

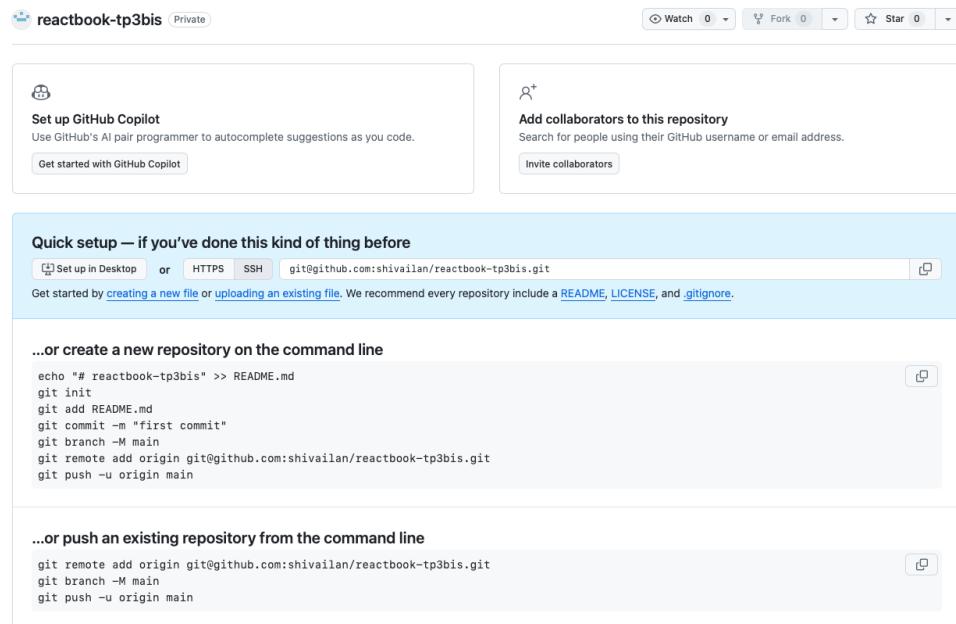
React JS TP3bis

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/\nndist/\nn.env\nn.DS_Store\n
1421 clear
1422 ls
1423 ls -al
1424 git add .
1425 git commit -m "Initial commit"\n
```

4 :



```
[~/De/_/M1/ReactJS/TP/TP3bis on master > git remote add origin git@github.com:shivailan/reactbook-tp3bis.git] at 10:53:16
~/De/_/M1/ReactJS/TP/TP3bis on master > git branch -M main

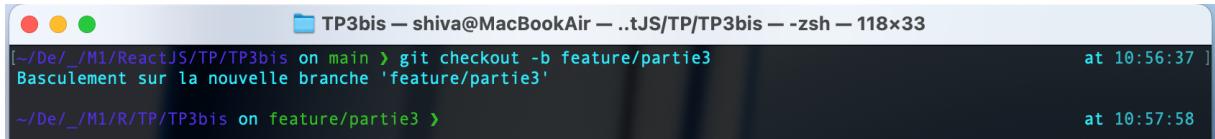
[~/De/_/M1/ReactJS/TP/TP3bis on main > git push -u origin main] at 10:53:28
Énumération des objets: 4, fait.
Décompte des objets: 100% (4/4), fait.
Compression par delta en utilisant jusqu'à 8 fils d'exécution
Compression des objets: 100% (3/3), fait.
Écriture des objets: 100% (4/4), 199.61 Kio | 2.06 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] at 10:53:42
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 >
```

at 10:56:37
at 10:57:58

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

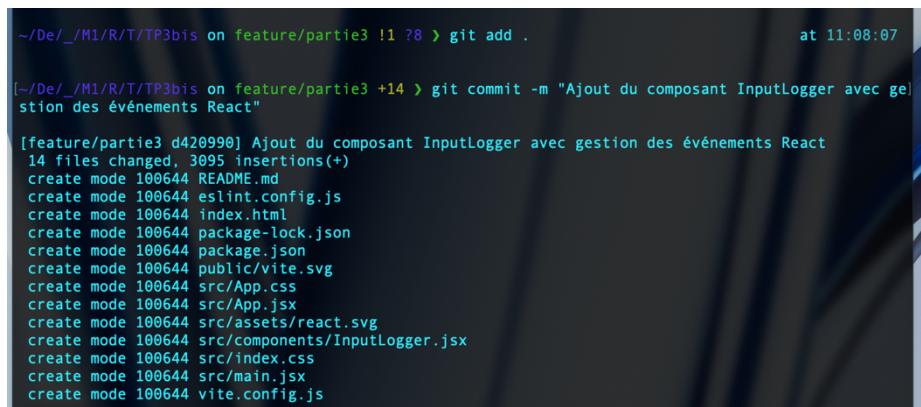
Valeur saisie : 10

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 .
[~/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

Valeur saisie : 50

Email :

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

Valeur saisie : 10

Email :

Mot de passe :

Se connecter

Message Board

Publier

- Bonjour

Shiva ILANCHEJIAN

```
[~/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) :

The screenshot shows the GitHub interface for creating a pull request. At the top, there's a placeholder 'Add a title' with a blue icon. Below it, the title 'Ajout de InputLogger, LoginForm et MessageBoard (Partie 3)' is entered. Under 'Add a description', there's a rich text editor with a 'Write' tab selected. The content area contains the following text:

Ce commit ajoute les composants de la Partie 3 :

1. InputLogger : champ texte affichant la valeur saisie en temps réel.
2. LoginForm : formulaire contrôlé avec email et mot de passe, affichage d'un message "Connexion réussie !" après soumission.
3. MessageBoard + MessageForm : composant parent stockant la liste des messages, composant enfant pour saisir un message via callback.

Below this, under 'Pour tester :', there's a list of instructions:

- Lancer le projet : npm run dev
- Tester InputLogger : taper du texte et vérifier la console.
- Tester LoginForm : saisir email et mot de passe, cliquer sur Se connecter, vérifier la console et le message temporaire.
- Tester MessageBoard : saisir un message et cliquer sur Publier, vérifier qu'il s'ajoute à la liste.

At the bottom, there are two buttons: 'Markdown is supported' and 'Paste, drop, or click to add files'. On the right, a large green button says 'Create pull request' with a dropdown arrow.

Remember, contributions to this repository should follow our [GitHub Community Guidelines](#).

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 : *****

Message Board

- Bonjour

Feed

Alice

Hello world!

Bob

React is cool!

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/TP/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_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/TP/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).