

سند نهایی معرفی و مشخصات پروژه

اپلیکیشن مدیریت برنامه بدنسازی برای مربی و شاگرد (MVP)

۱) هدف پروژه

در باشگاه‌ها مربی‌ها برنامه تمرینی و غذایی را روی کاغذ می‌نویسند و شاگرد مدام درباره اجرای حرکات سوال می‌پرسد. هدف این پروژه ساخت یک سیستم رایگان است که:

- مربی بتواند برای شاگرد **برنامه تمرینی/تغذیه/مکمل** بنویسد.
 - شاگرد در اپ بداند **امروز چه تمرینی دارد**، حرکت‌ها را با **ویدیو** ببیند، و اجرای تمرین را **تیک بزند**.
 - شاگرد بعد تمرین یک **بازخورد متنی** بدهد (مثل: عالی بود/بد بود...).
 - سیستم **یادآوری** (Push) برای تمرین و مکمل داشته باشد.
 - مربی از طریق یک **پنل تحت وب** شاگردها و برنامه‌ها را مدیریت کند.
- این سیستم پولی نیست و هدفش جذب کاربر و استفاده داخل باشگاه است.

۲) محصولات و اجزای سیستم

۲.۱ اپ شاگرد (Android)

- ورود با شماره موبایل + OTP
- صفحه Today (تمرین امروز، وضعیت انجام، یادآوری‌ها)
- نمایش برنامه تمرینی و ویدیوهای حرکات
- تیک زدن حرکات انجام‌شده
- پایان تمرین و ثبت "امتیاز متنی" + یادداشت اختیاری
- (اختیاری) نمایش برنامه تغذیه و تیک وعده‌ها

- ثبت و مشاهده عکس‌های پیشرفت (اختیاری و غیر اجباری)
- ذخیره آفلاین برنامه‌ها و سینک بعداً (برای اینترنت ضعیف باشگاه)

۲.۲ پنل مربی (Web)

- ورود مربی با شماره موبایل + OTP
- اضافه کردن شاگرد با شماره موبایل
- ارسال پیامک دعوت برای نصب و ورود شاگرد (لینک + توکن)
- تکمیل پروفایل شاگرد (قد/وزن/هدف/آسیب‌ها...)
- آپلود عکس (اختیاری)
- ساخت برنامه تمرینی چند هفته‌ای + تعداد روز تمرین در هفته
- چینش روزهای تمرین و انتخاب حرکات از بانک حرکات
- تنظیمات هر حرکت: ست، تکرار، استراحت، تمپو، توضیح مربی
- مشاهده تاریخچه انجام تمرین و بازخورد شاگرد

۲.۳ بک‌اند (Backend API)

- مدیریت احراز هویت (OTP/JWT)
- مدیریت نقش‌ها و دسترسی‌ها
- API های مربی، شاگرد، برنامه‌ها، جلسات تمرین
- API سینک برای آفلاین
- ذخیره و مدیریت توکن‌های FCM برای پوش‌نوتیف

۳ نقش‌ها و دسترسی‌ها

Athlete (شاگرد)

- فقط داده‌های خودش را می‌بیند

- می‌تواند تمرین را اجرا کند، تیک بزند، بازخورد ثبت کند
- می‌تواند نوتیفیکیشن را روشن/خاموش کند

Coach (مربی)

- می‌تواند شاگرد اضافه کند و برنامه‌ها را بسازد
- می‌تواند گزارش انجام تمرین شاگرد را ببیند
- دسترسی مربی به شاگرد وابسته به رابطه “فعال” است

Admin (فاز بعد)

- مدیریت کامل سیستم و محتوای حرکات و گزارش‌ها

(۴) مدل مالکیت داده

- شاگرد می‌تواند **چند مربی** داشته باشد.
- داده‌های شاگرد “متعلق به شاگرد” است.
- اگر رابطه مربی و شاگرد غیرفعال شود، مربی دیگر دسترسی ندارد.

(۵) منطق برنامه تمرینی

- مربی تعیین می‌کند برنامه چند هفته است (مثلاً ۴ هفته)
- مربی تعیین می‌کند هر هفته چند روز تمرین است (مثلاً ۳ روز)
- مربی فقط **روزهای تمرین** را می‌چیند (نه الزاماً ۷ روز هفته)
- شاگرد می‌تواند هر روز تمرینی را Start کند و با آن یک Session برای همان تاریخ ساخته می‌شود.

(۶) امتیازدهی تمرین (Feedback)

بعد از پایان تمرین، شاگرد فقط یک گزینه متنی انتخاب می‌کند:

- عالی بود
- خوب بود
- معمولی بود
- بد بود
- خیلی بد بود
- یک یادداشت اختیاری

۷) ویدیوها و بانک حرکات

- ویدیوها تولید اختصاصی هستند
- ویدیوها بی کلام و دارای متن/راهنما هستند
- در دیتابیس فقط URL ویدیو ذخیره می شود
- نسخه اول حدود ۱۰۰-۲۰۰ حرکت کافی است (قابل توسعه)

۸) نوتیفیکیشن ها

- Push Notification از طریق FCM
- یادآوری تمرین (روزهای برنامه دار)
- یادآوری مکمل ها (اگر فعال شود)
- SMS فقط برای Invite نصب شاگرد (در MVP کافی است)

۹) تکنولوژی های انتخابی

- Backend: **.NET ۸ Web API**
- DB: **SQL Server**
- ORM: EF Core

- Swagger: فعال با JWT Bearer
- Mobile: Android (Native Kotlin یا هر راه دیگر، ولی فعلاً تمرکز روی API)
- Web: Coach Panel (React یا هرچی، بعداً)

۱۰ MVP (نسخه اول قابل استفاده)

۷ قابلیت اصلی MVP:

1. ورود با شماره + JWT + OTP
2. دعوت شاگرد توسط مربی از پنل و ارسال SMS لینک نصب
3. ساخت پروفایل شاگرد (قد/وزن/هدف/آسیب‌ها)
4. ساخت برنامه تمرینی (چند هفته‌ای + چند روز در هفته + چینش حرکات)
5. اپ شاگرد: نمایش Today + لیست تمرین + ویدیو
6. تیک زدن حرکات و پایان تمرین با امتیاز متنی
7. ثبت Device Token و ارسال Push (حداقل زیرساخت)