

# *DreamRizz*

## Architectural Design

Software Engineering Course

מגישים: לירון בכשי, מתן כהן

ועדי אוחיון

מתרגל: אמיר לויט



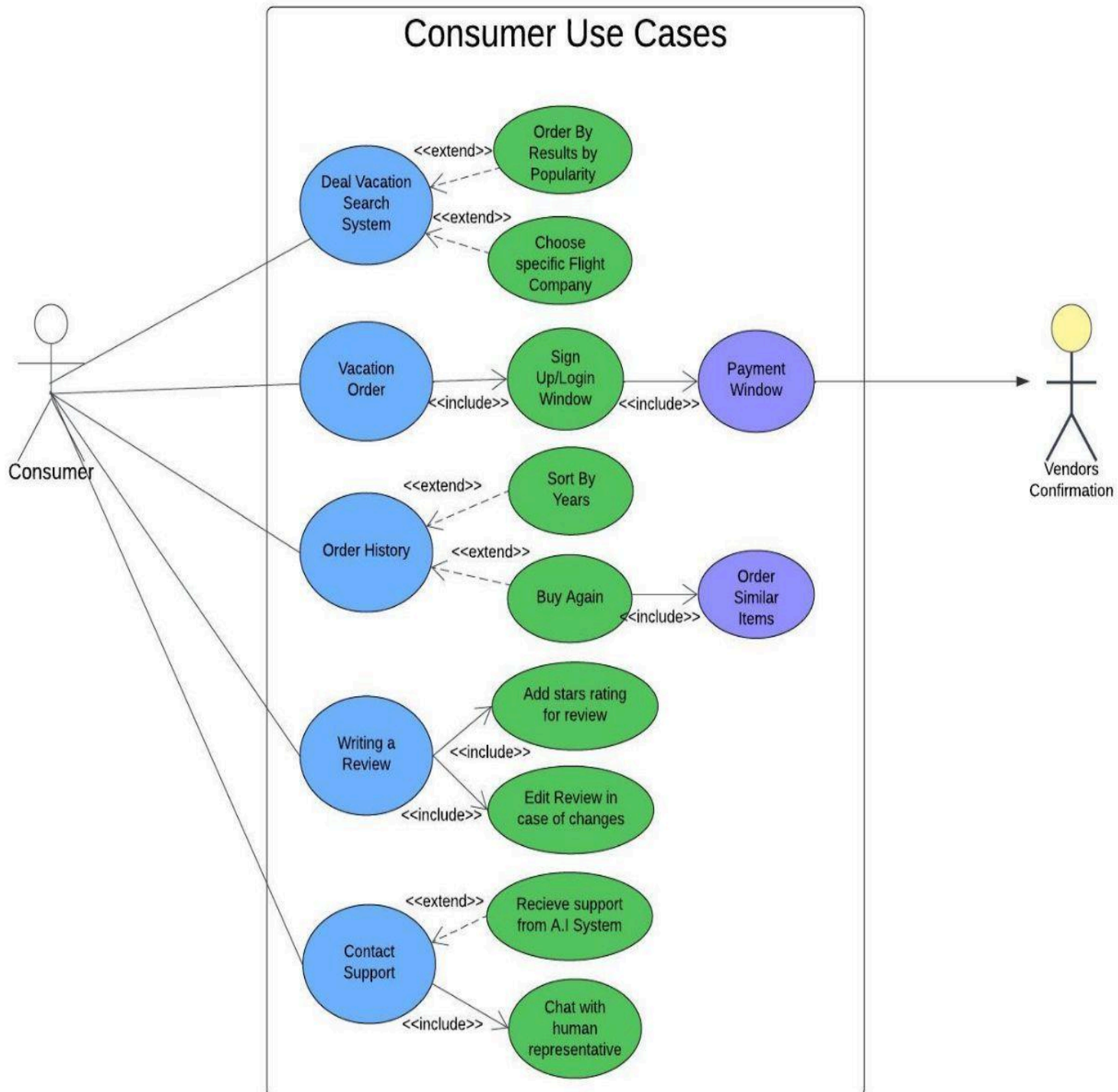
### תוצר 1: Use Case Model -

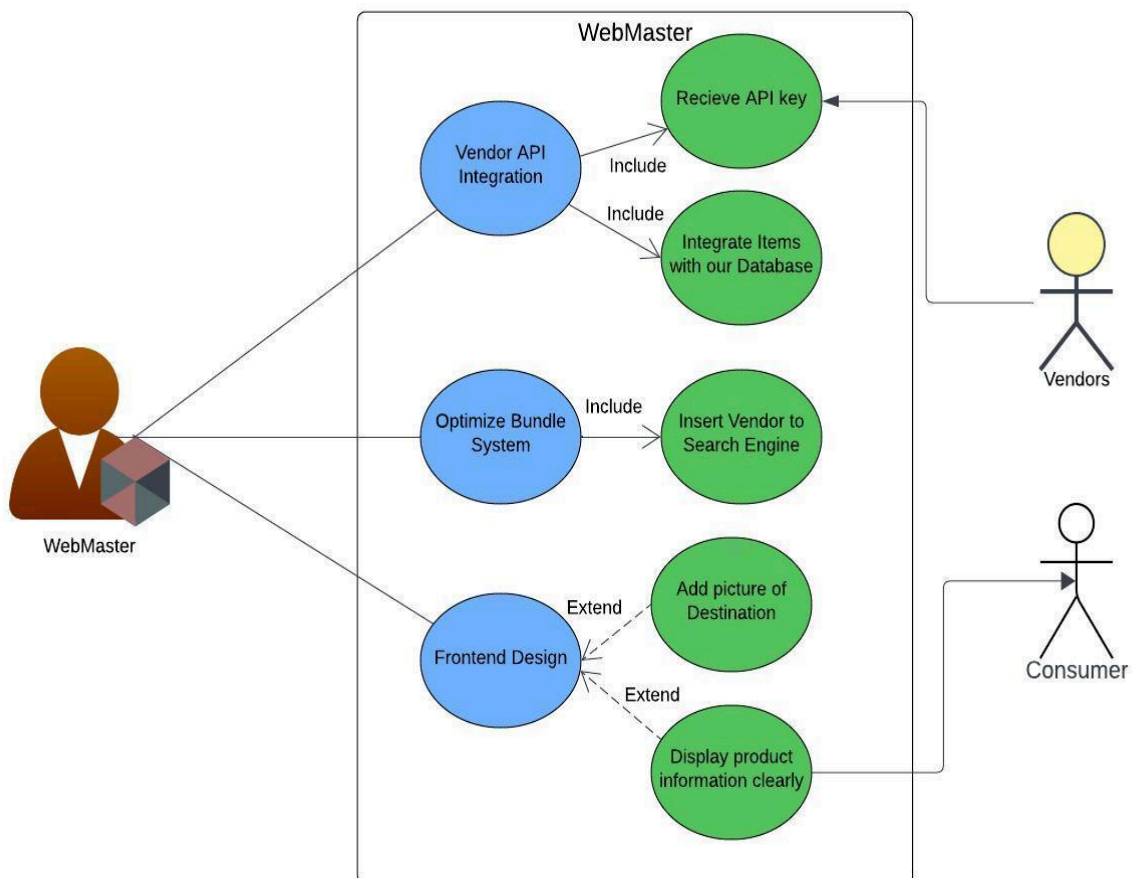
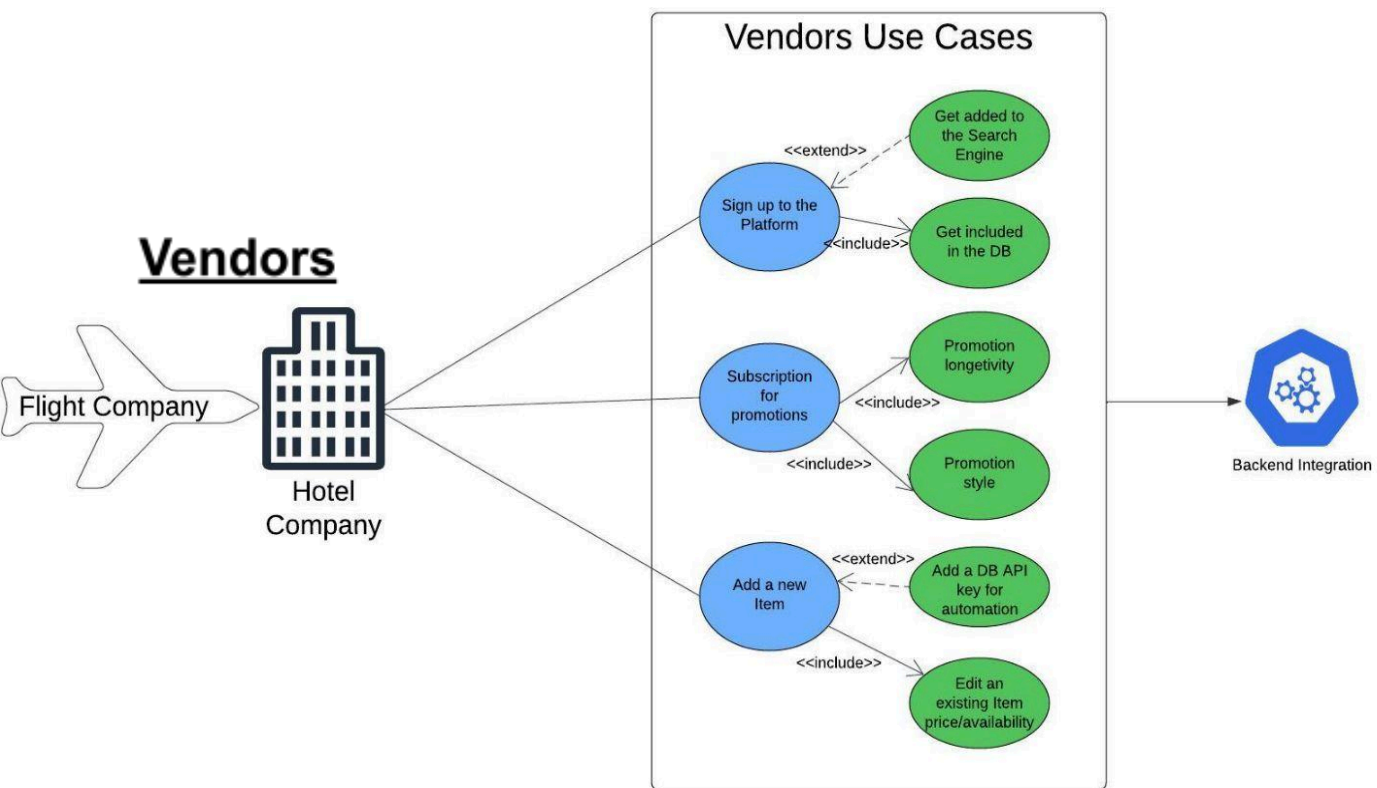
- בחרנו לפרט ב-3 דיאגרמות Use Cases את סוגי המשתמשים והיכולות שלהם:

- 1) Consumer Use Case
- 2) Vendor Use Case
- 3) Webmaster Use Case

כאשר Consumer הם הצרכנים שרוכשים חבילות נופש מן האתר,  
Vendor הם הספקים של האתר שלנו, כלומר בתי מלונות וחברות הטיסה, ומנהלי  
האתר אחראים לתפעול ה-Backend וה-Frontend.

# Use Case Diagrams





## תוצר 2: Sequence Diagrams -

- בחרנו לתאר את שלושת דיאגרמות ה-Use Cases לתוך 2 דיאגרמות Sequence:

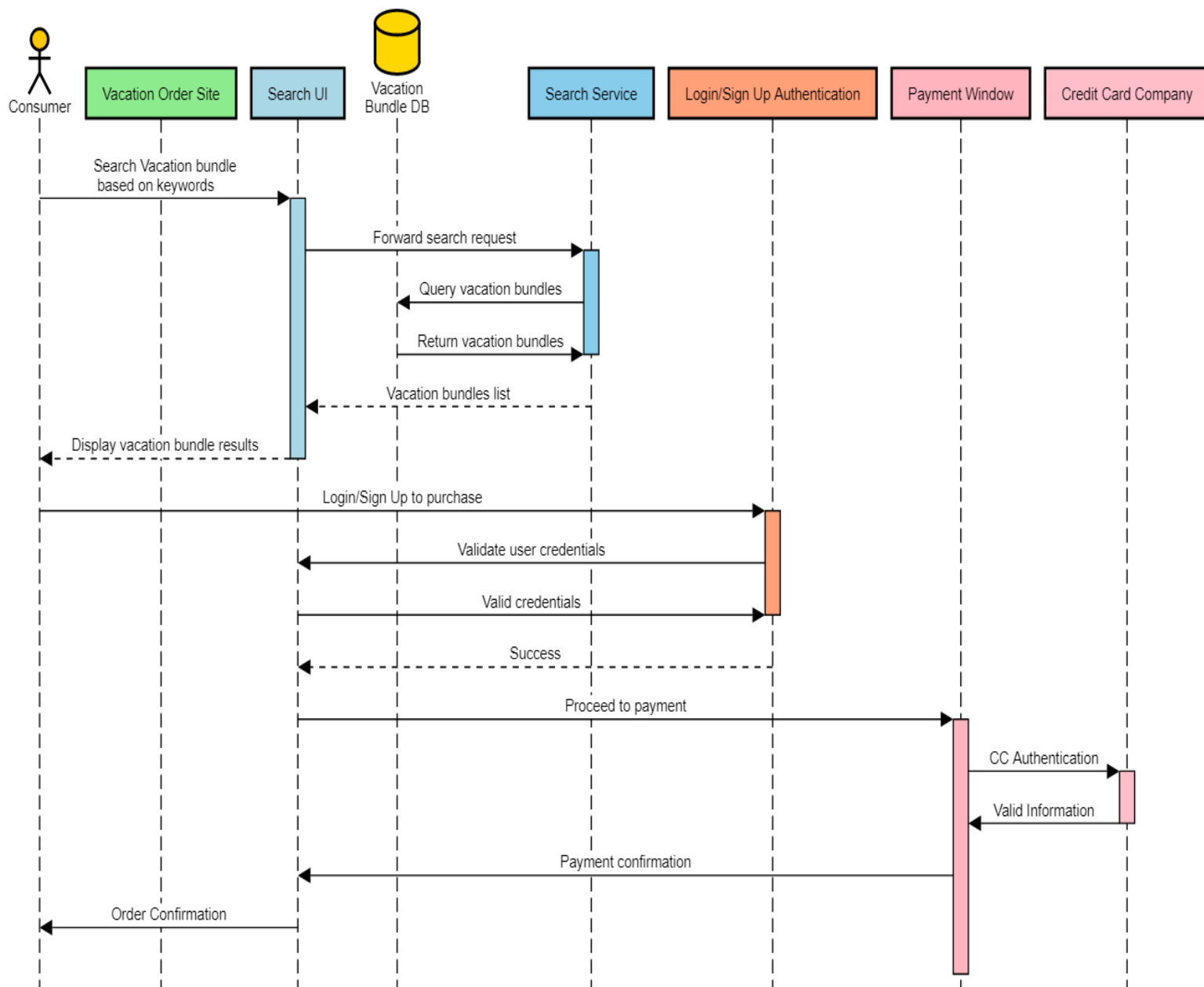
1) Consumer Sequence Diagram

2) Vendor + Webmaster Sequence Diagram

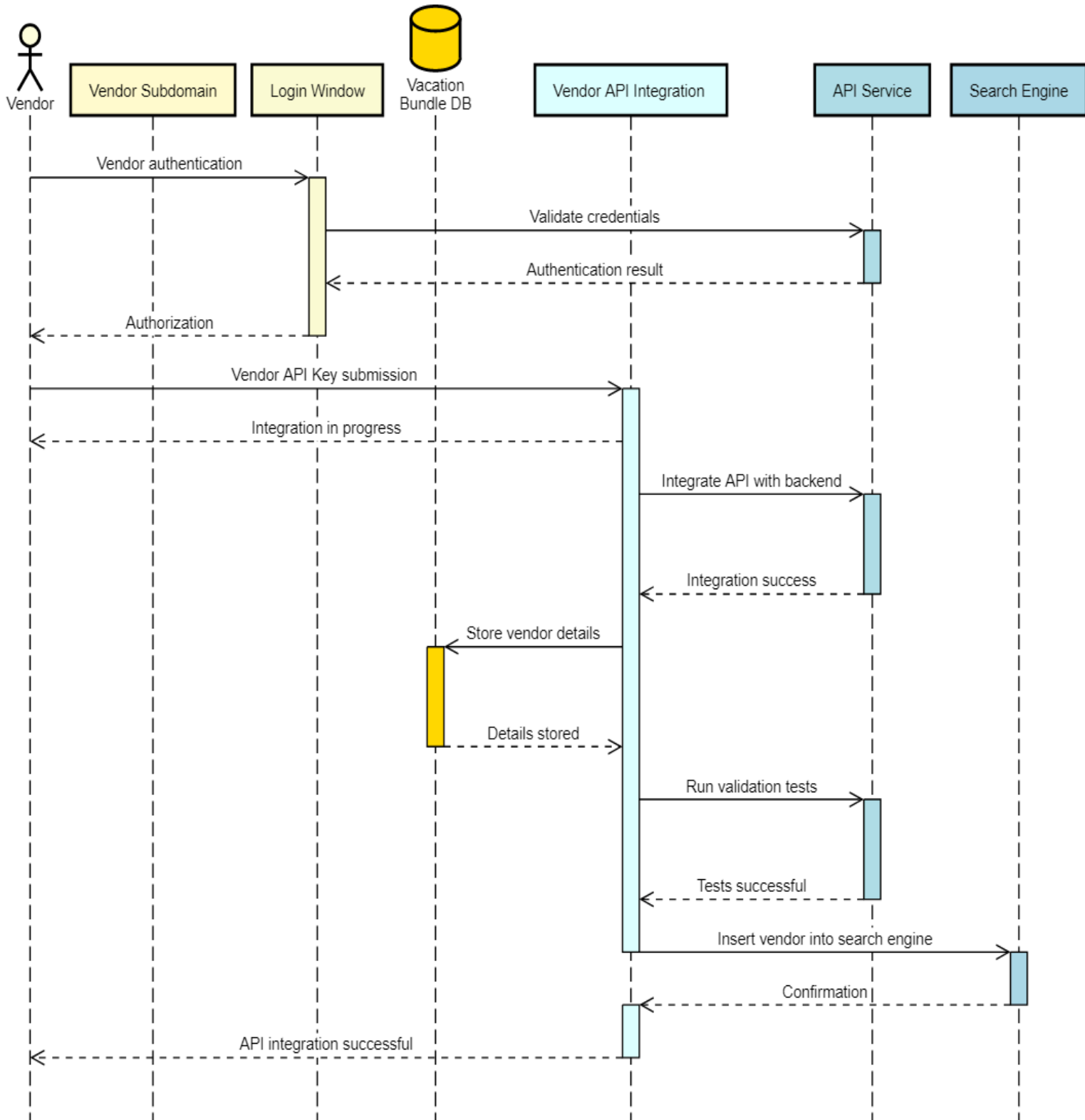
**דיאגרמת הצרכן מתארת את התהליך שלו באתר, ודיאגרמת הספק + מנהל האתר צוללת יותר לעומק של איך האתר עובד מאחורי הקלעים (Backend).**

**הערה:** גוון צבע דומה בין רכיבים מתאר מערכת יחסים קרובה ביניהם

Consumer Sequence Diagram - DreamRizz



## Vendor and Webmaster Sequence Diagram - DreamRizz



## תוצר 3: Class Diagram -

בשלב זה בחרנו לעצב את דיאגרמת המחלקות לפי עיצוב ארכיטקטוני של MVC:

Model: **Vacation Bundle DB**

View: **Website UI**

Controller: **שאר הרכיבים המתקשרים בין המודל והתצוגה**

Model: **Vacation Bundle DB**

- המודל הוא מאגר הנתונים של חבילות הנופש. הוא מכיל שתי טבלאות עיקריות: שהות בבית מלון ופרטי טיסה. הנתונים מאוחדים לחבילות נופש באמצעות מפתחות זרים (Foreign Keys).

View: **Website UI**

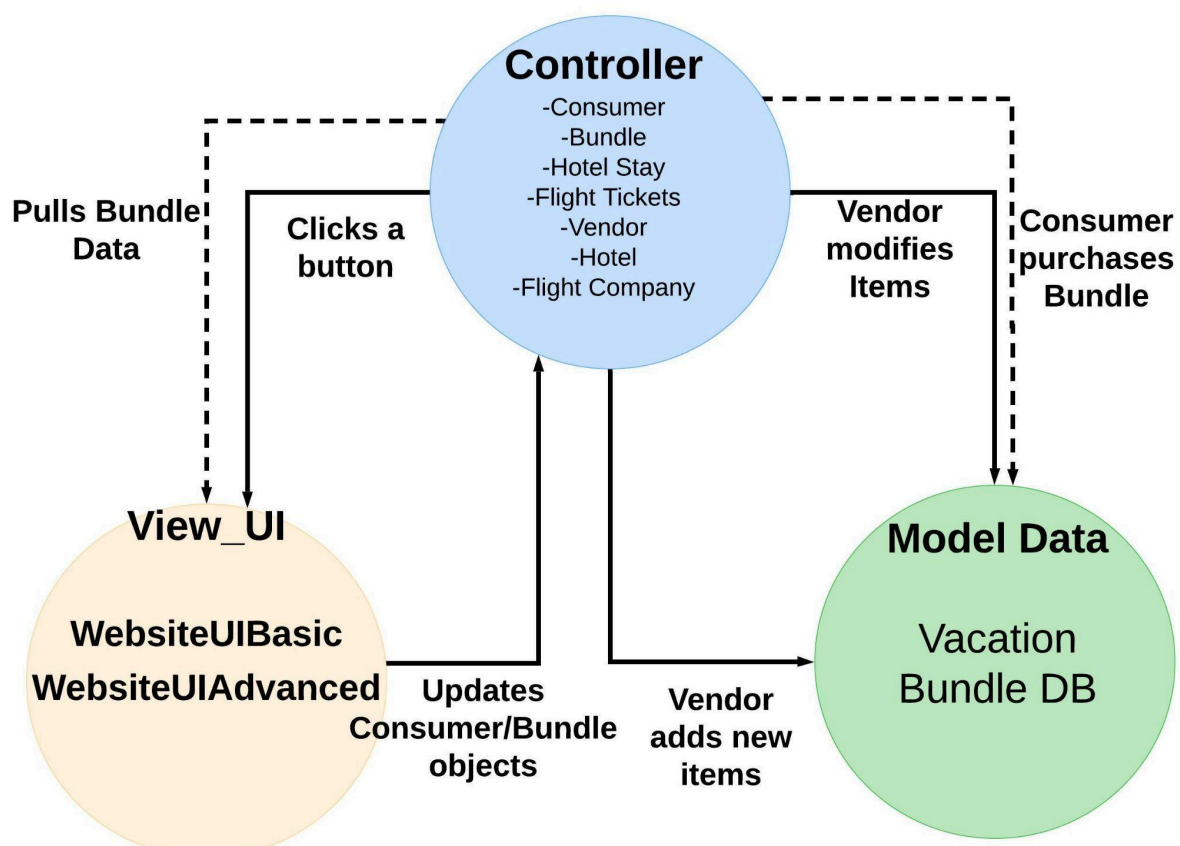
- התצוגה היא ממשק המשתמש של האתר, אשר מציג את חבילות הנופש ללקוח ומאפשר לו לבצע רכישה, התצוגה מחולקת ל-2 מחלקות - מחלקת פיצ'רים בסיסים, ופיצ'רים מתקדמים.

Controller: **Consumer and Bundle**

- אובייקט צרכן אשר רוכש אובייקט באנדל ממאגר הנתונים, המכיל כרטיסי טיסה ובית מלון.

Controller: **Vendor**

- אובייקט ספק (חברת תעופה או בית מלון) אשר מוסיף/מסיר פריטים למאגר הנתונים באופן ידני, או לפי API Access.





## DreamRizz Class Diagram

