Armedia Hackathon

Objectives

SUMMARY:

The overall objective of this exercise is to determine your ability to work in the IT industry. This exercise is split into three objectives:

- **Objective 1:** Ability to setup development environments using open source technologies to understand the variety of tools that can be used. We have provided one sample of what you could do using a tool called JHipster creating an integrated development environment (IDE).

- **Objective 2:** Test your ability to interface with services using RESTful interface technologies.

- **Objective 3:** Test your ability to design and present a system which is user friendly.

OBJECTIVE #1:

Installing Jhipster:

Pre-requisites: Java 8, Node.js, Yarn

- Install Java 8 from Oracle
- Install Node.js from http://nodejs.org
- Install Yarn from https://yarnpkg.com/en/docs/install

To install run:

1. yarn global add bower
2. yarn global add gulp-cli
3. yarn global add generator-jhipster

Creating the project using Jhipster:

1. From command line: mkdir **hackathon**, cd **hackathon**, **jhipster**
2. Q&A:
   a. Which type of application would you like to create?
      - Monolithic application
   b. What is the base name of your application?
      - hackathon
   c. What is your default Java package name?
      - com.armedia.hackathon
   d. Do you want to use the JHipster Registry to configure, monitor and scale your application?
      - Y
   e. Which type of database would you like to use?
      - An SQL database
   f. Which production database would you like to use?
      - MySQL
   g. Which development database would you like to use?
      - H2 with its data stored on disk
   h. Do you want to use Hibernate 2nd level cache?
      - No
   i. Would you like to use Maven or Gradle?

- Maven

j. Which other technologies would you like to use?
   - None (skip all) - Press Enter to skip all

k. Which Framework would you like to use for the client?
   - Pick your choice

l. Would you like to use the LibSass stylesheet preprocessor for your CSS?
   - N

m. Would you like to enable internationalization support?
   - N

n. Which testing frameworks would you like to use?
   - Select all using <a> key.

o. Would you like to install other generators from the JHipster Marketplace?
   - N

3. ./mvnw to build the snapshot of the configured Spring Boot Application
4. From your favorite web browser open http://localhost:8080 to access admin panel for Jhipster after successful build.

OBJECTIVE #2:

ArkCase is a tool developed by Armedia that provides several REST endpoints to retrieve data in JSON format. These data can be related to a case file, task, file or folder object among many others. You will be connecting to an instance of ArkCase to retrieve a specific set of data, and present it in Objective 3.

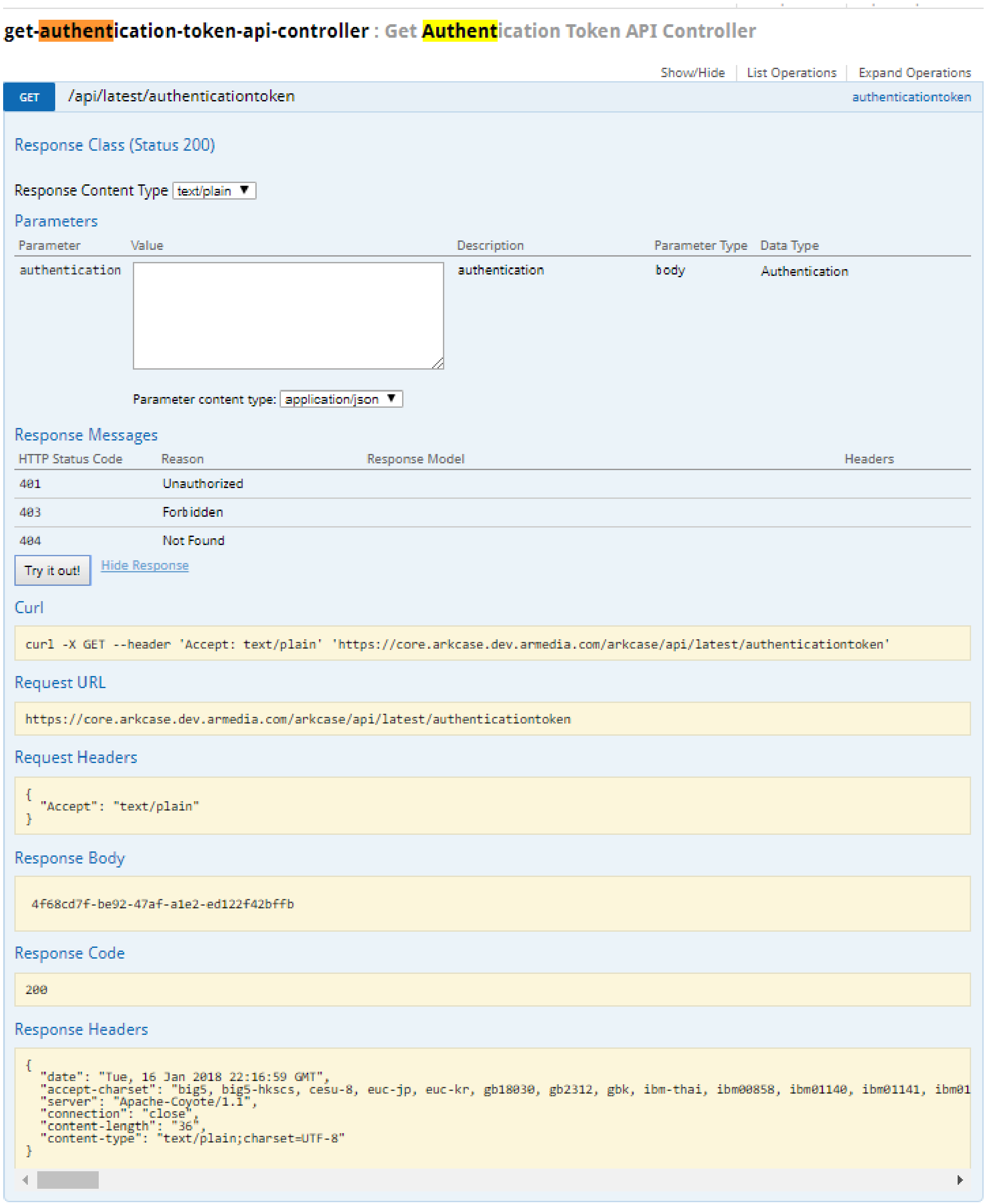
ArkCase provides various authentication methods like basic, token-based and multi-factor etc. You will be using a combination of the first two. In order to get authenticated to ArkCase, you will be using one of the APIs to generate an authentication token. You will be passing a basic authentication header to this REST end-point that will generate a token which can be used for any subsequent REST calls. Once the user has received the token, it is not necessary to pass the basic authentication header anymore when making REST calls to ArkCase. The token will expire after 30 minutes of inactivity and will have to renewed by following the process mentioned above.

You will use Swagger to look up the required API to connect to ArkCase.

To get to ArkCase Swagger interface, copy and paste the following link to your browser:

https://core.arkcase.dev.armedia.com/arkcase/swagger-ui.html

Enter the ArkCase login credentials and then look for interface called: **"Get Authentication Token API Controller"**. You can explore this interface to get an idea about the API end-point. There is no need to provide authentication and session fields values in ArkCase Swagger pages as they are set by Spring by default. Enter the rest of the parameter combinations and click **"Try it out!"**.



get-authentication-token-api-controller : Get Authentication Token API Controller
Show/Hide
List Operations
Expand Operations
GET
/api/latest/authenticationtoken
authenticationtoken

Response Class (Status 200)

Response Content Type text/plain ▼

Parameters

Parameter	Value	Description	Parameter Type	Data Type
authentication		authentication	body	Authentication

Parameter content type: application/json ▼

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Unauthorized		
403	Forbidden		
404	Not Found		

Try it out!
Hide Response

Curl

```
curl -X GET --header 'Accept: text/plain' 'https://core.arkcase.dev.armedia.com/arkcase/api/latest/authenticationtoken'
```

Request URL

```
https://core.arkcase.dev.armedia.com/arkcase/api/latest/authenticationtoken
```

Request Headers

```
{
  "Accept": "text/plain"
}
```

Response Body

```
 4f68cd7f-be92-47af-a1e2-ed122f42bffb
```

Response Code

```
200
```

Response Headers

```
{
  "date": "Tue, 16 Jan 2018 22:16:59 GMT",
  "accept-charset": "big5, big5-hkscs, cesu-8, euc-jp, euc-kr, gb18030, gb2312, gbk, ibm-thai, ibm00858, ibm01140, ibm01141, ibm01
  "server": "Apache-Coyote/1.1",
  "connection": "close",
  "content-length": "36",
  "content-type": "text/plain;charset=UTF-8"
}
```

Once you have a token, you will need to append the token to the end of the request URLs e.g.

https://cloud.arkcase.dev.armedia.com/arkcase/api/latest/plugin/search/advancedSearch?acm_ticket=<$token>

This token is required to further access ArkCase APIs for Objective 3.

The following is an example of using the Advanced Search API using the information you have already acquired.

In the Swagger UI, find **“Advanced Search API Controller”**.

ArkCase Advanced Search API is a wrapper around standard SOLR APIs. You can formulate SOLR queries to retrieve objects as below:

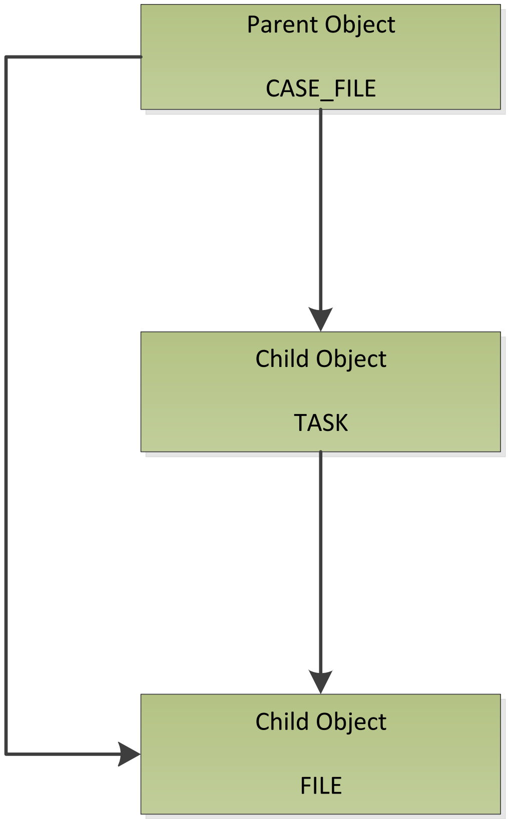
q=object_type_s:CASE_FILE

q=object_type_s:TASK

q=object_type_s:FILE

Child objects like FIlE and TASK will have a reference to the parent object which is maintained in the “parent_ref_s” field in the format “ID-OBJECT_TYPE” where object types could be **CASE_FILE**, **TASK** or **FILE**. In order to retrieve child objects for a parent object, you can formulate SOLR queries as below:

q=object_type_s:FILE AND parent_ref_s:101-CASE_FILE



Object types in ArkCase are case-sensitive i.e. **case_file** or **task** are **invalid** but **CASE_FILE** or **TASK** are **valid**.

All child objects are linked to parent objects by **“parent_ref_s”** field. E.g.

"parent_ref_s":"135-CASE_FILE"

A child object is related to a CASE_FILE object with ID 135.

You can explore various combinations of queries by using the fields in the response JSON.

Case File Sample JSON

Task Sample JSON

File Sample JSON

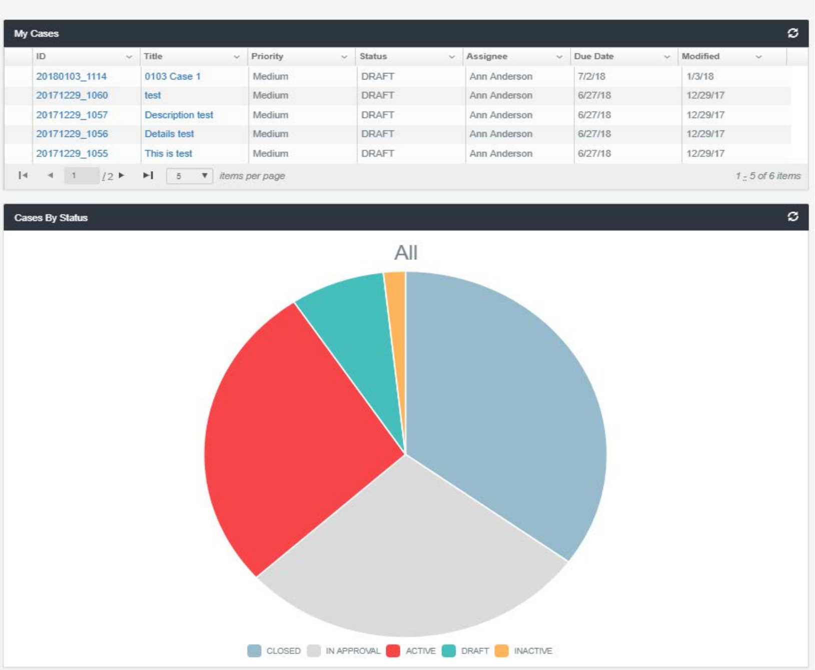
OBJECTIVE #3:

The purpose of Objective 3 is to test your creativity in using the data you have acquired and presenting it to users who are not as tech savvy as you are.

The sample at the right is an example of the data and data visualization.

You are now free to show your creativity and utilize the data in any manner you deem suitable using any technology for user presentation.

Expectation: You will be required to present your UI at the end to this audience. Please have a strategy in mind to deliver your design in a **6-minute**s presentation.



My Cases

ID	Title	Priority	Status	Assignee	Due Date	Modified
20180103_1114	0103 Case 1	Medium	DRAFT	Ann Anderson	7/2/18	1/3/18
20171229_1060	test	Medium	DRAFT	Ann Anderson	6/27/18	12/29/17
20171229_1057	Description test	Medium	DRAFT	Ann Anderson	6/27/18	12/29/17
20171229_1056	Details test	Medium	DRAFT	Ann Anderson	6/27/18	12/29/17
20171229_1055	This is test	Medium	DRAFT	Ann Anderson	6/27/18	12/29/17

1 / 2 5 items per page 1 - 5 of 6 items

Cases By Status