

## תרגיל בית הדסים 2 חלק ב

### **1. Requirements Review:**

1. אימות וטיפול בשגיאות:

ציין את תגובות השגיאה ואת קודי השגיאה הצפויים עבור תרחישים שונים, כגון שדות חסרים, תאריכים לא חוקיים או שגיאות בצד השרת.

האם יש פרוטוקול לטיפול בשגיאות בבקשות שרת?

אילו הודעות שגיאה מתאימות מסופקות כדי להדריך את המשתמש במקרה של נתונים חסרים או לא חוקיים?

2. נתונים גיאוגרפיים ובחירת תאריך:

האם יש צורך לשקול את משך המחלה או ההדבקה לצורך בידוד קבוצתי?

מה קורה אם מישהו בוחר בתאריך עתידי או לא רלוונטי?

מהו זמן החשיפה לסף להגדרת בידוד?

מהו הרדיוס האופייני לחשיפה לחולה?

האם ממשק המשתמש מספק שיטות אינטואיטיביות לבחירת נתונים ותאריכים גיאוגרפיים

אשר אם השילוב עם Google Maps מאפשר למשתמשים לבחור מיקומים ישירות על המפה?

3. חלון קופץ לאישור:

הגדר את אופן הפעולה של חלון האישור שמופיע כאשר לא נבחר תאריך או מיקום.

ודא שההודעה הקופצת היא אינפורמטיבית ומנחה את המשתמש לנקוט בפעולה מתקנת.

4. ניהול בידוד:

האם יודיעו למשתמשים על מיקומי החשיפה שלהם?

כיצד יקבלו הודעה למשתמשים על הכללה בבידוד קבוצתי?

לאחר כמה זמן אנשים יכולים לצאת מהבידוד הקבוצתי? בהתבסס על זמן או תוצאות בדיקות?

5. מקרי קצה:

איך המערכת מטפלת בערכים כפולים במערך האנשים שיש לעדכן?

איך מתנהלים מקרים שבהם אנשים כבר נמצאים בבידוד או החלימו מהנגיף?

כיצד המערכת מטפלת במקרי הקצה הללו מבלי לפגוע בפונקציונליות?

6. שיקולי אבטחה:

איך המערכת מוודאה שמידע רגיש, כגון מידע אישי, מטופל בצורה מאובטחת ומוגן מפני גישה לא מורשית?

7. אסטרטגיית בדיקה:

מהי גישת הבדיקה, כולל בדיקות יחידה, בדיקות אינטגרציה ובדיקות מקצה לקצה, כדי לאמת את הפונקציונליות והאמינות של התכונה?

האם המערכת יכולה לטפל במספר רב של בקשות ביעילות?

כיצד המערכת מטפלת בתאריכים עתידיים או עברו בבקשות?

האם המערכת מסוגלת להתמודד עם בידודי קבוצות חופפים?

כיצד המערכת מטפלת בשגיאות בבקשות שרת או בבעיות קישוריות לרשת?

8. הגדרות כלליות:

מה נחשב בקשה תקפה?

אילו סיכונים פוטנציאליים טומנת בחובה המערכת, וכיצד יפחיתו אותם?

מיהם המשתמשים במערכת?

למי תהיה גישה למערכת?

האם יהיו כמה סוגים של גישה למערכת?

## **2. מסמך בדיקות:**

1. בדיקת ממשק משתמש:

- אינטראקציה במפה:
- ודא שרכיב מפות Google מוצג כהלכה בממשק המשתמש.
- בדיקת אינטראקציה עם המפה, כולל התקרבות/התרחקות.
- ודא שהמפה משקפת במדויק את המיקום הגיאוגרפי שנבחר.

2. בחירת תאריך:

- בדוק את הפונקציונליות של בוחר התאריכים לבחירת תאריך החשיפה לנגיף ותאריך ההחלמה.
- ודא שפורמט התאריך עקבי ותואם לפורמט הצפוי.
- תרחישי בדיקה הכוללים טווחי תאריכים שונים ומקרי קצה שונים (למשל, בחירת תאריכים עתידיים, בחירת תאריכים קודמים).

3. לחצן יצירת בידוד קבוצה:

- בדוק את ההתנהגות של כפתור "צור בידוד קבוצה".
- ודא שלחיצה על הכפתור מפעילה את הגשת בקשת הבידוד.
- בדוק תרחישים שבהם לא נבחרו תאריכים או מיקומים כדי להבטיח טיפול נכון בשגיאות.

4. חלון קופץ לאישור:

- אמת את התנהגות חלון האישור כאשר לא נבחר תאריך או מיקום.
- אשר שהודעות שגיאה מתאימות מוצגות למשתמש.

5. בדיקות בצד השרת:

- טיפול בבקשות:
- בדוק את נקודת הקצה בצד השרת לקבלת בקשות בידוד.
- ודא שהשרת מנתח ומטפל כראוי במטען הבקשה.

- בדוק תרחישים עם שדות חסרים או נתונים לא חוקיים כדי להבטיח טיפול נכון בשגיאות.

- הבטח סנכרון עם תוצאות בדיקות COVID19 ועדכוני מסד נתונים.

6. אינטראקציה עם מסד נתונים:

- ודא שהשרת מעדכן את מסד הנתונים בפרטי הבידוד.

- מקרי בדיקה בהם יש צורך לבדוד ולעדכן מספר אנשים במסד הנתונים בו זמנית.

- ודא שרשומות מסד הנתונים מדויקות ועוקבות עם פרמטרי הבקשה.

7. טיפול בשגיאות:

- בדוק תרחישי שגיאה כגון כשלים בחיבור מסד נתונים, בקשות שגויות או שגיאות שרת.

- אשר שקודי מצב HTTP והודעות שגיאה מתאימים מוחזרים בתגובה לתרחישי שגיאה שונים.

8. מקרי קצה ותרחישים יוצאי דופן:

- תקופות בידוד חופפות:

- בדיקת תרחישים שבהם בקשות בידוד מרובות חופפות בזמן או מערבות את אותם אנשים.

- ודא שהמערכת מטפלת בתקופות בידוד חופפות בחן ללא התנגשויות או חוסר עקביות בנתונים.

9. נתונים גיאוגרפיים לא חוקיים:

- מקרי בדיקה עם נתונים גיאוגרפיים לא חוקיים או מעורפלים במטען הבקשה.

- ודא שהמערכת מטפלת בנתונים גיאוגרפיים לא חוקיים כראוי ומספקת הודעות שגיאה משמעותיות.

10. תאריך התאוששות לפני תאריך החשיפה:

- תרחישי בדיקה שבהם תאריך ההחלמה הוא לפני תאריך החשיפה.

- ודא שהמערכת מזהה ודוחה שילובי תאריכים לא חוקיים.

11. בדיקות אינטגרציה:

- שילוב מפות Google:

- ודא את האינטגרציה עם Google Maps API להצגת רכיב המפה.
- תרחישי בדיקה הכוללים עיבוד מפות, קידוד גיאוגרפי ומיקום סמנים.

12. שילוב API חיצוני:

- בדוק אינטגרציה עם ה-API של סמלי הטלפון כדי לאחזר מידע על אנשים חשופים.
- ודא את הדיוק והאמינות של הנתונים שהוחזרו מה-API החיצוני.

13. בדיקות אבטחה:

- אימות קלט:

- בדוק מנגנוני אימות קלט למניעת הזרקת SQL, התקפות XSS ופגיעות אבטחה אחרות.

- ודא שהמערכת מחטאת ומאמתת קלט משתמש לפני העיבוד.

14. אימות והרשאה:

- בדיקת מנגנוני אימות והרשאה לגישה לנקודת הקצה בצד השרת.
- ודא שרק משתמשים מורשים יכולים להגיש בקשות בידוד ולגשת לנתונים רגישים.

15. בדיקות קבלה:

- ודא שכל הדרישות המפורטות ב-RR-מתקיימות.
- בדיקת עמידה בתקני איכות וציפיות המשתמש.
- ודא שהתכונה פועלת כצפוי בתרחישים בעולם האמיתי.