**תרגיל בית 3 - חלק ב - מחשוב ענן**

**חלק קבוצתי:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Reason To Fail** | **Pass/Fail** | **Comments** | **Precondition** | **Expected Result** | **Description** | **Test Name** | **#** |
| Build a welcome page to give a background for the user | Not relevant | The user successfully directed | Login page exists | User directed to login page | User click on log in button | directedToLoginFromWelcomePage | 1 |
| Build signup page to let new user play our game | Not relevant | The user successfully directed | signup page exists | User directed to signup page | User click on sign up button | directedToSignUpPageFromWelcomePage | 2 |
| - | Pass | The user is logged in | User account exists | User directed to home page | User login with valid details | loginSuccess | 3 |
| - | Pass | User remains on the login page | User account does not exist | User receives an error message | User login with invalid details | loginFail | 4 |
| Build signup page to let new user play our game | Not relevant | User successfully create an account | User enter username that does not exists in the db | User receives a success message and directs to login page | User enter username and password | SignupSuccess | 5 |
| Build signup page to let new user play our game | Not relevant | User did not create an account | User enter username that exists in the db | User receives an error message | User enter username and password | SignupFail | 6 |
| - | Pass | User can see feedback for his answer | - The question is available in the system for user to answer  - The true answer is saved in the db | The system shows positive feedback | User select a correct answer | UserAnswerIsCorrect | 7 |
| - | Pass | User can see feedback for his answer | The question is available in the system for users to answer | The system shows a negative  feedback | User select an incorrect answer | UserAnswerIsWrong | 8 |
| In the system we directed to the Login page,  It will be nice for user to see his results even if he finishes his game in the middle | Not relevant | The user sees his result for that game | The user started to play | User directed to result page | User click on end game button | UserEndGame | 9 |
| - | Pass | User add question | The user has admin permission | The question successfully added to the db | User enter all details and click on add question button | AddQuestionSuccess | 10 |
| There is no input check | Fail | User did not add question | The user has admin permission | User receives an error message | User enter the fields, but not all of them and click on add question button | AddQuestionFail | 11 |
| After editing a question, the question remains the same | Fail | User successfully edit a question | The question's id exists in the db | The question changes in db respectively | User choose question from the dropdown, click on search button and add all fields | EditQuestionSuccess | 12 |
| There is no input check | Fail | User did not edit a question | The question's id exists in the db | User receives an error message | User choose question from the dropdown, click on search button but not fill all fields | EditQuestionFail | 13 |
| Add that option to allow more flexibility on the admin part | Not relevant | User successfully delete a question | The question's id exists in the db | The question deleted from the db | User choose question from the dropdown, click on search button and add all fields | DeleteQuestionSuccess | 14 |
| - | Pass | The user successfully directed | game page exists | User directed to game page | User press on the Start button | StartAGameSuccess | 15 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Reason To Fail** | **Pass/Fail** | **Comments** | **Precondition** | **Expected Result** | **Description** | **Test Name** | **#** |
| Add badges to create more motivation for the user | Not relevant | The user successfully see his badge | - User exits in db  - User's score in top 3 | User see a matched badge in his home page | User log in | BadgeDisplaySuccess | 1 |
| - | Not relevant | The user doesn't see badge | User exits in db  - User's score NOT in top 3 | User does not see badge in his home page | User log in | BadgeDoesNotdisplaySuccess | 2 |
| Add option for user to view videos to help him learn the subject well and keep play | Not relevant | The user successfully directed to Cinema page | The user on home page | User directed to Cinema page | User click on Cinema button | EntarToCinema | 3 |
| - | Not relevant | User successfully see an explanatory video | - User logged in  - The chosen video exist in db | New tab with the chosen video is start to play | User click on watch button | WatchAnExplanatoryVideo | 4 |
| **There is leader table** it's just not in the home page | Fail | The user successfully directed to Leader Table page | The user on home page | User directed to  Leader  Table page | User click on Leader Table button | EntarToLeaderTable | 5 |
| - | Pass | User can see his rank on the leader table | User logged in | User appears on the leader table | User click on Leader Table button | UserAppearsOnTheLeaderTable | 6 |
| - | Pass | User can see his rank on the top 5 at the leader table | User's score in top 5 | User appears on the top 5 leader table | User click on Leader Table button | UserAppearsOnTop5LeaderTable | 7 |
| - | Pass | User session details saved successfully | User start a session game | User session number and score saved on the db | User click on End Game button | UserSessionSaveOnDB | 8 |
| There is no indication in the code that any question ID is stored in the database | Fail | Question's id saved successfully | User start a session game | Question's id save in user details on the db | User click on the correct answer | UserQusetionThatBeenAnsweredCorrectSaveOnDB | 9 |
| Add that feature to create more motivation for the user | Not relevant | The number of questions that answered in a row is successfully display | User answered X questions in a row | The number of questions that answered in a row is X | User click on End Game button | DisplayQuestionThatAnsweredInARow | 10 |
| Add that feature to allow more flexibility to the user | Not relevant | The user successfully plays as he chose | User logged in | User play the amount of questions he chose | User choose the amount of questions for that session and click start button | StartASession | 11 |
| - | Pass | The user successfully receives a message | - The user has admin permission  - The user fills all the required fields | User receives a pop-up message | Admin add/delete/edit question | Admin SuccessfullyAdd/Delete/EditQuestion | 12 |
| **This feature exists** in the home page (just not after a game finished) | Fail | The user successfully sees details on his sessions score | User logged in | User directed to result page and see the bar chart | User click on End Game button | BarChartDisplay | 13 |
| - | Not relevant | The user add video | The user has admin permission | The video successfully added to the db | User enter all details and click on add video button | AddVideoSuccess | 14 |
| - | Not relevant | The user did not add video | The user has admin permission | User receives an error message | User enter the fields, but not all of them and click on add video button | AddVideoFail | 15 |

**חלק אישי:**

**שם הסטודנטית: שירן**

|  |  |
| --- | --- |
| **קריטריון** | **הסבר** |
| מימוש | לא תמיד, לדוגמא, כשעורכים שאלה אז היא לא מחליפה את השאלה שערכו אלא היא מתווספת כשאלה חדשה. |
| יעילות | הקוד יעיל, וכן המערכת סך הכל עובדת טוב. |
| פשטות | נראה כי הקוד כתוב בצורה ברורה, כמו כן הוא מחולק לחלקים מרכזיים עם תיעוד מתאים. |
| מודולריות | כן, יש חלוקה טובה של פונקציות. |
| תבניות עיצוב | קיים שימוש בסינגלטון, כאשר קוראים למסד הנתונים פעם אחת. |
| OOP | אין שימוש ב-OOP |
| באגים וטעויות | ניתן להכניס שאלה גם שלא כל השדות מלאים. |
| טיפול בשגיאות | כמו שציינתי לעיל, ניתן להכניס שאלות גם אם לא כל השדות מלאים, אין בדיקת קלט והודעת שגיאה למשתמש.  הבדיקה והדיבוג נוחים, וכן ההודעות שמוצגות למשתמש ברורות. |
| בדיקות | כן, אין בדיקות קלט לחלק מהמקרים. כמו שציינתי בהוספת/עדכון שאלה. היה צריך לבדוק את תקינות השדות שם.  בנוסף גם בהתחברות לא תמיד זה מציג הודעת שגיאה כאשר מכניסים נתונים לא תקינים. לפעמים המערכת עצמה (הגוגל קולאב) זורקת שגיאה. |
| שימושיות | מצד אחד כן, הממשק שימושי. הוא נראה טוב ובצבעים נוחים לעין. ולכן גם עושה רצון להמשיך לשחק בו.  מצד שני, אין כפתורי 'help' שיחלו אולי לעזור למשתמש להבין מה האפשרויות שלו לעשות במשחק. |
| תיעוד | הקוד עצמו כמו שצוין אכן מתועד.  לעומת זאת, כמו שציינתי קודם אין הסבר בממשק עצמו מה כל כפתור עושה, או כמה נקודות מרוויחים על שאלה וכו'. |
| אתיקה | לא מפרים שום פרטיות וכן אין שום ניצול או חולשה אנשות. בנוסף אין משהו כנגד קבוצה מסוימת. |
| אבטחה | לא, אין (הסיסמא בדף ההתחברות נסתרת). |
| ביצועים | ייתכן. אם זאת נראה כי הקוד מסודר ולכן ניתן יהיה להוסיף פי'צרים גם מבלי לפגוע בביצועים. |
| ביצועים | מבחינת מהירות המערכת אז הקוד עובד מהיר, ולא נתקע במעבר בין דפים. ולכן נראה כי אין מה לשפר בפן הזה.  אם זאת יש כמה תקלות במערכת שציינתי שאותם צריך לתקן. |
| קריאות - readability | קוד מאוד מסודר ומובן. |
| סעיף פתוח - התאמת במערכת לגודלי מסך שונים | ניתן להשתמש בגודלי מסך שונים, אם זאת במסך של המשחק עצמו איפה שמופיעות השאלות אז זה לא מתאים, לא ממש נוח. |
| סעיף פתוח - התאמת מערכת לכלל האוכלוסיה | המערכת אכן מתאימה לכלל האוכלוסיה, כלומר לא צריך לדעת רקע מסוים בתוכנה או רקע טכנולוגי כלשהו כדי לדעת להפעיל את המערכת. |

שאלות פתוחות:

1. היה מעניין לראות את המערכת של קבוצה נוספת. אם זאת מעבר על הקוד עצמו לא הרגיש רלוונטי והכרחי, בטח לא לקורס הזה.
2. כן. לא לתת להריץ את הבדיקות שקבוצה אחת כתבה על מערכת של קבוצה אחרת, מכיוון שנוצר מצב שיש המון בדיקות לא רלוונטיות. עדיף לקבל מהקבוצה הנבדקת את הבדיקות שלהן, ולתת סעיף של מתן שיפור למערכת במידה ויש.
3. הערות נוספת – אין.

**שם הסטודנט: דניאל גרוס**

|  |  |
| --- | --- |
| **קריטריון** | **הסבר** |
| מימוש | האם הקוד מבצע את הנדרש?  חישוב הנקודות אחרי כל משחק לא מתבצע כצפוי. ברגע שלוחצים על כפתור החזרה למסך הבית ומשם מתחילים משחק חדש המערכת ממשיכה לספור נקודות ושאלות על המשחק שהסתיים מקודם, ולא מתאפס אחרי כל משחק |
| יעילות | האם ניתן לשפר את יעילות הקוד?  הקוד יעיל וספציפית בשליפת נתונים מהמסד נתונים |
| פשטות | האם ניתן לפשט את הקוד?  כפי שאני רואה הקוד כתוב בצורה מסודרת ופשוטה מחולק לפונקציות קטנות  ויש תיעוד כמו שצריך |
| מודולריות | האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)  הקוד מחלק את הפונקציות והמחלקות למודולים שונים ומפריד את הפונקציות הקשורות לתחום מסוים או למשימה מסוימת לתוך מודולים נפרדים |
| תבניות עיצוב | האם ישנו מימוש ברור ונכון של תבניות עיצוב?  תבנית ה-:Singleton השימוש בתבנית ה-Singleton נמצא באתחול של ספריית Firebase Admin SDK בקוד. המודול firebase\_admin מאותחל רק פעם אחת |
| OOP | האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?  בקוד שסופק יש מספר עקרונות של תכנות מונחה עצמים OOP:  1. עקרון הייחודיות של האובייקטים Unique Objects: בשימוש בתבנית ה- Singleton ניתן לראות את העיקרון של הייחודיות של האובייקטים. הקוד מאפשר יצירת מופע יחיד מסוג מסוים, והוא נשמר ומשמש באופן יחיד בכל המקומות השונים בקוד. |
| באגים וטעויות | האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?  כן, פעולות הוספת ועריכת שאלות לא עובדת כצפוי |
| טיפול בשגיאות | האם ישנה התראה על שגיאות משתמש?האם ישנו שימוש במנגנון תפיסת שגיאות?האם הבדיקה ודיבוג הקוד נוחים?האם הודעות המשתמש ברורות וידידותיות למשתמש?  שדות log in ריקים לא מראה התראה, אין שימוש במנגנון תפיסת שגיאות בהתחברות, לא מופיע הודעת שגיאה כאשר השם משתמש לא קיים במסד נתונים |
| בדיקות | האם יש בדיקות שניתן להוסיף?  כן, כאשר לוחצים על כפטור ה log in ואין קלט זה לא עובד  לכן צריך לבדוק אם זה ריק ואם כן להציג הודעה |
| שימושיות | האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?  אישית לא הרגשתי שהממשק כל כך שימושי, אחת הסיבות היא כאשר מתחילים משחק לא יודעם מתי המשחק מסתיים למרות שזה מסתיים אחרי 5 שאלות אז אין אופציה בלהמשיך ולשחק עוד  ולא ניתן לראות כמה נקודות מקבלים עבור כל שאלה לדעתי זה היה יכול לתת מוטיבציה לנסות כמה שאפשר לענות נכון על השאלות.  מצד אחר יש הרגשה שכן רציתי להמשיך ולשחק כי השאלות לא היו כל כך קשות ולא קלות מדי לכן היה אתגר בזמן המשחק וזה נתן מוטיבציה להמשיך |
| תיעוד | האם הממשק מתועד?  כאשר מתחברים יש כמה כפתורים אחד מהם view וזה מציג את הגרף וזה לא היה ברור מה הכפתור הזה עושה |
| אתיקה | האם יש שימוש בנתוני משתמש אשר מפרים פרטיותו?האם יש ניצול של התנהגות או חולשה אנושית?האם ישנו algorithmic bias כנגד קבוצה או פרטים מסוימים?  לא, אין שימוש בנתוני משתמש אשר מפרים פרטיותו, אין ניצול של התנהגות או חולשה אנושית, אין algorithmic bias כנגד קבוצה או פרטים מסוימים |
| אבטחה | האם ישנו מידע אבטחה (שמות משתמש, ססמאות) גלויים?  בהתחברות הסיסמה תופיע נקודות ולא גלויה |
| ביצועים | האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?  כן, אם יבוצע שינוי באלגוריתם הקיים, יתכן שהוא יהיה פחות יעיל מהאלגוריתם הקודם. זה עשוי לגרום להאטה בביצוע הפעולות או להגברת הטעויות בקוד. |
| ביצועים | האם ניתן לשפר לדעתך את הביצועים?  בטח שניתן לשפר, וכבר רשמתי כמה דברים שלא עובדים כצפוי במערכת אז אפשר להתחיל משם. |
| קריאות - readability | האם הקוד מובן בקלות?אילו חלקים בקוד היו לא ברורים עבורך?כיצד ניתן לשפר את קריאות הקוד?  הקוד מובן בקלות  אבל ישנם מקומות בקוד שבהם שמות המשתנים לא מספקים מידע מספיק להבנה מידית של מה הם מייצגים. בשיפור קריאת הקוד, מומלץ להשתמש בשמות משתנים מובנים יותר |
| שליטה | האם יש למשתמש שליטה במערכת? אני חושב שכן יש שליטה כי בכל הדפים יש כפתורי חזרה או סיום משחק |
| קל ללמידה | האם קל למשתמשים לבצע את הפעולות הרצויות במערכת?  כן, המערכת פשוטה ולא עמוסה כך שניתן ללמוד אותה בקלות ולא גורשת זכרון חזק |

**שאלות פתוחות:**

1. כיצד הרגשת במהלך החוויה של מעבר על הקוד?

אישית, כאדם שעובר על הקוד, הייתי מרגיש כי הקוד נראה מאורגן ומובן בקלות. המבנה המודולרי שלו מאפשר לי לזהות את החלקים השונים של המערכת ואת הקשרים ביניהם. חלקים מסוימים של הקוד עשויים להיות מורכבים יותר והייתי צריך להתמודד עם קוד שהיה פחות ברור לי מבחינת המטרה והפעולה שלו

2. האם לדעתך ניתן לשפר תרגיל זה?

אני חושב שהתרגיל היה מלמד ונותן אופציה להסתכל על פרויקט דומה לפרויקט שאני והקבוצה שלי עשינו ולראות את הדברים מעיני בודק או משתמש.

וזה נתן לי לראות דרכי פתרון חדשות וגם להבין מה הדברים שחשובים להדגיש בפרויקט הבא

חסרון אחד זה שהמטלה ממש הייתה לקראת סוף סמסטר ויש לחץ של מבחנים הצגות ועוד פרויקטים, הייתה יותר שמח אם זה היה מוקדם יותר

3. הערות נוספות - אין

**שם הסטודנט: ראאד עטאריה**

|  |  |
| --- | --- |
| **קריטריון** | **הסבר** |
| מימוש | כן, הכל עובד |
| יעילות | לוקח יותר מדי זמן לעשות LOGIN וגם מציג הודעה שנעשה LOGIN לא מבין בשביל מה הם מציגים את ההודעה |
| פשטות | הקוד פשוט ומובן |
| מודולריות | יש חלוקה לקבצים, אבל אין מחלקות , הם שמו ה CSS בתיקיה נפרדת |
| תבניות עיצוב | ששאלנו אותם אמרו שלא השתמשו.  אבל מצאתי את התבניות הבאים (הם השתמשו בהם אבל לא ממשו אותם)  למשל:  :Decorator Pattern הקוד משתמש בשיטת " add\_class" כדי להוסיף באופן דינמי מחלקות CSS לווידג'טים |
| OOP | אין בכלל מחלקות בקוד |
| באגים וטעויות | בהדף של עדכון שאילה אין בכלל בדיקות.למשל אם המנהל רוצה לעדכן השאילה ואז הוא מוחק השדה של השאילה עצמה ולחץ על כפתור העדכון אז נתוני השאילה נשמרים בלי השדה של השאילה, וזה גם תקף לכל השדות (תשובה 1, 2 3 , 4) |
| טיפול בשגיאות | אין בכלל, אין התייחסות בכל הקוד לאופציה שבה המשתמש יכול לטעות, למשל לא מלא אחד השדות |
| בדיקות | יש רק בדיקה אם השם משתמש והסיסמה נמצאים |
| שימושיות | כן הוא מסודר והמשתמש יכול להבין איך לעשות הדברים, מאוד ברור, יפה, יש אלמנטים של משחוק |
| תיעוד | כן, יש תיעוד,אבל אפשר להוסיף עוד, לא מתועד כמו שצריך, למשל מעל כל פונקציה לא כתוב מה היא מקבלת, מחזירה ועושה |
| אתיקה | אין שימוש בנתוני משתמש אשר מפרים פרטיותו, אין ניצול של התנהגות או חולשה אנושית, אין algorithmic bias כנגד קבוצה או פרטים מסוימים? |
| אבטחה | לא, אין אבל להגיע אליהם מאוד פשוט, כל בן אדם יכול לשלוף כל הנתונים |
| ביצועים | הקוד מסודר, ואפשר להוסיף עליו דברים |
| ביצועים | הביצועים טובים, רק לוקח יותר די זמן לעשות login |
| קריאות - readability | הקוד מסודר ומובן, אבל היה עדיף להוסיף עוד תיעוד |
| סעיף פתוח | בדיקת אם המנהל מלא כל השדות, אם ב LOGIN המשתמש מלא כל השדות |
| סעיף פתוח | אפשר למשל להוסיף אפשרות שיחזור משאילה לשאילה אחרת באמצע המשחק |

**שאלות פתוחות:**

1. *כיצד הרגשת במהלך החוויה של מעבר על הקוד?*

*הקוד היה מסודר וקריא*

*צריך להוסיף קבצים לה GOOOGLE DRIVE , ושניתי קצת בקוד בשביל שזה יעבוד לא לקח זמן בגלל בקוד מסודר*

1. *האם לדעתך ניתן לשפר תרגיל זה?*

*שאנחנו נבדוק קבוצה שהייתה אייתנו בהצגה ואז נבין יותר מה הם עשו...*

**שם הסטודנטית: אמר דיאב**

|  |  |
| --- | --- |
| **קריטריון** | **הסבר** |
| מימוש | האם הקוד מבצע את הנדרש?  כן, הקוד מבצע את הפעולות הנדרשות עבור משחק השאלות. הוא מציג את ממשק המשחק, כולל השאלות, התשובות האפשריות והכפתורים לבחירת התשובה. גם נערך ספירה של התשובות הנכונות והלא נכונות, ומתעדכן מספר השאלה הנוכחית. כאשר מספר השאלות הנכונות מגיע ל-5, המשחק מסתיים ומציג סיכום עם מספר התשובות הנכונות, התשובות הלא נכונות והנקודות המרוכזות שהשחקן קיבל. מציג גם גרף של הלוח המובילים ומאפשר למשתמש לשחק שוב או לחזור לתפריט הראשי. |
| יעילות | האם ניתן לשפר את יעילות הקוד?  כן ניתן לשפר בכמה דרכים:   * יצירת תמונות: נראה שהקוד יוצר תמונות לכל תוצאה של המשחק ומציג אותן בממשק. יצירת תמונות עשויה לקחת זמן משמעותי ומשאבים. ניתן לשקול ליצור את התמונות מראש ולטעון אותן רק כאשר הן נדרשות להצגה. * אופטימיזציה של לולאות: ישנן לולאות בקוד שמבצעות פעולות על כמה איברים. ניתן לשקול שיפור ביצועים על ידי עשיית שימוש באיטרטורים (iterators) כך שלא יהיה צורך ליצור את רשימת האיברים מראש. |
| פשטות | האם ניתן לפשט את הקוד?   * מיזוג פונקציות: ניתן למזג פונקציות קטנות וחוזרות בתוך פונקציה אחת גדולה יותר. זה יכול לפשט את מבנה הקוד ולהקל על הקריאותיות. * הפקדה של משתנים: במקום ליצור משתנים גלובליים מראש, ניתן להפקיד אותם כפרמטרים לפונקציות. זה יכול להפחית את התלות במשתנים גלובליים ולפשט את תפקוד הפונקציות. |
| מודולריות | האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)   * ניתן לחלק את הקוד למודולים קטנים יותר עם תפקידים מוגבלים. זה יכול לקלקל את הקוד ליחידות נפרדות ולהקל על התחזוקה והשינויים בעתיד. |
| תבניות עיצוב | האם ישנו מימוש ברור ונכון של תבניות עיצוב?  לא |
| OOP | האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?  אין שימוש נכון |
| באגים וטעויות | האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?  כן, כאשר המשתמש מקבל נקודות עבור ביצועים שלו במשחק, הקוד מחשב את הניקוד בצורה לא תקינה. |
| טיפול בשגיאות | האם ישנה התראה על שגיאות משתמש?האם ישנו שימוש במנגנון תפיסת שגיאות?האם הבדיקה ודיבוג הקוד נוחים?האם הודעות המשתמש ברורות וידידותיות למשתמש?  טיפול בשגיאות :בצורה חלקית, חלק מהשגיאות מקבלים התראה וחלק לא, לדוגמא כאשר המשתמש לא מכניס שם משתמש וסיסמה (מחרוזת ריקה) לא מקבלים התראה  דיבוג הקוד: נוח  המשתמש לא מקבל הודעות בכלל על השגיאות לכן תשובה על השאלה האחרונה היא לא |
| בדיקות | האם יש בדיקות שניתן להוסיף?  כן, ישנן מספר בדיקות שניתן להוסיף כחלק מתהליך הפיתוח  בדיקות יחידה (Unit Testing): בדיקות יחידה מתבצעות על מנת לבדוק תפקוד נפרד של יחידות קוד קטנות כמו פונקציות או מחלקות בלבד. בדיקות אלו עשויות לבדוק שגיאות, תוצאות צפויות ותנאים מיוחדים בתוך יחידות הקוד הקטנות. |
| שימושיות | האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?  כן, המשחק קל לשימוש ומובן אבל לא מקבלים משוב או התראה אם נעשה משהו לא נכון לכן המשתמש לא ממשיך. |
| תיעוד | האם הממשק מתועד?  כן מתועד |
| אתיקה | האם יש שימוש בנתוני משתמש אשר מפרים פרטיותו?האם יש ניצול של התנהגות או חולשה אנושית?האם ישנו algorithmic bias כנגד קבוצה או פרטים מסוימים?  לא, אין ניצול או הפרת פרטיות במשחק. |
| אבטחה | האם ישנו מידע אבטחה (שמות משתמש, ססמאות) גלויים?  לא אין מידע אבטחה גלויים הכל שמור בדב |
| ביצועים | האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?  הוספת פונקציות נוספות או תהליכים מסובכים יותר עשויים להביא להאטת הביצועים. מצד שני, שיפורים באלגוריתמים או בשיטות המימוש עשויים להביא לשיפור בביצועים. |
| ביצועים | האם ניתן לשפר לדעתך את הביצועים?  כן תמיד ניתן לשפר את הפיצועים של הקוד, אפשר בכמה דרכים למשל:  אופטימיזציה של אלגוריתמים: ניתן לבחון את האלגוריתמים המרכזיים ולמצוא דרכים לשפר את ביצועיהם, להפחית את הזמן הריצה או לחסוך במשאבים.  קיבוץ ושילוב משאבים: ניתן לבחון את שימוש המשאבים כמו זיכרון, מעבד ושטח אחסון ולמצוא אופציות לשילובם ושימוש חכם יותר. לדוגמה, אופטימיזציה של מחסנית הקריאה, הגיבוי ושילוב קבצים לצורך קריאה יעילה וניהול משאבים. |
| קריאות - readability | האם הקוד מובן בקלות? אילו חלקים בקוד היו לא ברורים עבורך?כיצד ניתן לשפר את קריאות הקוד?  הקוד מובן בקלות והתיעוד מעל כל פונקציה עוזר להבין אותו יותר טוב, ניתן לשפר את קריאת הקוד על ידי הגדרת שמות משמעותיים מובנים לכל המשתנים ולא לחלק מהם. |
| סעיף פתוח  אתיקה ומוסר | הוסיפו קטגוריה נוספת לבדיקה ובצעו אותה  האם שינוי הקוד הזה מציג הטיה כלשהי של אלגוריתם, בינה מלאכותית או למידת מכונה?  לא הקוד אינו מציג הטיה של בינה מלאכותית או למידה מכונה הקוד הוא בסיסי. |
| סעיף פתוח  Readability | הוסיפו קטגוריה נוספת לבדיקה ובצעו אותה  האם ניתן להסיר כמה הערות על ידי הפיכת הקוד עצמו לקריאה יותר?  עדיף לא לשנות בקוד ולא לגעת בהערות, כי ההערות שבקוד מצומצמות ומובנות ועוזרות לקרוא את הקוד בצורה יותר יעילה. |

**שאלות פתוחות:**

1. *כיצד הרגשת במהלך החוויה של מעבר על הקוד?*

כחלק מכללת הבדיקה, עברתי על הקוד של הקבוצה, והרגשתי את החוויה הבאה:

* ניתוח הקוד: התחלתי בניתוח הקוד על מנת להבין את המבנה והלוגיקה שלו. ניתוח הקוד כלל קריאה והבנה של הפונקציות, המחלקות והקבצים השונים. בהתבוננות בקוד, הצלחתי לזהות רעיונות עיקריים, אלגוריתמים ותהליכים שממומשים בתוך הקוד.
* בניית תמונה כוללת: לאחר ניתוח הקוד, בניתי תמונה כוללת של כלל המערכת המוקדמת בקוד, והתחלתי להבין את דרכי התקשורת בין המודולים והפונקציות. זה כלל גילוי והבנה של המחלקות והפונקציות המרכזיות, כמו גם את הזרימות העיקריות במערכת הקוד.
* זיהוי קשיים ובעיות: במהלך הבדיקה, זיהיתי קטעים בקוד שיכולים לגרום לקשיים או שיביאו לבעיות בתהליך הרצת המערכת. זה כלל שגיאות לוגיות, תפקודים לא צפויים או קטעים של קוד שטחיים.
* עריכה ושיפור: כחלק מתהליך הבדיקה, ערכתי שינויים ושיפורים בקוד על מנת לתקן את הבעיות שזיהיתי.

1. *האם לדעתך ניתן לשפר תרגיל זה?*

*ניתן לשפר את התרגיל על ידי שיחה אישית עם המתכנתים כדי שתהיה הבנה עמוקה בין הבודק והמתכנת.*

1. *הערות נוספות* – אין

**שם הסטודנטית: אמנה סלאמה**

|  |  |
| --- | --- |
| **קריטריון** | **הסבר** |
| מימוש | האם הקוד מבצע את הנדרש?  בעיקרון כן, הקוד מתחיל עם פונקציה שמציגה את ממשק המשחק, מציגה שאלות ,אפשרויות תשובה וכפתורים עבור המשתמש לבחירת התשובה , עם כל בחירה מעניקים ניקוד ומחשבים תוצאה סופית . |
| יעילות | האם ניתן לשפר את יעילות הקוד?  כן, במקום לקרוא ל firebase כמה פעמים לטעינת השאלות היה אפשר לשלוף את כל השאלות פעם אחת ולשמור אותם במערך ולהשתמש בהם. |
| פשטות | האם ניתן לפשט את הקוד?  כן, למשל יכולים להימנע מכפילות לדוגמה הקוד שלהם שהם מייצרים לחצנים ומעצבים אותם חוזר על עצמו הרבה פעמים, אז ניתן לייצר פונקציה המקבלת מאפיינים של הלחצן כמו גדלים וכל זה ולהחזיר העיצוב שלה. |
| מודולריות | האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)  הקוד שלהם מודולרי אבל לדעתי ניתן לשפר גם, כך שיש חלקי קוד קטנים בתוך הפונקציות שניתן לחלץ אותם פונקציות עזר מקטנות יותר לדוגמה הקוד לעיצוב והצגת ודג'טים נראה חוזר על עצמו במהלך כל הקוד אז אפשר לייצר פונקציות נפרדות למשימות החוזרות האלה. |
| תבניות עיצוב | האם ישנו מימוש ברור ונכון של תבניות עיצוב?  לא,אין שימוש בתבניות עיצוב. |
| OOP | האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?  לא,אין שימוש ב OOP. |
| באגים וטעויות | האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?  כן, יש בעיה בחישוב הניקוד כך שהניקוד לא מחושב נכון. |
| טיפול בשגיאות | האם ישנה התראה על שגיאות משתמש?האם ישנו שימוש במנגנון תפיסת שגיאות?האם הבדיקה ודיבוג הקוד נוחים?האם הודעות המשתמש ברורות וידידותיות למשתמש?  בקוד שלהם לא מציגים תשובות אן הודעות למשתמש בצורה ישירה על שגיאות שיכולות להתרחש במהלך ההזנה , לדוגמה אם המשתמש הזין שם משתמש לא קיים או שגיאת קלט אחרת ,הקוד לא מציג הודעת שגיאה או התראה מסוימת.  דיבוג הקוד נוי כי הפונקציות מודולריות ולא מסובכות, אין שימוש במנגנון תפיסת שגיאות(היה אפשר להשתמש למשל ב try/catch. |
| בדיקות | האם יש בדיקות שניתן להוסיף?  כן, בדיקות קלט בהוספת שאלה או בעדכון שאלה. |
| שימושיות | האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?  כן הממשק שימושי, הוא נוח לשימוש אך לדעתי הוא מעצבן ולא מעורר רצון לשחק כי רוב השאלות חוזרות על עצמם ואין משוב על נקודות כל שאלה למשל. |
| תיעוד | האם הממשק מתועד?  כן הממשק מתועד. |
| אתיקה | האם יש שימוש בנתוני משתמש אשר מפרים פרטיותו?האם יש ניצול של התנהגות או חולשה אנושית?האם ישנו algorithmic bias כנגד קבוצה או פרטים מסוימים? לא אין שימוש בנתוני משתמש שמפרים פרטיותיו ואין ניצול של התנהגות או חולשה וגם אין algorithmic bias כנגד קבוצה או פרטים מסוימים. |
| אבטחה | האם ישנו מידע אבטחה (שמות משתמש, ססמאות) גלויים?  לא אין מידע אבטחה גלויים. |
| ביצועים | האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?  בעיקרון לא, הקוד מחולק וכל פונקציה עובדת לבד בלי תלות בפונקציות אחרות אז ניתן לשנות את הקוד ולהוסיף קטעים בלי לפגוע בביצועים. |
| ביצועים | האם ניתן לשפר לדעתך את הביצועים? כן אפשר להפחית את השימוש בפונקציות גלובליות כי הם איטיות מאשר פונקציות מקומיות . |
| קריאות - readability | האם הקוד מובן בקלות?אילו חלקים בקוד היו לא ברורים עבורך?כיצד ניתן לשפר את קריאות הקוד?  הקוד מובן באופן כללי כי הוא מתוכנת באופן מסודר ומכיל הסברים בעזרת הערות אך אפשר לשפר את קריאת הקוד למשל על ידי שימוש בשמות משתנים ברורים יותר לדוגמה להשתמש ב leaderboard במקום lb. |
| סעיף פתוח  Implementation | הוסיפו קטגוריה נוספת לבדיקה ובצעו אותה   * Could an additional framework, API, library, or service improve the solution?   כן אפשר להשתמש בשירותי API למשל לשאילתות אם ניקח מאגר שאלות מוכן זה יכול לחסוך זמן ביצירת שאלות. |
| סעיף פתוח  Logic Errors and Bugs | הוסיפו קטגוריה נוספת לבדיקה ובצעו אותה   * Can you think of any inputs or external events that could break the code?   בעיה בחיבור לרשת : אם אין חיבור לאינטרנט לא ניתן לשלוף נתונים ממסד נתונים, אז כל הפרעה או כשלון ברשת עשויים לגרום לכשלים בקוד . |

**שאלות פתוחות:**

1. *כיצד הרגשת במהלך החוויה של מעבר על הקוד?*

*הייתה דרך מאתגרת לעבוד עם הקוד והיה צורך בהשקעה נוספת לקריאת הקוד והבנת הלוגיקה.*

1. *האם לדעתך ניתן לשפר תרגיל זה?*

*כן ניתן לייצור תקשורת עם הקבוצה האחרונה ולבצע הבדיקות מולהם שיהיה מובן לנו בדיוק מה היתה הכוונה שלהם*

1. *הערות נוספות*

תודה על המאמץ שאתן משקיעות כדי ללמד אותנו .

**שם הסטודנט: עימאד סלימאן**

|  |  |
| --- | --- |
| **קריטריון** | **הקריטריון המרכזי של הקוד הוא יצירת ממשק משתמש למשחק שאלות ותשובות. הקוד מתחיל עם חלון התחברות שבו מתבקש מהמשתמש להזין את שם המשתמש והסיסמה. במידה והפרטים נכונים, המשתמש מועבר למסך הראשי של המשחק.במסך הראשי, המשתמש מתבקש לבחור באפשרות "התחל משחק" כדי להתחיל את המשחק. כאשר המשתמש לוחץ על האפשרות, המסך עובר למסך המשחק עצמו.במסך המשחק, המשתמש מתבקש לענות על שאלות. השאלות נטענות ממסד הנתונים של** Firebase**. המשתמש יכול לבחור את התשובה הנכונה מבין האפשרויות שמוצגות לו וללחוץ על הכפתור המתאים לאישור תשובה. לאחר כל תשובה, מתבצעת בדיקה של התשובה הנבחרת והתשובה הנכונה. על פי הבדיקה, מתעדכנים משתנים שמשמשים לספירת התשובות הנכונות והתשובות הלא נכונות.בנוסף, לאחר סיום השאלות, מוצג סיכום תוצאות המשחק וניתן לשחק מחדש או לחזור לתפריט הראשי.הקריטריון המרכזי בקוד הוא לספק ממשק משתמש נוח וידידותי למשתמש.** |
| מימוש | כן הוא מבצע את הנדרש מבחינת התחברות למשתמש וגם הצגת שאלות למשתמש והצגת תוצאות בסוף המשחק . |
| יעילות | תמיד אפשר לשפר את היעלות של הקוד , הקוד של הקבוצה יעיל מאוד מכל הבחינות ופשר ליעל יותר את קטע ההתחברות של המשתמש . |
| פשטות | ניתן לפשט אותו ולשפר את יעילותו וקריאתו. ניתן לזהות כמה אזורים בקוד שמכילים ריבוייות וחוזרים על עצמם, ובאמצעות הפקת פונקציות, מחלקות או קוד משותף ניתן לפשט את הקוד ולהפוך אותו לקוד יותר יעיל וקל יותר לתחזוקה. |
| מודולריות | ניתן לראות שהוא מחולק לפונקציות ובלוקים שונים. ישנם חלקים של הקוד שמתבצעים בתוך פונקציות נפרדות וניתן לראות חלוקה של הקוד ליחידות קטנות יותר. מבחינה זו, ניתן לציין שהקוד שהזנת במידת האפשר עומד במודולריות, שהיא היכולת לחלק את הקוד ליחידות קטנות וסלעות שמתבצעות בתוך פונקציות או מודולים נפרדים. |
| תבניות עיצוב | כו יש מימוש ברור ונכון נראה שהקוד מתמקד בלוגיקה ובפונקציות הקשורות למשחק ולשאלות. עיצוב הממשק הוא מוגבל וקיימים רק כמה ניסיונות לעיצוב הקלטות המשתמש עם כפתורים ותוויות.נראה שהקוד מתרכז בפונקציות הפועלות ובביצוע הפעולות המתאימות למשחק, ולא בתבניות עיצוב מובהקות וממומשות בצורה מלאה. זה יכול להיות מקור פשיטות הקוד והתמקדות בלוגיקה העיקרית של המשחק עצמו, ולא בעיצוב ובממשק המשתמש. |
| OOP | לא ניתן לראות שימוש מובהק בעקרונות התכנות המונחה עצמים (OOP). הקוד מופשט ולא מאורגן לפי מחלקות ואובייקטים, ואין בו שימוש בטכניקות כמו ירושה, חזור על עצמך (polymorphism), או חבויות (encapsulation) שקשורות לOOP. |
| באגים וטעויות | במעבר שלי על הקוד לא ראיתי באגים וטעיות בקוד . |
| טיפול בשגיאות | אין התראות ישירות על שגיאות משתמש (user errors). יתכן וישנם בדיקות ותנאים בקוד שבודקים תנאים מסוימים ומפעילים פעולות מתאימות בהתאם, אך אין הודעות מפורשות שמוצגות למשתמש במקרה של שגיאה משתמש. |
| בדיקות | בדיקת קלט תקין: ניתן להוסיף בדיקות שמוודאות שהקלט שהוזן על ידי המשתמש הוא תקין ותואם את הציפיות. זה יכול לכלול בדיקות כמו בדיקת אורך הקלט, סוג הקלט, או בדיקת התאמה לפורמט מסוים. |
| שימושיות | כן המשחק מעורר רצון להמשיך לשחק תוך השימוש ברקע ו סוג השאלות הוא מעורר רצון להמיך במשחק וללמוד עוד ועוד דברים . |
| תיעוד | כן הוא מתועד בצורה נכוכנה וברורה וניתן להביין מהתיעוד מה כל פונקציה מבצעת . |
| אתיקה | לגבי algorithmic bias, גם כאן, לא ניתן לראות באופן ישיר מתוך הקטעים שהוזנו אם ישנם איבייסים אלגוריתמיים כנגד קבוצה או פרטים מסוימים. זה דורש ניתוח עמוק של האלגוריתמים והמערכת כולה, ובדיקה של הפלט שלהם על פי קבוצות שונות ופרטים מסוימים. |
| אבטחה | הכל בנוי בצורה מאובטחת אין שם משתמש גלוי וגם אין ססמאות גלוית הוא בנוי בצורה מאובטחת ויעילה. |
| ביצועים | יתכן ששינוי עתידי בקוד יכול להשפיע על ביצועי המערכת, אך השפעתו תלויה בסוג השינויים ובמימוש המדויק שלהם. בדיקות מדויקות והערכה יכולות לסייע בזיהוי וטיפול בבעיות בביצועים ובמידת הצורך, לבצע שיפורים וביצועים מתאימים. |
| ביצועים | אפשר לשפר את מיטוב תקשורת עם מסדי הנתונים ושליפת הניתונים מתוך database. |
| קריאות - readability | באופן כללי הקוד קרי הוא מחולק בצורה טובה שניתנת להבנה , כל פונקציה יש עליה תיעוד והפעלות שבתוך הפונקציה ברורות ומובנות . |
| סעיף פתוח | קטגוריה נוספת לבדיקה: אבטחת מידע (Data Security)בבדיקה זו נבדוק את האפשרויות הנלוות לאבטחת המידע בקוד. נקודת המרכז תהיה השמירה על פרטיות המשתמש והמניעים למניעת גישה למידע לא מורשה. נבדוק האם ישנם מנגנונים להגנה על המידע הרגיש, האם מתבצעות בדיקות תקיפות ותפיסת שגיאות רלוונטיות, והאם משתמשים בכלים מתאימים להגנה על המידע. |
| סעיף פתוח | קטגוריה נוספת לבדיקה: ניהול טעויות (Error Handling)בבדיקה זו נבדוק את יכולת הקוד לטפל בשגיאות ולנהל טעויות בצורה יעילה ויוזמתית. נבדוק האם הקוד מזהה ומטפל בסוגיות שגיאה כמו קלט לא תקין, תקלות בשרת או תקשורת, וטעויות בלוגיקת התוכנה. נבדוק האם הקוד מציע הודעות ברורות ומותאמות למשתמש על טעויות שקרו, והאם הוא מספק מידע מועיל לצורך דיבוג ותיקון השגיאות. |

**שאלות פתוחות:**

1. *כיצד הרגשת במהלך החוויה של מעבר על הקוד?*

*במהלך החוויה של מעבר על הקוד שהוזן, הרגשתי שהקוד מסודר וקל לקריאה. השימוש בשפת פייתון הקלה על ההבנה של הקוד ועזרה לי לזהות את המבנה הכללי של היישום.במהלך הקריאה של הקוד, הייתה לי הבנה טובה של הפונקציות השונות והקשרים ביניהן. השימוש בפונקציות ומשתנים בשמות מתאימים והקוד הפשוט והידוע מאפשרו לי להבין בקלות את מהלך התהליך והלוגיקה של היישום.הקוד הופקד בצורה טובה ונראה כי התווספו הערות והסברים כלליים שהבהירו את מטרת כל קטע קוד ואת הפונקציה שלו במסגרת היישום הכללית.בסופו של דבר, החוויה של מעבר על הקוד הייתה חיובית ומרגישה כמו עבודה בסביבה נקייה ומסודרת של פיתוח תוכנה.*

1. *האם לדעתך ניתן לשפר תרגיל זה?*

*ניתן לזהות כמה אזורים שבהם ניתן לשפר את התרגיל:שימוש בעקרונות OOP: ניתן לשפר את הקוד על ידי המרתו לתכנון מונחה עצמים (OOP). זה יכול לאפשר קידום נוסף של יכולות התחזוקה, ההרחבה והשימושיות של הקוד.*

**שם הסטודנט: סוהיל חורי**

|  |  |
| --- | --- |
| **קריטריון** | **הסבר** |
| מימוש | כן המערכת פחות ויותר עומדת בכל הדרישות ומבצע את הדרישות גם נכון.   האם הקוד מבצע את הנדרש? |
| יעילות | הקוד מבחינת יעילות כן יעיל רק בעת הצורך ניגשים למסד הנתונים ולא כל פעם שמביאים שאלה למשל. תצוגת השאלות והתגובות במערכת היו מהירים מה שמראה שהקוד יעיל.  האם ניתן לשפר את יעילות הקוד? |
| פשטות | לא, הפתרון והקוד שלהם פשוט והכי יעיל.  האם ניתן לפשט את הקוד? |
| מודולריות | כן מחולק נכון ומודולרי, כלומר קובץ ה CSS ממומש בנפרד מהקוד של הלוגיקה, פונקציות מחולקות נכון ואין תלות ביניהן.  האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות) |
| תבניות עיצוב | כן היה שימוש ב MVC: Model: היה רכיב שאחראי על הנתונים והתקשורת עם מסד הנתונים ועם ה Firebase. View: קיום שכבה שאחראית על התצוגה למשתמש ושימוש ב IPython widgets. Controller: כן הייתה שכבה שאחראית על הטיפול בקלטים של המשתמש, היה שימוש ב event handler.  האם ישנו מימוש ברור ונכון של תבניות עיצוב? |
| OOP | כן הפונקציות כמעט לא תלויות אחת בשניה וגם השתמשו בסוג של זריקת תלויות (Dependency Injection) וגם ב Single Responsibility Principle וזה חשוב. אין שכפולים בקוד.  מבחינת OOP זה היה מספיק טוב. היה אפשר לשפר מבחינת העיקר הבא: Open-close principle האומר שהוספת פונקציונליות מסוימת גוררת מינימום שינויים בקוד, אולי זה מתגלה למשל אם רוצים להוסיף תצוגה דומה למשחק עצמו בפונקציה loadGame() שם הציגו את קוד ה html בתוך הפונקציה עצמה, היה עדיף לשים את קוד ה html בפונקציה נפרדת ואז לקרוא לפונקציה הזאת, כך שאם נצטרך להוסיף תצוגה דומה אז נשתמש בפונקציה הזאת.  האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר? |
| באגים וטעויות | כשעושים Edit לשאלה קיימת מנסים לשנות שדה מסוים לוחצים על edit(כלומר סיום הפעולה) ואז שוב פותחים את אותה שאלה, השאלה לא מעודכנת. ובמקום לעדכן את השאלה המערכת מוסיפה את השאלה במקום.  האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו? |
| טיפול בשגיאות | במסך ההתחברות אם מנסים להתחבר עם שדות ריקים אין הודעה על שגיאה ומראה שגיאה בקוד. אין מנגנון של תפיסת שגיאה בהתחברות. דיבוג כן נוח לא מסובך בקוד. כשמכניסים נתונים לא קיימים מופיע spinner אבל אין הודעה על אי קיום המשתמש. שאר ההודעות במערכת כן ידידותיות. |
| בדיקות | כן, בדיקת עומס Load Testing בדיקה שכאשר מתחברים מכמה משתמשים באותו הזמן, המערכת מתנהגת טוב תחת העומס. והמערכת הנבדקת כן עומדת בבדיקה זו.  האם יש בדיקות שניתן להוסיף? |
| שימושיות | המשחק מאוד יפה עם עיצוב יפה ומראה ההתקדמות שלי במשחק ומאפשר להתחרא עם אחרים בניקוד, אז מבחינה זו כן רוצה לשחק שוב. אבל הבעיה זה שכל פעם מנסים לשחק נותן לשחק רק 5 שאלות, אם רוצה עוד שאלות צריך לצאות ולשחק שוב וזה מעצבן. גם אם לא סיימנו את 5 השאלות ולוחצים exit לסיום המשחק, הייתי מצפה לדעת כמה עניתי נכון ולהציג את הגרפים אפילו אם יצאתי מה session. וגם אין הסבר על כמה נקודות זוכים על כל שאלה נכונה.  האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק? |
| תיעוד | כשלוחצים על כפתור View נפתח את גרף של התוצאות, ולא היה ברור מה זה View לראות מה בדיוק? כשמסיימים משחק Session מופע כמה נקודות זכה בכל ההיסטוריה שלו וזה לא ברור כי חשבתי שמופיע את הניקוד של כמה צברתי ב session עצמו. שאר הממשק כן מתועד נכון וברור.  האם הממשק מתועד? |
| אתיקה | לא אין הפרה בניתונים לא מוצג מידע אישי ואין ניצול או algorithmic bias.  האם יש שימוש בנתוני משתמש אשר מפרים פרטיותו?האם יש ניצול של התנהגות או חולשה אנושית?האם ישנו algorithmic bias כנגד קבוצה או פרטים מסוימים? |
| אבטחה | לא בעת ההתחברות ב login הסיסמה אנונימית וזה טוב. וגם אי אפשר לדעת פרטי השחקנים האחרים רק את ה username שלהם וזה ליגיטימי.  האם ישנו מידע אבטחה (שמות משתמש, ססמאות) גלויים? |
| ביצועים | לא חושב, כי הפונקציונליות והפונקציות של המערכת מופרדות בצורה טובה כך שאם רוצים להוסיף או לשנות בקוד לא יפגע בביצועים.  האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים? |
| ביצועים | כן להוסיף שאפשר לבחור על כמה שאלות לענות, אפשרות לצאת מהמשחק וגם עם זה לצפות בגרפים.  האם ניתן לשפר לדעתך את הביצועים? |
| קריאות - readability | כן הקוד מובן מאוד ומחולק נכון בין עיצוב ללוגיקה למסד נתונים. אין משהו לא ברור כל פונקציה מתועדת וגם השתמשו בשמות משמעותיים גם לפונקציות וגם למשתנים. אין מה לשפר.   האם הקוד מובן בקלות?אילו חלקים בקוד היו לא ברורים עבורך?כיצד ניתן לשפר את קריאות הקוד? |
| Accessibility | בדיקת האם המערכת מתאימה לאנשים עם מוגבליות למשל קושי בהבנה. המערכת קלה להבנה ובדקתי אותה עם חבר שיש לו קושי בהבנה והמערכת עברת את הבדיקה זו. |
| בדיקת נטוורק Network Testing | בדיקת איך המערכת מתנהגת ומתמודדת עם רשת איטית למשל. |

**שאלות פתוחות:**

1. **כיצד הרגשת במהלך החוויה של מעבר על הקוד?  
  
נהניתי מאוד כי ראיתי כמה הם מסודרים בקוד, והבנתי אותו בקלות כי הכל היה מתועד. וגם היה כיף ללמוד דברים חדשים בקוד עצמו.**

2. **האם לדעתך ניתן לשפר תרגיל זה?  
לא אין היה מעולה ומועיל ללמוד ולהסתכל על עבודות של האחרים, לא רק למשחק עצמו אלא גם על הקוד עצמו.**

3. **הערות נוספות  
אין.**