**אפיון מערכת: ניהול חומרים ושיעורים במכללה**

**1. תיאור המערכת**

מערכת "ניהול חומרים ושיעורים במכללה" מאפשרת למורים במכללה לנהל את המקצועות והשיעורים שלהם בצורה יעילה ומסודרת. במהלך כל שיעור, המורה יכול לפתוח את השיעור, לסמן נוכחות של סטודנטים ולנהל את תכני ההוראה בצורה דיגיטלית. בסיום השיעור, המורה מעלה את כל הקבצים והקוד שנלמדו (כגון מצגות, דפי עבודה, ,קבצי קוד, וכו'), והמערכת מעבירה אוטומטית את הקבצים לאזור האישי של כל סטודנט. בנוסף, המערכת משלבת בינה מלאכותית לתמלול אוטומטי של הקבצים המועלים ולזיהוי תוכן חשוב, ומספקת עדכונים בזמן אמת.

**2. מטרת הפרויקט**

**מטרות כלליות:**

* **יצירת פלטפורמה מרכזית:**  
  לספק מערכת אחידה לניהול חומרים ושיעורים למורים במכללה, במטרה לייעל את תהליכי ההוראה והלמידה.
* **ניהול חכם ונגיש של השיעורים:**  
  לאפשר למורים לנהל שיעורים בצורה נגישה, מסודרת ואוטומטית – החל מהפעלת השיעור, דרך ניהול נוכחות ועד העברת חומרי הלימוד לסטודנטים.
* **שילוב כלים טכנולוגיים מתקדמים:**  
  להטמיע טכנולוגיות כגון תמלול אוטומטי באמצעות AI וזיהוי תוכן חשוב, לצד עדכונים בזמן אמת, לשיפור חוויית המשתמש והיעילות התפעולית.
* **ניהול גישה והרשאות:**  
  להבטיח גישה מבוקרת לתוכן באמצעות הרשאות שונות, בהתאם לתפקיד המשתמש (מורה או סטודנט).

**3. קהל היעד ומטרות המשתמשים**

**3.1 קהל היעד:**

* **מורים במכללות:**  
  המעוניינים לנהל את השיעורים והחומרים בצורה מסודרת ולייעל את תהליכי ההוראה.
* **סטודנטים במכללות:**  
  המעוניינים לקבל גישה נוחה ומסודרת לחומרי הלימוד, לצפות בסיכומי השיעורים ולקבל את כל הקבצים שהועלו על ידי המורה באופן אוטומטי.

**3.2 מטרות המשתמשים:**

* **מטרות המורים:**
  + ניהול מלא ומרכזי של השיעורים והחומרים הנלמדים.
  + סידור תכנים בצורה אוטומטית (לפי תאריך, מקצוע וקטגוריה) והוספת תגים לשיפור החיפוש.
  + קבלת תמלול אוטומטי וסיכום תמציתי של החומר שנלמד, לתיעוד הנושאים המרכזיים.
  + קבלת עדכונים בזמן אמת על פעולות במערכת (כגון העלאת קבצים או הפקת תמלול).
* **מטרות הסטודנטים:**
  + קבלת גישה אוטומטית לאזור האישי עם כל החומרים והקבצים הרלוונטיים מהשיעורים.
  + צפייה בחומרים בצורה מאורגנת ונוחה, עם אפשרות לחיפוש לפי נושאים, תאריך או מקצוע.
  + קבלת סיכומי השיעורים המופקים אוטומטית, המאפשרים חזרה מהירה על הנושאים המרכזיים.
  + שיפור תהליכי הלמידה וההתעדכנות עם חומר עדכני וזמין בכל עת.

**4. פונקציונליות עיקרית**

1. **ניהול משתמשים והרשאות**
   * הרשמה, התחברות וניהול פרופיל אישי למורים ולסטודנטים.
   * אימות והרשאות מבוססי JWT, עם תפקידים נפרדים למורה ולסטודנט.
2. **ניהול קבצים**
   * העלאת קבצים (מסמכים, מצגות, קבצי קוד, תמונות, וידאו).
   * סידור קבצים אוטומטי לפי תאריך, מקצוע, קטגוריה והוספת תגים.
   * העברת הקבצים לאזור האישי של כל סטודנט בסיום השיעור.
3. **פיצ'ר AI – תמלול אוטומטי וזיהוי תוכן חשוב**
   * הפקת תמלול אוטומטי של קבצי השיעור באמצעות מנוע NLP.
   * זיהוי תוכן מרכזי והפקת סיכום תמציתי לחומר שהועלה.
4. **עדכונים והתראות בזמן אמת (Realtime)**
   * שילוב טכנולוגיות SignalR (בצד השרת) ו-Socket.io (בצד הלקוח) לקבלת התראות ועדכונים בזמן אמת.
   * עדכון דשבורד המשתמש באופן אוטומטי עם שינויים במערכת.

**5. טכנולוגיות בשימוש**

* **Backend:**
  + פיתוח API ב-.NET 9 לניהול משתמשים, קבצים ופיצ'ר ה-AI.
* **Frontend:**
  + אפליקציית React.js לממשק משתמש אינטראקטיבי ודינמי.
* **מסד נתונים:**
  + שימוש ב-PostgreSQL או SQL Server לאחסון נתוני המשתמשים והקבצים.
* **אחסון קבצים:**
  + אחסון בענן באמצעות Amazon S3 (או שירות ענן חלופי).
* **AI Services:**
  + אינטגרציה עם שירותי NLP (למשל OpenAI או HuggingFace) לתמלול אוטומטי וזיהוי תוכן חשוב.
* **Realtime:**
  + שילוב SignalR ב-.NET וב-Socket.io ב-React לקבלת עדכונים והתראות בזמן אמת.
* **Authentication:**
  + אימות והרשאות מבוססי JWT.

**6. API Routes עיקריים**

1. **ניהול משתמשים**
   * POST /api/auth/register – רישום משתמש חדש (מורה/סטודנט).
   * POST /api/auth/login – התחברות למערכת.
   * GET /api/auth/profile – קבלת פרטי משתמש.
2. **ניהול קבצים**
   * POST /api/files/upload – העלאת קובץ חדש.
   * GET /api/files – קבלת רשימת קבצים עם אפשרויות סינון (לפי תאריך, מקצוע, קטגוריה).
   * DELETE /api/files/{id} – מחיקת קובץ.
3. **AI – הפקת תמלול וסיכום**
   * POST /api/ai/summarize – הפקת תמלול וסיכום אוטומטי של תכני הקובץ.
   * פרמטרים: { fileId: string, content: string }
   * פלט: { summary: string }
4. **Realtime**
   * שילוב SignalR ו-Socket.io לקבלת עדכונים והתראות בזמן אמת על פעולות במערכת.

**7. הגדרת סבבי פיתוח (ספרינטים)**

**ספרינט 1 – תשתיות וניהול משתמשים (שבועות 1-2)**

* הקמת סביבת עבודה: יצירת פרויקטים ב-.NET, React והגדרת מסד נתונים.
* פיתוח API לניהול משתמשים (הרשמה, התחברות) עם JWT.
* עיצוב ראשוני של מסך כניסה/רישום.

**ספרינט 2 – פיתוח מודול ניהול קבצים (שבועות 3-4)**

* יצירת טבלאות Users ו-Files במסד הנתונים.
* פיתוח API לניהול קבצים – העלאה, עריכה, מחיקה.
* הטמעת סידור אוטומטי של הקבצים לפי תאריך, מקצוע וקטגוריה.
* עיצוב מסך דשבורד להצגת הקבצים.

**ספרינט 3 – הטמעת פיצ'ר AI – תמלול אוטומטי וזיהוי תוכן (שבועות 5-6)**

* פיתוח API ייעודי לקבלת תמלול וסיכום לתכני הקבצים.
* אינטגרציה עם שירות NLP להפקת תמלול וסיכום.
* עדכון ממשק המשתמש להצגת הסיכום עבור כל קובץ.

**ספרינט 4 – שיפורים, REALTIME ובדיקות (שבועות 7-8)**

* שילוב SignalR (בצד השרת) ו-Socket.io (בצד הלקוח) לקבלת עדכונים בזמן אמת.
* שיפור עיצוב הממשק והתאמתו למכשירים ניידים.
* ביצוע בדיקות איכות (QA), אבטחה ובדיקות עומס.

**ספרינט 5 – פריסה ופרסום (שבועות 9-10)**

* פריסת המערכת בענן (AWS, Azure או Google Cloud).
* הגדרת דומיין אישי, גיבויים וניטור ביצועים.
* פרסום דף בית לשיווק והדרכה למשתמשים.

**8. תרשים מבנה הטבלאות (Database Schema)**

**טבלת Users**

| **שם העמודה** | **סוג נתונים** | **מאפיינים** | **תיאור** |
| --- | --- | --- | --- |
| id | INT | PK, AUTO\_INCREMENT | מזהה ייחודי |
| full\_name | VARCHAR(100) | NOT NULL | שם מלא |
| email | VARCHAR(100) | UNIQUE, NOT NULL | כתובת דוא"ל |
| password\_hash | VARCHAR(255) | NOT NULL | סיסמה מוצפנת |
| role | VARCHAR(20) | NOT NULL | תפקיד (מורה / סטודנט) |
| created\_at | DATETIME | DEFAULT NOW() | תאריך יצירה |
| updated\_at | DATETIME | NULL | תאריך עדכון אחרון |

**טבלת Files**

| **שם העמודה** | **סוג נתונים** | **מאפיינים** | **תיאור** |
| --- | --- | --- | --- |
| id | INT | PK, AUTO\_INCREMENT | מזהה ייחודי |
| user\_id | INT | FK (Users.id) | מזהה המשתמש שהעלה את הקובץ |
| file\_name | VARCHAR(150) | NOT NULL | שם הקובץ |
| file\_type | VARCHAR(50) | NOT NULL | סוג הקובץ (PDF, תמונה, וידאו וכו') |
| url | VARCHAR(255) | NOT NULL | כתובת אחסון הקובץ בענן |
| subject | VARCHAR(100) | NOT NULL | מקצוע השיעור |
| category | VARCHAR(50) | NOT NULL | קטגוריה (שיעור, שיעורי בית וכו') |
| tags | VARCHAR(255) | NULL | תגים (מופרדים בפסיק) |
| ai\_summary | TEXT | NULL | תמלול וסיכום אוטומטי של תכני הקובץ |
| upload\_date | DATETIME | DEFAULT NOW() | תאריך העלאה |
| created\_at | DATETIME | DEFAULT NOW() | תאריך יצירה |
| updated\_at | DATETIME | NULL | תאריך עדכון אחרון |

**קשרים בין הטבלאות**

* **One-to-Many:**
  + משתמש אחד (מורה או סטודנט) יכול להעלות מספר קבצים.
  + קישור: Files.user\_id → Users.id

**9. סיכום**

מערכת "ניהול חומרים ושיעורים במכללה" מציעה פתרון כולל לניהול קבצי שיעורים וחומרי לימוד, כאשר:

* **למורים:** המערכת מאפשרת ניהול קל ונגיש של השיעורים, סימון נוכחות, העלאת קבצים, העברת החומרים לאזור האישי של הסטודנטים ותמלול אוטומטי עם זיהוי תוכן מרכזי.
* **לסטודנטים:** המערכת מבטיחה העברת כל החומרים באופן אוטומטי לאזור האישי שלהם, עם גישה מסודרת ונוחה לחומרי הלימוד וסיכומי השיעורים.

באמצעות שילוב טכנולוגיות מתקדמות (AI, Realtime, Cloud Storage) ואבטחת גישה מבוקרת, הפרויקט שואף לשפר באופן משמעותי את תהליכי ההוראה והלמידה במכללה, ולהעניק למשתמשים חוויית משתמש אינטואיטיבית, מהירה ומתקדמת.

מערכת ניהול שיעורים:  
יש עמוד בית ובו כניסה לאזורים אישיים ובו ניתן להיכנס כמנהל מורה וסטודנט  
המנהל-יוצר קבוצות ומכניס לכל קבוצה את המיילים של הסטודנט  
לכל מורה יש אפשרות להוסיף מקצוע לימוד ובתוכו לבחור קבוצת לימוד ובתוכו אפשרות להוסיף שיעורים ולהעלות לתוכו את קבצי השיעור ובעת לחיצת כפתור קובץ השיעור נשלח לכל התלמידות שבקבוצה  
סטודנט-לכל סטודנט יש אזור אישי שמסודר לפי תאריכים בכל תאריך יש את כל השיעורים-קבצים שנשלחו באותו תאריך  
ובנוסף לזה יש מנהל אחד על כל המערכת הזו המאפשר לנהל מוסד לימוד שיש לו את האפשרויות האלו:  כגון ניהול הגדרות מערכת, דוחות פעילות כלליים, צפייה בלוגים של המערכת.וכן המנהל יכול להוסיף מורים וקבוצות ומייל של סטודנט  
הערה😎:  
בכל פעם שכתוב להוסיף הכוונה להוסיף מחוק לעדכן ולקבל