

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);
    }

}
```

text
id 4

mainEdit.js

```
function getIndex(list, id) {
    for (let i = 0; i < list.length; i++) {
        if (list[i].id === id) {
            return i;
        }
    }
    return -1;
}

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

Vue.component('message-form', {
    props: ['messages', 'messageAttr'],
    data: function () {
        return {
            text: "",
            id: ""
        },
        watch: {
            messageAttr: function (newVal, oldVal) {
                this.text = newVal.text;
                this.id = newVal.id;
            }
        },
        template: "<div>" +
            "<input type='text' placeholder='Write something.' v-model='text'>" +
            "<input type='button' value='Save' @click='save'>" +
            "</div>",
        methods: {
            save: function () {
                let message = {text: this.text};
                if (this.id) {
                    messageApi.update({id: this.id}, message)
                        .then(result =>
                            result.json().then(data => {
                                let index = getIndex(this.messages, data.id);
                                this.messages.splice(index, 1, data);
                                this.text = "";
                                this.id = "";
                            })
                        )
                } else {
                    messageApi.save({}, message)
                        .then(result =>
                            result.json().then(data => {
                                this.messages.push(data);
                                this.text = "";
                            })
                        )
                }
            }
        }
    });

Vue.component('messages-row', {
    props: ['message', 'editMethod'],
    template: "<div>" +
        "<i>{{message.id}}</i>{{message.text}}</div>" +
        "<span id='s1'>" +
        "<input type='button' value='Edit' @click='edit'>" +
        "</span>" +
        "</div>",
    methods: {
        edit: function () {
            this.editMethod(this.message);
        }
    }
});

Vue.component('messages-list', {
    props: ['messages'],
    data: function () {
        return {
            message: null
        },
        template: '<div id="d1">' +
            '<message-form :messages="messages" :messageAttr="message"/>' +
            '<messages-row id="d2" v-for="message in messages" :key="message.id" ' +
            ':message="message" :editMethod="editMethod"/>' +
            '</div>',
        created: function () {
            messageApi.get().then(result => result.json()
                .then(
                    data => data.forEach(message => this.messages.push(message)
                )))
        },
        methods: {
            editMethod: function (message) {
                this.message = message;
            }
        }
    });

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

Write something.. Save

1). First message Edit

2). Second message Edit

3). Third message Edit

4). Fourth message Edit

```
<head>
  <meta charset="UTF-8">
  <title>Project</title>
  <link href="css/style.css" rel="stylesheet">
</head>
<body>

  <div id="d1">
    <div>
      <input placeholder="Write something.." type="text">
      <input value="Save" type="button">
    </div>
    <div id="d2"><i>1). </i>First message
      <span id="s1"><input value="Edit" type="button"></span></div>
    <div id="d2"><i>2). </i>Second message
      <span id="s1"><input value="Edit" type="button"></span></div>
    <div id="d2"><i>3). </i>Third message
      <span id="s1"><input value="Edit" type="button"></span></div>
    <div id="d2"><i>4). </i>Fourth message
      <span id="s1"><input value="Edit" type="button"></span></div>
    </div>

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

  </body>
```

Fourth message Save

1). First message Edit

2). Second message Edit

3). Third message Edit

4). Fourth message Edit

Edit Fourth message Save

1). First message Edit

2). Second message Edit

3). Third message Edit

4). Fourth message Edit

Write something.. Save

1). First message Edit

2). Second message Edit

3). Third message Edit

4). Edit Fourth message Edit

```
<head>
  <meta charset="UTF-8">
  <title>Project</title>
  <link href="css/style.css" rel="stylesheet">
</head>
<body>

  <div id="d1">
    <div><input placeholder="Write something.." type="text">
    <input value="Save" type="button">
    </div>
    <div id="d2"><i>1). </i>First message
      <span id="s1"><input value="Edit" type="button"></span></div>
    <div id="d2"><i>2). </i>Second message
      <span id="s1"><input value="Edit" type="button"></span></div>
    <div id="d2"><i>3). </i>Third message
      <span id="s1"><input value="Edit" type="button"></span></div>
    <div id="d2"><i>4). </i>Edit Fourth message
      <span id="s1"><input value="Edit" type="button"></span></div></div>

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

  </body>
```