جامعة اللاذقية...كلية الهندسة المعلوماتية

قسم البرمجيات ونظم المعلومات

تحليل النظم المالية

**وثيقة المتطلبات البرمجية (SRS)**

اسم النظام: 🏛️

نظام إدارة صالات الألعاب الرياضية

التاريخ: 📅

**5/7/202**5

إعداد الطلاب: ✍️

حلا نبيل حماده

راما ياسر علي

ريم عماد أسود

سلمان فرج علي

المقدمة:

1) الهدف من الوثيقة:

**1.1 تهدف الوثيقة إلى تحديد متطلبات نظام صالات الألعاب الرياضية بشكل واضح و دقيق تمثل مرجعاً للعميل والمطورين لفهم الوظائف المطلوبة لضمان جودة وسهولة تطوير النظام.**

نطاق النظام:

**1.2 يقوم النظام بإدارة جميع العمليات داخل الصالة الرياضية مثل: تسجيل الأعضاء، حجز الجلسات التدريبية، إعداد الخطط الغذائية ،الدفع الالكتروني، إصدار التقارير الصحية ويهدف إلى تسهيل العمل على الموظفين والمدربين وتحسين تجربة الأعضاء.**

تعريفات واختصارات:

**1.3 العضو :العميل الذي يستخدم خدمات الصالة.**

**المدرب: المسؤول عن وضع الجداول والخطط.**

**الطبيب: الاختصائي المسؤول عن التقارير الصحية.**

**الموظف: يشمل موظف الاستقبال والمحاسب.**

**الجلسة التدريبية: موعد محجوز للتمرين الفردي او الجماعي.**

2) وصف عام:

2.1 منظور النظام:

**النظام عبارة عن تطبيق ويب أو موبايل يستخدم من قبل الأعضاء ,الموظفين,المدربين,الأطباء لتنفيذ وإدارة المهام اليومية بسهولة.**

2.2 وظائف النظام الرئيسية:

**1-تسجيل الأعضاء وإنشاء حسابات جديدة**

**2-حجز الجلسات التدريبية وتتبع الحضور**

**3-إعداد ومتابعة الجداول الرياضية والخطط الغذائية**

**4-تسجيل بيانات الفحوصات الصحية وإصدار التقارير**

**5-الدفع الالكتروني ومتابعة الاشتراكات**

**6-تةفير تقارير إدارية لدعم اتخاذ القرار**

2.3 خصائص المستخدمين:

**أعضاء: مستخدمين غير تقنيين يحتاجون واجهه سهله الاستخدام**

**موظفو الاستقبال :لديهم خبرة متوسطة في استخدام الحاسوب**

**المدربون و الأطباء: يحتاجون الوصول السريع الى بيانات الأعضاء**

**الإدارة: تحتاج لوحات تحكم وتقارير مفصلة**

2.4 القيود:

**متوافق مع اجهزه الكمبيوتر والهواتف الذكية**

**متاح باللغتين العربية والانجليزية**

**يجب ان يعمل على مدار 24 ساعه دون توقف**

2.5الفرضيات

**جميع المستخدمين لديهم اتصال بالانترنيت**

**الاجهزه المستخدمة تلبي متطلبات النظام الدنيا**

**وظائف النظام الأساسية:**

**تحسين كفاءة الإدارة**

**تقليل العبء الورقي واليدوي**

**رفع جودة الخدمة المقدمة للأعضاء**

**تحقيق التكامل بين الخدمات الصحية والرياضية**

**توفير تقارير دقيقة تساعد في اتخاذ القرار**

**المستفيدون من النظام:**

**الأعضاء (العملاء)**

**المدربون**

**موظفو الاستقبال**

**الأطباء وأخصائيو التغذية**

**المتطلبات الوظيفية:**

-تسجيل الأعضاء: يمكن للمستخدم إنشاء حساب جديد

-حجز الجلسات التدريبية :يمكن للمستخدم تسجيل الدخول والخروج

-الجدول الرياضي: يمكن للمدرب إنشاء او تعديل جدول التمارين

-الخطط الغذائية: يمكن للعضو طلب خطة غذائية

-يمكن للطبيب تسجيل بيانات الفحوصات

-يمكن للعضو الدفع اونلاين

**المتطلبات الغير الوظيفية:**

-الاعتمادية: يجب للنظام ان يعمل بدون توقف

-الأمان :جميع بيانات الأعضاء يجب ان تكون ماشفرة

(التحقق من صلاحيات كل مستخدم قبل تنفيذ أي عملية)

-الاستجابه: يجب الا تتجاوز وقت استجابة أي عمليه 3 ثوانٍ

-قابلية الاستخدام: يجب النظام سهل الاستخدام للمستخدمين غير التقنيين

-الدعم متعدد اللغات :النظام يدعم اكثر من لغة

-قابلية التوسع :يجب ان يقبل النظام إضافة فروع جديدة للصالة دون تغييرات كبيرة

-الصيانة والتحديث: يجب ان يكون من السهل تحديث النظام وإضافة خصائص جديدة

**المخططات التصميمية:**

**1) مخطط حالة الاستخدام :**

يوضح مخطط حالة الاستخدام دورة حياة العضو داخل النظام

\*\*حاله الاستخدام: حجز جلسة تدريبية

الهدف :حجز جلسة تدريبية للعضو

الممثلون :عضو ,موظف استقبال

الشروط السابقة :العضو مسجل الدخول ولديه صلاحيات الحجز

العمليات :يختار العضو او الموظف الجلسة و التاريخ ويتم تأكيد الحجز

حالات الخطأ: عدم وجود موعد شاغر \_يقترح النظام موعداً بديلاً



**وصف حالة استخدام لحجز جلسة تدريب:**

الاسم**: حجز جلسة تدريب**

وصف موجز**: يحجز الزبون جلسة في صالة التدريب**

الشروط السابقة: -**الزبون مصرح له بحجز جلسة تدريب**

**- الزبون مسجل الدخول إلى النظام**

الشروط اللاحقة**: جلسة التدريب محجوزة**

حالات الخطأ**: لا يوجد موعد شاغر**

حالة النظام في حالة حدوث خطأ**: لم يحجز الزبون جلسة تدريب**

الجهات الفاعلة: **موظف الاستقبال**

المحفز**: يحتاج الزبون إلى جلسة تدريب**

العمليات القياسية**:- يختار الموظف الجلسة**

**- يختار الموظف التاريخ ومدة الحجز**

**- يؤكد النظام ان جلسة التدريب شاغرة**

**-يؤكد الموظف الحجز**

الإجراءات البديلة**: -جلسة التدريب في موعد غير مناسب**

**- يقترح النظام جلسة تدريب بديلة**

**- يختار موظف الاستقبال جلسة بديلة ويؤكد الحجز**

**وصف حالة استخدام لإصدار تقارير صحية:**

الاسم**: إصدار تقارير صحية**

وصف موجز**: يصدر الطبيب الرياضي تقرير صحي عن الزبون**

الشروط السابقة: -**الزبون مسجل عند الطبيب الصحي**

**- الزبون مسجل الدخول إلى النظام**

الشروط اللاحقة**: التقرير الصحي تم إصداره**

حالات الخطأ**: لا يوجد تقارير فارغة أو التأخر في إصدار التقرير**

حالة النظام في حالة حدوث خطأ**: لم يصدر للزبون تقرير صحي**

الجهات الفاعلة: **الطبيب الرياضي**

المحفز**: معرفة الحالة الصحية للزبون**

العمليات القياسية**:- يختار الطبيب الرياضي التاريخ لفحص الزبون**

**- يؤكد النظام على إنه تم فحص الزبون**

**- يؤكد الطبيب التقرير**

الإجراءات البديلة**: -إصدار تقرير لزبون أخر يحمل نفس الاسم**

**- يقترح النظام إدخال الاسم الثلاثي**

**- يختار النظام التقرير المناسب للزبون**

**Class diagram:**

*لدينا* الكيانات التالية:

عضو ، طبيب ، مدرب ، محاسب ، موظف استقبال ، موظف ، إدارة ، إيصال دفع ، جلسة تدريبية ، خطة غذائية ، تقرير طبي .

العضو يتفاعل مع الموظف لتسجيل العضوية

\* الارتباطات:

الطبيب يقوم بفحص العضو

المدرب يدرب العضو

العضو يدفع للمحاسب

\* الوراثة:

يرتبط الموظف بموظف الاستقبال والطبيب والمدرب والمحاسب.

\*التركيب:

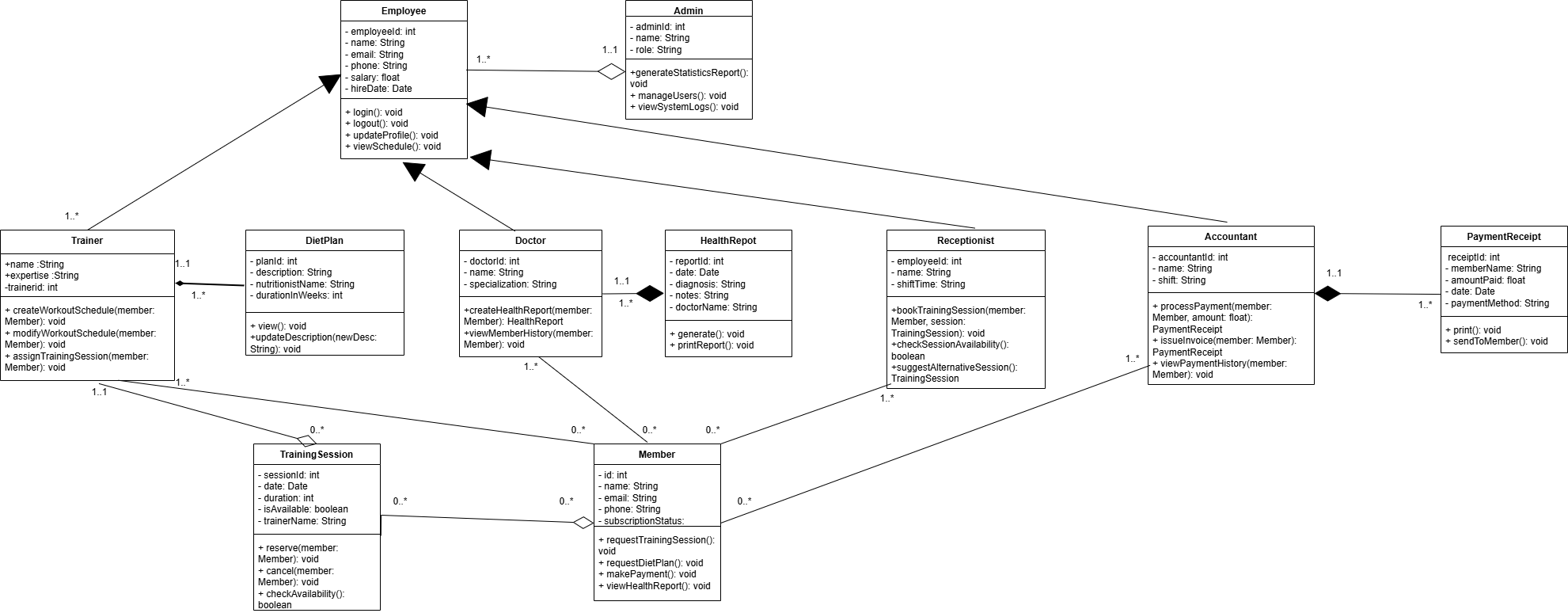
طبيب مع تقرير صحي(لا يتواجد تقرير صحي بدون طبيب)

مدرب مع خطة غذائية(لا تتواجد خطة بدون مدرب)

محاسب مع إيصال دفع.

\*التجميع: جلسة تدريب مع عضو

جلسة تدريب مع مدرب , إدارة مع موظف.

Class Diagram

Entity Relationship Diagram :

لدينا الكيانات التالية:

العضو ،المدرب ،الطبيب ،المحاسب ،الجلسة التدريبية ،إيصال الدفع ،تقرير صحي ،موظف استقبال .

العضو يحجز عدة جلسات

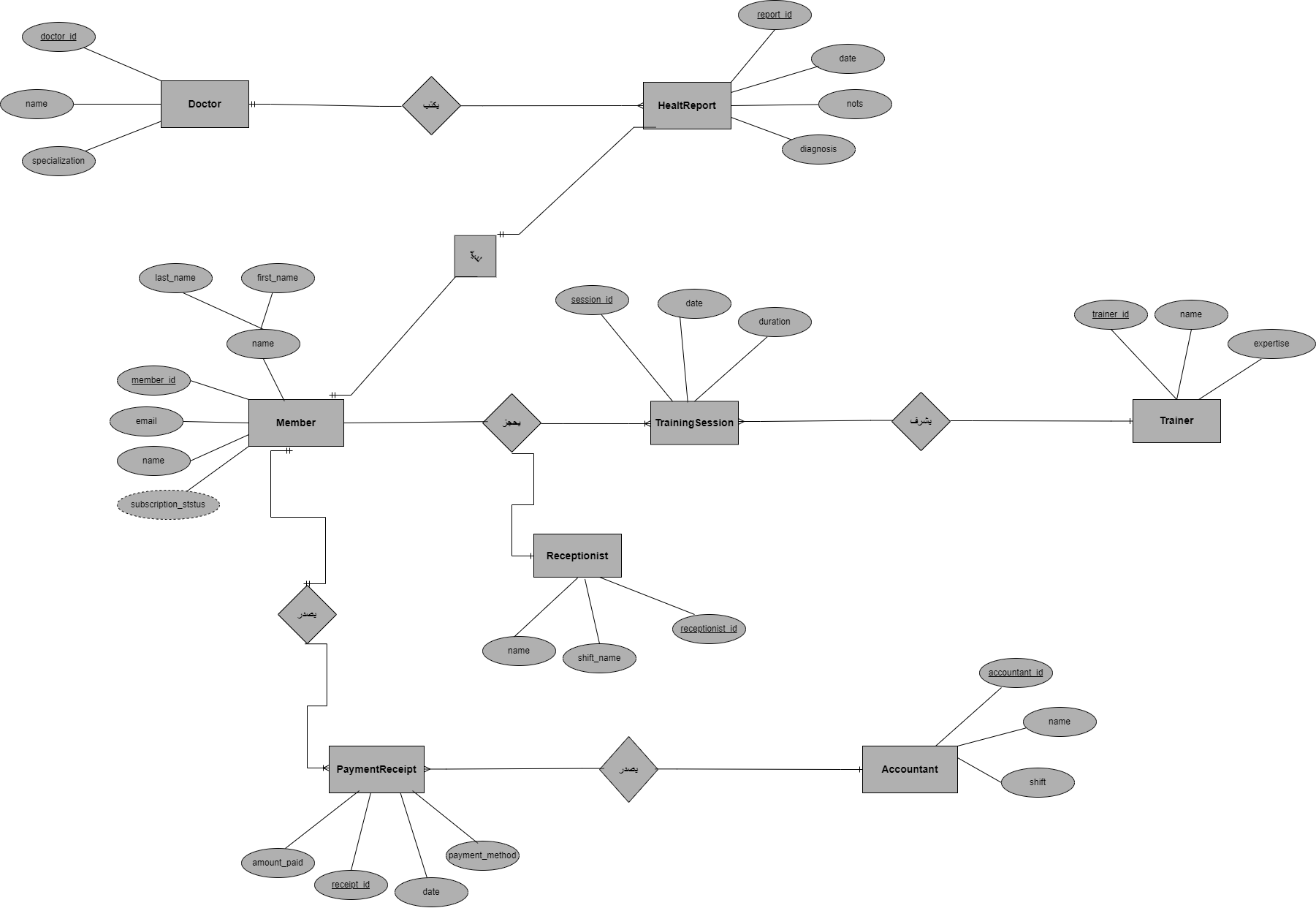
كل جلسة يشرف عليها مدرب

لكل عضو تقرير صحي

الموظف يحجز عدة جلسات

العضو يدفع اكثر من مره

المحاسب يصدر إيصالات الدفع

ERD Diagram k

**الكود الرئيسي لواجهة برمجة التطبيقات (API)**

تم تطوير واجهة برمجة تطبيقات بسيطة باستخدام إطار عمل Django REST Framework لإدارة البيانات . يتضمن الكود النماذج (Models)، المحولات

(Serializers)، طرق العرض (Views)، ومسارات URL.

**نموذج العضو member/models.py**

from django.db import models

Class Member(models.model):

Name =models.CharField(max\_length=100)

Subscription\_status=

models.CharField(max\_length=100)

def \_\_str\_\_(self):

return self.name

**member/serializers.pyمحول العضو**

from rest\_framework import serializers

from .models import member

class MemberSerializer(serializers.ModelSerializer):

class Meta:

model = Member

fields = '\_\_all\_\_'

**Member/view.py طرق عرض العضو**

from rest\_framework import view viewsts

from .models import Member

from .serializers import MemberSerializer

Class MemberViewSet(viewsets.ModelViewSet):

Queryset=Member.object.all()

Serializer\_class=MemberSerializer

**Member/urls.pyلتطبيق الأعضاء URLمسارات**

from rest\_framework.routers import DefaultRouter

from .view import MemberViewSet

Router= DefaultRouter()

Router.register(r’member ‘,MemberViewSet)

Urlpatterns=router.urls

**Member\_management\_system/urls.py**

from django.contrib import admin

from django.urls import path ,include

from member .urls import router

Urlpatterns=[

Path (“admin/”,admin.site.urle),

Path(“api/”, include(router.urls)),

]