## Technical Documentation:

**Introduction:**

Angular 2 application helps in creating single page App. The responsive design for the app is created by using the Bootstrap classes.

The front end code and design mainly lies in the “app” folder.

**HTML:**

The app.component.html file consists of the html template for paginated view, course view and CRUD operations on the courses.

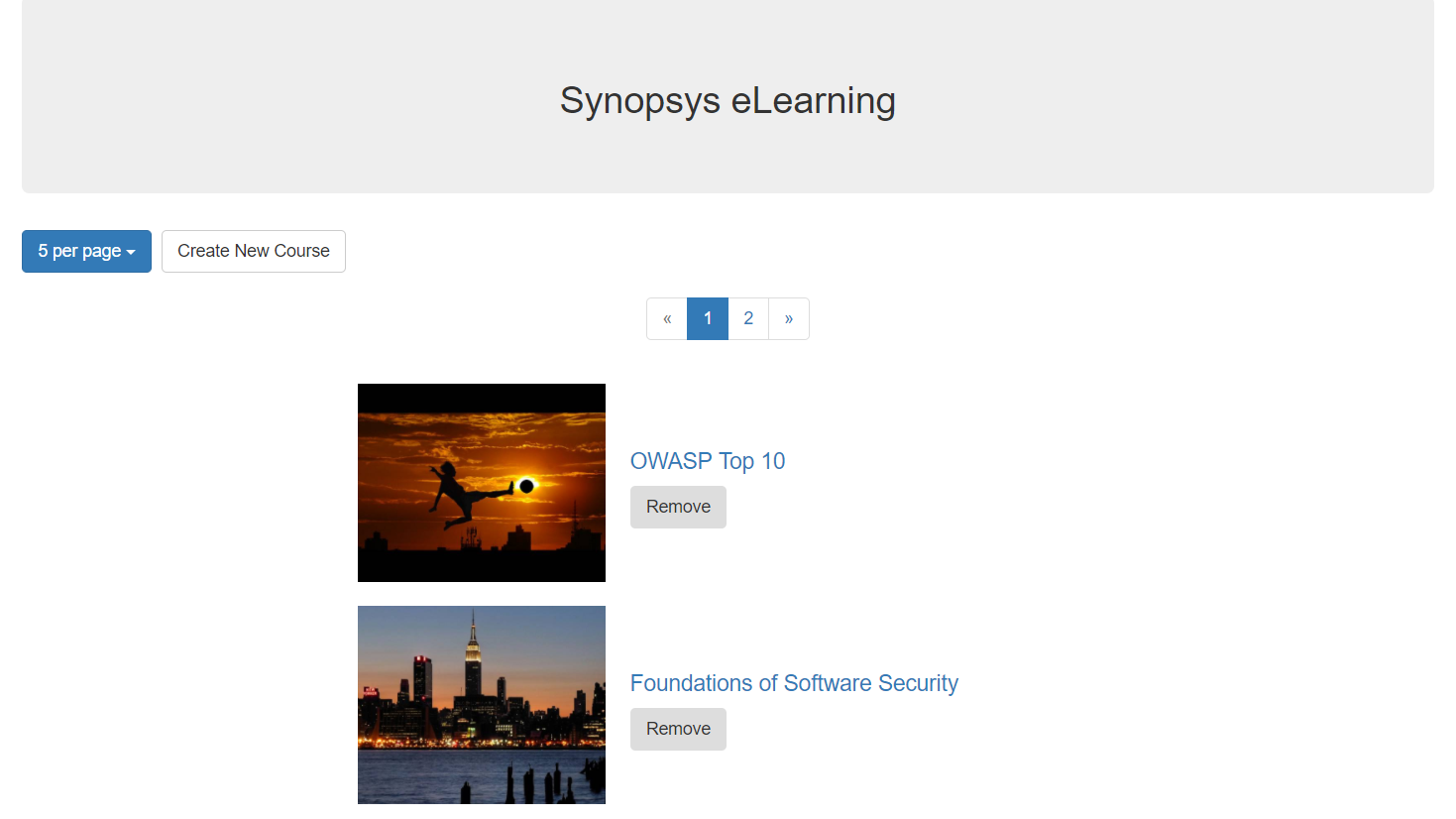
**App Service:**

App.service.ts file mainly provides an injectable service to make an HTTP GET call to the inventory.json file. The inventory.json data is ingested through this service in the app component. Inventory.json file is stored in the App folder.

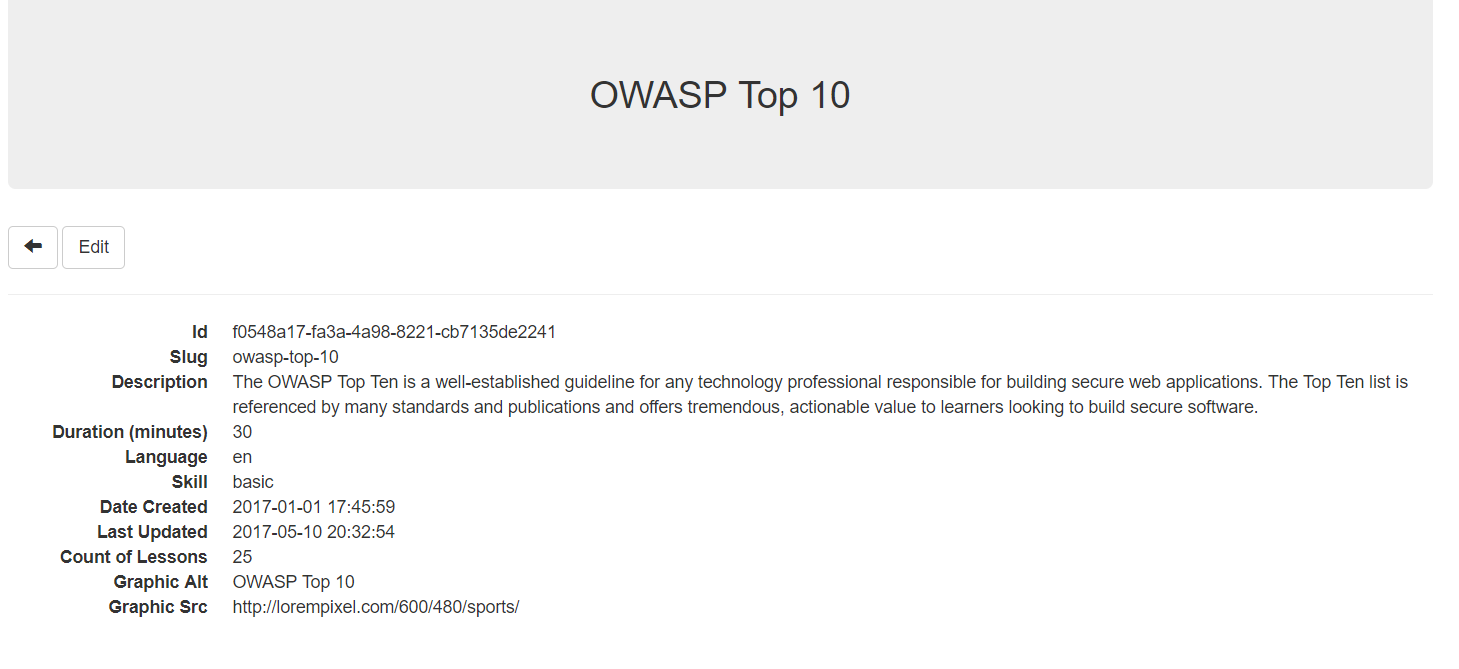
**App Component:**

The app.component.ts file defines the component for the angular app and all the functions used to render the courses data. The inventory service is called on app initialization and data is stored in a global variable.

**Paginated List View:** The paginate() method is called once the app is initialized. This method reads the default value for the no. of courses per page and sets the page values. SelectInventory() method is then called which filters the inventory to view only the first 5 courses.

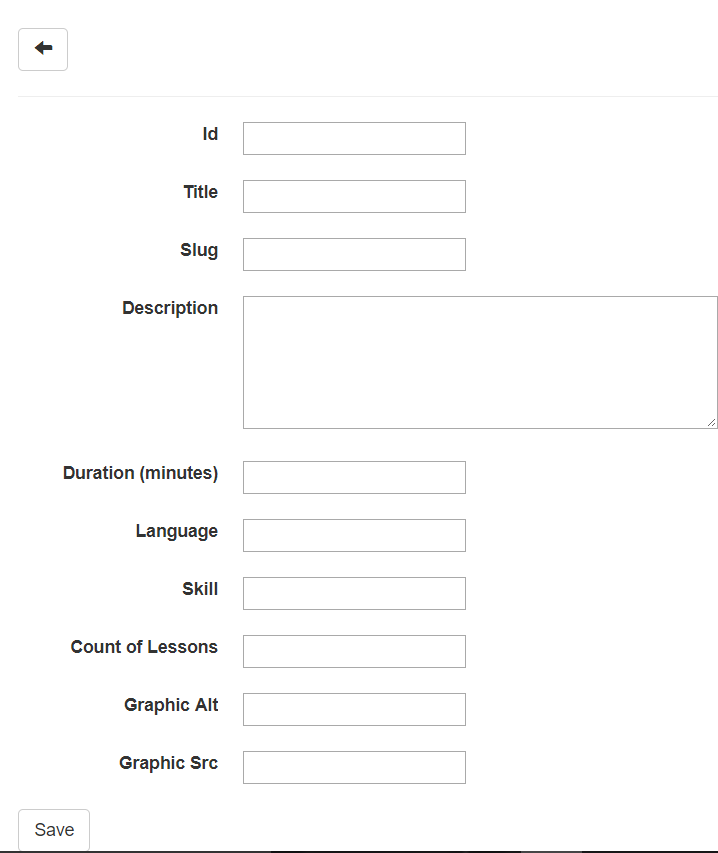
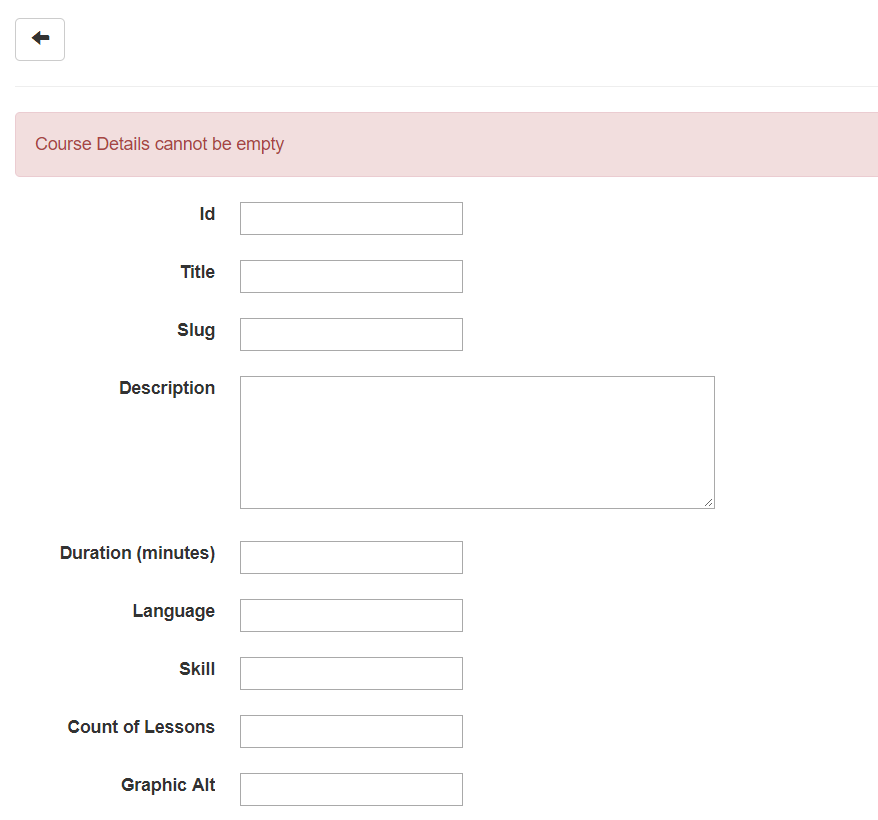


**CourseView:** Course view is rendered when the title of the course is clicked. The angular directives used on the html template hide and show the <div> elements depending upon the flag values.



The course details can be edited by clicking the edit button. On click, the edit elements are rendered on the screen with text boxes to update course values. The ngModel directive is used to enable 2 way binding of the data with the selected course.

**Create View:** Create new course button renders the div elements for new course. The data validation is performed for checking if the ID or Title are empty. The addCourse() method in the component performs a push on the inventory array and calls the Paginate() method to update the paginated list view.

**Delete Course:** The remove() method performs a splice() operation on the inventory and calls the selectInventory() method to refresh the displayed inventory data.