Achievement gym

يمثل هذا المشروع نظاماً برمجياً لإدارة الأندية الرياضية بعنوان Achievement GYM

تتمثل المشكلة الأساسية في غياب أنظمة رقمية متكاملة تساعد المدربين والإداريين على متابعة الأعضاء، تنظيم الجلسات التدريبية، وتسجيل التمارين بشكل آلي. الاعتماد على الطرق التقليدية يؤدي إلى ضياع البيانات وصعوبة متابعة تطور الأعضاء على المدى الطويل.

يهدف المشروع إلى تطوير تطبيق ويب متكامل يتيح للمدربين إدارة الأعضاء والجلسات التدريبية، مع توفير واجهة آمنة لتسجيل الدخول، وإضافة التمارين، وتقييم التقدم الفردي لكل لاعب. كما يوفر النظام خاصية متابعة تطور اللاعب في التمارين المختلفة بمرور الوقت.

تم بناء المشروع

Back-end: باستخدام nodejs وإطار عمل Express.js مع استخدام MongoDB لإدارة قواعد البيانات.

Front-end: باستخدام Reactjs

النتائج المتوقعة من النظام تتمثل في تسهيل الإدارة الرقمية للنادي، تقليل الأخطاء الناتجة عن التسجيل اليدوي، وتحقيق سهولة أكبر في متابعة أداء اللاعبين.

من خلال هذا النظام يمكن القول إن المشروع يشكل خطوة عملية نحو التحول الرقمي في المجال الرياضي ويسهم في تحسين جودة الخدمات المقدمة للأعضاء والمدربين على حد سواء.

فهرس المحتويات

الفصل الأول مقدمة 4
الفصل الثاني الدراسة النظرية 6
المفاهيم النظرية الأساسية 7
إدارة قواعد البيانات 7
الموثوقية والأمان 7
إدارة الجلسات التدريبية 7
مراقبة التطور 7
التقنيات وأطر العمل المستخدمة 8
الفصل الثالث التطبيق العملي 10
الأدوات وبيئة العمل المستخدمة في إنجاز المشروع 11
تحليل النظام البرمجي 11
Back-end 11
Front-end 11
التكامل بين Front-end وBack-end 12
التوصيف الفني للعمل 13
الواجهات 13
توصيف قاعدة البيانات 14
Coaches 14
Gym Members 15
Exercises 16
Sessions 16
Counters 16
Password Reset Token 16
العلاقات بين collections 17
التحديات أثناء التطوير 18
المقترحات والتوصيات 18

الفصل الأول مقدمة

يأتي هذا المشروع كخطوة لتسهيل عملية إدارة النوادي الرياضية وتسهيل مراقبة التطور للاعب وتسجيل جلساته. فكل لاعب في أي نادي رياضي لديه تمارين محددة يمارسها تختلف حسب نوع الرياضة التي يمارسها وطريقة مدربه في اختيارها ويكون هدفه التطور في هذه الرياضة من خلال جدول دقيق يحافظ عليه كتمارين ويزيده كشدة.

يقاس هذا التطور حسب نوع الرياضة (زيادة الحمل في بناء الأجسام ورفع الأثقال، زيادة المسافة المقطوعة في الجري والسباحة، زيادة السرعة ...)

وهنا تأتي أهمية موقع Achievement GYM حين يحدد المدربون المختصون المسجلون في الموقع التمارين بدقة مع الشدة التدريبة لكل منها ويصل إليها اللاعب بكل سهولة ليلتزم بها

إضافة إلى تمكين اللاعب من الوصول إلى مخططات بيانية دقيقة تبين له تطور الشدة التدريبية في كل تمرين على حدا ليتمكن من اكتشاف المشكلة التي تمنعه من التطور في حال وجودها وفي حال كان تطوره طبيعي فتسمح له بمراقبته وتحسينه بالتعاون مع المدرب المختص

الفصل الثاني الدراسة النظرية

المفاهيم النظرية الأساسية

إدارة قواعد البيانات

استخدام قاعدة بيانات MongoDB لتخزين معلومات اللاعبين والمدربين والجلسات الخاصة بكل لاعب إضافة إلى التمارين المنظمة حسب نوع الرياضة وربطها معاً بشكل واضح ومناسب.

الموثوقية والأمان

عند التسجيل في الموقع يختار المستخدم نوع حسابه (مدرب أو لاعب) تختلف صلاحية كل منهما عن الآخر يكون للمستخدم كلمة سر قوية يجب أن تحقق شروط تضمن الأمان وتخزن في قاعدة البيانات مشفرة لا يستطيع أحد الاطلاع عليها في حال نسيها المستخدم يمكن أن يقوم بإعادة تعيينها بعد طلب رمز يصله على بريده الإلكتروني وإدخاله للتحقق من هويته.

إدارة الجلسات التدريبية

تتكون الجلسة التدريبية في النظام من مجموعة من التمارين والأوزان الموافقة لكل تمرين حيث يحدد المدرب الجلسة لكل لاعب يتدرب لديه مع مدتها والتاريخ الذي سوف تكون فيه لا يستطيع أي أحد إضافة جلسة تدريبية للاعب إلا المدرب المشرف عليه فقط ولا يستطيع أحد الاطلاع على الجلسة سوا اللاعب الذي سوف يلعبها والمدرب الذي أنشأها

مراقبة التطور

في صفحة معلومات اللاعب هناك قسم مخصص لمراقبة التطور يحوي هذا القسم على قائمة بالتمارين الخاصة باللاعب يختار منها التمرين الذي يريد مراقبة تطوره أو حالته فيه عندها يرسم خط بياني محوره الشاقولي هو الوزن الذي تدرب فيه والأفقي هو تاريخ تدربه بهذا الوزن من خلال المخطط يراقب اللاعب هل أداءه يتجه للتحسن (الخط البياني صاعد) أم أنه ثابت منذ مدة (الخط البياني ثابت) أو يتجه نحو الأسوأ لسبب ما (الخط البياني هابط)

الوصول وخصوصية المعلومات

لا يمكن للاعب الوصول إلى معلومات أي لاعب آخر بما في ذلك التطور والجلسات ومعلومات الاتصال يصل اللاعب فقط إلى المعلومات التي تخصه

لا يستطيع المدرب الوصول إلى معلومات اللاعبين المتدربين عند مدرب اخر أيضا بما في ذلك معلومات الاتصال والجلسات ومخططات التطور

يستطيع اللاعب الوصول إلى معلوماته الشخصية بسهولة.

يستطيع المدرب الوصول إلى معلومات جميع اللاعبين المتدربين لديه

التقنيات وأطر العمل المستخدمة

Nodejs: منصة تشغيل java Script على الخادم تسمح بإنشاء تطبيقات سريعة وفعالة. استخدمناها كأساس لتشغيل الـ .Backend

Express.js: إطار عمل مبني على Node.js يُبسط بناء الـ APIs وإدارة الطلبات. وفر علينا الكثير من الوقت في هيكلة الخادم.

MongoDB: قاعدة بيانات NoSQL مرنة لتخزين البيانات على شكل وثائق (.(Documents اعتمدنا عليها لتخزين بيانات الأعضاء والجلسات.

MongoDB Atlas: خدمة سحابية لإدارة MongoDB بسهولة وأمان. وفرت لنا الوصول للبيانات من أي مكان بدون إعداد خوادم محلية.

MongoDB Compass: أداة رسومية لإدارة واستعراض بياناتMongoDB استخدمناها لفحص البيانات والتحقق من صحتها أثناء التطوير.

Mongoose: مكتبة لربط Node.js مع MongoDB وإدارة الـ Schemas ساعدتنا على تعريف الموديلات والتعامل مع البيانات بسلاسة.

Bcryptjs: مكتبة لتشفير كلمات المرور بطريقة آمنة. ضمنت لنا حماية حسابات المستخدمين عند التسجيل والدخول.

Cors: ميدل وير يسمح للواجهة الأمامية Reactبالتواصل مع الخادم بدون مشاكل Cross-Origin استخدمناه لتمكين الاتصال بين السيرفر والـ Frontend

Dotenv مكتبة لإدارة المتغيرات البيئية Environment Variablesاستخدمناها لتخزين المفاتيح السرية مثل مفتاح JWT ومعلومات البريد.

Joi: مكتبة للتحقق من صحة البياناتValidation وظيفتها منع إدخال بيانات خاطئة أو غير مكتملة للنظام.

jsonwebtoken (JWT): مكتبة لإنشاء والتحقق من tokens لتأمين الجلسات. استخدمناها للتحكم بالصلاحيات بين المدرب والعضو.

Multer: مكتبة للتعامل مع رفع الملفات مثل الصور. مكنت المستخدمين من رفع صورهم الشخصية

Nodemailer: مكتبة لإرسال البريد الإلكتروني من الخادم. اعتمدناها في ميزة استرجاع كلمة المرور والتحقق من الحساب.

Path: مكتبة مضمنة في Node.js لإدارة المسارات Paths والملفات. ساعدتنا في تحديد أماكن حفظ الملفات مثل الصور.

JSX: اختصار لـJavaScript XML ، وهي صيغة تتيح كتابة أكواد JavaScript و HTML معاً داخل React تجعل بناء الواجهات أسهل وأكثر وضوحاً.

CSS: لغة تنسيق الصفحات لتحديد مظهر وتصميم العناصر في واجهة المستخدم. تُستخدم لتخصيص الألوان والخطوط والتباعد.

JavaScript (JS): لغة برمجة ديناميكية تُستخدم لبناء التفاعلات على صفحات الويب. تشكل الأساس الذي بُنيت عليه React والتقنيات الأخرى.

React: مكتبة JavaScript لبناء واجهات المستخدم التفاعلية. تعتمد على المكوناتComponents وتُسهل إعادة استخدام الكود.

React Router: مكتبة لإدارة التنقل بين الصفحات داخل تطبيقات React أحادية الصفحة (SPA) تتيح إنشاء مسارات ديناميكية وسهلة التحكم.

JSON Server: أداة تُستخدم لإنشاء REST API وهمي باستخدام ملفات JSON مفيدة لاختبار الواجهات الأمامية أثناء التطوير.

React Charts: مكتبة لعرض الرسوم البيانية في تطبيقات React تُستخدم لتمثيل البيانات بشكل مرئي مثل الإحصاءات والتقارير.

useState: هو Hook مدمج في React لإدارة الحالة (state) داخل المكونات. يتيح تحديث البيانات وإعادة عرض الواجهة عند تغيرها.

useEffect: هو Hook يُستخدم لتنفيذ عمليات جانبية (Side Effects) مثل جلب البيانات أو تحديث DOMبعد كل إعادة عرض.

useParams: Hook من React Router يُستخدم لاستخراج المتغيرات الديناميكية من روابط الصفحات .(URLs) مفيد في استرجاع بيانات بناءً على معرف محدد.

الفصل الثالث التطبيق العملي

الأدوات وبيئة العمل المستخدمة في إنجاز المشروع

تم تطوير المشروع في بيئة Visual Studio code على نظام تشغيل Windows

Back-end باستخدام nodejs مع إطار عمل express.js

Data Base باستخدام MongoDB atlas لنسخة المنتج و MongoDB محلية للتطوير

Versions control system باستخدام Git & GitHub

API Testing and Documentation باستخدام Post Man

Front-end باستخدام Reactjs

تحليل النظام البرمجي

النظام ينقسم إلى جزئيين:

Back-end

وهو Rest Full APIs يستخدم بنية MVC في تنظيم ملفات الأكواد البرمجية مما يوفر سهولة صيانة وتوسع المشروع في الإصدارات القادمة يتألف من:

a. Models وهي الجزء المسؤول عن التعامل مع قاعدة البيانات وتحديد شكل كل collection

b. Routes تحديد End Points التي سوف يتم استخدامها في الربط

c. Controllers وهي الجزء الذي يعالج البيانات ويعدها بعد استدعاء end point محدد حتى تكون جاهزة للتعامل مع قاعدة البيانات والتعديل عليها.

Front-end

تم تطوير واجهة المستخدم باستخدام مكتبة React.js، وهي المسؤولة عن عرض البيانات القادمة من الخادم بشكل تفاعلي وسلس. تم تقسيم المشروع إلى مجموعة من المكونات (Components) القابلة لإعادة الاستخدام، مما يسهل عملية التطوير والصيانة.

a. المكونات (Components) تمثل كل جزء من واجهة المستخدم مثل النماذج، الجداول، والرسوم البيانية.

b. إدارة الحالة (State Management) تم الاعتماد على الـ Hooks مثل useState و useEffect لإدارة البيانات الداخلية وجلب البيانات من الخادم.

c. التنقل (Routing) باستخدام مكتبة React Routerالتي تتيح الانتقال بين الصفحات المختلفة (مثل صفحة تسجيل الدخول، صفحة الجلسات، صفحة معلومات اللاعب) دون الحاجة لإعادة تحميل كامل للتطبيق.

d. الرسوم البيانية :تم استخدام مكتبة React Chartsلعرض بيانات التمارين والجلسات بشكل مرئي لمتابعة تقدم اللاعبين.

e. تصميم الواجهات :تم استخدام CSSلتنسيق الصفحات وضبط الألوان والتخطيطات لتوفير تجربة مستخدم جذابة.

f. واجهة الـ Front-end تتواصل مع الـ Back-end عبر REST APIs، بحيث يتم جلب البيانات (مثل معلومات اللاعبين والجلسات) وإرسالها بشكل آمن ومنظم.

التكامل بين Front-end وBack-end

يعمل النظام بشكل متكامل بحيث يتواصل الـ Front-end مع الـ Back-end عبر بروتوكول HTTP باستخدام .RESTful APIsيقوم الـ Front-end بإرسال الطلبات مثل تسجيل الدخول، عرض الجلسات، إضافة لاعب جديد أو متابعة التمارين، بينما يقوم الـ Back-end بمعالجة هذه الطلبات والتحقق من صحتها والتعامل مع قاعدة البيانات.

عند قيام المستخدم بعملية تسجيل الدخول مثلًا، يرسل الـ Front-end بيانات البريد الإلكتروني وكلمة المرور إلى الخادم.

الخادم يتحقق من البيانات عبر قاعدة البيانات ويعيد رمز تحقق إذا كانت صحيحة.

الـ Front-end يخزن هذا الرمز ويستخدمه للوصول إلى باقي الخدمات بشكل آمن.

في حالة عرض التمارين أو تقدم اللاعب، يقوم الـ Front-end بجلب البيانات من الخادم وإظهارها للمستخدم على شكل جداول أو رسوم بيانية تفاعلية. هذا التكامل يضمن أن تكون البيانات آمنة، محدثة باستمرار، وقابلة للوصول بسهولة من خلال واجهة مستخدم واضحة وسهلة الاستخدام.

التوصيف الفني للعمل

الواجهات

- إنشاء حساب جديد

تتألف عملية تسجيل مستخدم جديد من خطوتين رئيسيتين في كل خطوة يدخل المستخدم مجموعة من المعلومات منها معلومات اتصال ومنها المعلومات الرياضية من اختيار المدرب (في حال كان المستخدم لاعب) ونوع الرياضة...

- تسجيل الدخول

يعتمد تسجيل الدخول على إدخال المستخدم لبريده الإلكتروني وكلمة السر ونوع حسابه

- طلب رمز إعادة تعيين كلمة السر

في حال نسي المستخدم كلمة السر الخاصة بحسابه فإنه سوف يقوم بإعادة تعيينها من هذه الصفحة التي يتم الوصول إليها من صفحة تسجيل الدخول عند الحاجة يدخل المستخدم فيها بريده الإلكتروني ونوع الحساب وينتظر الكود التي سوف يصله على البريد الإلكتروني

- إعادة تعيين كلمة المرور

وهي الخطوة التي ينتقل إليها المستخدم بعد وصول الكود إلى بريده الإلكتروني حتى يعيد تعيين كلمة المرور يطلب منه إدخال الكود وكلمة السر الجديدة وتكرارها لتجنب حدوث خطأ غير متوقع أثناء الإدخال

- عرض الجلسات

وهي الصفحة الرئيسية التي يصل إليها المستخدم عند اكتمال تسجيل الدخول تختلف هذه الصفحة كمحتوى بين اللاعب والمدرب حيث يحوي في كلا الحالتين على بطاقات كل بطاقة خاصة بجلسة تدريبية يحوي المعلومات الأساسية عنها (اسم اللاعب والعضلات المستهدفة والمدة والتاريخ وهل حالتها انتظار ام أنها تمت) ولكن الفرق أن اللعب يجب فيها الجلسات التدريبية الخاصة به أما المدرب يجد فيها الجلسات الخاصة بكل اللاعبين الموجودين لديه

- تفاصيل جلسة محددة

يتم الوصول إلى هذه الصفحة عند الضغط على أحد بطاقات الصفحة السابقة حيث تحوي هذه الصفحة على جدول يحوي اسم كل تمرين مع الوزن الذي كون الشدة التدريبة له حاليا

يقوم المدرب من هذه الصفحة أيضا بتغيير حالة الجلسة عندما يكون اللاعب أكملها فعلياً

- عرض اللاعبين

هذه الصفحة تظهر فقط للمستخدم صاحب الحساب من النوع مدرب وهي تحوي على بطاقات كل بطاقة تمثل أحد اللاعبين المتدربين لديه (صورة واسم كامل)

- الملف الشخصي

هذه الصفحة يصل إليها اللاعب من شريط التنقل تبويب profile ويصل إليها المدرب عند الضغط على أحد بطاقات الصفحة السابقة تحوي هذه الصفحة على معلومات اللاعب (اسمه ومعلومات الاتصال ونوع الرياضة واسم المدرب ونوع الرياضة) إضافة إلى الجزء الخاص بمراقبة التطور وهو يتكون من قائمة منسدلة تحوي على كل التمارين التي لعبها المستخدم منذ إنشاء الحساب حتى الوقت الحالي يختار المستخدم التمرين الذي يريد مراقبة تطوره فيه عندها يرسم مخطط بياني منظم يحوي على الاوزان المستخدمة على مدار الفترة التدريب

وإذا كان المستخدم مدرب تظهر له زر إضافي هو إضافة جلسة لهذا اللاعب

- إضافة جلسة جديدة

يصل إليها المدرب فقط عندما يضغط على زر Add Session ليضيف جلسة تدريبة للاعبين المدربين لديه حصراً ويحدد منها المدرب المدة والتاريخ ثم ينتقل لإضافة تمارين فيختار العضلة المستهدفة ويختار تمريناً من تمارينها ويدخل الوزن يكرر العملية على كل التمارين ويحفظ الجلسة

- مكونات مشتركة Header, Footer, Navigation Bar, Session Card, Member Card

توصيف قاعدة البيانات

تم استخدام قاعدة بيانات MongoDB لتنظيم البيانات، حيث تتألف من عدة Collections مترابطة فيما بينها هي:

Coaches

نخزن فيها المعلومات الخاصة بكل المستخدمين من نوع مدرب المعلومات وهي

- رقم ID فريد يميز كل مدرب
- الاسم الأول first name
- الاسم الأخير last name
- سيرة ذاتية خاصة بالمستخدم وهي نص طوله لا يتجاوز ال 500 محرف

- رقم الهاتف الخاص بالمدرب في حال احتاج أحد المتدربين التواصل معه لاستفسار أو اشتراك phone number
- البريد الإلكتروني الذي استخدمه للتسجيل واسترجاع كلمة السر عند الحاجة email
- كلمة السر التي سوف يستخدمها لتسجيل الدخول وتخزن مشفرة لحماية الخصوصية كلمة السر لها شروط أن تحوي حرف كبير واحد على الأقل ومحرف خاص على الأقل ورقم على الأقل ولا يقل طولها عن 8 محارف password
- نوع الرياضة حيث يحوي النظام حاليا على 3 رياضات أساسية (بناء الأجسام ورفع الأثقال والكاليستانيكس) spot type
- رابط الصورة وهي الصورة التي اختارها المدرب مخزنة في السيرفر ومسارها مخزن في هذا الحقل imageUrl

Gym Members

نخزن فيها المعلومات الخاصة بكل المستخدمين من نوع لاعب المعلومات وهي

- رقم ID فريد يميز كل لاعب
- الاسم الأول first name
- الاسم الأخير last name
- سيرة ذاتية خاصة بالمستخدم وهي نص طوله لا يتجاوز ال 500 محرف
- رقم الهاتف الخاص باللاعب في حال احتاج المدرب التواصل معه phone number
- البريد الإلكتروني الذي استخدمه للتسجيل واسترجاع كلمة السر عند الحاجة email
- كلمة السر التي سوف يستخدمها لتسجيل الدخول وتخزن مشفرة لحماية الخصوصية كلمة السر لها شروط أن تحوي حرف كبير واحد على الأقل ومحرف خاص على الأقل ورقم على الأقل ولا يقل طولها عن 8 محارف password
- نوع الرياضة حيث يحوي النظام حاليا على 3 رياضات أساسية (بناء الأجسام ورفع الأثقال والكاليستانيكس) spot type
- رابط الصورة وهي الصورة التي اختارها اللاعب مخزنة في السيرفر ومسارها مخزن في هذا الحقل imageUrl
- معرف المدرب الذي اختاره اللاعب عند التسجيل حتى يشرف عليهcoach id

Exercises

- معرف خاص بكل تمرين exercise id
- اسم التمرين exercise name
- نوع الرياضة sport type
- العضلة المستهدفة من هذا التمرينtarget muscle

Sessions

- معرف خاص بكل تمرين session id
- معرف اللاعب الذي سوف يلعب هذه الجلسة member id
- معرف المدرب الذي أنشأ هذه الجلسة session id
- المدة التقريبية التي تحتاجها هذه الجلسة duration
- التاريخ الذي يخطط أن تكون هذه الجلسة فيه date
- حالة الجلسة هي في حالة انتظار ام انها تمت state
- مصفوفة التمرين التي تحويها هذه الجلسة وهي مجموعة أغراض كل مهنا يحوي تمرين محدد مع وزن محدد له exercises

Counters

تحوي حقل لكل model اخر في قاعدة البيانات يكون نستخدمها لسهولة استدعاء كل غرض عند الحاجة وتحوي

- اسم العداد وهو اسم الcollection الموافق لهذا العداد counter name
- الرقم الذي وصلنا إليه وهو count

Password Reset Token

يخزن مؤقتا الرمز الذي وصل للمستخدم أثناء عملية إعادة تعيين كلمة السر وتحوي

- معرف المستخدم user id
- الرمز الذي أرسل على البريد الإلكتروني token

العلاقات بين collections

- اللاعب والمدرب(M-1):

كل مدرب يشرف على عدة لاعبين وكل لاعب لديه مدرب وحيد.

- المدرب والجلسات التدريبية(M-1):

كل مدرب يشرف على عدة جلسات تدريبية وكل جلسة تدريبية يشرف عليها مدرب وحيد

- اللاعب والجلسات التدريبية(M-1):

كل لاعب يلعب عدة جلسات تدريبية وكل جلسة تدريبية يملكها لاعب واحد

- التمارين والجلسات التدريبية(M-1):

كل جلسة تدريبية تحوي عدة تمارين وكل تمرين موجود في عدة جلسات تدريبية

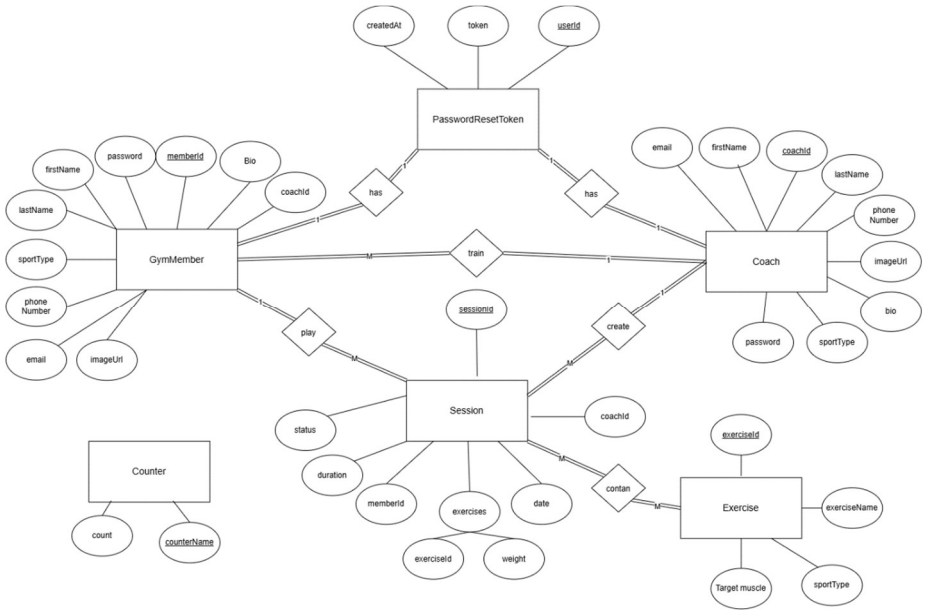
- رمز استعادة كلمة المرور واللاعبين(1-1) :

كل رمز يعود إلى لاعب وكل لاعب يملك رمز واحد

- رمز استعادة كلمة المرور والمدربين(1-1) :

كل رمز يعود إلى مدرب معين وكل مدرب لديه رمز وحيد

يوضح الشكل (1-3) مخطط ERD لقاعدة البيانات بكل الكيانات والعلاقات الموضحة بينها



الشكل (3-1) Entity Relationship diagram

التحديات أثناء التطوير

- ربط الواجهات الأمامية مع كل API بفعالية.
- إدارة الحالة بين مكونات React
- التعامل مع رفع الصور وعرضها
- تنسيق البيانات القادمة من قاعدة البيانات بشكل مناسب للعرض
- حظر الوصول إلى MongoDB atlas من بوابات الإنترنت الخاصة بالسورية للاتصالات

المقترحات والتوصيات

Achievement GYM 1.0.0 هو النسخة الأولى من النظام وهناك خطة لتطوير النظام وإضافة ميزات جديدة في الإصدارات القادمة منها:

- تحسين تجاوب الواجهات مع أحجام الشاشات المختلفة فالنسخة 1.0.0 موجهة للمستخدمين على PC
- إضافة خيار لطباعة البرنامج التدريبي او حفظه بصيغة PDF
- إضافة قسم مخصص للمنشورات حيث يستطيع المدربين نشر الأخبار الجديدة (اعتذار عن يوم معين – إعلانات لمكملات غذائية – فعاليات يقيمه النادي وبطولات ...)

- إضافة قسم للتواصل بين اللاعبين والمدربين لاستشارات سريعة تتعلق بالنظام الغذائي
- إضافة قسم لبيع المعدات الرياضية والمكملات الغذائية والمنتجات المختلفة مع بوابة دفع إلكتروني مناسبة حيث أصبح التعامل بها ممكن
- إضافة تقييم كامل للمتدرب في كافة التمارين معا