**תיעוד פרוטוקול**

הפרוטוקול עובר ב- TCP (בחרו את ה- port שלכם. אנחנו נשתמש ב-**445**)

**מבנה כללי של הודעה (Message format)**

CCCCCCCCCCCCCCCC|LLLL | MMM

C - במבנה הודעה זו 16 תווים של פקודה. תווים של פקודה מציינים את סוג ההודעה (למשל- LOGIN). שאר התווים שנשארו מ-16 התווים הם תווי רווח (" ").

| - הפרדה בין חלקי ההודעה. הפרדה זו הכרחית.

L - שדה זה הוא באורך של 4 תווים, המציינים את גודל השדה הבא - שדה ההודעות (M). הערך יהיה בין 0 ל-9999 (ערך שלילי אינו חוקי).

| - הפרדה בין חלקי ההודעה. הפרדה זו הכרחית.

M - מידע המיוצג על ידי תווים. בחלק זה נרשום את ההודעה. ההודעה תכיל את המידע שנרצה להעביר. פקודות מסוימות לא דורשות שדה הודעה (כמו למשל הודעת LOGOUT) ובמקרים כאלה לא נמלא את התווים הללו ושדה זה ישאר ריק.

**הודעות (Messages)**

חלק זה מתאר את ההודעות השונות שניתן להעביר בין הלקוח לשרת, ומהשרת ללקוח.

**לקוח -> שרת (Client -> Server)**

**הודעת LOGIN**

תיאור ההודעה: בקשת התחברות מהמשתמש

LOGIN : פקודה

מבנה המידע: UUU...#PPP...

U - שם משתמש (username)

# - הפרדה בין חלקי ההודעה. הפרדה זו הכרחית.

P - סיסמה (password)

דוגמה להודעה אמיתית בפרוטוקול:

 LOGIN           |   9|aaaa#bbbb

**[**

**הודעת LOGOUT**

תיאור: התנתקות מהשרת

LOGOUT : פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית בפרוטוקול:

LOGOUT          |   0|

**LOGGED**

תיאור: קבלת רשימת משתמשים שמחוברים ברגע זה למשחק

LOGGED : פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית בפרוטוקול:

LOGGED          |   0|

**GET\_QUESTION**

תיאור: פקודה שמבקשת שאלת טריוויה מהשרת

GET\_QUESTION: פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית בפרוטוקול:

GET\_QUESTION    |   0|

**SEND\_ANSWER**

תיאור: פקודה ששולחת תשובה לשאלת הטריוויה

SEND\_ANSWER: פקודה

מבנה המידע: id#choice

Id: מספר מזהה של שאלת הטריוויה. ערך זה הוא מספרי

# - הפרדה בין חלקי ההודעה. הפרדה זו הכרחית.

choice: התשובה לשאלת הטריוויה. ערך זה הוא מספרי, ומייצג אחת מ-4 תשובות אפשריות לשאלת הטריוויה

דוגמה להודעה אמיתית:

SEND\_ANSWER     |   3|2#4

**MY\_SCORE**

תיאור: בקשה לקבלת מצב הנקודות הנוכחי של המשתמש שמחובר כרגע למערכת

MY\_SCORE: פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית:

MY\_SCORE        |   0|

**HIGHSCORE**

תיאור: בקשה לקבלת טבלת ניקוד עם הניקוד הגבוה ביותר מהשרת

HIGHSCORE: פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית:

HIGHSCORE       |   0|

**שרת -> לקוח (Server -> Client)**

**LOGIN\_OK**

תיאור: תגובה להודעת LOGIN. שולח הודעה שמשמעותה היא שההתחברות בוצעה בהצלחה

LOGIN\_OK: פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית:

LOGIN\_OK        |   0|

**LOGGED\_ANSWER**

תיאור: תגובה להודעה מסוג LOGGED שנשלחת מהלקוח, כוללת את רשימת המשתמשים המחוברים כרגע למשחק הטריוויה

LOGGED\_ANSWER: פקודה

username1, username2…  :מבנה המידע

כל משתמש שמחובר כרגע- פרטיו נשלחים. שמות המשתמשים מופיעים בשרשור עם הפרדה של פסיק  ', ' בין השמות.

דוגמה להודעה אמיתית:

LOGGED\_ANSWER   |   4|user1,user2

**YOUR\_QUESTION**

תיאור: תגובה להודעת GET\_QUESTION.שולח שאלת טריוויה למשתמש

YOUR\_QUESTION: פקודה

מבנה המידע: id#question#answer1#answer2#answer3#answer4

Id: המספר המזהה (או ערך מספרי אחר) של שאלת הטריוויה

Question: מחרוזת שמייצגת את השאלה (טקסט השאלה)

answer1-answer4: מחרוזת המייצגת את התשובות האפשריות לשאלה

דוגמה להודעה אמיתית:

YOUR\_QUESTION   |  26|2#How much is 1+1?#5#6#7#2

**CORRECT\_ANSWER**

תיאור: תגובה להודעת SEND\_ANSWER. מציין שהתשובה לשאלה הייתה נכונה

CORRECT\_ANSWER: פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית:

CORRECT\_ANSWER  |   0|

**WRONG\_ANSWER**

תיאור: תגובה להודעת SEND\_ANSWER. מציין שהתשובה לשאלה אינה נכונה

WRONG\_ANSWER: פקודה

מבנה המידע: answer

answer: ערך מספרי המציין את מספר התשובה הנכונה

דוגמה להודעה אמיתית:

WEONG\_ANSWER  |   1|2

**YOUR\_SCORE**

תיאור: תגובה להודעת MY\_SCORE. שולח את הניקוד של המשתמש המחובר כרגע

YOUR\_SCORE: פקודה

מבנה ההודעה: score

score: ערך מספרי המייצג את הניקוד הנוכחי של המשתמש

דוגמה להודעה אמיתית:

YOUR\_SCORE      |   1|5

**ALL\_SCORE**

תיאור: תגובה להודעת HIGHSCORE. שולח את טבלת המשתמשים עם הציונים הכי גבוהים.

ALL\_SCORE: פקודה

מבנה המידע: ...user1: score1\nuser2: score2\n

user1: שם המשתמש

score: הניקוד של המשתמש

\n: מפריד בין הציונים השונים

דוגמה להודעה אמיתית:

ALL\_SCORE       |  47|bambababy: 5\nabc: 0\ntest: 0\nadmin: 0\nblabla: 0\n

**ERROR**

תיאור: הודעת שגיאה. כאשר הלקוח מקבל את ההודעה הזאת, החיבור אמור להתנתק

ERROR: פקודה

מבנה ההודעה: error\_msg

error\_msg: מחרוזת המתארת את השגיאה. הערך הזה יכול להיות מחרוזת ריקה

דוגמה להודעה אמיתית:

**NO\_QUESTIONS**

תיאור: תגובה להודעת GET\_QUESTION. מציינת שאין יותר שאלות זמינות ולכן המשחק נגמר (GAME OVER)

NO\_QUESTIONS: פקודה

מבנה המידע: ריק

דוגמה להודעה אמיתית:

NO\_QUESTIONS     |   0|

**מצבי פרוטוקול (Protocol State)**

בפרוטוקול יש את המצבים הבאים:

**התנתקות Disconnected**

במצב זה של התנתקות, אין תקשורת בין הלקוח לשרת

**מחובר Connected**

במצב זה, ה-SOCKET של הלקוח מחובר לשרת, אבל הלקוח עדיין לא התחבר (logged in)

ההודעה היחידה **שהלקוח**יכול לשלוח במצב זה היא הודעה להתחבר-  LOGIN message

ההודעה **שהשרת**יכול לשלוח במצב זה היא הודעת **LOGIN\_OK**כדי לציין שהתחברות בוצעה בהצלחה, או הודעת **ERROR**כדי לציין שגיאה מכל סוג שהוא.

**Logged-in**

במצב זה, ה-SOCKET של הלקוח מחובר, והלקוח מחובר **LOGGED IN**

הלקןח יכול לשלוח כל הודעה במצב זה, **חוץ מ**הודעת התחברות **LOGIN**

השרת יכול לשלוח כל הודעה במצב זה, **חוץ מ**הודעת LOGIN\_OK