

Лекция 4. Тестирование бекенда.

Тестирование. Практическое применение.

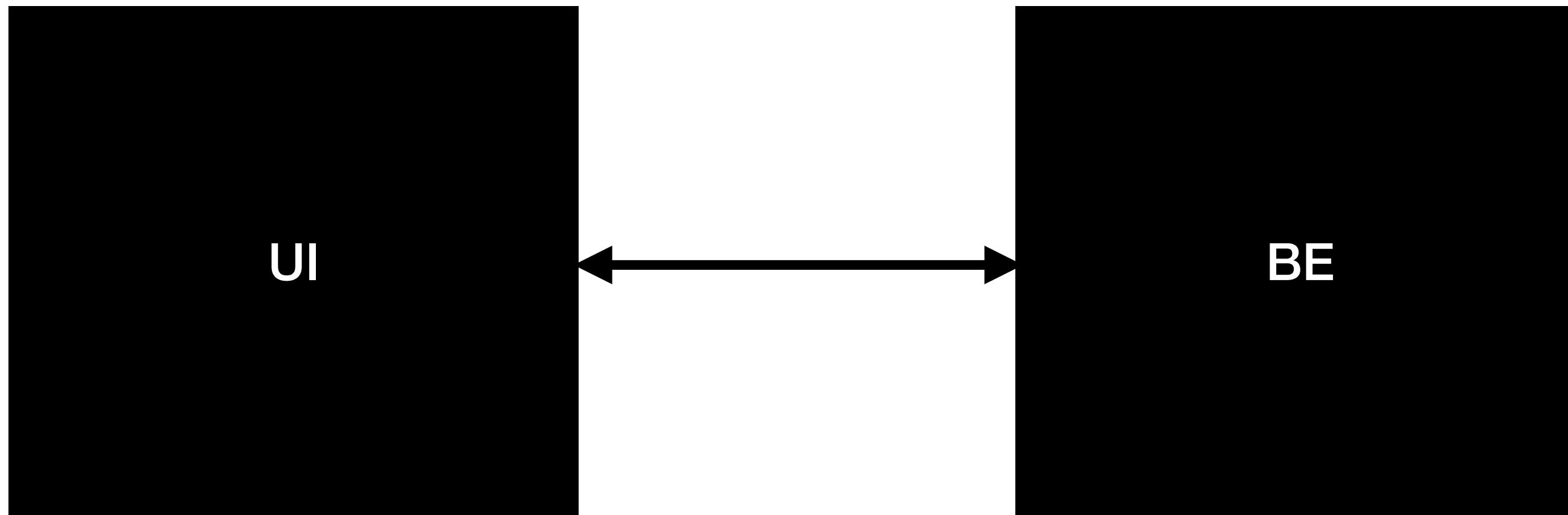
Брекелов Всеволод 30.09.2020

План

- Взаимодействие UI с ВЕ
- REST, SOAP
- Форматы данных JSON, Protocol Buffer, XML
- Тестирование REST API
- Тестирование Web Socket



Протоколы



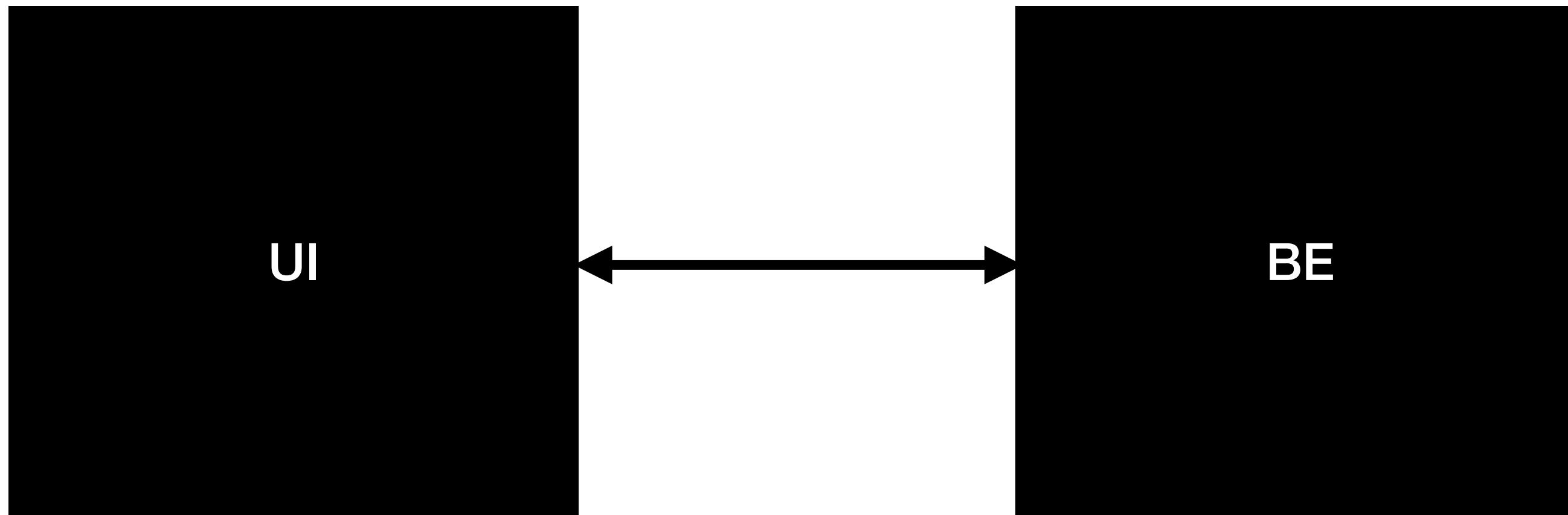
Http / Https

Binary protocols

Web Sockets



Форматы



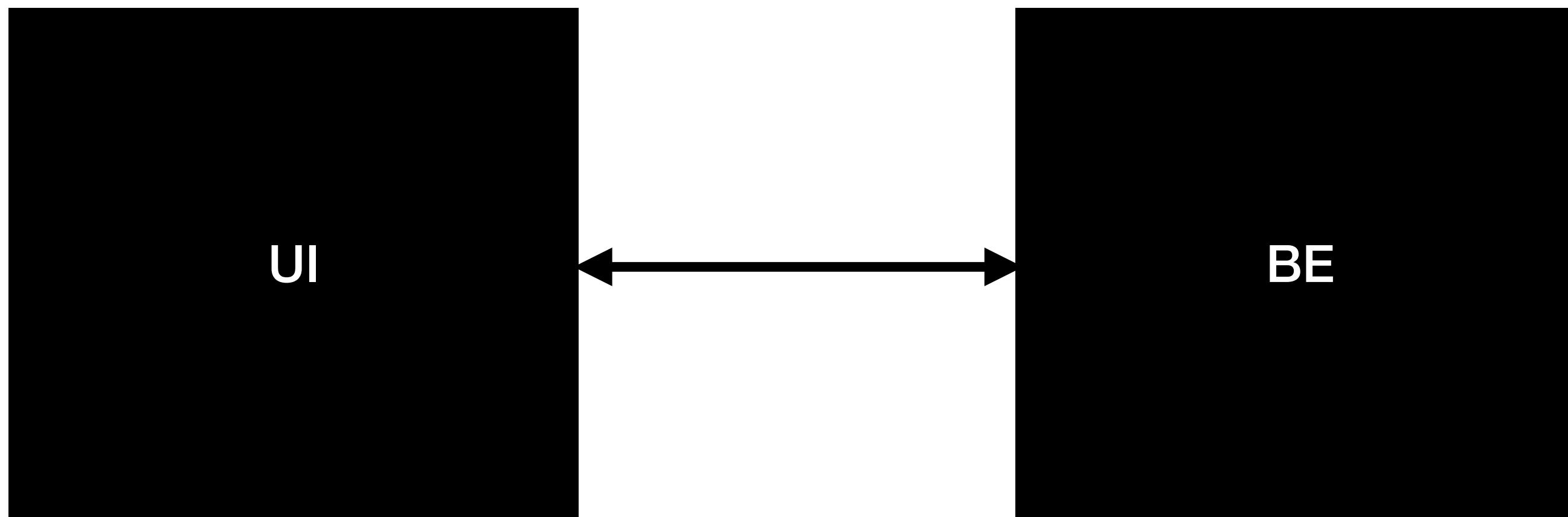
JSON

XML

Proto / ...



Простой тест



JSON

XML

Proto / ...

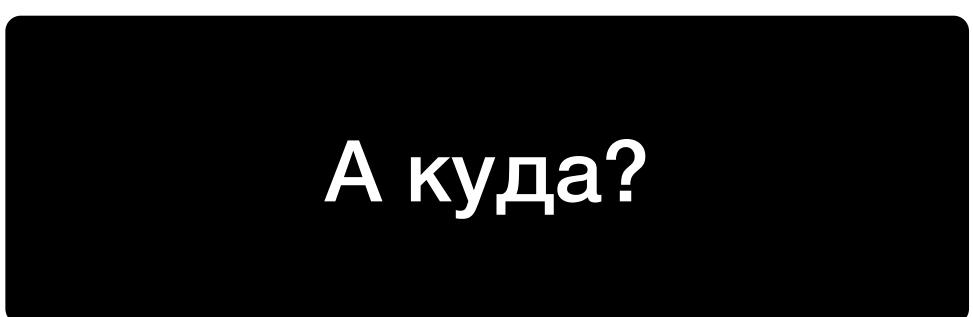
Просто так же как UI
передавать запросы





BE

/api



А куда?



Method	Endpoint
GET	https://jsonplaceholder.typicode.com/posts
POST	https://jsonplaceholder.typicode.com/posts
PUT	https://jsonplaceholder.typicode.com/posts/1
PATCH	https://jsonplaceholder.typicode.com/posts/1
DELETE	https://jsonplaceholder.typicode.com/posts/1



Ну можешь у меня
спросить

А как найти все
урлы?



SRSLY?





Шутка!
Есть много вариантов

Chrome DevTools



<https://developers.google.com/web/tools/chrome-devtools/network>



Fiddler



The screenshot shows the Fiddler Everywhere interface. The left sidebar lists sessions: WorkEasy (Session_Capture1, Session_Capture2, Session_Capture3), Kinvey, Data RPM, and New Folder. The main area displays 'Live Traffic (Capturing)' with 10 entries. The first entry is highlighted:

#	Result	Protocol	Host	URL	Body
01	200	HTTP	Tunnel 2	www.fiddler2.com	-1
02	200	HTTP	Tunnel 2	www.fiddler2.com	0
03	200	HTTP	Tunnel 2	www.fiddler2.com	0
04	200	HTTP	Tunnel 2	www.fiddler2.com	0
05	200	HTTP	Tunnel 2	www.fiddler2.com	0
06	200	HTTP	Tunnel 2	www.fiddler2.com	-1
07	200	HTTP	Tunnel 2	www.fiddler2.com	0
08	200	HTTP	Tunnel 2	www.fiddler2.com	-1
09	200	HTTP	Tunnel 2	www.fiddler2.com	0
10	200	HTTP	Tunnel 2	www.fiddler2.com	-1

The 'Request' pane shows the following HTTP request headers:

```
CONNECT telemetry.dropbox.com:443 HTTP/1.1
Connection: keep-alive
connection: keep-alive
host: telemetry.dropbox.com:443
```

The 'Response' pane shows the following HTTP response headers:

```
HTTP/1.1 200 Connection Established
FiddlerGateway: Direct
StartTime: 11:05:17.212
Connection: close
EndTime: 11:06:32.887
ClientToServerBytes: 1509
ServerToClientBytes: 657
```

Опция с сиифом
трафика

<https://www.telerik.com/fiddler>



Ну и у нас есть
Swagger API

Swagger

pet Everything about your Pets

Find out more ▾

POST /pet/{petId}/uploadImage uploads an image



POST /pet Add a new pet to the store



PUT /pet Update an existing pet



GET /pet/findByStatus Finds Pets by status



GET /pet/findByTags Finds Pets by tags



GET /pet/{petId} Find pet by ID



POST /pet/{petId} Updates a pet in the store with form data



DELETE /pet/{petId} Deletes a pet



store Access to Petstore orders

https://petstore.swagger.io/?_ga=2.163735615.1467617194.1601475562-856493072.1601475562





Ну или в проекте
посмотри сам

IDE и сорцы

Spring: Beans MVC Data

react-and-spring-data-re \${server.error.path:\$er}/ HomeController (com.greglturnquist.payroll.HomeController) @RequestMapping / ieController.index:String react-and-spring-data-rest-hypermedia

react-and-spring-data-re react-and-spring-data-re react-and-spring-data-re react-and-spring-data-re react-and-spring-data-re

6: Problems Build Terminal 9: Git 8: Services Java Enterprise TODO Spring





Ну можешь CURL
попробовать

```
curl -I --http2 https://www.keycdn.com
HTTP/2.0 200
server:keycdn-engine
date:Fri, 03 Jun 2016 17:04:55 GMT
content-type:text/html; charset=UTF-8
content-length:20634
vary:Accept-Encoding
set-cookie:keycdn=qhm4v0igli91fbpq949v3l7j73; expires=Mon, 06-Jun-2016 17:04:55
GMT; Max-Age=259200; path=/; domain=.keycdn.com; HttpOnly
expires:Fri, 03 Jun 2016 17:04:54 GMT
cache-control:no-cache
strict-transport-security:max-age=31536000; includeSubdomains; preload
content-security-policy:default-src 'self' 'unsafe-inline' 'unsafe-eval' https:
data:
x-frame-options:SAMEORIGIN
x-xss-protection:1; mode=block
x-content-type-options:nosniff
access-control-allow-origin:*
```

<https://www.keycdn.com/support/popular-curl-examples>

А как слать запросы?





Postman

The screenshot shows the Postman application interface. On the left, there is a sidebar titled "BUILDING BLOCKS" with options: "Request" (highlighted with a red arrow), "Collection", "Environment", "Documentation", "Mock Server", "Monitor", and "API". Below this is a section titled "collections" with the text "Collections let you group related requests, making them easier to access and run." and a "Create a collection" button. The main area shows a "GET Untitled Request" card with tabs for "Params", "Authorization", "Headers", "Body", "Pre-request Script", "Tests", and "Settings". The "Params" tab is active, showing a table with one row: "Key" (Value: Key). At the bottom, there is a "Response" section.

А есть вариант, где
сохранять можно?



<https://geekbrains.ru/posts/kak-testirovat-api-ili-postman-dlya-chajnikov>

Собрал в коллекцию все возможные
запросы и тестируй API



This REST Client is deprecated. Try our new HTTP Client in the editor. Convert request to the new format

HTTP method: GET Host/port: http://localhost:8080/resources Path:

Request Cookies Response Response Headers

Headers Request Parameters Request Body

Accept: */* Cache-Control: no-cache

Nothing to show

Empty Text File contents File upload (multipart/) File to send:

ПС, если юзаешь IDEA, то там есть
свой редактор запросов



Это мы разбираем как делать
интеграционный тест



- **Positive test**
 - Verify that the API receives input and returns the expected output as specified in the requirement.
 - Verify that the response status code is returned as specified in the requirement, whether it returns a 2xx or error code.
 - Specify input with minimum required fields and with maximum fields.
- **Negative test**
 - Verify that the API returns an appropriate response when the expected output does not exist.
 - Perform input validation test.
 - Verify the API's behaviors with different levels of authorization.



Не забываем про теорию



Я не очень люблю эту
тему

А что с SOAP?





Важно знать
следующее

XML

WSDL (Web Services
Description Language)



```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
        <getProductDetails xmlns="http://warehouse.example.com/ws">
            <productID>12345</productID>
        </getProductDetails>
    </soap:Body>
</soap:Envelope>
```

Ответ

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
    <soap:Body>
        <getProductDetailsResponse xmlns="http://warehouse.example.com/ws">
            <getProductDetailsResult>
                <productID>12345</productID>
                <productName>Стакан граненый</productName>
                <description>Стакан граненый. 250 мл.</description>
                <price>9.95</price>
                <currency>
                    <code>840</code>
                    <alpha3>USD</alpha3>
                    <sign>$</sign>
                    <name>US dollar</name>
                    <accuracy>2</accuracy>
                </currency>
                <inStock>true</inStock>
            </getProductDetailsResult>
        </getProductDetailsResponse>
    </soap:Body>
</soap:Envelope>
```



Да

Т.е. в WSDL вся информация
по API как Swagger?





SoapUI

А как тестировать?



<https://www.soapui.org/getting-started/soap-test/>



умеет

А POSTMAN умеет?

The screenshot shows the Postman application interface. At the top, it says "POST SOAP number convert". Below that, under the heading "SOAP number convert", there is a "POST" button highlighted in blue, followed by a URL "https://www.dataaccess.com/webservicesserver/NumberConversion.wso". To the left of the URL, there are buttons for "GET", "POST", "PUT", "PATCH", and "DELETE". To the right of the URL, there are tabs for "Headers (9)", "Body", "Pre-request Script", "Tests", and "Settings".



<https://learning.postman.com/docs/sending-requests/supported-api-frameworks/making-soap-requests/>

Разобрались как руками протестировать
Rest Api, Soap с Json/xml форматами

А что с бинарными протоколами?



RPC(Remote Procedure Call) стек

Avro
Protobuf
Thrift

Есть такое



JSON-RPC

gRPC



И такое

<https://github.com/sidshetye/SerializersCompare>

Придется писать код

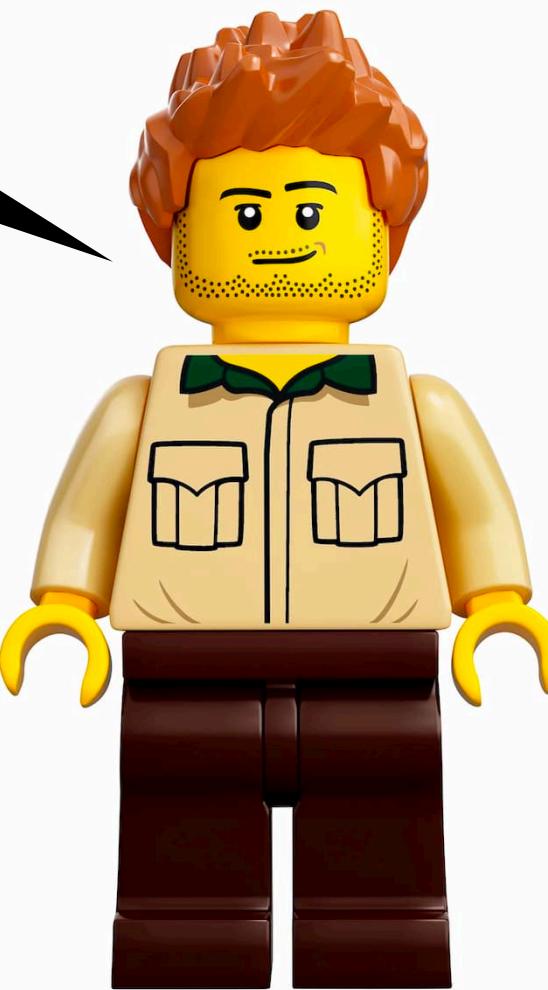


**О класс!
Я нашел такие готовые библиотеки
для тестирования**

<https://github.com/njpatel/grpcc>

<https://github.com/protobufUI/protobufUI>

<https://github.com/uber/prototool>



А не забыли про WebSocket?

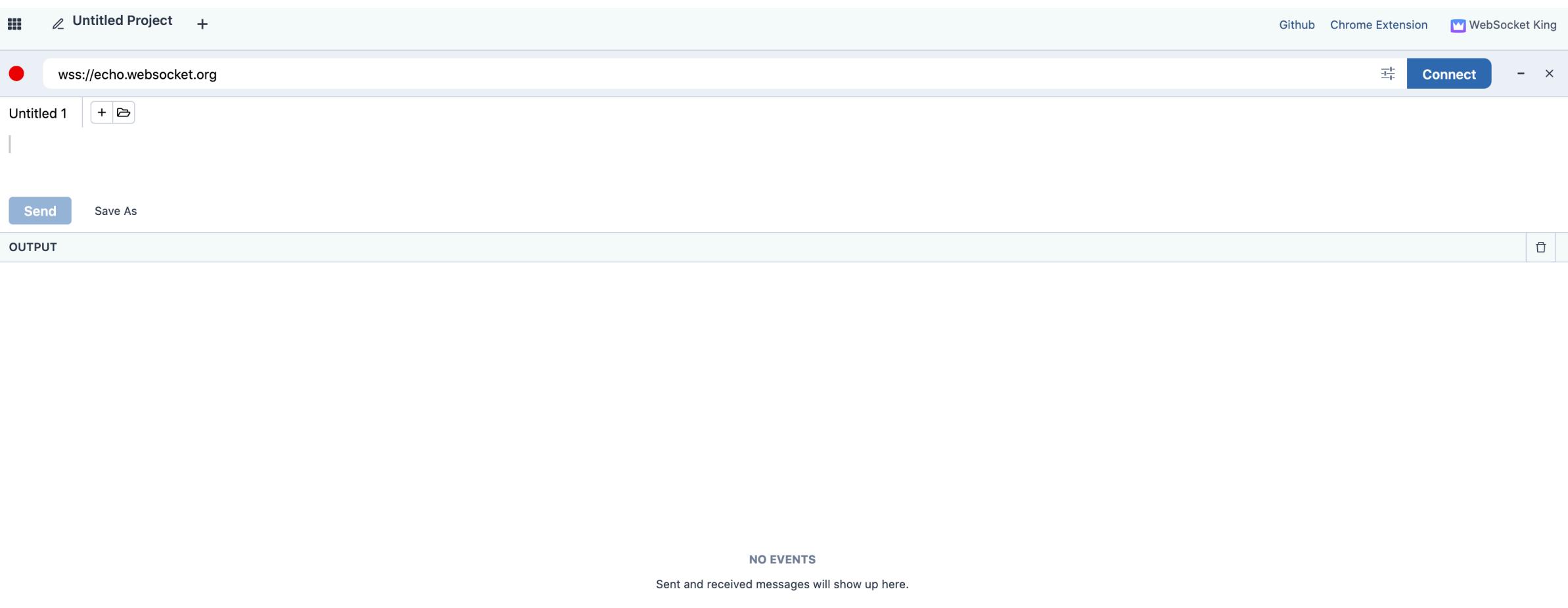


Можно и cURL



```
$ curl --include \
--no-buffer \
--header "Connection: Upgrade" \
--header "Upgrade: websocket" \
--header "Host: example.com:80" \
--header "Origin: http://example.com:80" \
--header "Sec-WebSocket-Key: SGVsbG8sIHdvcmxkIQ==" \
--header "Sec-WebSocket-Version: 13" \
http://example.com:80/
```

Можно поискать online штуки



<https://websocketking.com/>

Погрузить что Postman не умеет



Support WebSocket Endpoints #4009

! Open

matanpramp opened this issue on Dec 28, 2017 · 170 comments

<https://github.com/postmanlabs/postman-app-support/issues/4009>

Разобрались как тестировать по API



Все полезные ссылки на странице лекций

В следующих сериях

Spring / Java

Тестирование DB

Unit Тестирование

Component testing

Integration test with TestContainers

