**PackageDrop**

מסמך אפיון

<תום סימקין>

<גרסה 2.2>

<17.06.2017>

**היסטוריית גרסאות המסמך**

|  |  |  |
| --- | --- | --- |
| **תאריך** | גרסה | **תקציר השינויים** |
| 17.06.2017 | 2.2 | סיום הפרוייקט |
|  |  |  |
|  |  |  |
|  |  |  |



**1. הקדמה**

כיום יש הרבה אתרים בעולם המאפשרים לשמור קבצים אישיים לתוך האתר (כמו דיסק און קי אינטרנטי). המוצר שבניתי הוא אתר מהסוג הזה, אתר המאפשר להעלות קבצים ולשמור אותם על מנת להשתמש בהם בעתיד.

1.1 מטרה

*המטרה היא להסביר את הרעיון שבגללו פותחה התוכנה שבניתי, אילו שירותים היא מספקת ועל אילו צרכים היא עונה.*

קהל היעד של המסמך הוא כל משתמשי האתר. באמצעותו הם יבינו כיצד עובד האתר וכיצד להשתמש בו ולאילו מטרות.

1.2 המוצר

* *שם המוצר - PackageDrop*
* האתר מאפשר להעלות קבצים ולשמור אותם על מנת להשתמש בהם בעתיד.
* *המטרה של המוצר היא להביא למשתמש מקום בו הוא יכול לשמור את קבציו בביטחון. יתרונות האתר הם שסיסמאות המשתמשים שנרשמים לאתר מוצפנים על מנת למזער את הסיכוי לפריצתם וגם קאפצ'ה הנועדה להרחיק מתקפות אינטרנטיות.*

1.3 הגדרות

**GUI - Graphical User Interface,** הממשק הגרפי שאותו רואה המשתמש במהלך השימוש במוצר (דף אינטרנט במקרה זה).

**Database** – מאגר מידע, שומר את נתוני המשתמשים (שם משתמש וסיסמא).

**Flask** – פלאסק הוא מסגרת מיקרו בשביל פייתון המבוסס על עוד מסגרות מיקרו כגון **Werkzeug** ו **Jinja 2** המאפשר לחסוך את העבודה של "מאחורי הקלעים" בשביל המתכנתים (כגון בקשות GET) ומאפשר צפייה בתבניות HTML.

**SQL** - היא שפת מחשב הצהרתית לטיפול ועיבוד מידע בבסיסי נתונים יחסיים, שפותחה על ידי IBM, והתבססה במקור על אלגברה רלציונית. השפה מאפשרת שליפת נתונים ועדכונם ויצירת טבלה ושינויה.

**1.4** תקציר

*במסמך יש תיאור כללי של המוצר: מי קהל היעד שלו, אילוצים וכדומה.*

כמו כן המסמך מתאר את כל הדברים שהמוצר צריך לעשות.

1. תיאור כללי

2.1 פונקציונליות

- **המוצר מאפשר למשתמש להתחבר לאתר אם כבר קיים המשתמש, אחרת, הוא יכול ליצור משתמש חדש.**

**- המוצר מאפשר למשתמש להעלות את קבציו לאתר.**

**- המוצר מאפשר למשתמש להוריד את קבציו בחזרה למחשב.**

2.2 קהל היעד

**קהל היעד של המוצר הוא כל אדם אשר רוצה לשמור את קבציו באינטרנט.**

2.3 אילוצים עיקריים

*על המחשב צריך להיות מותקן דפדפן שיאפשר לו להיכנס לאתר והמוצר צריך לרוץ במערכת הפעלה WINDOWS.*

2.4 הנחות ותלויות

**כאשר המשתמש מעלה קובץ לאתר, העיצוב של האתר משתנה (רואים עוד קובץ שנמצא באתר).**

3.1 דרישות פונקציונליות

*תיאור של פיצ'רים ספציפיים במערכת. לכל דרישה פונקציונלית מומלץ לפרט:*

* *הסבר כללי – האתר יאפשר למשתמשים להעלות, לשמור ולהוריד את הקבצים שלהם.*
* *קלט – האתר קולט מן המשתמשים את השם משתמש והסיסמא שלהם כאשר הם נרשמים/נכנסים לאתר וגם את קבציהם כשהם מעלים אותם לאתר.*
* *תהליך – אם לא קיים משתמש, האדם יוצר אותם כאשר הוא לוחץ בדף הראשי של האתר על כפתור ה SIGN UP. כפתור זה מעביר את האדם לדף ההרשמה. שם האדם צריך להזין את פרטיו (שם משתמש, סיסמא, אימות סיסמא) וללחוץ על כפתור הקאפצ'ה שיאמת שזהו אדם. לאחר מכן הוא מועבר לדף ההעלאות שם הוא יוכל להעלות את קבציו. אם לאדם קיים משתמש, הוא יכול ללחוץ על כפתור ה LOG IN שיעביר אותו לדף ההתחברות. שם הוא יזין את פרטי המשתמש שלו ולאחר מכן יועבר לדף ההעלאות שם הוא יוכל להעלות את קבציו לשרת. יש גם אפשרות להוריד את קבציו למחשב ע"י לחיצת כפתור DOWNLOAD שיפנה אותו לראות את כל קבציו שהוא העלה. ע"י לחיצה על אחד הקבצים, זה יוריד אותו למחשב. כמובן שהמשתמש יכול להתנתק מהאתר ע"י לחיצת כפתור ה LOGOUT המצוי בדף ההעלאה וגם בדף ההורדה.*
* *פלט + טיפול בשגיאות – אם האדם שוגה בפרטי המשתמש שלו בדף ה LOG IN, השרת מחזיר לו שיש טעות בשם המשתמש שלו או בסיסמא שלו. בדף ה SIGN UP, אם השם משתמש תפוס, יוצא פלט שאומר שהוא תפוס, אם השם משתמש קצר מדי, יוצא פלט שאומר שהוא קצר מדי, אם האדם השתמש באותיות שלא נמצאות בשפה האנגלית ובדמויות לא מקובלות, יוצא פלט שאומר שאפשר להשתמש רק באותיות אנגליות ובספרות. אם הסיסמא שלו קצרה מדי, יוצא פלט שהסיסמא קצרה מדי. אם הסיסמא לא תואמת את האימות סיסמא, יוצא פלט שאומר שהסיסמאות צריכות להיות שוות. אם הקאפצ'ה לא נכונה/לא סומנה יוצא פלט שאומר שהקאפצ'ה לא נכון.*

3.2 דרישות של ממשקים חיצוניים

**3.2.1 ממשקי משתמש** (למשל: GUI) – ה GUI של המוצר הוא אתר אינטרנט ולכן חייב להיות דפדפן מותקן על המחשב.

**3.2.4 ממשקי תקשורת (למשל תקשורת בין שר**ת ללקוחות) – המחשב צריך להיות מחובר לאינטרנט.

3.3 דרישות לא פונקציונליות

*דרישות מהמערכת שלא מתבטאות בפיצ'ר ספציפי או בתהליך ספציפי שמתרחש במערכת אבל משפיעות על אופן עיצובה ומימושה, לדוגמא:*

**3.3.1 דרישות ביצועים (performance)** – האתר צריך לבצע את כל הפעולות של הפלט ובאותו הזמן להמשיך את החיבור בין השרת ללקוח.

**3.3.3 דרישות זמינות (Availability)** – האתר זמין כל עוד השרת פועל

**3.3.4 דרישות אבטחה (security)** – הסיסמא של המשתמש מוצפנת, יש קאפצ'ה ,כדי לגשת לקוד צריך מפתח סודי והקבצים שמועלים לשרת מוצפנים.

3.4 דרישות בסיס נתונים

* *הנתונים שיישמרו הם קבצי המשתמש*
* *פרטי המשתמש שמורים ב DB וקבציו שמורים בתיקייה ייחודית משלו.*
* *נפח האחסון הוא אינסופי והקובץ יכול להיות בגודל של עד mb16.*
* *רמת האבטחה הנדרשת היא גבוהה, לכן הסיסמאות מוצפנות, הקבצים מוצפנים ויש קאפצ'ה שהחיבור שלה עם השרת ועם דפי האתר הוא מוצפן. בנוסף לכך, על מנת להשתמש בקוד צריך מפתח סודי ויש שימוש ב SQL שקשה יותר לעשות SQLINJECTION.*

3.5 דרישות נוספות

*אין*

5. נספחים

**אין**