

28.8.22

עיצוב ופיתוח מערכות נתוני עתק ומחשוב ענן פרויקט סיום תשפ"ב – סמסטר קיץ

כללי:

בתקופה הנוכחית, בנוסף לנתונים המופקים ממערכות תפעוליות, מאותרים ונאספים נתונים רבים ממגוון מקורות כבסיס לניתוח, גיבוש תובנות, אימוץ אסטרטגיות וקבלת החלטות מבוססת ראיות.

מערכות Big Data Analytics דורשות משאבים ארגוניים רבים ולכן שימוש בספקי שירותי ענן לאחסון ועיבוד הם פרקטיקה נפוצה. בתכנון מערכת מבוצרת מסוג זה אין מניעה לטווח מגוון שירותים ממספר ספקים, בשילוב שירותים שמקורם בארגון עצמו, לכלל פתרון שלם.

בפרויקט הנוכחי ניצור פתרון לעיבוד ותצוגת נתונים בהשראת **תבנית למדה**, האוספת נתוני מכירת גלילות ממגוון סניפים, יוצרת מודל חיזוי צריכה ומסייעת בקבלת החלטות ייצור – הפתרון ישלב שירותי ענן ושירותי Web לכלל מערכת שלמה.

מהות הפתרון:

מערכת הבנויה משלוש תת מערכות שיחדיו מאפשרות ניטור Near Real Time למלאי טעמי גלילות בסניפים, באמצעות Dashboard, וכן יצירת מודל חיזוי, עבור סניף מסוים, מה עומד להיות פרופיל מכירותיו ביום מסוים.

דרישות פונקציונאליות:

1. רשת גלילות מציעה חמישה טעמי גלידה : שוקולד, וניל, תות, לימון וחלבה.
2. המערכת תרכז נתוני הסניפים ומאפייניהם (לפחות 100 סניפים) :
 - שם הסניף
 - יישוב
 - פרטי בעלים
 - מלאי נוכחי לכל טעם גלידה
3. המערכת תציג ב- Real Time באמצעות דשבורד וגראפים מתאימים מלאי כל אחד מהטעמים בכל סניף (Bar Chart), סה"כ מלאי הטעמים הכולל בכל הסניפים (Pie Chart) וכן תחזית מכירות סניף מסוים לטעם מסוים לשבוע הקרוב.
4. המערכת תאפשר ללמוד מנתוני עבר אודות צריכת טעמי גלידה ויצירת תחזית צריכה עתידית מתוך הנתונים הבאים:
 - יום בשבוע
 - חודש
 - עונה
 - האם שבוע חג
 - מזג אוויר באותו יום (חם מאוד, חם, נעים, קר, קר מאוד)
 - רמת צריכה (לכל טעם):
 - אפסית (מתחת לקילו)
 - מועטה (עד 20 קילו)
 - בינונית (עד 60 קילו)
 - גבוהה (עד 120 קילו)
 - גבוהה מאוד (מעל 120 קילו)
- גודל היישוב (יש להסתפק בשמונה ערכים בלבד, ניתן לבצע קיבוץ ערכים ולהגדיר סולם אורדינלי).
- סוג אוכלוסיה (יהודים, ערבים, מעורב).

- קבוצות גילאים ביישוב (ניתן לתרגם לאחוזים ולדרג כרצונכם)
 - פעוטים (גיל 0-5)
 - ילדים ונוער (גיל 6-18)
 - בוגרים (גיל 19-45)
 - מבוגרים (46-55)
 - עמידה (56-64)
 - זהב (65 ומעלה)

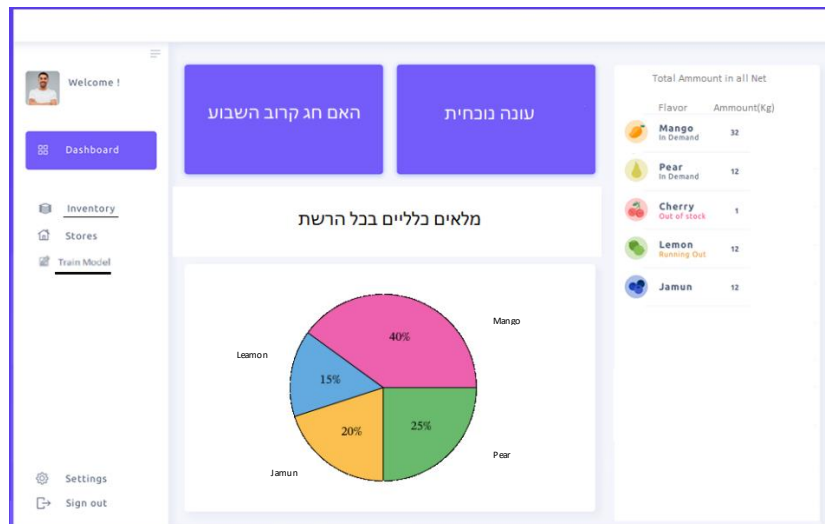
דרישות לא פונקציונאליות:

1. המערכת תפותח בתפיסה מונחית שירותים בגישת **Micro Services**.
2. ארגון השירותים בסביבת Node.js תתבסס על תבנית MVC (השתמשו ב-express כמרכיב View המוחלל תצוגה).
3. המערכת תתבסס על שלוש תת מערכות ושירותות Kafka שימש כ- Message Broker לאינטגרציה ביניהם.
4. שלוש תת מערכות הן:
 - א': תת מערכת תפעולית שתיוצג על ידי סימולטור השולח מסרים למתווך עם נתוני מכירות.
 - ב': תת מערכת Dashboards ונתוני זמן אמת.
 - תת המערכת תשמור במסד נתונים מסוג Redis את סטטוס מלאי כל טעם גלידה בכל אחד מהסניפים.
 - אחד המסכים ישמש לבקשת למידת מודל מתת מערכת ג' (שתכין סט נתונים ותעביר ל-BigML), כמוכן ניתן יהיה באותו מסך לבצע חיזוי צריכת גלידה בסניף מסוים, בתאריך מסוים לטעם גלידה מסוים.
 - ג': תת מערכת אחסון נתונים היסטוריים ונתוני רקע האחראית על תהליך ELT (Extract, Load, Transform):
 - תת המערכת תשמור במסד נתונים MySQL פרטים אודות הישובים ומאפייניהם (יש לטעון אותם חד פעמית מהקישורים להלן).
 - עם הגעת מסר עם נתוני מכירה, יחולצו נתוני מזג האוויר הנוכחי ביישוב ובירור אודות חג בשבוע הקרוב משירותי רשת (ראה להלן), נתוני המסר והנתונים הנוספים יישמרו במסגרת MongoDB כדוקומנט.
 - בהנתן הוראה מכינה המערכת סט נתונים, מוסיפה לכל אירוע מכירה את מאפייני היישוב ופונה לשירות BigML.Com כדי ליצור מודל חיזוי מסוג עץ החלטה (ראו הסברים בלינקים להלן).
5. ניתן להשתמש בחבילת Axios בכדי לגשת לשירותי מזג האוויר ומועדי ישראל.
6. ניתן להשתמש במקורות הבאים לקבלת נתונים רלוונטים:

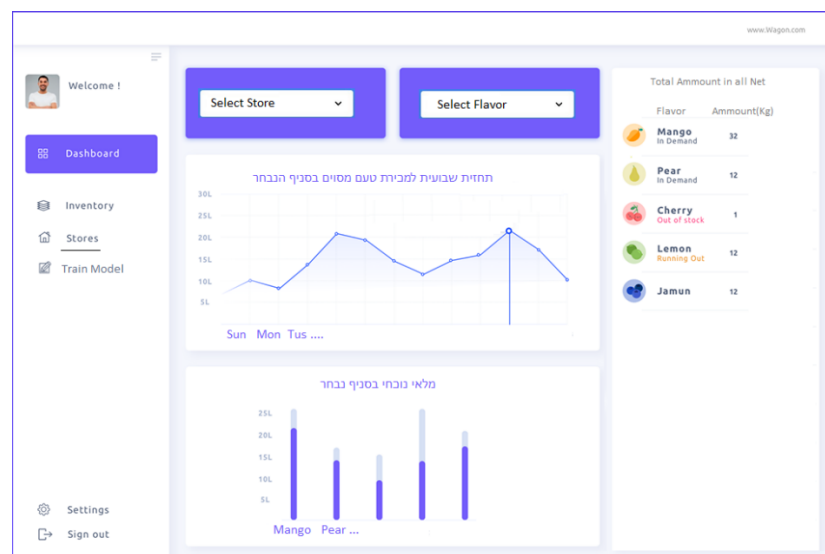
כתובת	סוג נתונים
https://www.cbs.gov.il/he/settlements/Pages/default.aspx?subject=מדד חברתיכלכלי	נתוני יישובים - כללי
https://data.gov.il/dataset/residents_in_israel_by_communities_and_age_groups	נתוני יישובים - גילאים
https://openweathermap.org/	נתוני מזג אוויר
http://lms.data.gov.il	נתוני מזג אוויר
https://www.hebcal.com/home/219/hebrew-date-converter-rest-api	מועדים וחגי ישראל

<https://www.softwaretestinghelp.com/decision-tree-algorithm-examples-data-mining/>
<https://www.keboola.com/blog/etl-process-overview> (also etl vs. elt)

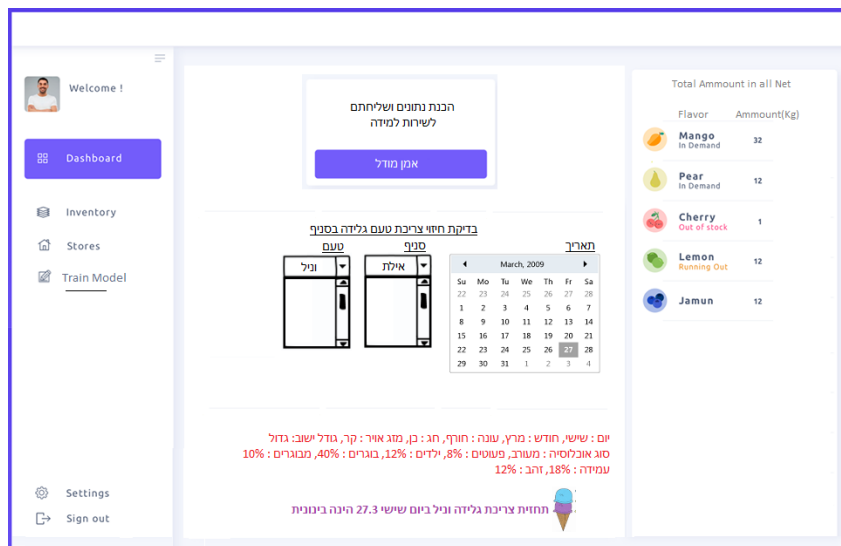
7. Kafka מומלץ להריץ בענן ולהשתמש בספק : <https://www.cloudkarafka.com/>
8. Redis מומלץ להריץ במסגרת Docker עם ה-image https://hub.docker.com/_/redis
9. MongoDB ניתן להריץ לוקאלית, ע"ג Docker Image או משירות ענן כגון : <https://www.mongodb.com/>
10. במסגרת תת מערכת ב' : הקשר בין השרת לדשבורד (דפדפן) יעדכן נתונים באמצעות פרוטוקול Socket.io
11. תת מערכת ג' תאפשר לנתח את הנתונים ההיסטוריים באמצעות אלגוריתם למידת מכונה וליצור מודל חיזוי מכירות, על ידי שימוש בשירותי BigML.com וחבילת Node המתאימה.
12. בונוס : תת מערכת ב' תשתמש בתשתית Bing Maps להצגת הסניפים ע"ג מפה.



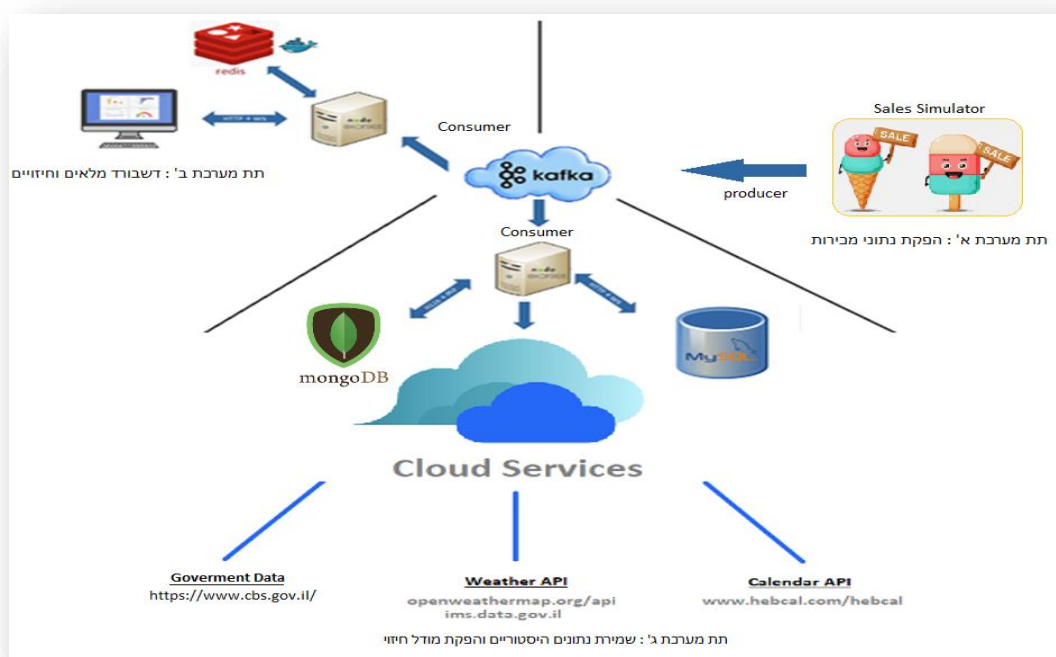
מסך 1: דשבורד הצגת נתוני מלאי כללי ברשת



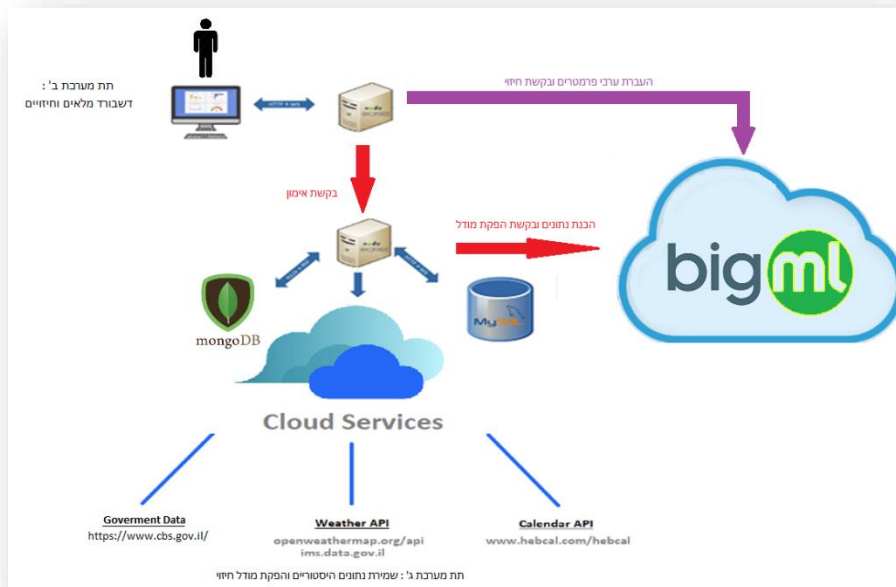
מסך 2: דשבורד תצוגת נתוני מלאי בסניף מסוים ותחזית שבוע קדימה



מסך 3: בקשות לאימון מודל ובדיקת תחזית ליום מסוים בסניף מסוים מטעם מסוים



דיאגרמה 1: שלושת תת המערכות בפתרון – זרימת נתונים לאחסון ותצוגה



דיאגרמה 2 : הפקת מודל חיזוי ובקשת חיזוי

הערות:

1. ההגשה בשלישיות אך ניתן גם בזוגות.
2. לכל סטודנט אחריות על מרכיב במערכת, הציונים בהגשה הם אינדיבידואלים.
3. **נא להרשם להגשה בגיליון הרישום בלינק זה.**
4. מימוש למידת מכונה באמצעות BigML הינה אופציה למגישים בזוגות אך חובה לשלישיה. מי שאינו משתמש ב-BigML יאתר חבילת ניתוחים לוקאלית לבניית מודל חיזוי.
5. מוזמנים להשתמש בדוגמאות הקוד במודל אך יש להתאימם לפתרון שלכם.
6. את הקוד ותאור הפרויקט יש לתעד ב- GitHub.
7. תאריך ההגשה : 19.10 באמצעות זום (יתכן שינוי נא לעקוב בגיליון הרישום).
8. בנוס: פריסת תת מערכת א' ע"ג ספק שירותי אירוח כגון <http://heroku.com> כמתואר כאן: <https://www.geeksforgeeks.org/deploying-node-applications/?ref=rp>

בהצלחה.