## **Задание 13.1**

**Для микросервисной архитектуры системы пиццерии, которую вы спроектировали в предыдущем модуле, продумайте:**

1. **Возможность асинхронного и синхронного взаимодействия:** опишите какие из запросов в рамках реализации сценариев должны выполняться синхронно, а какие могут асинхронно. Дайте как минимум по два примера каждого типа.
2. **Механику масштабирования —** подумайте, какие сервисы могут быть максимально нагружены и как их масштабировать, опишите механизм балансировки.
3. **Механизм кеширования данных в вашей архитектуре:** где будут располагаться кеши, какие данные кешировать, как будет происходить процедура сброса кеша.

**1.1 Асинхронное взаимодействие 1 -** отправка клиентских данных в клиентскую базу после прохождения регистрации/авторизации. В случае с регистрацией создается новая запись со штампом первого входа. В случае с авторизаций - запись с новым штампом входа и, возможно, другими деталями для работы аналитиков (потом... - не знаю, зачем написал это:)

**1.2 Асинхронное взаимодействие 2 -** уведомление пользователя (через СМС, например, об успешно произведенной оплате или снятии денег со счета клиента банком после процедуры оплаты на сайте (оплата товара).

**1.3 Асинхронное взаимодействие 3 -** возможно, передача заказа на кухню в одностороннем автоматизированном режиме после оплаты товара, а также в подобном ключе - уведомление курьера о готовности заказа для доставки от кухни. Но я бы в таком случае настроил возможность подтверждение взятия заказа курьером хотя бы - из контрольных/аналитических побуждений.

**1.4 Синхронное взаимодействие 1 -** запрос на авторизацию/регистрацию должен сопровождаться вводом данных пользователем для возможности дальше пользоваться системой полноценно.

**1.5 Синхронное взаимодействие 2 -** запрос на подтверждение правильности выбора заказа и последующее подтверждение пользователем правильности выбора товара

**1.6 Синхронное взаимодействие 3 -** запрос на ввод данных платежной карты и последующий ввод данных пользователем для произведения оплаты заказа.

**2.1** Мне кажется, что модули **каталог** и **оплата заказа** могут быть наиболее перегружены, но каталог больше, чем оплата, ибо в каталог попадают и производят действия все, а от оплаты пользователь может и отказаться в какой-то момент.

С другой стороны модуль “оплата” по моей схеме имеет в разы больше взаимодействия, чем “каталог” и… исходя из количества интеракций он выглядит более загруженным. Как это все сбалансировать - понятия не имею. Возможно, создать с десяток сервис-модулей под “оплату” и два-три под “каталог”. Или попытаться на стадии тестировки вывести общий коэффициент по результатам взаимодействий и времени, и по этим результатам вывести множитель для количества необходимых сервис-модулей под отдельные СОА решения.

Если же исходить из временных рамок - то для нового клиента и для возвращающегося клиента время, проведенное в каталоге будут разными, но все же более ресурсо-затратными, нежели модули оплаты или оформления доставки. К тому же, в каталоге есть возможность повесить рекламу, а это будет тоже оттягивать какой-то ресурс системы.

Оплату можно “разгрузить” через подключение модуля Джи-Пэй, например, Но я это так, наугад, фактически. Ибо такие решения не требуют ввода данных, а лишь подтверждение карты через биометрию, глаз и прочие опции верификации…

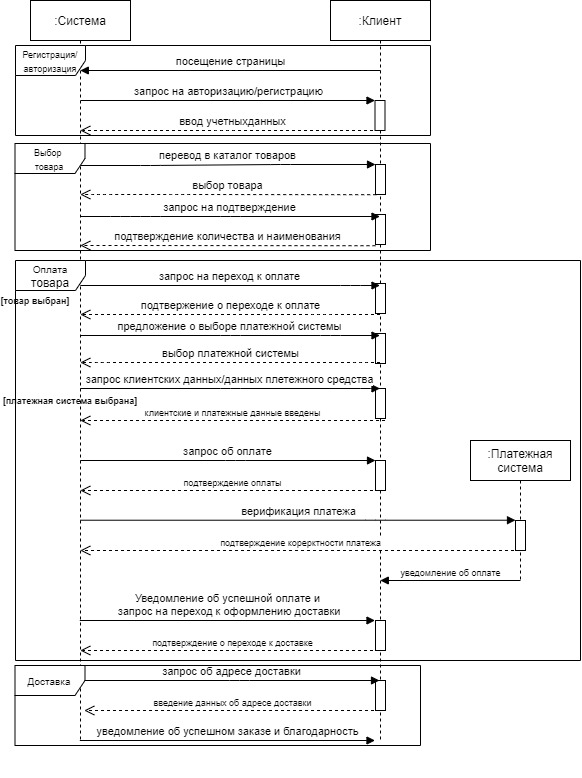
Механизм балансировки можно было бы подключить, в частности для оплаты, к факту выбора платежной системы пользователем и статуса нового/повторного пользователя. Исходя из варианта, то есть “по весу” - новый пользователь и ручной ввод карты более “ресурсны”, чем повторный пользователь или оплата через сохраненную карту.

Модуль заказа должен бы работать по циклу, ФИФО так сказать, ставя заказы в очередь для кухни.

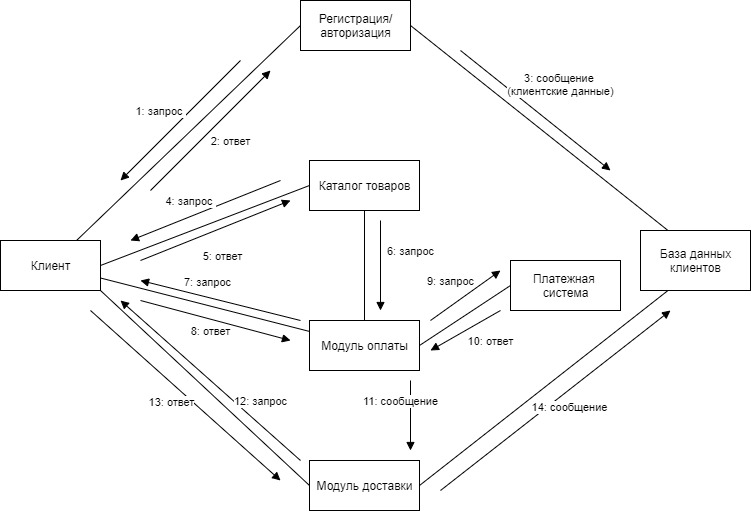
3.1 По кэшированию - скорее всего, это предметная ситуация для каталога товаров - картинки, состав, цены. Обновляться должно по сути обновления прейскуранта, наличия ингредиентов (подача с кухни), акций.

Возможно было бы неплохо в каталоге сделать рубрику “наиболее популярный товар” и кешировать его раз в 2-4-8 часов, таким образом наглядно давая пользователю возможную подсказку и разгружая систему. Подключив аналитику к переходам по этой рубрике можно было бы понять, работает ли задумка, есть ли смысл сделать в такой рубрике рейтинг популярность топ-3 продуктов, тем самым сохраняя кэш и управляя мнение/интересом пользователя через рейтинг. Или я заблуждаюсь сильно?

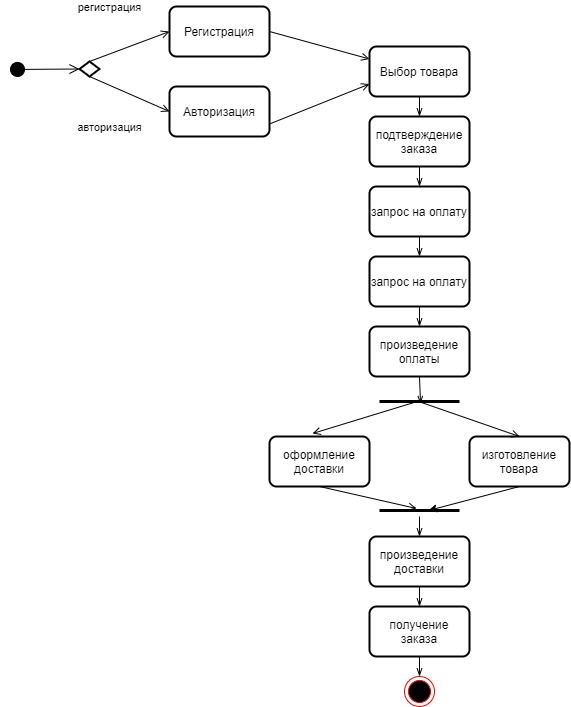
В любом случае актуальность кэша должна как минимум быть привязана к старту “нового рабочего дня”, когда уже есть готовый прейскурант и подбито наличие товара и его составляющих, актуальная информация обновлена на сайте.

****

* **смоделируйте основной поток этого же сценария с помощью диаграммы коммуникаций;**

****

* **смоделируйте этот же сценарий при помощи диаграммы активности.**

****