

SOA Module Assignment

Introduction

The assignment is to design a Service Oriented Architecture and approach for a given domain. You must show a good understanding of Service Oriented principles. In addition you must show knowledge and understanding of specific SOA techniques, practices and approaches in the design.

Domain – De-centralised Education

The world of education is rapidly becoming deconstructed and dis-intermediated by the Internet. Many prestigious universities including MIT are offering “Open Courseware” where students on the Internet can freely access teaching materials. Students may elect to take this (and other modules in the Software Engineering Programme) as one-off courses. Education – in England at least - is also changing from being a publicly provided offering to a paid-for service which will in the long run lead to much greater demands from students for greater choice and increased service. There is also a move towards long-term education, life-long learning and more part-time education.

A possible outcome in this evolution might well be that students start to take a “best of breed” approach to their education – choosing the best establishment to provide a given course. In this potential scenario, there would need to be a system whereby students could manage their qualifications, and where the college, university or school awarding a final degree, diploma, etc could contact other establishments to validate that students had passed their courses. In addition there would need to be some way of checking the level of a given course meets the criteria required to gain the diploma, degree, etc. And educational establishments – schools, universities, colleges and training establishments would need a way of making themselves known to the system.

This is a large domain problem with multiple solutions – there could be a single organization providing a system or there could be a completely federated model. Issues of security, identity and reputation are key in this model, and open APIs spring to mind as being an essential aspect.

Some aspects of this problem are a reality today – and in progress with an SOA solution – such as the need for universities to have a consistent way to check qualifications issued by other establishments.

Tasks

1. Schemas and message formats

A good Service Oriented Architecture has a clear set of message formats. For this assignment, document at least some of the relevant message formats and/or media types that your solution will use. These may be defined in XML Schema, as an XML infoset definition, UML, informal XML or JSON schema (as used in many standards) or using other schema approaches (such as JSON Schema). Discuss any decisions you had to

make (including your choice of description language) and why you chose your approach, as well as outlining what other formats you might define.

2. Services

Create some well-documented clear service definitions. These may use WSDL, WADL, Swagger or another approach, but must clearly convey how to use the service such that a third-party with domain knowledge should be able to construct an application that interacts with that service. You do not need to create descriptions for all the services, but you do need to create at least one complete service description. Ensure that there is a human readable presentation (e.g. a graphical view, or clear document) as well as the XML.

3. Service interactions

Capture some service interaction flows. You may use BPEL, BPMN, EIP, ESB flows or sequence diagrams to indicate how different services interact and if there are workflows, ESB flows, etc., how those orchestrate and mediate the services.

4. Security and other non-functional aspects

How is the system secured? Which services need security and what is the most appropriate technology for that? What directories, token servers, key management, etc does the solution require? What other non-functional aspects are important? Does the solution use policies and if so what is described in those policies?

5. SOA design and concepts

Did you choose a WS-* or a RESTful approach, or a mix of the two? Did you utilize asynchronous flow or event distribution? Is there an ESB, API management system, a registry, or a business process manager in your solution? How did you decide the granularity of your services? Are there components that are not available as services and how did you decide which components to expose and which to hide? What other SOA concepts such as compensation did you apply?

Deliverables

- Write a document of between 3000 and 5000 words, plus diagrams and code samples, outlining your design and architecture. Do not submit a document longer than 5000 words.
- You are not expected to completely implement a system! A real-life solution is out-of-scope.
- Clearly document any assumptions you make.

- You do not need to explore every aspect or every service. However, some well-chosen examples, explored to a good level of detail will demonstrate the understanding you have.
- You (and the examiner) must be confident that there are no major flaws in the design and that it is implementable.
- Prototyping some parts of the system in a common toolkit such as JAX-RS, JAX-WS or .NET is highly recommended.

Assessment Criteria

Assessment will be according to the following criteria:

- Have you understood the principles and design characteristics of a service-oriented architecture? By undertaking a design activity, can you show that you appreciate the strengths and weaknesses of the approach?
- Can you implement and deploy simple services using a development platform?
- Are you able to define and design applications as combinations of services, and be able to discuss the emergent properties of those composite services?
- Do you understand the challenges, emerging work and tradeoffs between different approaches? In particular, can you articulate clearly why different SOA technologies are better or worse for certain tasks?

Paul Fremantle, November 2012
paul@fremantle.org