Service-Oriented Architecture

Final Project Specs

Objective

The objective of this project is to be able to apply the concepts of service oriented architecture in the design and implementation of a web-based SOA system. Given an organization's information system, the group is expected to design services that would cater to the different processes as well as implement these services using the web framework of choice.

Instructions

Service Analysis and Modelling (30%)

Choose an information system currently being used by an organization. Gather as much information about the organization and how the information system plays a role in its operations. Perform an analysis on the process that the information system supports by creating the necessary flowcharts and DFD diagrams. Determine the entities that will be required by your service and create an ERD for it. Create a step by step list of processes and perform logical grouping to determine service candidates (entity-centric). Add new service candidates when necessary. Perform several iterations when necessary before finalizing them in your service inventory. These services should include inserting and retrieving information. Present the final set of services by providing a diagram for the service layer stack.

Service Contract (30%)

Your group will be tasked to implement the set of services you've identified after performing analysis and modelling. The expected SOA based implementation should follow the RESTful approach. Thus, your group is required to create a technical documentation (service contract) that contains two main parts:

a) URL References

- It must be domain driven (based on entity name)

- It should specify what HTTP verb is used
- In the case of a POST, state what parameters are needed

Example:

PackageService

URL	Description	HTTP Verb	Parameters
/packages	Retrieve a set of packages	GET	n/a
/packages/create	Create a new package record	POST	● code (string) ● order_date (string) ● weight (numerical) ● length (numerical) ● width (numerical) ● height (numerical)

b) Messaging Protocol

- Provide a format of the return messages using the JSON syntax for each service endpoint and its expected response
- State all possible conditions for a given service endpoint

Example:

PackageService

URL	On Success	On Failure
/packages/create	{ “success”: { “message”: “Successfully created package” } }	{ “error”: { “message”: “Missing attributes for creating a package” } }

Implementation (40%)

Using your framework of choice, implement the services based on your documentation. For testing, you can use the curl command to act as a client for accessing the different endpoints.

Expected Output

- Background of the Organization
- Background of the Information System
- Analysis
  - Flowcharts
  - DFD
  - ERD
  - List of processes (specify filters)
  - Service Inventory
  - Service Layer Diagram
- Service Contract
  - URL References
  - Messaging Protocol
- Implemented SOA system (provide the entire framework code)

Deadline for submission of project is on **March 20, 2013 5 p.m.**