# Enhance Home Component to Integrate ListOptions Components

## **Description:**

Enhance the Home component to integrate two instances of the ListOptions component. One instance will display a list of movie genres using the getMovieGenreData service, and the other will display a static list of two sorting options: asc and desc. When selecting an option from each ListOptions component, the getMovies service should be called appropriately to filter and sort the list of movies. The structure of the state should change to hold the selected filter and sort options: { filters: { pagination: { currentPage: number, totalPages: number }, genres: { value: string, label: string }, sort: { value: string, label: string } } }.

This task ensures that the Home component is enhanced to provide a more interactive and dynamic user experience by integrating two instances of the ListOptions component for filtering and sorting movies based on genres and sorting options. The state is appropriately updated to handle the selected options.

## Acceptance Criteria:

1. State Structure:
   1. Update the structure of the state in the Home component to hold selected filter and sort options.
   2. The state structure should be: { filters: { pagination: { currentPage: number, totalPages: number }, genres: { value: string, label: string }, sort: { value: string, label: string } } }.
2. Integration with ListOptions:
   1. Integrate two instances of the ListOptions component within the Home component.
   2. One instance should display a list of movie genres using the getMovieGenreData service.
   3. The other instance should display a static list of two sorting options: asc and desc.
3. ListOptions Props:
   1. Pass the necessary props to each ListOptions instance:
      1. For the genre options, use the getMovieGenreData service and pass the onChange callback to handle genre selection.
      2. For the sorting options, pass a static array of options (asc and desc) and the onChange callback to handle sorting selection.
4. Service Calls:
   1. Update the getMovies service calls in the Home component to include the selected filter and sort options.
   2. Call the service appropriately when the user selects options from the ListOptions components.
5. Unit Testing:
   1. Write unit tests for the updated Home component, covering scenarios such as selecting genres, selecting sorting options, and validating service calls.

## Definition of Done:

1. The state structure in the Home component is updated to hold selected filter and sort options.
2. Two instances of the ListOptions component are integrated within the Home component.
3. Props are passed to each ListOptions instance, including options and callbacks.
4. Service calls in the Home component are updated to include selected filter and sort options.
5. Unit tests for the updated Home component pass successfully, covering various scenarios.
6. The code is well-documented, adheres to coding standards, and includes comments where necessary.