Enterprise System Integration
(MTAT.03.229)

LECTURE 9: THICK CLIENT APPLICATIONS - VUE.JS - II

MOHAMAD GHARIB
UNIVERSITY OF TARTU
The Client side - recap
The Client side - recap

Client

GET /plants

HTTP request

Spring Boot Server

plantsRepository.findAll()

Entity

Controller

Service

Repository

HTTP response
The Client side - recap

GET /plants/id

HTTP request

plantsRepository.findById(id)

HTTP response
The Client side - recap

Client

POST /plants

Spring Boot Server

HTTP request

plantsRepository.save(plant)

HTTP response
The Client side - recap

PUT /plants/id
DELETE /plants/id

HTTP request

HTTP response

plantsRepository.save(plant)
plantsRepository.delete(id)
## The Client side - recap

### Client

<table>
<thead>
<tr>
<th>Method</th>
<th>Endpoint</th>
</tr>
</thead>
<tbody>
<tr>
<td>GET</td>
<td>plants/id/reservations</td>
</tr>
<tr>
<td>GET</td>
<td>plants/id/reservations/rid</td>
</tr>
<tr>
<td>POST</td>
<td>plants/id/reservations</td>
</tr>
<tr>
<td>PUT</td>
<td>plants/id/reservations/rid</td>
</tr>
<tr>
<td>DELETE</td>
<td>plants/id/reservations/rid</td>
</tr>
</tbody>
</table>

### Spring Boot Server

```java
reservationRepository.findByPlantId(plantId).forEach(res::add)
reservationRepository.findById(rid)
reservationRepository.save(res)
reservationRepository.save(res)
reservationRepository.deleteById(id)
```
Vue.js in nutshell - recap

Vue routes and Views
A view is supposed to contain root level component(s), where other components would be imported.

Components
Components are reusable Vue instances with custom HTML elements. Components can be reused as many times as you want or used in another component, making it a child component. Data, computed, watch, and methods can be used in a Vue component.

Methods declaration and use

Vue.js
Home | Plants | About

Header component
Body component
Side component
Footer component

Component structure
<template>
</template>
<script>
</script>

How we can use components?

Data declaration and use
Computed & Watched properties
Passing data to components (props: {})
Props type checks and other validators

Directives
A directive is a special token in the markup that tells the library to attach a specified behavior to that DOM element.

Server
Vue - practical example

Install and utilize Axios

**Axios** is a promise based HTTP client. Axios can be installed with the following command:

```
> npm install axios
```

Under **src** folder, you need to create a *.js file (http-axios.js) to utilize Axios.

The file content should include:

```javascript
import axios from "axios";
export default axios.create({
  baseURL: "http://localhost:8080/",
  headers: {
    "Content-type": "application/json"
  }
});
```
Create “Service” file

This service file will include functions/services that use axios as an HTTP client to send HTTP requests to the server.

```javascript
import http from "./http-axios";

class dataaxios{
    getAllPlants() {
        return http.get("/plants");
    }
    getPlantbyId(id) {
        return http.get("/plants/" + id);
    }
    .....  
}

export default new dataaxios();
```
Solve the CORS issue

If you are running both the front-end and back-end applications on the same computer, they will be using different ports. This cause a CORS (Cross-Origin Resource Sharing) error.

To solve this issue, you need to use `@CrossOrigin` tag to configure allowed origins.

It should be used before each `@RestController`, as follows:

```java
@CrossOrigin(origins = "http://localhost:8082")
@RestController
```
Thank You for your attention

Mohamad Gharib
mohamad.gharib@ut.ee

unitartu
tartuuniversity