

פרויקט גמר בסייבר

תיאור והצעת פרויקט

**שם התלמיד: עומר פרי**

**ת"ז: 318444528**

**בית הספר: אורט גוטמן**

**שמות המורים: מיכאל וגד**

**מועד הגשה: 12/2016**

**מבוא**

כאשר עובדים על פרויקט במחשב, מכל סוג, נתקלים לעיתים רבות בבעיות שונות. הבעיות הללו עוברות מחוסר גיבוי במידה ונהרס המחשב, ועד שינוי קטן שנרצה לבטל. מי מאתנו לא נתקל לפעמים בבעיה מסוג זה? לפתע הפרויקט שעבדת עליו כל כך הרבה זמן מפסיק לעבוד ואתה לא מצליח לברר מה הבעיה, אך לצערך לא שמרת את הגרסה האחרונה אשר עבדה. כמו כן, לפעמים נרצה אולי לעבוד על הפרויקט שלנו מעמדת עבודה שונה אך לא נוכל לעשות זאת בשל העובדה שלא שמרנו אותו על דיסק-און-קי, או שזה אינו נמצא בידינו. נוסף על כך, לעיתים נרצה לראות מתי שינינו משהו ספציפי בפרויקט, או שלא נזכור מתי ביצענו את השינוי ונרצה דרך נוחה לגלות זאת.  
  
המוטיבציה לפיתוח התוכנה מורכבת מכמה חלקים. ראשית, לעולם לא נגעתי בתחום הזה וזה נראה לי מאוד מעניין. אמנם קיים כבר כלי אשר נותן לנהל פרויקט בצורה מסודרת ולנהל גרסאות שונות של קבצים, אבל לא קיים כלי כזה שהוא נוח לשימוש וכל בן אדם (גם כזה שלא מבין כלום במחשבים) יוכל להשתמש בו. כמו כן, לא קיים כלי אשר מאפשר ויזואליזציה של הגרסאות. משתי הסיבות שצוינו לעיל הפרויקט הזה יהיה מאתגר ומעניין שכן עוד לא קיים דבר כזה, לכן אאלץ לחשוב על פתרונות לבעיות שעוד לא נתקלו בהן ודרכים לעשות דברים (כמו לדוגמה, איך להפוך את הכלי לאופטימלי) בעצמי מבלי שאוכל להיעזר באחרים.

התוכנה אמורה לאפשר ניהול נוח של פרויקט תוך כדי תמיכה בויזואליזציה של היסטוריית הקבצים והפרויקט ואף נתינת גיבוי של הפרויקט על שרת חיצוני.

**תיאור המוצר**

המטרה המרכזית של המוצר היא לאפשר ניהול נוח ככל האפשר של פרויקטים על המחשב. זאת על ידי גיבוי הפרויקט (ומתן טעינתו מכל מקום), שמירת גרסאות קודמות של קבצים ואף ויזואליזציה של היסטוריית הפרויקט והקבצים.

התרחישים שהמערכת צריכה להתמודד איתם הם:

* רישום והתחברות של משתמש. (ניהול מאגר מידע הכולל את המשתמשים ואת הפרויקטים שלהם)
* משתמש פותח פרויקט, מוסיף / מוחק קבצים, מתחיל גרסה חדשה של קובץ קיים, עורך גרסה ישנה של קובץ קיים וכו'...
* משתמש מבקש להוריד את הפרויקט שלו מהשרת אשר מגבה את הפרויקט בכל שלב.
* משתמש רוצה לראות בצורה ויזואלית את היסטוריית הפרויקט שלו / את ההיסטוריה של אחד הקבצים בפרויקט.

**אילוצים ודרישות**

הבעיות שהמערכת צריכה לדעת להתמודד איתן הן:

1. המערכת צריכה לשמור את הפרויקט לשרת ואף במחשב הלוקאלי מבלי להפריע לעבודת המשתמש.
2. המערכת צריכה להיות נוחה ופשוטה ככל האפשר.
3. מערכת הויזואליזציה צריכה לעבוד בצורה חלקה כך שהמשתמש לא יבחין בתקיעות בזמן השימוש בה.

היא נותנת להן מענה בכך ש:

1. המערכת תשמור כל זמן קצוב / מספר מסוים של שינויים את הקובץ במחשב של המשתמש ובסוף השימוש / בעת בקשת המשתמש היא תעלה את הפרויקט לשרת.
2. למערכת יהיה ממשק מאוד פשוט ומובן.
3. המערכת תטען חלקים מהקובץ בצורה חכמה כך שהשימוש יהיה חלק ככל האפשר.

**תיחום הפרויקט**

הפרויקט משלב את תחום גיבוי הקבצים, ויזואליזציה ותקשורת בין המחשבים.

הפרויקט יעשה שימוש במודולים שיאפשרו לו:

* תקשורת בין מחשבים
* גיבוי וגישה לקבצים
* GUI

**סביבת העבודה**

מירב הפרויקט ייכתב ב Python (התקשורת בין הלקוח לשרת, הטיפול בקבצים) ואילו ה GUI ייכתב בשפת C#.

**ניהול פרויקט עתידי**

|  |  |
| --- | --- |
| **יעדים** | **זמן ביצוע** |
| מחקר לגבי הרכיבים הנדרשים בשביל המערכת | ? |
| הגשת מסמך עיצוב | ? |
| כתיבת שרת ב – Python שמאפשר הרשמה והתחברות של משתמשים (גם כמה במקביל), ולקוח שיכול להתחבר לשרת זה. | ? |
| כתיבת קוד אשר מעביר תיקייה נבחרת לשרת או מוריד את תיקיית הפרויקט שלך מהשרת. | ? |
| כתיבת קוד שיאפשר יצירת מספר גרסאות לכל קובץ ואף מתן גישה לכל גרסה. | ? |
| כתיבת קוד אשר ייתן ויזואליזציה של קבצים. | ? |
| כתיבת דו"ח ביניים. | ? |
| חיבור כל המודולים ביחד, בדיקות ותיקון שגיאות. | ? |
| כתיבת ה- GUI וחיבורו לחלק שנכתב ב- Python. | ? |
| ייעול הויזואליזציה והפיכתה לחלקה יותר. | ? |
| עוד בדיקות ותיקון שגיאות. | ? |
| כתיבת תיק הפרויקט. | ? |
| תוספות ושיפור הפרויקט. | ? |