Experience Report

Working with JSF was a lot harder than I expected, even though Brandon and I already had some background in web development. For myself in particular, I found it very difficult to wrap my head around how everything worked in JSF, since a lot of things are managed by the framework behind the scenes and it’s difficult to see understand what is happening. For instance, when the user clicks on a link the URL doesn’t change to reference the new URL, which doesn’t make any sense. Also, it is very difficult to find documentation online for the things we were struggling with. Many tutorials online say to do one thing and other say to do something else. It took us a while to realize that many tutorials online were outdated. For instance, in trying to use dependency injection, which we recommend future students should use, some guides say to use the @Named annotation while others say to use the @ManagedProperty annotation. They mean different things and we found that some of the things we were trying to implement worked with one and not the other. For this reason, instead of trying to find documentation online, we recommend sticking to the recommended book, and may be not even other books out there since they may reference different versions of JSF.

Another thing we recommend is taking the time to design the application before developing. It may feel tempting to build new functionality by copying and pasting old code and making minor modifications instead of refactoring constantly and extracting common functionality. However, this will cause issues later on when you are trying to modify the code. For instance, Brandon and I made the mistake of just copying and pasting DAO methods to create new ones, and when it came back to deploying the application and having to change the DB URLs on all of those methods we encountered a ton of small issues. What we should have done from the beginning is stored the DB name and DB URL in a different file as a constant, and reference that anytime we needed it. That way changing them would require one simple change and everything would just work.

On a similar note, we also recommend templating the XHTML files. You can have a base template that includes things like the header with any stylesheets or JS files that are needed for all of the pages as well as things that are going to be common across different pages like a navigation bar or a footer. In this way, developing the UI becomes easier and in the end it will look a lot more consistent.