

Please visit website: <http://cxyroad.com>

SpringBoot+Spring WebFlux响应式开发，实现打字效果

=====

Hi,大家好，我是抢老婆酸奶的小肥仔。

日常逛网站时，我们有时候会用到在线客服，这种功能大部分都是采用机器人来进行实现，也会出现打字效果。因此，我们仿照实现这样的效果。先看效果图：

至于咨询那部分实现，我们就不在此篇文章中实现了。大家可以接入ChatGPT或通义千问来实现。

1、Spring WebFlux简介

Spring WebFlux是实现响应式编程，基于Servlet3.x之后异步响应处理所提供的更简化的实现模式，该组件是一个重新构建的且基于Reactive Streams标准实现的**异步非阻塞**Web开发框架，以Reactive开发框架为基础，可以更加容易的实现**高并发**访问下的请求处理模式。

在SpringBoot 2.x版本中你那个提供了“Spring-webflux”依赖模块，该模块有两种编程模式实现：一种基于功能性端点方式，一种基于SpringMVC注解方式。

Spring WebFlux在进行具体请求处理前，需要首先配置一个请求终端，而后依据路由匹配的地址找到指定终端类中所提供的处理方法进行操作。

2、Spring WebFlux与SpringMVC区别

区别	**Spring WebFlux**	**SpringMVC**
编程模型	采用异步、非阻塞的编程模型。基于Reactive Streams标准，使用反应式编程的理念，可以更有效地处理大量并发请求，减少线程资源的浪费。	采用同步、阻塞的编程模型，每个请求都会在一个单独的线程中处理，线程会一直阻塞直到请求完成。
并发处理	使用非阻塞IO，通过少量的线程处理大量的并发请求。	使用Servlet API中的阻塞IO来处理请求
适用场景	适用于需要处理大量并发请求，对实时性要求高的场景，必然要实时通信等。	适用于传统的同步IO的应用场景，特别是那些对实时性要求不是很高的场景。

3、Spring WebFlux与Spring MVC是否同时存在？

可以同时存在。

4、Spring WebFlux关键概念

* `Flux(流)`：表示包含零个或多个元素的异步序列的主要类型。在WebFlux中，Flux表示请求和响应的数据流。

...

```
Flux<Object> obj = Flux.just(1,2,3);
```

...

* `Mono(单值)`：表示零个或一个元素的异步序列的类型。

...

```
Mono<Object> obj = Mono.just("Spring WebFlux");
```

...

* `Scheduler(调度器)`：Reactor提供了调度器来控制异步操作的执行。调度器用于在不同线程或线程池中执行操作，以避免阻塞。

...

```
Scheduler scheduler = Schedulers.parallel();
```

...

* `Operators(操作符)`：Reactor提供了丰富的操作符，用于在Flux和Mono上执行各种转换和操作。

...

```
Flux<Integer> val = obj.map(n -> n+1);
```

...

5、SpringBoot集成WebFlux

5.1 引入jar包

...

```
<!--Spring WebFlux-->
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-webflux</artifactId>
  <version>3.2.5</version>
</dependency>
```

...

5.2 实现Flux工具

该工具提供了Mono，Flux返回方法，同时提供了Function的执行方法。

```
...
/**
 * @author: jiangjs
 * @description:
 * @date: 2024/5/17 10:41
 **/
@Component
public class ChatMsgFluxUnit<T,R> {

    public Mono<R> getMonoChatMsg(T t, Function<T,R> function){
        return Mono.just(t).map(function).onErrorResume(e->Mono.empty());
    }

    public Flux<R> getMoreChatMsg(T t, Function<T,R> function){
        return Flux.just(t).map(function).onErrorResume(e->Flux.empty());
    }
}
...
```

5.3 测试

创建Controller类，提供测试方法。

```
...
/**
 * @author: jiangjs
 * @description:
 * @date: 2024/5/16 14:44
 **/
@RestController
@RequestMapping("/chat")
public class ChatMsgController {
    @Resource
    private ChatMsgFluxUnit<String,Map<String,String>>
chatMsgFluxUnit;

    @GetMapping(value = "/flux",produces =
MediaType.TEXT_EVENT_STREAM_VALUE)
```

```
@CrossOrigin(origins = "*")
public Flux<Map<String,String>> flux(@RequestParam("content")
String content){
    return chatMsgFluxUnit.getMoreChatMsg("你要问啥？请详细描述一下你的问题
.....,wqwqwqwqwqwwwwwqwqwqwqwqwqwqwqwqwqwqwqwqwqwqwqwqwqwqwqwqsasasasawqwqw", s -> {
        Map<String, String> map = new HashMap<>(1);
        map.put("msg", s);
        return map;
    });
}
```

测试结果：

测试结果会出现乱码，小伙伴可自行处理。

6、接入前端并实现打字效果

前端调用WebFlux时，需要创建EventSource，连接WebFlux地址。

```

    this.eventSource = new
    EventSource('http://127.0.0.1:8008/word/chat/main?content='+this.input
    tText);

```

实现打字效果，需要将WebFlux返回的数据进行拆分嘛，然后通过setTimeout每隔一段时间取一个字进行显示，即可实现改效果。

```

...
const strings = data.msg.split("");
strings.forEach((obj,index) => {
  setTimeout(() => {
    this.messages[this.messages.length - 1].text += obj;
    if (index > 0 && index % 20 === 0) {
      this.messages[this.messages.length - 1].text += "\n";
    }
  },index*80);
})
...

```

前端完整代码：

```

...
<template>
  <div class="bg-gray-100 h-screen flex flex-col max-w-lg mx-auto">
    <div class="bg-blue-500 p-3 text-white flex justify-between items-center">
      <button id="login" class="hover:bg-blue-400 rounded-md p-1">
        <svg width="25px" height="25px" viewBox="0 0 24 24" fill="none" xmlns="http://www.w3.org/2000/svg"><g id="SVGRepo_bgCarrier" stroke-width="0"></g><g id="SVGRepo_tracerCarrier" stroke-linecap="round" stroke-linejoin="round"></g><g id="SVGRepo_iconCarrier"> <circle cx="12" cy="6" r="4" stroke="#ffffff" stroke-width="1.5"></circle> <path d="M15 20.6151C14.0907 20.8619 13.0736 21 12 21C8.13401 21 5 19.2091 5 17C5 14.7909 8.13401 13 12 13C15.866 13 19 14.7909 19 17C19 17.3453 18.9234 17.6804 18.7795 18" stroke="#ffffff" stroke-width="1.5" stroke-linecap="round"></path> </g></svg>
        </button>
      <span>咨询平台</span>
      <div class="relative inline-block text-left">
        <button id="setting" class="hover:bg-blue-400 rounded-md p-1">
          <svg width="30px" height="30px" viewBox="0 0 24 24" fill="none" xmlns="http://www.w3.org/2000/svg"><g id="SVGRepo_bgCarrier" stroke-width="0"></g><g id="SVGRepo_tracerCarrier" stroke-linecap="round" stroke-linejoin="round"></g><g id="SVGRepo_iconCarrier"> <path fill-rule="evenodd" clip-rule="evenodd" d="M14.1395 12.0002C14.1395 13.1048 13.2664 14.0002 12.1895 14.0002C11.1125 14.0002 10.2395 13.1048 10.2395 12.0002C10.2395 10.8957 11.1125 10.0002 12.1895

```

```
10.0002C13.2664 10.0002 14.1395 10.8957 14.1395 12.0002Z"
stroke="#ffffff" stroke-width="1.5" stroke-linecap="round" stroke-
linejoin="round"></path> <path fill-rule="evenodd" clip-rule="evenodd"
d="M7.57381 18.1003L5.12169 12.8133C4.79277 12.2907 4.79277
11.6189 5.12169 11.0963L7.55821 5.89229C7.93118 5.32445 8.55898
4.98876 9.22644 5.00029H12.1895H15.1525C15.8199 4.98876 16.4477
5.32445 16.8207 5.89229L19.2524 11.0923C19.5813 11.6149 19.5813
12.2867 19.2524 12.8093L16.8051 18.1003C16.4324 18.674 15.8002
19.0133 15.1281 19.0003H9.24984C8.5781 19.013 7.94636 18.6737
7.57381 18.1003Z" stroke="#ffffff" stroke-width="1.5" stroke-
linecap="round" stroke-linejoin="round"></path> </g></svg>
</button>
<div id="dropdown-content" class="hidden absolute right-0
mt-2 w-48 bg-white border border-gray-300 rounded-lg shadow-lg p-
2">
    <a href="#" class="flex items-center px-4 py-2 text-
gray-800 hover:bg-gray-200 rounded-md">
        <svg width="20px" height="20px" viewBox="0 0 24 24"
fill="none" class="mr-2" xmlns="http://www.w3.org/2000/svg"><g
id="SVGRepo_bgCarrier" stroke-width="0"></g><g
id="SVGRepo_tracerCarrier" stroke-linecap="round" stroke-
linejoin="round"></g><g id="SVGRepo_iconCarrier"> <path d="M9
21H12M15 21H12M12 21V18M12 18H19C20.1046 18 21 17.1046 21
16V7C21 5.89543 20.1046 5 19 5H5C3.89543 5 3 5.89543 3 7V16C3
17.1046 3.89543 18 5 18H12Z" stroke="#000000" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"></path>
</g></svg>Appearance
    </a>
    <a href="#" class="flex items-center px-4 py-2 text-
gray-800 hover:bg-gray-200 rounded-md">
        <svg width="20px" height="20px" viewBox="0 0 24 24"
class="mr-2" fill="none" xmlns="http://www.w3.org/2000/svg"><g
id="SVGRepo_bgCarrier" stroke-width="0"></g><g
id="SVGRepo_tracerCarrier" stroke-linecap="round" stroke-
linejoin="round"></g><g id="SVGRepo_iconCarrier"> <path
d="M11.2691 4.41115C11.5006 3.89177 11.6164 3.63208 11.7776
3.55211C11.9176 3.48263 12.082 3.48263 12.222 3.55211C12.3832
3.63208 12.499 3.89177 12.7305 4.41115L14.5745 8.54808C14.643
8.70162 14.6772 8.77839 14.7302 8.83718C14.777 8.8892 14.8343
8.93081 14.8982 8.95929C14.9705 8.99149 15.0541 9.00031 15.2213
9.01795L19.7256 9.49336C20.2911 9.55304 20.5738 9.58288 20.6997
9.71147C20.809 9.82316 20.8598 9.97956 20.837 10.1342C20.8108
10.3122 20.5996 10.5025 20.1772 10.8832L16.8125 13.9154C16.6877
14.0279 16.6252 14.0842 16.5857 14.1527C16.5507 14.2134 16.5288
14.2807 16.5215 14.3503C16.5132 14.429 16.5306 14.5112 16.5655
14.6757L17.5053 19.1064C17.6233 19.6627 17.6823 19.9408 17.5989
20.1002C17.5264 20.2388 17.3934 20.3354 17.2393 20.3615C17.0619
20.3915 16.8156 20.2495 16.323 19.9654L12.3995 17.7024C12.2539
```

```
17.6184 12.1811 17.5765 12.1037 17.56C12.0352 17.5455 11.9644
17.5455 11.8959 17.56C11.8185 17.5765 11.7457 17.6184 11.6001
17.7024L7.67662 19.9654C7.18404 20.2495 6.93775 20.3915 6.76034
20.3615C6.60623 20.3354 6.47319 20.2388 6.40075 20.1002C6.31736
19.9408 6.37635 19.6627 6.49434 19.1064L7.4341 14.6757C7.46898
14.5112 7.48642 14.429 7.47814 14.3503C7.47081 14.2807 7.44894
14.2134 7.41394 14.1527C7.37439 14.0842 7.31195 14.0279 7.18708
13.9154L3.82246 10.8832C3.40005 10.5025 3.18884 10.3122 3.16258
10.1342C3.13978 9.97956 3.19059 9.82316 3.29993 9.71147C3.42581
9.58288 3.70856 9.55304 4.27406 9.49336L8.77835 9.01795C8.94553
9.00031 9.02911 8.99149 9.10139 8.95929C9.16534 8.93081 9.2226
8.8892 9.26946 8.83718C9.32241 8.77839 9.35663 8.70162 9.42508
8.54808L11.2691 4.41115Z" stroke="#000000" stroke-width="2"
stroke-linecap="round" stroke-linejoin="round"></path>
</g></svg>Favorite
    </a>
    <a href="#" class="flex items-center px-4 py-2 text-
gray-800 hover:bg-gray-200 rounded-md">
        <svg width="20px" height="20px" viewBox="0 0 24 24"
class="mr-2" fill="none" xmlns="http://www.w3.org/2000/svg"><g
id="SVGRepo_bgCarrier" stroke-width="0"></g><g
id="SVGRepo_tracerCarrier" stroke-linecap="round" stroke-
linejoin="round"></g><g id="SVGRepo_iconCarrier"> <g id="Warning /
Info"> <path id="Vector" d="M12 11V16M12 21C7.02944 21 3 16.9706
3 12C3 7.02944 7.02944 3 12 3C16.9706 3 21 7.02944 21 12C21
16.9706 16.9706 21 12 21ZM12.0498 8V8.1L11.9502
8.1002V8H12.0498Z" stroke="#000000" stroke-width="2" stroke-
linecap="round" stroke-linejoin="round"></path> </g> </g></svg>More
    </a>
</div>
</div>
</div>

<div class="flex-1 overflow-y-auto p-3">
    <div class="flex flex-col space-y-2">
        <div v-for="(message, index) in messages" :key="index"
class="flex" :class="{ 'justify-end': message.isMine}">
            <div :class="{ 'bg-blue-200': message.isMine, 'bg-gray-
300': !message.isMine}" style="white-space: pre-wrap;" class="text-
black p-2 rounded-lg max-w-xs">
                {{ message.text }}
            </div>
        </div>
    </div>
</div>

<div class="bg-white p-4 flex items-center">
    <input type="text" v-model="this.inputText" placeholder="请输
```

```

入您的问题或需求..." class="flex-1 border rounded-full px-4 py-2
focus:outline-none">
    <button class="bg-blue-500 text-white rounded-full p-2 ml-2
hover:bg-blue-600 focus:outline-none" @click="sendSSEMessage">
        <svg width="20px" height="20px" viewBox="0 0 24 24"
fill="none" xmlns="http://www.w3.org/2000/svg" stroke="#ffffff"><g
id="SVGRepo_bgCarrier" stroke-width="0"></g><g
id="SVGRepo_tracerCarrier" stroke-linecap="round" stroke-
linejoin="round"></g><g id="SVGRepo_iconCarrier"> <path
d="M11.5003 12H5.41872M5.24634 12.7972L4.24158
15.7986C3.69128 17.4424 3.41613 18.2643 3.61359 18.7704C3.78506
19.21 4.15335 19.5432 4.6078 19.6701C5.13111 19.8161 5.92151
19.4604 7.50231 18.7491L17.6367 14.1886C19.1797 13.4942 19.9512
13.1471 20.1896 12.6648C20.3968 12.2458 20.3968 11.7541 20.1896
11.3351C19.9512 10.8529 19.1797 10.5057 17.6367 9.81135L7.48483
5.24303C5.90879 4.53382 5.12078 4.17921 4.59799 4.32468C4.14397
4.45101 3.77572 4.78336 3.60365 5.22209C3.40551 5.72728 3.67772
6.54741 4.22215 8.18767L5.24829 11.2793C5.34179 11.561 5.38855
11.7019 5.407 11.8459C5.42338 11.9738 5.42321 12.1032 5.40651
12.231C5.38768 12.375 5.34057 12.5157 5.24634 12.7972Z"
stroke="#ffffff" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round"></path> </g></svg>
    </button>
</div>

</div>
</template>

<script>

export default {
  // eslint-disable-next-line vue/multi-word-component-names
  name: 'Home',
  data() {
    return {
      inputText: null, // 要发送的问题
      // 对话数组
      messages: [
        { text: "你好", isMine: true },
        { text: "你好，我是咨询小哥，有什么我能帮助你的吗？", isMine:
false },
      ],
      responseFirst: true, // 对话第一次回复
      eventSource: null,
    };
  },
  beforeUnmount() {
    if (this.eventSource) {

```

```

        this.eventSource.close();
    },
    methods:{
        sendSSEMessage() {
            // 只有当eventSource不存在时才创建新的EventSource连接
            if (!this.eventSource) {
                this.messages.push({text: this.inputText, isMine: true});
                this.messages.push({text: "", isMine: false});
                // 创建新的EventSource连接
                //this.eventSource = new
EventSource('http://127.0.0.1:8080/completions?messages='+this.inputT
ext);
                this.eventSource = new
EventSource('http://127.0.0.1:8008/word/chat/main?content='+this.inpu
tText);

                // 设置消息接收的回调函数
                this.eventSource.onmessage = (event) => {
                    this.messages.text = "";
                    const data = JSON.parse(event.data);
                    console.info(data.msg);
                    const strings = data.msg.split("");
                    strings.forEach((obj,index) => {
                        setTimeout(() => {
                            this.messages[this.messages.length - 1].text += obj;
                            if (index > 0 && index % 20 === 0) {
                                this.messages[this.messages.length - 1].text +=
"\n";
                            }
                        },index*80);
                    })
                };

                // 可选：监听错误事件，以便在出现问题时能够重新连接或处理错
误
                this.eventSource.onerror = (event) => {
                    console.error("EventSource failed:", event);
                    this.eventSource.close(); // 关闭出错的连接
                    this.eventSource = null; // 重置eventSource变量，允许重建
连接
                };
            }
        }
    }
}
</script>
<style scoped>

```

</style>

...

谢谢大家听我唠叨，如果觉得有用的话，记得点赞、收藏哦！我是抢老婆酸奶的小肥仔，下次见.....

参考：小帅、[[gitee.com/xshuai/qian...](https://gitee.com/xshuai/qianfan-sse-demo)](<http://cxyroad.com/>
"https://gitee.com/xshuai/qianfan-sse-demo")

原文链接: <https://juejin.cn/post/7369791641915670537>