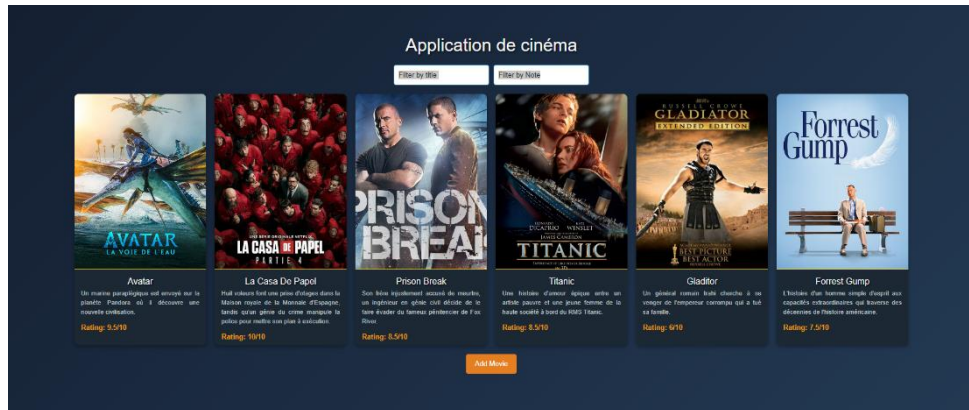


Application de cinéma avec React Debugging

1) Je vais travailler sur cette application : Application de cinéma



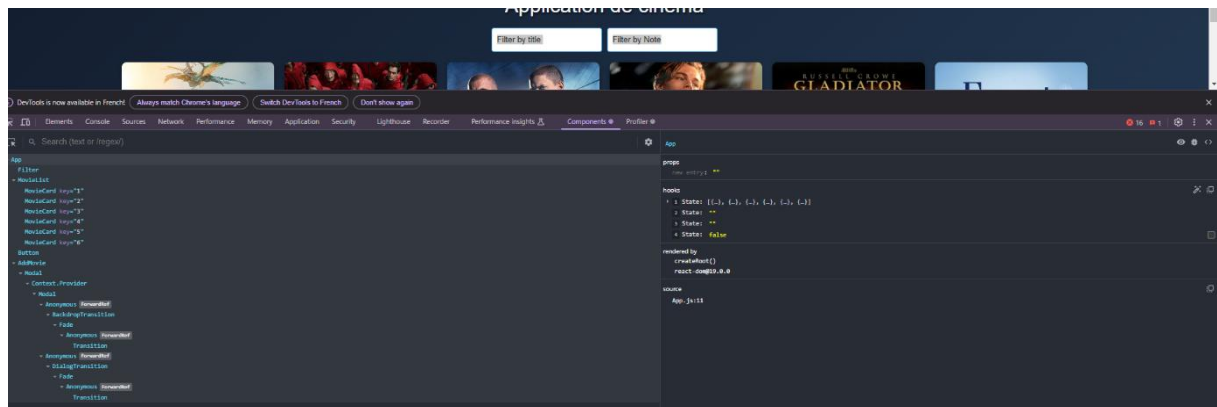
2) Installation de React Developer Tools :

Etape1 : Installation de l'extension React dev tools dans chrome.

Etape2 : Installation de l'outil react-devtools en utilisant le terminal de VsCode.

Etape3 : npm start sur le terminal et faire inspecter pour voir l'extension components

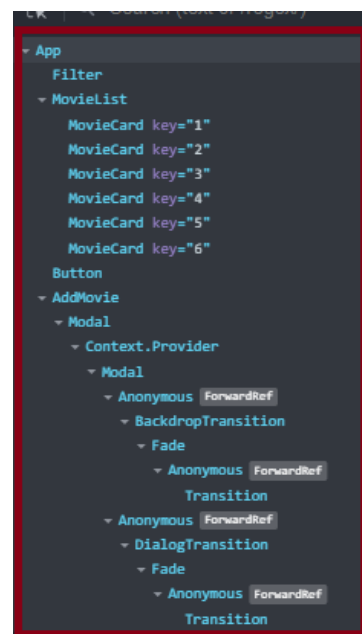
Maintenant on peut naviguer entre les composants, les states et les props de notre application



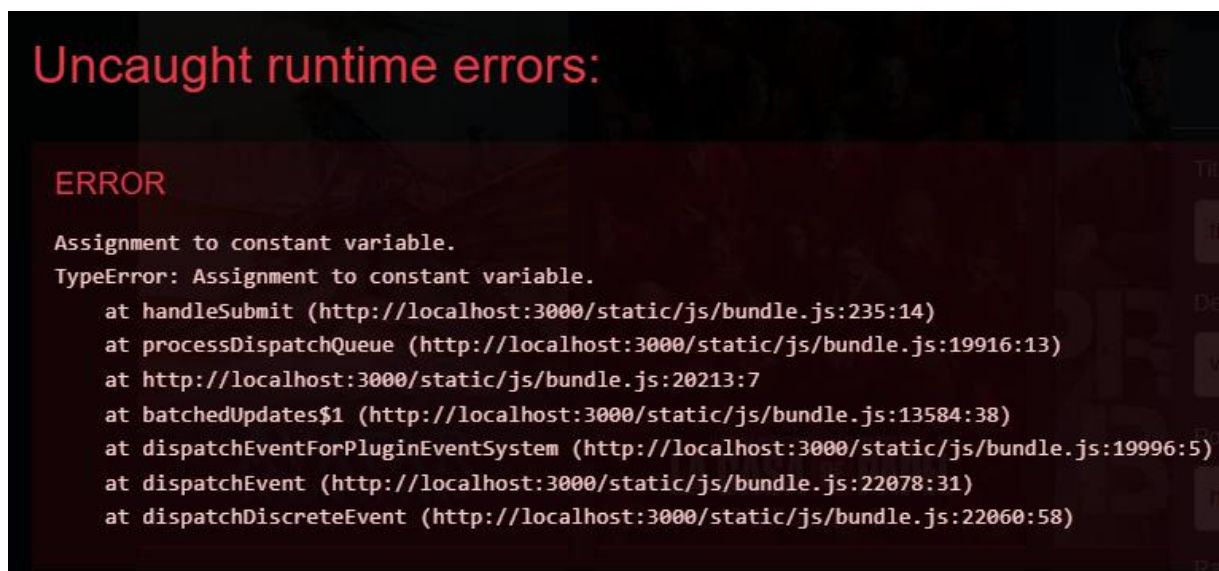
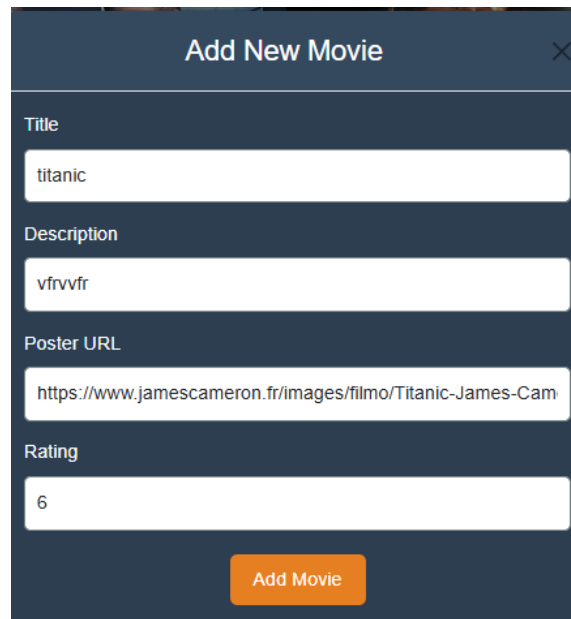
3) L'arbre des composants :

-Les composants à inspecter :

- App : contient le state principale et gère le passage des props.
- Filter : gère les filtres pour les titres et les notes des films.
- MovieList : reçoit la liste filtrée des films et rend chaque carte de film.
- AddMovie : permet d'ajouter un nouveau film à la liste.

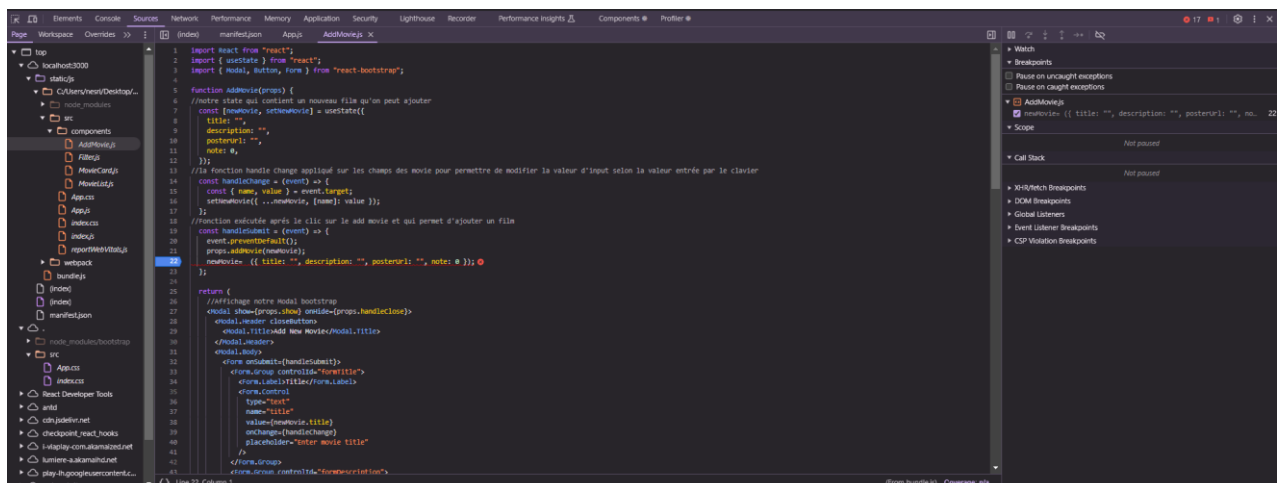


- 4) Quand j'essaie d'ajouter un film il m'affiche une erreur alors j'essaie de debugger le problème :



- 5) Commençons le débogage :

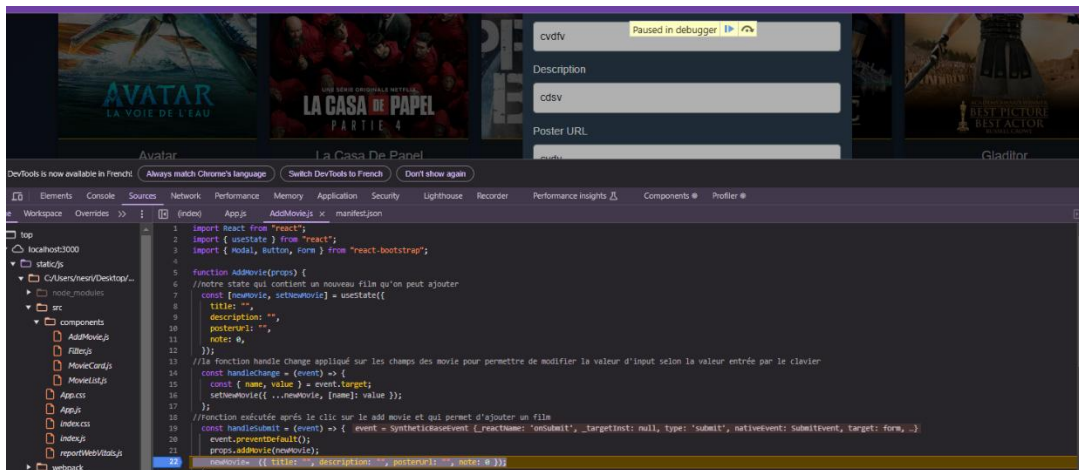
Le problème est apparu lors du click sur le bouton Add Movie alors il est intéressant de commencer par voir le code de ce composant :



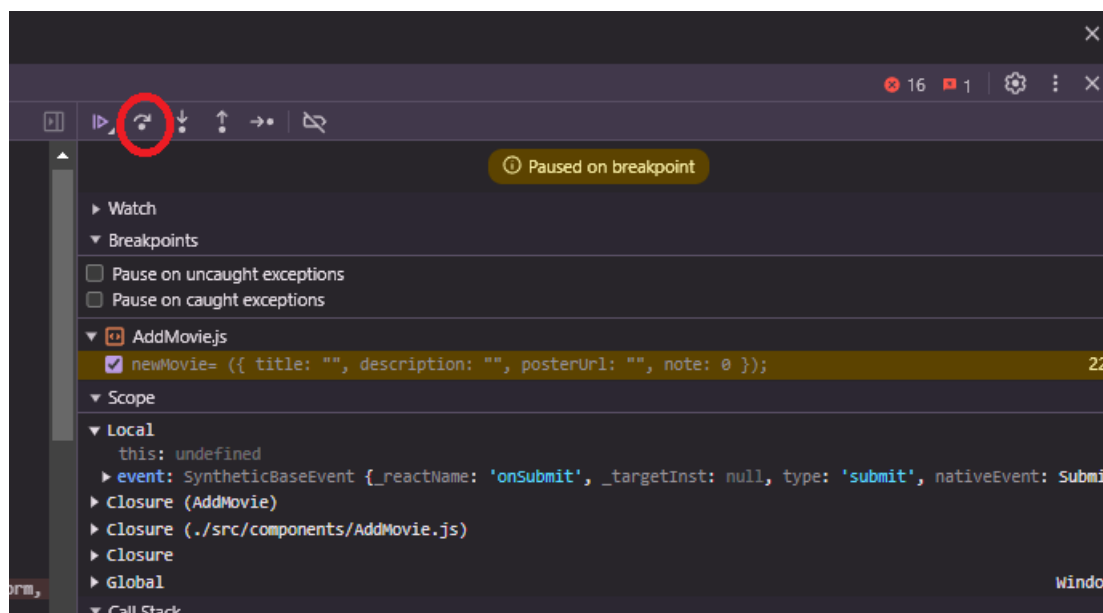
J'ai mis un breakpoint sur la ligne pour le new movie

```
18 //Fonction exécutée après le clic sur le add movie et qui permet d'ajouter un film
19 const handleSubmit = (event) => {
20   event.preventDefault();
21   props.addMovie(newMovie);
22   newMovie= ({ title: "", description: "", posterUrl: "", note: 0 });
```

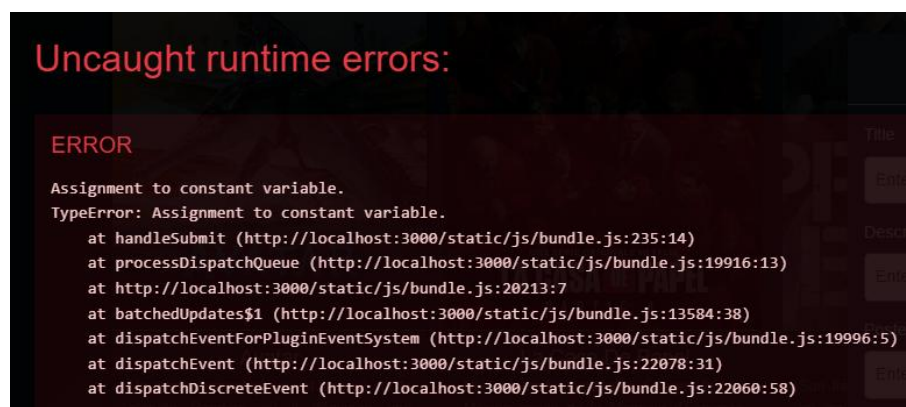
On essaie d'exécuter le code avec ce breakpoint et on clique sur le bouton Add movie, il n'y a pas de problème.



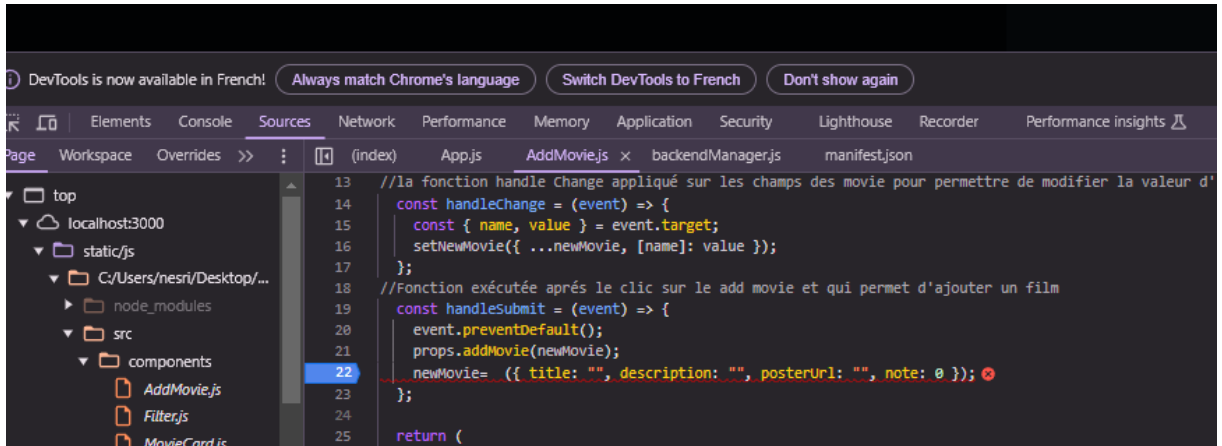
Passant maintenant à la ligne de code suivante :



L'erreur s'affiche :



Et le code se bloque au niveau de cette ligne de code :



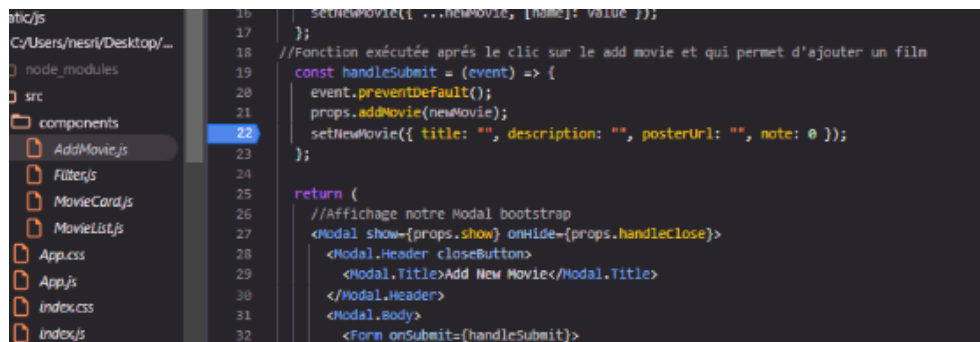
```
13 //la fonction handle Change appliqué sur les champs des movie pour permettre de modifier la valeur d'  
14 const handleChange = (event) => {  
15   const { name, value } = event.target;  
16   setNewMovie({ ...newMovie, [name]: value });  
17 };  
18 //Fonction exécutée après le clic sur le add movie et qui permet d'ajouter un film  
19 const handleSubmit = (event) => {  
20   event.preventDefault();  
21   props.addMovie(newMovie);  
22   newMovie= ({ title: '', description: '', posterUrl: '', note: 0 });  
23 };  
24  
25 return (
```

Alors la ligne où nous avons mis le breakpoint pose le problème.

- 6) Revenons sur notre code, on remarque que la mise à jour de state newMovie est fait d'une façon incorrecte.

Revenons à notre code et corrigeons l'erreur :

On utilise setNewMovie au lieu de faire la mise à jour directement sur newMovie



```
16 setNewMovie({ ...newMovie, [name]: value });  
17 };  
18 //Fonction exécutée après le clic sur le add movie et qui permet d'ajouter un film  
19 const handleSubmit = (event) => {  
20   event.preventDefault();  
21   props.addMovie(newMovie);  
22   setNewMovie({ title: '', description: '', posterUrl: '', note: 0 });  
23 };  
24  
25 return (  
26   //Affichage notre Modal bootstrap  
27   <Modal show={props.show} onHide={props.handleClose}>  
28     <Modal.Header closeButton>  
29       <Modal.Title>Add New Movie</Modal.Title>  
30     </Modal.Header>  
31     <Modal.Body>  
32       <Form onSubmit={handleSubmit}>
```

On clique sur le bouton on remarque que le code ne se bloque pas au niveau de ligne 22 (update de state) et on passe à la ligne suivante. Maintenant notre code fonctionne de la bonne manière.

