

## ReactJS TP3

Lien GIT :

<https://github.com/shivailan/reactbook-tp3bis>

### Partie 1 : Introduction aux Hooks et à l'État de Base

1,2,3 :



Shiva

Je suis Shiva

❤️ 7 J'aime

J'aime ❤️      Réinitialiser

Julie

Je suis julie

❤️ 4 J'aime

J'aime ❤️      Réinitialiser

4)

Parce que React n'applique pas immédiatement la mise à jour de l'état.

Il regroupe plusieurs mises à jour pour des raisons de performance.

En utilisant la version fonctionnelle, on est sûr de toujours obtenir la valeur la plus récente de likes, même si plusieurs clics se produisent très vite.

5)

Si on écrivait `setLikes(likes + 1)`, et que l'utilisateur clique deux fois très vite, les deux clics utiliseraient la même valeur ancienne de likes avant la mise à jour — le résultat serait `+1` au lieu de `+2`.

Avec `setLikes(prevLikes => prevLikes + 1)`, chaque clic prend la dernière valeur mise à jour, donc le compteur est toujours correct.

6)

Parce que si l'utilisateur clique très vite deux fois, les deux clics lisent l'ancienne valeur avant qu'elle soit mise à jour.

La forme fonctionnelle `setLikes(prev => prev + 1)` évite ce problème en utilisant toujours la dernière valeur à jour.

8)

Si on modifie directement la variable d'état sans utiliser `setX()`, la valeur change en mémoire mais l'affichage ne se met pas à jour. React ne re-render le composant que lorsqu'on utilise la fonction d'état, ce qui permet de synchroniser l'interface avec les données.

9)

Le composant se re-render après un `setState()` parce que React détecte que l'état a changé. Il relance alors le rendu du composant pour que l'interface reflète la nouvelle valeur de l'état.

## Communication Parent/Enfant et `useEffect`

### ReactBook

Ajouter un Post

Alice

Premier post !

❤️ 5 J'aime

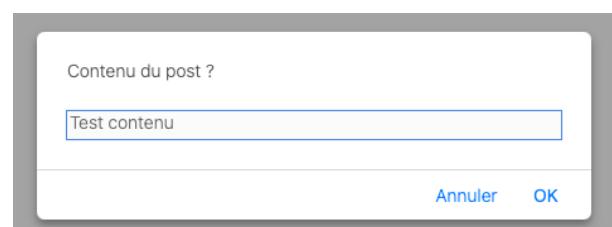
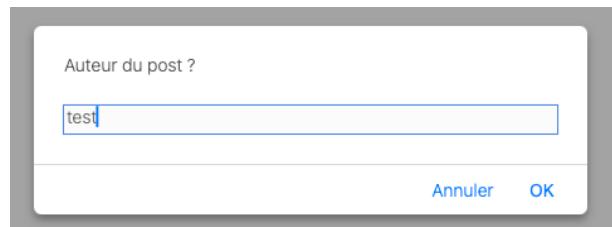
J'aime ❤️      Réinitialiser

Bob

J'adore React

❤️ 2 J'aime

J'aime ❤️      Réinitialiser



# ReactBook

Ajouter un Post

**Alice**

Premier post !

❤️ 5 J'aime

J'aime ❤️      Réinitialiser

**Bob**

J'adore React

❤️ 2 J'aime

J'aime ❤️      Réinitialiser

**test**

Test contenu

❤️ 0 J'aime

J'aime ❤️      Réinitialiser

14)

Le tableau de dépendances [posts] fait exécuter l'effet à chaque changement de posts.

Si vide [] → l'effet s'exécute une seule fois au montage.

Si omis → l'effet s'exécute à chaque re-render, même si rien n'a changé.

15)

Le code dans useEffect() s'exécute :

1. **Au montage** du composant (premier rendu).
2. **À chaque mise à jour** des dépendances.
3. **Au démontage** si un nettoyage (return) est défini.

16)

Le return dans un useEffect sert à définir une fonction de nettoyage qui s'exécute avant le prochain effet ou au démontage du composant. Il permet de libérer des ressources, comme supprimer des timers, des écouteurs d'événements ou annuler des requêtes, pour éviter les fuites mémoire.

## Requêtes API et Cycle de Vie

### Leanne Graham

Email : Sincere@april.biz

Entreprise : Romaguera-Crona

## ReactBook

Ajouter un Post

### Alice

Premier post !

 5 J'aime

J'aime  Réinitialiser

### Bob

J'adore React

 2 J'aime

J'aime  Réinitialiser

20)

Si on oublie le tableau de dépendances, le useEffect s'exécute à chaque re-render, ce qui relance la requête en boucle infinie.

21)

On utilise un AbortController et, dans le return du useEffect, on appelle controller.abort(). Cela annule la requête si le composant se démonte avant la réponse, évitant les erreurs et les mises à jour sur un composant non monté.

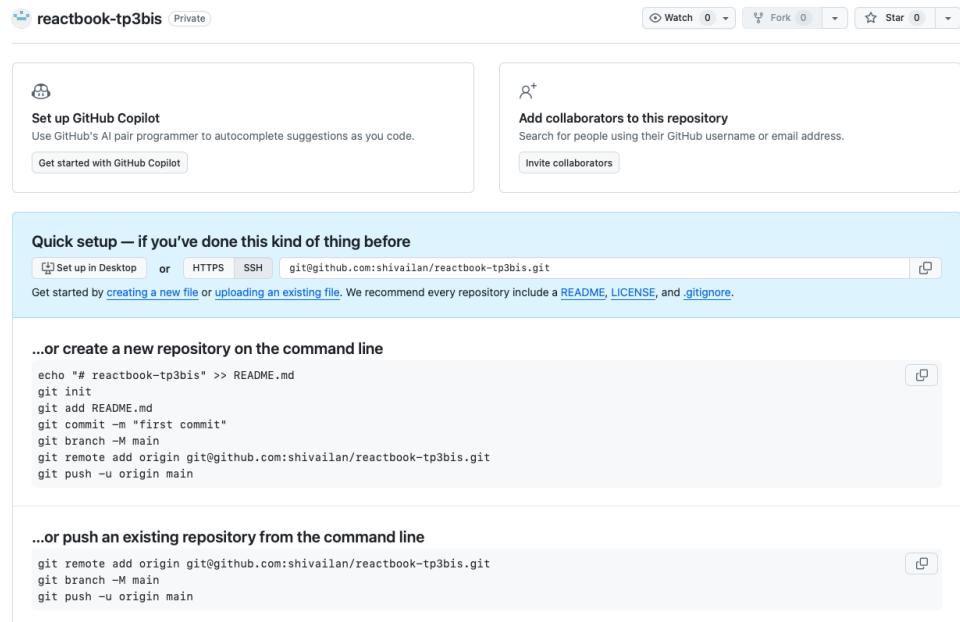
## React JS TP3bis

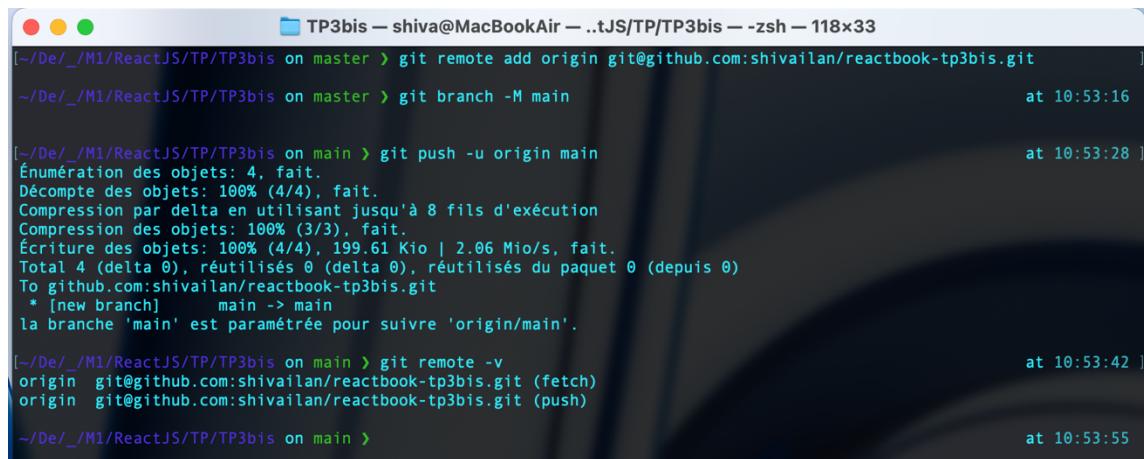
### Partie 2 : Configuration Git et GitHub

1,2,3 :

```
1415 git init
1416 ls
1417 touch .gitignore\n
1418 ls
1419 touch .gitignore\n
1420 node_modules/\n\ndist/\n.n.env\n.DS_Store\n
1421 clear
1422 ls
1423 ls -al
1424 git add .\n
1425 git commit -m "Initial commit"\n
```

4 :





```
TP3bis — shiva@MacBookAir — ..tJS/TP/TP3bis — -zsh — 118x33
[~/De/_/M1/ReactJS/TP/TP3bis on master > git remote add origin git@github.com:shivailan/reactbook-tp3bis.git
~/De/_/M1/ReactJS/TP/TP3bis on master > git branch -M main
[~/De/_/M1/ReactJS/TP/TP3bis on main > git push -u origin main
Énumération des objets: 4, fait.
Décompte des objets: 100% (4/4), fait.
Compression par delta en utilisant jusqu'à 8 fils d'exécution
Compression des objets: 100% (3/3), fait.
Écriture des objets: 100% (4/4), 199.61 Kio | 2.06 Mio/s, fait.
Total 4 (delta 0), réutilisés 0 (delta 0), réutilisés du paquet 0 (depuis 0)
To github.com:shivailan/reactbook-tp3bis.git
 * [new branch]      main -> main
la branche 'main' est paramétrée pour suivre 'origin/main'.
[~/De/_/M1/ReactJS/TP/TP3bis on main > git remote -v
origin  git@github.com:shivailan/reactbook-tp3bis.git (fetch)
origin  git@github.com:shivailan/reactbook-tp3bis.git (push)
~/De/_/M1/ReactJS/TP/TP3bis on main >
```

## Flux de Travail (GitFlowSimplifié)

5 :



```
TP3bis — shiva@MacBookAir — ..tJS/TP/TP3bis — -zsh — 118x33
[~/De/_/M1/ReactJS/TP/TP3bis on main > git checkout -b feature/partie3
Basculement sur la nouvelle branche 'feature/partie3'
~/De/_/M1/R/TP/TP3bis on feature/partie3 >
```

10 :

Elles permettent d'isoler le développement, éviter les conflits, tester sans impacter le code principal et garder un historique clair.

## Partie 3 : Gestion des Événements et Communication

1,2,3,4 :

# ReactBook - Partie 3

10

Valeur saisie : 10

5 :

Un SyntheticEvent est une version normalisée d'un événement DOM, gérée par React pour fonctionner de manière uniforme sur tous les navigateurs, alors qu'un événement DOM classique dépend directement du navigateur.

6 :

React utilise ses événements synthétiques pour garantir un comportement uniforme sur tous les navigateurs, améliorer les performances grâce au pooling d'événements, et faciliter la gestion du cycle de vie des composants.

```
~/De/_/M1/R/T/TP3bis on feature/partie3 !1 ?8 > git add . at 11:08:07

[~/De/_/M1/R/T/TP3bis on feature/partie3 +14 > git commit -m "Ajout du composant InputLogger avec gestion des événements React"

[feature/partie3 d420990] Ajout du composant InputLogger avec gestion des événements React
 14 files changed, 3095 insertions(+)
 create mode 100644 README.md
 create mode 100644 eslint.config.js
 create mode 100644 index.html
 create mode 100644 package-lock.json
 create mode 100644 package.json
 create mode 100644 public/vite.svg
 create mode 100644 src/App.css
 create mode 100644 src/App.jsx
 create mode 100644 src/assets/react.svg
 create mode 100644 src/components/InputLogger.jsx
 create mode 100644 src/index.css
 create mode 100644 src/main.jsx
 create mode 100644 vite.config.js
```

## Formulaires Contrôlés et Prévention

7,8,9,10 :

# ReactBook - Partie 3

50

Valeur saisie : 50

Email : shiva77127@gmail.com

Mot de passe : \*\*\*\*\*

Se connecter

Connexion réussie !

11 :

Pour éviter que la page se recharge et permettre à React de gérer le formulaire sans perdre l'état de l'application.

```
[~/De/_/M1/R/T/TP3bis on feature/partie3 !1 ?1 > git add . at 11:23:28 ]

[~/De/_/M1/R/T/TP3bis on feature/partie3 +2 > git commit -m "Ajout du LoginForm et finalisation InputLogger pour la Partie 3"
[feature/partie3 189a0ca] Ajout du LoginForm et finalisation InputLogger pour la Partie 3
 2 files changed, 47 insertions(+)
 create mode 100644 src/components/LoginForm.jsx

~/De/_/M1/R/T/TP3bis on feature/partie3 > █ at 11:24:12
```

Communication Parent → Enfant → Parent (Remontée d'État)

12,13,14,15 :

## ReactBook - Partie 3

10

Valeur saisie : 10

Email : shiva77127@gmail.com

Mot de passe : \*\*\*\*\*

Se connecter

### Message Board

Tape ton message...

- Bonjour

```
[~/De/_/M1/R/T/TP3bis on feature/partie3 !1 ?2 > git add .] at 11:29:37
[~/De/_/M1/R/T/TP3bis on feature/partie3 +3 > git commit -m "Ajout de MessageBoard et MessageForm avec gestion des messages via callback"
[feature/partie3 830eb3e] Ajout de MessageBoard et MessageForm avec gestion des messages via callback
 3 files changed, 56 insertions(+)
  create mode 100644 src/components/MessageBoard.jsx
  create mode 100644 src/components/MessageForm.jsx

[~/De/_/M1/R/T/TP3bis on feature/partie3 > git push -u origin feature/partie3] at 11:29:56
  Énumération des objets: 35, fait.
  Décompte des objets: 100% (35/35), fait.
  Compression par delta en utilisant jusqu'à 8 fils d'exécution
  Compression des objets: 100% (30/30), fait.
  Écriture des objets: 100% (33/33), 30.79 Kio | 6.16 Mio/s, fait.
  Total 33 (delta 7), réutilisés 0 (delta 0), réutilisés du paquet 0 (depuis 0)
  remote: Resolving deltas: 100% (7/7), done.
  remote:
  remote: Create a pull request for 'feature/partie3' on GitHub by visiting:
  remote:     https://github.com/shivailan/reactbook-tp3bis/pull/new/feature/partie3
  remote:
  To github.com:shivailan/reactbook-tp3bis.git
   * [new branch]      feature/partie3 -> feature/partie3
  la branche 'feature/partie3' est paramétrée pour suivre 'origin/feature/partie3'.

~/De/_/M1/R/T/TP3bis on feature/partie3 >] at 11:30:08
```

Créer la Pull Request (PR) :

# Shiva ILANCHEJIAN

shivailan Merge pull request #1 from shivailan/feature/partie3		4f87b6e · now	5 Commits
	public	Ajout du composant InputLogger avec gestion des événements	18 minutes ago
	src	Ajout de MessageBoard et MessageForm avec gestion de l'état	5 minutes ago
	.gitignore	Ajout du composant InputLogger avec gestion des événements	18 minutes ago
	README.md	Ajout du composant InputLogger avec gestion des événements	18 minutes ago
	TP3__Concepts_avancés_de_React_(1).pdf	Initial commit	46 minutes ago
	eslint.config.js	Ajout du composant InputLogger avec gestion des événements	18 minutes ago
	index.html	Ajout du composant InputLogger avec gestion des événements	18 minutes ago
	package-lock.json	Ajout du composant InputLogger avec gestion des événements	18 minutes ago
	package.json	Ajout du composant InputLogger avec gestion des événements	18 minutes ago
	vite.config.js	Ajout du composant InputLogger avec gestion des événements	18 minutes ago

**9, Fusionnez la PR dans la branche main puis supprimer cette branche :**

```
~/De/_/M1/R/T/TP3bis on feature/partie3 > git checkout main at 11:35:45
Basculement sur la branche 'main'
Votre branche est à jour avec 'origin/main'.

~/De/_/M1/R/TP/TP3bis on main ?1 > git pull origin main at 11:35:46

remote: Enumerating objects: 1, done.
remote: Counting objects: 100% (1/1), done.
remote: Total 1 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
Dépaquetage des objets: 100% (1/1), 936 octets | 936.00 Kio/s, fait.
Depuis github.com:shivailan/reactbook-tp3bis
 * branch      main      -> FETCH_HEAD
   86a533e..4f87b6e  main      -> origin/main
Mise à jour 86a533e..4f87b6e
Fast-forward
 .gitignore          |  24 +
 README.md           |  16 +
 eslint.config.js    |  29 +
 index.html          |  13 +
 package-lock.json   | 2826 ++++++=====
 package.json         |  27 +
 public/vite.svg     |    1 +
 src/App.css          |   42 +
 src/App.jsx          |   16 +
 src/assets/react.svg|    1 +
 src/components/InputLogger.jsx |  19 +
 src/components/LoginForm.jsx  |  45 +
 src/components/MessageBoard.jsx |  26 +
 src/components/MessageForm.jsx |  28 +
 src/index.css        |  68 ++
 src/main.jsx         |   10 +
 vite.config.js       |    7 +
17 files changed, 3198 insertions(+)
create mode 100644 README.md
create mode 100644 eslint.config.js
create mode 100644 index.html
create mode 100644 package-lock.json
create mode 100644 package.json
create mode 100644 public/vite.svg
create mode 100644 src/App.css
create mode 100644 src/App.jsx
create mode 100644 src/assets/react.svg
create mode 100644 src/components/InputLogger.jsx
create mode 100644 src/components/LoginForm.jsx
create mode 100644 src/components/MessageBoard.jsx
create mode 100644 src/components/MessageForm.jsx
create mode 100644 src/index.css
create mode 100644 src/main.jsx
create mode 100644 vite.config.js

~/De/_/M1/R/TP/TP3bis on main ?1 > git branch -d feature/partie3 at 11:35:54
Branche feature/partie3 supprimée (précédemment 830eb3e).

~/De/_/M1/R/TP/TP3bis on main ?1 > █ at 11:35:57
```

**Partie 4**

:

**Contexte global et état partagé**

**Gestion d'État Structuré et Avancé**

1, 2, 3, 4 :

## ReactBook - Partie 4

10

Valeur saisie : 10

Email : shiva77127@gmail.com

Mot de passe : \*\*\*\*\*

[Se connecter](#)

### Message Board

Tape ton message... [Publier](#)

- Bonjour

### Feed

[Ajouter un post](#)

**Alice**

Hello world!

[J'aime](#) [Supprimer](#)

**Bob**

React is cool!

[J'aime](#) [Supprimer](#)

5 :

Parce que `useReducer` centralise la logique de mise à jour d'un état complexe, comme un tableau d'objets post avec plusieurs actions (ajout, like, suppression), ce qui rend le code plus clair, prévisible et facile à maintenir qu'avec plusieurs `useState` séparés.

6 :

La fonction `reducer` est une fonction pure qui reçoit l'état actuel et une action, et retourne un nouvel état. Elle définit comment l'état doit évoluer en réponse aux actions dispatchées, centralisant toute la logique de mise à jour de l'état.

```
[~/De/_/M1/R/TP3bis on main !6 ?3 > git add . at 11:49:41]
[~/De/_/M1/R/TP3bis on main +16 > git commit -m "Partie 4 : Feed avec useReducer (ADD_POST, TOGGLE_E_LIKE, DELETE_POST)" at 11:50:03

[main 9c4b2e8] Partie 4 : Feed avec useReducer (ADD_POST, TOGGLE_E_LIKE, DELETE_POST)
 16 files changed, 21629 insertions(+), 43 deletions(-)
  create mode 100644 .vite/deps/_metadata.json
  create mode 100644 .vite/deps/chunk-TDVGZL.js
  create mode 100644 .vite/deps/chunk-TDVGZL.js.map
  create mode 100644 .vite/deps/package.json
  create mode 100644 .vite/deps/react-dom_client.js
  create mode 100644 .vite/deps/react-dom_client.js.map
  create mode 100644 .vite/deps/react.js
  create mode 100644 .vite/deps/react.js.map
  create mode 100644 src/components/Feed.jsx
  create mode 100644 src/feedReducer.js

~/De/_/M1/R/TP3bis on main +1 >
```

## Hooks Personnalisés (Custom Hooks) et Logique de Formulaire

1, 2 :

### ReactBook - Partie 4

Tapez ici...

Valeur saisie :

Email :

Mot de passe :

### Message Board

Tape ton message...

### Feed

**Bob**

React is cool!

**Nouvel Auteur**

Contenu du nouveau post

3 :

useFormInput factorise la logique de gestion d'un champ de formulaire (onChange + mise à jour de la valeur), mais chaque appel crée son propre état indépendant. Donc si deux composants utilisent ce hook, ils ont chacun leur propre variable d'état, ils ne partagent rien entre eux.

4 :

Un Hook comme useFormInput peut utiliser d'autres hooks (useState, useEffect, etc.) et est lié au cycle de vie du composant React, alors qu'une simple fonction JS ne peut pas gérer d'état React ni réagir aux mises à jour du composant.

5 :

Un Hook doit commencer par use pour que React le reconnaisse comme un Hook et applique correctement ses règles (ordre d'appel, association à un composant, suivi de l'état).

## Gestion de l'état global (Context API)

```
~/De/_/M1/R/T/TP3bis on partie4-context !4 > git add .                                at 12:27:29 ]  
~/De/_/M1/R/T/TP3bis on partie4-context +4 > git commit -m "Gestion de l'état global (Context API)" ]  
erreur : le spécificateur de chemin '-' ne correspond à aucun fichier connu de git  
erreur : le spécificateur de chemin 'm' ne correspond à aucun fichier connu de git  
erreur : le spécificateur de chemin 'Gestion de l'état global (Context API)' ne correspond à aucun fichier connu de git  
~/De/_/M1/R/T/TP3bis on partie4-context +4 > git commit -m "Gestion de l'état global (Context API)" [partie4-context b3f1ffc] Gestion de l'état global (Context API)  
4 files changed, 26 insertions(+), 4 deletions(-)  
~/De/_/M1/R/T/TP3bis on partie4-context >                                         at 12:27:48
```

## ReactBook

Connecté en tant que : Invité

[Changer de thème](#)

### Feed

[Ajouter un post](#)

**Alice**

Hello world!

[J'aime](#) [Supprimer](#)

**Bob**

React is cool!

[J'aime](#) [Supprimer](#)

9 :

Il faut imbriquer les Providers pour que tous les composants enfants puissent accéder aux contextes via useContext. Sans cette imbrication, un composant ne pourrait pas consommer le contexte correspondant.

10 :

Avoir deux contextes séparés permet :

- Une meilleure lisibilité et maintenance du code.
- De limiter les re-renders inutiles : changer le thème ne force pas un re-render des composants qui n'utilisent que l'utilisateur, et vice versa.

11 :

Si plusieurs Providers identiques sont imbriqués, le composant le plus profond consommera la valeur du Provider le plus proche dans la hiérarchie. Les Providers supérieurs sont ignorés pour ce composant.

## Partie 5 :

```
~/Desktop/_/_/_/Etudes/M1/ReactJS/TP/TP3bis on main > git push -u origin partie5-routing
erreur : le spécificateur de référence source partie5-routing ne correspond à aucune référence
erreur : impossible de pousser des références vers 'github.com:shivailan/reactbook-tp3bis.git'
~/De/_/E/M1/ReactJS/TP/TP3bis on main :3 > git branch
(~/_/_/_/E/M1/ReactJS/TP/TP3bis on main :3) > git branch
~/De/_/E/M1/ReactJS/TP/TP3bis on main :3 > git checkout -b partie5-routing
Basculement sur la nouvelle branche 'partie5-routing'
~/De/_/E/M/R/TP/TP3bis on partie5-routing > git branch
(~/_/_/_/E/M1/ReactJS/TP/TP3bis on partie5-routing) > git add .
~/De/_/E/M/R/TP/TP3bis on partie5-routing > git commit -m "Partie 5 : ajout du routage et des ProtectedRoute"
Sur la branche partie5-routing
rien à valider, la copie de travail est propre
~/De/_/E/M1/ReactJS/TP/TP3bis on partie5-routing > git push -u origin partie5-routing
Énumération des objets: 28, fait.
Décompte des objets: 100% (28/28), fait.
Compression par delta en utilisant jusqu'à 8 fils d'exécution
Compression des objets: 100% (20/20), fait.
Écriture des objets: 100% (20/20), 7.78 Mio | 3.28 Mio/s, fait.
Total 20 (delta 4), réutilisés 0 (delta 0), réutilisés du paquet 0 (depuis 0)
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
remote:
remote: Create a pull request for 'partie5-routing' on GitHub by visiting:
remote:     https://github.com:shivailan/reactbook-tp3bis/pull/new/partie5-routing
remote:
To github.com:shivailan/reactbook-tp3bis.git
 * [new branch]      partie5-routing -> partie5-routing
la branche 'partie5-routing' est paramétrée pour suivre 'origin/partie5-routing'.
~/De/_/E/M/R/T/TP3bis on partie5-routing >
```

## Extension TP

### Filtre :

## ReactBook

Connecté en tant que : Invité

[Changer de thème](#)

### Feed

ali

Alice

Hello world!

[♥ J'aime](#) [Supprimer](#)

Commentaires (0)

[Ajouter un commentaire...](#) [Envoyer](#)

### Ajout de commentaires :

**Alice**

Hello world!

[♥ J'aime](#)

[Supprimer](#)

**Commentaires (0)**

Hello

[Envoyer](#)

Shiva ILANCHEJIAN

## Alice

Hello world!

### Commentaires (1)

Hello

Pagination :

Charlie

Début de la pagination

Commentaires (0)

David

Post numéro 4

Commentaires (0)

## Shiva ILANCHEJIAN

Tests d'intégration avec Jtest :

Ajout d'un fichier «Feed.test.jsx » pour le test

```
[~/Desktop/reactbook-tp3bis on main !3 ?1 > npm test at 14:25:22 ]
> tp3bis@0.0.0 test
> jest --passWithNoTests

sh: jest: command not found

[~/Desktop/reactbook-tp3bis on main !3 ?1 > npm install --save-dev jest jest-environment-javascript babel-jest @babel/co
re @babel/preset-env @babel/preset-react
npm warn deprecated inflight@1.0.6: This module is not supported, and leaks memory. Do not use it. Check out tru-c
ache if you want a good and tested way to coalesce async requests by a key value, which is much more comprehensive
and powerful.
npm warn deprecated glob@7.2.3: Glob versions prior to v9 are no longer supported
npm warn deprecated abab@2.0.6: Use your platform's native atob() and btoa() methods instead
npm warn deprecated whatwg-encoding@2.0.0: Use @exodus/bytes instead for a more spec-conformant and faster impleme
ntation
npm warn deprecated domexception@4.0.0: Use your platform's native DOMException instead

added 382 packages, and audited 560 packages in 12s

66 packages are looking for funding
  run 'npm fund' for details

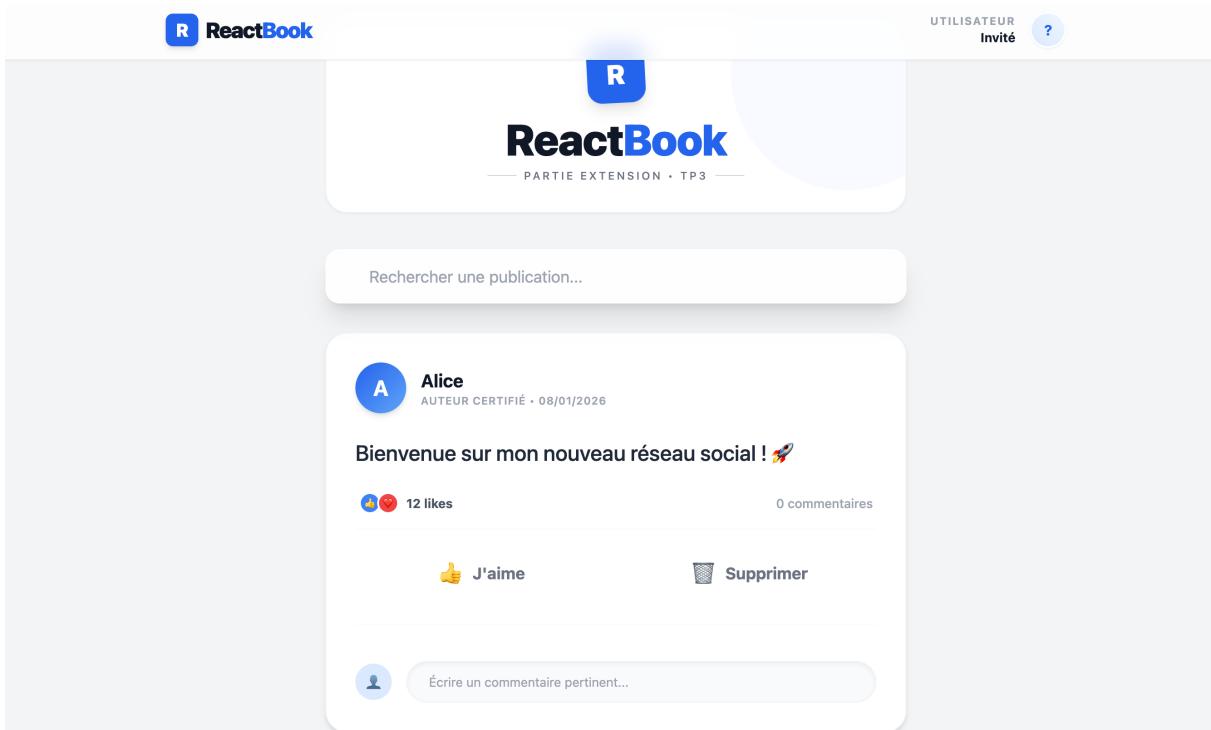
found 8 vulnerabilities
```

```
[~/Desktop/reactbook-tp3bis on main !4 ?1 > npm test at 14:30:42 ]
> tp3bis@0.0.0 test
> jest --env=jsdom --passWithNoTests

PASS  src/components/Feed.test.jsx
  Tests d'Intégration ReactBook
    ✓ vérifie que la liste se filtre lors d'une recherche (60 ms)
    ✓ vérifie l'ajout d'un nouveau commentaire (7 ms)

Test Suites: 1 passed, 1 total
Tests:       2 passed, 2 total
Snapshots:  0 total
Time:        0.865 s, estimated 1 s
Ran all test suites.
```

Le design avec l'utilisation de tailwind CSS :



## DOCKER :

```

[~/Desktop/reactbook-tp3bis on main ?] > docker build -t reactbook-frontend:1.0 .
[+] Building 17.9s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 484B
=> [internal] load metadata for docker.io/library/node:22-alpine
=> [internal] load .dockerignore
=> => transferring context: 93B
[1/6] FROM docker.io/library/node:22-alpine@sha256:0340fa682d72068edf603c305bfbc10e23219fb0e40df58d9ea4d6f33a9798bf
=> => resolve docker.io/library/node:22-alpine@sha256:0340fa682d72068edf603c305bfbc10e23219fb0e40df58d9ea4d6f33a9798bf
=> sha256:11118a8e50bb2ee57ce661a68336ae987af986f9b42f77d4e187799ed7197f8 1.72kB / 1.72kB
=> sha256:b17305803b632762c3f033ad3df0203a93d753d644fb38bc60012c449e329 6.54kB / 6.54kB
=> sha256:f604fb9446345fcad2d026eac181fe6c0a919c8a4fcccc3bea5deb7f6df67e 4.20MB / 4.20MB
=> sha256:03ae695e7697f612446465b65f70d9423d774ccfc0436ba741a6fff45a0e7f 52.24MB / 52.24MB
=> sha256:45b4862b6891b640b4dd4cd5d85ef65840da8a4e00ff5734fa7cf1b0eaa59479c 1.26MB / 1.26MB
=> sha256:10c60c56762a4b48671956ae4ebfc29a11040543b8b3672631a03309309c78 446B / 446B
=> => extracting sha256:03ae695e7697f612446465b65f70d9423d774ccfc0436ba741a6fff45a0e7f
=> => extracting sha256:45b4862b6891b640b4d4c5d85ef65840da8a4e00ff5734fa7cf1b0eaa59479c
=> => extracting sha256:10c60c56762a4b48671956ae4ebfc29a11040543b8b3672631a03309309c78
=> [internal] load build context
=> => transferring context: 2.16kB
[2/6] WORKDIR /app
[3/6] COPY package*.json .
[4/6] RUN npm install
[5/6] COPY ...
[6/6] RUN npm run build
=> exporting to image
=> exporting layers
=> => writing image sha256:e3f39b90c85d9c4b1a77cf6a96485ee27fd23772df5ef15053e65e4e82577d16
=> => naming to docker.io/library/reactbook-frontend:1.0

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/vqzuqw664ua517kxia2up48of

What's next:
  View a summary of image vulnerabilities and recommendations --> docker scout quickview

[~/Desktop/reactbook-tp3bis on main ?] > docker run --rm -p 5173:5173 reactbook-frontend:1.0
took 19s at 11:46:17

> tp3bis@0.0.0 dev
> vite --host 0.0.0.0 --port 5173

  VITE v7.3.1 ready in 121 ms
  > Local: http://localhost:5173/
  > Network: http://172.17.0.2:5173/

```

Le docker run est maintenant fonctionnel

## Shiva ILANCHEJIAN

Build :

```
docker build -t reactbook-frontend:1.0 .
```

Run

```
docker run --rm -p 5173:5173 reactbook-frontend:1.0
```

Github Action :



J'ai joint à ce rendu une **vidéo de l'output** qui montre l'application en fonctionnement.

Pour tester l'application, voici les identifiants de connexion :

- **Login** : shiva
- **Mot de passe** : shiva