוץ שמות של אורחים

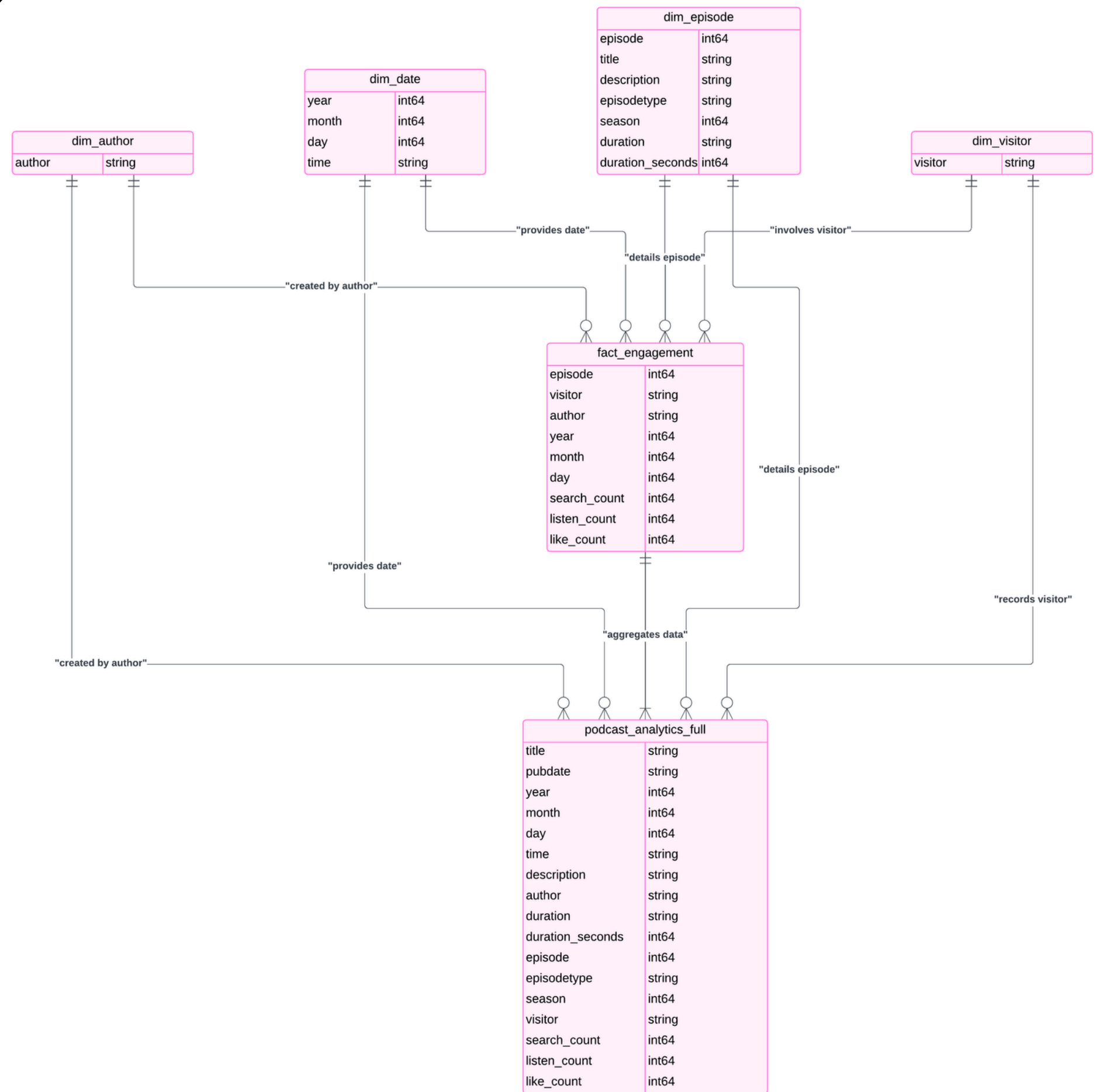
● איחוד לוגים מכל המקורות

● ארגון אירועים לפי זמן

● יצירת מדדי לייקים, חיפושים, והשמעות.

● אחסון נתונים מעובדים כקבצי Parquet
ב-S3 Silver

יצירת מחסן נתונים



בקרת איכות נתונים

בדיקת שלמות נתונים, תיקוף ערכים, זיהוי כפילויות, ועקביות פורמט.

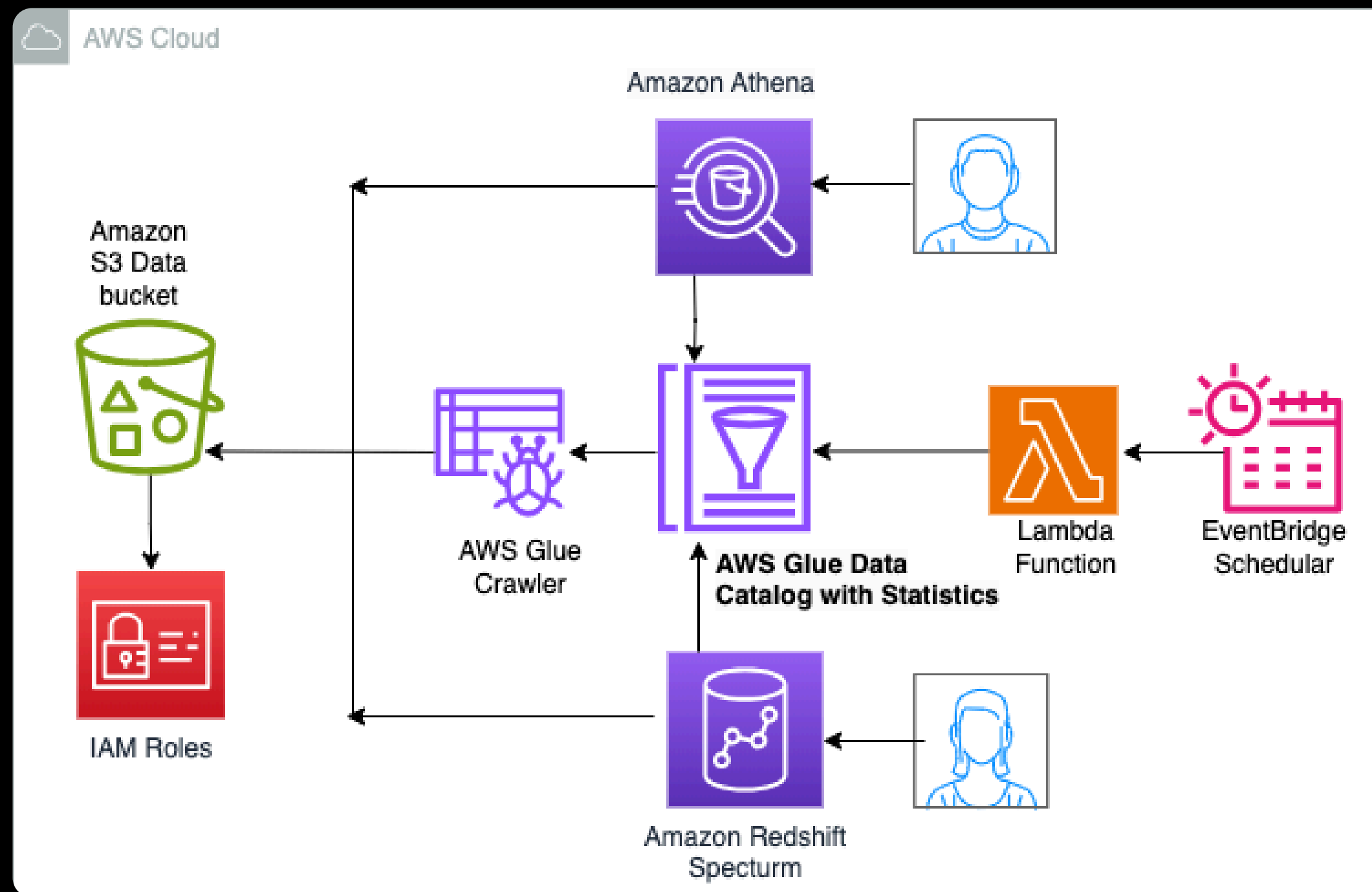
דוגמה:

Data Quality Check Results:

- ✓ No null values found
- ✓ All episodes within 1 hour limit
- ✓ All titles are unique
- ✓ Episodes continuous from 1 to 112
- ✓ All date values valid
- ✓ All expected columns present



יצירה של קטלוג



ניתוח נתונים

● ניתוח טמפורלי

● ביצועי פרקים

● השפעת משך זמן

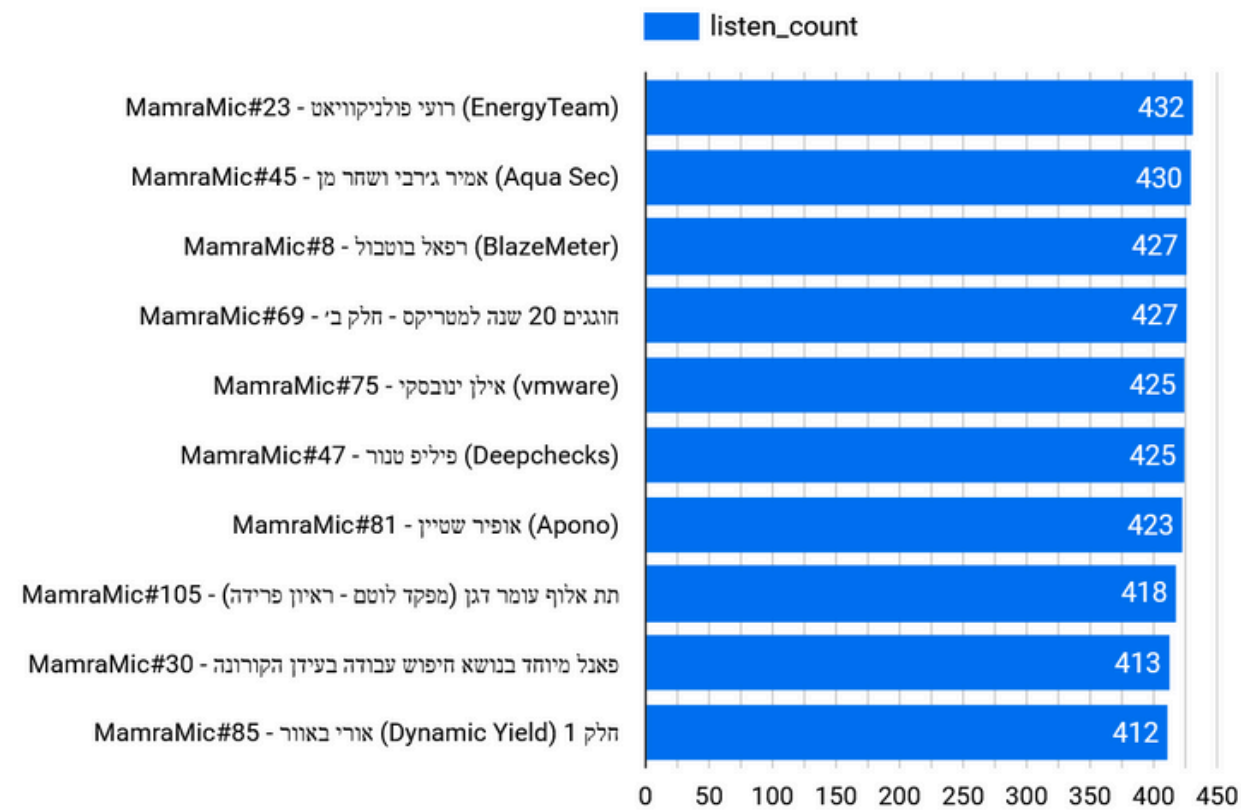
● ניתוח תוכן

● ניתוח נושאים ומגמות

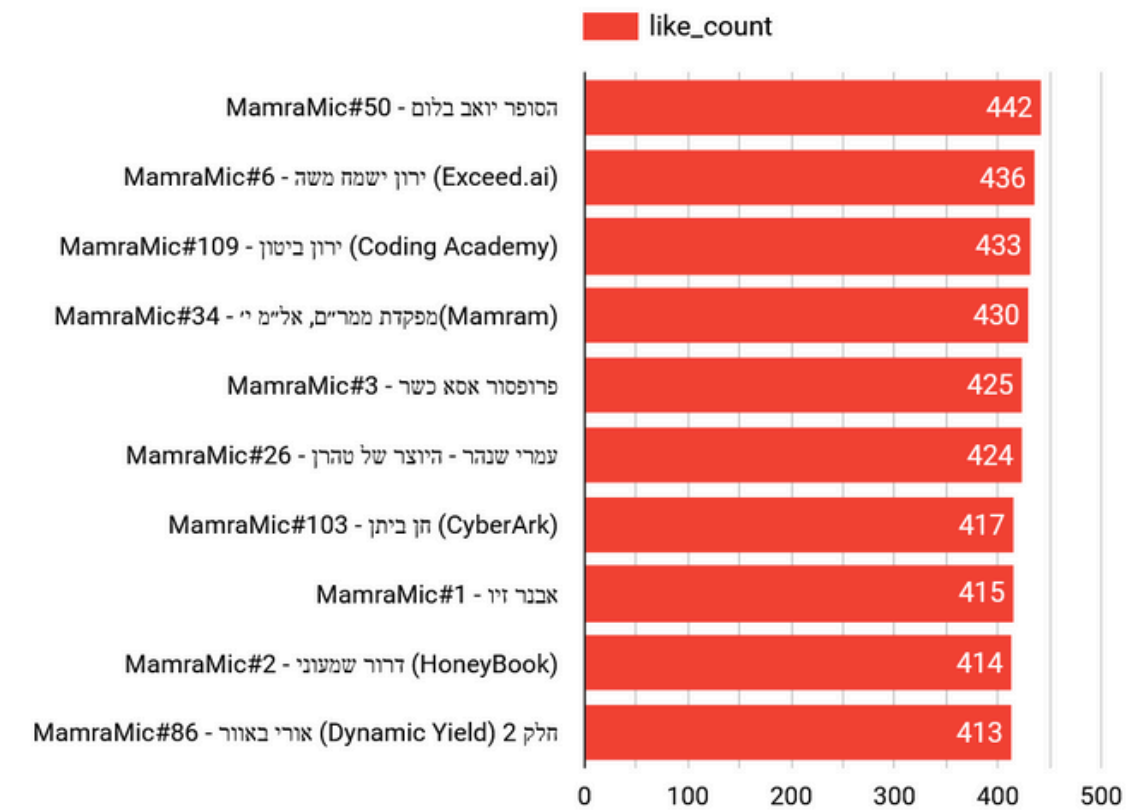
● ביצועי אורחים



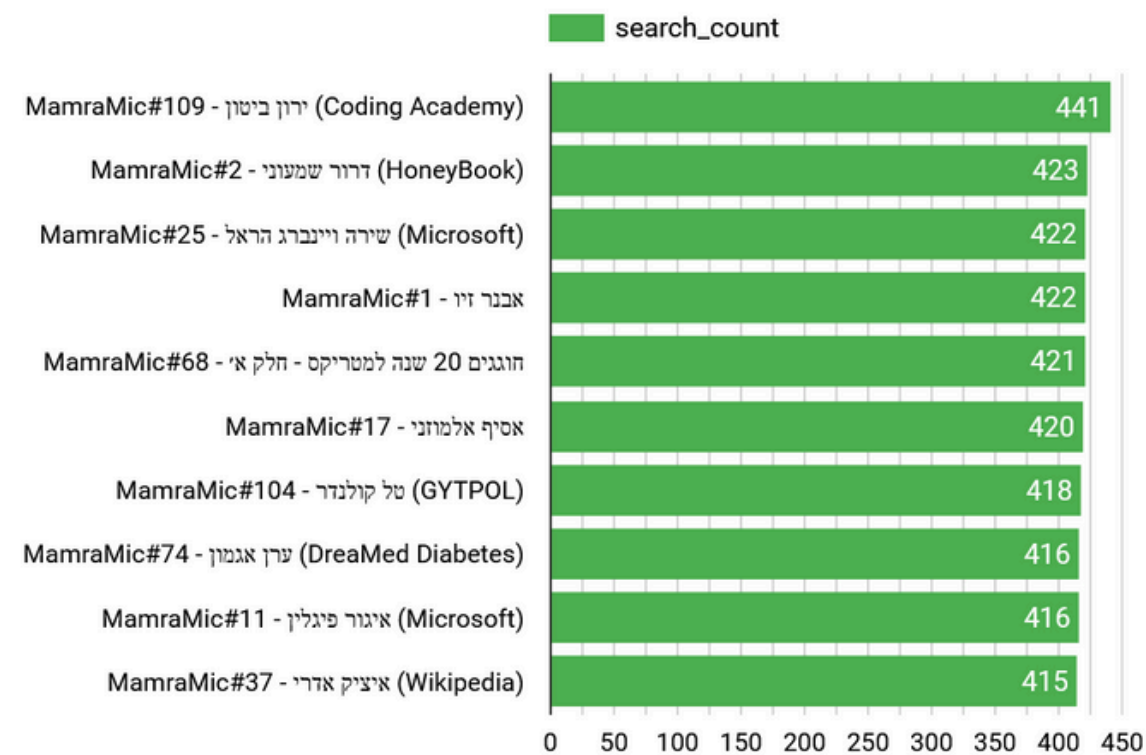
Top 10 Most Listened Podcasts



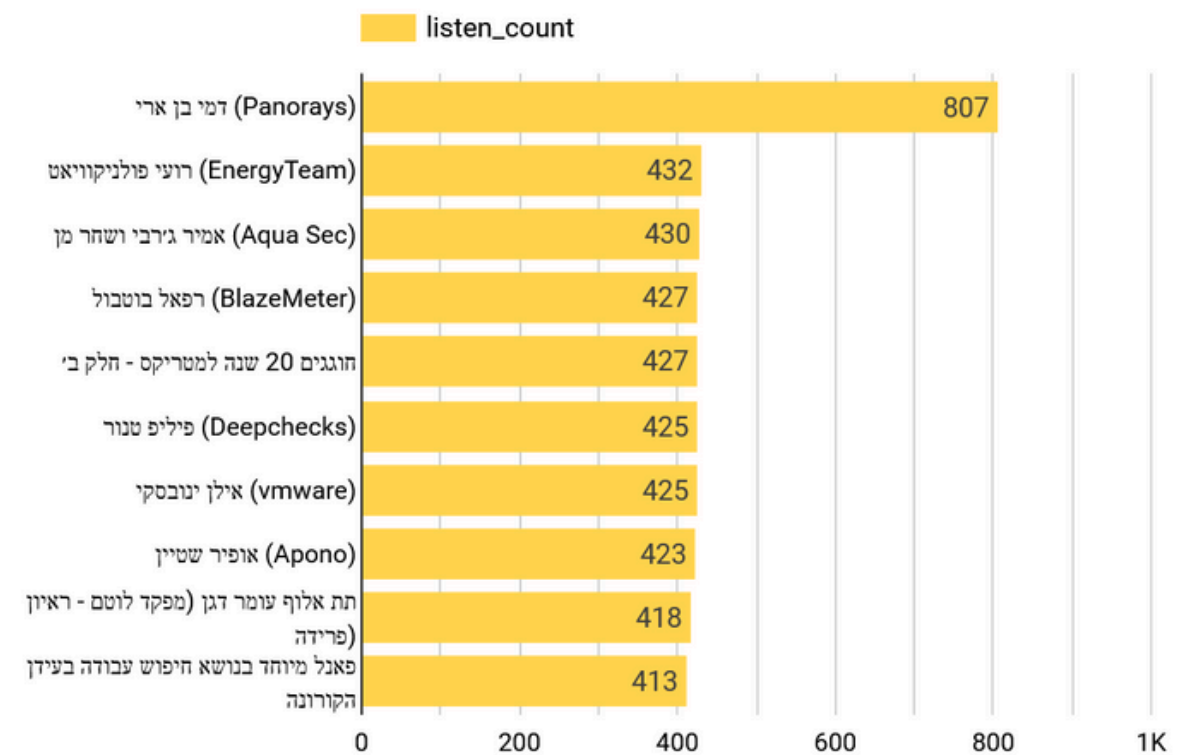
Top 10 Most Liked Podcasts



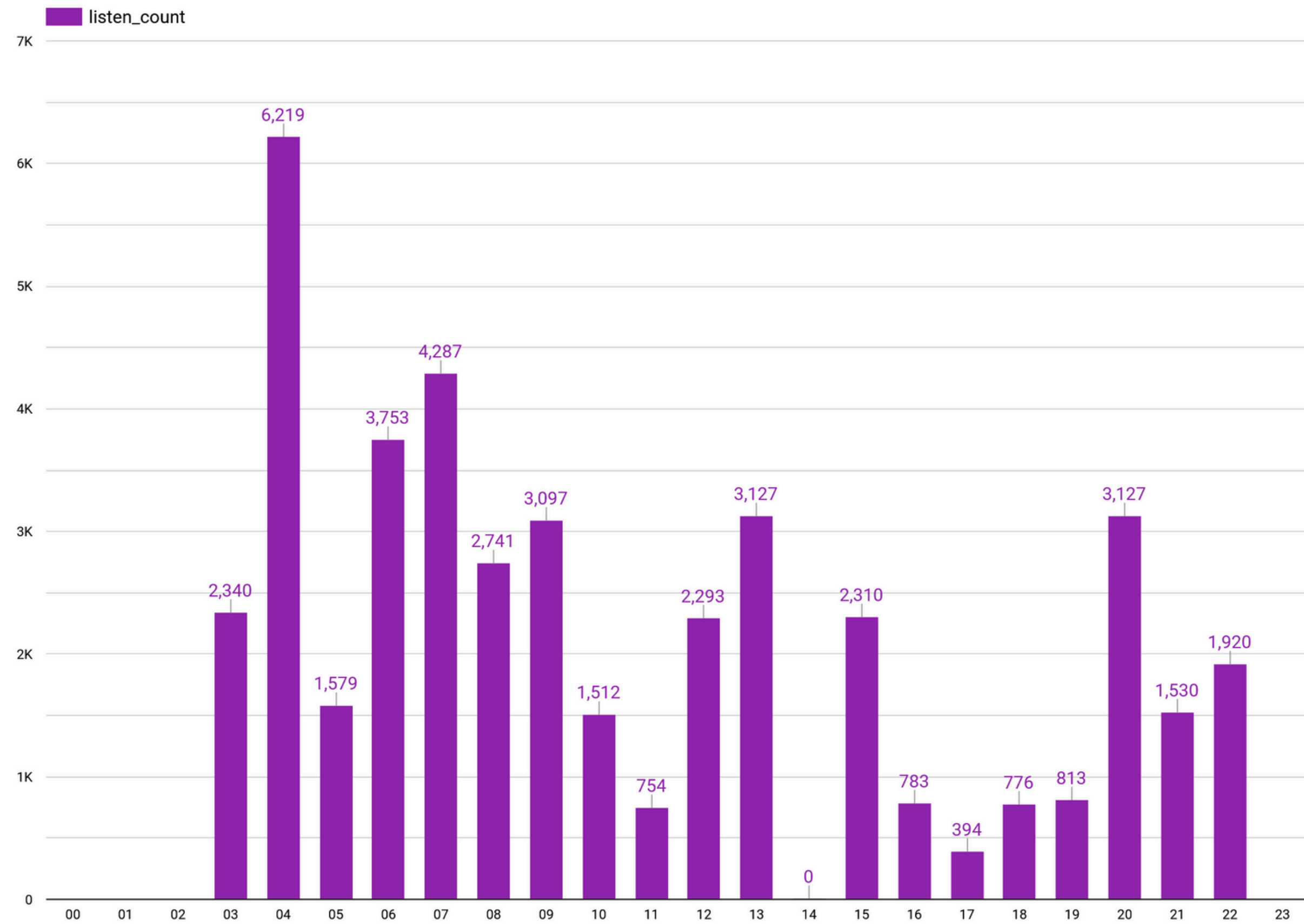
Top 10 Most Searched Podcasts



Top 10 Most Listened Visitors



Number of Listeners by Hour



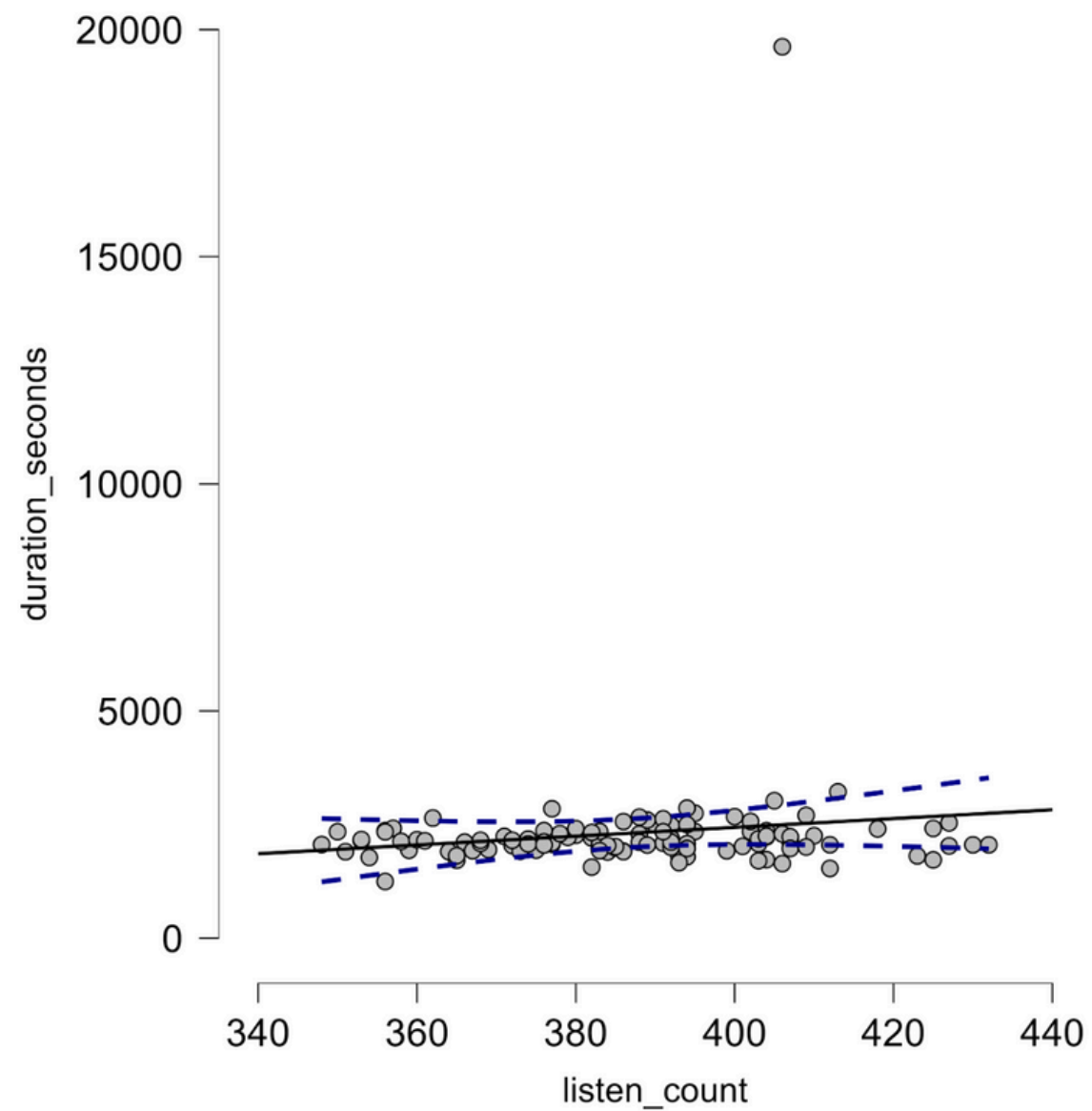
Pearson's Correlations

			Pearson's r
listen_count	-	duration_seconds	0.114

* $p < .05$, ** $p < .01$, *** $p < .001$

Scatter plots ▼

listen_count vs. duration_seconds ▼



נמצא כי אין קורלציה בין כמות
ההשמעות לאורכם של הפודקאסטים
מכיוון שהמתאם נמוך ואינו מובהק.

נתונים נוספים

נשתמש גם בחבילת NLTK לעיבוד
שפה טבעית עבור מציאת המילים
הכי נפוצות מתיאורי הפרקים.

המילים הכי נפוצות הן:

- שחר: 4
- מיקרוסופט: 3
- כהן: 3
- אלוף: 3
- עומר: 3
- סא"ל: 3
- בן: 3
- עודד: 3
- אבטחה: 2
- יו"ר: 2

זמן ממוצע בין פרקים חדשים: 15 ימים

אורך בדקות של הפרקים

Min: 20.8

Max: 60.0

ממוצע: 36.2

חציון: 35.2

חמשת השעות בהם הפודקאסטים הכי מושמעים

04:00 :6219 listens

07:00 :4287 listens

06:00 :3753 listens

13:00 :3127 listens

20:00 :3127 listens

על בסיס הנתונים הקיימים:

- קהל היעד של הפודקאסטים כולל בעיקר את קהילת בוגרי ממר"ם, אנשי טכנולוגיה, יזמים ואנשים בעלי עניין בתעשיית ההייטק, בהתבסס על השמות של האורחים, התפקידים שלהם, והקשרים שלהם לעולם הטכנולוגי והחדשנות.
- הצעת הערך הייחודית ש-Mamramic מספקת למאזינים היא היכולת לגשת לתובנות ייחודיות וידע שנצבר על ידי בוגרי ממר"ם בעלי ניסיון בתחומים מגוונים. הפודקאסט נותן במה לא רק למידע טכני אלא גם לדיונים על מסלולי קריירה, הצלחות, ואתגרים מקצועיים.
- המילים שחוזרות ומדגישות את המיתוג כוללות "בוגרי ממר"ם", "טכנולוגיה", "חדשנות", ו"קריירה". זה מעיד על מיתוג חזק כפורמט שמחבר בין היסטוריה מקצועית של בוגרים לבין העולמות העסקיים והטכנולוגיים המודרניים.

● את ההצלחה של הפודקאסט ניתן למדוד באמצעות מדדים כמו מספר חיפושים, האזנות, לייקים, ומשובים כתגובות. כדי לשפר את המדידה, ניתן לאסוף נתונים על זמן האזנה ממוצע, שיעור האזנות חוזרות, ותובנות מהרשתות החברתיות בהן הפודקאסט מופץ.

● המאפיינים שתרמו לפודקאסטים עם הכי הרבה לייקים הם ככל הנראה נושאים מעוררי השראה, ראיונות עם דמויות מפתח בתעשייה, ותוכן רלוונטי לקהל היעד.

● שדות נוספים שיכולים לתרום להפקת תובנות כוללים את קהל היעד המדויק, תגובות על התוכן, ואפיון הנושאים של כל פרק מבחינת מידת הטכניות, העניין הציבורי, ומידת ההשראה שהם מספקים. היה גם מעניין לראות על ציר זמן עבור כל פרק מתי היו הכי הרבה מאזינים, ומתי המאזינים "פורשים" או מפסיקים להאזין במהלך הפרק.

● לגבי פודקאסט עם מספר חיפושים גבוה, ניתוח התיאור מראה שהתיאור מספק מידע בסיסי על נושא הפרק והאורחים, אך ניתן לשפר על ידי הוספת מילות מפתח חיפוש נפוצות או שאלות שיעוררו סקרנות.

תודה רבה!