

```
MessageController

package com.example.restdemo.controller;

import com.example.restdemo.exceptions.NotFoundException;
import org.springframework.web.bind.annotation.*;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

@RestController
@RequestMapping("message")
public class MessageController {

    private int counter = 4;

    private List<Map<String, String>> messages = new ArrayList<Map<String, String>>() {{
        add(new HashMap<String, String>() {{
            put("id", "1");
            put("text", "First message");
        }});
        add(new HashMap<String, String>() {{
            put("id", "2");
            put("text", "Second message");
        }});
        add(new HashMap<String, String>() {{
            put("id", "3");
            put("text", "Third message");
        }});
    }};

    @GetMapping
    public List<Map<String, String>> list() {
        return messages;
    }

    @GetMapping("/{id}")
    public Map<String, String> getOne(@PathVariable String id) {
        return getMessage(id);
    }

    private Map<String, String> getMessage(@PathVariable String id) {
        return messages.stream()
            .filter(map -> map.get("id").equals(id))
            .findFirst()
            .orElseThrow(NotFoundException::new);
    }

    @PostMapping
    public Map<String, String> create(@RequestBody Map<String, String> message) {

        message.put("id", String.valueOf(counter++));

        messages.add(message);
        return message;
    }

    @PutMapping("/{id}")
    public Map<String, String> update(@PathVariable String id,
                                     @RequestBody Map<String, String> message) {

        Map<String, String> messageFromDB = getMessage(id);
        messageFromDB.putAll(message);
        messageFromDB.put("id", id);

        return messageFromDB;
    }

    @DeleteMapping("/{id}")
    public void delete(@PathVariable String id) {
        Map<String, String> message = getMessage(id);
        messages.remove(message);
    }

}
```

```
<head>
  <meta charset="UTF-8">
  <title>Project</title>
  <script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/vue-resource@1.5.1"></script>
</head>
<body>

  <div>
    <div><i>1</i>. </i>First message</div>
    <div><i>2</i>. </i>Second message</div>
    <div><i>3</i>. </i>Third message</div>
  </div>

  <script src="js/main.js"></script>
  <!--<script src="js/script.js"></script>-->

</body>
```

```
index.html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Project</title>
  <script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
  <script src="https://cdn.jsdelivr.net/npm/vue-resource@1.5.1"></script>
</head>
<body>

  <div id="app">

  </div>

  <script src="js/main.js"></script>

</body>
</html>
```

```
main.js

var messageApi = Vue.resource('/message/{id}');

Vue.component('messages-row', {
  props: ['message'],
  template: "<div><i>{{message.id}}</i>. </i>{{message.text}}</div>"
});

Vue.component('messages-list', {
  props: ['messages'],
  template: '<div>' +
    '<messages-row v-for="message in messages" :key="message.id" :message="message"/>' +
    '</div>',
  created: function () {
    messageApi.get()
      .then(result => result.json().then(data =>
        data.forEach(message => this.messages.push(message)
      )
    )
  }
});

var app = new Vue({
  el: '#app',
  template: '<messages-list :messages="messages"/>',
  data: {
    messages: [],
  }
});
```