

IADI 2015 - Test 1

1:30h

January 1, 2016

Unless explicitly written, to answer the following questions, you should consider a very simplified version of Google Drive where you can have folders and files, and the natural relationship between them. For simplicity, you should consider that all files are of the same type: text files. Consider that, for optimization reasons, a folder must know its “parent” folder.

You should answer using Java code under the Spring Boot framework, or IFML/WebRatio notation, whenever applicable.

Answer each group in a different sheet of paper (use as many as you need). Identity each sheet you want to be evaluated with your name and student number.

Group 1

1 Define the model(s) of the scenario described above (simplified Google Drive).

2 (In this question you shouldn't consider the Google Drive example.) Given the following models, define a possible relational model that could be used to save the corresponding data.

```
public class A {
    @Id
    private String a;
    @OneToOne(mappedBy="a")
    private B b;
    @ManyToMany
    private Collection<A> as;
}
public class B {
    @Id
    private int b;
    private A a; }
```

Group 2

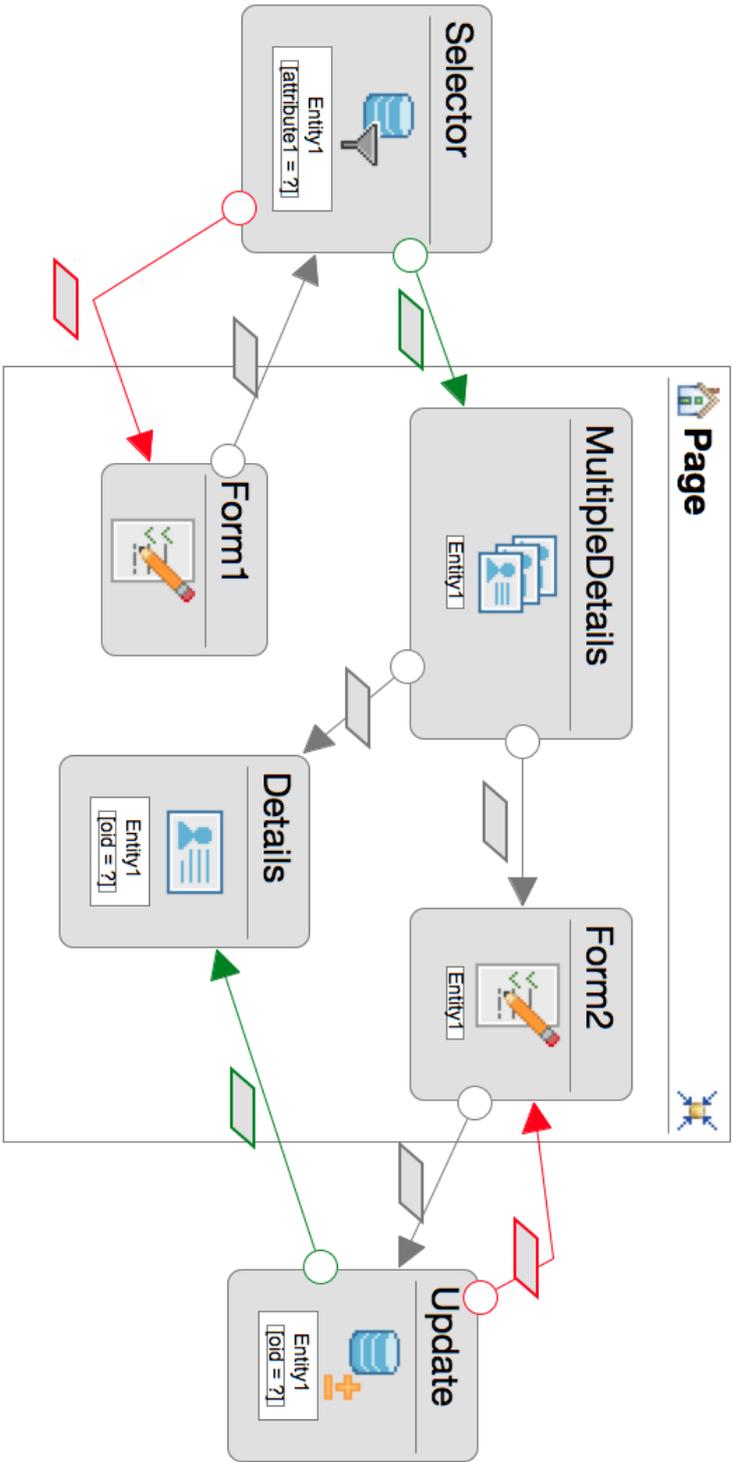
3 Create the IFML/WebRatio specification for the use case where a new file is created. There are two possibilities to create a file:

1. inside a specific folder, naturally being inside it, or
2. outside a folder, but after its creation it must be immediately connected to the “parent” folder.

Besides the specification, you should describe it as well as you can.

4 (In this question you should not consider the Google Drive example.) Consider the following IFML/WebRatio specification, page 3, and its details, page 4 (the test continues in page 5):

1. Name and describe with your own words the view(s) necessary to implement such specification. You should do this using the names shown in the specification and in its details, whenever necessary/possible, to make the description of the view clear.
2. Implement the necessary controllers to fully realize the specification. Use the names defined in the previous question to define the controllers. You can consider there is a repository of `Entity1` called `rep`. Don't forget to define which controller implement which part of the specification.



- ▼  Page
 - ▼  Details
 - ▼  Conditional Expression
 -  KeyCondition8
 - ▼  Form2
 -  Field1
 - ▼ → Flow14
 -  Entity Object_Entity Object
 -  Field1_attribute1
 -  oid_PASSING_KeyCondition7 [oid]
 - ▼  Form1
 -  Field11
 - ▼ → Flow17
 -  Field11_AttributeCondition1
 - ▼  MultipleDetails
 - ▼ → Flow13
 -  attribute1_PASSING
 -  oid_PASSING
 - ▼ → Flow15
 -  oid_KeyCondition8 [oid]
- ▼  Selector
 - ▼  Conditional Expression
 -  AttributeCondition1
 - ▼ → OKFlow9
 -  attribute1_PASSING
 -  oid_PASSING
 - ▼ → KOFlow7
 -  attribute1_PASSING
- ▼  Update
 - ▼  Conditional Expression
 -  KeyCondition7
 - ▼ → OKFlow8
 -  oid_KeyCondition8 [oid]
 - ▼ → KOFlow6
 -  oid_Key Condition [oid]

Group 3

5 (In this question you should not consider the Google Drive example.) Define the complete REST mapping for the application defined in the IFML/WebRatio specification given in question 4. Although not given in the specification, consider it is also possible to delete and add entries of the entity **Entity1**. You should define the URL, the request method, and the purpose (e.g. URL: /sample/exam, Method: Get, Purpose: show an exam).

6 Consider you have a repository of folders, called **folders**, and a repository of files called **files**.

1. Write the methods' signature to find/select folders and files by a given name.
2. Write the methods' signature to find/select all the files of a given folder parent name.

Note you can also add annotations to the method signature if you think they are necessary.

Group 4

7 Explain why should a developer use frameworks to develop web applications and not simply programming languages.

8 Are web service applications and RESTful applications the same thing? If so, why? If not, when and why should we use one or the other.