**אפיון מערכת ArchiSync**

1. **תיאור כללי  
   ArchiSync היא אפליקציה SaaS לניהול פרויקטים עבור אדריכלים. המערכת מאפשרת למשתמשים ליצור ולנהל פרויקטים אישיים, להעלות קבצים, ולבצע תהליכים אוטומטיים להפקת תמונות מתוך סקיצות באמצעות AI. המשתמשים יכולים לשתף תיקי עבודות עם ציבור או בקשות גישה, ולנהל משתמשים ואירועים.**

**1.1 מטרת הפרויקט**

* **לאפשר לאדריכלים לנהל פרויקטים בצורה נוחה ובטוחה.**
* **לאפשר למשתמשים להעלות קבצים, כולל סקיצות, ולהמירם לתמונות בעזרת AI.**
* **לשפר את נגישות המידע ולייעל את תהליך שיתוף הפרויקטים.**
* **להקל על ניהול משתמשים והרשאות גישה.**
* **לספק ממשק ניהול שמציג נתוני כניסה ואירועים לחודשים האחרונים.**

**1.2 קהל היעד**

* **אדריכלים הזקוקים לניהול פרויקטים וסקיצות.**
* **משתמשים שמעוניינים להציג תיקי עבודות.**
* **מנהלי משתמשים המבקשים לנטר פעילות במערכת.**
* **גופים מקצועיים המעוניינים בגישה לפרויקטים לצורך שיתופי פעולה.**

**1.3 פונקציונליות של המערכת**

* **יצירת פרויקטים חדשים וניהול פרויקטים קיימים.**
* **העלאת קבצים (כולל סקיצות) ושמירה בענן.**
* **הפקת תמונות מתוך סקיצות באמצעות AI.**
* **הצגת תיקי עבודות עם אפשרות לצפייה ציבורית או פרטית.**
* **ניהול משתמשים, כולל הרשאות גישה וסטטוס.**
* **הצגת נתוני כניסה וסטטיסטיקות בחודשים האחרונים.**

**1.4 בעיות שהמערכת פותרת**

* **ריכוז כל המידע והקבצים במקום אחד ונגיש מכל מקום.**
* **שיפור שיתוף הפרויקטים בצורה נוחה ומהירה.**
* **ביטול הצורך בניהול ניירת פיזית.**
* **צמצום טעויות בניהול פרויקטים באמצעות אוטומציה.**

**1.5 טכנולוגיות בשימוש**

* **Backend: .NET Core לניהול API.**
* **Frontend: Angular לניהול משתמשים ו-React לניהול פרויקטים והעלאת קבצים.**
* **Database: PostgreSQL/MongoDB.**
* **Cloud Storage: AWS S3/Firebase.**
* **AI: מודלים לניתוח והמרת סקיצות לתמונות.**
* **Security: JWT, OAuth2, הצפנת נתונים.**
* **CI/CD: GitHub Actions לפריסה אוטומטית.**

**1.6 מתודולוגיית פיתוח (Agile)**

* **עבודה לפי ספרינטים בני שבוע.**
* **שימוש ב-Jira/Trello לניהול משימות.**
* **בדיקות איכות והפצות מדורגות.**
* **שילוב CI/CD לפריסה מהירה.**
* **שימוש במתודולוגיית TDD לבדיקת רכיבים מרכזיים.**

1. **עיצוב ממשק משתמש ויזואלי (Wireframes)  
   📌 מסכים מרכזיים:**

* **מסך התחברות/הרשמה – הזדהות מאובטחת.**
* **מסך ניהול פרויקטים – הצגת פרויקטים, העלאה, מחיקה ועידכון.**
* **מסך העלאת קבצים – העלאת קבצים וסקיצות.**
* **מסך הצגת תיקי עבודות – תצוגה של פרויקטים ציבוריים ופרטיים.**
* **מסך ניהול משתמשים (Angular) – ניהול משתמשים והרשאות.**
* **מסך הצגת נתוני כניסה – גרפים וסטטיסטיקות על פעילות המשתמשים.**

1. **איפיון תמציתי**

**3.1 רשימת פונקציות המערכת**

| **פונקציה** | **Route** | **פרמטרים** | **פלט** |
| --- | --- | --- | --- |
| **העלאת פרויקט** | **POST /api/projects/upload** | **projectData: JSON** | **{ projectId, url }** |
| **חיפוש פרויקטים** | **GET /api/projects/search** | **query: string** | **רשימת פרויקטים** |
| **הפקת תמונה מסקיצה** | **POST /api/sketch/convert** | **sketchId: string** | **{ imageUrl }** |
| **ניהול משתמשים** | **GET /api/users** | **-** | **רשימת משתמשים** |

**3.2 אימות והרשאות**

* **JWT Authentication להגנה על API.**
* **רמות גישה: אדריכל, משתמש רגיל, מנהל מערכת.**
* **הרשאות מבוססות תפקידים (RBAC).**

**3.3 אפליקציית ניהול**

* **צפייה וניהול משתמשים והרשאות (Angular).**
* **עדכון פרטי פרויקט.**
* **הצגת נתוני כניסה וסטטיסטיקות למנהלים.**

**📂 טבלאות המערכת:**

**4.1 טבלת משתמשים (Users)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה משתמש** |
| **full\_name** | **VARCHAR(100)** | **שם מלא** |
| **email** | **VARCHAR(100)** | **אימייל ייחודי** |
| **role** | **ENUM('architect', 'user', 'admin')** | **סוג משתמש** |
| **created\_at** | **TIMESTAMP** | **תאריך יצירה** |

**4.2 טבלת פרויקטים (Projects)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה פרויקט** |
| **user\_id** | **INT (FK)** | **מזהה משתמש** |
| **project\_name** | **VARCHAR(255)** | **שם הפרויקט** |
| **project\_url** | **TEXT** | **מיקום הפרויקט בענן** |
| **created\_at** | **TIMESTAMP** | **תאריך יצירה** |
| **updated\_at** | **TIMESTAMP** | **תאריך עדכון** |

**4.3 טבלת קבצים (Files)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה קובץ** |
| **project\_id** | **INT (FK)** | **מזהה פרויקט** |
| **file\_name** | **VARCHAR(255)** | **שם הקובץ** |
| **file\_url** | **TEXT** | **מיקום הקובץ בענן** |
| **uploaded\_at** | **TIMESTAMP** | **תאריך העלאה** |
| **updated\_at** | **TIMESTAMP** | **תאריך עדכון** |
|  |  |  |

**בהחלט! ניתן להוסיף עוד טבלאות ולפרט את המבנה שלהן באפיון. הנה דוגמה להרחבה על טבלאות נוספות, כולל טבלת הרשאות, טבלת פעילויות, וטבלת הודעות.**

**4. טבלאות המערכת (המשך)**

**4.1 טבלת משתמשים (Users)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה משתמש** |
| **full\_name** | **VARCHAR(100)** | **שם מלא** |
| **email** | **VARCHAR(100)** | **אימייל ייחודי** |
| **password\_hash** | **VARCHAR(255)** | **סיסמה מאוחסנת** |
| **role** | **ENUM('architect', 'user', 'admin')** | **סוג משתמש** |
| **created\_at** | **TIMESTAMP** | **תאריך יצירה** |

**4.2 טבלת פרויקטים (Projects)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה פרויקט** |
| **user\_id** | **INT (FK)** | **מזהה משתמש** |
| **project\_name** | **VARCHAR(255)** | **שם הפרויקט** |
| **project\_url** | **TEXT** | **מיקום הפרויקט בענן** |
| **created\_at** | **TIMESTAMP** | **תאריך יצירה** |
| **updated\_at** | **TIMESTAMP** | **תאריך עדכון** |
| **visibility** | **ENUM('public', 'private')** | **סטטוס נראות של הפרויקט** |

**4.3 טבלת קבצים (Files)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה קובץ** |
| **project\_id** | **INT (FK)** | **מזהה פרויקט** |
| **file\_name** | **VARCHAR(255)** | **שם הקובץ** |
| **file\_url** | **TEXT** | **מיקום הקובץ בענן** |
| **uploaded\_at** | **TIMESTAMP** | **תאריך העלאה** |
| **updated\_at** | **TIMESTAMP** | **תאריך עדכון** |
| **file\_type** | **VARCHAR(50)** | **סוג הקובץ (PDF, תמונה וכו')** |

**4.4 טבלת הרשאות (Permissions)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה הרשאה** |
| **user\_id** | **INT (FK)** | **מזהה משתמש** |
| **project\_id** | **INT (FK)** | **מזהה פרויקט** |
| **permission\_type** | **ENUM('view', 'edit', 'delete')** | **סוג הרשאה** |

**4.5 טבלת פעילויות (Activities)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה פעילות** |
| **user\_id** | **INT (FK)** | **מזהה המשתמש שביצע את הפעולה** |
| **action** | **VARCHAR(255)** | **פעולה שבוצעה (כמו "העלה קובץ", "עדכן פרויקט")** |
| **project\_id** | **INT (FK)** | **מזהה פרויקט (אם רלוונטי)** |
| **timestamp** | **TIMESTAMP** | **תאריך ושעת ביצוע הפעולה** |

**4.6 טבלת הודעות (Messages)**

| **עמודה** | **סוג נתונים** | **תיאור** |
| --- | --- | --- |
| **id** | **INT (PK)** | **מזהה הודעה** |
| **sender\_id** | **INT (FK)** | **מזהה השולח** |
| **receiver\_id** | **INT (FK)** | **מזהה המקבל** |
| **message\_text** | **TEXT** | **תוכן ההודעה** |
| **sent\_at** | **TIMESTAMP** | **תאריך ושעת שליחה** |

1. **📌 ספרינט 1 - תשתיות ופיתוח בסיסי  
   מטרות:**

* **הקמת פרויקט .NET Core לפיתוח API.**
* **הגדרת מסד נתונים PostgreSQL/MongoDB.**
* **יצירת מערכת אימות משתמשים (JWT) והצפנת נתונים.**
* **פיתוח API ראשוני לניהול משתמשים ופרויקטים.**

**תוצרים: ✅ בסיס נתונים מוגדר עם טבלאות: Users, Projects, Files  
✅ API ראשוני: הרשמה, התחברות, העלאת פרויקטים, חיפוש פרויקטים  
✅ אימות והצפנה: JWT, OAuth2  
✅ תשתית CI/CD לפריסה ראשונית**

**📌 ספרינט 2 - פיתוח ממשק משתמש וניהול פרויקטים  
מטרות:**

* **פיתוח ממשק ניהול פרויקטים ב-React (העלאה, חיפוש, תצוגה).**
* **פיתוח מערכת ניהול משתמשים ב-Angular (הרשאות, עדכון מידע).**
* **חיבור ה-Frontend ל-API בעזרת Axios/Fetch.**

**תוצרים: ✅ ממשק ניהול פרויקטים (React)  
✅ מסך ניהול משתמשים והרשאות (Angular)  
✅ חיבור מלא של ה-Frontend ל-Backend**

**📌 ספרינט 3 - הפקת תמונות ואינטגרציה עם AI  
מטרות:**

* **פיתוח מערכת להפקת תמונות מסקיצות.**
* **אינטגרציה עם API של מודלים לניתוח סקיצות.**
* **ניהול התראות ועדכון משתמשים על פרויקטים חדשים.**

**תוצרים: ✅ הפקת תמונות מסקיצות באמצעות AI  
✅ מערכת התראות למשתמשים על פרויקטים חדשים**

**📌 ספרינט 4 - אבטחה מתקדמת וניהול משתמשים  
מטרות:**

* **שיפור אבטחת המערכת והגנה על נתוני המשתמשים.**
* **הגברת השליטה בניהול משתמשים והרשאות.**

**תוצרים: ✅ אבטחה מתקדמת (הצפנת נתונים, גישה לפי הרשאות)  
✅ ממשק ניהול משתמשים עם פונקציות מתקדמות**