



נושא העבודה: בוט לשימור והצגת נתונים

התמחות: הגנת סייבר

לוגו :



שם התלמיד: מיכאל ספקטור

ת.ז. התלמיד: 325112738

שם בית ספר ועיר: אורט שפירא, כפר סבא

שמות המנחים והמלווים: צביקה שטרקמן ומוטי גורנשטיין

מועד הגשת הספר: 03.05.2020

2. תוכן עניינים

3. מבוא ורקע כללי..... 3
3. תקציר כללי – תיאור הפרויקט..... 3
3. מה היו המניעים לפיתוח הפרויקט..... 3
4. דרישות הפרויקט..... 4
4. תיאור דרישות פונקציונליות..... 4
4. מה הפרויקט אמור לעשות..... 4
4. סוגי המשתמשים השונים..... 4
5. דרישות המערכת..... 5
5. מה קיים בשוק :..... 5
6. טכנולוגיות בהן השתמשתי בפרויקט :..... 5
5. שפת התכנות :..... 5
6. חבילות תוכנה (ספריות, מודולים ועוד)..... 6
10. היבטי אבטחת מידע..... 6
15. הרחבות שהייתי רוצה להשלים בהמשך הפרויקט..... 7
16. מקורות מידע..... 7
17. רפלקציה..... 7
7. איך הייתה העבודה עבורי..... 7
7. נושאים שחקרתי ולמדתי בעצמי..... 7

3. מבוא ורקע כללי

תקציר כללי – תיאור הפרויקט

פיתחתי מערכת בוט (Bot) הנקראת Analitica Bot המיועדת לשימוש של מנהלי שרתים בפלטפורמות מרובות משתמשים מבוססות VoIP (Voice Over IP).

התמקדתי בפלטפורמת דיסקורד (<https://discordapp.com/>) המתמחה בתקשורת בין משתמשים באמצעות התכתבות, שיחות וידאו ושיחות שמע בערוץ צ'אט בעיקר עבור קהילות משחקי וידאו. הפלטפורמה מורכבת משרתים וירטואליים, שכל משתמש של הפלטפורמה יכול ליצור שרתים כאלו ולהצטרף לשרתים של משתמשים אחרים. כדוגמה, אני הכנתי שרת שהנושא שלו הוא משחק מסוים, ובשרת זה יש לי כל מיני ערוצים שבהם חברי ואני יכולים לדבר על המשחק. לדיסקורד כרגע יותר מ-250 מיליון משתמשים רשומים וכ-14 מיליון משתמשים שפעילים בכל יום, והוא הוקם בשנת 2015. דיסקורד עובד על וינדוס, מאק, אנדרואיד, לינוקס וכו'. אלו גם כמה מהסיבות שבחרתי להתמקד בפלטפורמה זו. דיסקורד החלו לא מזמן בעקבות המצב גם להגדיל את סוגי האנשים שהם רוצים שישתמשו בפלטפורמה שלהם והם התחילו להציע שרתים מוכנים לסוגי משתמשים שונים, לדוגמה: שרת בשביל כיתה, שרת לשיעורי ריקוד, שרת בשביל משפחות שיתקשרו וכו'. בכל שרת יש לבעל השרת יש אפשרות להגדיר תפרידים שונים ולתת את התפרידים לאנשים שהוא מחליט, אחד מהתפרידים שבעל שרת יכול ליצור הוא מנהל שרת או עוזר שרת שלרוב יש להם יותר יכולות וגישה ממשתמשים רגילים, ובשבילם הבוט שלי.

הבוט מתחבר ומשתמש ב API של דיסקורד שפותח עבוד שפת פייתון. דרך ה API עוברת כל התקשורת בין הבוט שלי לבין פלטפורמת הדיסקורד.

הבוט אוסף מידע באופן רציף משרתי הדיסקורד ואוגר אותו בבסיס נתונים. הבוט מתוכנת מראש לאסוף את כל סוגי המידע האפשריים מבחינה חוקית מפלטפורמת הדיסקורד ושומר אותם בבסיס הנתונים כמידע גולמי. כל בעל שרת יכול להוסיף את הבוט לשרת שלו ולהשתמש בשירותיו. לאחר שבעל השרת הוסיף את הבוט לשרת, הבוט יתחיל באיסוף מידע באופן רציף על המשתמשים שנמצאים באותו שרת. כעת, יכול כל מנהל בשרת לשלוח בקשות לבוט על מידע שונה שהוא רוצה, ואז הבוט ניגש לבסיס הנתונים, שולף את המידע, מנתח אותו, ומוציא דוח בהתאם לבקשה שמוצג בצא"ט שבו המנהל ביצע את הבקשה של הנתונים.

מה היו המניעים לפיתוח הפרויקט

אני משתמש בדיסקורד כבר מספר שנים, ובמהלך אותם שנים הייתי בעל מספר שרתים שונים. בכל פעם הייתה לי הבעיה שהייתי צריך נתונים וסטטיסטיקה על המשתמשים בשרת שלי אבל לא הייתה לי דרך להשיג את המידע הזה. אז חשבתי אם כבר אני צריך לעשות פרויקט גמר, למה לא לעזור לעצמי ולאחרים בנושא הזה. ידעתי גם שזה יהיה מאתגר לעשות את הפרויקט הזה ואצטרך ללמוד הרבה דברים חדשים לבד, משהוא שגם משך אותי לפרויקט הזה.

4. דרישות הפרויקט

תיאור דרישות פונקציונליות

מה הפרויקט אמור לעשות

בקצרה, מה הפרויקט אמור לעשות? – הפרויקט אוסף מידע מפלטפורמת דיסקורד ומאחסן אותו. כאשר מנהל שרת מבקש מידע, הבוט מוציא רק את המידע הנדרש, ואז הוא מציג אותו בצורה שמנהל השרת ביקש.

הבוט שולף מדיסקורד את כל המידע המתון באותו רגע על כל משתמש בכל שרת בו הבוט נמצא, ואז הוא שומר רק את המידע שהוגדר: שם המשתמש, מה הוא עושה באותו רגע, מה המצב שלו כרגע, ומה הזמן בו בוצעה השליפה.

בבוט קיימת גם מערכת הרשאות שעליה אפרט בפרק על בסיסי הנתונים.

כעת נפרט על כל סוג:

שם המשתמש – לכל משתמש יש שם ומספר שמחובר למשתמש, לדוגמה השם שלי הוא HolyRidek והמספר שלי הוא 9770, ולכן הבוט יקבל את המידע: HolyRidek#9770.

מה המשתמש עושה באותו רגע – דיסקורד בודק בכל רגע נתון מה המשתמש עושה, אם הוא עושה משהו בנושאים מסוימים: האם הוא מקשיב למוזיקה באפליקציית ספוטיפיי? האם הוא משחק במשחק מסוים, ואם כן, באיזה משחק הוא משחק? למשתמש גם אפשרות לשים פעולה שהוא הגדיר בדיסקורד.

מה מצב המשתמש באותו רגע – לכל משתמש בדיסקורד יש היכולת להיות באחד מארבעה מצבים: online, offline, idle, do not disturb. הבוט לוקח את מצב המשתמש באותו רגע ומאחסן אותו.

מה הזמן בו בוצעה השליפה – הזמן הנוכחי שבו בוצעה הבדיקה לפי פורמט Epoch, זאת אומרת מספר השניות שעבר מאז 1 בינואר, 1970.

סוגי המשתמשים השונים

נגדיר את סוגי המשתמשים השונים: מנהל המערכת, בעל שרת, מנהל שרת, עוזר שרת, משתמש.

כעת נפרט על כל סוג:

מנהל המערכת: מנהל המערכת מנהל ומפעיל את הבוט. הוא קובע מתי הבוט פועל, מתי לא. המנהל יכול לראות דוחות שונים: לכמה שרתים הבוט מחובר, כמה משתמשים קיימים בכל השרתים ביחד, מה אחוז המשתמשים שפעילים ברגע זה, כמה אחוז מהמשתמשים עושים פעולה מסוימת ועוד דוחות שיאפשרו לו לפקח על הנעשה במערכת. חשוב לי לציין, שעקב מורכבות הפרויקט ישנם אפשרויות שאותם עדיין לא יישמתי, ולכן אותם מנהל המערכת עדיין לא יכול לבצע. על מנהל המערכת גם האחריות לבדוק שהבוט והתוכנה פועלים באופן שותף ובלי תקלות, ולדווח על כל תקלה למתכנת הבוט.

בעל שרת: לבעל השרת גישה לכל המידע הקיים על השרת שלו, על כל משתמש שנמצא בשרת שלו, וגם יש לו גישה לשנות את הגישה לסוגי מידע שונים לכל מנהל שרת, עוזר שרת ומשתמש בשרת שלו. סוגי גישה שמנהל השרת יכול לתת או לקחת הם: היכולת לבקש נתונים על כל השרת, לבקש נתונים על משתמש מסוים ועוד. כמובן שבעל השרת יכול לבקש כל סוג נתונים אפשריים, כמו מידע על השרת, ורשימה של כל המשתמשים ומה ההרשאות וסוג המשתמש של כל אחד מהם ועוד.

מנהל שרת: תפקיד שמשתמש יכול לקבל רק מבעל השרת. המנהל יכול לתת תפקידים כמו עוזר שרת למשתמשים. לכל מנהל שרת גם גישה לכל אותו המידע שבעל השרת יכול לבקש, כגון: מידע על כל השרת, מידע על משתמש מסוים, ורשימה של כל המשתמשים ומה הסוג של כל אחד, או משתמש מסוים, מה סוג אותו משתמש ומה ההרשאות הספציפיות של אותו משתמש.

עוזר שרת: תפקיד שמשתמש יכול לקבל מבעל שרת או מנהל שרת. לעוזר שרת היכולת לבקש מידע מוגבל, כמו מידע על משתמשים מסוימים, ורשימה של כל המשתמשים ומה הסוג של כל אחד, או משתמש מסוים ומה סוג אותו משתמש.

משתמש: הסוג הכי בסיסי של משתמש, ולו היכולת רק לבקש מידע/נתונים על עצמו, ורשימה של כל המשתמשים ומה הסוג של כל אחד, או משתמש מסוים ומה סוג אותו משתמש.

דרישות המערכת

על מנת להפעיל את התוכנה ישנן מספר דרישות:

מערכת הפעלה: התוכנה רצה בכל מערכות ההפעלה הגדולות אבל עובדת הכי טוב ב: Windows Vista לפחות, Linux, macOS 10.6 לפחות או FreeBSD 10 לפחות.

שפות תכנות: בשביל להריץ את התוכנה נדרשת התקנה של Python 3 על המחשב.

ספריות: על המחשב בחיבור עם פייתון חייבות להיות מותקנות הספריות הבאות: matplotlib, numpy, discord.py. הספרייה הראשונה נדרשת כי היא ה API שמקשר בין הבוט לדיסקורד, הספרייה השנייה והשלישית נדרשות בכדי ליצור את הגרפים שהתוכנה לפעמים מייצרת.

אינטרנט: הפרויקט הינו מבוסס על תקשורת עם פלטפורמת דיסקורד, על כן התוכנה חייבת גישה לאינטרנט.

מחשב שעליו התוכנה יכולה לפעול בלי הפסקה: בתוכנה יש איסוף מידע רציף ולכן כדי להציג למנהלים ולמשתמשים תוצאות מדויקות, ועל מנת שהם בכלל יוכלו לבקש את המידע הזה, על התוכנה לפעול באופן רציף ככל האפשר בלי הפסקות.

5. מה קיים בשוק:

התחלתי לעבוד על הפרויקט הזה ספציפית מכיוון שלא מצאתי מוצרים דומים בשוק.

6. טכנולוגיות בהן השתמשתי בפרויקט:

שפת התכנות:

כחלק מתהליך פיתוח הפרויקט, היה עלי תחילה להחליט מה הן שפות התכנות בהן אשתמש כדי להכין את הפרויקט. בזמן זה מכל שפות התכנות שידעתי (C#, Python, Assembly, Lua) הייתי בקיא ביותר בשפת התכנות Python, לכן החלטתי שאני אתחיל לכתוב את הפרויקט בשפה זו. בנוסף לכך, Python היא שפה שנועדה להיות נוחה, יעילה, קלה לשימוש וקלה ללמוד. לכן חשבתי שעבור פרויקט בתחומים שאני ככל הנראה לא אכיר ואצטרך ללמוד בעצמי עליהם, יהיה לי יעיל לעשות זאת בצורה קלה וידידותית.

כדי לכתוב את הפרויקט עצמו, כלומר את הקוד, בחרתי לתכנת ב-Pycharm, שכן זו היא סביבת פיתוח יחסית מפורסמת ונוחה לעבודה, שבה השתמשתי כבר לאורך זמן בתיכון ולכן אני היא גם מוכרת לי. למעשה סביבת הפיתוח בעלת השפעה זניחה על המוצר לפי דעתי, שכן הינה רק מספקת תנאי עבודה נוחים יותר, לכן גם לא טרחתי לחפש תוכנות אחרות שבהן אוכל לרשום קוד.

חבילות תוכנה (ספריות, מודולים ועוד)

Matplotlib – ספרייה plotting בשביל שפת פייתון שמאפשרת להציג נתונים באמצעות גרפים ומאפשר הצגה של אותם גרפים דרך סביבת הפיתוח או שמירת אותם גרפים כקבצי תמונה למחשב. כשחיפשתי ספרייה בשביל לצייר גרפים מצאתי הרבה מאוד ספריות שיכולות לעשות זאת, אבל משאלות של אנשים אחרים, רוב המתכנתים הציעו את הספרייה שבחרתי מכיוון שהיא הכי פשוטה ונוחה, וגם לא צריך שרת אינטרנט שדרכו מעבירים את התמונות, בניגוד לרוב הספריות האחרות.

SQLite3 – ספרייה עבור תקשורת עם מסדי נתונים – Databases. הפרויקט לא צריך מעבר לקישור של מסד נתונים אחד בזמן אחד. לכן נוח לבחור במודול מוכן ומוכר עבור עבודה עם מסדי הנתונים.

SQLite Studio – תכנה שמאפשרת לערוך ולצפות באופן וויזואלי במסדי נתונים. השתמשתי בה כדי לעקוב אחר הפעולות שנעשות בפרויקט שלי על מסדי הנתונים ולצורכי דיבוג.

GitHub – אתר וכלי שהשתמשתי בו בכדי לגבות גרסאות הקוד של הפרויקט שלי, כדי לשמור עותקים שלו למקרים שאצטרך לעבוד עליו במקומות שונים מלבד הבית ולמקרים שמשוהו יקרה בקוד שלי ואצטרך לחזור לגרסה קודמת. בחרתי בכלי זה מכיוון שאני מכיר אותו טוב וקל לי לעבוד איתו.

discord.py – הספרייה המרכזית בה השתמשתי בכדי לתקשר בין הבוט שלי לבין אפליקציית דיסקורד. אין אלטרנטיבה אחרת לפייתון ולכן בחרתי בספרייה זו.

10. היבטי אבטחת מידע

האבטחת מידע שקיימת כרגע בפרויקט היא בהפעלה של התוכנית. צריך טוקן שאפשר לקבל אותו רק מכניסה למשתמש שלי בדיסקורד ששם צריך משתמש, סיסמה וקוד שמגיע אליי לטלפון. בלי הקוד של הדיסקורד אי אפשר להפעיל את הבוט.

בנוסף, אחרי שאני מכין ושולח את הגרפים עם הנתונים, אני מוחק אותם כדי שאם מישהו יקבל גישה למחשב שלי הוא לא יוכל לראות מידע על אנשים שלא קשורים אליו.

14. באגים פתוחים \ בעיות קיימות + מגבלות שקיימות

- מגבלה אחת גדולה היא על כמות הבקשות שיכולות להגיע לבוט באותו הזמן, כלומר ככל שיש יותר בקשות באותו הזמן, יקח לו יותר זמן להחזיר תשובה
- מגבלה שנייה שקיימת היא כמות השרתים שבהם הבוט יכול להיות קיים, לפי מספר האנשים, כי יקח לו יותר זמן לעבור על המסד נתונים כאשר יש יותר נתונים.

15. הרחבות שהייתי רוצה להשלים בהמשך הפרויקט

בהמשך הפרויקט הייתי רוצה:

- להכניס מערכת הרשאות לפרויקט כדי שלא כל אחד שנמצא בשרת, יוכל להשיג מידע או סטטיסטיקות על אנשים אחרים אלא רק אנשים המורשים לכך.
- הוספה של עוד סוגי מידע שאפשר לבקש כגון מתי המשתמש היה בשיחת קול ביום.
- אפשרות למנהל המערכת לקבל גרפים ודיווחים על המערכת כולה.
- מתן אפשרות למשתמשים לשלוח באמצעות הבוט הצעות לשיפור וביקורת על הבוט.
- שיפור של מערכת LOGGING מכיוון שכרגע כל הלוגינג הוא בקונסול והייתי רוצה להעביר את לקובץ log.

16. מקורות מידע

- <https://stackoverflow.com> – לפתרון בעיות נקודתיות.
- <https://discordpy.readthedocs.io/en/latest> - ללמידת מודול discord.py.
- <https://matplotlib.org> – ללימוד מודול.
- [https://en.wikipedia.org/wiki/Discord_\(software\)](https://en.wikipedia.org/wiki/Discord_(software)) – ללמידה יותר עמוקה על אפליקציית דיסקורד.
- <https://docs.python.org/3> - ללמידה נוספת על מודולים.
- שרתי דיסקורד של תכנות שעזרו לי בפתרון בעיות מסוימות.

17. רפלקציה

איך הייתה העבודה עבורי

העבודה הייתה מאתגרת, אך באותו הזמן נהניתי לעבוד עליה מאוד.

נושאים שחקרתי ולמדתי בעצמי

SQL & Databases – למדתי SQL ברמה הבסיסית, וכיצד להשתמש בה ובמסדי נתונים.

Discord.py – למדתי איך להשתמש ולתפעל את הספרייה המאוד מורכבת הזאת, למראת שהשתמשתי בחלק יחסית קטן ממנה.

Matplotlib – למדתי את הספרייה הזאת ואיך להכין ולצייר גרפים.

Program Architecture – למדתי איך לשרטט נכון דיאגרמות ותרשימי זרימה של תכנית.

פייתון בכללי – שמת לי לב שלאחר הפרויקט הקוד שלי יותר איכותי וקריא.

הצעת פרויקט ראשונית

תיאור הפרויקט

הפרויקט הוא בוט לפלטפורמות תקשורת כמו "דיסקורד" שאוסף כל מיני נתונים על משתמשים בשרתים, שומר אותם וכאשר מגיעה בקשה, מעבד אותם ומחזיר את הנתונים הרלוונטיים למבקש.

הפרויקט בא לפתור את הבעיה של מנהלים של שרתים גדולים ובינוניים בפלטפורמת "דיסקורד" בלדעת את הנתונים של האנשים שמשתמשים בשרת שלהם, ולדעת מתי הכי הרבה אנשים מחוברים ומדברים בשרת שלהם, וכך לדעת מתי לפרסם הודעות חשובות וגם לדעת מתי אנשים מסוימים יתחברו וכו'.

בוט זה מחדש הרבה דברים כי לפי הבדיקה שלי לא קיים בוט כזה בדיסקורד, או לפחות לא קיים בוט ברמה הזאת עם יכולות אלו.

דרישות פונקציונליות

- המשתמשים במערכת יהיו משתמשים רגילים בשרת דיסקורד, המנהלים בשרת הזה ומפתח הבוט.
- הלקוח יוכל לבקש מהבוט להציג את המידע שהוא אסף בדרכים שונות וגם יוכלו לבקש ממנו לחזות את הפעולות של משתמש מסוים.
- מבחינת ניהול ועיבוד מידע השרת יקח את כל המידע שהוא משיג מהשרת דיסקורד, ישמור בשרת בטבלה מסודרת, ולפי זה ועם machine learning ועם אלגוריתמים יחזה את הפעולות העתידיות של המשתמש ליום הקרוב.
- מה שיוצג למשתמש זה תמונה עם כל המידע שהוא ביקש, שתמונה זו תישלח לערוץ בשרת שבו המנהל ביקש את המידע.
- הבוט עצמו ישב על שרת חיצוני באינטרנט וכל מנהל יכניס את הבוט לשרת שלו והוא יפעל שם (כל המידע ישמור ב-database).
- בעיות שצפויות הן: איך להפוך את המידע לתמונה, איך לאסוף את המידע מדיסקורד.
- מגבלות: כמות שרתים שבו יוכל הבוט לעבוד.

תיאור המימוש ארכיטקטורת הפרויקט

מרכיבי הפרויקט:

- השרת – בו יתקיים כל שימור המידע עיבוד חלק מהמידע ושליחת השאר ל"משינלרנינג" לעיבוד שאר המידע, ואז קבלת המידע מאותו מודול והפיכת המידע לתמונה ושליחה חזרה לבוט
- הבוט – יאסוף את כל הנתונים וישלח לשרת וגם יקבל את התמונות הסופיות של השרת ויפרסם אותן בשרת הדיסקורד
- משינלרנינג – בו מתבצע עיבוד המידע וחזיית הפעולות של המשתמש ושליחת המידע חזרה לשרת

אני אעבוד בסביבת העבודה "Pycharm" ואשתמש בשפת פייתון לתכנות הפרויקט. אני אשתמש בספריות של threading, ול machine learning, דיסקורד,

תוכנית עבודה לפרויקט

- יש ללמוד את ה api של דיסקורד, machine learning, ויש להעמיק בנושא של threading
- אני אתחיל בבניית איסוף המידע של הבוט, אז אני אתחיל לעבוד על שמירת המידע ואז על עיבוד המידע ואז על החזייה של הפעולות ולבסוף גם שליחת המידע חזרה.
- במערכת בסיסית עובדת אני אכין את המערכת שאוספת המידע, עושה עיבוד בסיסי ושולחת חזרה את המידע
- בשילוב DB אני אכין שרת ואעשה שהנתונים גם ישמרו בשרת
- בשילוב כלים של מערכות הפעלה אני אכניס threading שבאמצעותו הבוט יעבוד על מספר שרתים באותו זמן
- בשילוב UI אכין אתר בסיסי שבו מנהל השרת יוכל לשנות כל מיני אפשרויות של הבוט
- בקטע של האבטחת מידע אכניס encryption בין השרת לבין הבוט

