

# Java Backend Assessment



## Introduction

This is the Sogeti Back-end developer assessment. The assessment is intended to give the reviewer insight into your technical abilities, development approach and general technical working habits. We view your performance on this assessment as indicative of the work you will deliver as a Sogeti Back-end developer.

The assessment can be made privately and is to be later discussed during an interview. We have prepared a business case, further described below. Read the case carefully and approach it as if it was a real project. We expect that you will need about 5 to 8 hours to finish the assessment. For the assessment discussion, please bring a laptop with a working development environment so that we can look at parts of the application together. Also, we expect you to provide the source code and some documentation before the interview.

The assignment should be implemented in Java. You can choose between JEE and Spring. In case of JEE, the application needs to be runnable in a current Wildfly version. The data needs to be saved in an in-memory database, e.g. H2. You have to adhere to current clean-code and secure coding principles.

Make sure to prepare yourself to tell something about the following subjects:

- the overall approach you took to implement the assignment
- the architecture of the solution
- the technologies used for your solution for each of the user stories.

Good luck with the assignment!

# The assessment

Filmland is a company that offers a streaming service for movies and tv-series. Customers pay a monthly subscription fee to access all the content available on Filmland. However, due to the fierce competition in the industry, Filmland is seeing a rapid decline in customer numbers. Therefore, Filmland has decided to set up a new subscription structure and have asked you to implement it into their back-end application.

In the new subscription structure, Filmland's content is divided into three separate categories: 'Dutch Series', 'Dutch Films' and 'International Films'. Each category has its own price and customers can subscribe to one or more categories. The new subscription structure also imposes a monthly limit: only a certain number of films or series can be watched each month. If a subscriber has any remaining content at the end of the month, this does **not** carry over to the next month.

Please note that everyone who has a login is a customer and a subscriber is someone who also has an active subscription.

## 1. Customer can log in

- **Precondition:** Login credentials are stored in the database (not hashed).
- **HTTP Method:** POST
- **Request:**

```
{
  "email": "info@filmland-assessment.nl",
  "password": "Javaiscool90"
}
```

- **Action:** Check if the login is valid.
- **Response:**

```
{
  "status": "Login successful/Login failed",
  "message": "make a nice message"
}
```

- **BONUS:** The password is hashed into the database.
- **BONUS:** Make an authorisation key and use this key with all requests to the server.

## 2. Getting categories

- **HTTP Method:** GET
- **Prerequisite:** The user is authenticated. Only the username needs to be sent to the back-end.
- **Action:** Getting all the available categories and the subscribed categories of the authenticated subscriber.
- **Response:**

```
{
  "availableCategories": [
    {
      "name": "Dutch Films",
      "availableContent": 10,
      "price": 4.0
    },
    {
      "name": "Dutch Series",
      "availableContent": 20,
      "price": 6.0
    }
  ],
  "subscribedCategories": [
    {
      "name": "International Films",
      "remainingContent": 5,
      "price": 8.0,
      "startDate": "01-01-2018"
    }
  ]
}
```

### 3. Customer can subscribe to a category

- **HTTP Method:** POST
- **Request:**

```
{
  "email": "info@filmland-assessment.nl",
  "availableCategory": "Dutch Films"
}
```

- **Action:** The first month is free; after this month, the subscriber needs to pay for the service. You may add a payment date to the database. Make some logical business rules, e.g. a subscriber cannot subscribe to a service he/she is already subscribed to.
- **Response:**

```
{
  "status": "Login successful/Login failed",
  "message": "make a nice message"
}
```

- **BONUS:** Instead of using the payment date in the database, solve the problem by using a scheduler.

### 4. Subscriber can share a category

- **HTTP Method:** POST
- **Request:**

```
{
  "email": "info@filmland-assessment.nl",
  "customer": "client@filmland-assessment.nl",
  "subscribedCategory": "Dutch Films"
}
```

- **Action:** It is possible to share a subscription with an existing customer. This will start as soon as this option is selected. The costs and the remaining content will need to be shared among both customers.
- **Response:**

```
{
  "status": "Login successful/Login failed",
  "message": "make a nice message"
}
```

## About Sogeti

---

Sogeti is a leading provider of technology and engineering services. Sogeti delivers solutions that enable digital transformation and offers cutting-edge expertise in Cloud, Cybersecurity, Digital Manufacturing, Digital Assurance & Testing, and emerging technologies. Sogeti combines agility and speed of implementation with strong technology supplier partnerships, world class methodologies and its global delivery model, Rightshore®. Sogeti brings together more than 25,000 professionals in 15 countries, based in over 100 locations in Europe, USA and India. Sogeti is a wholly-owned subsidiary of Capgemini SE, listed on the Paris Stock Exchange.

Learn more about us at [www.sogeti.com](https://www.sogeti.com)