Системный дизайн агрегатора такси(Яндекс-Такси/Uber...)

# Требования и граничные условия

# Оценки (расчеты)

Трафик

• Хранилище

• Память

• Пропускная способность

• …

• Возможные доп вопросы:

• Количество активных пользователей в день

• Количество новых пользователей в месяц

# API дизайн системы (модель данных)

Разработка API интерфейса (минимум концептуально)

• Модель данных

• Отдаем

• Получаем

• Потоки данных

• С какими компонентами / системами связаны и как?

• На что можем повлиять?

# Дизайн БД

Один из важнейших пунктов

• Эффективность решения во многом зависит от хранения

• Что нужно: БД NoSQL или SQL?

• Схема данных

# High-level design (потоки данных между компонентами системы)

Состав компонентов

• Как взаимодействуют друг с другом

• \* часто описывается в виде диаграмм

# Детализированный дизайн

• Итерационные проходы

• Как правило, детализируют лишь определенные компоненты

# Узкие места

Есть ли в вашей системе единая точка отказа?

• Что будет, если мы потеряем несколько серверов?

• Достаточное ли количество процессов (инстансов) сервисов у нас

запущено на каждом из серверов?

• Будет ли система работать также стабильно, если нагрузка

увеличится? А если она увеличится резко?

# Мониторинг