**EduShare**

**אפיון מערכת לניהול חומרי לימוד דיגיטליים**

**תיאור כללי של המערכת**

המערכת מאפשרת למורים לנהל ולארגן חומרי לימוד בצורה דיגיטלית, כולל העלאת קבצים (מצגות, סיכומים מוקלדים, PDF, Word), עיבוד ושמירת המידע מהקבצים, ושיתוף החומרים עם אחרים. בנוסף, המערכת תכלול ממשק מנהל לניהול משתמשים ולצפייה בנתונים.

**מטרות המערכת:**

1. לאפשר למורים להעלות ולנהל חומרי לימוד בקלות.
2. לאפשר הפקת מידע מתוך קבצים (סיכומים סרוקים, מצגות וכו').
3. לשמור מידע מהקבצים באופן מובנה ונגיש.
4. לאפשר שיתוף חומרים עם מורים אחרים.
5. לספק ממשק מנהל לניהול משתמשים וניהול תוכן.

**מבנה המערכת**

**1. ממשקי משתמש:**

* **ממשק למורים (React)** – מאפשר העלאת חומרים, ניהול שיעורים ועריכת מידע שהופק מהקבצים.
* **ממשק למנהל (Angular)** – מאפשר צפייה וניהול משתמשים, הסרתם וניהול נתוני המערכת.

**2. Backend:**

* **API ראשי (Minimal API ב-.NET Core)** – אחראי על ניהול משתמשים, חומרי לימוד ושיעורים.
* **שרת AI לעיבוד קבצים (Python)** – מקבל קבצים, מנתח את המידע ושולח תוצאות מעובדות ל-DB.

**3. מסד נתונים:**

* שוקל להשתמש ב-Clever Cloud לניהול הנתונים.

**4. חיבור לענן:**

* שימוש באחסון בענן (AWS או שירות אחר) לשמירת הקבצים.

**מבנה בסיס הנתונים**

**טבלת Users – ניהול משתמשים**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| Id | int (PK) | מזהה משתמש ייחודי |
| Name | varchar | שם מלא |
| Email | varchar | כתובת אימייל |
| PasswordHash | varchar | סיסמה מוצפנת |
| Role | varchar | תפקיד (מורה/מנהל) |

**טבלת Lessons – ניהול שיעורים**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| Id | int (PK) | מזהה שיעור ייחודי |
| Title | varchar | כותרת השיעור |
| Subject | varchar | נושא השיעור |
| Level | varchar | רמת השיעור |
| UserId | int (FK) | מזהה המורה שיצר את השיעור |

**טבלת Materials – ניהול חומרי לימוד**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| Id | int (PK) | מזהה חומר לימוד ייחודי |
| LessonId | int (FK) | מזהה השיעור אליו שייך החומר |
| Title | varchar | כותרת החומר |
| Type | varchar | סוג החומר (PDF, מצגת, Word) |
| FileUrl | varchar | קישור לקובץ בענן |
| ProcessedData | text | מידע מעובד שהופק מהקובץ |

**טבלת SharedMaterials – שיתוף חומרים בין מורים**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| Id | int (PK) | מזהה שיתוף ייחודי |
| MaterialId | int (FK) | מזהה החומר המשותף |
| SharedWith | int (FK) | מזהה המשתמש עמו החומר שותף |

**תהליכים מרכזיים במערכת**

**1. העלאת קובץ**

1. המורה מעלה קובץ דרך הממשק ב-React.
2. הקובץ נשמר בענן ונשלח לשרת ה-AI לעיבוד.
3. השרת מחלץ מידע רלוונטי ושומר אותו ב-DB.

**2. יצירת שיעור והוספת חומרי לימוד**

1. המורה יוצר שיעור חדש ומוסיף לו חומרים.
2. ניתן להעלות קובץ חדש או לבחור חומר קיים ממאגר החומרים.
3. ניתן לערוך את המידע המעובד לפני השמירה.

**3. שיתוף חומרים**

1. המורה בוחר חומר לימוד ומשתף אותו עם מורה אחר.
2. המערכת יוצרת רשומה בטבלת SharedMaterials.
3. המורה המקבל יכול לגשת לחומר ולנהל אותו.

**4. ניהול משתמשים (מנהל בלבד)**

1. המנהל יכול לצפות ברשימת המשתמשים ולבצע פעולות כמו חסימה או מחיקה.
2. המנהל יכול להוסיף משתמשים חדשים ידנית.

**סיכום**

המערכת נותנת פתרון מלא לניהול חומרי לימוד בצורה דיגיטלית, כולל העלאת קבצים, ניתוח המידע שלהם, שמירה נוחה במאגר ושיתוף עם משתמשים אחרים. ממשק המורה ב-React מספק נוחות וקלות שימוש, בעוד שממשק המנהל ב-Angular נותן שליטה על כל המערכת.

תכנון הזמן/ספרינטים

### שבוע 1: תכנון ובניית החלקים הבסיסיים

#### יום 1-2: תכנון ארכיטקטורת המערכת

* **הבנת הדרישות והאפיון:** עיון במערכת הקיימת והבנת הדרישות ממנה.
* **תכנון מאגר הנתונים:** יצירת תרשימי ER (Entity-Relationship) שמפרטים את כל הטבלאות, הקשרים והכוחות של המערכת.
* **תכנון ממשקים:** חשיבה על עיצוב ה-UI למורים ולמנהל, סידור רכיבי הממשק בעדיפות גבוהה.
* **הגדרת שיטות עבודה:** כיצד ייראה תהליך העלאת קבצים, חיבור לשרת ה-AI והעברת המידע לבסיס הנתונים.

#### יום 3-4: הגדרת סביבת פיתוח

* **הכנה של VS Code ל-React ו-Angular:** התקנה של כל הכלים הדרושים, יצירת פרויקטים ריקים עבור שני הממשקים.
* **הכנה של .NET Core API:** יצירת API בסיסי באמצעות Minimal API, הגדרת קונפיגורציות ראשוניות.

#### יום 5-7: פיתוח ממשק המורה ב-React

* **יצירת עמודים בסיסיים:** יצירת דפים בסיסיים כמו דף הבית, דף העלאת קבצים ודף ניהול שיעורים.
* **שילוב API עם React:** התחברות ל-API הראשי ב-.NET Core לשליחת בקשות ולקבלת תשובות.
* **תהליך העלאת קבצים:** יצירת רכיב שיעזור למורים להעלות קבצים ולקבל את המידע מהקובץ.

### שבוע 2: פיתוח ופתרון בעיות

#### יום 8-10: פיתוח הממשק המנהל ב-Angular

* **יצירת דפים וממשקים:** יצירת דפי ניהול משתמשים ודפי הצגת נתונים.
* **שליטה במשתמשים:** אפשרות להוסיף, למחוק או לחסום משתמשים.
* **חיבור ל-API:** חיבור ממשק Angular ל-API, כולל פעולות CRUD (יצירה, קריאה, עדכון ומחיקה) על משתמשים.

#### יום 11-12: פיתוח ה-Backend (Minimal API)

* **פיתוח פונקציות ניהול משתמשים:** יצירת מסלול עבור ניהול משתמשים (הוספה, עדכון, הסרה).
* **פיתוח פונקציות לניהול שיעורים:** יצירת מסלול ליצירת שיעור והעלאת חומרי לימוד אליו.
* **חיבור למאגר הנתונים:** הגדרת החיבורים למסד הנתונים (Clever Cloud) והתחלת בניית פעולות CRUD על טבלאות ה-Users וה-Lessons.

#### יום 13-14: שרת AI ועיבוד קבצים

* **בניית השרת ב-Python:** יצירת API ב-Python עבור עיבוד קבצים (OCR, ניתוח טקסטים, וכדומה).
* **שליחת קבצים לשרת AI:** יצירת חיבור בין הממשק ב-React לשרת ה-Python לעיבוד קבצים.
* **שמירת המידע בבסיס נתונים:** השרת שולח את המידע המעובד ל-DB (כמו טקסט מתוך PDF) אחרי עיבוד.

### שבוע 3: פיתוח מתקדם, חיבור לענן ושיתוף חומרים

#### יום 15-16: חיבור לענן (AWS)

* **הגדרת שירותי AWS:** הכנה של שירותים כגון S3 לשמירת קבצים בענן.
* **חיבור המערכת לאחסון:** יצירת חיבור בין המערכת לשירותי AWS לשמירת הקבצים שהועלו על ידי המורים.
* **העלאת קבצים לענן:** ייבוא ושמירה של קבצים בענן, כולל הוצאת כתובת URL עבור כל קובץ.

#### יום 17-18: פיתוח פונקציות שיתוף חומרים

* **יצירת פונקציה לשיתוף חומרי לימוד:** יצירת מערכת שתאפשר למורים לשתף חומרים עם מורים אחרים.
* **בקרת שיתוף:** יצירת ממשק ב-React שמאפשר למורים לבחור את החומרים ולשתף אותם עם מורים אחרים.
* **שמירת שיתוף ב-DB:** תהליך יצירת רשומות בטבלת SharedMaterials כשחומר לימוד משותף.

#### יום 19-21: שדרוג UI ופתרון בעיות

* **שדרוג ממשק המורה והמנהל:** עיצוב חווית משתמש בצורה שתהיה יותר נוחה וממוקדת.
* **בקרת שגיאות:** בדיקת המערכת לאיתור בעיות ותיקון באגים (לדוגמה: בעיות בהעלאת קבצים, עיבוד לא נכון של מידע).
* **ביצוע בדיקות אינטגרציה:** חיבור בין כל החלקים, ביצוע בדיקות של העלאת קבצים, שיתוף חומרים, ניהול משתמשים.
* **חיבור עם AWS:** סיום החיבור למאגר הנתונים בענן ולהעלאת קבצים.

#### סיום:

* **הגדרת גישה למערכת:** אפשרות למורים ומנהלים להתחבר למערכת ולבצע את כל הפעולות הנדרשות.
* **בדיקות סופיות:** ביצוע בדיקות שימושיות וסופיות לוודא שהמערכת פועלת כמצופה.

בהצלחה ענקית!!!!

והמון המון ס"ד!!!