# Create MovieDetail React Component

## Description:

Create a React component named MovieDetail to integrate the MovieDetailCard component with the getMovieDetail service. Display this component on the route "/movie/:id" and implement features to show a loader while fetching data from the API and display a notification in the case of an error. Extract the id of the movie from the URL and use it in the getMovieDetail service. Provide a way for the user to navigate back to the Home component from the MovieDetail component.

This task ensures the creation of a MovieDetail component that integrates the MovieDetailCard component with the getMovieDetail service. It provides a detailed view of a single movie based on its id, handling loading and error states gracefully.

## Acceptance Criteria:

1. Component Creation:
   1. Create a file named MovieDetail.tsx within the components folder.
2. Integration with MovieDetailCard:
   1. Integrate the MovieDetailCard component within the MovieDetail component.
3. Service Integration:
   1. Use the getMovieDetail service to fetch detailed information about the movie based on its id.
   2. Extract the movie id from the URL (route parameter).
4. Loader Display:
   1. Display a loader (e.g., a spinner) while fetching data from the API.
5. Error Handling:
   1. Implement error handling to display a notification in case of an error during data fetching.
6. Navigation Back:
   1. Provide a clear and intuitive way for the user to navigate back to the Home component.
7. Route Configuration:
   1. Configure the route for the MovieDetail component to be "/movie/:id", where ":id" is the id of the movie.
8. Unit Testing:
   1. Write unit tests for the MovieDetail component to ensure correct integration with the MovieDetailCard, proper data fetching, and error handling.

## Definition of Done:

1. The MovieDetail.tsx file is created within the components or pages folder.
2. The MovieDetail component is developed using functional component syntax.
3. The MovieDetailCard component is integrated within the MovieDetail component.
4. The getMovieDetail service is used to fetch detailed information about the movie based on its id.
5. The movie id is extracted from the URL (route parameter) and used in the service call.
6. A loader is displayed while fetching data from the API.
7. Error handling is implemented to display a notification in case of an error during data fetching.
8. The route for the MovieDetail component is configured to be "/movie/:id".
9. Unit tests for the MovieDetail component pass successfully, covering various scenarios.