

הסבר כללי על המערכת ומבנה הקוד

מערכת הц'אט מבוססת פרוטוקול TCP ומיעדת לניהול תקשורת טקסטואלית בזמן אמת בין מספר משתמשים המוחברים לשרת מרכז. המערכת מתוכננת באופן מודולרי כדי לאפשר קרינות, ניהול חיבורים ותיעוד של פעולות המשתמשים.

✓ רכיבי המערכת:

1. שרת (server.py):

- מנהל חיבורים רבים בו-זמןית (ריבוי תהליכי במאיצעות Threads).
- מתווך בין הלקוחות, מפיץ הודעות לכל המשתמשים.
- מטפל במקרים של ניתוק לא צפוי.
- מתעד את הפעולות (חיבור, ניתוק, ושליחת הודעות) בקובץ log.txt.

2. לקוח (client.py):

- מאפשר למשתמש להתחבר לשרת באמצעות שם משתמש ייחודי.
- שולח הודעות לשרת ומציג את ההודעות המתתקבלות.
- מודיע למשתמש אם השרת התנתק או אם לקוח אחר עזב את השיחה.

3. קובץ תיעוד (chat_log.txt):

- זמן החיבור והניתוק של כל משתמש.
- תוכן ההודעות שנשלחו במהלך השיחה.

✓ תכונות עיקריות:

1. תקשורת דו-כיוונית:

- משתמשים יכולים לשלוח הודעות טקסט בזמן אמת ולקבל תשובות מלוקחות אחרים דרך השירות.

2. תיעוד שיחה מלא:

- כל האירועים (חיבורים, ניתוקים, והודעות) נרשומים בקובץ לצורך מעקב.

3. ניהול חיבורים:

- תמיכה במספר חיבורים בו-זמןית תוך טיפול בשגיאות וניהול משאבים נכון.

4. טיפול בשגיאות:

- השירות מזהה מקרים של ניתוק לקו או תקלה בראשת ומשדר הודעות מתאימות ללקחות.

✓ מבנה הקוד:

1. פונקציות מרכזיות בשרת:

- `handle_client` – מנהלת כל לקוח בנפרד.
- `broadcast` - שולחת הודעה לכל לקוחות המתחברים.
- `log_chat_activity` - מתעד את הפעולות בקובץ התייעוד.

2. פונקציות מרכזיות בלקוח:

- `receive_messages` - מאזינה להודעות מהשרת.
- `send_messages` - שולחת הודעות לשרת.

✓ שימושים אפשריים:

- מערכות תקשורת פשוטות בין משתמשים.
- מערכות לניהול לוגים או בקרה של משתמשים רבים.