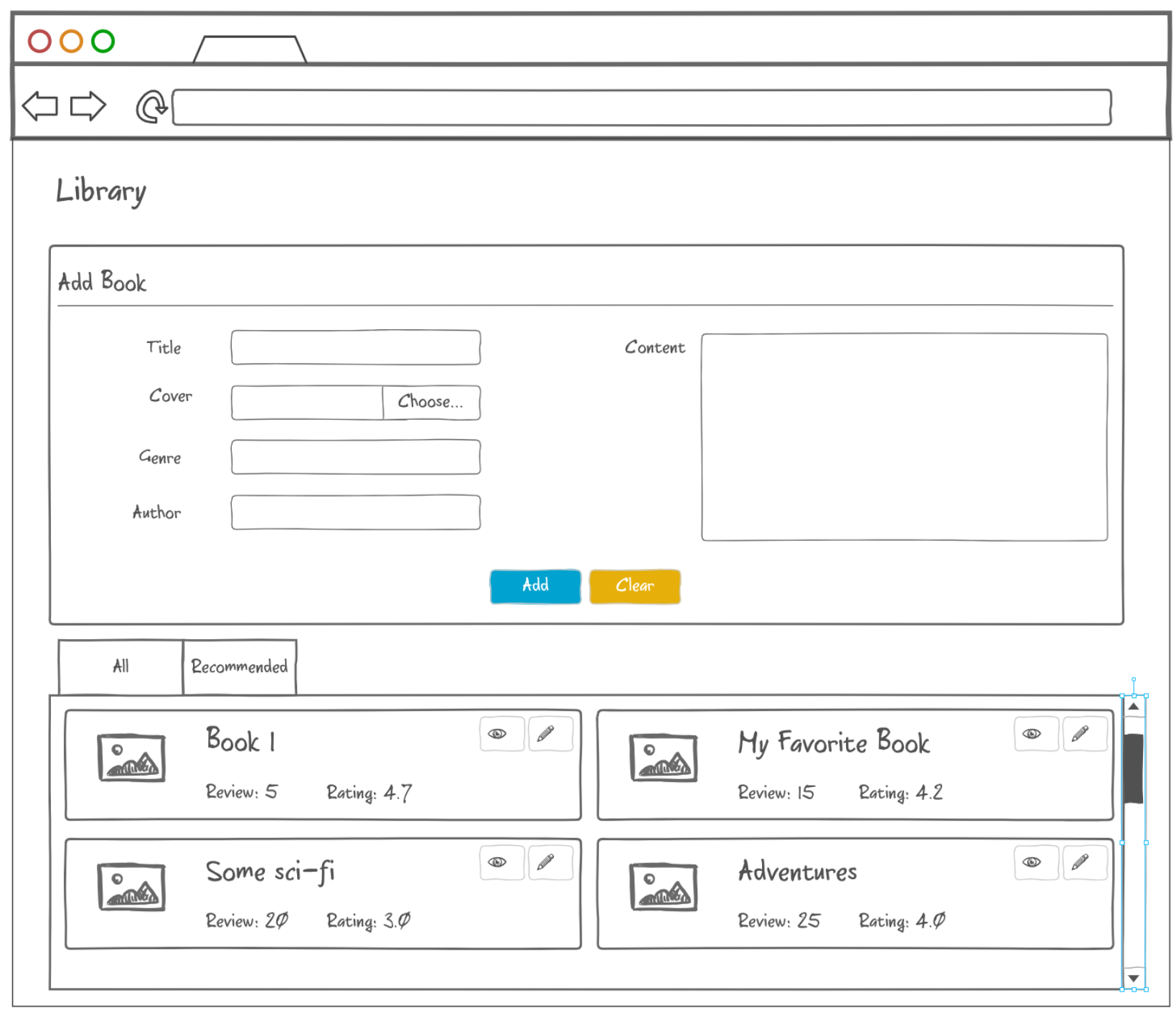
# Home task #3 | Angular | Library

Your task is to create an Angular application based on the .NET backend you’ve built for the 2nd task. You can connect it to the server, but this is not required. It would be enough if your services provide dummy (fake) data. Within this task, you need to:

1. [Create](https://angular.io/guide/setup-local) an [Angular](https://angular.io/start) application using [Angular CLI](https://angular.io/cli).
2. Using Angular CLI generates a set of components based on wireframes (pic.1, pic.2). Suggested composition:
   1. app
   2. books-page
   3. book-list
   4. book-list-item
   5. edit-book
   6. view-book
3. For each component, add a markup (use pic. 1 and pic. 2 as references). You can use either your own styles or a library like [Angular Material](https://material.angular.io/). We will not decrease the score because of the UI design.
4. Add a set of actions for each component:
   1. book-list
      1. Show All or Recommended list
   2. book-list-item
      1. Edit item - show book in the form on the top, change Label from Add to Edit
      2. View item - show modal with book details
   3. edit-book
      1. Choose pic - upload a picture and convert it to base64
      2. Add/Edit book - save book on backend and reload the list
      3. Clear book - clear all inputs
   4. view-book
      1. Close modal - dismiss modal dialog
5. Generate a service (or services) to manipulate data: load, save, delete books. Try to keep components as simple as possible.



Pic. 1 - Main page of the Library



Pic. 2 - Book details modal dialog

## Notes

* It would be great if you use an [async pipe](https://ultimatecourses.com/blog/angular-ngif-async-pipe) instead of an explicit subscription.
* Use [display: grid](https://css-tricks.com/snippets/css/complete-guide-grid/) or [display: flex](https://css-tricks.com/snippets/css/a-guide-to-flexbox/) to align your elements
* Use [Angular DevTools](https://angular.io/guide/devtools) during development
* There are no requirements to the project structure, but you can follow [this](https://blogs.halodoc.io/angular-best-practices/) guideline.
* If you connect the app to your backend part, do not forget to set up [CORS](https://medium.com/@alistairrobson_36819/how-to-add-cors-to-a-dot-net-core-web-api-15312734c796) middleware