

مستند معماری پروژه

این مستند به صورت خلاصه معماری کلی پروژه را برای درک بهتر ساختار و روابط بین اجزا توضیح می‌دهد.

معماری پروژه: طرح‌بندی لایه‌ای

این پروژه از معماری چندلایه (Layered Architecture) پیروی می‌کند تا وظایف و مسئولیت‌ها به صورت واضح از هم تفکیک شوند.

۱. لایه ارائه

- **کامپوننت‌ها:** شامل Controller ها.
- **وظیفه:** دریافت درخواست‌های HTTP از کلاینت و ارسال پاسخ. این لایه مسئول مدیریت مسیرها، اعتبارسنجی ورودی‌ها و تبدیل آن‌ها به DTO است.

توضیحات مربوط به api ها و نحوه صدا زدن آنها در فایل [api.md](#) در ریپازیتوری قابل دسترسی است.

۲. لایه منطق

- **کامپوننت‌ها:** شامل Service ها.
- **وظیفه:** پیاده‌سازی منطق اصلی و قوانین تجاری برنامه. این لایه با استفاده از ریپازیتوری‌ها داده‌ها را از پایگاه داده می‌خواند و پس از پردازش، آن‌ها را به کنترلرها برمی‌گرداند. احراز هویت کاربر از طریق SecurityContext نیز در این لایه انجام می‌شود.

۳. لایه دسترسی به داده

- **کامپوننت‌ها:** شامل Repository ها.
- **وظیفه:** ارائه یک رابط انتزاعی برای تعامل با پایگاه داده. این لایه مسئول اجرای عملیات CRUD (ایجاد، خواندن، به‌روزرسانی و حذف) روی مدل‌های داده است. Spring Data JPA پیاده‌سازی این لایه را به صورت خودکار انجام می‌دهد.

۴. لایه مدل

- **کامپوننت‌ها:** شامل کلاس‌های Entity و DTO ها.
- **وظیفه:** Entity ها ساختار داده‌های موجود در پایگاه داده را نشان می‌دهند، در حالی که DTO ها برای انتقال داده‌ها بین لایه‌های مختلف (مثلاً بین فرانت‌اند و بک‌اند) استفاده می‌شوند.

توضیحات بیشتر در خصوص دیتا مدل ها نیز در فایل [DataModel.md](#) در ریپازیتوری فایل دسترسی است.

پیکربندی‌ها و فیلترهای امنیتی

- **SecurityConfig**: این کلاس قوانین دسترسی و فیلترهای امنیتی را تعریف می‌کند.
- **JwtAuthFilter**: این فیلتر یک جزء کلیدی در معماری امنیتی است که قبل از رسیدن درخواست به کنترلرها، توکن JWT را بررسی و اعتبار سنجی می‌کند.
- **UserDetailsServiceImpl**: این سرویس جزئیات کاربر را از پایگاه داده برای Spring Security بارگذاری می‌کند.