urlMock=http://localhost:5555/api/

|  |  |  |
| --- | --- | --- |
| **Методы-заглушки** | **Request** | **Response** |
| Получение списка пользователей | /user/get/all | { "name":"Test user", "cource":"QA", "email":"[test@test.test](mailto:test@test.test)" "age": 23 } |
| Получение списка курсов | /cource/get/all | [ { "name":"QA java", "price": 15000 }, { "name":"Java", "price": 12000 } ] |
| Получение оценки пользователя | /user/get/{id} | { "name":"Test user", "score": 78 } |

* создать тест с http-helper
* создать тест с soap-helper

дополнительно (по желанию):

* создать тест с sql-helper
* создать тест с mq-helper

## Домашнее задание

ДЗ #3: Stub & Api Testing (API-helpers)

**Цель:**

1.Реализовать Stub сервер со следующими методами-заглушами

* получение списка пользователей
* получение списка курсов
* оценка пользователя

1. Создать "хелперы" (API-helpers) для работы с API.  
   Это можно сделать в отдельном проекте. API-helpers должны тестировать реальные сервисы в интернете или сервисы развернутые на вашей локальной машине.  
   Нужно поработать с:

* http-helper
* soap-helper  
  дополнительное, более сложное задание (по желанию) :
* создать рабочий sql-helper (для получения данных из БД)
* создать рабочий mq-helper (для отправки данных в менеджер сообщений)

Вышеуказанные задачи можно реализовать с помощью Citrus Framework.  
По http запросам можно применить RestAssured, Retrofit или Citrus (на ваше усмотрение).  
Для заглушек можно использовать Wiremock или Citrus Framework (на ваше усмотрение).

**Описание/Пошаговая инструкция выполнения домашнего задания:**

1. Необходимо реализовать Stub сервер на Wiremock со следующими endpoint'ами:

* /user/get/{id} для получение оценки пользователя
* /cource/get/all для получения списка курсов
* /user/get/all - для получения списка всех пользователей  
  Контракты  
  Для user  
  {  
  "name":"Test user",  
  "cource":"QA",  
  "email":"[test@test.test](mailto:test@test.test)"  
  "age": 23  
  }  
  Для оценки:  
  {  
  "name":"Test user",  
  "score": 78  
  }  
  Для курсов:  
  [  
  {  
  "name":"QA java",  
  "price": 15000  
  },  
  {  
  "name":"Java",  
  "price": 12000  
  }  
  ]  
  Подключить stub frontend, написать тесты для проверки json cхем.

2.

* создать тест с http-helper
* создать тест с soap-helper  
  дополнительно (по желанию):
* создать тест с sql-helper
* создать тест с mq-helper

**Критерии оценки:**

1.Максимальная оценка 10 баллов

* 2 балла за реализованные метод для получения списка пользователей
* 2 балла за реализованные метод для получения списка курсов
* 2 балла за реализованные метод для получения оценки пользователя
* 1 балла за подключенный stub frontend
* 1 балла за подключенный Selenoid
* 1 балла за реализацию тестов через Cucumber
* 1 балла за написание тестов для проверки схем пользователей, оценки, курсов
* 1 балла работающий http-helper
* 1 балла работающий soap-helper
* 1 бал за не реализованный метод stub backend
* 1 бал за каждое действие преподавателя для запуска тестов
* 0.5 баллов за не написанный тест  
  Сдача:  
  Сдача ДЗ происходит в git в ветке homework\_3  
  Дополнительные баллы будут отмечены в сертификате, как задания повышенной сложности.