**الخطة التفصيلية لتنفيذ مشروع إدارة الأصول والعهد باستخدام ASP.NET Core MVC**

**التقنيات المستخدمة**

* **لغة البرمجةC# (.NET 9):**
* **إطار العمل ASP.NET Core MVC:**
* **قاعدة البيانات SQL Server:**
* **إدارة البياناتEntity Framework Core:**
* **واجهة المستخدم HTML, CSS, JavaScript, Bootstrap:**
* **التقارير iTextSharp: لإنشاء PDF وEPPlus لتصدير Excel**
* **المصادقة والصلاحيات ASP.NET Identity + JWT Authentication:**

**الجدول الزمني لتنفيذ المشروع (10 أيام عمل)**

**اليوم 1-2: تصميم قاعدة البيانات (Database Design)**

* **تحليل البيانات المطلوبة** وإنشاء مخطط قاعدة البيانات (ERD - Entity Relationship Diagram).
* تصميم الجداول الأساسية والعلاقات بينها.
* كتابة **ملف SQL Script** لإنشاء الجداول بحيث يمكن تنفيذه بسهولة عند نشر المشروع.
* **إنشاء قاعدة البيانات في SQL Server**
* إعداد **Migrations** لتمكين التحديثات المستقبلية بسهولة.

**اليوم 3-4: بناء البنية الخلفية (Backend Development)**

* إعداد **مشروع ASP.NET Core MVC** مع **Entity Framework Core**.
* **إنشاء النماذج (Models) والـ DTOs** لكل من الأصول، الحركات، المستخدمين، إلخ.
* **بناء طبقة الوصول إلى البيانات (Repository Pattern) لإدارة العمليات على قاعدة البيانات.**
* **تطوير الخدمات (Services) المسؤولة عن تنفيذ العمليات المختلفة، مثل:**
  + **إضافة، تعديل، حذف، تكهين الأصول.**
  + **نقل الأصول وتسجيل العمليات في سجل الحركات.**
  + **إدارة المستخدمين والصلاحيات.**
  + **تسجيل كافة الأنشطة في سجل التدقيق (Audit Log).**
* **إنشاء وحدات الاختبار (Unit Tests) للكود الخلفي باستخدام xUnit.**

**اليوم 5-6: بناء الواجهة الأمامية (Frontend Development)**

* استخدام **Razor Pages** لإنشاء **تصميم متجاوب وحديث** يدعم **اللغة العربية**
* إنشاء صفحة **عرض الأصول مع الفلترة والبحث** حسب القسم، المستلم، النوع، إلخ.
* تصميم **النماذج اللازمة** لإدخال البيانات بنفس تجربة المستخدم المتوفرة في ملف Excel.
* **تطوير صفحات رئيسية تشمل:**
  + لوحة التحكم (Dashboard)
  + إدارة الأصول (إضافة/تعديل/حذف/تكهين)
  + إدارة المستخدمين والأدوار
  + سجل الحركات والتدقيق
  + البحث والتصفية filter

**اليوم 7: إدارة المستخدمين والصلاحيات**

* **تطوير نظام تسجيل الدخول والتسجيل (Authentication & Authorization) باستخدام Identity.**
* **تعيين أدوار وصلاحيات مختلفة للمستخدمين (Admin - مشرف - مسؤول - مدخل بيانات - مستخدم).**
* **إضافة تسجيل نشاطات المستخدمين في Audit Log.**

**اليوم 8: التقارير والإحصاءات + استيراد وتصدير البيانات**

* **تطوير نظام تقارير شامل (Reports) يتضمن:**
  + تقرير الأصول حسب القسم أو المستخدم أو الحالة (قيد الاستخدام / مكهن).
  + تقرير الحركات التي تمت على الأصول.
  + سجل أنشطة المستخدمين.
* **إضافة ميزة تصدير البيانات بصيغ Excel و PDF باستخدام مكتبات مثل EPPlus و iTextSharp.**
* **إضافة ميزة استيراد البيانات من ملفات Excel** لتحميل بيانات الأصول الجديدة مباشرة إلى النظام.

**اليوم 9: الاختبار وضمان الجودة (Testing & Quality Assurance)**

* **اختبار جميع الميزات يدويًا باستخدام Postman و SQL Profiler.**
* **إجراء اختبارات أداء (Performance Testing) للتأكد من سرعة النظام.**
* **إجراء اختبارات أمان (Security Testing) لمنع أي ثغرات أمنية مثل SQL Injection و Cross-Site Scripting.**

**اليوم 10: تسليم المشروع + الوثائق**

* تحضير ملف التسليم (Documentation) وشرح كيفية التشغيل و إعدادات الاتصال بقاعدة البيانات (Connection String).
* **تسليم الكود بالكامل** مكتوب بطريقة نظيفة وقابلة للتطوير مع SQL Script لإنشاء قاعدة البيانات.

**الكود المصدري بالكامل**

✔ **مشروع ASP.NET Core MVC جاهز للتشغيل.**  
✔ مكتوب بأسلوب نظيف وقابل للتطوير (**Clean Code & Best Practices**).  
✔ يحتوي على جميع الميزات المطلوبة، بما في ذلك **إدارة الأصول، الصلاحيات، التقارير، والاستيراد والتصدير**.  
✔ **مقسم إلى طبقات (Layered Architecture)** لضمان سهولة الصيانة والتطوير المستقبلي.

**الملفات التي ستتسلمها:**  
**المجلد الرئيسي للمشروع** يحتوي على:

* **Controllers/** → جميع الـ **Controllers** المسؤولة عن منطق التطبيق.
* **Models/** → الكيانات (Entities) وملفات **DTOs**.
* **Views/** → صفحات العرض (Razor Views) مع تصميم متجاوب.
* **wwwroot/** → ملفات **CSS و JS والصور**.
* **Services/** → الخدمات التي تنفذ العمليات (CRUD, Authentication, Reports).
* **Repositories/** → طبقة الوصول للبيانات باستخدام **Entity Framework Core**.
* **Migrations/** → ملفات **Entity Framework Migrations** لإنشاء الجداول تلقائيًا.
* **appsettings.json** → ملف الإعدادات، بما في ذلك **Connection String**.

**قاعدة البيانات (SQL Server Database)**

✔ **ملف SQL كامل لإنشاء الجداول والعلاقات والبيانات الافتراضية (Seed Data).**  
✔ **جداول تشمل:** الأصول، المستخدمين، الحركات، الصلاحيات، السجلات، التقارير.  
✔ دعم **Entity Framework Core Migrations** لتحديث قاعدة البيانات بسهولة.

**الملفات التي ستتسلمها:**  
**Database/** يحتوي على:

* **database\_schema.sql** → سكريبت SQL لإنشاء الجداول والعلاقات يدويًا.
* **database\_seed.sql** → سكريبت لإضافة بيانات افتراضية (Users, Roles, Assets).

# **قوالب**

<https://demos.creative-tim.com/argon-dashboard/pages/rtl.html>

<https://adminlte.io/themes/v3/index2.html>

<https://adminlte.io/themes/v3/index3.html>

<https://adminlte.io/themes/v3/index.html>

<https://coreui.io/demos/bootstrap/5.1/free/?theme=dark>

<https://coreui.io/demos/bootstrap/5.1/free/?theme=light>

<https://startbootstrap.com/previews/sb-admin-2>

**تحياتي،**

**مؤمن حسن**