

Нефункциональные требования (НФТ)

НФТ обычно относятся к атрибутам качества системы, таким как производительность, надежность, безопасность, удобство использования и масштабируемость. Они не связаны напрямую с конкретными функциями системы, но определяют, как эти функции должны быть реализованы и как они должны работать вместе, чтобы обеспечить желаемый уровень сервиса.

НФТ играют важную роль в проектировании архитектуры, так как они влияют на выбор технологий, структуру системы, и какие меры безопасности и оптимизации необходимы для обеспечения целостности и надежности системы. Например, если система должна обрабатывать большие объемы данных в режиме реального времени, это может потребовать выбора определенных технологий и подходов к архитектуре, чтобы обеспечить требуемую производительность. Если НФТ игнорировать, система может оказаться неспособной удовлетворять потребности пользователей или бизнеса, что может привести к ухудшению удовлетворенности пользователей и ущербу для бизнеса.

В основном выделяют разные группы нефункциональных требований, в которые в том числе входит и требования к дизайну (например, в приложении должна поддерживаться темная тема). Но мы с вами поговорим о самых важных для архитектора - нефункциональных требованиях к качеству ПО (атрибуты качества).

Всего атрибутов качества очень много (https://en.wikipedia.org/wiki/List_of_system_quality_attributes). Некоторые из них будут уникальны для вашего продукта. Рассмотрим основные атрибуты, на которые обычно опираются архитекторы.

- Производительность(Performance): Каково будет время загрузки приложения для пользователей? Как мы можем справиться с сетевой задержкой?
- Отказоустойчивость (Fault tolerance): Как мы можем обеспечить стабильную работу приложения? Как мы можем сделать приложение устойчивым к отказам?
- Доступность(Availability): Как мы можем обеспечить высокую доступность приложения?
- Масштабируемость(Scalability): Как мы можем удовлетворить растущий спрос на ресурсы? Как мы можем обеспечить хорошую масштабируемость при внезапном увеличении нагрузки?
- Безопасность (Security): Как мы можем защитить приложение от несанкционированного доступа? Как мы можем защитить приложение от злонамеренных атак? Как мы можем соответствовать местным законам и требованиям аудита?

Однако, в зависимости от характера проекта, могут быть НФТ, которые подходят только для этого проекта (например, четкость голоса для решения для колл-центра или любые другие).

Архитектор решений привлекается к проекту на очень ранней стадии, что означает, что ему нужно проектировать решение, оценивая все требования и ограничения к проекту. Нефункциональные требования позволяют уменьшить степень неопределённости при проектировании решения - потому что на основе них понятно, какие технологии точно не надо использовать (например, для сильной строгой асинхронности REST не подходит, а брокеры Kafka - вполне.)

Нефункциональные требования являются неотъемлемым и важным аспектом проектирования решения, который часто упускается из виду, когда команды слишком сосредоточены на бизнес-требованиях, и это может плохо сказаться на пользовательском опыте. Хороший архитектор решений несет основную ответственность за объяснение важности НФТ и убеждается, что они реализованы как часть доставки решения.

Сложность заключается в том, что мало кто от бизнеса знает, какие точные цифры по НФТ нужно учесть. Скорей всего заказчики вам будут говорить цифры вроде“нужна 100% доступность приложения!”. Но такого не бывает в реальности, системы и сеть (интернет) ненадёжны. Поэтому архитектор должен в том числе отталкиваться от своего опыта и бизнес-требований, чтобы выделить метрики для НФТ (тут уже можно самостоятельно предлагать требования и метрики для измерения их достижения).