המואזין - المؤذن

במהלך השנים האחרונות, גורמי בטחון ישראלים מתמודדים עם אתגרים מורכבים בזירות פעולה שונות. לאחרונה, אנו עדים להתפתחות של איום חדש, המאופיין בשימוש בפלטפורמות דיגיטליות, ובראשן פודקאסטים, כדי להפיץ הסתה נגד ישראל ומוסדותיה ברחבי העולם.

איום זה מהווה סיכון ישיר לביטחונן של שגרירויות ונציגויות ישראליות בחו"ל. בעבר, השתמשנו במערכת ייעודית שפיתחנו על מנת לנטר תכנים מסוג זה ולספק התרעות מבעוד מועד, ובכך הבטחנו את שלומם של אנשינו.

נכון ליום חמישי האחרון, המערכת קרסה.

כתוצאה מכך, אנו נותרנו ללא כלי הניטור החיוני, וקיים סיכון מוגבר על אתרינו בעולם.

לפיכך, אנו פונים אליכם, אנשי ה ,Data-לצורך פיתוח מערכת ניטור חלופית, שתשמש כפתרון זמני.

משימתכם, היא לבנות מערכת לניטור תוכן פודקאסטים, ולאפשר יכולת התרעה על בסיס התוכן הקיים בהם בכדי למנוע פעולות חבלניות עוינות בגורמים ישראליים ברחבי העולם.

בשל דחיפות המשימה,

אתם נדרשים לפתח פתרון מהיר בטווח זמן של <u>ארבעה ימים</u>. הבטחת ביטחונם של אנשינו תלויה בהצלחתכם המהירה.

המואזין - المؤذن

הנחיות כלליות למבחן:

- **שימוש נכון ב– Git-**לאורך כל המבחן, בצעו קומיטים מסודרים עם הודעות ברורות. הקפידו על שלבים קטנים ומשמעותיים, כדי שיהיה קל לעקוב אחרי התקדמות הפיתוח.
- **ארגון הפרויקט** – חלקו את הקוד למודולים, חבילות וקבצים בצורה הגיונית ונקייה. חשוב שמי שיגיע אחרי יוכל להבין את המבנה ואת האחריות של כל רכיב בלי להתאמץ.
- **עקרונות SOLID** – שמרו על כתיבת קוד גמיש, קריא וניתן להרחבה.
- **קונסיסטנטיות** – שמרו על שמות אחידים, סגנון אחיד של קוד ותיעוד קצר היכן שנדרש.
- **Naming** – בחרו שמות משמעותיים וברורים למשתנים, פונקציות וקבצים. שם טוב חוסך הסברים ומקל על מי שיקרא את הקוד אחריכם.
- **החלטות פיתוח** – כל החלטה שתעשו (מבנה, ספרייה, טכנולוגיה) תשאירו עקבות.
  - בקובץ **README** מסודר – תעדו מה בחרתם ולמה.
  - ב– **Git-**הקפידו שהקומיטים ישקפו את שלבי הפיתוח, חשוב לנו לא רק לראות את הקוד, אלא גם להבין את **כיוון החשיבה שלכם** לאורך הדרך.
- **חשוב מאוד: בסוף, כל מה שבניתם חייב לרוץ בתוך.Docker containers**

זה כולל גם את השירותים עצמם וגם את התלויות (DB, Elastic, Kibana וכו.('

שימו לב להחלטות שאתם מקבלים בדרך, יש להם השלכות.

המואזין - المؤذن

שלב א

לכל סיפור יש התחלה, במקרה שלנו נצטרך להתחיל לקלוט קבצים, בכדי שלא "יתפוצץ לנו חדר הדואר" הווירטואלי שלנו מכמויות חומרים אדירים. משימתכם היא - לבנות שירות שימשוך קבצים ממערכת הקבצים הלוקאלית במחשב, המערכת תעבד אותם, ותשלח את הפלט ל TOPICשל.KAFKA

את הקבצים תורידו מהקישור) הבא מומלץ לשמור את הקבצים בתיקיה הראשית של המחשב האישי.(

תהליך העיבוד יצטרך לבצע את הפעולות ההבאות:

1. **קריאת הקובץ** – קריאה של הקובץ לפי הנתיב המדויק שלו בתיקיה המקומית (לרכיב הבא שלכם אתם תצטרכו לשלוח נתיב לקובץ ולא את הקובץ עצמו, שם כבר תטפלו בקובץ עצמו.(
2. **יצירת פרטי המעטפת – (Metadata)** את שליפת נתוני ה - metadata תכלו ליישם למשל באמצעות ספריית ,pathlib כדי לקבל מידע על הקובץ (גודל, שם, תאריך יצירה וכו.('
3. **בניית JSON ייעודי** – שילוב כל הנתונים (נתיב הקובץ ו-metadata) למבנה JSON מסודר, תוך שמירה על מידול הנתונים.
4. **שליחה ל-Kafka** – פרסום ה-JSON ל-TOPIC המתאים ב-.Kafka

המטרה היא שהשירות יהיה **יציב, מסודר ויעיל**, ויוכל להתמודד עם כמויות גדולות של קבצים מבלי לפגוע במערכת או במידע.

המואזין - المؤذن

שלב ב

כעת, לאחר שיש לנו תשתית יציבה שמכניסה את הקבצים ל-Kafka, מגיע החלק המרכזי – **שירות הצריכה והעיבוד של המידע**.

השירות צריך לקחת את המידע שמגיע ל-TOPIC, לפרק אותו לחלקים השונים, ולשלוח כל חלק למאגר המתאים בצורה מסודרת:

- **חישוב מזהה חד-חד ערכי** (Unique ID)
  - לכל מסמך שנכנס נחשב מזהה ייחודי על בסיס הנתונים של המסמך עצמו. זה חשוב כדי שנוכל לזהות את אותו מסמך בכל מאגר בו הוא נשמר.
- **שליחת חלקי המסמך:**
  - כל פרטי הקובץ (שם, גודל, תאריך יצירה וכו' או בקצרה Metadata) יישלחו לאינדקס שתיצרו ב-Elasticsearch עם המזהה המחושב, כדי שנוכל לבצע חיפושים וניתוחים יעילים.
  - התוכן הממשי של הקובץ יישמר ב MongoDB, גם כן עם המזהה מהסעיף הקודם, כדי לשמור על קשר בין Metadata לתוכן.(שימו לב חקרו איך אתם מעלים למאגר קובץ)

שימו לב!! חשוב שגם אנחנו, אנשי ה־Data, נוכל לראות תמונה ויזואלית וברורה של המידע שהמערכת אוספת ומנתחת.
לכן, עליכם להוסיף את **Kibana** לסביבת ה־Elasticsearch שלכם, ולוודא שב־**Discover** יוגדרו האינדקסים הרלוונטיים.