מבוא

הפרויקט שלי הוא צאט בין 2 משתמשים.

הצאט מבוסס על p2p במקום צאט רגיל שבדרך כלל בנוי על שרת שאליו נשלח ויוצא כל המידע. בחרתי בדרך זו בגלל מספר סיבות

פרטיות ואנונימיות: מאחר והתקשורת ברשתות P2P ישירה בין משתמשים, ישנה אפשרות לשמור על פרטיות ואנונימיות גבוהה יותר, בהשוואה לתקשורת דרך שרתים מרכזיים שעלולה להיות פחות פרטית.

גמישות בתפעול: כאשר אין תלות בשרת מרכזי, יש גמישות רבה יותר בתפעול הרשת. זה מאפשר למשתמשים להתחבר ולתקשר בצורה קלה ומהירה, גם בתנאים של רשתות מוגבלות או פתאומיות.

בדומה לאפליקציית Whatsapp יצרתי תכנית לשמירת היסטוריית הצאטים גם לאחר שעבר זמן מאז ההתחברות האחרונה.

שומר על היסטוריית הצאטים

תיאור פרויקט

כאשר פותחים את הפרויקט מופיע מסך התחברות עם אפשרויות להזין אימייל וסיסמא, אם ברצונך ליצור משתמש צריך ללחוץ register

כדי לעבור לצאט צריך ללחוץ על Login לאחר הזנת פרטי המשתמש

כאשר הועברת למסך הצאט צריך להזין כתובת הסוקט ולהתחיל להתכתב עם המשתמש השני

סקרת ספרות

* pyQt5

ספריית PyQt5 היא ספריית פייתון שמאפשרת לך ליצור ממשקי משתמש גרפיים (GUI) בצורה יעילה ויפה. היא מבוססת על הספרייה Qt, שפותחה על ידי חברת Qt Company. PyQt5 מציעה גרסה מודרנית וקלה יותר לשימוש בפייתון.

הנה כמה מהיתרונות של PyQt5:

1. CrossPlatform עובד על מערכות הפעלה שונות כמו Windows, macOS ו-Linux.

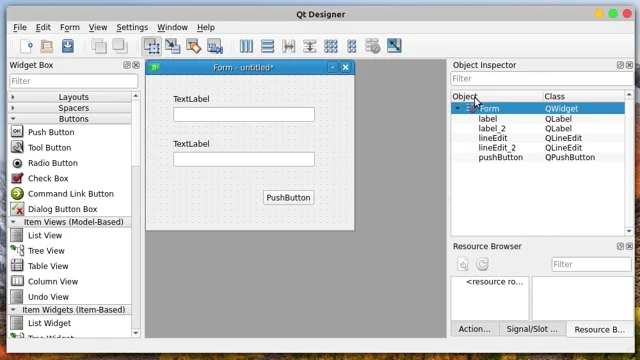
2. מגוון רחב של רכיבים ואפשרויות עיצוב: PyQt5 מספקת מגוון רחב של אלמנטים גרפיים וכלים ליצירת ממשקים משתמש מתקדמים. זה כולל כפתורים, תיבות טקסט, טבלאות, ועוד.

3. תכנות מונחה עצמים: PyQt5 משתמשת באובייקטים, מה שמאפשר לך ליצור קוד יעיל ומסודר.

4. למידה מהירה: יש לספריית PyQt5 קהילה רחבה ותיעוד טוב, כך שזה יכול להיות קל ללמוד איך להשתמש בה.

5. ריבוי תכונות: PyQt5 מאפשרת לך להוסיף תכונות תצורה לאלמנטים גרפיים, כמו צבעים, גדלים, ועוד.

6. Designer: היא תוכנה נוספת שאפשר להתקין והיא מאפשרת למתכנת ליצור UI בעזרת כלי ויזואלי דרכו קל מאוד להוסיף ולמקם widgets (אלמנטים) במסך על ידיד גרירה.



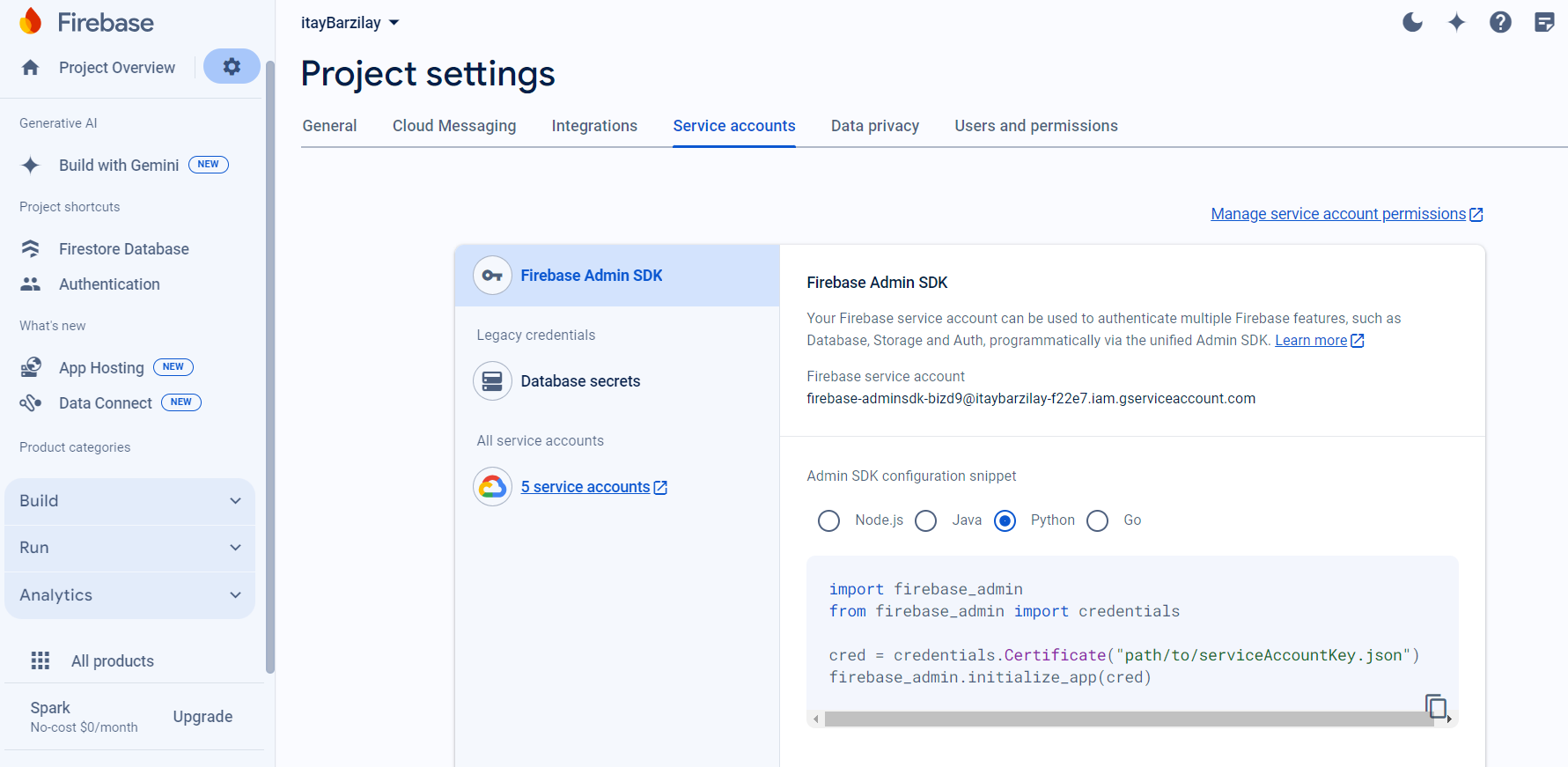
בפרויקט שלי התלבטתי אם להשתמש בו אבל היות והממשק הגרפי שלי מאוד פשוט (כמה תיבות טקסט וכפתורים) לא ראיתי בזה צורך.

חשוב לי לציין שבתחילת הפרויקט עבדתי עם tkinter והיה הרבה פחות נוח ולכן מהלצת המורה שלי עברתי לשימוש ב pyQt5

השימוש ב-PyQt5 יכול להקל על הפיתוח של ממשקי משתמש עבור יישומים רבים, כולל יישומי משחקים, יישומי עסקים, ויישומי תוכנה בדידה ורשת.

* Firebase הוא פלטפורמה שפותחה על ידי Google המספקת מגוון שירותים לפיתוח יישומים ניידים ואינטרנטיים. פופולריותה גדלה בשנים האחרונות בזכות היכולת שלה לספק שירותים מתקדמים בתחום של ניהול משתמשים, איחסון נתונים בזמן אמת, ניתוב הודעות, ועוד...

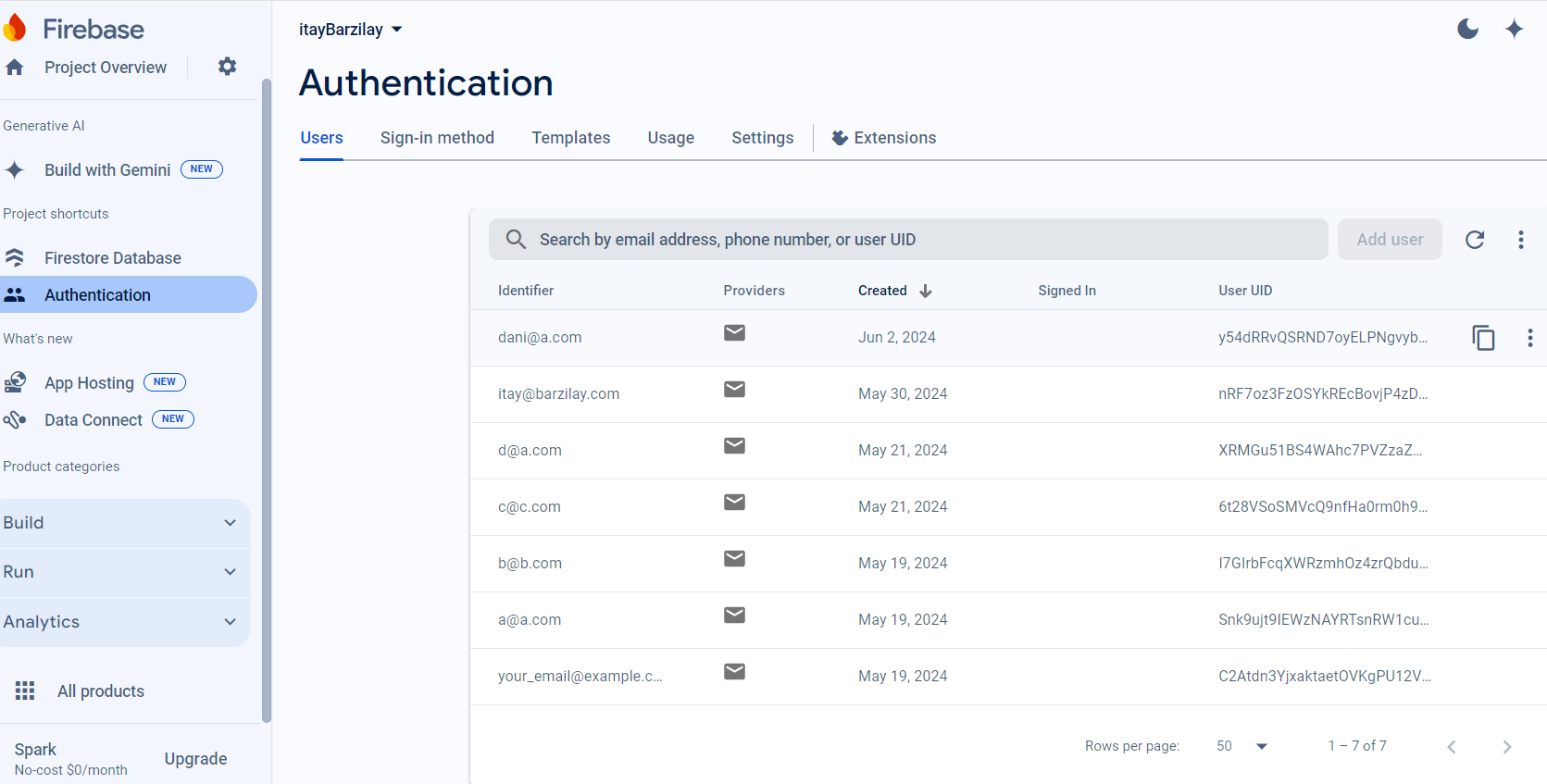
כאשר אתה רוצה להשתמש בשירותי firebase בפרויקט ב-Python על המפתח להוריד מהמסוף קובץ json ובו "מפתח" סודי שיאפשר לי לגשת ל-firebase.



* + Authentication Firebase Authentication הוא שירות המאפשר לך להוסיף תהליכי אימות משתמשים ביישומים שלך. השירות מספק אפשרויות רבות לאימות משתמשים, כולל אימות דואר אלקטרוני, גוגל, פייסבוק, טוויטר ועוד. זה מאפשר למפתחים להשתמש בתהליכי אימות מוכנים ובטוחים ביישומים שלהם במהירות וביעילות.

כך נראה מסך במסוף firebase ובו כל המשתמשים.

כפי שניתן לראות, הסיסמא אינה מופיעה היות והיא מוצפנת.

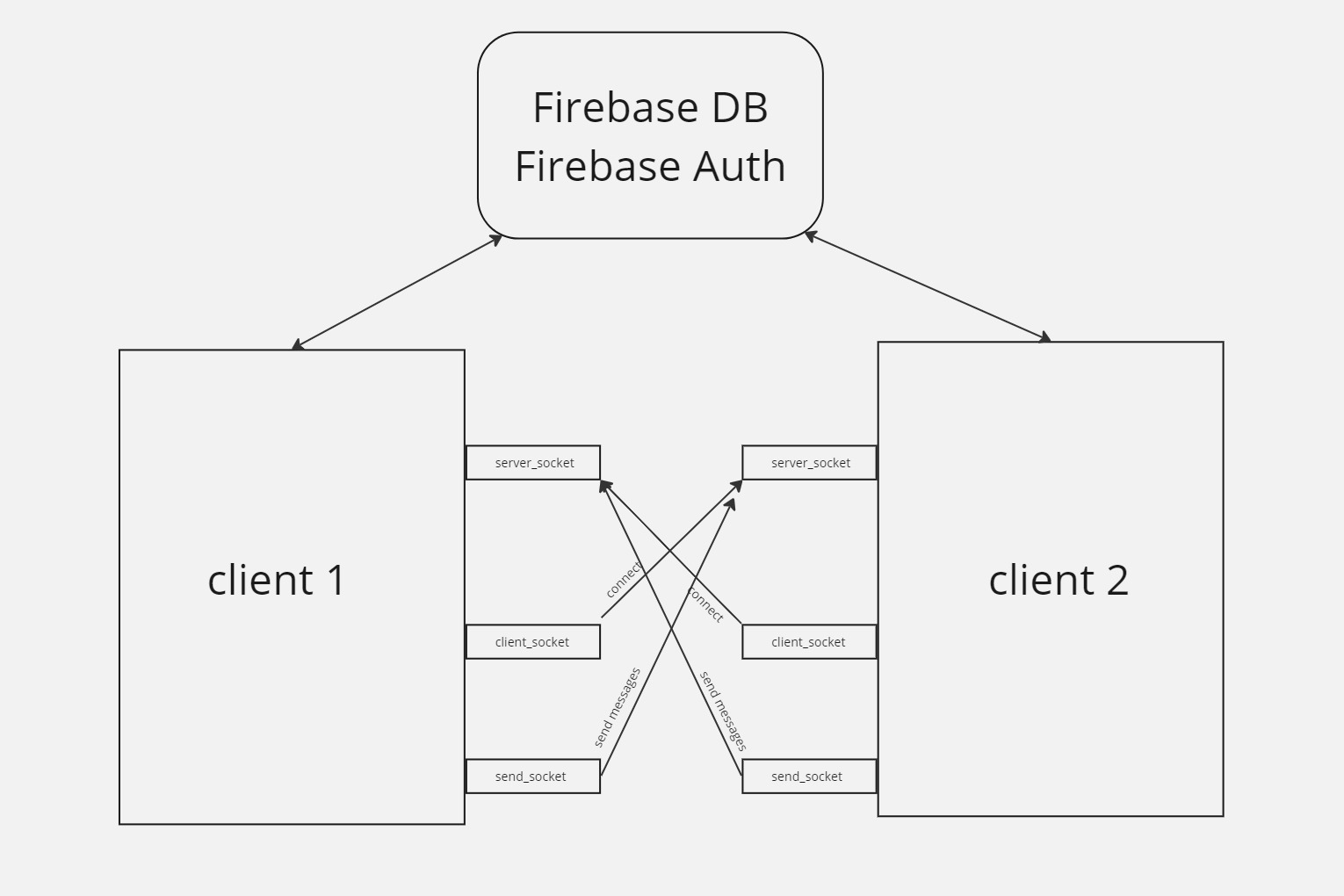


* + Firebase Database **(Database)** מציעה שירות מסד נתונים בזמן אמת בשם Firebase Realtime Database. השירות מאפשר לך לאחסן ולסנכרן נתונים בין יישומים שונים, כולל יישומים ניידים ואתרי אינטרנט. הוא מבוסס על תשתית של JSON, מה שהופך אותו לפשוט לשימוש ולאיחוד עם רוב השפות והפלטפורמות
* t כשמדברים על תקשורת P2P (Peer-to-Peer), מבנה ה־Socket משמש כאמצעי ליצירת תקשורת ישירה בין שני משתמשים או בין מכשירים שונים ברשת באמצעות התחברות ישירה, ללא צורך בשרת מרכזי. זה יכול להיות שימושי במקרים שבהם רצוי להעביר נתונים ישירות בין משתמשים, כמו בשיתוף קבצים או שיחות וידאו, ובמקרים כמו רשתות תפעוליות פיתוח פירותיות.

התקשורת P2P בנויה סביב הרעיון שכל משתמש או מכשיר ברשת יכול לפעול כשרת ולקוח כאחד. מבנה ה־Socket כאן משמש להקים חיבורים ישירים בין המשתמשים, כך שהם יכולים לשלוח ולקבל נתונים ישירות ביניהם.

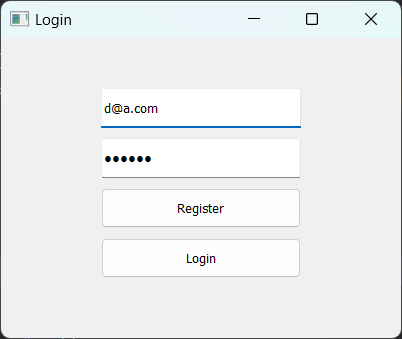
בעזרת מבנה ה־Socket, מתאפשרת התקשורת ישירה בין המשתמשים, כשכל משתמש מאתחל חיבור חדש כשרת או כלקוח ביחס למשתמש השני. כל משתמש יכול לשלוח ולקבל נתונים על ידי ה-Socket שלו.

כדי לבצע תקשורת P2P עם Socket, משתמשים צריכים להגדיר את הפרוטוקול שלהם ואת התהליכים להתחברות, תקשורת, וניהול שגיאות. זה דורש מימוש תוכנה עבור שני הצדדים של התקשורת, כאשר כל אחד מן המשתמשים מחליט אם לפתוח חיבור כשרת או כלקוח לצורך התקשורת.

ארכיטקטורה

מדריך למשתמש

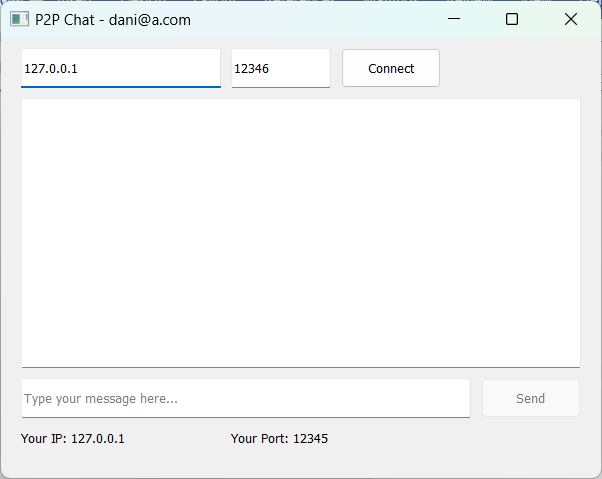
הרצת האפליקציה פותחת מסך login:



בחלון המשתמש יזין את הכתובת מייל שלו ואת הסיסמא אותה הוא יבחר

לחיצה על Register המידע תקין לשמירה ואחר כך יצור משתמש חדש

לחיצה על Login תעביר אותך על הצאט בתנאי שישנו משתמש קיים עם הפרטים התואמים להזנת המשתמש



בצד שמאל ניתן להזין את כתובת הip של הclient השני כדי להתחבר לצאט

מימין לכתובת הip ניתן להזין את הport של ה client השני. הזנת ה PORT רלוונטית רק לגרסת הפיתוח היות ואני בודק את הפרויקט על אותו מחשב ולכן אי אפשר להאזין מאותו פורט. כמובן שבבחינה נציג את המערכת על שני מחשבים אבל היה לי חשוב להשאיר את השדה של ה PORT על מנת להראות את תהליך הפיתוח.

לחיצה על CONNRCT תחבר אותך אל הצאט בתנאי שהכל תקין (פותח סוקט מול ה"שרת" של הלקוח השני.

בחלון הריק ניתן לכתוב את תוכן הצאט

לחיצה על כפתור הSEND יעבוד בתנאי ששני הCLIRNTES מחוברים אחד לשני

מחלקות פונקציות

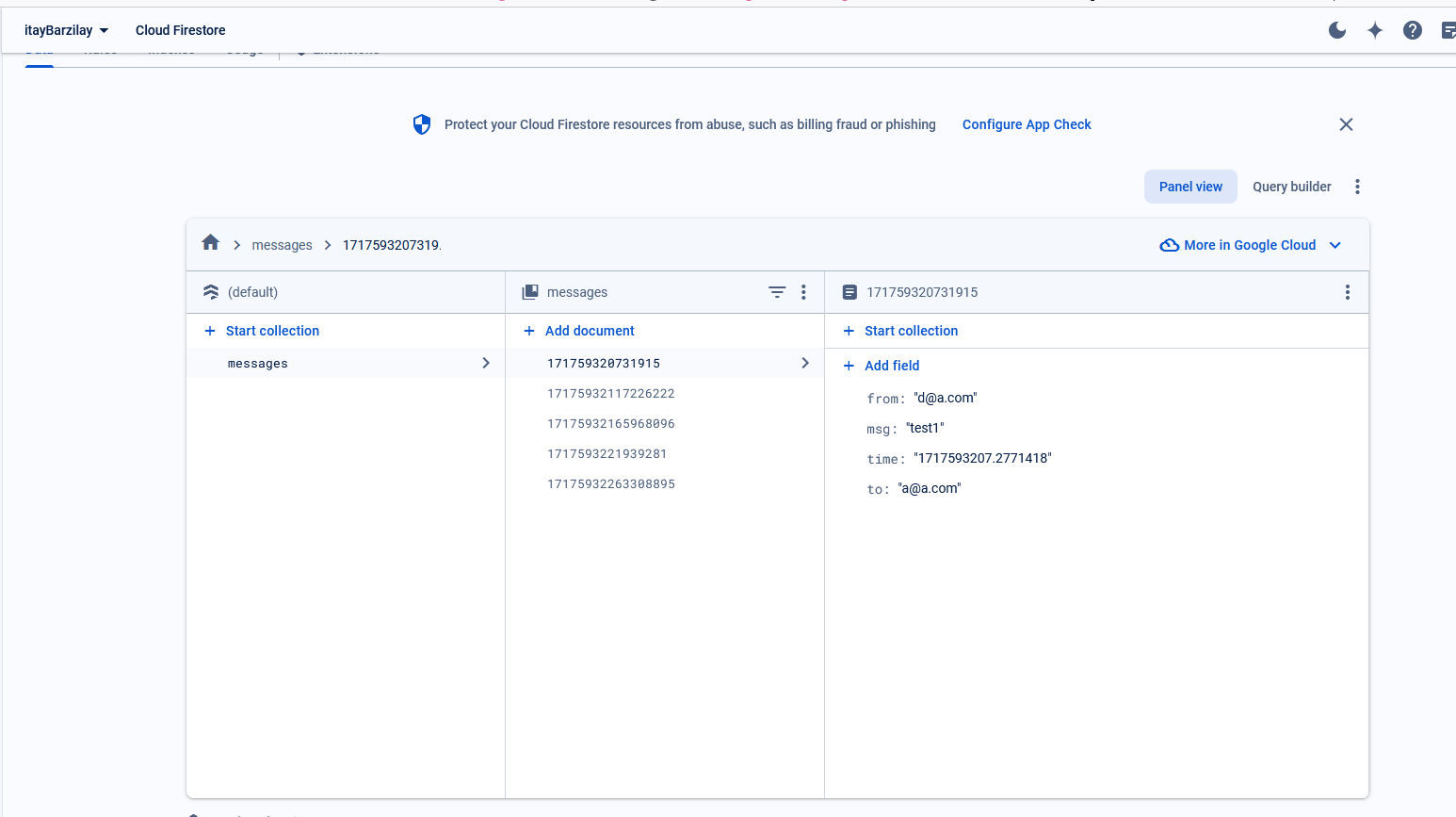
בסיס נתונים

בסיס הנתונים שלנו די פשוט. הוא מכיל רק הודעות.

כל הודעה מכילה 4 פרמטרים:

1. From-כתובת המשתמש ששלח את ההודעה
2. Msg- מה ההודעה מכילה
3. Time-זמן שליחת ההודעה בזמן UNIX
4. To- כתובת היעד אליו ההודעה נשלחה

כל הודעה נשמרת כ document ב- DB



כאשר המזהה (id) של כל הודעה הוא הזמן epoch בו ההודעה התקבלה.

- Unix דרך למדוד זמן במחשבים. זמן Unix מתייחס למספר השניות שעברו מתאריך מסוים בתוכנית המחשב המקורית של Unix, משומשת כיום במערכות הפעלה רבות. תאריך התחלת הזמן Unix הוא הרצף (Epoch) של 1 בינואר 1970, מה שמאפיין את זמן Unix הוא שהוא משמש כדי לייצג זמן בצורה של מספר שניות שחלפו מהרצף הזמני הזה ועד לרגע הנוכחי.