

# Spec driven development in Microservices

Nikita Galkin

softserve

**DATAART**

 **SIGMA**  
Software

GlobalLogic

infopulse

upwork™

 **Svitla**  
systems

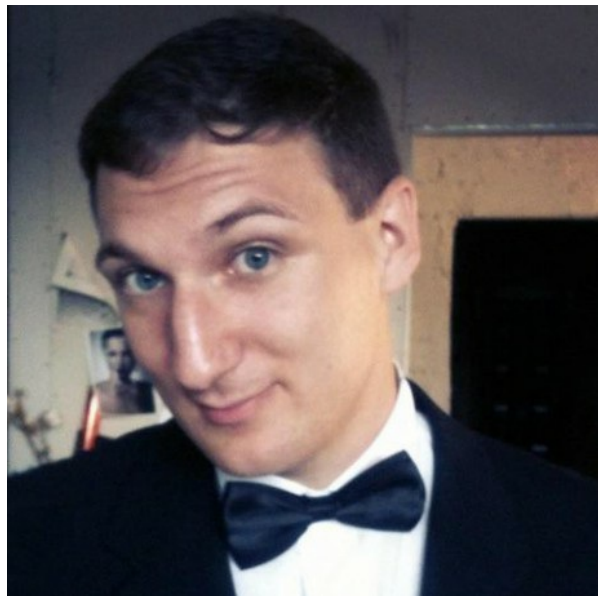
IT SERVICES  
**Levi**  
9

 **LOHIKA**

 **PIRAEUS  
BANK**  
Πιραιεύς Банк  
Надійний з 1916 року

stfalcon.com 

  
frameworks  
days



Никита  
Галкин

# Backend разработчик

Люблю Highload, Opensource и Хакатоны

Умею делить монолиты на микросервисы  
на PHP и Node.js

в компания  Ciklum (ThomasCook)

---

 galk-in  galkin  galk\_in  nikitagalkin

В чем проблема?

Бизнес решает любую проблему новым  
процессом

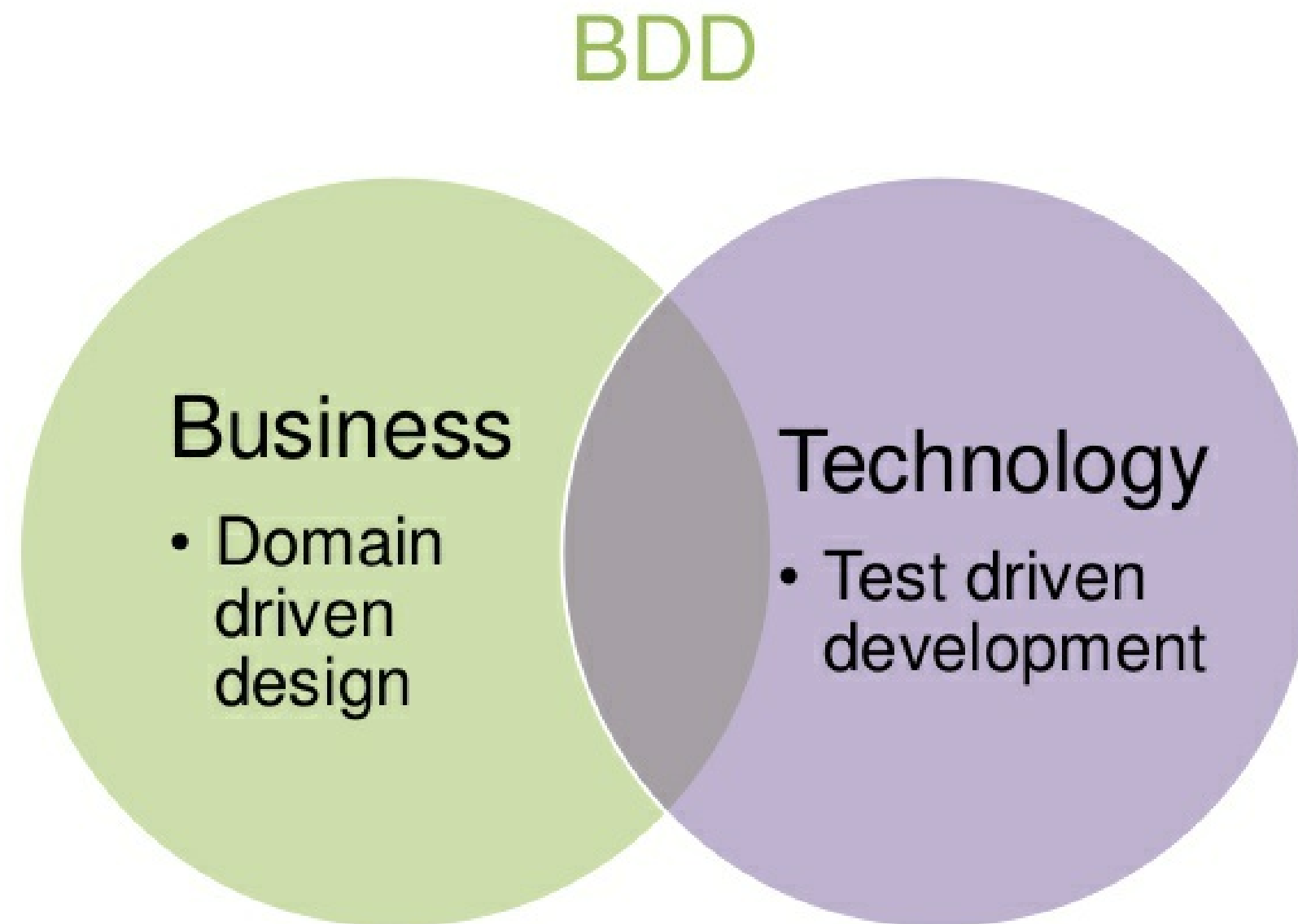
В чем  
проблема?

Бизнес решает любую проблему новым  
процессом:

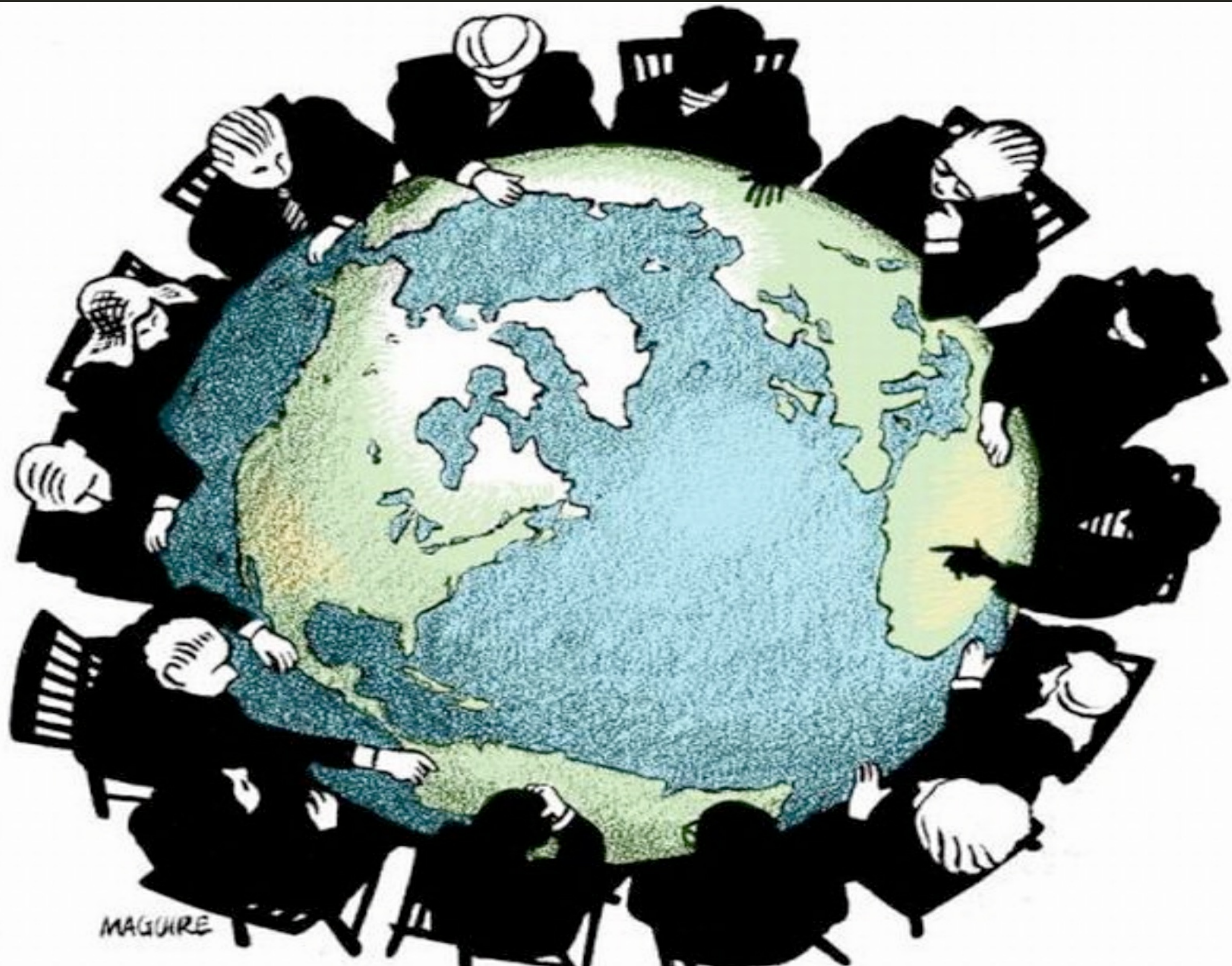
- Плохо => Будем проверять => **TDD**
- Не то => Делаем вместе => **Scrum**
- Долго => Автоматизируем => **CI/CD**
- Дорого => Наймем других => **Outsourcing**
- Не понятно => Говорите понятно => **DDD**

В чем  
проблема?

И так далее







В чем  
проблема?

Организовывать

совместную работу

разработчиков **эффективно**



Что такое спецификация?

Термины

Спецификация

это контракт

между разработчиками.

Термины

Спецификация

не описывает

бизнес логику.

Термины

Спецификация

фиксирует договоренности,

а не описывает код.



- QA информирует о баге
- Ответственных нет, ни ВЕ, ни FE не могут воспроизвести
- Фронтэнд считает, что виноват бэкенд
- Бэкенд считает, что виноват фронтэнд
- "И вообще у нас своих багов хватает" (с)





Различайте  
баги в логике и  
баги в реализации  
спецификации

# Составные части спецификации

1. Структура данных
2. Методы работы с данными
3. Информирование об ошибках

# На примере REST:

1. Структура данных

**JSON schema**

2. Методы работы с данными

**endpoints**

3. Информирование об ошибках

**Response codes**

# Требования к спецификации



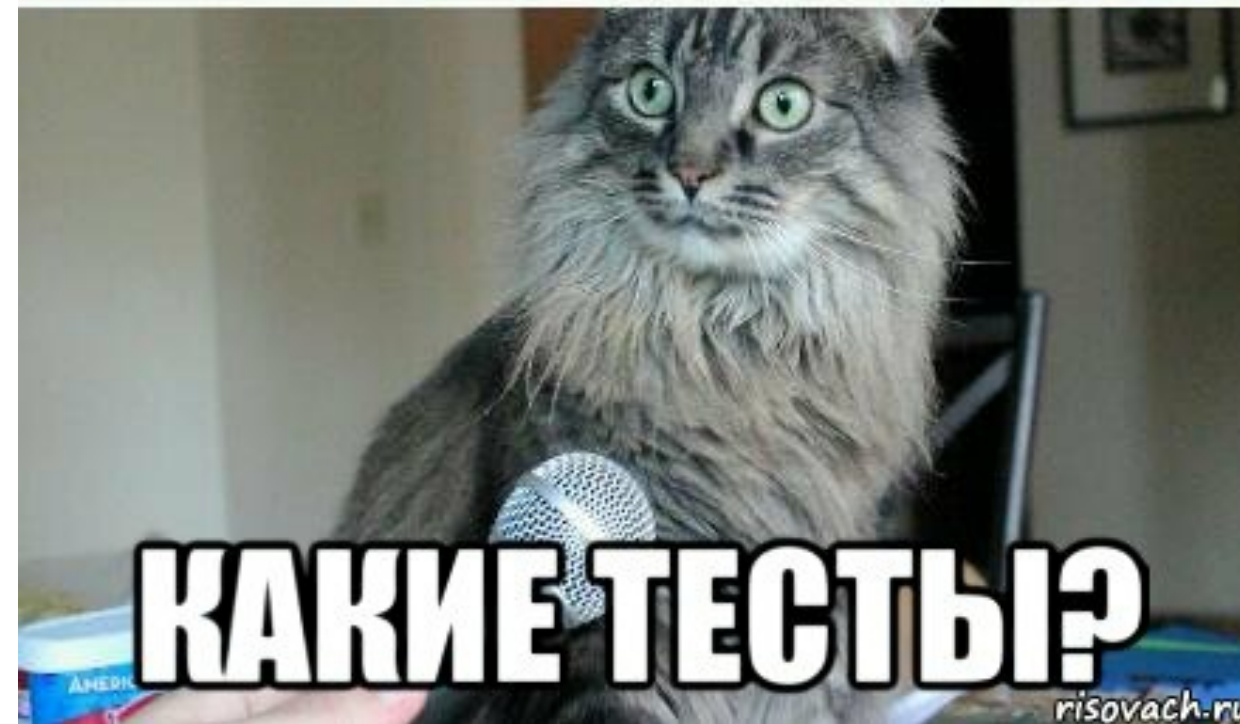
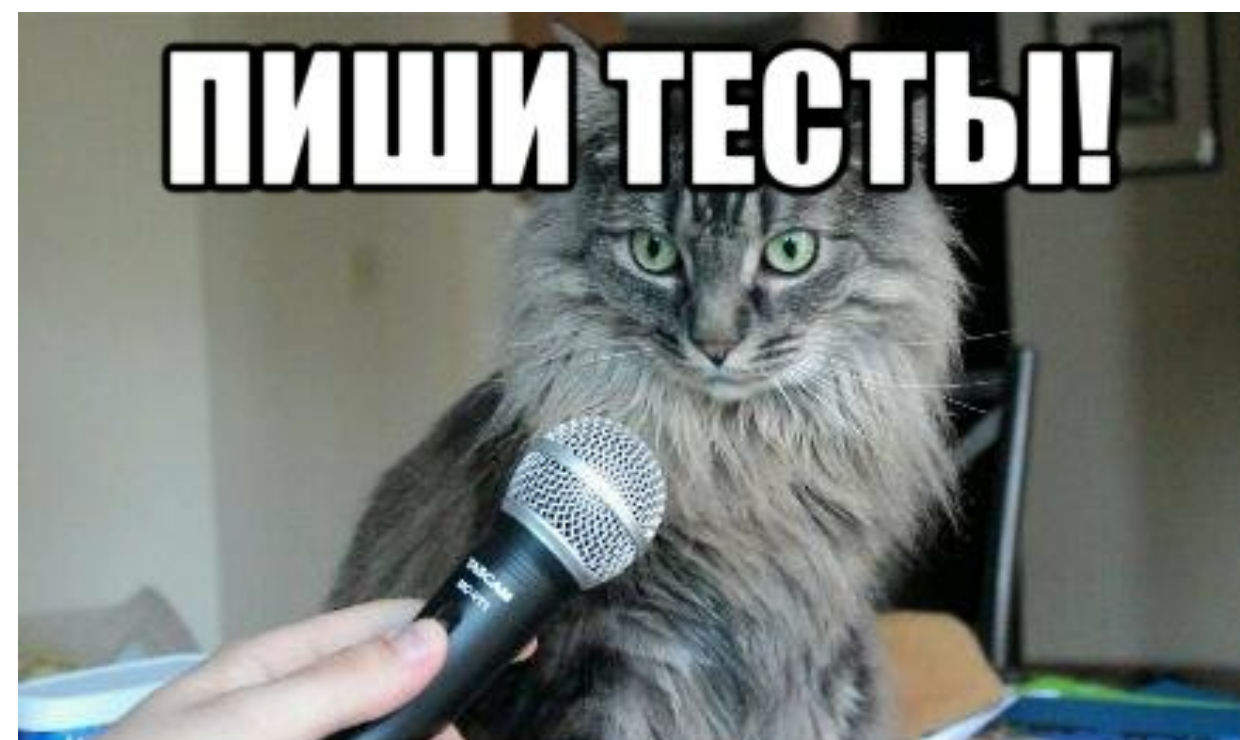
1. Использование стандарта
2. Консистентность (согласованность)
3. Версионность
4. Семантичность (компилируемость)
5. Актуальность

Как все начиналось...

- Есть монолит Backend для iOS/Android приложения
- Большой накопленный технический долг
- Скоро будет рефакторинг Backend-а
- Бэклог пустой
- Что делать?

БУДЬ МУЖИКОМ  
БЛЕЯТЬ







# Как тестировать

Как  
тестировать

- Юнит
- Интеграционные
- Функциональные
- Приемочные

Стандартное деление тестов

Как  
тестировать

- Unit Testing
- Integration Testing
- Component Testing
- Contract Testing
- End to End Testing

Testing strategies in MicroServices by [Martin Fowler](#)

Как  
тестировать

- TDD Unit Testing
- TDD Integration Testing
- TDD Component Testing
- SDD Contract Testing
- BDD End to End Testing

Что можно делать на основании  
спецификации?



Валидировать входные данные

Автоматизировать контрактное тестирование

Делать mock микросервисов

Генерировать stub данные

Генерировать документацию

Автоматизировать типовые действия

# Наш опыт

# Структура данных

## JSON schema

```
{
  "id": "http://example.com/entry-schema#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "title": "Example Schema",
  "type": "object",
  "properties": {
    "firstName": {
      "type": "string"
    },
    "lastName": {
      "type": "string"
    },
    "age": {
      "description": "Age in years",
      "type": "integer",
      "minimum": 0
    }
  },
  "required": ["firstName", "lastName"],
  "additionalProperties": false
}
```

# Валидация

```
var schema = require('../schemas/user.json');
var validate = require('jsonschema').validate;

Controller.prototype.validate = function(req, res, next) {
  var validatorResult = validate(req.body, schema);
  if (validatorResult.valid) return next();
  next({code: 400, error: validatorResult.errors});
};
```

## Спецификация

- Наш выбор RAML
- Swagger
- Api Blueprint

Выбирать вам

Abao

Контрактное тестирование RAML – Abao

Под капотом coffeescript и mocha

Валидирует схемы и коды

Пример

To be continued in Log Driven  
Development