Presentation Layer Patterns

Daniel R. Cender

Grand Canyon University: CST-361-O500

March 14, 2021

**Write-Up**

**MVC Explanation**

The MVC (Model-View-Controller) design pattern allows for a separation of concerns between individual portions of an application. The View portion controls what gets displayed to the user after all request processing has resolved. The Controller portion manages the request from a user, deciding what business logic or data operations to apply before sending a designated response back to the user agent. The Model elements are representations of the data that will be manipulated through requests. In this example, the Track and Album classes are parts of the Model layer. The AlbumController class fits into the Controller layer, of course. The JSF templating pages all fit into the View layer of the application, naturally.

**MVC Type I**

The first iteration of the MVC pattern involved a tight coupling of the view and controller layers. All the business logic and session management logic are embedded directly into the templating pages themselves.

**MVC Type II**

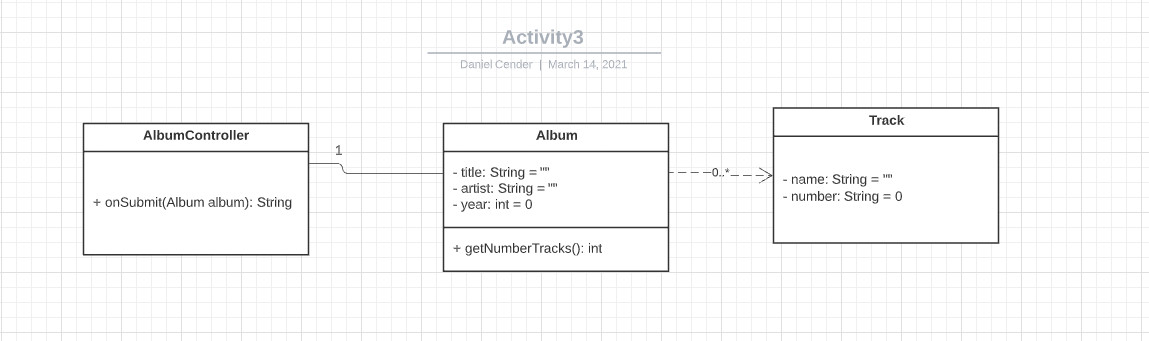
The second iteration of the MVC pattern is the one that moves all the business logic out of the presentation layer, and into the controller layer (in a servlet class).

**MVC For This Assignment**

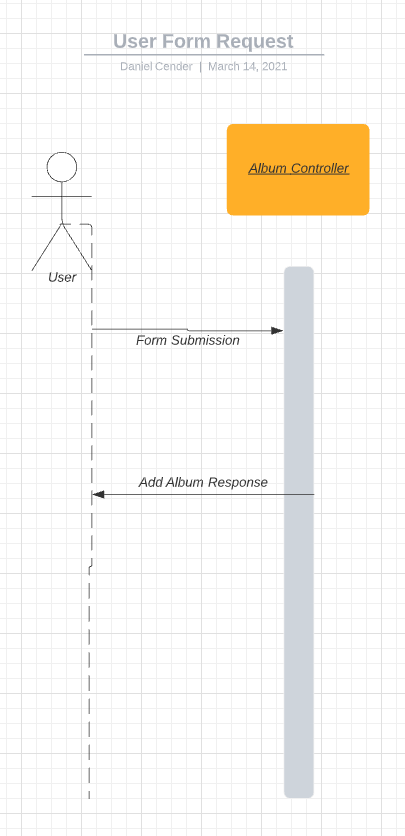
This version of MVC is version II. We are dividing business logic and connection to the backing bean behind the UI form using a controller servlet. The resulting UI page only knows what element is accessible because the servlet class placed the new Album class into the persistent FacesContext object.

**Deliverables**

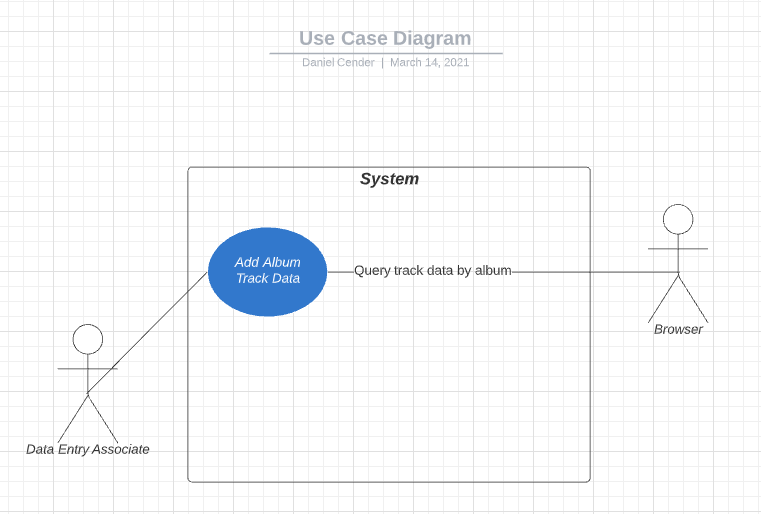
**Class UML**



**Process/Sequence UML**



**Use Case UML**



References