*Owen Lindsey*

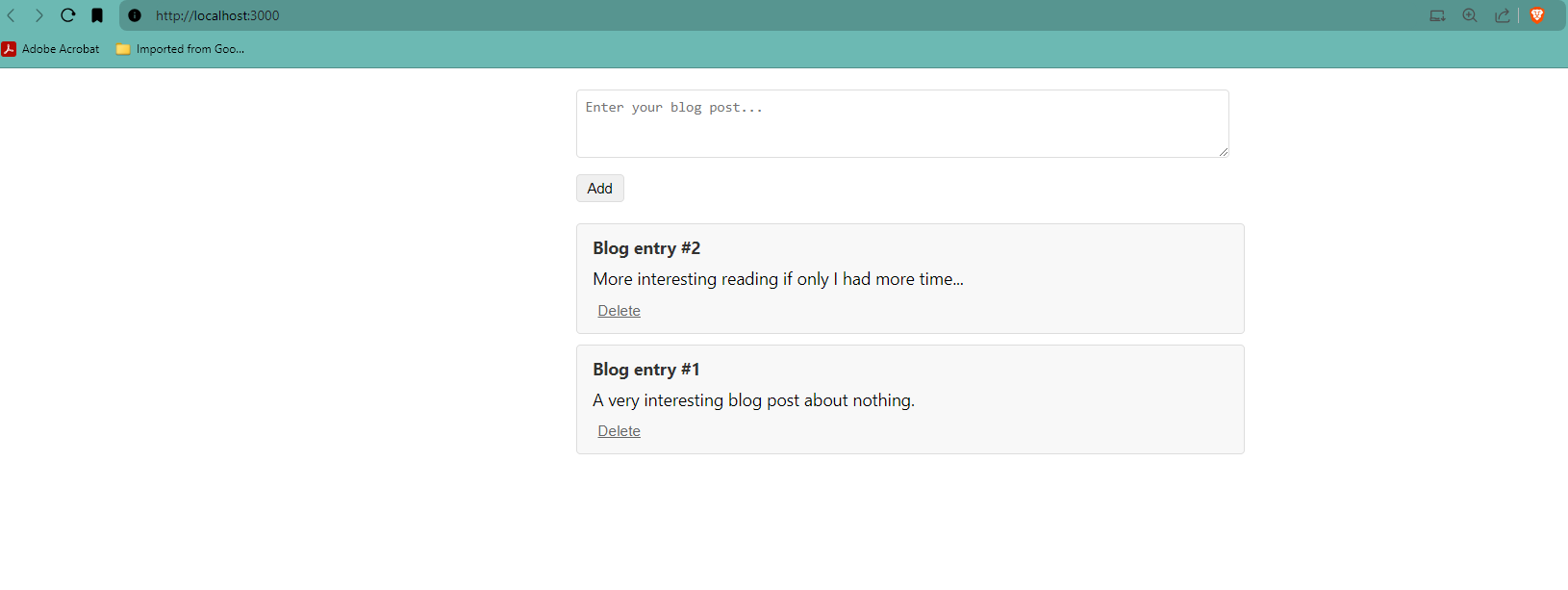
*Professor Sparks, James*

*CST-391 Activity 7*

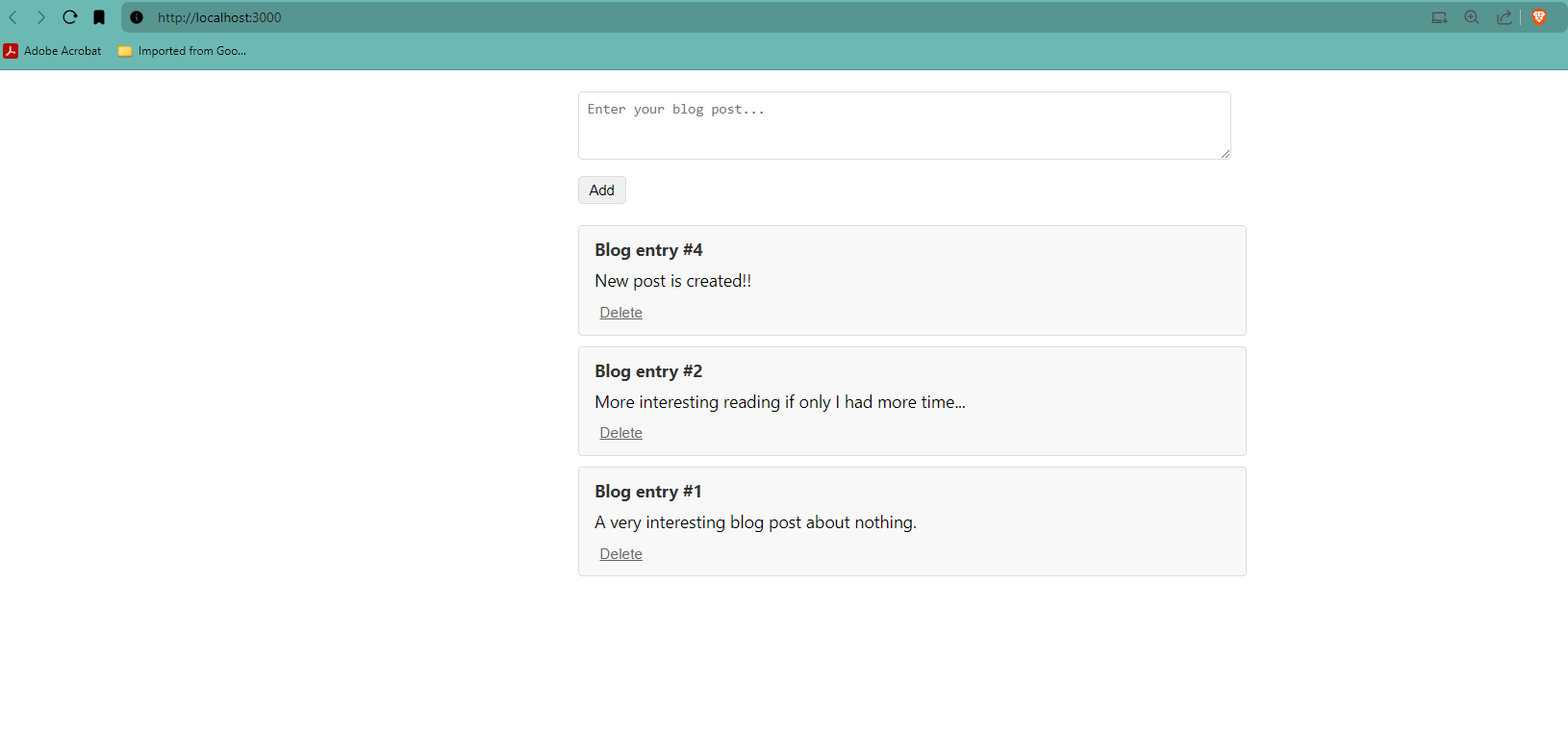
*12/1/2024*

**Dynamic Components Mini Application:**

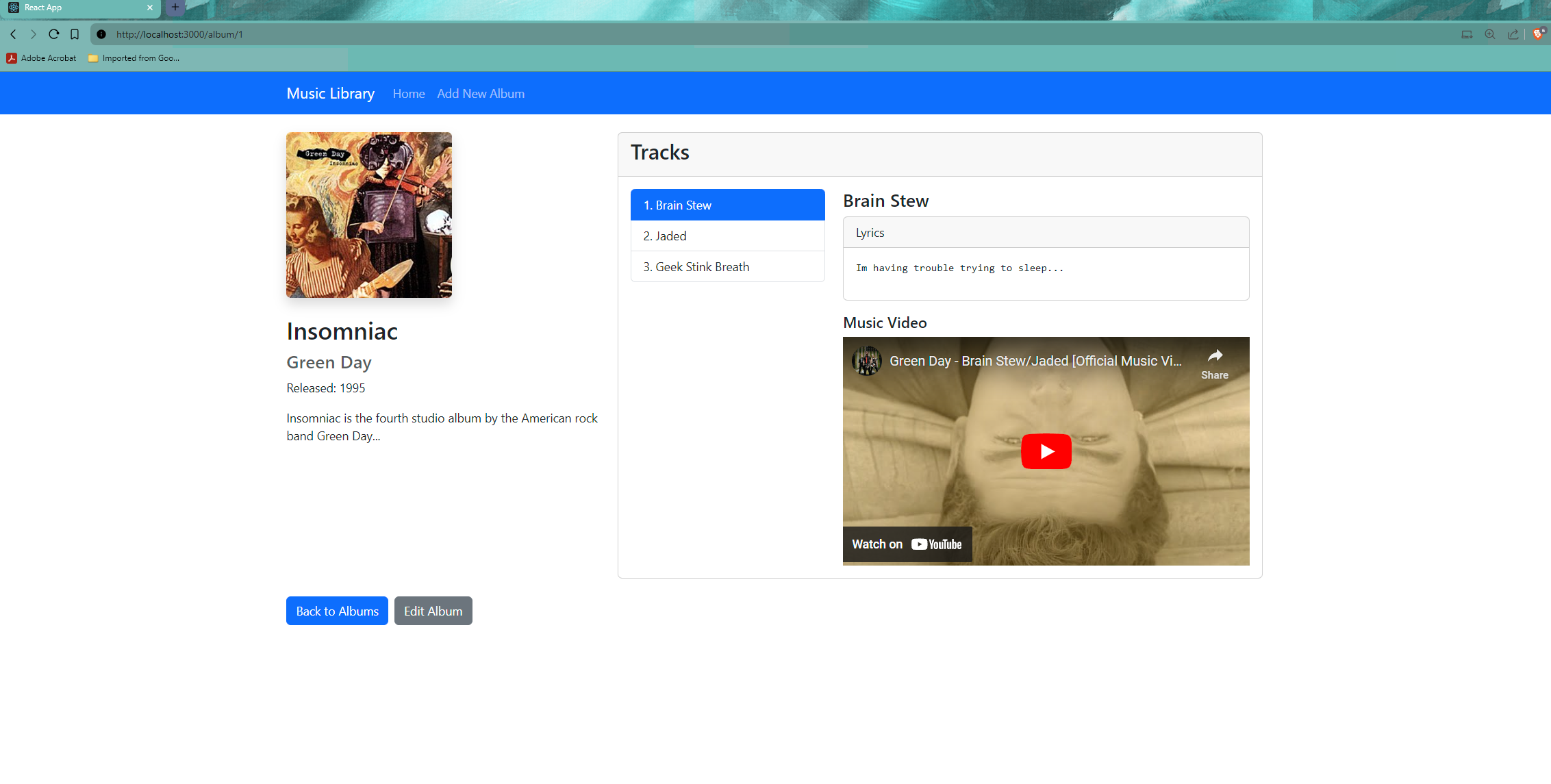
Screenshot displays the ability to remove blog posts as blog entry number 3 is no longer present.

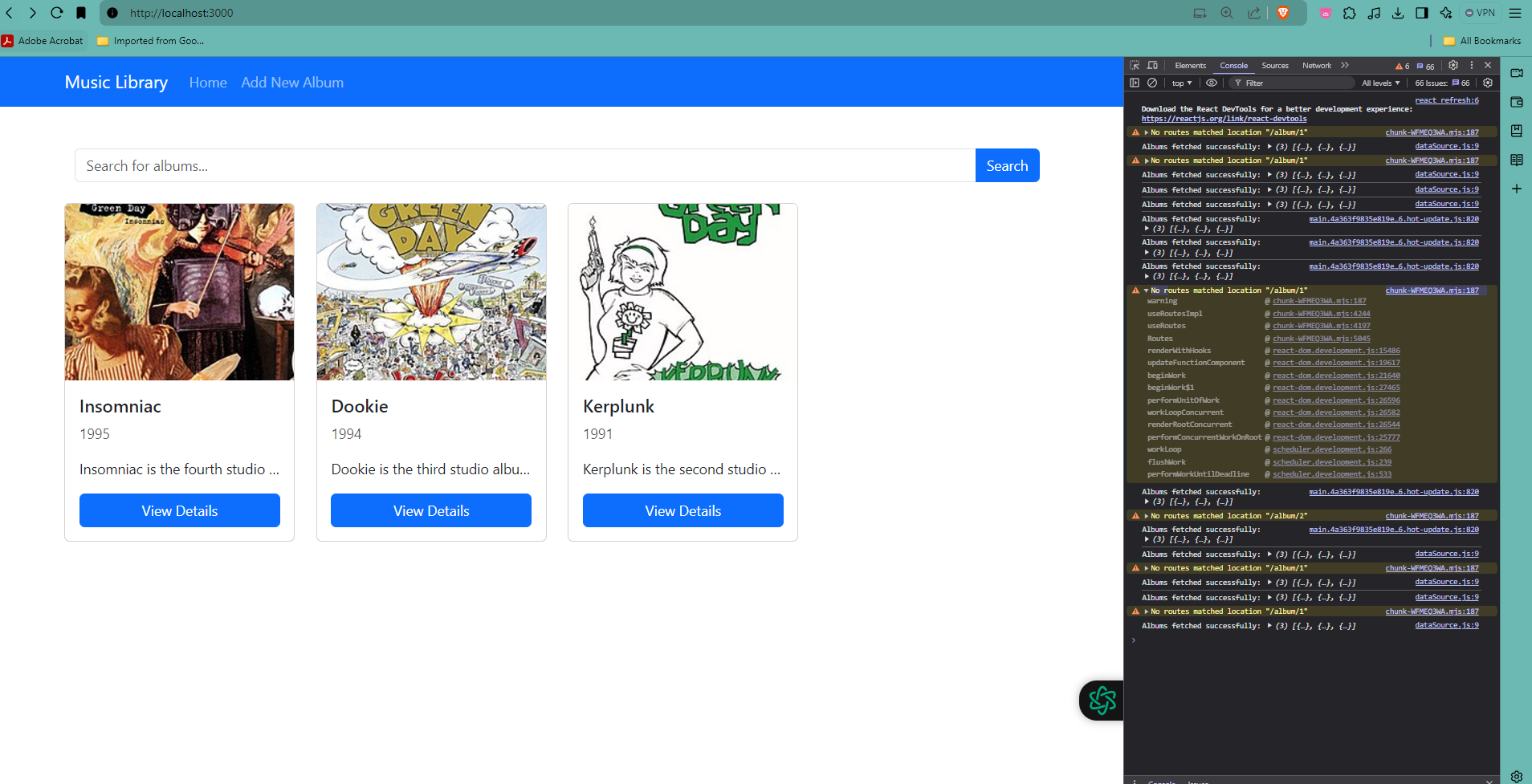


Adding a blog entry shows it as Blog entry #4 along with the text body.

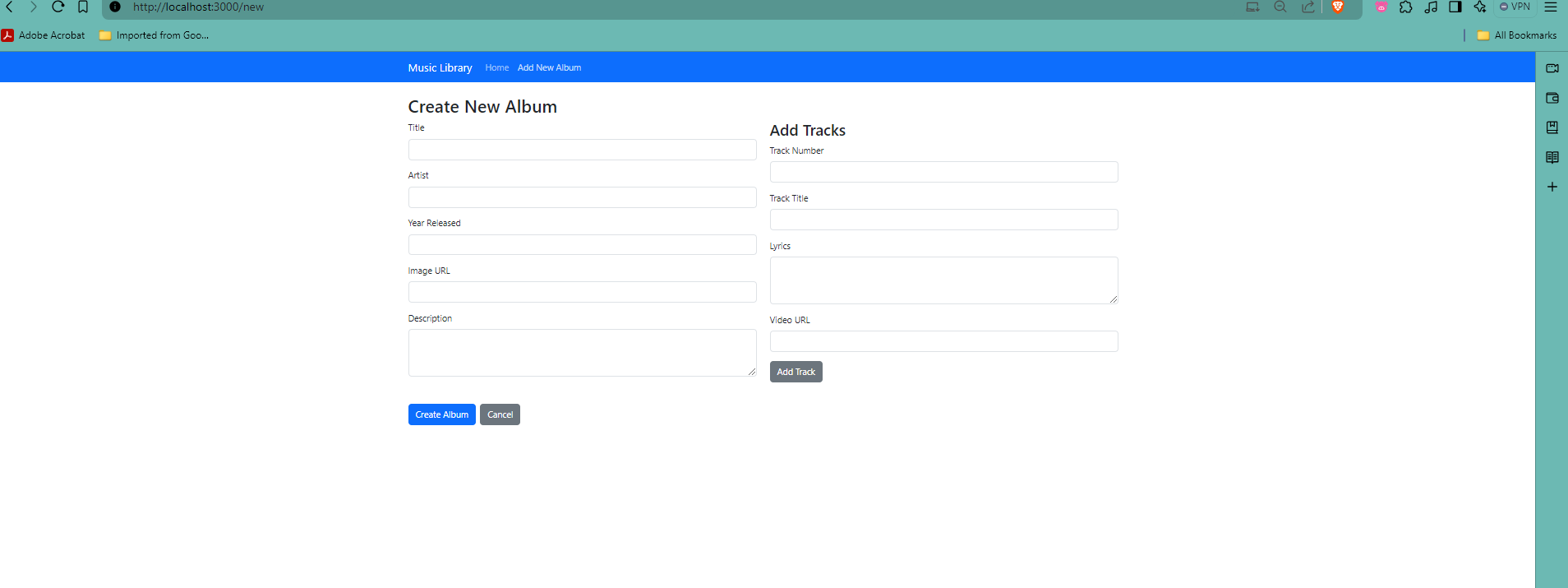
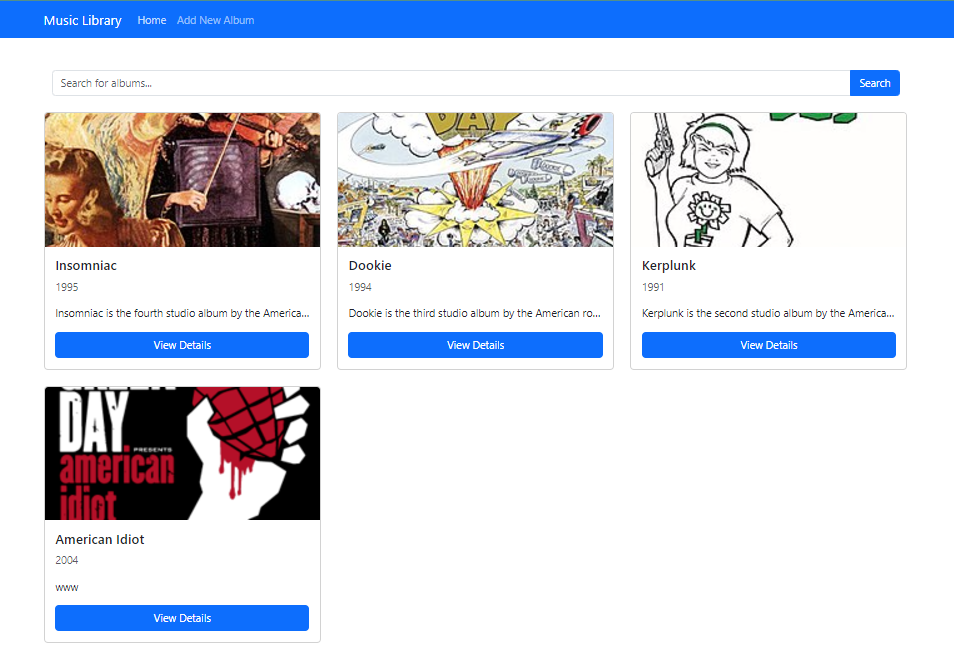
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
  
**Stopping Point #5 – Dynamic Components Demo**

This screenshot shows the album details along with the track list, lyrics, and music video.

  
  
**Summary of new features:**  
The dynamic components demo added the ability to dynamically add and remove blog posts using React state and props. Functional components were used, which are simpler and more concise than class components. The props object allows passing data from a parent component to its children, while state enables components to manage their own data that can change over time. The map function and spread syntax were employed to render lists of components and update state arrays immutably.  
  
**New terminology:**  
map function & spread syntax.

**Stopping Point #6 – New Album**  
This screenshot shows the website before the addition of a new album   


This screenshot shows the form the user must fill out to create a new album.

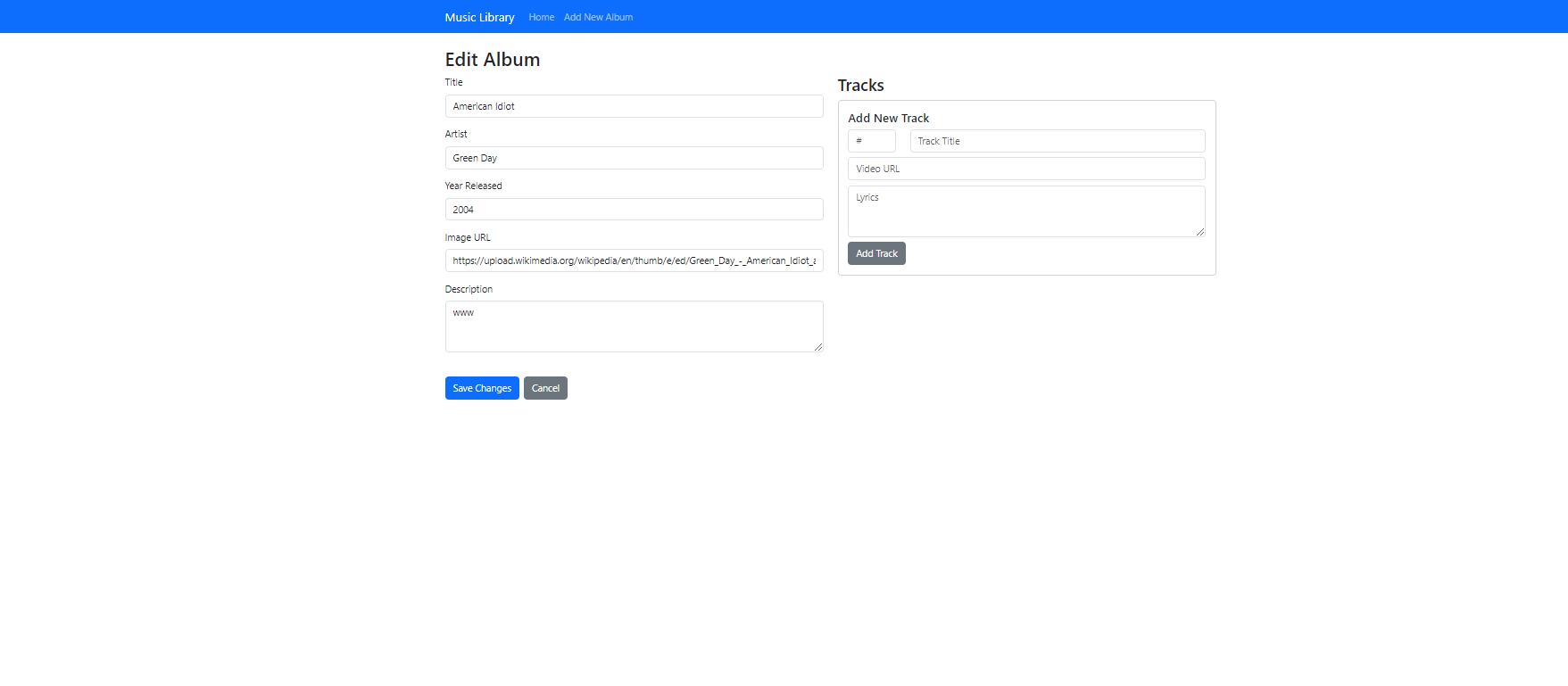
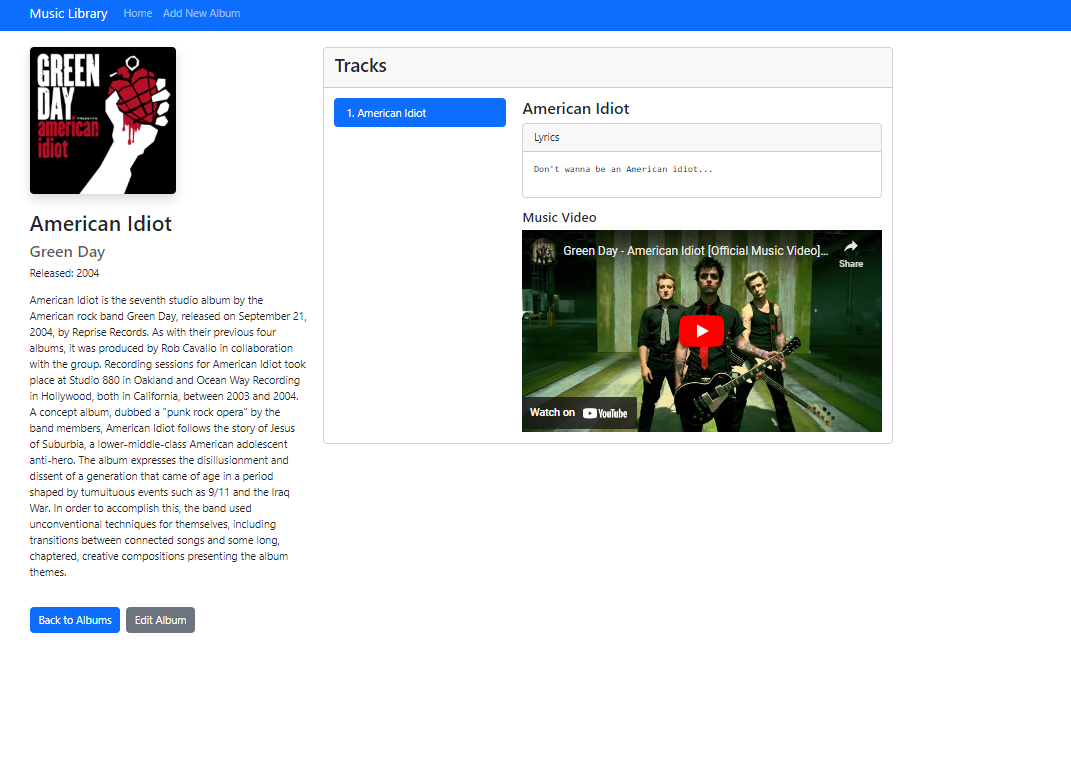
  
  
This screenshot shows the addition of a new album   


**Stopping Point #6 – New Album**

Summary of new features:  
In this section, a form was created to allow adding new albums to the collection. The form leveraged Bootstrap for consistent, responsive styling. Controlled components were implemented, where form inputs are controlled by React state rather than the DOM, enabling a "single source of truth" for the form data. The onChange event was utilized to update the state whenever form fields are modified by the user  
  
New terminology:

controlled components and onChange event

**Stopping Point #7 – Edit Albums:**

This screenshot allows us to edit the current contents of the American Idiot album and tracks.   
  
  
I have added a Track and put the proper information in the description section of the album.  


**Stopping Point #7 – Edit Albums:**

**Summary of new features:**  
The album form was enhanced to support both adding new albums and editing existing ones. The component was generalized to differentiate between "add" and "edit" modes based on the presence of an album prop. REST API conventions were followed, using POST requests to create new resources and PUT requests to update existing ones. The album ID was maintained when editing to map to the correct database record.

**New terminology:**

This was leveraging REST API conventions, so this is mostly familiar.