

Санкт-Петербургский государственный электротехнический университет им.
В.И. Ульянова (Ленина)

ИССЛЕДОВАНИЕ МОДЕЛИ ОБРАЩЕНИЯ К WEB-СЛУЖБЕ ASP.NET ЧЕРЕЗ ПРОКСИ СБОРКУ НА ПРИМЕРЕ РАЗРАБОТКИ СЕРВИСА ПО АРЕНДЕ ВИДЕОФИЛЬМОВ

Выполнил: Маслов Никита Дмитриевич, гр. 5304
Руководитель: Попова Елена Владимировна, к.т.н., доцент

Санкт-Петербург, 2021

Цель и задачи

Цель: исследование модели обращения Web-приложения к Web-службе

Актуальность: Для того что бы web-приложение быстро и без ошибок выполняло возложенные на него функции необходимо правильно разработать модель взаимодействия его клиентской и серверной частей. Поэтому исследование модели взаимодействия Web-службы и Web-приложения будет являться всегда актуальной.

Задачи:

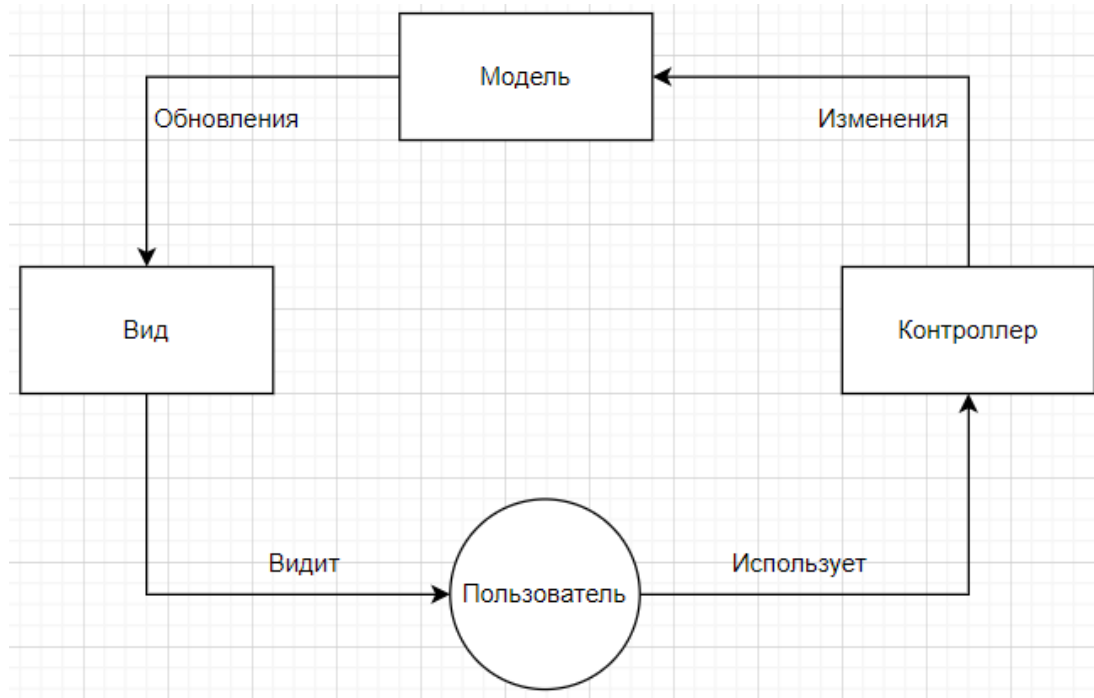
1. Изучить теоретические аспекты разработки web-приложений с использованием технологии ASP.NET
2. Разработать модель взаимодействия Web-службы и Web-приложения
3. Разработать Web-приложение и Web-службу и провести исследование модели их взаимодействия

Технология ASP.NET

Плюсы:

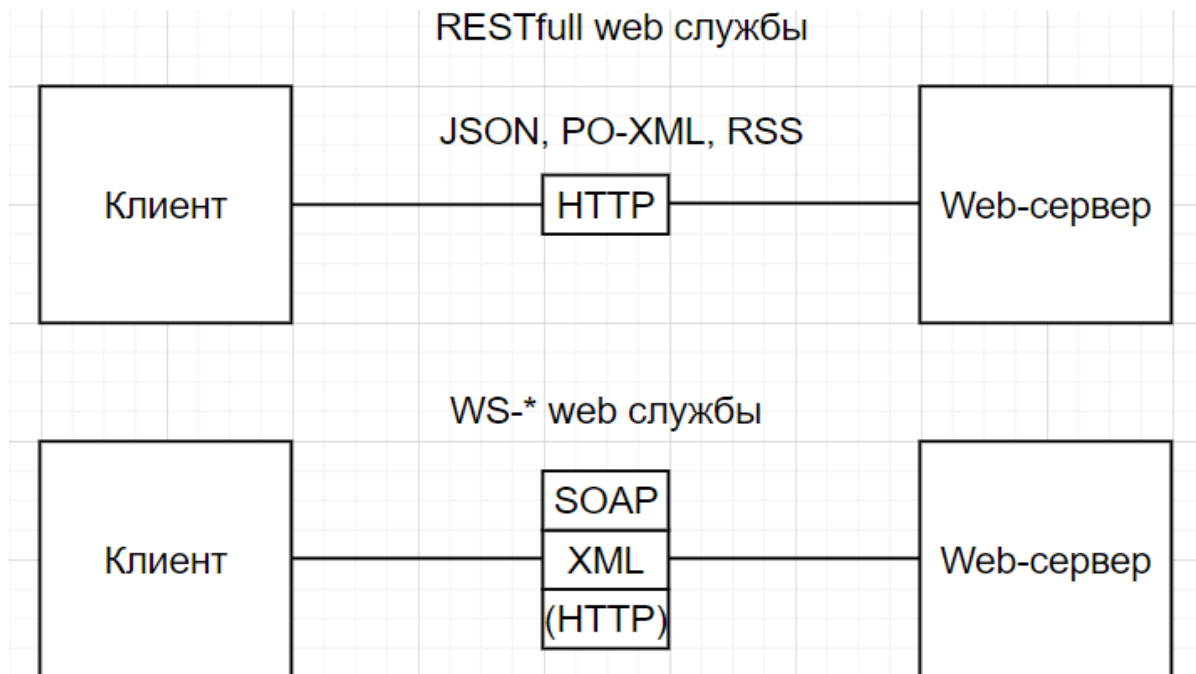
- Ускоренный процесс разработки
- Компилируемый код
- Большой набор готовых компонентов
- Документация

Архитектура MVC



Визуализация паттерна MVC

Разработка модели взаимодействия между web-приложением и web-службой



Архитектура клиент-сервер, основанная на web-технологии при соединении REST и SOAP

Прокси

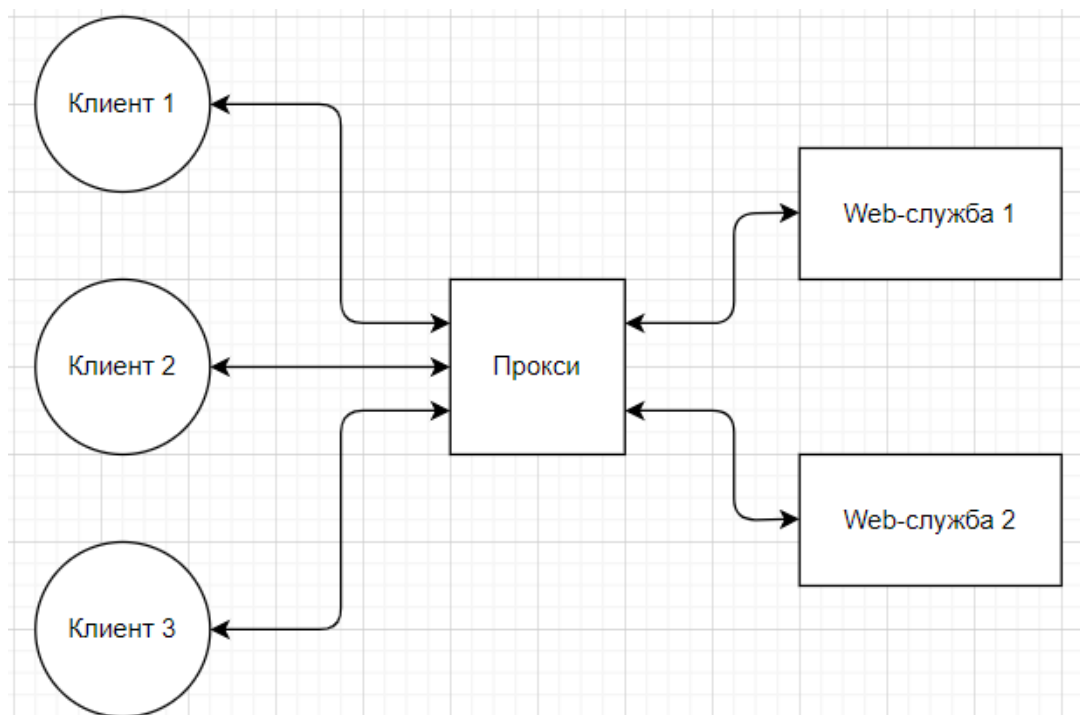
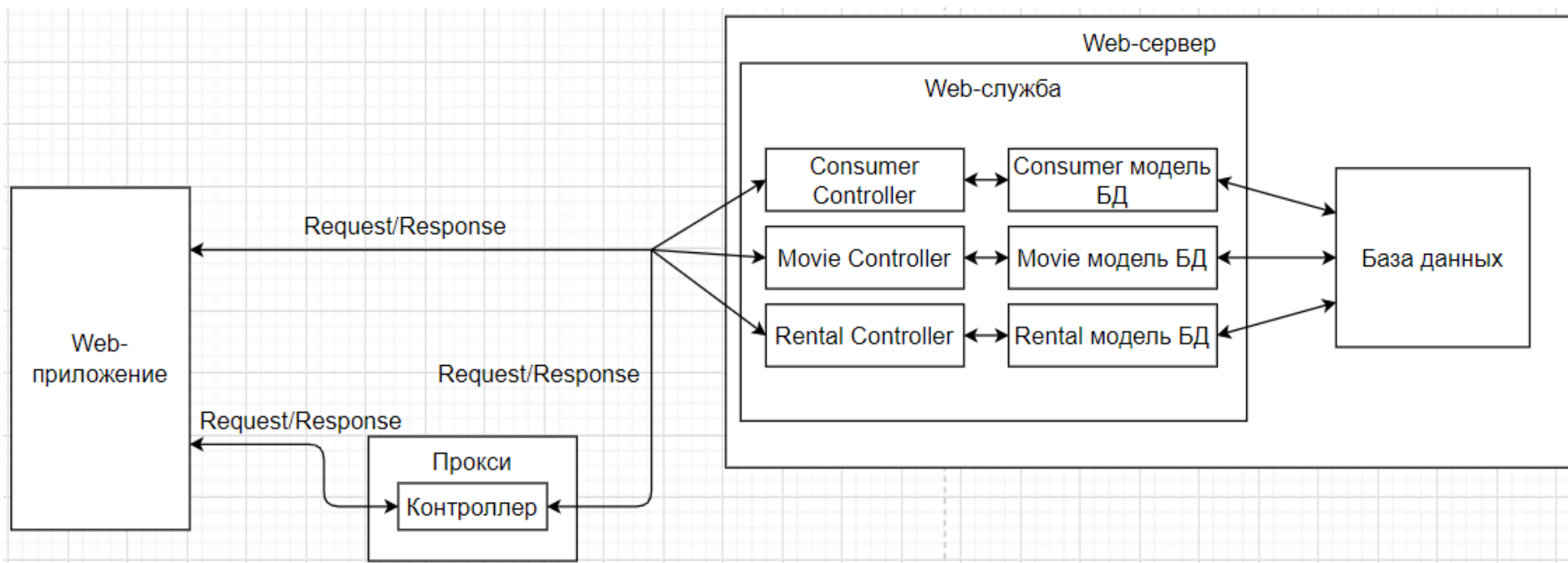


Схема связи клиентов и служб через прокси

Модель взаимодействия web-приложения с web-службой



Модель взаимодействия Web-приложения с Web-службой

Разработка web-приложения

MovieRent

New Rental

Customers

Movies

Register

Log in

MovieRent

Renting a movie has never been easier. Start with us now

© 2021 - MovieRent

Главная страница web-приложения

Разработка web-службы

Основные маршруты службы:

Контроллер Customers(тарифный план для аренды):

GET [URL web-службы]/api/customers

POST [URL web-службы]/api/customers

PUT [URL web-службы]/api/customers/[:id]

DELETE [URL web-службы]/api/customers/[:id]

Контроллер Movies(фильмы):

GET [URL web-службы]/api/movies

POST [URL web-службы]/api/movies

PUT [URL web-службы]/api/movies/[:id]

DELETE [URL web-службы]/api/movies/[:id]

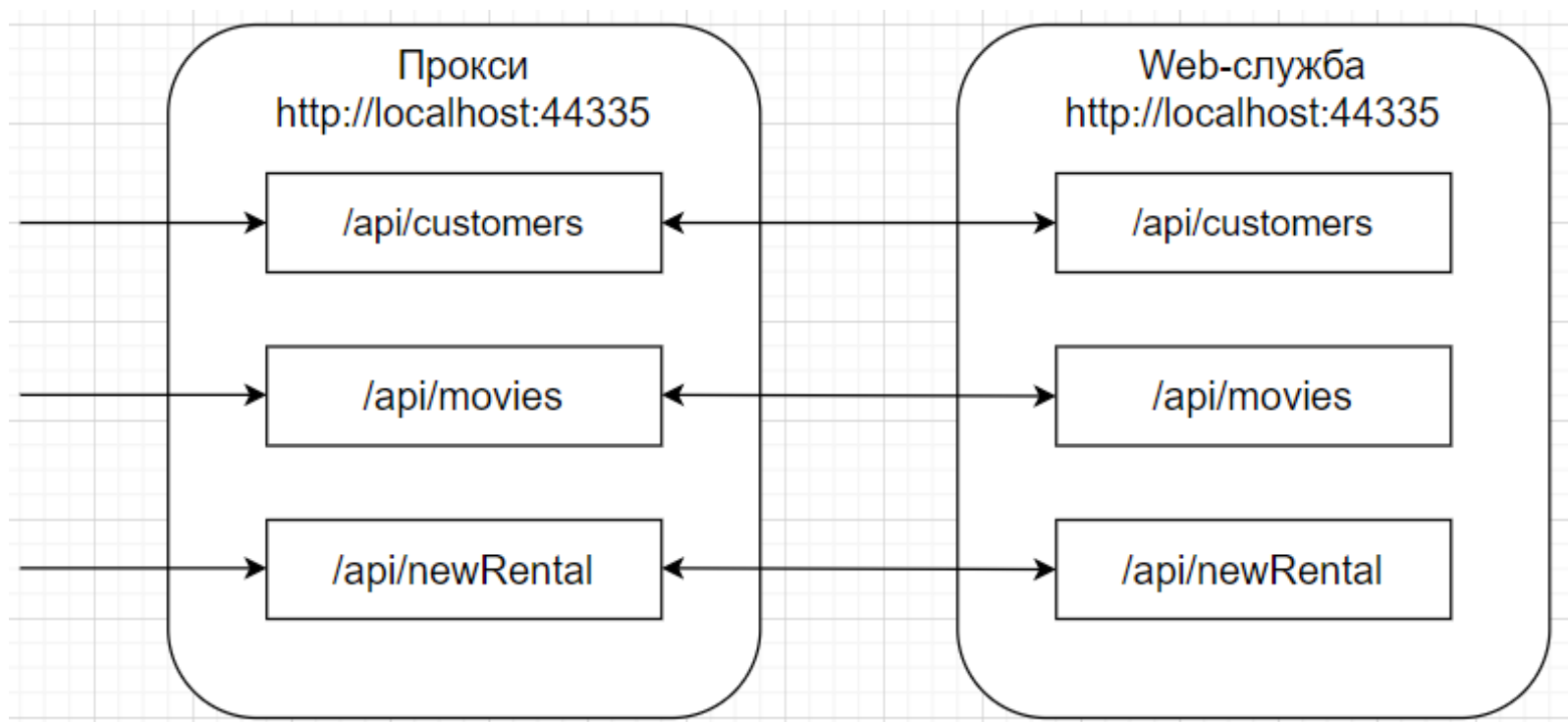
Контроллер NewRental(аренда фильма):

POST [URL web-службы]/api/NewRental

Реализация прокси

Адрес приложения - <https://localhost:44300/>

Адрес прокси - <https://localhost:44355/>



Графическое представление перенаправления запросов

Исследование модели

The screenshot displays a REST client interface with a 'Tests' tab selected. The test script is written in JavaScript and performs 1000 iterations of a GET request to a local API endpoint. The results section shows five individual test passes for response times and one overall summary pass for the average response time and iteration count.

Params Authorization Headers (7) Body Pre-request Script Tests ●

```
1 iterations = 1000;  
2 delay = 100;  
3 responseTimes = [];  
4  
5 i=0;  
6  
7 function sendRequest() {  
8   pm.sendRequest({  
9     url: "https://localhost:44300/api/movies",  
10    method: 'GET'  
11  }, function (err, res) {
```

Body Cookies (1) Headers (10) Test Results (1001/1001)

All Passed Skipped Failed

- PASS Response time is 9
- PASS Response time is 13
- PASS Response time is 6
- PASS Response time is 12
- PASS Average response time is 14.845 ms; the number of iterations is 1000

GET запрос к службе напрямую

Исследование модели

Аппаратное обеспечение: Windows 10, 64-разрядная (10.0, сборка 19041), ОЗУ – 16гб, процессор – Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz (8CPUs)

Программное обеспечение: «Postman»

Результаты исследования

	Время среднее время 1000 GET запросов	Время среднее время 1000 POST запросов	Соккрытие адреса Web- службы
Напрямую	14.845 ms	23.048 ms	Нет
Через прокси	28.405 ms	37.927 ms	Да

Исследование авторизации показало, что запросы себя ведут аналогичным образом и при запросе напрямую и через прокси, авторизованный пользователь может выполнять действия с моделями базы данных, а не авторизованный не может, и пересылается на страницу логина.

Заключение

1. Были изучены теоретические основы разработки web-приложения с использованием технологии ASP.NET
2. Разработана модель взаимодействия web-приложения и web-службы
3. Разработаны Web-приложение и Web-служба и проведено исследование модели их взаимодействия

Исследование модели показало, что с использованием прокси увеличивается время ответа сервера. Для GET запроса в среднем на 13.5ms, а для POST 14.9ms. Из этого можно сделать вывод, что прокси необходимо использовать для реализации какой-либо дополнительной функциональности, например для увеличения безопасности приложения, упрощения взаимодействия между несколькими службами или снижения нагрузки на сервер, путём кэширования запросов пользователя в прокси, в противном случае внедрение прокси может замедлить время ответов сервера и как следствие замедлить работу всего приложения в целом.

В дальнейшем можно провести аналогичное исследование для SOAP

Апробация работы

Репозиторий проекта

<https://github.com/NikitaMasl/MovieRent>



Спасибо за внимание

Дополнительные слайды

<div> <div>Manifest</div> <div>Service Workers</div> <div>Storage</div> </div> <div>Storage</div> <div> <div>Local Storage</div> <div>Session Storage</div> <div>IndexedDB</div> <div>Web SQL</div> <div>Cookies</div> </div> <div>https://localhost:44300</div>	Name	Value	Do...	P...	Exp...	Size	Http...	Sec...	Sa...	Sa...	Pri...
	1P_JAR	2021-05-08-14	.gst...	/	20...	19		✓	No...		Me...
	.AspNet.Applicatio...	RcBUYtnu89p-vJ8gjbGSpR26eO...	loc...	/	Ses...	516	✓	✓			Me...
	next-i18next	en	loc...	/	20...	14			Strict		Me...
	__RequestVerificati...	XRku1eU1MBrEFhxeMdj_ckrfa8y...	loc...	/	Ses...	134	✓				Me...
	_ga	GA1.1.1871714445.1586808233	loc...	/	20...	30					Me...
	vc	6	loc...	/	20...	3					Me...

Куки