

دليل نشر نظام إدارة الصيانة

📋 نظرة عامة

كقاعدة بيانات، مع إمكانية العمل أونلاين ودعم الهواتف Google Sheets هذا النظام تم تطويره ليعمل بالكامل مع المحمولة.

🔧 الإعدادات المطلوبة

1. إعداد Google Sheets

إنشاء شيت المستخدمين:

- جديد باسم "نظام المستخدمين Google Sheet إنشاء"
- في الورقة الأولى، أضف الأعمدة التالية:

...A: الرقم الوظيفي
...B: الاسم
...C: كلمة المرور
...D: الصلاحية

3. أضف بيانات تجريبية:

000001 | م. فهد صالح | quality123 | quality
...000002 | د. محمد العميد | dean123 | dean
...000003 | أحمد المشرف | super123 | supervisor
000004 | سالم الفني | tech123 | technician
...000005 | خالد الموظف | emp123 | employee

إنشاء شيت الطلبات:

- جديد باسم "طلبات الصيانة Google Sheet إنشاء"
- سيتم إنشاء الأعمدة تلقائياً عند أول طلب

2. إعداد Google Apps Script

1. اذهب إلى script.google.com
2. إنشاء مشروع جديد
3. المرفق Google Apps Script انسخ كود
4. شيت الطلبات ID بـ (REQUESTS_SHEET_ID) استبدل
5. "Maintenance System API" احفظ المشروع باسم

Web App نشر ك

1. اضغط على "Deploy" > "New deployment"
2. اختر Type: "Web app"
3. Execute as: "Me"
4. Who has access: "Anyone"
5. اضغط "Deploy"
6. انسخ Web App URL

3. إعداد Google Sheets API

1. اذهب إلى Google Cloud Console
2. إنشاء مشروع جديد أو اختيار موجود
3. تفعيل Google Sheets API
4. إنشاء API Key:
 - Credentials > Create Credentials > API Key
 - للمواقع المحددة API Key اختياري: تقييد

النشر على الإنترنت

1 (مجاني) GitHub Pages: الخيار

1. إنشاء Repository:

```
bash
...git init
...git add .
...git commit -m "Initial commit"
...git branch -M main
...git remote add origin https://github.com/username/maintenance-system.git
...git push -u origin main
```

2. تفعيل GitHub Pages:

- Settings > Pages
- Source: Deploy from a branch
- Branch: main
- Path: / (root)

3. الرابط النهائي:

```
...https://username.github.io/maintenance-system/
```

2 (مجانى) Netlify: الخيار

1. ربط Repository:

- netlify.com اذهب إلى
- "New site from Git"
- اختر GitHub repository

2. إعدادات البناء:

- Build command: (فارغ)
- Publish directory: /
- اضغط "Deploy site"

3. دومين مخصص (اختياري):

- Domain settings > Add custom domain

3 (مجانى) Vercel: الخيار

1. ربط Repository:

- vercel.com اذهب إلى
- "New Project"
- Import من GitHub

2. النشر التلقائي:

- push سيتم النشر تلقائياً عند كل

التكوين النهائي ⚙️

config.json: تحديث ملف

```
json
{
  "sheetsConfig": {
    "usersSheetId": "YOUR_USERS_SHEET_ID",
    "requestsSheetId": "YOUR_REQUESTS_SHEET_ID",
    "apiKey": "YOUR_API_KEY",
    "appsScriptUrl": "YOUR_APPS_SCRIPT_URL"
  }
}
```

تحديث الروابط في الكود:

استبدل HTML في ملف

```
javascript
```

```
SHEETS_CONFIG: {  
  API_KEY: 'YOUR_API_KEY',  
  USERS_SHEET_ID: 'YOUR_USERS_SHEET_ID',  
  REQUESTS_SHEET_ID: 'YOUR_REQUESTS_SHEET_ID'  
}
```

التحسين للهواتف

النظام مُحسّن تلقائياً للهواتف المحمولة ويتضمن:

- تصميم متجاوب: يتكيف مع جميع أحجام الشاشات
- عمل أوفلاين: حفظ البيانات محلياً عند انقطاع الإنترنت
- مزامنة تلقائية: تحديث البيانات كل 30 ثانية
- واجهة سهلة: أزرار كبيرة ونماذج مبسطة

الأمان والخصوصية

API Key حماية:

1. للدومين المحدد فقط API Key قيود المجال: قيد
2. المسموحة (اختياري) IP إضافة عناوين IP: قيود
3. كل 3-6 أشهر API Key تجديد دوري: غير

حماية البيانات:

- (HTTPS) جميع البيانات مشفرة في النقل
- عدم حفظ كلمات المرور في المتصفح
- انتهاء جلسة العمل تلقائياً بعد الخمول

الاختبار

اختبار محلي:

```
bash  
# تشغيل خادم محلي  
python -m http.server 8000  
# أو  
npx serve .
```

اختبار الإنتاج:

1. تجربة جميع المميزات على الهاتف والكمبيوتر
2. اختبار العمل أوفلاين

3. اختبار المزامنة
4. اختبار صلاحيات المستخدمين

إطلاق النظام

قائمة التحقق النهائية:

- ☐ بالبيانات الصحيحة Google Sheets تم إعداد
- ☐ وتجربته Google Apps Script تم نشر
- ☐ تم تحديث جميع المعارف في الكود
- ☐ تم اختبار النظام على أجهزة مختلفة
- ☐ تم إعداد النسخ الاحتياطي للبيانات
- ☐ تم تدريب المستخدمين الأساسيين

توزيع الروابط:

- للإدارة: رابط النظام + دليل الاستخدام
- للفنيين: رابط النظام + تعليمات مبسطة
- للموظفين: رابط النظام + شرح إرسال الطلبات

الدعم والصيانة

مراقبة الأداء:

- Google Apps Script مراجعة دورية لسجلات
- Google Sheets API متابعة استخدام
- مراقبة سرعة الاستجابة

النسخ الاحتياطي:

- Google Sheets نسخ احتياطي أسبوعي من
- نسخ احتياطي شهري من الكود
- توثيق جميع التغييرات

التحديثات:

- تحديثات أمنية دورية
- تحسينات الأداء حسب الحاجة
- مميزات جديدة بناءً على طلبات المستخدمين

نصائح للنجاح

1. ابدأ تدريجياً: اختبر مع مجموعة صغيرة أولاً.
2. تدريب المستخدمين: قدم تدريباً شاملاً للفرق.

- دعم مستمر: كن متاحاً للدعم في الأسابيع الأولى. 3
- جمع التغذية الراجعة: استمع لملاحظات المستخدمين واعمل على التحسين. 4
- وثّق كل شيء: احتفظ بسجل مفصل للإعدادات والتغييرات. 5
-

معلومات إضافية

تطوير: م. فهد صالح - وكيل ضبط الجودة

الإصدار: 3.0

آخر تحديث: يناير 2025

للدعم التقني، راجع دليل المستخدم أو تواصل مع فريق تقنية المعلومات.