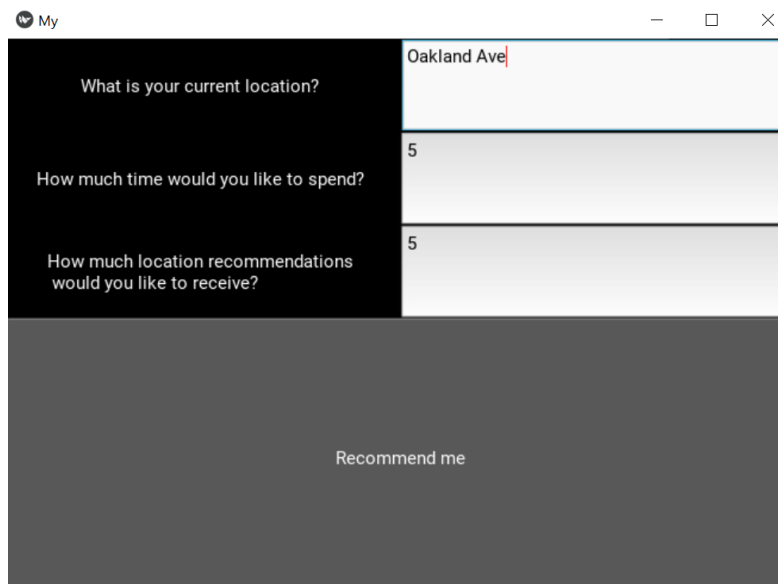


## עבודת בית 2 – GUI, Data-Base, Web-Service

במטלה זו עליכם לבנות אפליקציה להמלצות לטיולי אופניים בעיר ניו-יורק.

דוגמה לממשק **בסיסי**:



איור 1 – דוגמה ל-GUI בסיסי של המערכת

### קבצים וארכיטקטורה:

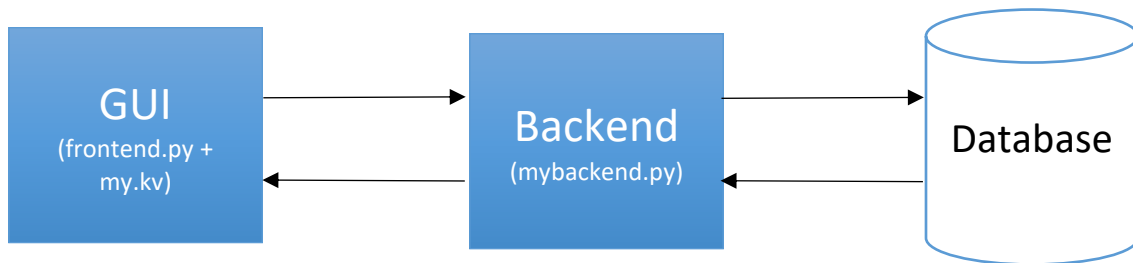
יש לממש את האפליקציה עם הקבצים הבאים:

1. frontend.py – קוד ה-GUI (שם נעשה import ל-backend)
2. mybackend.py – הקוד שמתממשק עם ה-DB, אותו יש ליצור באמצעות SQLITE כפי שנלמד בכיתה.
3. mywebservice.py – הקוד שמתממשק עם שירות ההמלצות עבור נסיעה חדשה.
4. my.kv – קונפיגורציה עבור עיצוב הדף על ידי שימוש ב-kivy.

### עליך לתמוך בפונקציונאליות הבאה:

חלק א' - my.kv + frontend.py – GUI:

עליכם לממש את חלק זה על סמך הארכיטקטורה הבאה:

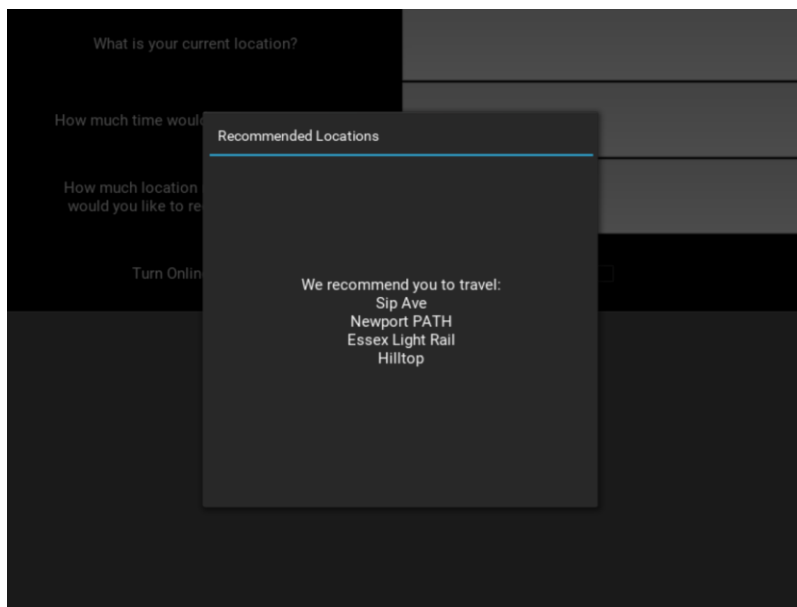


איור 2 – ארכיטקטורת מערכת עבור חלק א'

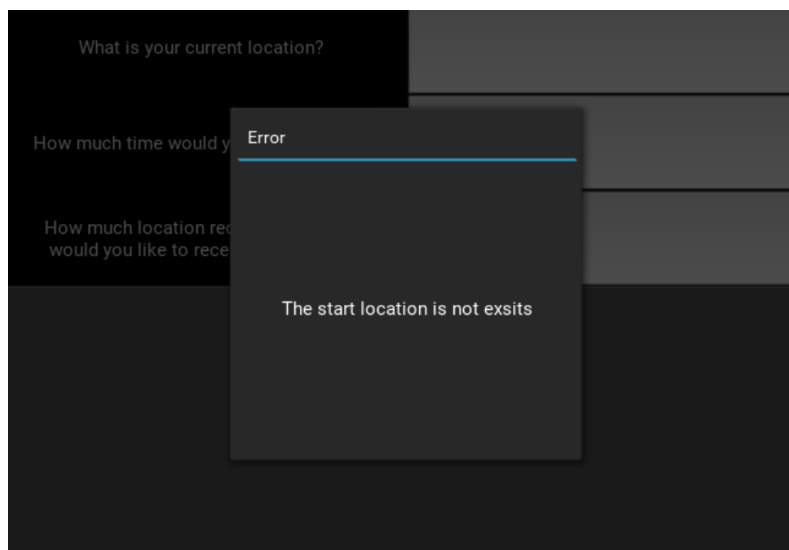
1. עליך לתמוך ב-grid בשביל למקם את האלמנטים השונים (ראה איור 1).
2. למשתמש תהיה את האפשרות להזין טקסט עבור שם של יעד התחלת הטיול, אורך זמן הטיול שהוא מעוניין לבצע, וכמות ההמלצות שהוא מעוניין לקבל. ההמלצות שיקבל המשתמש יהיו ממוינות מהיעד הכי מומלץ (יופיע בראש הרשימה) ליעד הכי פחות מומלץ (יופיע בתחתית הרשימה).
3. לאחר לחיצה על כפתור Submit יוקפץ חלון (Popup) עם מספר המלצות כפי שניתן לראות באיור 3.
4. מספר דגשים:
  - לאחר לחיצה על כפתור Submit, הטקסט בכל התיבות יימחק.
  - יש לבצע בדיקות קלט – לדוגמה: אם המשתמש הזין טקסט שאינו מספר בשדה של מספר המלצות שהוא מעוניין לקבל, יש להקפיץ הודעה מתאימה עבור שגיאה זו (Popup).
  - יש לבצע בדיקות תקינות לוגיקת התוכנית – לדוגמה: אם המשתמש הזין מקום שאינו קיים בבסיס הנתונים יש להקפיץ הודעה מתאימה עבור שגיאה זו (Popup).

#### :Backend – mybackend.py

1. בקובץ זה עליך להקים מחלקה בשם Database אשר רק באמצעותה ניתן לתקשר עם בסיס הנתונים.
2. בתחילת ריצת התוכנית - בסיס הנתונים ייווצר באמצעות טעינת קובץ csv (BikeShare.csv) אשר מצורף למטלה) אל מאגר הנתונים הרלציוני (sqlite3) של האפליקציה אשר ייווצר בשם database.db.  
**בסיס הנתונים ייווצר באופן חד פעמי – כלומר, יש לוודא כי לאחר יצירת בסיס הנתונים, בסיס הנתונים אינו ייווצר בשנית.**
3. כל השאילתות הדרושות לעבודה תקינה של האפליקציה (כלומר, הכנסת נתונים, קריאת נתונים וכולי) יהיו בפורמט של SQL.



איור 3 – דוגמה לקבלת 4 המלצות מהמערכת



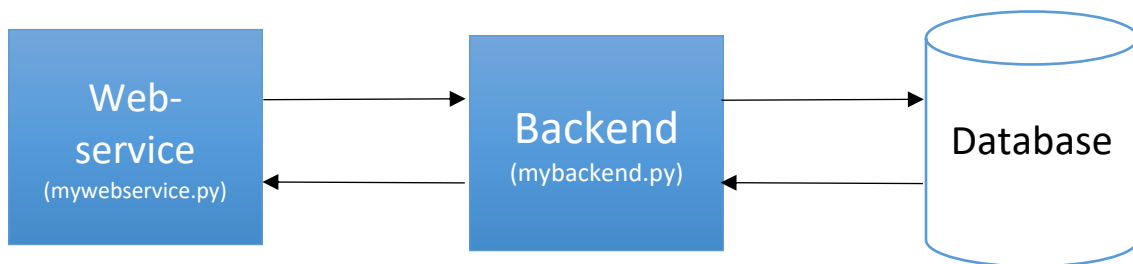
איור 4 – דוגמה להודעה מתאימה עבור מיקום שאינו קיים



### חלק ב' - Web-service – mywebservice.py:

שימוש בקובץ זה יצור המלצות במקום שימוש ב-GUI.

עליכם לממש את חלק זה על סמך הארכיטקטורה הבאה:



איור 3 – ארכיטקטורת מערכת עבור חלק ב'

בעזרת חבילת flask בקובץ זה תממשו web service של מערכת המלצה כך שכל הפונקציות הדרושות לצורך הפעלת השירות ייכתבו בקובץ זה בלבד.

השירות יאפשר גישה ל URL בשיטת GET. כאשר הפרמטרים שישלחו לכתובת הינם:

- startlocation - יעד התחלת הטיול
- timeduration - אורך הטיול (מבחינת זמן - בדקות)
- k - מספר המלצות שברצונו לקבל.

דוגמה לבקשת GET כאשר מקום התחלת הטיול הינו Oakland Ave, הזמן שברצון המשתמש לטייל הינו 5 דקות, ומספר ההמלצות שהוא מעוניין לקבל עומד על 5 המלצות:

`http://127.0.0.1:5000/?startlocation=Oakland+Ave&timeduration=5&k=5`

השיטה תחזיר מבנה מסוג JSON אשר יכיל את שמות K היעדים המומלצים לטיול, לדוגמה:

```
[  
  "Christ Hospital",  
  "City Hall",  
  "Sip Ave",  
  "Newark Ave",  
  "Exchange Place"  
]
```

שירות ה-Web שתצרו יתקשר עם המחלקה Database עם כל שאילתות ה-SQL הדרושות עבור אחזור ההמלצות. בנוסף, יש להשתמש בדיוק באותו אלגוריתם שמבצע המלצות ב-GUI.



### חלוקת הניקוד במטלה תתחשב בדברים הבאים:

1. יש לשמור על תקינות הקלט ולהציג הודעות שגיאה מתאימות.
2. יש לתעד את הקוד ([דוגמה](#)).
3. קוד מסודר - יש לשמור על חלוקה נכונה לפונקציות ועל מודולאריות האפליקציה.
4. יינתן ניקוד גבוה יותר על יצירתיות במימוש האלגוריתם של ההמלצות, ועיצוב מתקדם של ה-GUI.

### הוראות - הגשה:

- יש להגיש את המטלה עד לתאריך 09.12.2020 בשעה 23:55.
- ההגשה תתבצע ע"י קובץ zip המכיל את הקוד שנכתב בהתאם לדרישות. פורמט קובץ ה-zip יהיה ID01\_ID02.zip כאשר הקובץ יכיל:
  - פרויקט - כל הקבצים הדרושים לריצה תקינה של הפרויקט (py, kv).
  - דו"ח - קובץ pdf - בדו"ח יש לפרט:
    - o באיזה אלגוריתם השתמשו לצורך מתן המלצות, השתדלו לפרט בדו"ח על דרך המחשבה שלכם ולהצדיק את השימוש באלגוריתם שפיתחתם. בנוסף, במידה וביצעתם ניתוחים כלשהם על ה-Data הציגו אותם בדו"ח.
    - o יש לפרט בדו"ח על שתי דוגמאות של המלצות שניתנו למשתמש עבור שאילתות שונות (ניתן לצרף תמונות מסך של ה-GUI שיצרתם).
- ההגשה הינה בזוגות.
- אין להגיש את קובץ בסיס הנתונים (db) וקובץ ה-csv.
- חבר קבוצה אחד בלבד יעלה את הפתרון לאתר.
- בעיות אישיות בנוגע למועד ההגשה יש להפנות לבודק התרגילים הקורס טרם מועד ההגשה.
- כל חריגה מנהלים אלו, ללא אישור בכתב מצוות הקורס, מהווה עילה לפסילת המטלה או להפחתת נקודות.
- אין להעתיק פתרונות ואין לשתף קוד בין סטודנטים. אין להעתיק קוד מוכן באינטרנט!
- לפתרון המטלה יש להשתמש בגרסת Python 3.6 בלבד.
- להבהרות, הכוונות או כל עזרה אחרת יש לפנות לפורום המטלות במודל.
- בדיקת המטלה תתייחס בין השאר לפרמטרים הבאים: תיעוד הקוד, נכונות הקוד, יעילות הקוד וזמני ריצה. יש לבדוק מקרי קצה.

בהצלחה!