**תרגיל בית 3 חלק ב':**

**מגישים:**

נוי ביטון - 312238678

דולב סרי - 313428005

דבי דובינקר - 318179033

אייל קובי - 209291418

נופר בן-סימון - 308468826

אילנית חנוכו - 208657320

| **Results** | **Precondition** | **Expected Result** | **Description** | **TestID** | **#** |
| --- | --- | --- | --- | --- | --- |
| Success | user is defined in the db | user logs in  update user’s status in db | open login window  enter username and password  click ok | login\_success | **1** |
| Success | user is not defined in the db | user not logs in  error message appears in the screen | open login window  enter username and password  click ok | login\_fail wrong-password | **2** |
| Success | user is not defined in the db | user not logs in  error message appears in the screen | open login window  enter username and password  click ok | login\_fail\_wrong-username | **3** |
| Success | the game is logged to the db  the correct answer matches the user’s choice | success message appears on the screen  the user’s score increases by one | click on one of the optional answers  submit the choice | correct\_answer\_to\_question | **4** |
| Success | the game is logged to the db  the correct answer doesn’t match the user’s choice | a message with the correct answer appears on the screen (so that the user can learn for the future)  the user’s score does not change | click on one of the optional answers  submit the choice | incorrect\_answer\_to\_question | **5** |
| Success | the topic exist in the db | a new question is added to the selected topic | get to the question editor page and select a topic  click “add new question”  add the question and the four optional answers | add\_new\_question(manager) | **6** |
| Success | the topic exist in the db | the question is deleted from the db | get to the question editor page and select a topic  select an existing question and click delete | remove\_question\_fromDB(manager) | **7** |
| Success | the topic exist in the db | the question is updated from the db | get to the question editor page and select a topic  select an existing question and click update and save | update\_question\_fromDB(manager) | **8** |
| Success | None | a new topic is added to the game | get to the question editor page  add new topic | add\_new\_topic(manager) | **9** |
| Success | the selected topic existed in the db | the selected topic is deleted from the db | get to the question editor page  select a topic and click delete | remove\_existing\_topic\_fromDB(manager) | **10** |
| Success | user is defined in the db as logged in | the user logs out  the user’s status updates in the db | click on the logout button | logout | **11** |

**חלק ראשון – חלק קבוצתי:**

**מסקנות:**

* ניתן להסיק מתוצאות ההרצה כי הקוד עומד בכל הדרישות.
* טופלו כל מקרי הקצה אותם העמדנו לבדיקה.
* מקום לשיפור – החלפת סביבת העבודה (Google Colab) מאחר והמשחק דורש ביצועים גרפיים גבוהים יותר, סביבה עננית שרצה על גבי דפדפן איננה מספיקה.

**חלק שני – עבודה אישית:**

**אייל קובי 209291418**

1. **האם הקוד מבצע את הנדרש?**

לגמרי, הקוד מבצע את כל הדרישות מגרפים מעניינים דרך session-ים למסד הנתונים וכלה בעדכוני שאלות.

1. **האם ניתן לשפר את יעילות הקוד?**

הקוד מסודר ויעיל ברמה גבוהה מאוד (!!!). אם חייב להעיר משהו על יעילות הייתי אולי מעיר על דברים ממש קטנים וזניחים ביעילות כמו שורות 662-663 שבהן יש if מקונן שלא לצורך לדוגמא: את ה-if המקונן ניתן היה להחליף באמצעות and (כי נכנסים ל-branch רק אם שני התנאים מתקיימים).

יש הרבה מה ללמוד מהם על יעילות קוד.

1. **האם ניתן לפשט את הקוד?**

מהבחינה הזאת לשמחתי גם אין לי משהו להגיד כמעט בכלל.

אם בכל-זאת חייב להעיר משהו הייתי מעיר על חלקי קוד מתועדים (טיוטות ככל הנראה) ורווחים מוגזמים שניתן היה למחוק.

1. **האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)**

כפי שציינתי לעיל גולת הכותרת של הקוד היא המודולריות שלו. הקוד מחולק לשני cell ראשיים, שבכל אחד מהם מחלקות, אובייקטים, פונקציות והכל מסודר מאוד!!!

ניתן לראות כי חברי הצוות צלחו את קורס 'מונחה עצמים' בהצטיינות – הביצועים מרשימים מאוד ומזכירים אפילו עבודה שלמה ב-Java.

1. **האם ישנו מימוש ברור ונכון של תבניות עיצוב?**

ניתן לראות שימוש בתבנית העיצוב Singleton בעת יצירת חלון ה-popup במסך המבוך של השחקן. ה-popup מיוצר פעם אחת בלבד בעזרת המתודה show.

1. **האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?**

כמצוין לעיל העבודה מונחית העצמים בפרויקט זה ראויה להערכה, מדויקת מאוד, מסודרת מאוד.

אין לי הצעת ייעול כלל בנושא זה לצערי וגם לשמחתי😊

1. **האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?**

לא צפיתי במקרה בו הקוד מתנהג לא כצפוי, זהו קוד גדול ומאסיבי במיוחד (ניכר שעבדו עליו הרבה מאוד זמן) אז יכול להיות גם שפספסתי, אני גם בן אדם..

1. **א. האם ישנה התראה על שגיאות משתמש?**

**ב. האם ישנו שימוש במנגנון תפיסת שגיאות?**

**ג. האם הבדיקה ודיבוג הקוד נוחים?**

**ד. האם הודעות המשתמש ברורות וידידותיות למשתמש?**

**א.** כן, יש התראות מושקעות, יפות ומצחיקות:

לדוגמא, במידה וטעיתי בשאלה מסוימת כתוב לי ya noob (משהו כזה..) עם רקע אדום מקווקוו, שזה די נחמד, ואם צדקתי אז כתוב לי Good Job עם רקע ירוק.

תענוג לשחק ככה!

אם מדובר על שגיאה לא משחקית אלא טכנית גם לזה יש התראות – לדוגמא ישנה הודעה למשתמש במידה והוא מזין username ו-password שאינם שייכים לאף user ב-db. במקרה זה נזרקת הודעת שגיאה עם קווים אדומים ברקע..

**ב.** כן, הקוד ממודל בצורה מאוד נוחה לתפיסת שגיאות. הקוד מופרד לחלקים קטנים שכל אחד מהם ניתן לדבג בנוחות וביעילות ללא חשש.

**ג.** עקב הסדר של הקוד והחלוקה בו, כאשר ניסיתי לבדוק האם דברים עלולים ליפול או ליצור בעיות היה לי מאוד נוח ופשוט לעשות כן. בנוסף-על-כך שמות המשתנים ברורים וטבעיים דבר שמקל מאוד על ההתמצאות בו.

**ד.** הודעות המשתמש ברורות וידידותיות מאוד לעניות דעתי. אולי יש שימצאו את המילה noob כפוגענית, ברם אני רואה בזה משהו חיובי, מצחיק, ומראה שמי שכתב את הקוד באמת נהנה ובא לתת לי חווית משתמש כיפית וקלילה.

1. **האם יש בדיקות שניתן להוסיף?**

ניתן לבדוק האם הגרף מוצג בסוף משחק או לא..

במקרה הכינותי מראש מה שנקרא ובדקתי זאת – יש גרף, הוא צבוע בירוק או באדום לתשובות נכונות או שגויות בהתאמה ויש לו אפילו אלמנט של hover בו ניתן לראות באיזה נושא.

בנוסף, הם לא הסתפקו בחיווי גרפי רגיל של נכון ולא נכון ונתנו אינדיקציה לשחקן באיזה נושא הוא היה טוב יותר, ובאיזה הוא היה טוב פחות (!!!).

1. **האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?**

הממשק שימושי מאוד, מעורר צורך לשחק בו (!!!).

אולי הדבר היחיד שהייתי משפר זה מעבר מ-Google Colab לסביבת עבודה תומכת יותר כפי שמצוין בביקורת הקבוצתית..

מאוד אהבתי את החיווי והשינויים כשאני לוחץ על דברים במסך, את התמונות הנחמדות המצוירות, את הסאונד המפחיד, את השליטה בסאונד, חווית משתמש עוצרת נשימה 😊 😊 !!!

1. **האם הממשק מתועד?**

הממשק מתועד אמנם חלקית מאוד, אבל שמות המשתנים ברורים, הקוד ברור, יש לו flow מאוד נהיר ולכן על אף התיעוד הלוקה בחסר האפליקציה נוחה לשימוש עד מאוד.

1. **א. האם יש שימוש בנתוני משתמש אשר מפרים פרטיותו?**

**ב. האם יש ניצול של התנהגות או חולשה אנושית?**

**ג. האם ישנו algorithmic bias כנגד קבוצה או פרטים מסוימים?**

**א.** אין שום שימוש בנתוני משתמש אשר מפרים את פרטיותו בשום צורה.

האפליקציה מספקת מידע במשחק רק לגבי ניקוד..

אולי ניתן להסתכל על ה- Leader Board כסוג של פגיעה בפרטיות אם ממש מנסים

לחפש סכנה לפרטיות אבל לא מעבר. יתר-על-כן בעיני זו אפילו לא פגיעה בפרטיות וזה feature (מאפיין) חמוד שקיים בהמון משחקים שאנו מכירים באינטרנט לדוגמא.

**ב.** אין שום ניצול של התנהגות וחולשה אנושית במערכת בשום צורה שהיא.

אין לי אפילו מה להרחיב על כך.

**ג.** אין כאן שום algorithmic bias כנגד קבוצה או פריט.

למה שסטודנט שנה ג' בקורס מבוא למחשוב ענן ישתמש ב-algorithmic bias בפרויקט סוף של משחק בנושא Python?

1. **האם ישנו מידע אבטחה (שמות משתמש, סיסמאות) גלויים?**

שמות המשתמש והסיסמאות גלויות אך ורק למנהלנים וגם זה באופן חלקי – על-מנת למחוק, לערוך, להוסיף, וגם להפוך למנהלן משתמש מסוים עלייך לדעת במדויק את פרטיו. זה אמנם אולי לא user friendly במיוחד אבל הרבה יותר מאובטח מגישה מוחלטת של מנהלן לכל פרטי המשתמשים ללא כל מאמץ.

לכל הפחות על המנהלן יהיה לדעת את שם המשתמש והסיסמא של אותו אדם.

1. **האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?**

דבר ראשון באופן עקרוני תמיד שינויים בקוד יכולים לפגוע בכל-מיני דברים, צריך לדעת מה היקף השינוי בקוד? מה התווסף? מה ירד? שינוי אפילו הכי קטן גם בקוד הכי איכותי שנראה יכול לגרור תקלות רבות אם מחקנו לדוגמא 3 שורות קריטיות..

אם נשים בצד לרגע את הגנריות הבעייתית בשאלה, לטעמי שינויים מבוקרים בקוד זה לא יפגעו בביצועיו, ממה שראיתי הקוד כתוב בצורה מסודרת, יעילה וניתן לשנות, להוסיף, ולהוריד פונקציונאליות באופן כירוגי, מדויק מאוד, מבלי לפגוע בחלקים אחרים בקוד.

לדוגמא אם ארצה לשנות משהו בכניסת המשתמש (להוסיף בדיקה או הערה נניח) אלך אל ה-class ששמו log\_user\_in ואשנה בעדינות משהו בו או ב-class דומה לו באותו נושא (מחלקות בנושאים דומים מקובצות ביחד, עוד דבר שמקל על עריכות בקוד).

1. **האם ניתן לשפר לדעתך את הביצועים?**

על-מנת לשפר את מהירות ואיכות הביצועים לטעמי ניתן לעשות אחד משני הדבר הבאים:

**א.** למחוק חלק מה-features הגרפיים והקוליים הקיימים, כולל תמונות וציורים, ובכך לוותר על עבודה רבה, מושקעת וטובה לטובת ביצועים.

**ב.** להחליף סביבת עבודה (רעיון שמצוין על-ידי בפעם השנייה) מ-Google Colab לסביבה יותר תומכת בעבודות מסוג זה.

אני אישית הייתי בוחר בסעיף ב'.

1. **א. האם הקוד מובן בקלות?**

**ב. אילו חלקים בקוד היו לא ברורים עבורך?**

**ג. כיצד ניתן לשפר את קריאות הקוד?**

**א.** כמו שאמרתי כבר קודם, לעניות דעתי הקוד מאוד מובן ומאוד קריא.

**ב.** אישית לי לקח יותר זמן לעשות את ה-switch מה שנקרא לתבניות ה-OOP שלפיהן הקוד בנוי. אז מהבחינה הזאת כל החלקים עם המחלקות לקחו לי יותר זמן להבין, להסתגל ולהיזכר, אך ברגע שנזכרתי דברים הסתדרו הרבה יותר והקריאה זרמה.

**ג.** ניתן להוסיף יותר תיעוד בחלקים קריטיים אך כמו שציינתי לפני - בקוד זה באופן ספציפי לא הורגש באופן משמעותי כלל חסרונו של חלק מהתיעוד.

1. **בדיקה ראשונה (סעיף פתוח ראשון) – לחיצה על כפתור ה-credit ובדיקה האם מומש.**

לחצתי על כפתור ה-credit ובדקתי שאכן מומש. לאחר שלחצתי עליו הופעלה מנגינה יפנית עם תמונה של כלב וכל-מיני דברים משעשעים שבה הם הודו לחברי הקבוצה. זה היה נחמד, משעשע, וכיפי.

אהבתי מאוד את היצירתיות של חברי הקבוצה.

1. **בדיקה שנייה (סעיף פתוח שני) – לחיצה במסך ה'מגרש' של השחקן ב'אזורים מתים' במסך על-מנת לוודא שאכן קיימים כאלה ולא כל לחיצה תכניס אותי לסוג משחק מסוים.**

ביצעתי בדיקה זו ולא התאכזבתי. הכפתורים אינם פרושים לאורך כל המסך, התצוגה מאוד ידידותית למשתמש ונהירה, ואפילו ניתן ללחוץ על '?' באמצע המסך אם אני צריך הסבר פשוט ומפורט לדרך שבה צריך לשחק במשחק. נחמד מאוד!

**שאלות פתוחות:**

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

תחילה הייתי צריך להתרגל לסגנון הכתיבה ומה שנקרא 'להיכנס לראש'. עד מהרה מצאתי את עצמי מבין את הקוד ומתעניין מאוד בדברים קטנים שאפילו גרמו לי לבקש מחברי הקבוצה עוד קצת הסברים לא פעם.

קוד מסודר, נקי, כמו שנאמר כבר למעלה – תענוג!!

1. **האם לדעתך ניתן לשפר תרגיל זה?**

לטעמי השאלות קצת חוזרות על עצמן ולא הכי נהירות, הייתי משאיר שאלות מרכזיות ומנסח יותר ברור.

1. **הערות נוספות..**

אין הערות נוספות, שמח שבדקתי את העבודה שבעיני אחת הכי איכותיות שראיתי לקורס זה.

**נוי ביטון 312238678**

1. **האם הקוד מבצע את הנדרש?**

       הקוד מבצע את הנדרש.

1. **האם ניתן לשפר את יעילות הקוד?**

       תמיד יש מקום לשפר קוד, לדעתי המשחק כתוב בצורה יעילה מאוד.

1. **האם ניתן לפשט את הקוד?**

       לפי מבט עיני, הקוד כתוב בצורה ברורה.

1. **האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)**

      הקוד בהחלט מודולרי ומחולק ל3 תאים.

1. **האם ישנו מימוש ברור ונכון של תבניות עיצוב?**

תבנית העיצוב שהשתמשו בו הוא סינגלטון, ניתן לראות במסך של המבוך שקופץ חלון פעם אחת                                      בלבד.

1. **האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?**

       אכן יש שימוש נכון בעקרונות OOP, אין לי מה להוסיף על מנת לשפר.

1. **האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?**

במענה הראשוני על שאלות מקבלים פידבק של מענה נכון או לא, בחלקכם רק כתוב מלל שקופץ וחלק קופצת הודעה עם רקע.

1. **א. האם ישנה התראה על שגיאות משתמש?**

           יש התראה, קופצות הודעות popup

**ב. האם ישנו שימוש במנגנון תפיסת שגיאות?**

כן

**ג. האם הבדיקה ודיבוג הקוד נוחים?**

כן, הקוד מחולק לחלקים אשר מקלים על הבדיקה ודיבוג

**ד. האם הודעות המשתמש ברורות וידידותיות למשתמש?**

לא אהבתי את הפידבק הניתן למשתמש בעת מענה שגוי, אין סיבה לרשום למשתמש ״you are noob", ההודעות לא ברור במדויק מכיוון ולפעמים הן לא קופצות כמו שרצו עם רקע, אלא רק מלל שמאוד מקשה לראות מה כתוב בגלל הרקע.

1. **האם יש בדיקות שניתן להוסיף?**

כן, האם הכפתור קרדיט פועל

1. **האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?**

      המשחק מעורר רצון להמשיך לשחק בעקבות המוזיקה ובזכות הגרפיקה של המשחק.

1. **האם הממשק מתועד?**

      הממשק אינו מתועד באופן מלא, מתומצת למדי.

1. **א. האם יש שימוש בנתוני משתמש אשר מפרים פרטיותו?**

            אין שימוש בנתוני משתמש שמפרים את פרטיותו.

**ב.  האם יש ניצול של התנהגות או חולשה אנושית?**

אין ניצול.

**ג.  האם ישנו algorithmic bias  כנגד קבוצה או פרטים מסוימים?**

לא.

1. **האם ישנו מידע אבטחה (שמות משתמש, סיסמאות) גלויים?**

      אין.

1. **האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?**

      לא.

1. **האם ניתן לשפר לדעתך את הביצועים?**

 כן, המשחק די איטי, מעבר בין דפים לא חלק, לוקח זמן לדברים לעלות. אני חושבת שזה בגלל הסביבת עבודה בה הם עבדו.

1. **א. האם הקוד מובן בקלות?**

          כן, הקוד די מובן בזכות שמות המשתנים שהם משמעותיים.

**ב. אילו חלקים בקוד היו לא ברורים עבורך?**

לא היה משהו שהוא לא ברור

**ג.  כיצד ניתן לשפר את קריאות הקוד?**

לחלק את המימוש של המשחק לבלוקים נוספים.

1. **בדיקה ראשונה (סעיף פתוח ראשון) – מחיקת יוזר מבלי להכניס פרטים**

נכנסתי דרך משתמש מנהל וניסיתי למחוק משתמש, קיבלתי הודעה שהפרטים שהזנתי לא נכונים, הודעה שציפיתי לקבל.

1. **בדיקה שנייה (סעיף פתוח שני) – ניסיון הוספת משתמש חדש מבלי להכניס מידע כלל**

נכנסתי דרך משתמש מנהל, ניסיתי להוסיף משתמש חדש מבלי להכניס שום פרט עליו, המצופה לקבל הודעת שגיאה שאי אפשר להוסיף עקב ואין פרטים, אך כן נוסף יוזר עם שם משתמש שנבחר לו באופן אוטומטי.

קפצה הודעה עם השם שהוכנס למערכת.

**שאלות פתוחות:**

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

       קושי בלהבין נקודות עמוקות של הקוד עקב ואין תיעוד מלא.

1. **האם לדעתך ניתן לשפר תרגיל זה?**

       לדעתי לחבר את חלקי המטלה יחדיו מבלי להמתין זמן להגשת חלק א

1. **הערות נוספות..**

       אין

**דבי דובינקר 3181790333**

|  |  |
| --- | --- |
| קריטריון | הסבר |
| **מימוש** | **האם הקוד מבצע את הנדרש?**  כן, האתר מכיל את כל האלמנטים שהוא צריך להכיל לפי הדרישות שקיבלנו. |
| **יעילות** | **האם ניתן לשפר את יעילות הקוד?**  נראה שהקוד כתוב בסך הכל בצורה די יעילה, כמובן שתמיד אפשר למצוא שיפורים קטנים שאפשר לעשות אבל בגדול הקוד מבצע את מה שנדרש ונראה יעיל. |
| **פשטות** | **האם ניתן לפשט את הקוד?** אם הפרויקט לא היה בגוגל קולאב והייתה אופציה לחלק את הקוד לקבצים אז לדעתי הקוד היה הוך לפשוט יותר, אבל במגבלות העבודה בקולאב נראה שהקוד די פשוט איך שהוא. |
| **מודולריות** | **האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)**  כן, הקוד מודולרי מספיק. |
| **תבניות עיצוב** | **האם ישנו מימוש ברור ונכון של תבניות עיצוב?**  בקוד נעשה שימוש בתבנית עיצוב סינגלטון, בעת יצירת חלון פופ-אפ. המימוש ברור ונכון. |
| **OOP** | **האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?**  כן, ישנה חלוקה מאוד ברורה ומסודרת. |
| **באגים וטעויות** | **האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?**  מהבדיקות שהרצנו על הקוד (גם בדיקות קבלה וגם ניסיון שימוש באתר), נראה שהקוד עובד בצורה תקינה ואין שום באגים או טעויות. |
| **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?**  כן יש התראה למשתמש על שגיאות – למשל במסך התחברות, אם המשתמש הזין פרטים לא נכונים תקפוץ לו הודעה מתאימה, ההודעות ברורות וידידותיות למשתמש.  נראה שקיים מנגנון תפיסת שגיאות, בכל מקום בו יכולה להופיע שגיאה יש קטע קוד שמטפל בה. |
| **בדיקות** | **האם יש בדיקות שניתן להוסיף?**  **ממעבר על הקוד לא ראיתי עוד בדיקות שאפשר להוסיף, אבל כמובן שייתכן שיש עוד בדיקות שפשוט פספסתי.** |
| **שימושיות** | **האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?**  כן הממשק מאוד קל לשימוש. המשחק מכיל הרבה אלמנטים של משחוק שלמדנו עליהם בקורס ולכן אכן מעודד את השחק להמשיך לשחק. |
| **תיעוד** | **האם הממשק מתועד?**  התיעוד הוא די מינימלי, אפשר היה לתעד יותר בהרחבה.  עם זאת, נראה ששמות הפונקציות משמעותיים מה שמקל על קריאת הקוד. |
| **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?**  אין שימוש בנתוני המשתמש שמפרים את פרטיותו, אין algorithmic bias כנגד קבוצה או פרטים מסוימים. |
| **אבטחה** | **האם ישנו מידע אבטחה (שמות משתמש, ססמאות) גלויים?**  לא, אין שום מידע גלוי. |
| **ביצועים** | **האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?**  כן. תמיד שינויים/תוספות לקוד יכולים לפגוע בביצועים. במידה ורוצים לבצע שינויים צריך לעשות זאת בצורה חכמה ולבצע הרבה בדיקות תוך כדי על מנת לשמר את רמת הביצועים של הקוד. |
| **ביצועים** | **האם ניתן לשפר לדעתך את הביצועים?**  כן, פיתוח אתרים בסביבת גוגל קולאב אינו הכי יעיל, לכן אם נעביר את הקוד לסביבת עבודה אחרת, אשר יותר מתאימה לפיתוח אתרים, אפשר יהיה לשפר ביצועים. |
| **קריאות – readability** | **האם הקוד מובן בקלות? אילו חלקים בקוד היו לא ברורים עבורך? כיצד ניתן לשפר את קריאות הקוד?**  במסגרת המגבלות של גוגל קולאב, הקוד ברור וקריא.  אם הפרויקט היה נעשה בסביבה אחרת, המאפשרת חלוקה לקבצים, הדבר היה מקל על קריאת והבנת הקוד.  כמו כן, כפי שכבר ציינתי, התיעוד הוא מינמלי – הרחבת התיעוד גם הייתה עוזרת בהבנת הקוד. |
| **סעיף פתוח 1 - Reusability** | כפי שכבר ציינתי, בגלל האתגרים עם העבודה בקולאב, יהיה קשה להשתמש בקטעי קוד מפרויקט זה עבור פרויקטים אחרים. |
| **סעיף פתוח 2 - Maintainability** | הקולאב היא סביבה שעדיין נמצאת בפיתוח, לכן בגרסאות עתידיות שלה ייתכן ואלמנטים שעובדים עכשיו בצורה מסוימת, יעבדו בצורה אחרת, ויהיה צורך לעשות שינויים בהתאם – דבר אשר הופך את תחזוק הקוד למשימה מייגעת. זאת בנוסף לכך שהעבודה עם הקולאב בגרסה הנוכחית היא לא פשוטה – לתחזק קוד ארוך כל כך תמיד יהיה משימה מאתגרת. |

**שאלות פתוחות:**

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

לעבור על קוד של מישהו אחר זה תמיד מאתגר ובמיוחד במקרה זה בו הקוד מאוד ארוך ואין חלוקה לקבצים בגלל המגבלות של גוגל קולאב, דבר שאנחנו לא רגילים אליו, כי בכל קורס אחר עבדנו עם סביבות שמאפשרות זאת. אם זאת הקוד כן היה מסודרת ומדולרי כמה שאפשר, דבר אשר הפך את תהליך המעבר על הקוד לפשוט יותר.

1. **האם לדעתך ניתן לשפר תרגיל זה?**

חלק מהשאלות נוגעות באותן נקודות, ואז גם התשובות קצת חוזרות על עצמן. בנוסף, היה קשה למצוא עוד 2 סעיפים פתוחים בנוסף לסעיפים הנתונים שגם ככה מקיפים את כל הקריטריונים הרלוונטים.

1. **ערות נוספות**

אין.

**נופר בן סימון 308468826**

|  |  |
| --- | --- |
| קריטריון | הסבר |
| מימוש | האם הקוד מבצע את הנדרש?   * **אכן הקוד מבצע את מה שהגדירו במשימה.**   **יש אופציה של התחברות למשתמש ומשחק.** |
| יעילות | האם ניתן לשפר את יעילות הקוד?   * **הקוד נראה יעיל אבל קצת אורך מדי.** |
| פשטות | האם ניתן לפשט את הקוד?   * **אני חושבת שרמת המשחק והאלמנטים שהם הציגו במשחק שלהם רשומים בצורה הכי טובה שניתן.** |
| מודולריות | האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)   * **כן הקוד מחולק בצורה מסודרת לפי מסכים והפונקציות הנדרשות** |
| תבניות עיצוב | האם ישנו מימוש ברור ונכון של תבניות עיצוב?   * **ניתן לראות כי יש מימוש של** singleton. |
| **OOP** | האם ישנו שימוש נכון בעקרונות **OOP**? האם ניתן לשפר?   * **כן, ישנה חלוקה למחלקות ואובייקטים בצורה מסודרת** |
| באגים וטעויות | האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?   * **נראה כי ישנה תגובה ומעבר למסך/פידבק רק לאחר חיווי מהמשתמש ולכן הקוד מתנהל תקין** |
| 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?   * **כן ישנה התראה בעת חיבור למשתמש עם שם משתמש וסיסמא לא תקינים.יש מנגנון תפיסת שגיאות. הבדיקה והדיבוג נוחים. בנוסף, ניתן לתת יותר זמן תצוגה להודעות המשתמש.** |
| בדיקות | **האם יש בדיקות שניתן להוסיף?**   * **נראה כי הבדיקות הקיפו את כל מקרי הקצה.** |
| שימושיות | האם הממשק שימושי? (**usability**)? האם המשחק מעורר רצון להמשיך לשחק?   * **המשחק אכן שימושי. המשחק נראה מאוד אטרקטיבי, מושך את העין ויוצר עניין באמצעות השמע ברקע.** |
| תיעוד | האם הממשק מתועד?   * **קיים תיעוד בקוד אבל יחסית מינימלי. ניתן יותר לפרט על הפונקציות ותפקידם.** |
| 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?   * **אני חושבת שהשימוש בנתוני המשתמש הם מידתיים ומאפשרים למשתמש לראות את קצב התקדמות למידתו, לכן אין פגיעה בפרטיותו.** |
| אבטחה | האם ישנו מידע אבטחה (שמות משתמש, ססמאות) גלויים?   * **אין מידע גלוי אך רמת האבטחה היא מינימלית מכיוון שהקולב לא מאפשר ואפילו מקשה על הצפנת הנתונים.** |
| ביצועים | האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?   * **כן מכיוון שסביבת העבודה של הקולב אינה רחבה ומכיוון שכמות האלמנטים בממשק רבים זה יעמיס על המערכת.** |
| ביצועים | האם ניתן לשפר לדעתך את הביצועים?   * **לא, גוגל קולב מאוד מוגבל ולכן לא ניתן לשפר את הביצועים מחשש לקריסת המערכת.** |
| קריאות – readability | האם הקוד מובן בקלות?אילו חלקים בקוד היו לא ברורים עבורך?כיצד ניתן לשפר את קריאות הקוד?   * **הקוד מאוד ארוך ולכן קשה לקרוא את כולו אבל בסך הכל הקוד היה מובן. ניתן לשפר את רמת התיעוד וכך יהיה קל יותר להבין מה כל פונקציה מבצעת.** |
| תחזוקה | * **סביבת העבודה של הקולב מקשה על שילוב שפות ותוכנות חיצוניות ובכך אין אפשרות לתחזק את הממשק כמו שצריך.** |
| שימוש חוזר | * **הקוד משלב הרבה אלמנטים שקצת מקשים על שימוש חוזר בקטעי הקוד של הממשק.** |

שאלות פתוחות:

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

*- היה בסדר. מאוד קשה לקרוא קוד בסביבת העבודה של הקולב מכיוון שאין חלוקה לקבצים וכל הקוד נמצא בבלוק אחד ארוך מאוד.*

1. ***האם לדעתך ניתן לשפר תרגיל זה?***

*- ניתן לתת יותר זמן להגשת חלק זה של התרגיל. בנוסף, בגלל הקולב, קריאת הקוד מאוד מסורבלת ומקשה על הבודק, לכן אפשר לחלק את העבודה בין חברי הצוות או לתת לבדיקה חלקים מסוימים מהקוד*

1. ***הערות נוספות***

*-* אין

**דולב סרי 313428005**

|  |  |
| --- | --- |
| **קריטריון** | **הסבר** |
| **מימוש** | **האם הקוד מבצע את הנדרש?**  כן, הקוד ממלא את כל דרישות הפרויקט. |
| **יעילות** | **האם ניתן לשפר את יעילות הקוד?**  כמו שנאמר לנו הרבה פעמים בעבר תמיד ניתן לשפר את יעילות הקוד, בנוגע לקוד המדובר לדעתי הוא יעיל מאוד ובמגבלות הידע שלי איני יכול לתת הצעה כיצד ניתן לשפר אותו. |
| **פשטות** | **האם ניתן לפשט את הקוד?**  כן, הקוד מפושט למדי, אך ניתן היה לפשט יותר - למשל הוצאת קטעי קוד משוכפלים לפונקציה חיצונית שבה נוכל להשתמש במידת הצורך. |
| **מודולריות** | **האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)**  כן, הקוד מחולק ל-3 cells("קבצים"), בנוסף יש שימוש נרחב במחלוקות והמון פונקציות שעושות דברים ספציפיים. |
| **תבניות עיצוב** | **האם ישנו מימוש ברור ונכון של תבניות עיצוב?**  כן, נעשה שימוש ב-singleton בחלון הפופ אפ. |
| **OOP** | **האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?**  כמו שכבר הזכרתי נעשה שימוש ב-class ועבודה עם אובייקטים אך תמיד ניתן לשפר. |
| **באגים וטעויות** | **האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?**  לא נתקלתי במקרים כאלו. |
| **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?**  כן, ישנן הודעות המוצגות למשתמש במידה והאינפוט אינו תקין/נכון (לדוגמא שדות ריקים/ שאלה לא קיימת ב-update question).  כן, נעשה שימוש במנגנון try - except - לדוגמא כאשר מגיעים לשאלה האחרונה בעת המעבר על השאלות שמציגים למשתמש.  כן, כמו שכבר אמרתי הקוד מפושט למדי אז הוא יחסית נוח לבדיקה.  כן, הודעות המשתמש ברורות ואף חלקן הומוריסטיות. |
| **בדיקות** | **האם יש בדיקות שניתן להוסיף?**  כן, למשל ב-add question ניתן היה לבדוק אם השאלה כבר קיימת. |
| **שימושיות** | **האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?**  כן, הממשק די פשוט לתפעול, העיצוב של המסכים השונים קונסיסטנטי , ישנם פידבקים למשתמש ובגדול די נוח להתמצאות.  העיצוב, פידבקים, מוזיקה (במיוחד ההודעה על טעות בתשובה) נתנו לי מוטיבציה להמשיך לשחק. |
| **תיעוד** | **האם הממשק מתועד?**  בממשק קיים תיעוד מינימלי. |
| **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?**  אין שום שימוש בנתוני המשתמש אשר מפרים את פרטיותו.  כמו כן אין שום ניצול של חולשה אנושית/ algorithm bias כנגד קבוצה או פרטים מסוימים. |
| **אבטחה** | **האם ישנו מידע אבטחה (שמות משתמש, ססמאות) גלויים?**  שמות המשתמשים והסיסמאות גלויים אך ורק למנהל וגם זה אך ורק מתוקף תפקידו. |
| **ביצועים** | **האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?**  כן, לדעתי שינויים כגון הוספה של פיצ'רים יכולים לגרום להאטה של המשחק. |
| **ביצועים** | **האם ניתן לשפר לדעתך את הביצועים?**  כן, לדעתי אם נשתמש בסביבת עבודה אחרת ,המתאימה יותר לפרויקטים מסוג זה, ניתן לשפר את הביצועים. |
| **קריאות - readability** | **האם הקוד מובן בקלות? אילו חלקים בקוד היו לא ברורים עבורך? כיצד ניתן לשפר את קריאות הקוד?** כן, הקוד מובן די בקלות אך ארוך למדי ולכן לדעתי חלוקה ל-cells נוספים היו משפרים את קריאות הקוד. |
| **אתיקה** | **Does this code change introduce anygender/racial/political/religious/ableist bias?**  לא, בקוד אין שום אפליה (אפילו אין אזכור למושגים אלו או משהו שמתייחס אליהם). |
| **קריאות - readability** | **Is there any commented-out code**  כן, ישנה מחלקה שהיא  commented-out (ככל הנראה פיצ'ר שוויתרו עליו ושכחו למחוק). |

**שאלות פתוחות:**

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

בהתחלה היה לי די קל לקרוא את הקוד מכיוון שהוא מאוד מסודר ובנוסף מאוד עניין אותי לראות איך אנשים אחרים מימשו דברים בצורה שונה ממני, אבל מכיוון שהקוד ארוך לאחר זמן מה די איבדתי את הריכוז ולכן הייתי צריך הפסקה כדי לחזור לפוקוס.

1. **האם לדעתך ניתן לשפר תרגיל זה?**

כן, כמו שאמרתי בשאלה הקודמת מכיוון שהקוד ארוך ניתן היה אולי לתת רק חלק מהקוד לבדיקה.

כמו כן לדעתי חלק מהשאלות לא כל כך רלוונטיות/חוזרות על עצמן ולכן ניתן להוריד חלק מהקריטריונים שאנו בודקים.

1. **הערות נוספות**

אין.

**אילנית חנוכו 208657320**

1. **האם הקוד מבצע את הנדרש?**

כן, המשחק רץ והמעבר בין המסכים עובר בצורה חלקה.

1. **האם ניתן לשפר את יעילות הקוד?**

לדעתי הקוד כתוב בצורה יעילה. יש חלוקה למתודות נפרדות עבור פעולות קטנות יותר, יש מימוש של עקרונות מונחה עצמים כמו שימוש במחלקות. ממה ששמתי לב יש פונקציות שמקבלות המון פרמטרים, מה שיכול מעט לבלבל, אך באופן כללי הקוד כתוב בצורה יעילה.

1. **האם ניתן לפשט את הקוד?**

הקוד כתוב בצורה מופשטת ולא מסורבלת שכן כפי שנאמר בשאלה 2, מממש עקרונות של תכנות מונחה עצמים.

1. **האם הקוד מודולרי מספיק? (חלוקה תקינה לקבצים, פונקציות ומחלקות)**

כן, בקוד יש כמות גדולה של פונקציות כאשר לכל אחת תפקיד אחר, כמו גם מחלקות שהוגדרו בתחילת הקוד.

1. **האם ישנו מימוש ברור ונכון של תבניות עיצוב?**

כן, יש שימוש בתבנית עיצוב Singletone ליצירת חיבור יחיד לדאטה-בייס באמצעות המשתנה FIREBASE\_CONNECTION.

1. **האם ישנו שימוש נכון בעקרונות OOP? האם ניתן לשפר?**

כן, ישנה חלוקה למחלקות, חלוקה חכמה של ה-API לפונקציות קטנות.  
היה ניתן לממש עיקרון של הורשה ע"י זה שהמחלקה InvokeButtonWithPlace תירש מהמחלקה InvokeBackground, שכן הן חולקות מאפיינים ומתודות זהות, אך לראשונה מאפיינים נוספים.

1. **האם ישנם מקרים בהם הקוד לא מתנהג כצפוי? אילו?**

מאחר והמשחק כולל אלמנטים גרפיים רבים לוקח מעט זמן לטעון אובייקטים שונים כמו רקע הכפתור או חלונית הפופ-אפ המופיעה בעת מענה על שאלה.

1. **א. האם ישנה התראה על שגיאות משתמש?**

כן, למשל כאשר מזינים שם משתמש וסיסמה לא נכונים, קופצת התראה מתאימה.

**ב. האם ישנו שימוש במנגנון תפיסת שגיאות?**

כן, קיים בלוק try catch בקוד שתפקידו לעצור איטרטור מלרוץ מעבר לגבולותיו.

**ג. האם הבדיקה ודיבוג הקוד נוחים?**

הבדיקות שנכתבו כתובות בצורה ברורה ומסודרת (למשל מתי להציג את הודעת השגיאה שציינתי מקודם) לגבי דיבאגינג, גוגל קולאב הוא לא כלי נוח לדיבוג כמו ב-IDE קלאסי ולכן התשובה היא לא, ללא קשר לאיכות הקוד של הקבוצה.

**ד. האם הודעות המשתמש ברורות וידידותיות למשתמש?**

באופן אישי, לא התחברתי. מבחינה עיצובית, רקעי המטריקס האדומים/ הירוקין להודעות פופ-אפ עם קונטקסט שלילי/ חיובי (בהתאמה) מעט הקשו על קריאת הטקסט שנכתב בפונט לבן. מעבר לכך, תוכן ההודעה כאשר משתמש ענה לא נכון היה מעט לא נעים מאחר והיה שימוש במילה "noob" (טירון). ברור לי כי זו נכתבה בקונטקסט הומוריסטי, אך ישנם משתמשים שעשויים להיעלב.

1. **האם יש בדיקות שניתן להוסיף?**

ניתן להוסיף בדיקות על המעבר בין המסכים - לעטוף את הפקודה display בבלוק try-catch.

1. **האם הממשק שימושי? (usability)? האם המשחק מעורר רצון להמשיך לשחק?**

כן, ראשית הוא משרת את המטרה לה נועד והיא למידה בצורה חווייתית. מעבר לכך, האלמנטים הגרפיים והמעט מפחידים עושים חשק להישאר ולגלות מה עוד עשוי להיחשף במהלך המשחק.

1. **האם הממשק מתועד?**

קיים תיעוד חלקי. באופן כללי שמות המשתנים והפונקציות ברורים ואין צורך לבצע refactoring משמעותי.

1. **א. האם יש שימוש בנתוני משתמש אשר מפרים פרטיותו?**

לא. המשתמש לא מזין פרטים אישיים מעבר לשמו הפרטי, שם משתמש וסיסמה הנשמרים בדאטה בייס.

**ב. האם יש ניצול של התנהגות או חולשה אנושית?**

לא. המשחק אמנם מעט מפחיד מבחינת אלמנטים גרפיים, אך ניצול אינו חלק ממנו.

**ג. האם ישנו algorithmic bias  כנגד קבוצה או פרטים מסוימים?**

לא. האלגוריתמיקה בפרויקט בסיסית ולא מכוונת להעדפה מסוימת של קבוצות כמו מחלקות/ אובייקטים על פני אחרים.

1. **האם ישנו מידע אבטחה (שמות משתמש, סיסמאות) גלויים?**

לא, בעת הקלדת סיסמה במסך ההתחברות הסיסמה לא חשופה לעיני כל מאחר ונכתבת ב-Password field. הסיסמאות במלואן חשופות רק במסד הנתונים, אליו יש גישה רק למנהלי המשחק.

1. **האם לדעתך שינוי עתידי בקוד יכול לגרום לפגיעה בביצועים?**

מאחר והקוד כתוב בצורה מודולרית ומסודרת, אי-אלו שינויים יוכלו להשתלב בצורה טובה. כמובן שבאופן טבעי עשויות לצוץ בעיות כמו בכל אינטגרציה אך הבסיס טוב וקל להבנה.

1. **האם ניתן לשפר לדעתך את הביצועים?**

בהחלט, ע"י בחירת עיצוב פשוט יותר שעדיין משרת את המטרה. נכון, עיצוב הוא חלק גדול בחוויית המשתמש אך כאשר לוקח זמן לתמונות לעלות במהלך ריצת המשחק זה עשוי לגרום לחוסר סבלנות אצל המשתמש. בכל אופן אני מאמינה כי גם לסביבת העבודה יש השפעה על רמת הביצועים, ובמידה והיינו עובדים על סביבה מקומית (ולא עננית) הביצועים היו עולים על-אף השימוש האינטנסיבי באלמנטים גרפיים.

1. **א. האם הקוד מובן בקלות?**

כן.

**ב. אילו חלקים בקוד היו לא ברורים עבורך?**

מאחר ולא למדתי פייתון במסגרת הלימודים, אני חווה מעט פערי ידע בשפה, ולכן החלק שכולל כתיבת קוד מונחה עצמים היה מעט יותר מאתגר מבחינת הקונספט (רגילה לכתוב בג'אווה).

**ג. כיצד ניתן לשפר את קריאות הקוד?**

בעיניי יש צורך בעוד תיעוד.

1. **סעיף פתוח - קטגוריית בדיקה חדשה**

בלחיצה על כפתור "My Progress" מופיע גרף המתאר את כמות הנקודות שנצברו לשחקן בכל קטגוריה. נבדוק האם לחיצה על האיזור בגרף שמכסה את החלק במסך הקודם בו מוקמו הכפתורים, לא גוררת מעבר למסך אחר.

הבדיקה לא עברה מאחר וכאשר מרחפים מעל האיזור בו מכוסה הכפתור "Play" סמן העכבר משתנה לסמן המגדיר לחיצה, ובלחיצה בפועל מועברים למסך בחירת הנושא.

1. **סעיף פתוח - קטגוריית בדיקה חדשה**

נבדוק האם יציאה פתאומית ממשחק לא גוררת שמירת נקודות (שכן צבירת הנקודות תלוייה בהשלמת מענה על כל השאלות).

בלחיצה על "My Progress" בדקתי את מצב הנקודות בקטגוריית Json והוא 190.  
התחלתי משחק חדש בקטגורייה זו, וצברתי 10 נקודות ע"י מענה נכון על שאלה. לחצתי על כפתור היציאה. בדקתי שוב ב"My Progress" את מצב הנקודות וראיתי כי לא השתנה ועדיין עומד על 190, ולכן הבדיקה עברה.

**שאלות פתוחות:**

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

הייתי צריכה מעט להסתגל שכן עליי היה לתפוס את הקוד באופן ביקורתי יותר ולנסות להבין את כוונת המשורר גם עם מעט תיעוד.

1. **האם לדעתך ניתן לשפר תרגיל זה?**

לדעתי הבדיקות שביצענו לקבוצה השנייה היו צריכות להתבצע עם הבדיקות שהם כתבו.

1. **הערות נוספות**

אין. תודה רבה על פרויקט מעניין וחוויתי.