ב-.NET Core Web API:

**ספרינט 1: פריסה בענן והקמת API ב-.NET Core Web API**

**מטרת הספרינט:**  
הכנת התשתית לפיתוח בענן, כולל חיבור למסד נתונים, אחסון קבצים ופריסת ה-API.

**שלבים עיקריים:**

1. **הכנה לפריסת שירותים בענן (Clever Cloud ו-AWS):**
   * יצירת חשבון ב-Clever Cloud ו-AWS.
   * יצירת באקט ב-AWS S3 לאחסון קבצים.
   * יצירת שירות PostgreSQL ב-Clever Cloud.
2. **העלאה ל-GitHub:**
   * יצירת ריפוזיטורי ב-GitHub לפרויקט.
   * העלאת הקוד לכל פרויקט (צד שרת וצד לקוח).
   * הגדרת .gitignore למניעת העלאת קבצים לא רצויים.
3. **הקמת פרויקט .NET Core Web API:**
   * יצירת פרויקט ASP.NET Core Web API.
   * הגדרת שירות Web ב-Render.
   * חיבור ל-PostgreSQL ב-Clever Cloud והגדרת טבלאות בסיסיות (Users, Worksheets, Ratings).
   * הגדרת חיבור ל-Amazon S3 להעלאת קבצים.
4. **חיבור ה-API למסד נתונים ו-Amazon S3:**
   * הגדרת חיבור למסד נתונים ב-Clever Cloud.
   * חיבור לאחסון קבצים ב-AWS S3 והעלאת קבצים.
5. **פריסת ה-API בענן:**
   * העלאת ה-API ל-Render.
   * הגדרת משתני סביבה (API URL, חיבור למסד נתונים ו-S3).
   * בדיקות חיבור ופעולה של ה-API בענן.

**זמן צפוי:** 12-15 שעות

**ספרינט 2: פיתוח צד שרת (API)**

**מטרת הספרינט:**  
פיתוח פונקציות בסיסיות ב-API, חיבור למסד נתונים ו-Amazon S3, ניהול דפי עבודה ודירוגים.

**שלבים עיקריים:**

1. **פיתוח פונקציות ב-API:**
   * יצירת פונקציות CRUD (הוספה, הצגה, עדכון, מחיקה) לדפי עבודה.
   * יצירת פונקציות לניהול דירוגים.
   * פיתוח לוגיקה לפעולות דינמיות.
2. **חיבור למסד נתונים (PostgreSQL):**
   * חיבור ה-API למסד הנתונים ב-Clever Cloud.
   * יצירת טבלאות וכתיבת שאילתות קריאה וכתיבה.
3. **חיבור ל-Amazon S3:**
   * הגדרת מפתחות גישה ל-S3.
   * כתיבת לוגיקה להעלאת קבצים (כגון דפי עבודה ב-PDF, PNG).
4. **בדיקות אינטגרציה:**
   * ביצוע בדיקות חיבור למסד הנתונים ו-S3.
   * ווידוא שה-API פועל כראוי עם כל השירותים המשותפים.

**זמן צפוי:** 12-15 שעות

**ספרינט 3: פיתוח צד לקוח ב-React**

**מטרת הספרינט:**  
פיתוח צד לקוח ב-React, יצירת ממשק משתמש עבור משתמשי אתר הציורים והדפי עבודה.

**שלבים עיקריים:**

1. **הקמת פרויקט React:**
   * יצירת פרויקט חדש ב-React.
   * חיבור ה-React API לשרת (באמצעות REST API).
2. **פיתוח מסכים ופונקציות:**
   * דף התחברות/רישום: אפשרות למשתמשים להירשם ולהתחבר.
   * דף הבית: הצגת קטגוריות של דפי עבודה, אפשרות לחיפוש וסינון.
   * דף דפי עבודה: הצגת דפי עבודה לבחירה, הוספת דירוגים והערות.
   * דף מועדפים: הצגת דפי עבודה מועדפים של המשתמש.
   * הורדת קבצים: אפשרות להוריד דפי עבודה בפורמטים שונים (PDF, PNG).
3. **חוויית משתמש:**
   * פיתוח אינטרקציות פשוטות ונוחות: התחברות, חיפוש, דירוג והוספת דפים למועדפים.
4. **בדיקות UI ו-UX:**
   * בדיקות אינטגרציה על ה-UI לוודא שהחוויות של המשתמש פועלות בצורה תקינה.

**זמן צפוי:** 12-15 שעות

**ספרינט 4: פיתוח צד לקוח ב-Angular – ממשק למנהל מערכת**

**מטרת הספרינט:**  
פיתוח ממשק ניהול למנהל המערכת ב-Angular, ניהול דפי עבודה וציורים.

**שלבים עיקריים:**

1. **הקמת פרויקט Angular:**
   * יצירת פרויקט חדש ב-Angular.
2. **פיתוח מסכים למנהל מערכת:**
   * דף ניהול דפי עבודה: הצגת רשימה, עדכון ומחיקת דפי עבודה.
   * דף ניהול קטגוריות: הוספה, עדכון ומחיקה של קטגוריות.
   * דף ניהול משתמשים: ניהול משתמשים, הרשאות ודירוגים.
3. **פונקציות ניהול:**
   * אפשרות להעלות דפי עבודה חדשים (PDF, PNG).
   * ניהול משוב ודירוגים על דפי עבודה.
   * שליטה בתצוגה (מיון, חיפוש, סינון).
4. **חוויית משתמש:**
   * ממשק ניהול אינטואיטיבי ונוח עם אמצעי ניווט פשוטים.
5. **בדיקות אינטגרציה ו-UI:**
   * בדיקות UI ו-UX לוודא שהממשק פועל בצורה תקינה ונוחה.

**זמן צפוי:** 10-12 שעות

**ספרינט 5: סיום ותחזוקה**

**מטרת הספרינט:**  
אופטימיזציה, ביצוע תיקונים, ופריסת המערכת לסביבה חיה.

**שלבים עיקריים:**

1. **אופטימיזציה וביצועים:**
   * שיפור ביצועים של ה-API וצד הלקוח.
   * אופטימיזציה של חיפושים וסינונים בממשק המשתמש.
2. **בדיקות סופיות:**
   * ביצוע בדיקות QA מלאות למערכת (בדיקות אינטגרציה, בדיקות משתמש).
   * תיקון בעיות שהתגלו במהלך הבדיקות.
3. **פריסה לסביבה חיה:**
   * פריסת ה-API לצד השרת לסביבה חיה.
   * פריסת הממשק (React ו-Angular) לסביבה חיה.
4. **ממשק משתמש סופי:**
   * שיפור חוויית המשתמש ויישום שיפורים קטנים לפי משוב מהמשתמשים.
5. **תחזוקה לאחר ההשקה:**
   * ניטור המערכת ופתרון בעיות (במקרה הצורך).
   * מתן תמיכה טכנית למשתמשים.

**זמן צפוי:** 10-15 שעות

**סיכום כל הספרינטים:** הספרינטים כולם מתמקדים בבניית המערכת בצורה מסודרת, החל מהקמת ה-API בענן ועד לפיתוח צד הלקוח, כולל ממשק ניהול למנהל המערכת. כל ספרינט מתרכז בהיבט אחר של המערכת (צד שרת, צד לקוח, אופטימיזציה ותחזוקה), וכולל את הצעדים המפורטים הנדרשים לפיתוח והפריסה.