**מסמך אפיון (עדיין בתהליכי כתיבה//)- מערכת הצפנת מסמכי Word ו-PDF(DocLock)**

בס"ד

1. **תיאור המערכת**

המערכת מאפשרת הצפנה ופענוח של מסמכים בפורמט Word ו-PDF, בהתבסס על קוד סודי שהמשתמש מזין. הקבצים המוצפנים יאוחסנו בענן, וניתן יהיה לפתוח ולצפות בהם רק בתוך המערכת לאחר הזנת הקוד הנכון. המערכת תתמוך בניהול משתמשים, אחסון מאובטח, וניהול הרשאות צפייה וגישה לקבצים. ההצפנה תתבצע באמצעות אלגוריתמי הצפנה חזקים תוך מימוש DRM , תוכנו הבינארי של הקובץ יקודד על מנת להקשות על הפריצה, הקובץ יתאפשר לקריאה רק למשתמש שמזין את הקוד הנכון.

1. **מטרות המערכת**

* הצפנה ופענוח של קבצים
* אחסון מאובטח בענן
* ניהול משתמשים והרשאות
* גישה מאובטחת וממשק משתמש אינטואיטיבי

1. **קהל יעד**

* אנשי עסקים
* משתמשים פרטיים
* מפתחים
* מוסדות אקדמיים
* ארגונים ממשלתיים

1. **עיצוב ממשק משתמש ויזואלי**

* מסך התחברות והרשמה: שדות להזנת דוא"ל, סיסמה ואימות סיסמה.
* מסך ראשי: רשימת קבצים מוצפנים ופתוחים, עם אפשרויות העלאה, מחיקה ותיוג.
* מסך צפייה בקובץ: תצוגת מסמך לאחר הזנת קוד נכון, ללא אפשרות להורדה.
* תפריט ראשי: ניווט בין מסכי המערכת, ניהול משתמשים, יציאה מהחשבון.

המערכת פונה בעיקר למי שזקוק לשמירה על פרטיות ואבטחת מידע במסמכים דיגיטליים, תוך שמירה על גישה מאובטחת וניהול הרשאות מתקדמות.

1. **דרישות המערכת**

**5.1 דרישות פונקציונליות**

1. **התחברות והרשמה**:
   * המשתמשים יוכלו להירשם למערכת באמצעות הזנת כתובת דוא"ל, סיסמה ואימות סיסמה.
   * המשתמשים יוכלו להתחבר למערכת באמצעות דוא"ל וסיסמה.
2. **ניהול קבצים**:
   * המשתמשים יוכלו להעלות קבצים מסוג Word ו-PDF למערכת.
   * כל קובץ יישמר עם מפתח הצפנה ייחודי שנוצר עבורו.
   * המשתמשים יוכלו לצפות בקבצים מוצפנים בתנאי שהזינו את הקוד הנכון.
   * לא תינתן אפשרות להוריד את הקבצים המוצפנים, אלא רק לצפות בהם בתוך המערכת.
   * המשתמשים יוכלו למחוק קבצים שהעלו או לעדכן את פרטי הקובץ (כגון שם או תיוגים).
3. **ניהול משתמשים**:
   * המערכת תאפשר למנהלים להוסיף, לעדכן ולמחוק משתמשים.
   * המערכת תומך בהגדרת תפקידים (כגון משתמש רגיל, מנהל) עם הרשאות שונות.
   * המערכת תאפשר למנהלים לנהל את ההרשאות של המשתמשים בנוגע לגישה לקבצים.
4. **הצפנה ופענוח**:
   * המערכת תבצע הצפנה א-סימטרית עבור קבצים (לפי קוד המשתמש).
   * כל קובץ יהיה מוצפן עם מפתח הצפנה ייחודי שנשמר מאובטח.
   * כאשר המשתמש יזין את הקוד הנכון, המערכת תבצע פענוח של הקובץ ותאפשר צפייה בו.
5. **תצוגת קבצים**:
   * קבצים שנפתחו יהיו מוצגים על גבי מסך הצפייה בממשק, ולא ניתן יהיה להורידם.
   * תהיה אפשרות להציג את תוכן הקובץ בממשק הקריאה, כולל כל עיצוב או פורמט (לדוגמה: תמונות בטקסט).
6. **אחסון בענן**:
   * המערכת תומכת באחסון קבצים בענן, באמצעות שירותי AWS S3 או Azure Blob Storage. או render
   * הקבצים יאוחסנו בענן עם הצפנה ברמה גבוהה.
7. **אינטגרציה עם API**:
   * המערכת תספק API לניהול קבצים, משתמשים והגדרת הרשאות.
   * כל הגישה לנתונים תתבצע באמצעות HTTPS כדי להבטיח אבטחת מידע.

**5.2 דרישות לא פונקציונליות**

1. **אבטחת מידע**:
   * המערכת תשתמש בהצפנה א-סימטרית כדי להבטיח שמירה על פרטיות המידע.
   * כל הגישה לנתונים תתבצע דרך HTTPS כדי להבטיח חיבור מאובטח.
   * המערכת תבצע ניהול הרשאות גישה על מנת למנוע גישה לא מורשית לקבצים.
   * הסיסמאות של המשתמשים יהיו מוצפנות בבסיס הנתונים באמצעות אלגוריתמים מודרניים.
2. **ביצועים**:
   * המערכת תספק תגובה מהירה בעת העלאת קבצים, צפייה בהם וניהול הרשאות.
   * המערכת תהיה מסוגלת להתמודד עם אחסון של מאות אלפי קבצים, תוך שמירה על ביצועים גבוהים.
   * קבצים ייטענו במהירות, בזכות שימוש ב-CDN להאצת טעינת קבצים.
3. **קנה מידה**:
   * המערכת תוכל להתרחב בקלות כדי לתמוך בכמות משתמשים גבוהה ובכמויות גדולות של קבצים.
   * תתמוך בהוספת יכולות נוספות בעת הצורך, כגון שילוב של AI לניהול גישה חשודה.
4. **שימושיות**:
   * המערכת תהיה נוחה לשימוש, עם ממשק משתמש אינטואיטיבי וברור.
   * המערכת תכלול תיעוד למשתמשים ולמפתחים, עם הסברים ברורים לשימוש בפונקציות השונות.
5. **תחזוקה**:
   * המערכת תהיה קלה לתחזוקה, עם קוד מסודר ומודולרי.
   * המערכת תכלול אוטומציה לעדכונים והתקנות כדי לשפר את ביצועי המערכת לאורך זמן.
6. **ארכיטקטורת המערכת וטכנולוגיות**

המערכת מבוססת על ארכיטקטורת Full Stack עם .NET API בצד השרת, Angular למסך ההתחברות, ו-React לצד הלקוח.

* צד שרת: .NET Minimal API לניהול משתמשים, פרויקטים וניתוחים.
* מסך התחברות: Angular עם עיצוב Material-UI.
* צד לקוח: React עם דף לוח פרויקטים.
* מסד נתונים: MySQL לניהול משתמשים וקבצים.

**3.1 תתי-פרויקטים**

1. **צד שרת - .NET Minimal API**  
   • פיתוח REST API לניהול משתמשים, פרויקטים וניתוחי כתב יד.  
   • שימוש ב-Entity Framework לניהול מסד הנתונים (MySQL).  
   • מנגנוני אימות משתמשים (JWT Authentication).  
   • תמיכה בהעלאת קבצים ושמירתם בענן.  
   • אבטחת מידע ושמירה על פרטיות הנתונים.
2. **מסך התחברות - Angular**  
   • דף התחברות מאובטח עם JWT.  
   • הרשאות גישה בהתאם לרמות משתמשים (מנהלים, לקוחות).  
   • עיצוב עם Material-UI להתאמה למובייל ודסקטופ.
3. **צד לקוח - React**  
   • תצוגת לוח פרויקטים עם רשימת הניתוחים הקיימים.  
   • מסך "Create New" להעלאת קובץ להצפנה/צפייה באמצעות הזנת קוד על ידי המשתמש
4. **מסד נתונים - MySQL**  
   • טבלת Users לניהול משתמשים והרשאות.  
   • טבלת Files לניהול הקבצים של כל משתמש.
5. **תרשים מבנה הטבלאות**

**טבלת משתמשים**

|  |  |  |  |
| --- | --- | --- | --- |
| **עמודה** | **סוג** | **מאפיינים** | **תיאור** |
| **id** | **INT** | **PRIMARY KEY, AUTO\_INCREMENT** | **מזהה ייחודי לכל משתמש** |
| **full\_name** | **VARCHAR(255)** | **NOT NULL** | **שם מלא של המשתמש** |
| **email** | **VARCHAR(255)** | **UNIQUE, NOT NULL** | **כתובת דוא"ל ייחודית** |
| **password** | **VARCHAR(255)** | **NOT NULL** | **סיסמה מוצפנת** |
| **role** | **ENUM('admin', 'user')** | **NOT NULL** | **תפקיד המשתמש** |
| **created\_at** | **TIMESTAMP** | **DEFAULT CURRENT\_TIMESTAMP** | **תאריך יצירת המשתמש** |

**טבלת קבצים**

|  |  |  |  |
| --- | --- | --- | --- |
| **עמודה** | **סוג** | **מאפיינים** | **תיאור** |
| **id** | **INT** | **PRIMARY KEY, AUTO\_INCREMENT** | **מזהה ייחודי לקובץ** |
| **user\_id** | **INT** | **FOREIGN KEY** | **מזהה המשתמש שהעלה את הקובץ** |
| **file\_name** | **VARCHAR(255)** | **NOT NULL** | **שם הקובץ** |
| **file\_path** | **TEXT** | **NOT NULL** | **נתיב האחסון בענן** |
| **encrypted\_key** | **TEXT** | **NOT NULL** | **מפתח הצפנה ייחודי** |
| **created\_at** | **TIMESTAMP** | **DEFAULT CURRENT\_TIMESTAMP** | **תאריך העלאת הקובץ** |
| **//להגדיר פרטי מחלקת קובץ מוצפן** | **מחלקה** |  | **פרטי קובץ מוצפן** |

1. **ממשק משתמש (UI)**

**1. מסך התחברות והרשמה:**

* שדות התחברות:
  + שדה להזנת דוא"ל (Email).
  + שדה להזנת סיסמה (Password).
  + כפתור "התחבר" (Login).
* שדות הרשמה (למשתמש חדש):
  + שדה להזנת דוא"ל (Email).
  + שדה להזנת סיסמה (Password).
  + שדה לאימות סיסמה (Confirm Password).
  + כפתור "הירשם" (Sign Up).
* קישור לשכחת סיסמה: יאפשר למשתמש לאפס את סיסמתו במקרה של אובדן.
* אופציה להתחברות באמצעות שירותים חיצוניים: כמו Google/Facebook (אופציונלי(.

**2. מסך ראשי:**

* רשימת קבצים:
  + הצגת כל הקבצים המוצפנים והפתוחים של המשתמש.
  + לכל קובץ יופיע שם הקובץ, תאריך העלאה, וסטטוס הצפנה.
  + אפשרות למחוק קובץ או להוסיף תיוגים לקובץ.
* כפתור העלאת קובץ: מאפשר למשתמש להעלות קבצים מסוג Word/PDF.
* תפריט ניווט:
  + גישה לעמוד ניהול משתמשים (עבור מנהלים).
  + גישה למסך ההגדרות.
  + כפתור יציאה (Logout).

**3. מסך צפייה בקובץ:**

* תצוגת קובץ: הצגת תוכן הקובץ המוצפן לאחר הזנת קוד הצפנה נכון.
* כפתור חזרה למסך הראשי: מאפשר חזרה למסך הראשי.
* ממשק קריאה ברור: הצגת תוכן הקובץ בצורה נוחה עם עיצוב ברור וקריא (למשל, תמונות או טבלאות אם יש).

**4. מסך ניהול קבצים:**

* העלאת קובץ: כפתור להעלאת קבצים חדשים למערכת.
* מחיקת קובץ: אפשרות למחוק קבצים קיימים.
* תיוגים: הוספת תיוגים לקבצים לצורך סינון וניהול.
* סינון קבצים: אפשרות לסנן את הקבצים לפי שם, תאריך, תיוגים ועוד.

**5. מסך ניהול משתמשים (עבור מנהלים):**

* הוספת משתמש: כפתור להוספת משתמשים חדשים למערכת.
* עדכון פרטי משתמש: אפשרות לעדכן פרטים של משתמשים קיימים.
* מחיקת משתמש: אפשרות למחוק משתמשים.
* ניהול הרשאות: אפשרות להקצות תפקידים (משתמש רגיל/מנהל) ולנהל את הרשאות הגישה.

**6. תפריט ראשי:**

* ניווט בין המסכים השונים במערכת: התחברות, ניהול קבצים, ניהול משתמשים (עבור מנהלים), הגדרות.
* כפתור יציאה מהחשבון.
* גישה למסך הגדרות המערכת.

1. **עתיד הפיתוח**
2. **שיפור חוויית משתמש (UI/UX):**
   * עיצוב רספונסיבי: התאמה מלאה לכל המכשירים (מובייל, טאבלט, מחשבים שולחניים) עם ממשק אינטואיטיבי וידידותי למשתמש.
   * הוספת עיצוב מתקדם: שילוב של אנימציות ומעברים חלקים כדי לשפר את חוויית השימוש.
   * שיפור תמיכה בשפות מרובות: הוספת תמיכה בשפות נוספות להנגשה לעולמות בינלאומיים (כגון אנגלית, ספרדית, צרפתית).
3. **שילוב טכנולוגיות נוספות:**

**אפשרויות להטמעת AI:**

* + AI לזיהוי סוגי מסמכים: פיתוח אלגוריתמים חכמים לזיהוי אוטומטי של סוגי מסמכים (חוזים, דוחות כספיים, מסמכים משפטיים).
  + מעקב אחרי דפוסי השימוש של המשתמשים.
  + ניתוח ניסיונות כושלים להזנת קוד.
  + שימוש במודל למידת מכונה לזיהוי התנהגות חריגה ומניעת גישה לא מורשית בזמן אמת.

AI-Assistant: אם המשתמש שוכח קוד, אפשר לשלב מודל AI שמציע רמזים לפי נתונים שהוזנו בעבר.

שיפור הצעות הרשאות גישה: המערכת תוכל להמליץ למי לשתף מסמך על בסיס שימוש קודם.

בדיקת חולשות בהצפנה על ידי מודלים של AI Security Analysis.

1. **פיצ'רים מתקדמים:**
   * תמיכה בניהול קבצים משותפים: אפשרות לשתף קבצים עם משתמשים אחרים תוך שמירה על הצפנה מאובטחת.
   * הוספת חיפושים חכמים: אפשרות לחיפוש מתקדם בתוך הקבצים (כגון חיפוש לפי תוכן מסמך).
   * תמיכה בקבצים נוספים: הוספת תמיכה בהצפנת קבצים מסוגים נוספים (כגון Excel, PowerPoint).
2. **שיפור אבטחת מידע:**
   * הצפנה מתקדם: שדרוג אלגוריתמים לאבטחה גבוהה יותר, כמו הצפנה עם מפתחות יותר ארוכים ואלגוריתמים א-סימטריים מתקדמים.
   * אימות דו-שלבי: הוספת תמיכה בהגנה נוספת על ידי אימות דו-שלבי בעת התחברות למערכת.
3. **הרחבת יכולות הענן:**
   * שיפור תמיכה ב-CDN: כדי לשפר את מהירות טעינת הקבצים ויכולת התמודדות עם עומסים גדולים של משתמשים.
   * פריסת המערכת על שירותי ענן נוספים: יכולת להרחיב את התמיכה בענן עם שירותים נוספים כמו Google Cloud או IBM Cloud.
4. **מעבר לענן:**
   * שדרוג יכולות השיתוף והגישה בזמן אמת באמצעות פתרונות ענן מתקדמים, מה שיאפשר למשתמשים לגשת לקבצים בצורה חלקה יותר.
5. **הגדרת סבבי פיתוח**

**סקירת תכנון ספרינט**  
כל ספרינט נמשך בדרך כלל 2-4 שבועות, והמטרה היא לספק תוספת מוצר שניתן להוציא לשוק בסוף כל ספרינט. להלן פירוט מוצע:

**ספרינט 1: הקמת סביבת פיתוח ופיתוח מסך הכניסה**

* יעד כללי: הקמת סביבה פיתוחית עובדת ותחילת פיתוח מסך הכניסה לאפליקציה.
* **משימות:**
  + הגדרת סביבת עבודה: התקנת כלים נדרשים (IDE, Docker, ניהול גרסאות).
  + יצירת פרויקט בסיסי ב-React/Angular.
  + פיתוח מסך הכניסה עם אפשרות להזין שם משתמש וסיסמה.
  + חיבור למערכת ניהול משתמשים בסיסית (למשל, באמצעות JWT).

**ספרינט 2: פיתוח ממשק ראשי וניהול משתמשים**

* יעד כללי: פיתוח הממשק הראשי של האפליקציה והתחלת פיתוח מערכת ניהול משתמשים.
* **משימות:**
  + עיצוב ופיתוח ממשק ראשי עם ניווט בין דפים.
  + חיבור למערכת ניהול משתמשים (הכנסת פרטי משתמשים ותחום ניהול המידע).
  + חיבור למסד נתונים (למשל MySQL/PostgreSQL) לצורך שמירת פרטי משתמשים.

**ספרינט 3: פיתוח מנגנון הצפנה והעלאת קבצים**

* יעד כללי: פיתוח מנגנון הצפנה עבור קבצי Word ו-PDF.
* **משימות:**
  + פיתוח אפשרות להעלות קובץ (Word/PDF) למערכת.
  + פיתוח מנגנון הצפנה עם קוד משתמש (הצפנה על בסיס קוד משתמש).
  + יצירת קישור להורדת הקובץ המוצפן לאחר ההעלאה.

**ספרינט 4: הצגת הקובץ המוצפן ואימות סיסמה**

* יעד כללי: פיתוח מערכת להצגת קבצים מוצפנים ואימות סיסמה.
* **משימות:**
  + פיתוח מנגנון להצגת קובץ מוצפן תוך כדי הצגת הודעה על הצורך בהזנת סיסמה.
  + פיתוח מערכת לאימות סיסמה ולפתיחת הקובץ המוצפן לצפייה בלבד.
  + חיבור לקוד ההצפנה ולמערכת ההורדה והצפייה.

**ספרינט 5: אינטגרציה עם AI (אפשרי)**

* יעד כללי: פיתוח אינטגרציה ראשונית עם AI לצורך שיפור המערכת.
* **משימות:**
  + חיבור למערכת AI (למשל, לצורך זיהוי של תבניות בטקסטים או סיווג קבצים).
  + יצירת ממשק אינטגרציה עם יכולות AI בשלב הראשון (למשל, ביצוע ניתוחים ראשוניים של הקבצים).

**ספרינט 6: בדיקות ושיפור המערכת**

* יעד כללי: ביצוע בדיקות
  + ביצוע בדיקות על הצפנה ופתיחה של קבצים.
  + תיקון באגים ושיפורים לפי תוצאות הבדיקות.
  + שיפור ביצועים והכנת האפליקציה לעבודה בסביבה פרודקשן.

**תהליך פיתוח ספירלי כזה מאפשר לא רק להגיע לתוצאה סופית, אלא גם לשפר ולהתאים את המערכת לאורך כל הדרך.**