פרויקט בהנדסת תוכנה התמחות בהגנת סייבר  
סמל מקצוע: 883589



**משחק ניחוש באמצעות ציור רב משתתפים**

נושא העבודה: Multiplayer Drawing And Guessing Game

שם התלמיד: רון אייזן

תעודת זהות: 213412422

בית ספר: ראשון לציון, מקיף ה' אמירים.

שם המנחה: מוטי מתתיהו

מועד הגשה: מאי 2021

[מסמך ייזום 3](#_Toc62808722)

[מבוא ורקע כללי לנושא העבודה 4](#_Toc62808723)

[נושא הפרויקט 4](#_Toc62808724)

[תקציר הפרויקט 4](#_Toc62808725)

[הגדרות 4](#_Toc62808726)

[מטרת המערכת ויעדיה 5](#_Toc62808727)

[ארכיטקטורת המערכת 5](#_Toc62808728)

[הגדרת רכיבי המערכת 6](#_Toc62808729)

[תיחום המערכת 6](#_Toc62808730)

[אילוצים ודרישות: 7](#_Toc62808731)

[פירוט סביבת העבודה והכלים הנדרשים לפיתוח: 7](#_Toc62808732)

[תיאור המוצר המוגמר: 8](#_Toc62808733)

[ניהול פרויקט עתידי: 8](#_Toc62808734)

[מסמך אפיון 9](#_Toc62808735)

[פונקציונליות המערכת 10](#_Toc62808736)

[דרישות בסיס נתונים 12](#_Toc62808737)

[סביבת עבודה 12](#_Toc62808738)

[מסמך עיצוב 13](#_Toc62808739)

[תיאור בסיס מערכת הנתונים במערכת 14](#_Toc62808740)

[שדות בסיס הנתונים 14](#_Toc62808741)

[הסבר השדות 14](#_Toc62808742)

[מודולים 15](#_Toc62808743)

[ביבליוגרפיה 15](#_Toc62808744)

מסמך ייזום

מבוא ורקע כללי לנושא העבודה

בחרתי לעשות את הפרויקט הזה מפני שלדעתי, הוא שילוב מעניין של החומר שנלמד במהלך

השנים- רשתות ומערכות הפעלה. הפרויקט משלב מספר קונספטים שונים שנלמדו במהלך

השנה האחרונה והשנים שקדמו לה. בנוסף, בסוף הפרויקט יוצא מוצר שמיש וייחודי שגם

מעניין מבחינת הנושא שלו. את הרעיון לפרויקט קיבלתי מהאפליקציה skribbl.io שהיא אפליקציה אינטרנטית חינמית בעלת רעיון דומה לרעיון עליו מתבסס הפרויקט שלי. כמו כן, בעת חיפוש רעיון לפרויקט תחילה בחרתי ברעיון "עוזרת אישית קולית", אך לאחר מכן התברר כי רעיון זה נתפס על ידי תלמיד אחר בכיתה, אז החלטתי להתמקד בתחום המשחקים כך שהמשחק יכיל את כל דרישות החובה שרצויות בפרויקט. לבסוף חשבתי על רעיון הדומה לאפליקציית skribbl.io, ונעזרתי בה לרעיונות עיצוביים ותוכניים עבור הפרויקט שלי.

נושא הפרויקט

משחק ניחוש באמצעות ציור, רב משתתפים.

תקציר הפרויקט

הפרויקט הוא משחק רב משתתפים, בו כל משתמש נרשם באמצעות יצירת שחקן חדש או שמתחבר לשחקן קיים ובאחת מהדרכים נכנס למאגר המשתמשים, משתמש אחד נבחר מבין שאר המשתתפים ומוצגת בפניו באופן פרטי (ולא בפני שאר המשתתפים) מילה ממאגר מידע מוכן של פריטים, חפצים, בעלי חיים, מקצועות ועוד... ועליו להביע את המילה על ידי ציורה על הלוח בעוד שעל שאר המשתתפים לנחש את המילה על ידי כתיבתה בצ'אט.

הגדרות

משתמשי המערכת

המערכת מיודעת לכל אדם עם מחשב המעוניין לשחק במשחק ניחוש מהנה.

מטרת המערכת ויעדיה

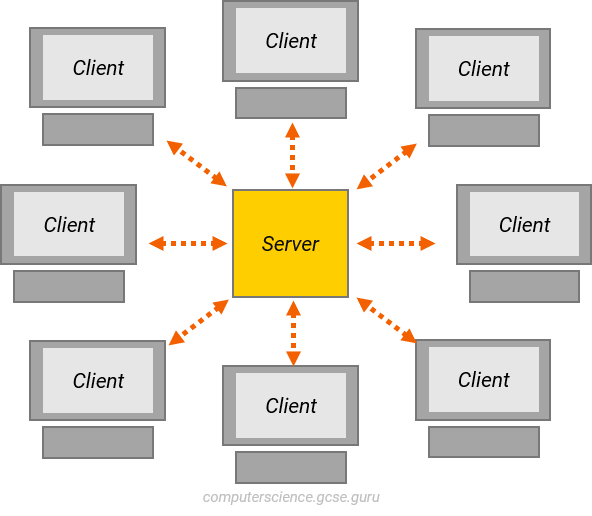
מטרת המערכת היא לתת למשתמש חווית משחק נוחה ומהנה, שליטה ובקרה נוחה.

במשחק זה רוכז רעיון מהנה וייחודי, ונעטף בממשק גרפי נוח, ברור וקל לתפעול.

בנוסף היא גם מאפשרת משחק רב משתתפים, למשחק בחברותא.

ארכיטקטורת המערכת

המערכת נבנתה בארכיטקטורת שרת-לקוח. המערכת תומכת במספר רב של לקוחות, תוך תמיכה בריבוי תהליכים במקביל- הן מספר תהליכים שונים על אותו לקוח והן מספר תהליכים שונים בין לקוחות שונים. המערכת מטפלת בכל לקוח בעזרת תהליך (THREAD) שונה.

**תרשים מספר 1: ארכיטקטורת המערכת – שרת לקוח**

מקור התמונה: <https://www.computerscience.gcse.guru/theory/client-server-networks>

הגדרת רכיבי המערכת

1. רשת מקומית- (Network Area Local- LAN) רשת מקומית. רשת מחשבים המתפרסת על אזור קטן ומוגבל, לרוב בתוך בניין.
2. שרת- (server) ממנו מופעלות הפעולות השונות של הלקוחות, מודל אשר מדומה לצינור המחבר על ידי IP ו Port לשני מכשירים שונים ומאפשר להעביר [חבילות נתונים](https://he.wikipedia.org/wiki/%D7%97%D7%91%D7%99%D7%9C%D7%AA_%D7%9E%D7%99%D7%93%D7%A2) (data packets) מאחד לשני .
3. לקוח- (client) מחשב של משתמש המחובר למערכת.
4. :Client-Server-model [ארכיטקטורת תוכנה](https://he.wikipedia.org/wiki/%D7%90%D7%A8%D7%9B%D7%99%D7%98%D7%A7%D7%98%D7%95%D7%A8%D7%AA_%D7%AA%D7%95%D7%9B%D7%A0%D7%94) ל[חישוב מבוזר](https://he.wikipedia.org/wiki/%D7%97%D7%99%D7%A9%D7%95%D7%91_%D7%9E%D7%91%D7%95%D7%96%D7%A8), אשר מגדירה את היחס בין [תוכנות](https://he.wikipedia.org/wiki/%D7%AA%D7%95%D7%9B%D7%A0%D7%94) משתפות פעולה. המודל מחלק את המשימות או עומס העבודה בין ספק השירות או המשאבים - ה[שרת](https://he.wikipedia.org/wiki/%D7%A9%D7%A8%D7%AA_(%D7%9E%D7%97%D7%A9%D7%91%D7%99%D7%9D)), לבין מבקש השירות – הלקוח.

תיחום המערכת

הפרויקט עוסק ומשלב בתוכו תחומים רבים וביניהם:

1. תקשורת- נעשית תקשורת בין מספר מחשבים (3-5) שרת ו2-4 לקוחות (כמספר המשתתפים).  
2. בסיס נתונים- נעשה שימוש בבסיסי נתונים למאגר שמות השחקנים במשחק.  
3. אבטחת מידע- הצפנת מידע רגיש של השחקן במשחק, לדוגמה סיסמת השחקן.  
4.חלק ויזואלי (GUI)- יצירת ממשק גרפי בעל מקשים נוח וידידותי למשתמש.

המודולים ותתי הנושאים בהם אני משתמש בפרויקט:

1. בתחום התקשורת נעשה שימוש במודולים socket וthreads כשאר שניהם מאפשרים יחד תקשורת בין שרת למספר לקוחות.
2. בתחום בסיס הנתונים נעשה שימוש במסד הנתונים SQlite.
3. הצפנת המידע תעשה באמצעות הצפנת RSA.
4. יצירת הממשק הגרפי תעשה על ידי שימוש במודול tkinter.

אילוצים ודרישות:

1. על המערכת לעבוד על מחשב שמערכת ההפעלה שלו היא WINDOWS.
2. לוח זמנים צפוף, מתכוון לבצע שני מועדי פסיכומטרי בעת תקופת ביצוע הפרויקט.
3. קהל היעד הוא עבור ילדים בגילאים 8-18.

פירוט סביבת העבודה והכלים הנדרשים לפיתוח:

שפת התכנות: Python 3.7.3

סביבת העבודה: WINDOWS. ו- PYCHARM

כלים הנדרשים לפיתוח:

1. בתחום התקשורת נעשה שימוש במודולים socket וthreads כשאר שניהם מאפשרים יחד תקשורת בין שרת למספר לקוחות.
2. בתחום בסיס הנתונים נעשה שימוש במסד הנתונים SQlite.
3. אימות דו שלבי, הצפנת המדע תעשה באמצעות הצפנת RSA.
4. יצירת הממשק הגרפי תעשה על ידי שימוש במודול tkinter.

הכלים הנדרשים לבדיקה וסיוע בפיתוח:

1. מספר מחשבים על מנת לבדוק את תחום התקשורת.
2. תיעוד והסבר בקוד מה מקבלת כל פונקציה או כל אובייקט ומתי הלקוח שולח מידע לשת ומתי השרת אל הלקוח וכו'...
3. תכנות מונחה עצמים (OOP).

תיאור המוצר המוגמר:

שם המוצר: Draw & Guess.  
מה המוצר המוגמר אמור לבצע: על המוצר המוגמר להיות משחק רב משתתפים של בין 2 ל4 שחקנים, שבעבור כל משתתף מוצג לוח הציור של אותו השחק שפניו מוצגת המילה.  
מטרות מרכזיות:

1. חשיפה למודולים חדשים בשפת פייתון.
2. התנסות ראשונית ביצירת פרויקט בהיקף של חמש יח"ל.
3. העמקה והרחבת הידע בתחום הסייבר והצפנת המידע.
4. התנסות בתכנות עם בסיסי נתונים.
5. יצירת ממשק גרפי (GUI).

המערכת עונה על **כל** המטרות המרכזיות שציינתי וניתן לראות זאת בהרחבה בחלק תיחום הפרויקט.

ניהול פרויקט עתידי:

**תרשים מספר 2: טבלת ניהול פרויקט עתידי**

|  |  |
| --- | --- |
| יעד | תאריך יעד |
| קישור בין שרת ולקוח ויצירת בסיס נתונים מלא עם מילים, ובסיס נתונים ריק של שחקנים. | 10.11.20 |
| קישור בין בסיסי הנתונים לשרת.  יצירת פעולה שמקבלת את המילה מהשרת | 30.11.20 |
| פעולה ששולחת את הניחוש לשרת | 15.12.20 |
| הוספת ממשק גרפי פעולה שבאמצעותה מצייר הלקוח את המילה שקיבל. | 15.1.21 |
| שיפורים/ תיקונים אחרונים | 1.4.21 |

מסמך אפיון

פונקציונליות המערכת

פעולות רקע:

1 .חיבור כל המשתמשים לשרת.

2 .שמירת נתוני המשתמשים.

3 .פתיחת מסך הפתיחה.

פעולות השרת:

1. שידור מסכים בזמן אמת.
2. שידור הציור לכלל המשתמשים.
3. שידור הניחושים לשאר המשתתפים.
4. קבלת ניחושים משאר המשתתפים.
5. שליחת הניקוד ושמירתו במאגר הנתונים.
6. ושליחת המילה שהוגרלה ממאגר הנתונים.

פעולת הלקוח:

1. התחברות לשרת.
2. שליחת הניחוש לשרת.
3. מסך פתיחה.
   1. לחצן עם סימן שאלה, מכיל מידע אודות המשחק.
   2. לחצן להרשמה/ לחצן לרישום למשחק.
4. הרשמה ראשונית למשחק.
   1. הכנסת שם משתמש, אימייל, סיסמה ואימות סיסמה.
5. כניסה למשחק עם משתמש קיים (log in).
   1. הכנסת שם משתמש וסיסמה.
6. המשחק עצמו.
   1. המשתמש שתורו לצייר מקבל מילה המוגרלת ממאגר הנתונים.
   2. שאר המשתמשים מנחשים את המילה, ורושמים את ניחושם בצ'אט.
      1. במישה והניחוש נכון, השחקן מקבל הודעה כי ניחש נכונה.
      2. אחרת, מנחש שוב, מספר הניחושים לא יעלה על 3.
   3. כאשר הסיבוב נגמר, תורו של השחקן הבא לקבל מילה ממאגר הנתונים ולצייר אותה על הלוח.
   4. לאחר שמסתיים סיבוב שלם, כל חברי הקבוצה שיחקו, מוצגת טבלת הניקוד של כל המשתתפים והשרת שואל את הלקוחות אם ברצונם לשחק משחק נוסף, במידה ויש הסכמה של כל חברי הקבוצה ישנו משחק נוסף, במידה ולא חוזרים למסך הבית.

דרישות פונקציונליות :

**תרשים מספר 3: דרישות פונקציונליות**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| # | דרישה | | | תהליך | קלט | | | פלט | טיפול בשגיאות | | |
| 1 | התחברות לserver | | | המשתמש תחבר לserver של המערכת על ידי socket. | Portו IP של ה server | | | התחברת למערכת | הודעת שגיאה : נסה להתחבר למערכת מחדש. | | |
| 2 | | הרשמה ראשונית למערכת | המשתמש יכניס את פרטיו יבצע בדיקת סיסמא ולאחר מכן פרטיו ישמרו בשדה הנתונים . | | | פרטי המשתמש (שם משתמש, סיסמא, אמייל) | הלקוח נרשם למערכת בהצלחה. | | | הודעה שגיאה :  - אמייל או שם משתמש שגוי, סיסמה קצרה מדי. |
| 3 | התחברות למשתמש קיים | | | המשתמש יכניס את שם המשתמש וסיסמתו אשר יבדקו בשדה הנתונים ולאחר מכן יכנס למערכת . | שם המשתמש והסיסמא של המשתמש. | | | התחברת למערכת בהצלחה. | במקרה שהמשתמש יכניס פרטים שגויים הוא יתבקש לחזור על הפעולה | | |
| 4 | שינוי פרטי משתמש | | | המשתמש יכניס את פרטי המשתמש החדשים , אשר יעדכנו בשדה הנתונים . | פרטי המשתמש החדשים | | | פרטיך עודכנו בהצלחה. | הודעה שגיאה :  - אימייל או שם משתמש שגוי, סיסמה קצרה מדי. | | |
| 5 | שליחת מילה ממאגר הנתונים ללקוח | | | השרת מגריל מילה ממאגר הנתונים, ושולח אותה ללקוח באמצעות socket | - | | | המילה המוגרלת | - | | |
| 6 | שליחת הניחוש לשרת | | | הלקוח שולח את הניחוש לשרת באמצעות socket | הניחוש | | | - | - | | |

דרישות בסיס נתונים

בסיס הנתונים ימומש בשפת – SQL וישמרו בו נתוני המשתמש, ונתוני הקבצים ששותפו .

סביבת עבודה

שפת פרויקט – . Python 3.7

מערכת הפעלה – WINDOWS.

ממשק גרפי – Tkinter .

בסיס הנתונים - SQLite.

PyCharm סביבת עבודה - לכתיבת הקוד, Debugging , והרצה.

מסמך עיצוב

תיאור בסיס מערכת הנתונים במערכת

-Users.dbמאגר נתונים שמכיל את פרטי המשתמשים במשחק. השרת מתחבר אל מאגר הנתונים, מוסיף אליו משתתפים, בודק אם משתתפים קיימים, מקבל הרשאה להכנסת משתתפים רשומים כלומר יש לו גישה מלאה בכל הקשור למאגר הנתונים.

שדות בסיס הנתונים

**תרשים מספר 4: שדות בסיס הנתונים**

|  |  |
| --- | --- |
| Type | Field Name |
| TEXT | username |
| TEXT | password |
| TEXT | email |

הסבר השדות

Username – שם המשתמש, מסוג טקסט, מפתח ראשי, חייב להיות ייחודי (כלומר לכל משתמש יש שם ואין כפילויות), לא יכול להיות ריק.

Password – סיסמת המשתמש, מסוג טקסט, לא חייבת להיות ייחודית (כלומר יכול לקרות מצב בו למספר משתמשים ישנה אותה הסיסמה), לא יכולה להיות ריקה.

Email – הדואר האלקטרוני של המשתמש, מסוג טקסט, חייב להיות ייחודי (כלומר לכל משתמש יש דוא"ל ואין כפילויות), לא יכול להיות ריק.

**תרשים מספר 5: דוגמה לבסיס נתונים**

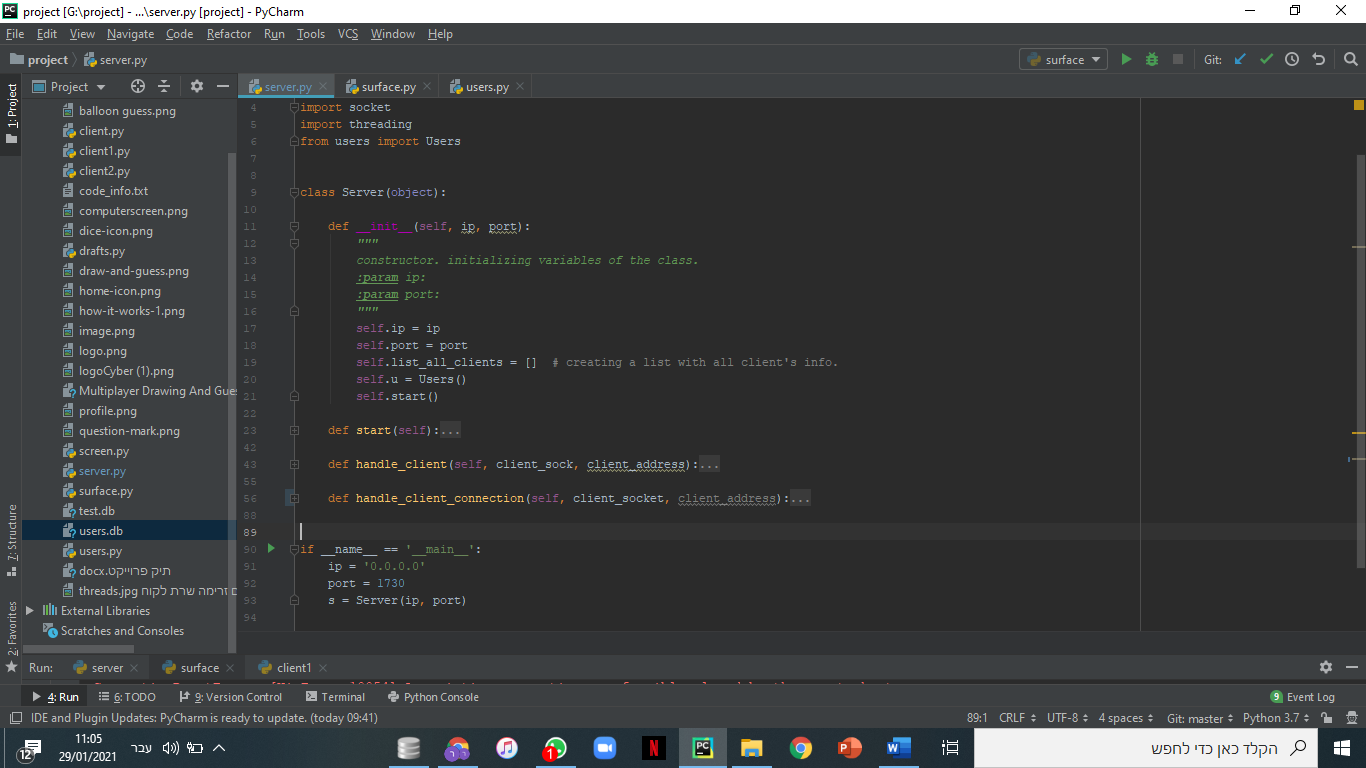
|  |  |  |
| --- | --- | --- |
| email | password | username |
| ronaizen@gmail.com | Hello123 | Ron |
| jasmin@gamil.com | Papaya12 | Jasmin |
| Moti@outlook.com | Hello123 | Moti |
| yarin@walla.com | Noder2424 | Yarin |
| hadas@yahoo.com | H2905 | Hadas |

מודולים

* Server.py- הקובץ הראשי והעיקרי במחשב. כולל את השרת, את פעולותיו, את התקשורת שלו עם הלקוחות וכו'..
* Surface.py- הקובץ המכיל בתוכו את הממשק הגרפי ואת הלקוח. הקובץ מתחבר לשרת באמצעות חיבור socket.
* Screen.py- הקובץ המכיל בתוכו את הקוד שיוצר את המסך עליו מצייר המשתמש.
* Users.py- הקובץ שפונה אל מאגר הנתונים, מכניס משתמשים, שולף משתמשים וכו'...
* Users.db- הקובץ שמכיל את מאגר המשתמשים.

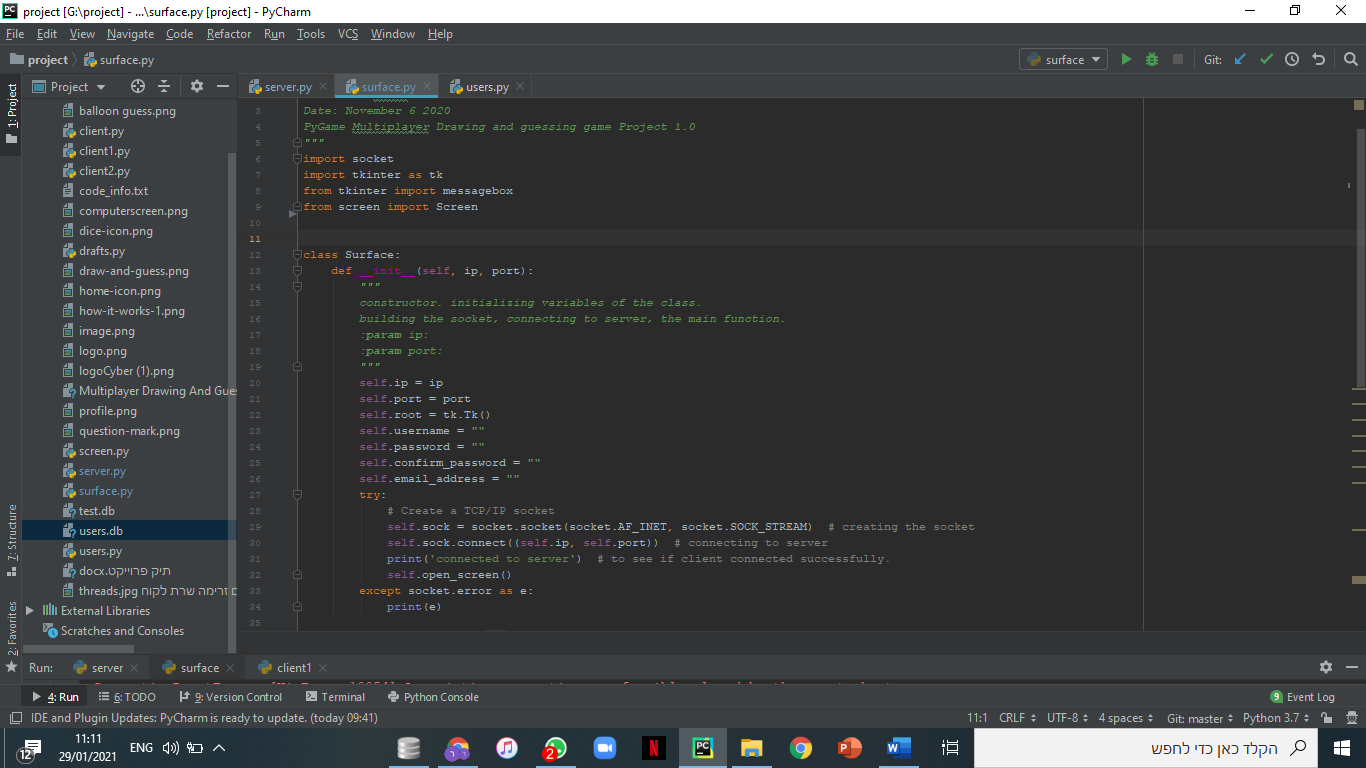
Server

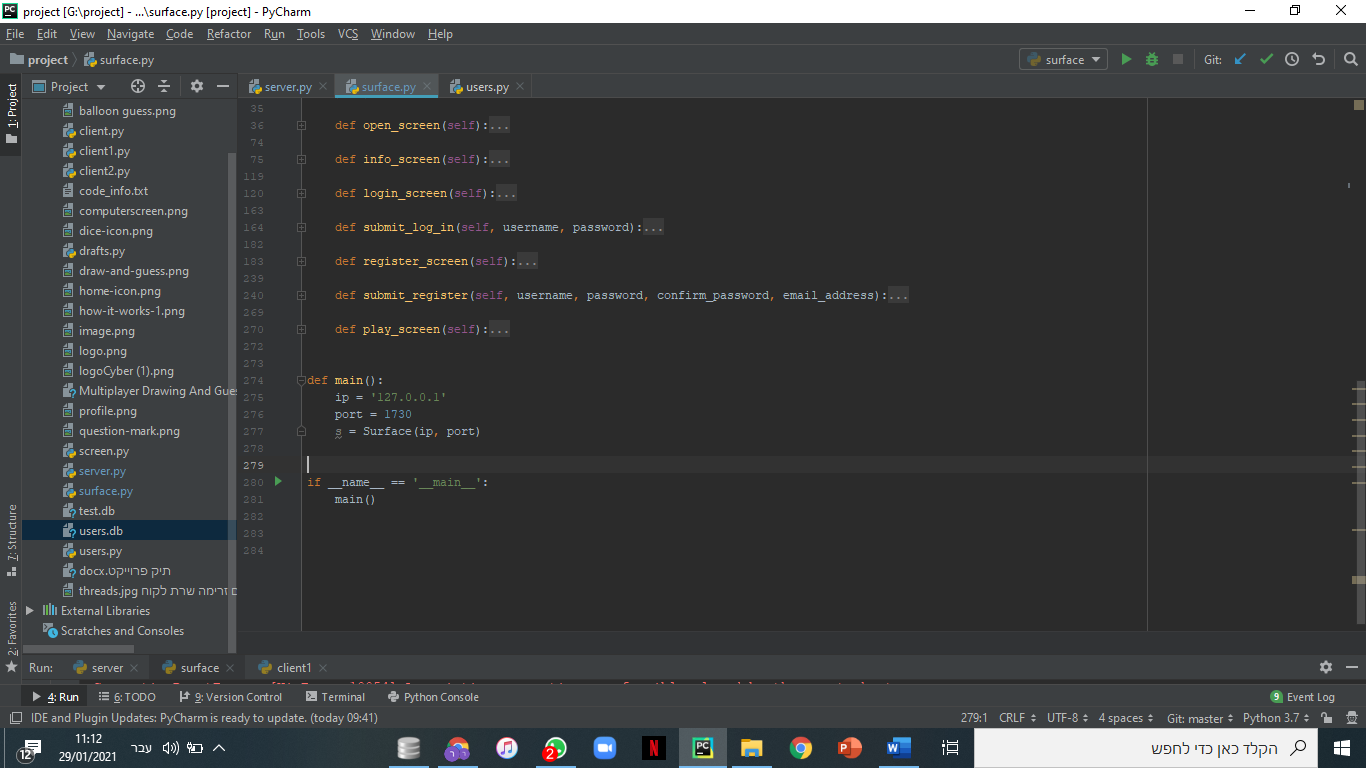
**תרשים מספר 5: דוגמה לבסיס נתונים**



המחלקה server מייצגת, כמובן, את השרת והפעולות השונות שלו. מחלקה זו מכילה את  התכונות השונות של השרת, ואת האופן בו הוא מטפל בלקוחות המתחברים או מתנתקים  ממנו.

Surface





ביבליוגרפיה

* [תהליכונים](https://he.wikipedia.org/wiki/%D7%AA%D7%94%D7%9C%D7%99%D7%9B%D7%95%D7%9F)
* [תקשורת שרת לקוח](https://he.wikipedia.org/wiki/%D7%A9%D7%A7%D7%A2_(%D7%AA%D7%A7%D7%A9%D7%95%D7%A8%D7%AA_%D7%9E%D7%97%D7%A9%D7%91%D7%99%D7%9D))
* [שרת לקוח](https://he.wikipedia.org/wiki/%D7%A9%D7%A8%D7%AA%E2%80%93%D7%9C%D7%A7%D7%95%D7%97)
* [GUI - Tkinter](https://docs.python.org/3/library/tkinter.html)
* [מסדי נתונים](https://www.sqlitetutorial.net/sqlite-python/)