INFO323
Distributed Information Systems
Course Project

1 Deliverables

Phase 2 – 5:00 pm, Friday 10 May. The router documentation is worth 5%, and the composite service and AJAX client is worth 15%.

2 Phase 2

2.1 Composite Service

- Create the composite service that you described in your phase 1 documentation. The composite service should be implemented using Apache Camel and must make use of the two services created in phase 1.
- The Camel router should be exposed as a web-based service using a WebSocket as shown in lab 9.

2.2 AJAX Client

- Create an AJAX client that sends user input to the Camel router via a WebSocket as done in lab 9.

  If it makes sense to do so; the AJAX client should pull its data from the RESTful service from phase 1 as done in labs 5 and lab 9. If the data that the AJAX client needs to display to the user can not be supplied by the REST service you should hard code it in the JavaScript as shown in lab 5.

  The AJAX client should make use of AngularJS for UI binding and interacting with the RESTful service as done in labs 5 and 9.

  There is no need to create a full AJAX RESTful client that sends POST, PUT and DELETE requests as done in lab 5. The AJAX client should just do what it needs to do to provide a workable user interface for submitting requests to the composite service, and no more than that.
- Feedback should be sent back to the AJAX client via a WebSocket and displayed to the user as done in lab 9. You are free to decide how to display the feedback to the user. The feedback could be the result of the composite service's processing, or could be some kind of confirmation to the user that the composite service has handled their request.

2.3 Router Documentation

Explain the operation of your Camel router. For each endpoint in your router you should describe what data enters the endpoint, and what each route does to the data before it is moved to the next endpoint(s). To clarify; an 'endpoint' is defined in Camel by either a `from`, `to`, or `recipientList` method call.

Your answer should include:

- What data format is the data in as it enters each endpoint.
- What does the data represent as it enters each endpoint. For example; does it represent an order, or an order item, or a book?
- What transformations, modifications, or additions are made to the data by the route before it is sent to the next endpoint.
- What networking protocols are used to communicate with the service endpoints? We consider the following to be service endpoints:
  - Java RMI services.
  - SOAP/RPC services.
  - RESTful services.
  - WebSocket endpoints. Both incoming and outgoing.

  We have explained at length what protocols are used for communicating with web services. We will leave it to you as a research exercise to discover the protocols that WebSockets and Java RMI use.

2.4 Expectations for submitted work

Your work should be your own. It is fine to use the lab work and projects that we have given you as guides for creating your own work, but you should be using them as guides only – not basing your work completely from them.

Avoid doing the following:

- Submitting lab work with only minor modifications. You are expected to put a reasonable amount of effort into coming up with your own solution.
- Submitting projects that we have given you with only minor modifications. This is plagiarism, and will be reported to the Head of Department in line with the University's Dishonest Practice Procedures.
- We shouldn't have to tell you this, but submitting work that comes from other students (either students in your class or students from previous years), or copied from other sources without attribution is also plagiarism.

Since you have limited experience with HTML, CSS, and JavaScript we don't mind if you base your work on the HTML, CSS, and JavaScript files that we have given you. You should however make a reasonable effort to modify the HTML and JavaScript files to suit the requirements of your system. You are free to use our CSS without modification.

2.5 Phase 2 Submission

We will be using the INFO323 Wiki to submit the project.

Click the 'Project Submission' link in the navigation menu, and then click 'Phase 2'. You should see a page with your user name as the title. This is your personal project submission page. Click your user name to open the page.

Add your router documentation directly to the page by clicking the 'Edit Page' button and adding your content via the Wiki editor. Leave your user name as the title of the page.

You should submit ALL of your projects related to the course project. This includes the service projects that you created in phase 1. Chances are you will have had to make some changes to the phase 1 projects so we will need the updated versions. Even if you have made no changes to the phase 1 projects you should submit them anyway so that we have all of your projects in a single location ready for marking.

Prepare your projects for submission by right clicking each project in the NetBeans project view and selecting *Clean*.

Zip up your NetBeans projects, and attach them to the page by clicking the 'attach file or image' button.