نظام طيبة الصحي المتكامل

Taiba Healthcare System

نظام طيبة هو نظام صحي متكامل مبني على بنية الخدمات المصغرة (Microservices) باستخدام FastAPI وPython، مع دعم كامل للمعايير الطبية العالمية والتكامل مع نظام نفيس السعودي.

🏥 المميزات الرئيسية

📋 إدارة شاملة للرعاية الصحية

- **إدارة المرضى**: تسجيل وإدارة بيانات المرضى مع الرقم الصحي الموحد
- **السجلات الطبية الإلكترونية (EHR)**: تسجيل التشخيصات والعلاجات والملاحظات السريرية
- **إدارة المواعيد**: حجز وإدارة المواعيد مع الإشعارات التلقائية
- **إدارة المختبرات (LIS)**: طلب الفحوصات وتتبع العينات وإدارة النتائج
- **إدارة الصيدليات (PMS)**: الوصفات الطبية وإدارة المخزون
- **الطب الاتصالي**: الاستشارات عن بعد والمراقبة المنزلية

🔒 الأمان والمصادقة

- **OAuth 2.0 / JWT**: مصادقة آمنة ومتقدمة
- **تشفير البيانات**: حماية شاملة للبيانات الحساسة
- **إدارة الصلاحيات**: نظام صلاحيات متقدم (RBAC)
- **تدقيق العمليات**: تسجيل جميع العمليات للمراجعة

🌐 المعايير الطبية المدعومة

- **HL7 FHIR R4**: تبادل البيانات الصحية
- **ICD-10-AM**: تصنيف التشخيصات والأمراض
- **SNOMED CT**: المصطلحات السريرية الموحدة
- **معايير نفيس**: التوافق الكامل مع المركز الوطني للمعلومات الصحية

📊 المراقبة والتحليل

- **Prometheus & Grafana**: مراقبة الأداء في الوقت الفعلي
- **ELK Stack**: تجميع وتحليل السجلات
- **التقارير المتقدمة**: تقارير شاملة وإحصائيات تفاعلية

🏗️ البنية المعمارية

الخدمات المصغرة (Microservices)

```
┌──────────────────┐  ┌──────────────────┐  ┌──────────────────┐
| Auth Service | | Patient Service | | EHR Service |
| :8001 | | :8002 | | :8003 |
└──────────────────┘  └──────────────────┘  └──────────────────┘


┌──────────────────┐  ┌──────────────────┐  ┌──────────────────┐
|Appointment Svc | |Laboratory Svc | | Pharmacy Svc |
| :8004 | | :8005 | | :8006 |
└──────────────────┘  └──────────────────┘  └──────────────────┘


┌──────────────────┐  ┌──────────────────┐  ┌──────────────────┐
|Telemedicine Svc | | Nafis Integration| |Medical Standards|
| :8007 | | :8008 | | :8009 |
└──────────────────┘  └──────────────────┘  └──────────────────┘


┌──────────────────┐
| Reporting Svc |
| :8010 |
└──────────────────┘
```

قواعد البيانات

- **PostgreSQL**: البيانات المهيكلة الرئيسية

- **MongoDB**: السجلات الطبية غير المهيكلة
- **Redis**: التخزين المؤقت والجلسات
- **Elasticsearch**: البحث والفهرسة

البنية التحتية

- **Nginx**: API Gateway وموازن الأحمال
- **RabbitMQ**: Message Broker للاتصالات غير المتزامنة
- **Docker**: حاويات التطبيقات
- **Prometheus**: مراقبة الأداء
- **Grafana**: لوحات المعلومات

🚀 التثبيت والتشغيل

المتطلبات الأساسية

- Docker & Docker Compose
- Python 3.11+
- Git

التثبيت السريع

```
# استنساخ المشروع
git clone <repository-url>
cd taiba_system

# إعداد بيئة التطوير
make dev-setup
```

التشغيل اليدوي

```
# تثبيت التبعيات
make install-deps
```

```
# بناء الحاويات
make build

# تشغيل النظام
make up

# ترحيل قاعدة البيانات
make db-migrate

# إدخال بيانات تجريبية
make db-seed
```

📖 الاستخدام

الوصول للنظام

- **الواجهة الأمامية**: http://localhost:3001
- **API Gateway**: http://localhost
- **Grafana**: http://localhost:3000 (admin/admin)
- **RabbitMQ Management**: http://localhost:15672 (taiba_user/taiba_password)

واجهات برمجة التطبيقات (APIs)

```
POST /api/auth/login # تسجيل الدخول
GET /api/patients/ # قائمة المرضى
POST /api/patients/ # إضافة مريض جديد
GET /api/ehr/{patient_id} # السجل الطبي للمريض
POST /api/appointments/ # حجز موعد
```

```
GET /api/laboratory/tests # قائمة الفحوصات المتاحة
```

أمثلة على الاستخدام

```
import httpx


# تسجيل الدخول
response = httpx.post("http://localhost/api/auth/login", json={
"username": "doctor@taiba.com",
"password": "password123"
})
token = response.json()["access_token"]


# إضافة مريض جديد
headers = {"Authorization": f"Bearer {token}"}
patient_data = {
"name": "أحمد محمد",
"national_id": "1234567890",
"phone": "+966501234567",
"email": "ahmed@example.com"
}
response = httpx.post("http://localhost/api/patients/",
json=patient_data, headers=headers)
```

🔧 التطوير

إضافة خدمة جديدة

```
# إنشاء خدمة جديدة
make create-service

# أو يدوياً
mkdir services/new_service
cd services/new_service
# إضافة الملفات المطلوبة
```

تشغيل الاختبارات

```
# جميع الاختبارات
make test

# اختبارات خدمة محددة
cd services/auth_service
python -m pytest tests/ -v
```

فحص جودة الكود

```
# فحص الكود
make lint

# تنسيق الكود
```

```
make format
```

📊 المراقبة

Prometheus Metrics

- معدل الطلبات (Request Rate)
- زمن الاستجابة (Response Time)
- معدل الأخطاء (Error Rate)
- استخدام الموارد (Resource Usage)

Grafana Dashboards

- لوحة معلومات النظام العامة
- مراقبة قواعد البيانات
- مراقبة الخدمات المصغرة
- تحليل الأداء

🔐 الأمان

أفضل الممارسات المطبقة

- تشفير البيانات في حالة السكون والنقل
- مصادقة متعددة العوامل
- تدقيق جميع العمليات
- فصل الصلاحيات
- حماية من OWASP Top 10

إعدادات الأمان

```
# تحديث كلمات المرور في .env
JWT_SECRET_KEY=your-super-secret-jwt-key
DATABASE_PASSWORD=strong-database-password

# تفعيل HTTPS في الإنتاج
# تكوين جدار الحماية
```

```
# تحديث النظام بانتظام
```

🌍 التكامل مع نفيس

المعايير المدعومة

- **HL7 v2.x & FHIR R4**: تبادل البيانات
- **ICD-10-AM**: ترميز التشخيصات
- **SNOMED CT**: المصطلحات السريرية
- **الرقم الصحي الموحد**: ربط هوية المريض

تكوين التكامل

```
# في ملف .env
NAFIS_BASE_URL=https://api.nafis.gov.sa
NAFIS_CLIENT_ID=your-client-id
NAFIS_CLIENT_SECRET=your-client-secret
NAFIS_API_KEY=your-api-key
```

📚 التوثيق

الوثائق المتاحة

- دليل المطور
- دليل المستخدم
- دليل النشر
- مرجع API

توثيق API التلقائي

- **Swagger UI**: http://localhost/docs
- **ReDoc**: http://localhost/redoc

🤝 المساهمة

خطوات المساهمة

1. Fork المشروع

2. إنشاء فرع للميزة الجديدة (`git checkout -b feature/amazing-feature`)
3. Commit التغييرات (`git commit -m 'Add amazing feature'`)
4. Push للفرع (`git push origin feature/amazing-feature`)
5. فتح Pull Request

معايير الكود

- اتباع PEP 8 لـ Python
- كتابة اختبارات للميزات الجديدة
- توثيق الكود والوظائف
- استخدام Type Hints

📄 الترخيص

هذا المشروع مرخص تحت رخصة MIT - راجع ملف LICENSE للتفاصيل.

📞 الدعم والتواصل

- **البريد الإلكتروني**: support@taiba-health.com
- **الوثائق**: https://docs.taiba-health.com
- **المجتمع**: https://community.taiba-health.com

🙏 شكر وتقدير

نشكر جميع المساهمين في تطوير هذا النظام والمجتمع الطبي السعودي لدعمهم المستمر.

نظام طيبة الصحي - نحو رعاية صحية رقمية متقدمة 🏥✨