

عنوان مقاله: پیاده سازی API Gateway؛ راهکار حل مشکلات

نویسنده مقاله: تیم فنی نیک‌آموز

تاریخ انتشار: مرداد ماه ۱۴۰۱

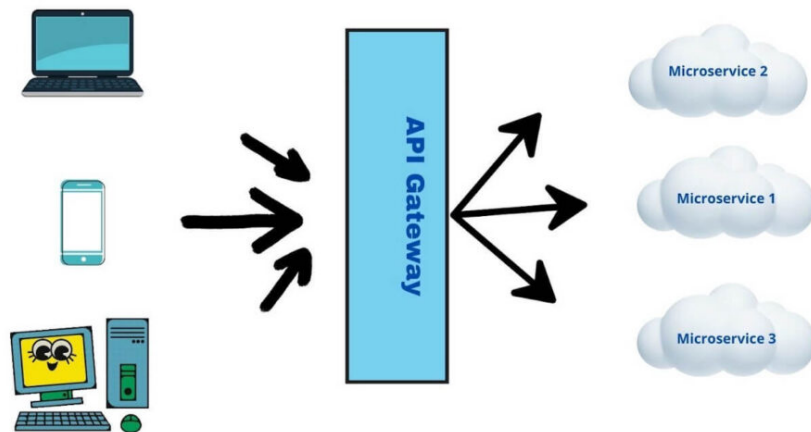
منبع : <https://nikamooz.com/api-gateway-implementation/>

زمانی که تصمیم می‌گیرید برنامه خودتان را به‌عنوان مجموعه‌ای از میکروسرویس‌ها بسازید، باید تصمیم بگیرید که مشتریان شما چگونه با این میکروسرویس‌ها تعامل خواهند داشت. در معماری میکروسرویس‌ها، هر میکروسرویس دارای مجموعه‌ای از نقاط پایانی است که آن‌ها را در معرض نمایش قرار می‌دهد. این امر می‌تواند بر روی ارتباط مشتریان با نرم‌افزار تأثیرگذار باشد. شما می‌توانید به کمک پیاده سازی api gateway بر چنین مشکلاتی غلبه کرده و به یک برنامه یکپارچه و با رابط کاربری مناسب دست پیدا کنید.

الگوی API Gateway چیست؟

API که مخفف Application Program Interface است در واقع مجموعه از دستورات عمل‌ها، پروتکل‌ها و ابزارهایی برای ساخت نرم‌افزار بوده که نحوه تعامل اجزای نرم‌افزار را مشخص خواهد کرد. API Gateway یک سرور است که به‌عنوان یک نقطه ورود واحد به سیستم شناخته می‌شود. api gateway می‌تواند بر روی معماری داخلی سیستم تأثیرگذار بوده و برای هر مشتری یک API ارائه دهد.

همچنین پیاده سازی api gateway می‌تواند مسئولیت‌های دیگری مانند احراز هویت، نظارت، متعادل‌سازی بار، ذخیره‌سازی، شکل‌دهی، مدیریت درخواست و پاسخ راه هم به عهده داشته باشد. api gateway از سوی دیگر مسئول مسیریابی درخواست، ترکیب و ترجمه پروتکل بوده و باید بداند که تمام درخواست‌های ارسال شده توسط مشتری از طریق دروازه API انجام می‌شود.



بعد از این مرحله است که api gateway می تواند درخواست های ارائه شده را به میکروسرویس مناسب هدایت کند. همان طور که گفتیم api gateway می تواند موجب شود تا درخواست های مناسب از طریق آن به میکروسرویس ها هدایت شوند. باید بدانید که این امر به کمک یکی از دو روش زیر امکان پذیر است:

- درخواست ها به سرویس مناسب هدایت و یا پروکسی می شوند.
- درخواست ها به چندین سرویس هدایت و یا فرستاده خواهند شد.

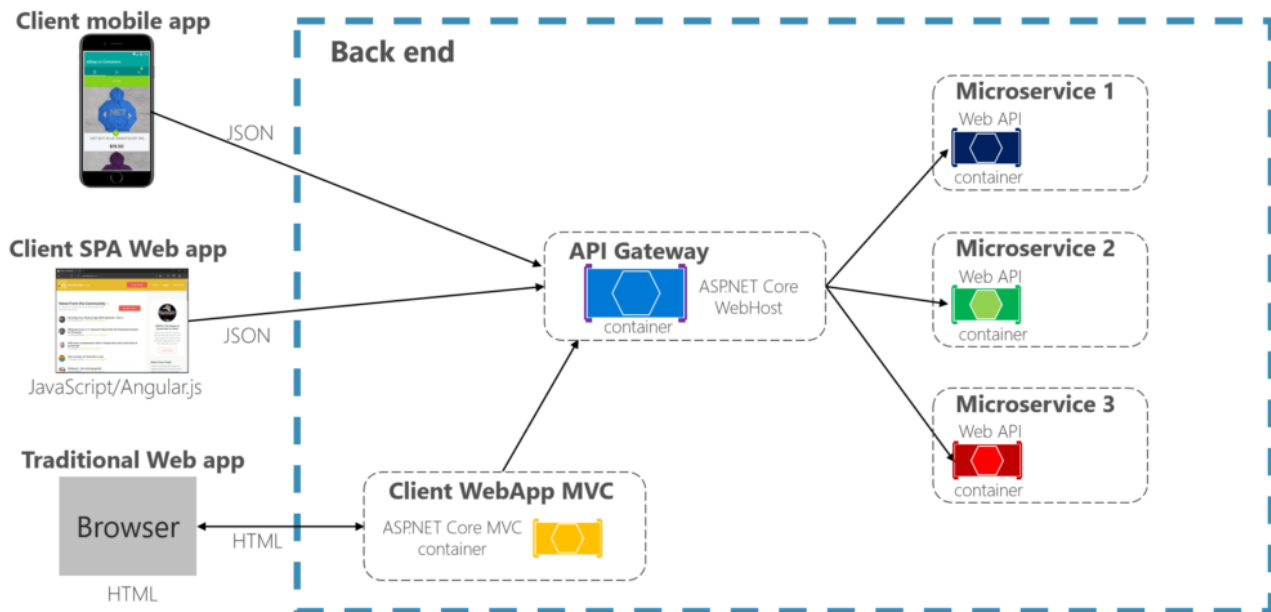
مزایا و معایب پیاده سازی API Gateway

همان طور که انتظار می رود، استفاده از الگوی api gateway می تواند مزایا و معایب خاص خودش را داشته باشد. مهم ترین مزیت استفاده از دروازه API این است که ساختار داخلی برنامه محصور خواهد شد. از این رو مشتریان به جای درخواست از خدمات خاص، می توانند به سادگی با این دروازه تعامل داشته باشند. از سوی دیگر api gateway برای هر نوع کلاین یک API خاص ارائه خواهد کرد که این کار باعث کاهش تعداد عملیات رفت و برگشت بین برنامه و مشتری خواهد شد.

از دیگر مزایای پیاده سازی api gateway می توان به ساده سازی کدهای برنامه نویسی، افزایش کارآمدی برنامه، کاهش خطاها و ... اشاره کرد. البته که api gateway دارای اشکالاتی نیز است api gateway. در واقع یک مؤلفه بسیار در دسترس بوده که نیاز به توسعه، مستقر شدن و مدیریت دارد. از سوی دیگر این خطر وجود دارد که دروازه API به یک گلوگاه توسعه نیز بدل شود. توسعه دهندگان باید api gateway را به روزرسانی کرده تا بتوانند نقاط پایانی هر میکروسرویس را مشخص کنند.

از این رو رو بسیار مهم است که روند به روزرسانی api gateway تا حد امکان ساده و سبک انجام شود. در غیر این صورت، توسعه دهندگان برای به روزرسانی و آپدیت دروازه API مجبور خواهند شد تا در صف منتظر بمانند. اما با وجود چنین مشکلات و معایبی برای استفاده از دروازه های API، این الگو می تواند به عنوان راهکار حل مشکلات برنامه های کاربردی معرفی شود.

Using a single custom API Gateway service



نکات مهم پیاده‌سازی API Gateway

حال که به صورت کامل با api gateway آشنا شدید و متوجه شدید که استفاده از آن دارای چه مزایا و معایبی است، زمان آن رسیده است تا به مسائل طراحی مختلفی که باید در زمان پیاده سازی این الگو مدنظر داشته باشید، اشاره کنیم:

عملکرد و مقیاس‌پذیری:

باید بدانید که امروزه تعداد انگشت شماری شرکت‌ها در مقیاس‌های عظیم و گسترده فعالیت می‌کنند که نیاز به رسیدگی کردن به میلیاردها درخواست در روز را دارند. با این حال برای اکثر برنامه‌های کاربردی موجود، عملکرد و مقیاس‌پذیری api gateway بسیار مهم است. کاملاً طبیعی خواهد بود که api gateway را بر روی پلتفرمی پیاده سازی کنیم که ورودی/خروجی ناهمزمان و غیر مسدودکننده را پشتیبانی کند.

امروزه انواع مختلفی از فناوری وجود دارد که می‌توان از آن‌ها برای پیاده سازی api gateway مقیاس‌پذیر استفاده کرد که از مهم‌ترین آن‌ها می‌توان به موارد زیر اشاره کرد:

- در JVM می‌توانید از یکی از چارچوب‌های مبتنی بر NIO مانند Netty، Vertx، Spring Reactor یا JBoss Undertow استفاده کنید.
- NGINX Plus

- جاوا اسکریپت کروم

استفاده از مدل برنامه نویسی واکنشی:

api gateway برخی از درخواستها را به سادگی با مسیریابی آنها به سرویس Backend مناسب، رسیدگی می کند. همچنین می تواند با فراخوانی چندین سرویس پشتیبان و جمع آوری نتایج، سایر درخواستها را نیز مدیریت کند. اما برخی از درخواستها کاملاً از یکدیگر مستقل بوده و برای به حداقل رساندن زمان پاسخ گویی، الگوی api gateway باید درخواستهای مستقل را همزمان انجام دهد.

در صورتی که کد api gateway را با استفاده از روش واکنشی به سبک اعلانی بنویسید، می توانید سرعت پاسخ گویی به درخواستها را تا حد بسیار زیادی افزایش دهید. استفاده از مدل برنامه نویسی واکنشی شما را قادر می سازد تا کد api gateway بسیار ساده و درعین حال کارآمد بنویسید.

فراخوان خدمات:

یک برنامه کاربردی مبتنی بر میکروسرویس یک سیستم توزیع شده بوده که باید از مکانیزم ارتباط بین فرایندی استفاده کند. دو سبک ارتباط بین فرایندی عبارتند از: استفاده از مکانیزم ناهمزمان مبتنی بر پیام، استفاده از یک مکانیزم همزمان. یک سیستم معمولاً از هر دو سبک ارتباط فرایندی همزمان و ناهمزمان استفاده می کند و حتی می تواند چندین سبک پیاده سازی را مورد استفاده قرار دهد. برای پیاده سازی api gateway باید توجه داشت که این دروازه از مکانیزمهای ارتباطی مختلفی پشتیبانی کند.

کشف خدمات:

api gateway باید مکان (آدرس IP و پورت) هر میکروسرویس که با آن در ارتباط است را بداند. در برنامه های میکروسرویس مدرن مبتنی بر ابر (cloud-based)، نیز باید به این نکته توجه کرد. با اینکه سرویس های زیرساخت دارای یک مکان ثابت هستند، اما تعیین مکانیک سرویس چندان کار ساده ای نیست. سرویس های برنامه دارای مکان هایی به صورت پویا بوده که می تواند تغییر پیدا کنند. در نتیجه api gateway مانند هر سرویس گیرنده دیگری در سیستم باید از مکانیزم کشف سرویس سیستم استفاده کند.

رسیدگی به شکست های جزئی:

یکی دیگر از مسائلی که باید هنگام پیاده سازی api gateway به آن بپردازید، مشکل شکست جزئی است. این موضوع در همه سرویس های توزیع شده، زمانی که یک سرویس با سرویس دیگری تماس می گیرد که یا کند پاسخ می دهد و یا در دسترس نیست، ایجاد می شود. api gateway هرگز نباید به طور نامحدود منتظر یک سرویس پایین دستی قرار بگیرد و در این زمانها، باید سایر درخواستها را انجام دهد.

برای برطرف کردن این مشکل api gateway می‌تواند داده‌های کش شده را در صورت موجود بودن، بازگرداند. با بازگرداندن داده‌های پیش‌فرض یا داده‌های موجود در حافظه پنهان، پیاده سازی api gateway تضمین می‌کند که خرابی‌های سیستم بر تجربه کاربر تأثیرگذار نخواهد بود.

نتیجه‌گیری:

برای اکثر برنامه‌های مبتنی بر میکروسرویس، پیاده سازی api gateway که به‌عنوان یک دروازه ورودی واحد به سیستم عمل می‌کند، منطقی به نظر می‌رسد. api gateway. مسئول مسیریابی درخواست، ترکیب و ترجمه پروتکل است. این سرویس می‌تواند به هر یک از مشتریان یک API سفارشی ارائه داده و با بازگرداندن داده‌های ذخیره شده یا پیش‌فرض، مشکلات موجود در سرویس‌های پشتیبان را پنهان کند. استفاده از الگوی api gateway دارای مزایا و معایب خاص خود بوده ولی همچنان به‌عنوان راه‌حل بسیاری از مشکلات شناخته می‌شود.