

تشریح جزییات پروژه #زاگرس یا تغییر معماری دیدگاه

تغییر معماری مجموعه دیدگاه و آغاز مهاجرت نرم افزارهای مختلف به این معماری، خبر بزرگی بود که بسیاری از مدعوین هفتمین گردهمایی مشتریان چارگون را هیجان زده کرد.

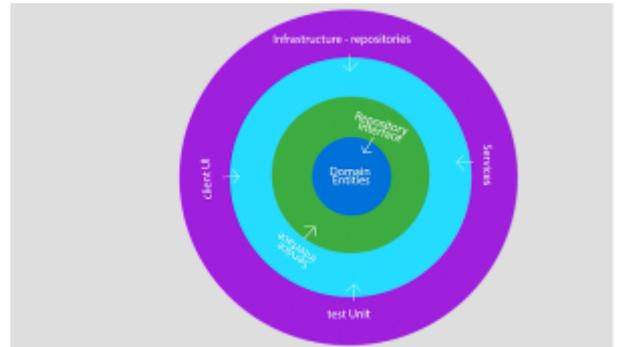
هر چند این تغییرات از سال گذشته و با انتقال نرم افزار مکاتبات به معماری جدید کلید خورده بود؛ اما آگاهی از جزییات فنی و قابلیت های آن برای همه سازمان های مشتری، شنیدنی تر از هر سخنی بود. این مقاله تالیفی ست از توضیحات تیم توسعه زیرساخت دیدگاه درباره جزییات پروژه زاگرس که برای مهمانان هفتمین گردهمایی سالانه مشتریان چارگون ارائه شد:

چرا تغییر معماری؟

وقتی می خواهیم یک بنا را بسازیم ابتدا به اسکلت بندی آن فکر میکنیم. اسکلت اصلی سازه باید محکم و با استقامت باشد چون قرار است کل سازه را نگه دارد. اگر نرم افزار را تمثیلی از سازه در نظر بگیریم، معماری نرم افزار معادل اسکلت سازه است. به این ترتیب وقتی صحبت از تغییر معماری نرم افزار می شود در واقع با پروژه ای به اندازه بازنویسی کامل آن روبه رو هستیم؛ اما پرسش اصلی این است که چرا چارگون به تغییر معماری رضایت داد؟

از مهمترین عواملی که موجب شد تا چارگون چنین تصمیم بزرگی را بگیرد، پایین آوردن هزینه تغییرات و کم کردن زمان Deliver کردن محصول به مشتری است. از اهداف دیگر در این تحول بزرگ می‌توان به امکان بروزرسانی نرم‌افزارهای مختلف به شکل مستقل از یکدیگر اشاره کرد که در معماری پیشین دیدگاه، ممکن نبود. معماری جدید، مدلی مشابه به معماری پیازی (onion architecture) است که قسمت‌های مختلف نرم‌افزار در آن، حول یک هسته متمرکز قرار می‌گیرند. در این معماری، اجزای مختلف تنها هسته مرکزی را می‌بینند و از آن شناخت دارند و از وجود قسمت‌های دیگر سیستم آگاه نیستند. این عدم آگاهی باعث می‌شود تا رشد و بلوغ آن بجز از سیستم به اجزای دیگر وابسته نباشد. به این شکل هر جزء می‌تواند به طور مستقل Develop شود، از تکنولوژی دلخواه استفاده کند، به طور مستقل تست شود و تغییرات آن روی قسمت‌های دیگر سیستم تاثیر نگذارد.

انتظار شرکت چارگون از معماری جدید دیدگاه، نرم‌افزارهایی سریع‌تر و کم‌باگ‌تر بود که تغییرات مورد نیاز مشتری با سرعت بیشتری در آن پیاده‌سازی شوند و زودتر به دست او برسند. برای این امر، تحقق 3 هدف تولید نرم‌افزارهای چابک با سرعت انتقال بالا، قابلیت تست‌پذیری و امکان توسعه‌پذیری به صورت جداگانه در دستور کار قرار گرفت.



انتخاب یک مکانیزم مطمئن ارتباطی

در یک سیستم نرم‌افزاری مبتنی بر مدیریت داده‌های کاربران، ارتباط با یک یا چند پایگاه داده بزرگ از مهم‌ترین دغدغه‌های معماری سیستم است. برای انتخاب یک مکانیزم مطمئن ارتباطی بین پایگاه داده و نرم‌افزار ملاحظات متعددی وجود داشت که از آن جمله می‌توان به حجم بسیار بالای تراکنش‌های بین سرور وب و پایگاه داده، حجم شدن تراکنش، ارتباط توزیع‌شده تراکنش‌ها در پایگاه داده‌های مختلف، پاسخ به نیازهای کسب و کار پیچیده سیستم‌های ۴ حوزه دیدگاه به ویژه در حوزه فرآیندها و محاسبات مالی اشاره کرد. مجموع این ملاحظات، دقت و حساسیت بالایی را طلب می‌کرد.

تحقیقات نشان داد تیم توسعه زیرساخت دیدگاه 5 برای تعیین مکانیزم مطمئن ارتباطی بین پایگاه داده و نرم‌افزارها 2 انتخاب پیش رو دارد:

- 1- انتخاب یک ORM پیشرفته با قابلیت‌های فراوان که کدنویسی و نگهداری کد را بسیار کم‌هزینه می‌کند؛ اما کاهش سرعت پردازش را نیز به همراه دارد.
- 2- انتخاب و توسعه یک Micro ORM با قابلیت‌های پایه که با وجود افزایش هزینه کدنویسی و نگهداری، سرعت و بهینگی نرم‌افزار را به طور منحصر به فردی بالا می‌برد.

در نهایت با بررسی تفاوت‌های دو گزینه، انتخاب و توسعه یک Micro ORM از سوی شرکت چارگون برگزیده شد تا با حداکثر سرعت و بهینگی داده‌ها را به مشتریان چارگون ارائه کنیم.

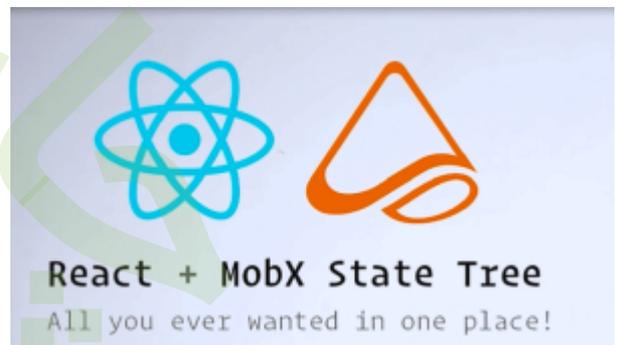


تمرکز بر کاهش هزینه نگهداری و افزایش توسعه‌پذیری نرم‌افزارها در طول یکسال گذشته با هدف پاسخگویی به نیاز رو به رشد نرم‌افزارهای جدید دیدگاه، کار بر روی نسل جدید کنترل‌ها کلیدخورد. به این منظور، مشکلات موجود بررسی و تلاش شد تا در نسل جدید، تمرکز تیم بر روی ایجاد فرم‌هایی باشد که سریعتر رندر شوند، ایمن باشند، هزینه نگهداشت کمتر و توسعه‌پذیری بالاتری داشته باشند. تیم توسعه زیرساخت دیدگاه 5 بر روی تکنولوژی‌های روز دنیا بررسی‌های فراوانی داشت و در نهایت تصمیم گرفت که MVVM Pattern موجود که با استفاده از Knockout پیاده‌سازی شده بود را کنار بگذارد و با استفاده از React کنترل‌های جدید را توسعه دهد.

معماری جدید، نه تنها نیاز به تغییر در Stack تولید داشت؛ بلکه نیازمند تغییر در تفکر طراحی بود. کنترل‌های React به دلیل ذات Component Base خود، قابلیت

توسعه‌پذیری بالایی دارند به همین دلیل جداسازی Business از Presentation در سطح کنترل‌های React راه را برای تست‌پذیری نسل جدید هموار می‌کنند. دیدگاه 5 با توجه به قابلیت نمایش هم‌زمان فرم‌های مختلف از نرم‌افزارهای دیدگاه نیاز به زیرساخت قدرتمندی داشت که توانایی پاسخگویی این حجم از اطلاعات را داشته باشد و کامپوننت‌های جدید با استفاده از تکنولوژی Virtual DOM در این زمینه به صورت بسیار کارآمدتری عمل کنند.

مجموع این نیازها موجب شد تا در نهایت با بهره‌گیری از ابزارهای تولید جدیدی مثل NodeJS رابط کاربری دیدگاه 5 توسعه داده شود تا مشتریان دیدگاه از مزیت‌های تکنولوژی روز دنیا بهره‌مند شوند.



رابط کاربری ساده و زیبا

معروف است که زیبایی در سادگیست؛ چراکه سادگی از مُد نمی‌افتد. مطالعه‌ای بر روی رفتار کاربرها نشان می‌دهد هرچا رابط کاربری ساده‌تر و تجربه کاربرانش به تجربه‌های دنیای واقعی نزدیک بوده، کاربر راحت‌تر توانسته است با نرم‌افزار ارتباط برقرار کند و از کارکردن با آن راضی‌تر بوده است.

استانداردی ساده که زیبا و در عین حال، کار کردن با آن راحت باشد و همچنین ارائه پلتفرم‌هایی متفاوت که تجربه یکپارچه‌ای را نمایش دهد و کار کردن طولانی‌مدت با آن، کاربر را خسته نکند از ویژگی‌هایی بود که تیم توسعه زیرساخت دیدگاه 5 به دنبال آن بود. در نهایت این تیم از میان استانداردهای متنوع برای پیاده‌سازی رابط کاربری، Material Design شرکت گوگل را انتخاب کرد که تمام جوانب مورد نظر را یکجا در خود جای داده است.

نقاط قوت material design گوگل دستمایه طراحی استاندارد جدیدی برای رابط کاربری دیدگاه شد که هم نیازهای پیچیده نرم‌افزارها را پوشش می‌دهد و هم کار کردن با آن برای کاربر خوشایند و راحت است.

