

Министерство науки и высшего образования РФ
Государственное образовательное учреждение высшего
профессионального образования
Белгородский Государственный Технологический Университет им
В. Г. Шухова

Кафедра программного обеспечения вычислительной техники и
автоматизированных систем.

Лабораторная работа №3 по дисциплине
«Технологии web-программирования»

Серверное программирование.

Выполнил:
студент группы ПВ-42
Волобуев И.К.
Проверил:
Картамышев С.В.

Белгород 2020 г.

Цель: познакомиться с основами backend разработки web-приложений.

Выполнение.

В ходе выполнения лабораторной работы был разработан контроллер для контентной страницы web-приложения с использованием паттерна DTO. DTO (data transfer object) – это объекты, которые содержат в себе данные отправляемые на фронт.

```
@Data
@FieldDefaults(level = AccessLevel.PRIVATE)
public class MovieResponse {
    Long id;
    String name;
    String poster;
    GregorianCalendar release;
    List<CountryResponse> country;
    List<GenreResponse> genre;
    List<PersonResponse> director;
    List<PersonResponse> writer;
    List<PersonResponse> producer;
    List<PersonResponse> operator;
    List<PersonResponse> compositor;
    List<PersonResponse> actor;
    Long budget;
    Long box_office;
    String duration;
    String synopsis;
    Double rating;
    String trailer;
}
```

```
@Data
@FieldDefaults(level = AccessLevel.PRIVATE)
public class IdNameDTO {
    Long id;
    String name;
}
```

На данном этапе контроллер содержит в себе только обработку GET запроса. То есть, пользователь, перейдя на страницу с конкретным id фильма получает информацию о нем (на данном этапе только статическую).

```
@RestController
@RequestMapping("/movie")
public class ContentPageController {

    @GetMapping("/{id}")
    MovieResponse getMovie(@PathVariable Long id) {
        MovieResponse movieResponse = new MovieResponse();

        movieResponse.setId(1L);
        movieResponse.setName("TENET");
        movieResponse.setPoster("https://images-na.ssl-images-
amazon.com/images/I/71W2aEcrxxL._AC_SL1334_.jpg");

        movieResponse.setRelease(new GregorianCalendar(2020,
Calendar.SEPTEMBER, 3));
        List<CountryResponse> countryResponses = new ArrayList<>();
    }
}
```

```
CountryResponse countryGreatBritain = new CountryResponse();
countryGreatBritain.setId(1L);
countryGreatBritain.setName("Great Britain");
countryResponses.add(countryGreatBritain);
CountryResponse countryUSA = new CountryResponse();
countryUSA.setId(2L);
countryUSA.setName("USA");
countryResponses.add(countryUSA);
movieResponse.setCountry(countryResponses);
```

```
List<GenreResponse> genreResponses = new ArrayList<>();
GenreResponse genreFantasy = new GenreResponse();
genreFantasy.setId(1L);
genreFantasy.setName("Fantasy");
genreResponses.add(genreFantasy);
GenreResponse genreAction = new GenreResponse();
genreAction.setId(2L);
genreAction.setName("Action");
genreResponses.add(genreAction);
movieResponse.setGenre(genreResponses);
```

```
List<PersonResponse> directorResponses = new ArrayList<>();
PersonResponse personChristopherNolan = new PersonResponse();
personChristopherNolan.setId(1L);
personChristopherNolan.setName("Christopher Nolan");
directorResponses.add(personChristopherNolan);
movieResponse.setDirector(directorResponses);
```

```
List<PersonResponse> writerResponses = new ArrayList<>();
writerResponses.add(personChristopherNolan);
movieResponse.setWriter(writerResponses);
```

```
List<PersonResponse> producerResponses = new ArrayList<>();
PersonResponse personIvoFelt = new PersonResponse();
personIvoFelt.setId(2L);
personIvoFelt.setName("Ivo Felt");
PersonResponse personThomasHayslip = new PersonResponse();
personThomasHayslip.setId(3L);
personThomasHayslip.setName("Thomas Hayslip");
producerResponses.add(personChristopherNolan);
producerResponses.add(personIvoFelt);
producerResponses.add(personThomasHayslip);
movieResponse.setProducer(producerResponses);
```

```
List<PersonResponse> operatorResponses = new ArrayList<>();
PersonResponse personHoyteVanHoytema = new PersonResponse();
personHoyteVanHoytema.setId(4L);
personHoyteVanHoytema.setName("Hoyte Van Hoytema");
operatorResponses.add(personHoyteVanHoytema);
movieResponse.setOperator(operatorResponses);
```

```
List<PersonResponse> compositorResponses = new ArrayList<>();
PersonResponse personLudwigGorannson = new PersonResponse();
personLudwigGorannson.setId(5L);
personLudwigGorannson.setName("Ludwig Gorannson");
compositorResponses.add(personLudwigGorannson);
movieResponse.setCompositor(compositorResponses);
```

```
List<PersonResponse> actorResponses = new ArrayList<>();
PersonResponse personJohnDavidWashington = new PersonResponse();
personJohnDavidWashington.setId(6L);
personJohnDavidWashington.setName("John David Washington");
PersonResponse personRobertPattinson = new PersonResponse();
personRobertPattinson.setId(7L);
```

```

        personRobertPattinson.setName("Robert Pattinson");
        PersonResponse personElizabethDebicki = new PersonResponse();
        personElizabethDebicki.setId(8L);
        personElizabethDebicki.setName("Elizabeth Debicki");
        actorResponses.add(personJohnDavidWashington);
        actorResponses.add(personRobertPattinson);
        actorResponses.add(personElizabethDebicki);
        movieResponse.setActor(actorResponses);

        movieResponse.setBudget(205000000L);
        movieResponse.setBox_office(300000000L);
        movieResponse.setDuration("2:30:00");
        movieResponse.setSynopsis("Armed with only one word—Tenet—and
fighting for the survival of the entire" +
        " world, the Protagonist journeys through a twilight world
of international espionage on a mission" +
        " that will unfold in something beyond real time.\n" +
        " Not time travel. Inversion.");
        movieResponse.setRating(8.5);

    movieResponse.setTrailer("https://www.youtube.com/embed/L3pk_TBkihU");
    return movieResponse;
}
}

```

Результат работы программы: JSON передаваемый на клиентскую часть

```

{
  "id": 1,
  "name": "TENET",
  "poster": "https://images-na.ssl-images-amazon.com/images/I/71W2aEcrxL_AC_SL1334_.jpg",
  "release": "2020-09-02T21:00:00.000+00:00",
  "country": [
    {
      "id": 1,
      "name": "Great Britain"
    },
    {
      "id": 2,
      "name": "USA"
    }
  ],
  "genre": [
    {
      "id": 1,
      "name": "Fantasy"
    },
    {
      "id": 2,
      "name": "Action"
    }
  ],
  "director": [
    {
      "id": 1,
      "name": "Christopher Nolan"
    }
  ]
}

```

```
    }
  ],
  "writer": [
    {
      "id": 1,
      "name": "Christopher Nolan"
    }
  ],
  "producer": [
    {
      "id": 1,
      "name": "Christopher Nolan"
    },
    {
      "id": 2,
      "name": "Ivo Felt"
    },
    {
      "id": 3,
      "name": "Thomas Hayslip"
    }
  ],
  "operator": [
    {
      "id": 4,
      "name": "Hoyte Van Hoytema"
    }
  ],
  "compositor": [
    {
      "id": 5,
      "name": "Ludwig Gorannson"
    }
  ],
  "actor": [
    {
      "id": 6,
      "name": "John David Washington"
    },
    {
      "id": 7,
      "name": "Robert Pattinson"
    },
    {
      "id": 8,
      "name": "Elizabeth Debicki"
    }
  ],
  "budget": 205000000,
  "box_office": 300000000,
  "duration": "2:30:00",
```

"synopsis": "Armed with only one word—Tenet—and fighting for the survival of the entire world, the Protagonist journeys through a twilight world of international espionage on a mission that will unfold in something beyond real time.\n Not time travel. Inversion.",

"rating": 8.5,

"trailer": "https://www.youtube.com/embed/L3pk_TBkihU"

}