Week3-FAQ

***How do Service layer and Integration layers differ?***

In this course, you’ll observe that both Integration and Service layers are used synonymously. In the industry, technologists hold different positions on its definition.

Integration layer performs ‘integration’ with heterogeneous systems (in-house or external). Example, Amazon would integrating their web application to Facebook, Twitter, and a host of business partners and vendors (Banks for CC processing for example, or Paypal).

Service Layer services provide either a standalone coarse grain functionality (say, processing delinquency rules in the case of a Mortgage provider) or it could be ‘interacting or integrating’ with other ‘services’ (in-house or external).

Some view Integration Layer as an architecture patterns, and Services Layer services as software components that perform the integration.

***How to communicate details about successes and failures beyond T/F***

Thinking about services, I am thinking about all the information an application sometimes wants to know after a method call.  Would we want to group return values to return more than one thing.  For instance, let’s assume I just entered data in a view for a new student and wanted to save.  Let's also assume the first name is invalid (blank).  Upon returning from the request to save (create) the new student, I might want a small collection of three pieces of information such as:

1. "valid" = T/F (for success for failed save),

2.  "data" = reference to the new (or modified) student object, and

3. "fields" - A Map (Hash, etc.) containing all the fields that have an error, if any.  They could be identified by attribute name (i.e., "firstname") as a key, with corresponding (user friendly) messages (i.e., First name is a required field.) to be displayed.  This way, for instance, a view can know which fields have errors and display them at the top of the view or near the fields for the user to know what changes to make and attempt to save again.  Perhaps it could be useful elsewhere (I can't think of any right know).

Is this something someone would want to do?

**Answer:**

So, the method return type can typically take the following forms:

- void

- primitive type/or it's class equivalent (ex: int or Integer, boolean or Boolean, etc.)

- complex type object (this would be your user-defined class)

What you are thinking off is the last one, with the object holding the 3 fields you've identified.

All three options are valid and one chooses one over the other depending on what one wants to return to the caller.  Your choice of a complex object for the scenario described is perfectly fine