

## گزارش کار پروژه نهایی طراحی پایگاه داده

بخش اول: طراحی پایگاه داده

در پوشه database تصاویر نمودارهای ER و شمای رابطه ای آورده شده است.

بخش دوم: Backend

به ازای هر رابطه یا Relation در پایگاه داده یک Entity و DAO و Route مرتبط با آن در بکند اضافه شده و با توجه به نیاز های موجود در فرانتند تابع های مطلوب را طراحی و اطلاعات مناسب را از پایگاه داده دریافت یا به آن اضافه می کنیم و یا در آن تغییر ایجاد می کنیم.

به ازای هر Route یک بلوپرینت ساخته و آن را در تابع create\_app در \_\_init\_\_ اضافه کرده و در نهایت در app این تابع را صدا می کنیم تا درخواست های مرتبط با هر روت به آن هدایت شود.

چند تابع مورد نیاز به عنوان مثال:

تابع request\_match که یک بازیکن را با توکن او شناخته و با توجه به نام کاربری

وارد شده یک درخواست بازی ارسال می کند. در ادامه بازیکن مقابل با تابع

get\_pending\_match\_requests\_route اقدام به دریافت درخواست های بازی

خود کرده و با تابع accept\_match درخواست بازی دریافت شده را رد یا تایید می کند

اگر رد کند تریگر موجود در پایگاه داده بازی فوق را از لیست بازی ها حذف می کند و اگر

تایید کند شش دور مرتبط با آن مسابقه توسط تریگر دیگری در پایگاه داده به پایگاه داده

اضافه می شود و دور اول با نفر دوم که درخواست را تایید کرده آغاز می شود و باید با

استفاده از تابع set\_round\_category\_route موضوع دور را از بین سه موضوعی

که به طور تصادفی با تابع `get_random_categories` دریافت کرده، انتخاب کند و یک تریگر دیگر برای هر دو بازیکن سه `player_answer` اضافه می‌کند و ...

### بخش سوم: Frontend

در این بخش هر صفحه طراحی شده اقدام به ارسال درخواست به بکند و با توجه به پاسخ دریافتی اقدام به تعامل با بازیکن می‌کند. به عنوان مثال `TopPlayers.js` را بررسی می‌کنیم:

ابتدا توکن بازیکن را از حافظه داخلی یا `localStorage` می‌خواند و با کمک آن به سرور درخواست [https://localhost:5000/api/players/top\\_winners](https://localhost:5000/api/players/top_winners) را ارسال کرده و منتظر پاسخ می‌ماند و پس از دریافت بازیکنان برتر آن‌ها را نمایش می‌دهد.