



معرفی یک صفحه‌ای

سامانه یکپارچه‌سازی و مدیریت واسط برنامه‌سازی

دُرسا

در سازمان‌های بزرگ معمولاً بخش‌های مختلف، به صورت جداگانه سرویس‌ها و خدمات خود را به شرکت‌های ثالث، همکاران خود در بخش‌های دیگر و یا مردم ارائه می‌کنند. همچنین ارتباط بین سامانه‌های پشتیبان کسب و کار به طور مثال، پایگاه داده‌ی اصلی سازمان به خودی خود حجم زیادی از ارتباطات و فعالیت‌های واحدهای IT سازمان را به خود اختصاص می‌دهد. نگهداری، مدیریت و بروزرسانی هر بخش روی باقی بخش‌ها موثر است. مدیریت دسترسی‌ها غالباً به علت یکپارچه و متمرکز نبودن بخش‌ها، موثر نیست و همیشه نمی‌توان تضمین کرد که سازمانی بزرگ، می‌تواند جلوی دوباره‌کاری را در زیرمجموعه‌های خودش بگیرد. این موارد سبب می‌شود نیاز به پلتفرمی که بتواند قابلیت رصد همه بخش‌ها را داشته باشد و با یکپارچه‌سازی سرویس‌ها و مدیریت همه‌ی بخش‌ها از طریق یک محل، تصمیم‌گیری را اثر بخش کند؛ بیش از پیش احساس شود. سامانه یکپارچه‌سازی و مدیریت واسط برنامه‌سازی، *درسا*، برای این تغییر در سه بخش اصلی با ظرفیت پیام‌رسانی بالا طراحی و پیاده‌سازی شده است.

۱- **API Self-Service**: در این سلف-سرویس، تمام سرویس‌های سازمان از طریق پرتالی برای کاربران بیرونی (سازمانی، شخصی) در دسترس قرار می‌گیرد. از طریق پنل مدیریتی این پرتال امکان تعریف سطح دسترسی برای هر کاربر یا گروه کاربری فراهم شده است و حتی می‌توان برای هر سرویس یا متد، جزئیات مهمی را تعیین نمود، مانند این که چه سرویسی و یا چه متدی، در دسترس چه اشخاص یا گروه‌هایی قرار بگیرد؛ فراخوانی هر یک از گروه‌های کاربری حداکثر چند TPS باشد و قابلیت‌های دیگری از این دست.

۲- **Integration Layer**: این لایه خودش چند بخش اصلی دارد، مهمترین بخش آن ESB است که برای یکپارچه‌سازی در دو سطح 3rd party و داخلی عمل می‌کند. به این معنی که یک سطح از ESBها وظیفه‌ی مدیریت ارتباطات بیرونی دارند و در سطح دیگر، ارتباطات داخلی بین مجموعه سامانه‌ها و زیرسامانه‌ها مدیریت می‌شود. بخش دوم در این لایه، وب اپلیکیشن سرورها هستند که هیچ محدودیتی برای بهره‌برداری از وب اپلیکیشن‌ها ندارند می‌توان از انواع مختلف وب اپلیکیشن‌ها استفاده کرد. به صورت پیش‌فرض *درسا* امکان میزبانی از سرویس‌های جاوایی با تمام تکنولوژی‌ها و شخصی‌سازی‌ها را دارد. بخش سوم و آخر، **Data Access Layer** می‌باشد که برای مدیریت دسترسی به دیتابیس استفاده می‌شود. برای استفاده از این DAL محدود به استفاده از هیچ نوع دیتابیس خاصی نیستیم و صرفاً وجود **jdbc driver** برای استفاده از آن کافی است. *درسا* به صورت پیش‌فرض از اوراکل استفاده می‌نماید.

۳- **Monitoring and logging**: این بخش در سطح سخت افزار وضعیت کار سرورها را مانیتور می‌کند. در سطح سرویس به مدیر سامانه این امکان را می‌دهد که رصد کند کدام سرویس یا حتی متد یا منبع، چه مقدار فراخوانی می‌شود، چه تعداد خطا داشته و مدت زمان اجرایش چقدر بوده است. حتی می‌تواند **abnormality** را هم تشخیص دهد. و از آن مهم‌تر این که این ویژگی‌ها برای تمام لایه‌های فوق‌الذکر می‌تواند خدمت رسانی کند. هر گونه تحلیل برخط از داده‌ها و رفتار سامانه با استفاده از پروتکل‌های بروز و سریع که برای داده‌های حجیم طراحی شده‌اند، در این سامانه امکان پذیر است و می‌توان نتایج آن‌را در داشبورد ساز موجود در این لایه به صورت مجسم و گرافیکی نمایش داد.

بخش دیگر این لایه، یکپارچه‌سازی و مدیریت مرکزی لاگ است. لاگ‌های ماشین‌ها و ابزارهای مختلف همگی یک جا جمع می‌شوند، اطلاعات آنها، پارس، ایندکس و دسته بندی می‌شود؛ سپس در پنل مدیریتی می‌توان لاگ‌هایی که از هر جای سیستم جمع شده را بررسی کرد و در میان آن‌ها جستجو نمود. مثلاً یک پیام خطا را جستجو می‌کنیم و متوجه می‌شویم روی چه سرور/ماشین مجازی‌هایی در کدام لایه پیش آمده، زمان رخداد کی بوده و آیا برطرف شده است یا خیر. از قابلیت‌های داشبوردساز در این بخش هم می‌توان استفاده کرد، مثلاً می‌توانیم بگوییم نموداری از تعداد لاگ‌های سطح **ERROR** یا سطح **FATAL** در لایه‌ی **integration** داشته باشیم که مثلاً در هر لحظه چند عدد از این لاگ‌ها ساخته شده.