

# Lab 10 - Composite Components and Action Links from Data Tables

## Summary

This assignment is the final milestone towards your final project. It incorporates JSF `h:dataTable` and JSF composite component functionality. I want to make sure you don't wait until the last minute to incorporate these into your projects!

## Requirements

### Documentation

There is no further documentation required for Lab 10.

### Database Setup

Use your `itmd4515` database and user from Lab 1

### Project Setup

Your `uid-fp` repository should already be setup, and you should continue pushing your commits into GitHub for this lab. Use a prefix on your commit messages of "Lab 10"

### Project Requirements

You should add these changes to your `uid-fp` project.

1. Using `h:dataTables` components, present information relevant to the authenticated user on your application's JSF pages. In my sample application, this would be on the welcome or landing page for each area of the app.
  1. You should introduce new JSF Controllers / Backing Beans to call your services, and bind fields from your new JSF Controllers / Backing Beans to the `h:dataTable`
    1. In simple cases, you can follow the short-cut I used during class to directly invoke `@Named @Stateless` EJB components where appropriate.
  2. Within the data table, include links to View, Edit and Delete each record. Include an appropriate action link to create a new record.
  3. Implement JSF Controller functionality to handle these actions.
2. Make appropriate use of `f:convertDateTime` and other components/converters to assist you in proper formatting of your data
3. Consider usage of the ternary operator to render parts of your content or template conditionally. See examples in Week 13 lecture!

4. Make appropriate use of a JSF composite component to use on your create, read, update and delete pages. In class, I have shown how composite components can be parameterized. You can use the parameterized composite component on multiple pages, or you can follow the view/edit page pattern I showed in class. It is entirely up to you how to design the pages, but remember the principal of DRY (Don't Repeat Yourself)
5. Submit to Canvas or Beacon
  1. Right your **uid-fp** project and select "Clean"
  2. Go to your NetBeans Projects directory. Create a zip file of the **uid-fp** folder and submit it to the Canvas or Beacon assignment.