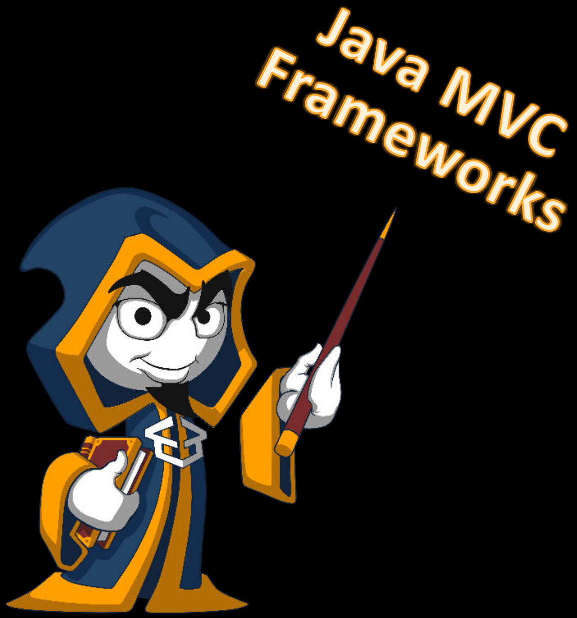
Java MVC Frameworks

AJAX, Spring Data REST

SoftUni Foundation



SoftUni Team

Technical Trainers

Software University

http://softuni.bg

Table of Contents

AJAX and JQuery

Consuming JSON

REST with Spring

Creating REST API with Spring

Questions

SoftUni Foundation

sli.do

#java-web

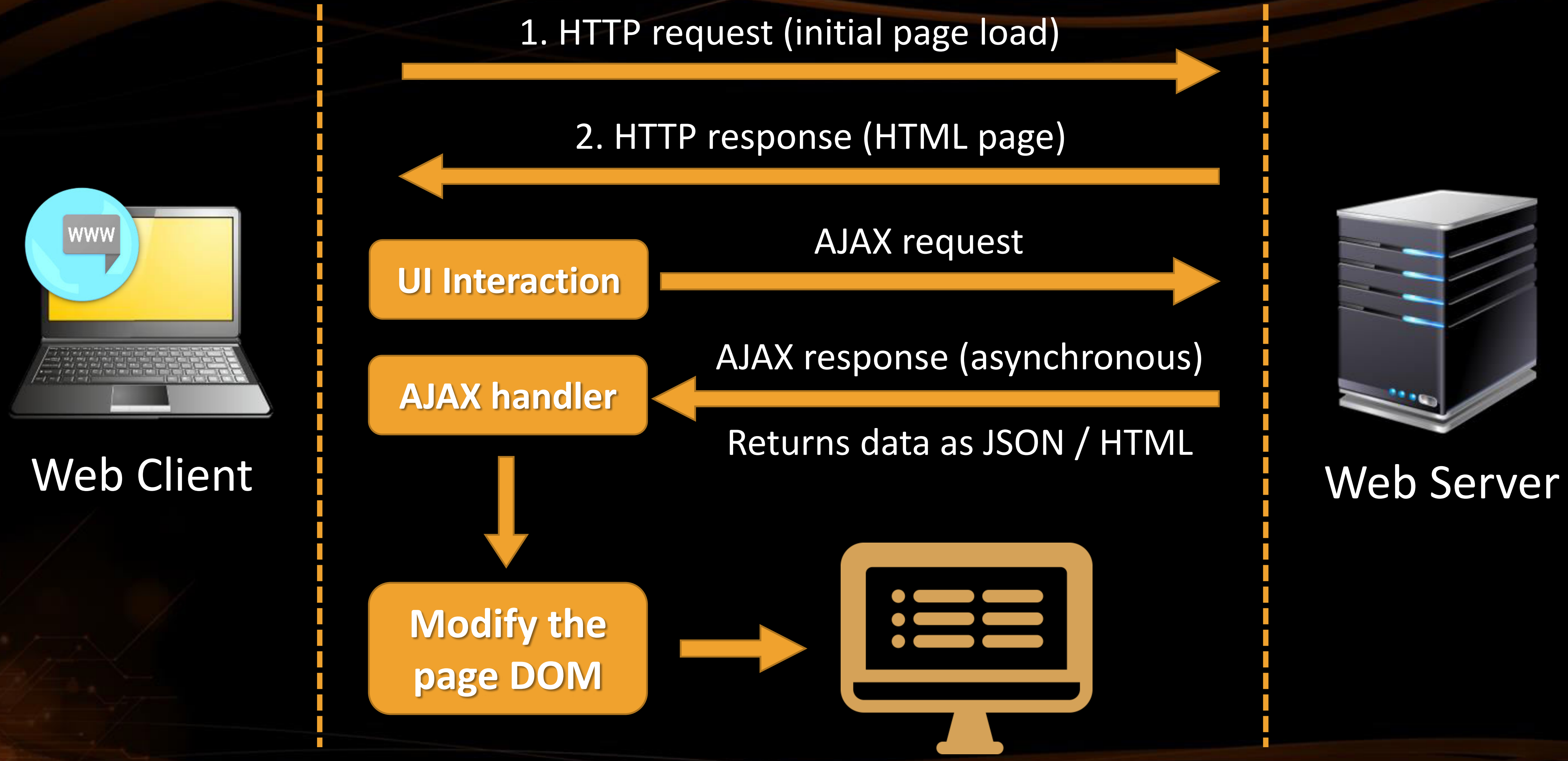
AJAX and JQuery

Consuming JSON

What is AJAX?

- AJAX == **Asynchronous JavaScript And XML**
  - Technique for background loading of dynamic content / data
  - Web pages / apps load data from the Web server and render it
- Two types of AJAX
  - **Partial page rendering**
    - Load HTML fragment + show it in a `<div>`
  - **JSON service**
    - Load JSON object and display it with JavaScript / jQuery

AJAX: Workflow



1. HTTP request (initial page load)

2. HTTP response (HTML page)

UI Interaction

AJAX request

AJAX handler

AJAX response (asynchronous)

Returns data as JSON / HTML

Modify the page DOM

Web Client

Web Server

jQuery Selectors

- Some of the jQuery selectors:
- `$("#title");` - gets the element with attribute `id="title"`
- `$(".m-0");` - gets the elements with attribute `class="m-0"`
- `$("a");` - selects all of the `<a>` tags
- The selectors return jQuery object/collection
- You can create complex selections

jQuery AJAX

- jQuery dramatically simplifies how developers make AJAX calls
- Supports GET and POST requests:
- Cross-Browser support
- Load JSON, XML, HTML or even scripts
- Simple API
- Allows parts of a page to be updated

jQuery Get and Post

- Example GET request:

```
$.get("/games/5").done(function (data) {
  alert(JSON.parse(data));
});
```

- Example POST request:

```
$.post(url, data).fail(function () {
  alert("An error occurred. Please, try again later.");
});
```

jQuery Get JSON

- Sending a request that will return JSON:

```
let url = "/games/" + id + "/info";

$.getJSON(url).done(function (info) {
  $("#title").text(info.title);
  $("#trailer").attr("src",
        "https://www.youtube.com/embed/" + info.trailer);
});
```

SoftUni Foundation

REST with Spring

Creating REST API with Spring

Response Body on MVC Controller

- Returning plain-text in MVC controller:

```java
@GetMapping("{id}/info")
@ResponseBody
public String getInfo(@PathVariable Long id){

  GameInfoView gameInfo = this.gameService.getInfoById(id);

  return new Gson().toJson(gameInfo);
}
```

Response Status

- Setting the correct Response Code

```
@GetMapping("{id}/info")
@ResponseBody
@ResponseStatus(HttpStatus.OK)
public String getInfo(@PathVariable Long id){

  GameInfoView gameInfo = this.gameService.getInfoById(id);

  return new Gson().toJson(gameInfo);
}
```

REST Controllers

- @RestController is essentially @Controller + @ResponseBody:

```
@RestController
public class OrderController {

  @GetMapping("{id}/info")
  public String getInfo(@PathVariable Long id){
    ...
  }
}
```

Response Entity

- Controlling the entire response object:

```
@GetMapping("{id}/title")
public ResponseEntity<String> getTitle(...){
  ...

  return new ResponseEntity.ok(game.getTitle());
}
```

- The **ResponseEntity<>** object allows you to change the response body, response headers and response code

Spring Data REST

- Include Maven Dependency

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-data-rest</artifactId>
</dependency>
```

- Spring Data REST scans your project and provides REST API for your application using HAL as media type

Configuring Repositories

- You can configure repository settings using the **@RepositoryRestResource** annotation:

```
@RepositoryRestResource(path = "gameIssues")
public interface IssueRepository extends
                              JpaRepository<Issue, Long> {
    Issue getById(@Param("id") Long id);
    List<Issue> getAllByOrderByDateDesc();
}
```

Summary

- AJAX sends asynchronous HTTP requests from JS
- jQuery simplifies how developers use AJAX
- Actions in controllers can return plain text via **@ResponseBody**
- **@RestController** is used when all actions will return JSON or XML.
- Spring Data REST uses Spring HATEOAS to automatically expose resources for entities managed by Spring Data repositories

Java MVC Frameworks – Web API

SoftUni Foundation

Questions?

xssoftware

SmartIT

NETPEAK
SEO and PPC for Business

SUPERHOSTING®.BG

INDEAVR
Serving the high achievers

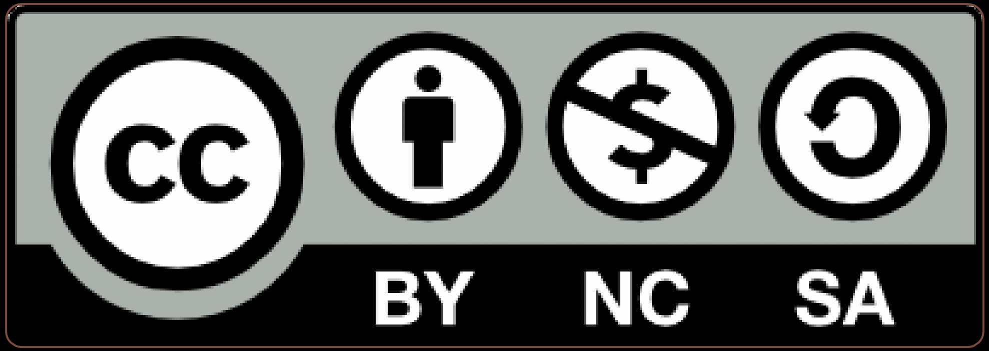
telenor

SOFTWARE GROUP

INFRAGISTICS™
DESIGN / DEVELOP / EXPERIENCE

https://softuni.bg/courses/java-mvc-frameworks-spring

License



Trainings @ Software University (SoftUni)

- Software University – High-Quality Education, Profession and Job for Software Developers
  - softuni.bg
- Software University Foundation
  - http://softuni.foundation/
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg

Software University

SoftUni Foundation

FORUM