# Python Full Stack Web Developer

# Project – Part II



**הגשה 20.9.2025**

# כללי



**זהו החלק השני של הפרויקט המתגלגל.**

**בחלק זה עליכם לבנות את כלל המערכת על גבי Python Flask ולממש את המסכים הדרושים עבור המערכת.**

**טכנולוגיות פיתוח:**

* מסד הנתונים: MySQL.
* צד השרת:Python – Flask .
* צד הלקוח: HTML, CSS, JavaScript. react

**סוג הפרויקט:**

* Python ע"י שימוש ב-Flask Framework.
* יש לייצר סביבה וירטואלית המכילה את כלל ההתקנות הדרושות ולעבוד מתוך סביבה זו.
* על ההתקנות להיות מוגדרות בקובץ requirements.txt.

**להלן תזכורת לפונקציונליות שבניתם בחלק הראשון של הפרויקט:**

* הרשמה למערכת.
* כניסת משתמשים רשומים.
* החזרת כלל החופשות.
* הוספת חופשה חדשה.
* עדכון חופשה קיימת.
* מחיקת חופשה קיימת.
* ביצוע Like לחופשה.
* ביצוע Unlike לחופשה.

שימו לב: עליכם להעתיק את הקוד המתאים מהפרויקט הקודם לפרויקט החדש, שכן הפרויקט החדש בנוי בסביבת Flask. כמו כן, אין בעיה להוסיף או לשנות קודים שכתבתם אם לדעתכם יש מימוש נכון יותר (לדוגמה אם בוצעה בדיקת וולידציה ב-Façade אך היא בעצם צריכה להיות ב-Model).

**תוכן מסד הנתונים בהגשת הפרויקט:**

* שני תפקידים: Admin ו-User.
* שני משתמשים, האחד מסוג Admin והשני מסוג User.
* לפחות 10 מדינות אמיתיות.
* לפחות 12 חופשות, עם נתונים הגיוניים.
* טבלת Likes ריקה.

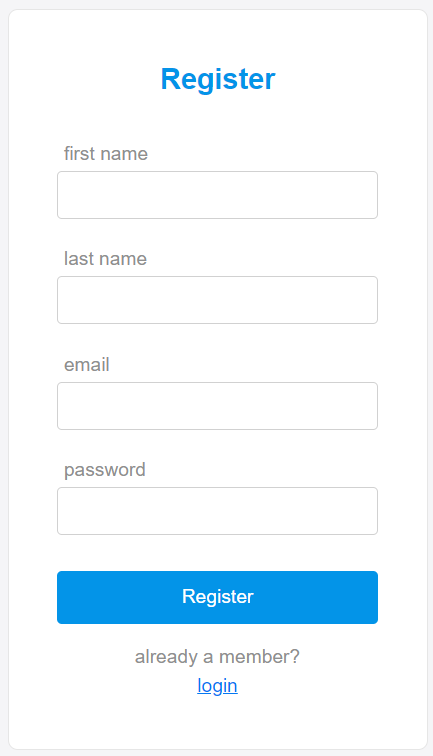
# פירוט המסכים



להלן המסכים הדרושים עבור המערכת בהתאם לפונקציונליות שבניתם בחלק הראשון.

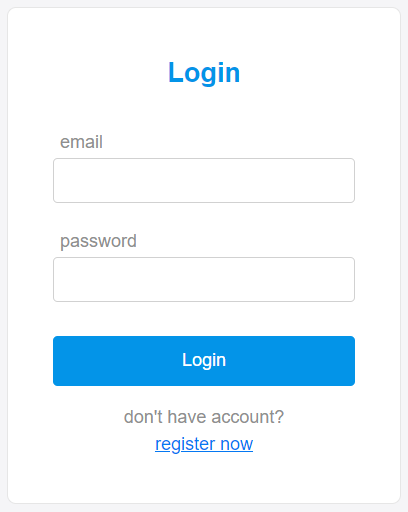
שימו לב: ממשק המשתמש המוצג כאן הינו להמחשה בלבד. אין צורך לבצע בדיוק את אותו הדבר. ניתן לבנות כל ממשק משתמש אחר שהינו אסתטי ועונה על האפיון.

**דף הרשמה למשתמשים חדשים:**

****

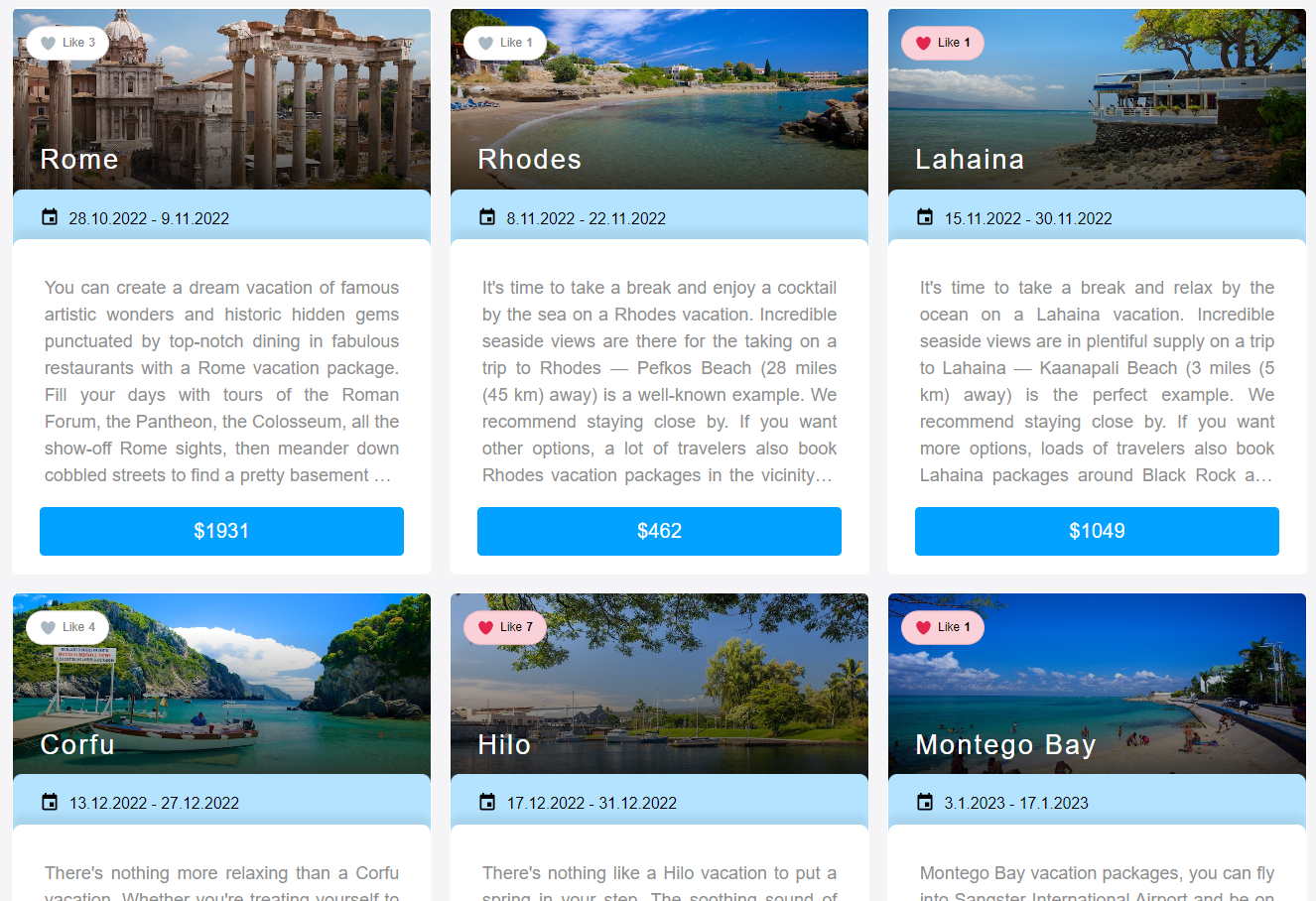
* **כלל השדות הינם שדות חובה.**
* **יש לוודא שהאימייל חוקי.**
* **סיסמה חייבת להיות מינימום 4 תווים.**
* **יש לבצע בדיקה שהאימייל אכן פנוי. אין לאפשר להירשם ע"י אימייל תפוס.**
* **בסיום ההרשמה יש לנתב לדף החופשות.**

**דף כניסה למשתמשים רשומים:**

****

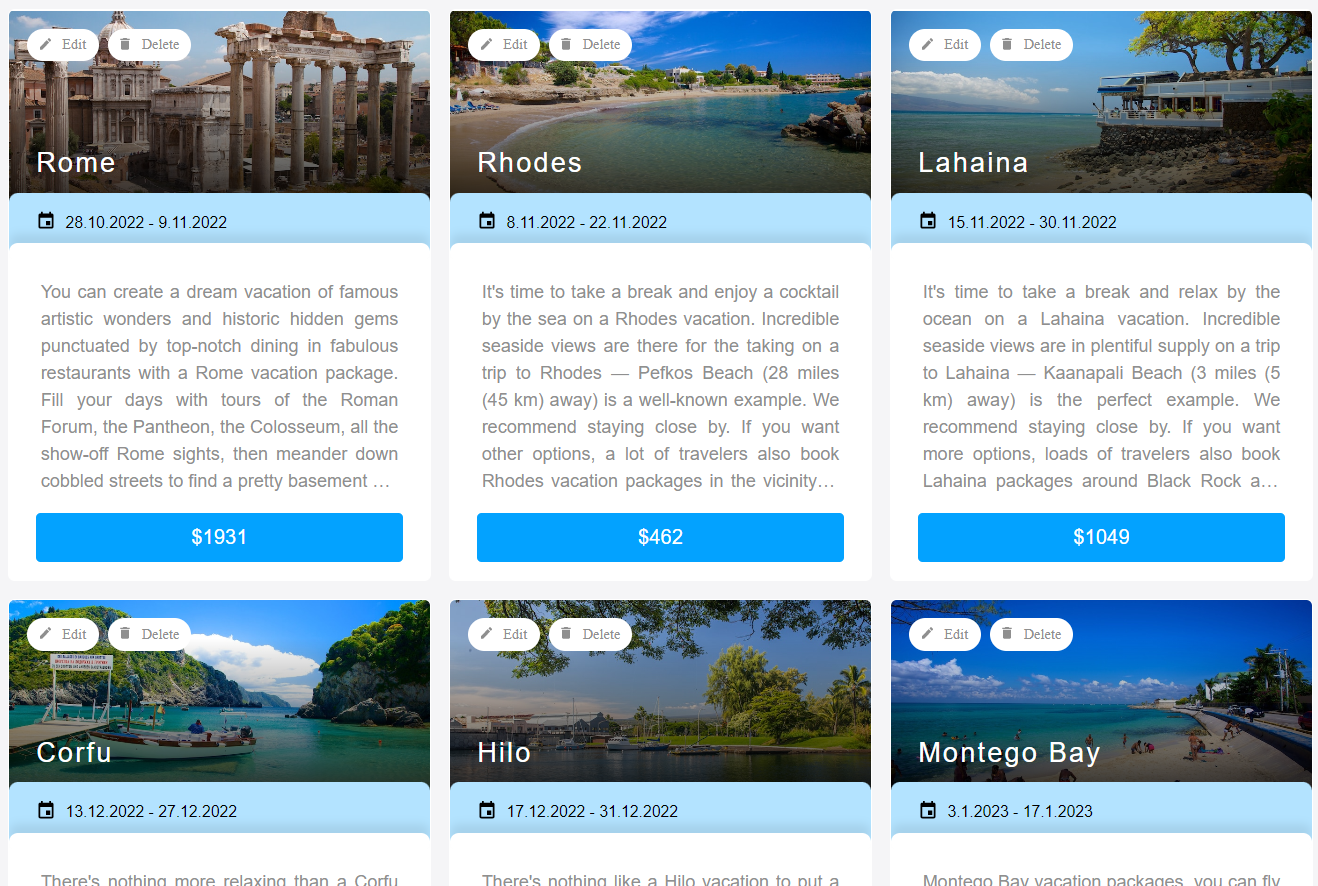
* **כלל השדות הינם שדות חובה.**
* **יש לוודא שהאימייל תקין.**
* **סיסמה חייבת להיות מינימום 4 תווים.**
* **עבור פרטי כניסה שגויים יש להציג הודעת שגיאה מתאימה.**
* **אם הפרטים נכונים יש לנתב לדף החופשות.**
* **ניהול token ופרטי משתמש באמצעות context**
* **סיסמא מוצפנת בצד שרת**

**דף החופשות:**



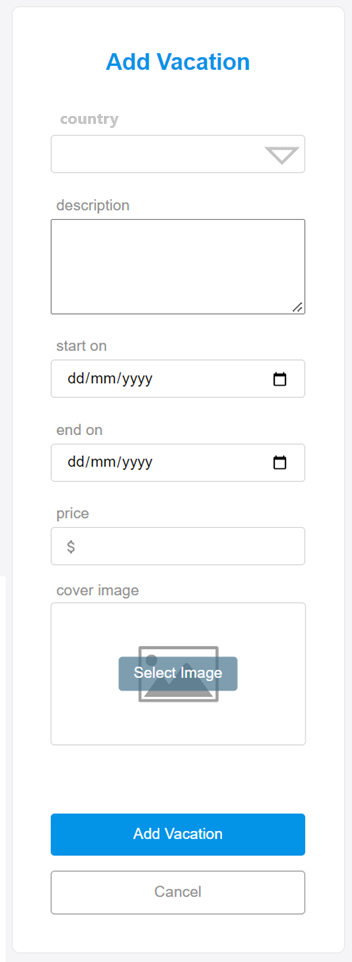
* דף זה נגיש לכולם אך ניסיון למחוק לערוך ןליצור חופשה מחייב admin ולוגאין .
* משתמש המנסה להגיע לדף עריכה או עדכון חופשה Logged-In מנותב לדף Login.
* על החופשות להיות מוצגות בכרטיסיות (לא בטבלה).
* בכל כרטיסיה ניתן לראות את כלל הפרטים של החופשה (ללא קודים של מסד הנתונים כמובן).
* בכל כרטיסיה ניתן לראות את מספר ה-Likes שיש לאותה החופשה מכלל המשתמשים.
* בכל כרטיסיה ניתן לראות האם המשתמש הנוכחי ביצע Like אחרי חופשה זו או לא.
* על החופשות להיות ממוינות לפי תאריך התחלת החופשה בסדר עולה.
* משתמש יכול לבצע Like לחופשה.
* משתמש יכול לבצע Unlike לחופשה עבורה הוא ביצע Like (כלומר להוריד Like שביצע).
* יש להציג מקסימום 3 חופשות בשורה.

**דף מנהל:**



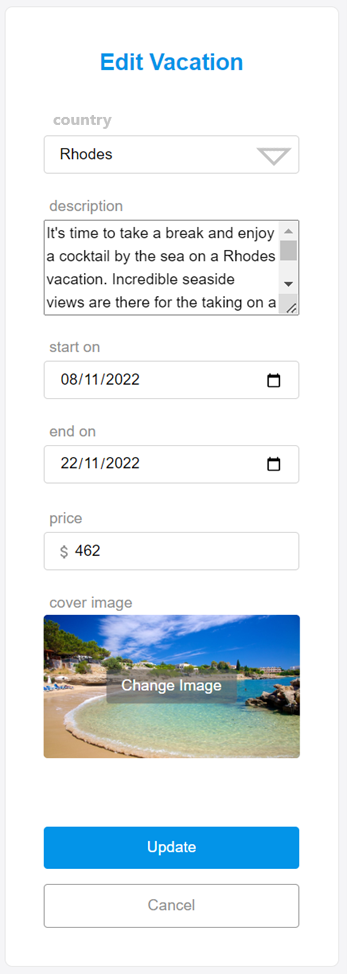
* המנהל **לא** יכול לבצע Like או Unlike ואין להציג למנהל אפשרויות אלו בכרטיסיה.
* בראש הדף יש להציג לחצן עבור הוספת חופשה חדשה המפנה לדף הוספת חופשה.רק בתנאי שאתה אדמין
* בכל כרטיסיה יש להציג לחצן עבור עריכת החופשה המפנה לדף עריכת חופשה.רק בתנאי שאתה אדמין
* בכל כרטיסיה יש להציג לחצן עבור מחיקת החופשה.רק בתנאי שאתה אדמין
* אם המנהל מנסה למחוק חופשה, יש לוודא האם הוא בטוח, ואם לא – אין לבצע את המחיקה.

**דף הוספת חופשה:**



* דף זה נגיש אך ורק למנהל.
* כלל השדות הינם שדות חובה.
* אין לאפשר להכניס מחיר שלילי או גבוה מ-10,000.
* אין לאפשר לבחור תאריך סיום המוקדם מתאריך ההתחלה.
* אין לאפשר לבחור תאריכי עבר.
* על התמונות להישמר בתיקייה בצד השרת.
* במסד הנתונים יש לשמור אך ורק את שם קובץ החופשה (כולל סיומת).

**דף עריכת חופשה:**



* דף זה נגיש אך ורק למנהל.
* כלל השדות הינם שדות חובה פרט לשדה התמונה, אותה יש להציג ולאפשר שינוי אך לא לחייב זאת.
* אין לאפשר להכניס מחיר שלילי או גבוה מ-10,000.
* אין לאפשר לבחור תאריך סיום המוקדם מתאריך ההתחלה.
* יש לאפשר בחירת תאריכי עבר מפני שניתן לערוך חופשה שכבר הסתיימה.

**תפריט ראשי:**

* יש להציג תפריט ראשי התלוי במשתמש.
* יש להציג את השם המלא של המשתמש עבור משתמשים רשומים (כולל מנהל).
* משתמשים לא רשומים יכולים לבצע הרשמה או כניסה בלבד.
* משתמש רשום יכול לצפות בדף החופשות, לבצע Like או Unlike לחופשה ולבצע Logout מהמערכת.
* מנהל יכול לצפות בחופשות, להוסיף חופשה, לעדכן חופשה, למחוק חופשה ולבצע Logout מהמערכת.

# הגשת הפרויקט



* יש לבצע Export למסד הנתונים ולשמור אותו בתיקיית הפרויקט.
* יש לוודא שבקובץ ה-requirements.txt קיימות כלל ההתקנות הדרושות (ביצוע pip freeze).
* יש למחוק את תיקיית הסביבה הווירטואלית.
* יש לבצע קובץ zip אחד ויחיד מתיקיית הפרויקט ולהעלות ללשונית הגשת הפרויקט באתר הקורס.
* מועד הגשה: 15/3/2024.

# בהצלחה!

# 

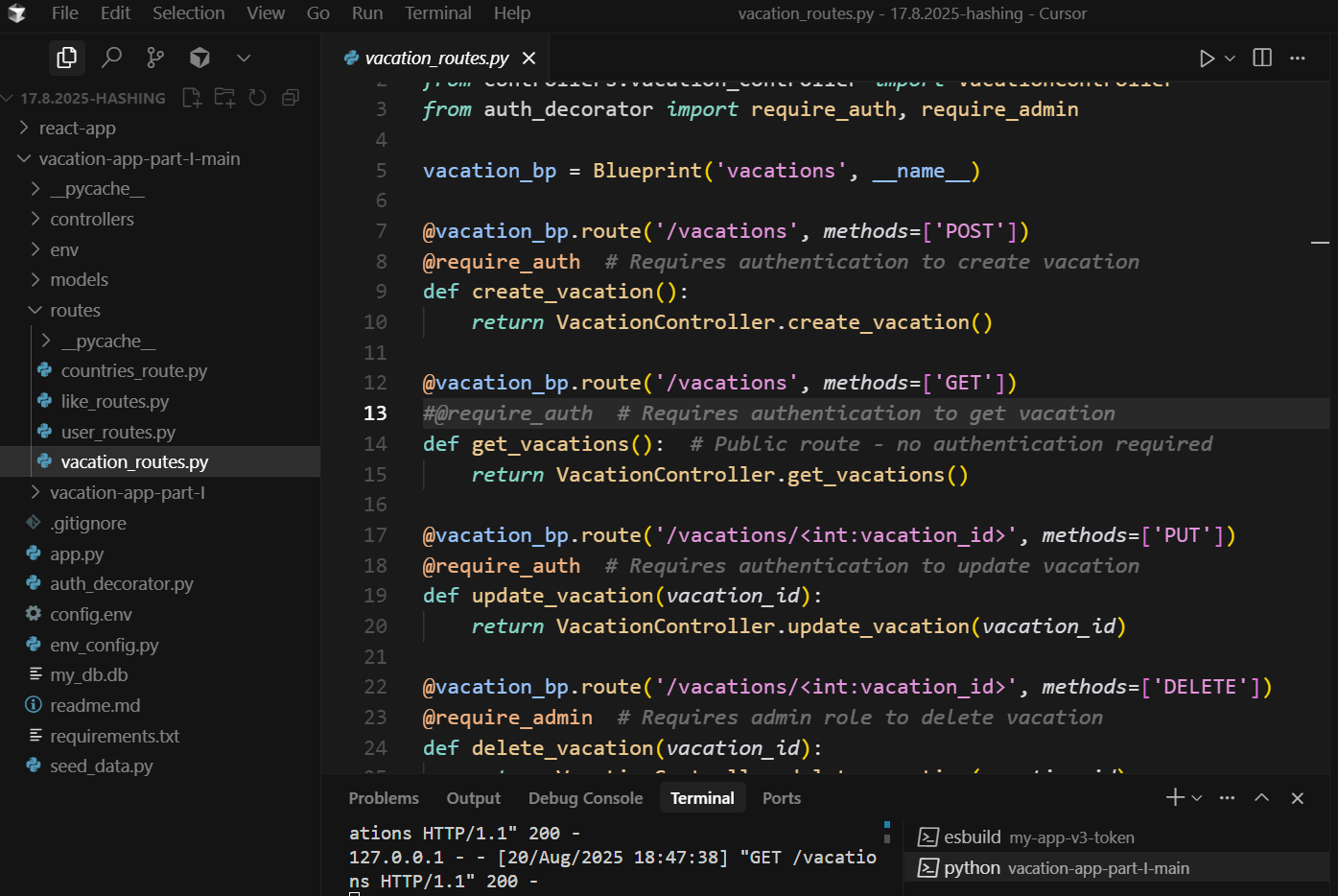
# 

# לפני שמתחילים

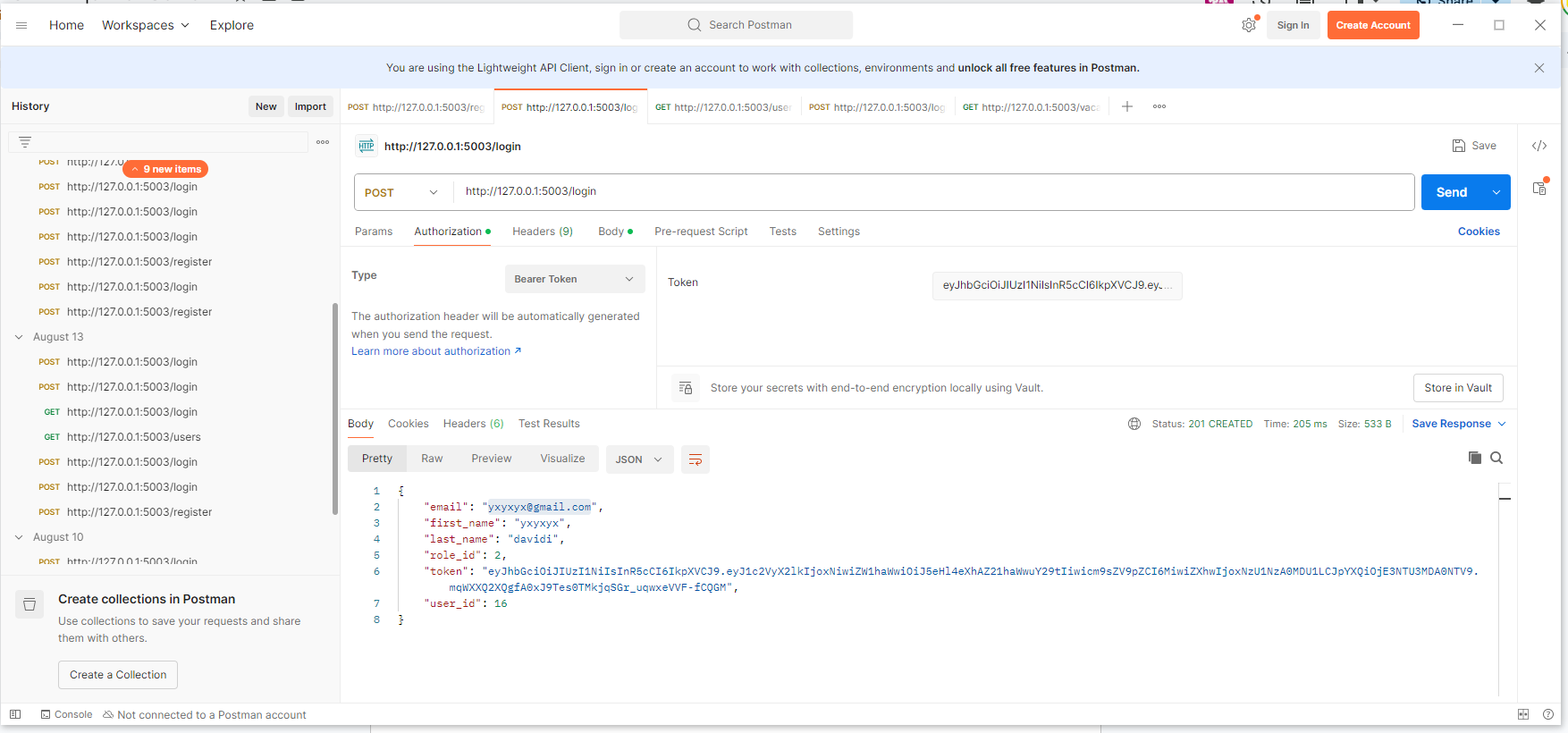
פתחו גיט עבור **הקליינט** של הפרויקט ועבדו על גיט מסודר . וודאו שיש לכם גיט אחר עבור צד השרת (אמור להיות לכם מהגשת פרויקט 1 עבור צד שרת )

סדר פעולות יצירת הפרויקט

1. טיפול בצד שרת מוקלט בתאריך 17/8
2. לא לשכוח בצד שרת להתקין cors
   1. הוספת יכולת הצפנת סיסמא בקונטרולר של user
   2. טיפול בפונקצית login - החזרת token
   3. כתיבת auth\_decorator.py
   4. חסימת routing באמצעות auth\_decorator.py



* 1. בדיקה שהכל עובד מושלם עם postman כולל שליחת autherization



טיפול בקליינט

1. פתיחת פרויקט חדש
2. פתיחת תיקיות contexts, component, page, api
3. פתיחת כל הקומפוננטות הדרושות ריקות
4. התקנת react-router-dom
5. בניית מעטפת כולל navbar שמנווט בין הקומפוננטות
6. יצירת context user
7. מתחילים קומפוננטת register
8. קומפוננטת login
9. בדיקות שהכל עובד
10. ביצוע crud עבור vacation

אתגרים למי שרוצה,

1. טעינת הפרויקט ל gpt ושאלה כגון תן לי features נוספים שאני יכול לבצע בפרויקט
2. דוגמא ל feature : שליחת בקשה ל agent
   1. נתחיל ממחקר gpt או gemini או אחר
   2. נבקש מ gpt כיצד אני יכול לשלוח מהאתר שלי בקשות אלי. כלומר לשלוח ל gpt req ולקבל תשובה לקליינט בריאכט ולהציג את התשובה למשתמש
3. שילוב מפה בפרויקט. רעיונות למסלולים במדינה ליד מיקום החופשה
4. הצגת מטבע המדינה באמצעות api agent