
ארץ עיר

מסמך עיצוב

דביר אלפי

גרסה 1.3

22.06.2017

היסטוריית גרסאות המסמך

תאריך	גרסה	תקציר השינויים
04\16	1.1	תכנית בצורת אלגוריתם. יש קשר בין השרת ללקוח אחד בלבד.
05\18	1.2	יש קשר בין השרת לכמות לקוחות שהוא בוחר.. הקשר עובד רק לפעמים.(באמצעות Threading)
06\19	1.3	שינוי שיטת הקשר מ Threading ל Select .. הפעם יש קשר יציב בין השרת לכמות הלקוחות שאחד מן הלקוחות בוחר (את כמות השחקנים שישחקו אתו במשחק)
-	-	-



הקדמה

המסמך בא להציג את הצורה הלוגית ואת הדרך בה כתבתי את המוצר שלי ברמה של פונקציות, תכונות.

המוצר נקרא ארץ עיר .

ע"ש המשחק המוכר והידוע ארץ עיר , שבו על השחקן להתאים מילים שמתחילות באות מסוימת לקטגוריות . השחקן מקבל אות רנדומלית ועליו, בזמן מוגדר, לענות על כל קטגוריה בתשובה מתאימה שמתחילה באות שניתנה לו.

במסמך זה אתאר את מבנה המערכת, פירוטים שונים והסברים מפורטים על המוצר שפיתחתי.

קישור למסמך האפיון :

<https://docs.google.com/document/d/1HTxa9fDVeCxUXdAstdzHRTTdMVYKVjM327ldyZ5ITdQ/edit>

חבילה כלולה בפיתוח שמכילה אפשרויות גרפיקה - *PyGame*.

תיקיה נדרשת להפעלת המוצר : "media" ובה כל קבצי המדיה הנדרשים.

ארכיטקטורת המערכת

חלק זה כולל את תיאור מבנה המערכת ופירוט המודולים השונים בה.

מבט על

החלוקה מתבצעת ע"י שרת ולקוח, שמתחלקים בקבצים שונים. השרת מנהל את המשחק, ומעביר מידע ממנו ללקוחות ומלקוח ללקוח. ממשק המשתמש מתבצע ע"י Pygame. המוצר מתחלק ל-3 קבצים שונים. קובץ השרת שבו יש מימוש של השרת ע"י Socket ו Select ו Thread. החלטתי לשלב בין Select ו Thread, משיקול דעת של Blocking באמצע בדיקה של קלט מן המשתמש, לבין דרישה של Blocking, בטיחות, והגעה שלימה ובטוחה של המידע בין השרת והלקוחות. יש בחלק הלקוח של המוצר לולאות ריצה While שמקבלות במקרה הצורך תווים ופעולות מן המשתמש.

צד הלקוח:

לולאת ה WHILE של המשחק עצמו :

הלולאה רצה ומקבלת קלט מהמשתמש, ומסתיימת רק במצב של סיום הזמן המוקצב, קבלת אנטר מהמשתמש, או, במקרה של משחק רב-משתתפים, קבלה של GAME OVER מהשרת.

הפונקציה `get_and_update_answer_from_player` :

מקבלת קלטים מתאימים מהמשתמש, ומעדכנת את המצב הגרפי ומצב התשובה לקטגוריה העדכנית לאור השינויים מפאת הפעולות של המשתמש. הפונקציה מקבלת את רשימת האובייקטים של הקטגוריות (יוסבר למטה) ואובייקט עדכני, לפי מקום סמן המקלדת. (המשתמש יכול לשנות את מקום הכתיבה ע"י לחיצה על מקשי החיצים (למעלה ולמטה) או לחיצה על תיבת הטקסט באמצעות העכבר).

מחלקת Category :

מחלקה שמגדירה אובייקט שמכיל תכונות: ערכי X Y של התשובה שרשם המשתמש, ערכי X Y של תיבת הטקסט בו רשומה התשובה, והתשובה לקטגוריה המתאימה. לכל תכונה יש פונקציית get, ולתכונת התשובה יש פונקציית set, מכיוון שכל פעם שהמשתמש מעדכן את תשובתו צריך לעדכן זאת באובייקט.

בחירות בשיטות העבודה

החלטתי להשתמש בשפת התכנות Python מכיוון שהיא יותר ידידותית למשתמש, ואני מרגיש נוח לכתוב בה. החלטתי ליצור קשר בין הלקוח לשרת באמצעות חיבור TCP כדי שהחיבור והמידע שיועבר יהיה יותר בטיחותי, לא כמו בחיבור UDP.



הצדדים השונים מורכבים מפונקציות שונות ואפרט את העיקריות שבהן.

צד השרת:

GetPlayerDetailsAndSendLetterThread - מחלקה שיורשת מ Threading :

מחלקה זו בונה Thread שמקבל את חיבורו של הלקוח, מתוך כך הוא מקבל את פרטי הלקוח (כולל אובייקט ה Socket המחובר בינו לבין השרת) ומכניס אותם למילון פרטי השחקנים.

ReceivingPlayerResultsThread עוד מחלקה דומה, שיורשת מ Threading ובונה thread שמקבל את תוצאות הלקוח.

send_to_players

פונקציה המקבלת טקסט ורשימת פרטי השחקנים (כולל אובייקט ה Socket המחובר בין הלקוח לשרת), ושולחת את הטקסט לכל לקוח ולקוח המחובר לשרת (דרך אובייקט ה Socket).
פונקציה זו משומשת במקרה שאחד השחקנים לחץ על Enter, ויש צורך לשלוח לכל השחקנים הודעת "Game Over", או בתחילת המשחק שהשרת שולח לכל השחקנים את האות הרנדומלית(האקראית).

get_random_letter

פונקציה המחזירה אות אקראית (אות קטנה) מהאלפבית האנגלי.
האות נשלחת לשחקנים והם משחקים את המשחק לפיה.

main

הפונקציה הראשית המנהלת את פעילות השרת.
הפונקציה מדפיסה עדכונים בפעולות השרת, לקבלת משוב על פעולותיו.

הצד הגרפי:

הצד הגרפי כולל בתוכו קבועים של כתובות קבצי המדיה, ערכים של צבעים(RGB), ועוד קבועי טקסט וערכי מספרים חשובים.

הוא כולל גם פונקציות עזר גרפיות, שלרוב מתחילות במילה ... draw_something.
כל פונקציה כזו מקבלת את אובייקט ה Display, ומעדכנות/מציירות על המסך תכנים מתאימים.
צד הלקוח משתמש בחלק הגרפי בהתאם למתרחש בצד שלו (בצד הלקוח, המושפע מהשתמש).
בשפת Python, כדי שקובץ יוכל להשתמש בפונקציות ובקבועים של קובץ אחר, הוא צריך לעשות לו import.

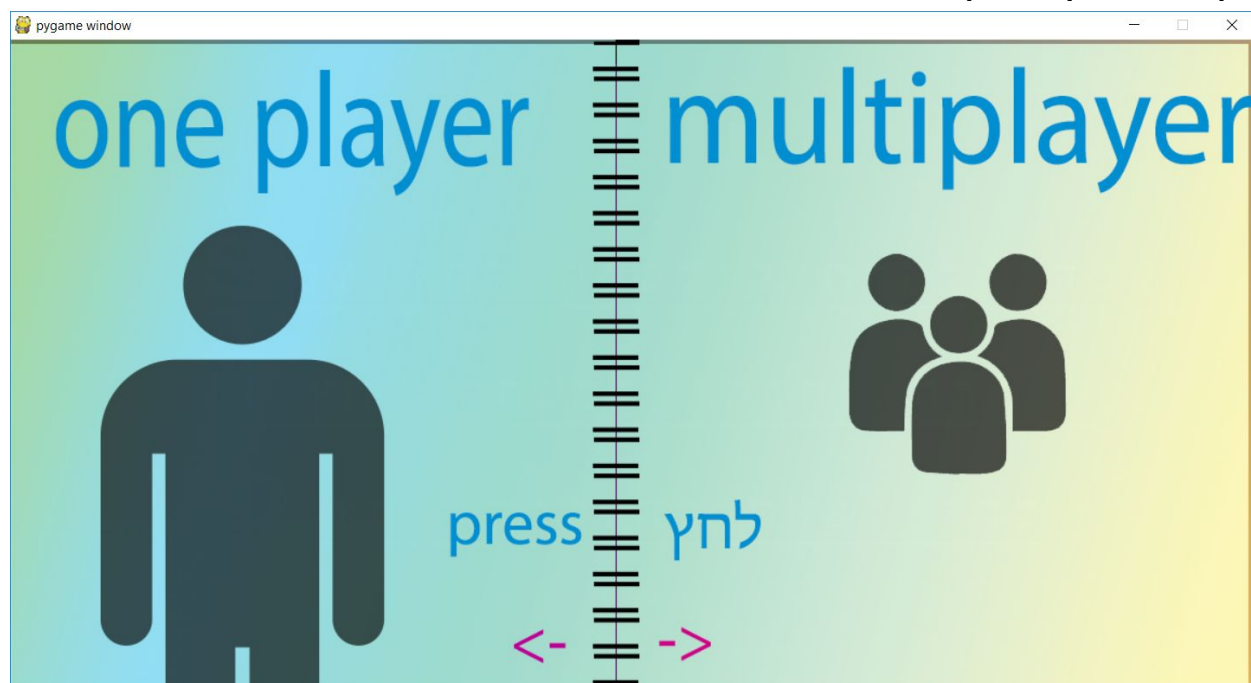
ממשק משתמש

המסך הגרפי עצמו בגודל 600*1152
מסך הפתיחה : (המשתמש לוחץ על כל מקש כדי להמשיך.)

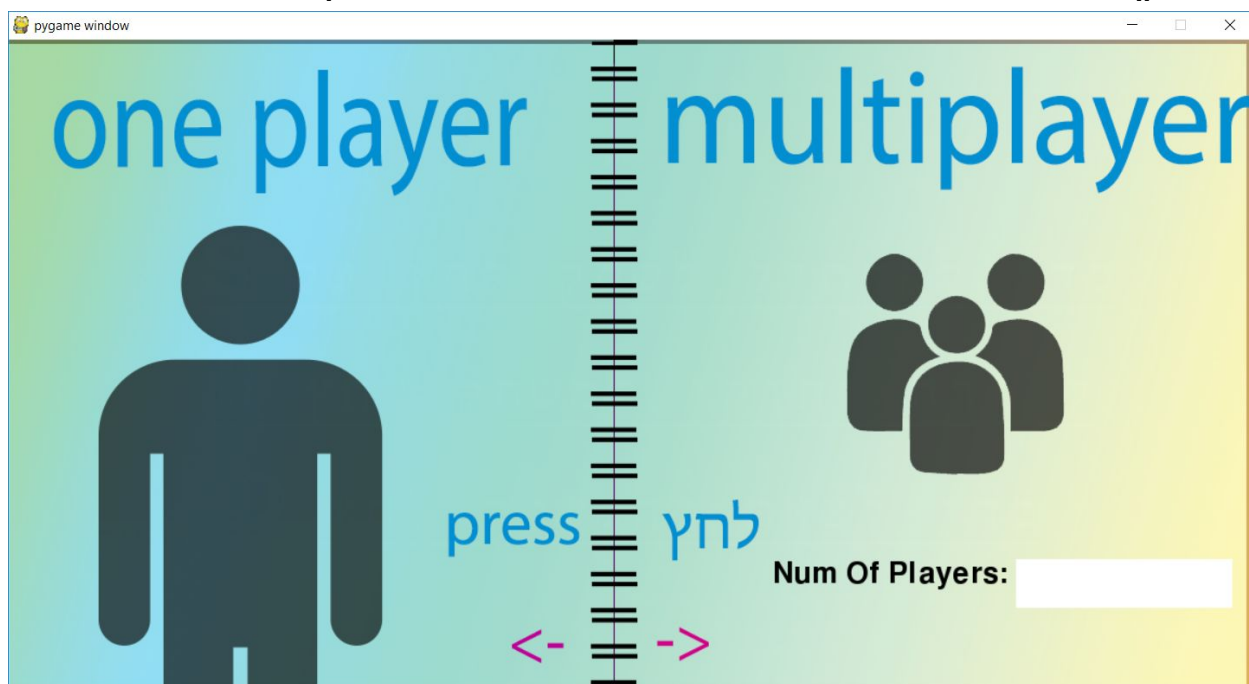


מסך התפריט:





אם השחקן בחר רב משתתפים, באמצעות לחיצה על ימינה, בחירת מספר השחקנים:



במקרה שלא כל השחקנים הצפויים לשחק התחברו, השחקנים שהתחברו רואים מסך המתנה:



מסך המשחק עצמו :

השחקן מזין תשובות בתוך תיבות הטקסט, ויכול להעביר בין תיבות טקסט שונות באמצעות לחיצה על תיבת הטקסט בעכבר, או באמצעות לחיצה על מקשי החיצים (למטה ולמעלה).

pygame window

Time: 35.17

אָרֵץ עִיר

האות: d

Boy Name: dan

Girl Name: dana

Animal Name: donkey

Country Name: bla bla bla

כדי לסיים, לחץ אנטר!

לאחר לחיצה על אנטר: (הזמן נעצר, ויש ואים בהתאם לתשובות.)

pygame window

Time: 28.12

אָרֵץ עִיר

האות: d

Boy Name: dan

Girl Name: dana

Animal Name: donkey

Country Name: bla bla bla

כדי לסיים, לחץ אנטר!

מסך התוצאות:

