

В современной микросервисной архитектуре количество отдельных сервисов может быстро увеличиваться. В нашем проекте, состоящем из BookService, AuthorService, OrderService, UserService и NotificationService, каждая служба отвечает за свою бизнес-логику. Однако клиенту не всегда удобно напрямую обращаться к каждому сервису. Для решения этой задачи используется API Gateway.

Роль API Gateway заключается в том, чтобы выступать единым входом для всех внешних запросов к микросервисам. Вместо того чтобы клиент напрямую вызывал нужные ему сервисы, он обращается к одному URL. Gateway анализирует запрос и маршрутизирует его к нужному сервису на основе правил маршрутизации.

Например:

1. /books/** - для взаимодействия с BookService
2. /users/** - для взаимодействия с UserService

И так далее.

Spring Cloud Gateway предоставляет фильтры для обработки запросов и ответов, что позволяет решать задачи аутентификации и авторизации. Например, можно настроить фильтр, который проверяет JWT-токен в заголовке запроса, прежде чем пропустить его к микросервису. Это означает, что каждый микросервис не будет самостоятельно проверять токены - ответственность централизована в Gateway.

API Gateway может выполнять дополнительные функции:

1. Логирование и мониторинг запросов.
2. Балансировку нагрузки между несколькими инстансами одного микросервиса.
3. Кэширование.