Jonathan Roland

Web 425 Angular with TypeScript

Prof. Richard Krasso

2/12/2021

Discussion 7.1 – Template Driven and Reactive Forms

In enterprise environments, applications are often based heavily on forms. The main concerns of a forms-based application are with keeping track of progress state, validation, and error display. Angular has two prominent forms libraries with differing models of execution: template driven and reactive.

Angular’s FormsModule contains the template driven functionality of the now-deprecated AngularJS ng-model directive. This was a two-way data binding model and has been repackaged as ngModel. Whenever FormsModule is invoked in Angular, the NgForm directive will automatically be bound to any <form> element in the DOM of the component. It also makes the directives ngForm, ngModel, and ngSubmit available which track validity state, bound values, and click events respectively. Once a click event is fired, the ngSubmit directive prevents a standard HTTP request and instead utilizes whatever method it has been wired with to handle form submission. The primary disadvantage of this template driven model is that testing can only be done by actual interaction with the DOM.

Angular’s ReactiveFormsModule allows for the active approach to handling forms. It utilizes the formGroup directive on a form element to bind to a specified variable in the component. This is used to keep track of all form elements as attributes of a FormGroup type. Each element is tracked in the component by instantiating a FormControl type in the component that matches a formControlName attribute provided in the element. This allows a programmatic approach to managing the form and provides for cleaner code. Reactive forms are so named because the basis of their functionality is an Observable type of architecture which reacts to bound events.

Template based forms are convenient for smaller forms since we do not have to worry as much about the logic and can leverage pre-built functionality. Reactive forms provide better separation of template and component roles and clearer code, in addition to scaling better to larger forms. While template based forms may be useful for certain cases such as smaller forms, reactive forms generally provide a better basis for form building.

**Reference:**

Angular University. (2021, January 26). *Angular Forms Guide: Template Driven and Reactive Forms*. Angular University Blog. https://blog.angular-university.io/introduction-to-angular-2-forms-template-driven-vs-model-driven/