

Git 講習



Gitとは？

ファイルの編集履歴どのようにしていますか？

編集前のファイルをコピーしておく方法



ファイル名を日付や名前、変更内容等
を書いて区別できるようにする

起き得る問題点

- ・ 変更点がよくわからない
- ・ フォルダのストレージがパンパン
- ・ どのファイルが最新で
いつ誰が更新したのか不明
- ・ 共有ファイルで変更点が競合して
変更が残っていない・・・

Gitとは？

- ・バージョン管理システム
ファイルの変更履歴を管理するツール



- ・分散型システム
各ユーザが完全な履歴を持ち、オフラインでも作業ができる
- ・オープンソースソフトウェア



Gitとは？

誰がどんな変更をいつしたのか、一目瞭然

https://github.com/Hibikino-Musashi-Home/RoboCup-TDP/commits/2026_incheon/?before=7f3a9f7bd4b36ba31926a824ed312bfa68c25192+70



Merge pull request #161 from Hibikino-Musashi-Home/159-tdp-2026-object-recognition-nano-sam

Amos2610 authored 3 weeks ago · ✓ 1 / 1

Verified a90e0eb ⌂ ↗

-o Commits on Nov 7, 2025

update: r.kobayashi->our team, add:cite hsrtx(rsj2024) paper

ry0hei-kobayashi committed 3 weeks ago · ✓ 1 / 1

23f69af ⌂ ↗

change the task_planning fig

tera7718 committed 3 weeks ago · ✓ 1 / 1

3a7c4b8 ⌂ ↗

ラベルつけしたい、コンフリクトしたら削ってください。

yuzoo0226 committed 3 weeks ago · ✓ 1 / 1

b415030 ⌂ ↗

cutieに関する記述をアップデート

yuzoo0226 committed 3 weeks ago

e2398cf ⌂ ↗

文章修正

carrobo24_athome committed 3 weeks ago · ✓ 1 / 1

160c5e1 ⌂ ↗

この変更点すべてに
その時点でのファイルが
サーバに保存されている

-o Commits on Nov 6, 2025

usdのロゴを変更

Wataki0824 committed 3 weeks ago · ✓ 1 / 1

3cc850e ⌂ ↗

YOLO v11 → YOLO11

Amos2610 committed 3 weeks ago · ✓ 1 / 1

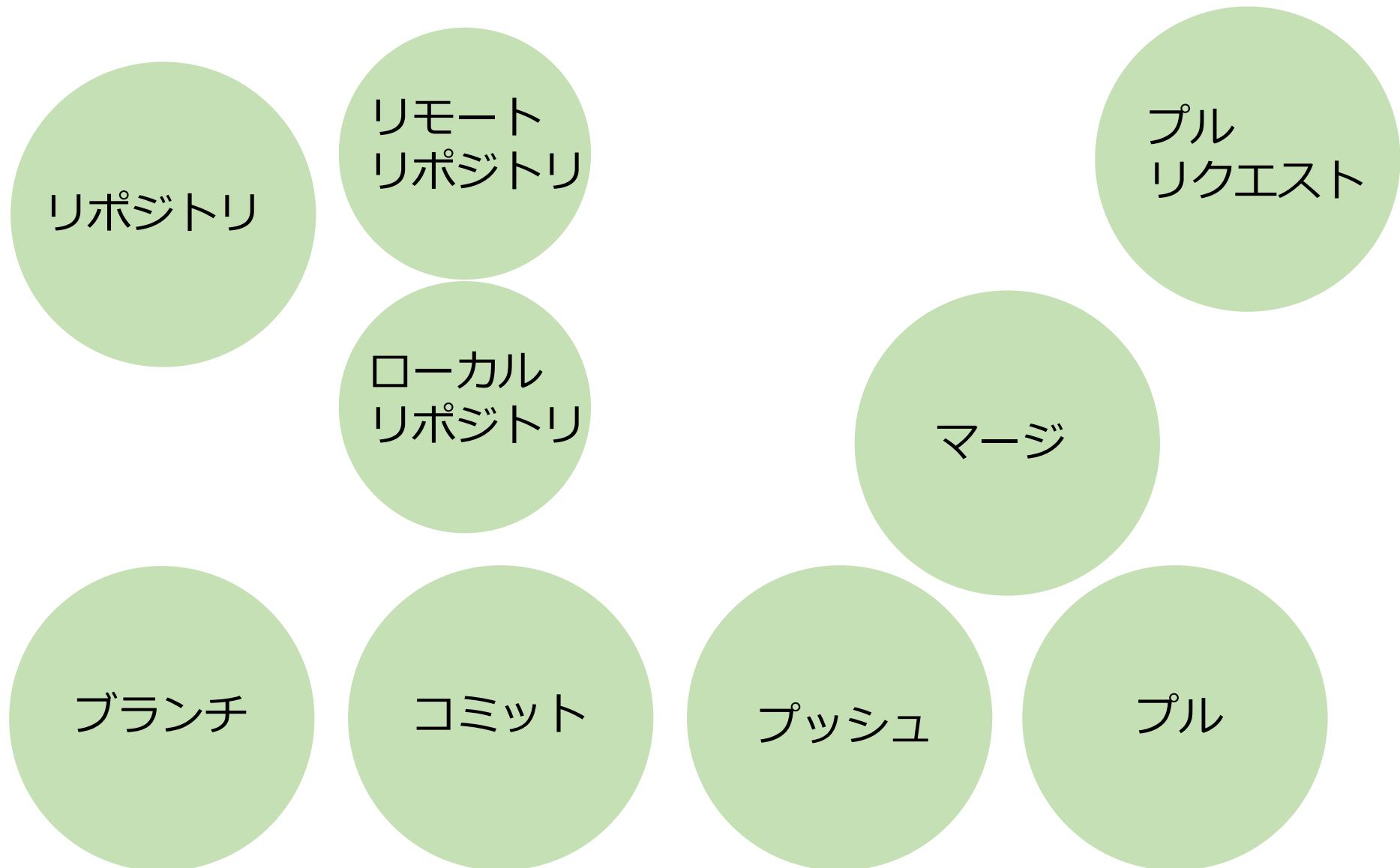
25f54e8 ⌂ ↗

初稿v2添削後の修正 (Merge from 2026_incheon-fumoto-remove-and-fix-all-check)

Amos2610 authored 3 weeks ago · ✓ 1 / 1

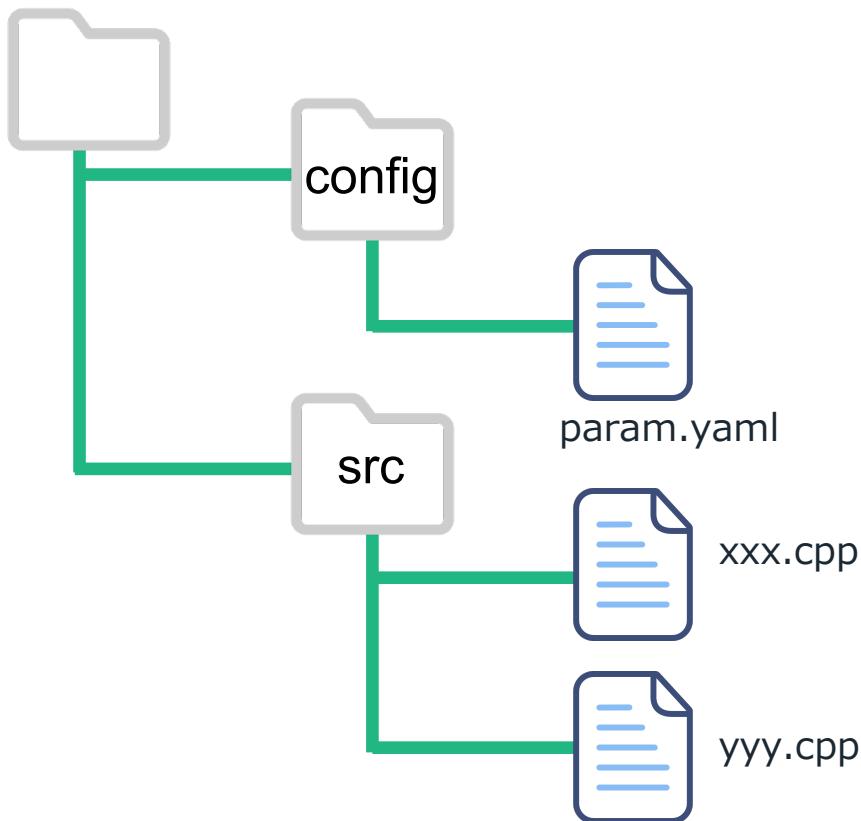
Verified 63c4932 ⌂ ↗

Gitの基本用語



リポジトリ(repository)

管理したいファイル(ソースコード)のまとめ

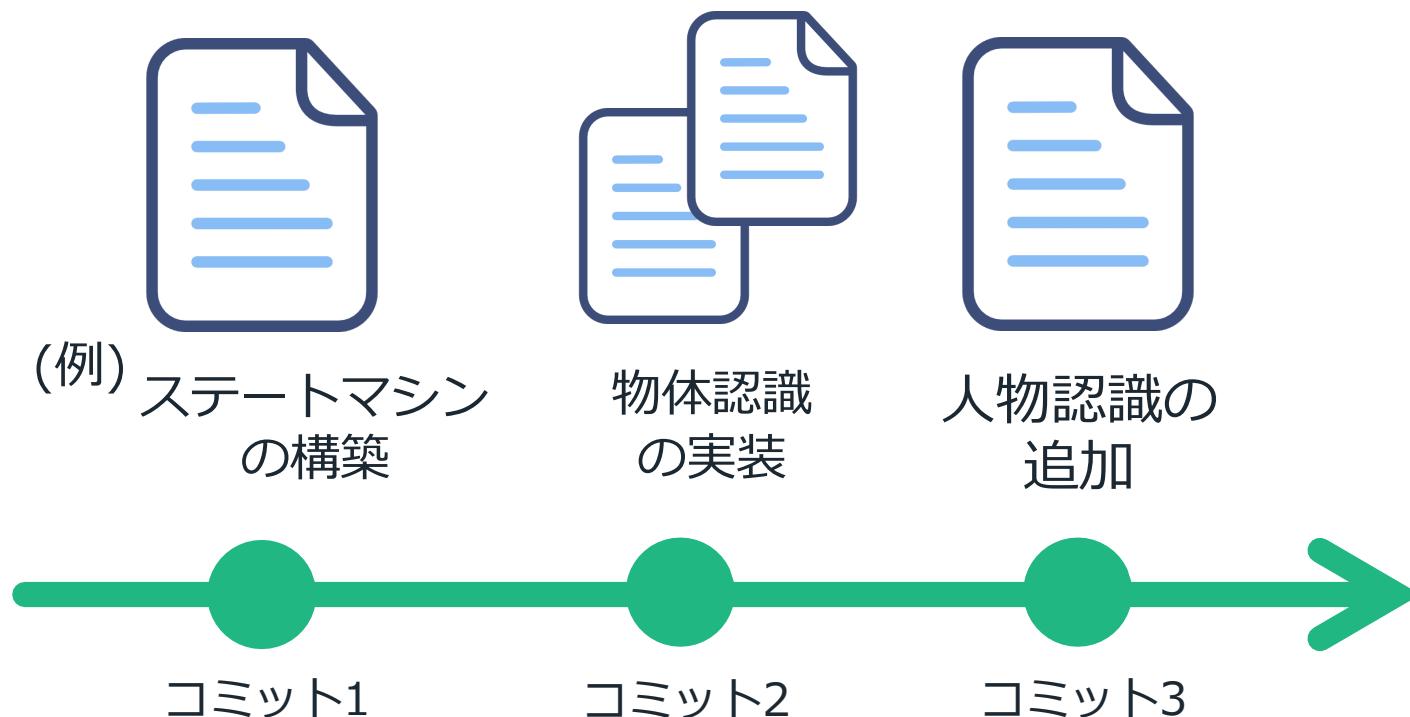


turtlebot3の例

turtlebot3 Public		
humble	4 Branches 29 Tags	Watch 63
This branch is 7 commits ahead of main .		
robotpilot	Merge pull request #1127 from ROBOTIS-GIT/main	4ab470d · last week 1,018 Commits
.github	remove rolling ci	5 months ago
docker	add nav2-route install command	2 weeks ago
turtlebot3	Bump	last month
turtlebot3_bringup	Bump	last month
turtlebot3_cartographer	Bump	last month
turtlebot3_description	Bump	last month
turtlebot3_example	Bump	last month
turtlebot3_navigation2	Bump	last month
turtlebot3_node	Bump	last month
turtlebot3_teleop	Bump	last month
.gitignore	gitignore: Update .gitignore to include .DS_Store file excl...	27 days ago
CONTRIBUTING.md	modified ament_copyright	5 years ago
ISSUE_TEMPLATE.md	update version info	3 years ago
LICENSE	modified ament_copyright	5 years ago
README.md	Modified the description of this repo	7 months ago
turtlebot3.repos	Updated the CHANGELOG and version to release binary p...	9 months ago
turtlebot3_ci.repos	Updated the CHANGELOG and version to release binary p...	9 months ago

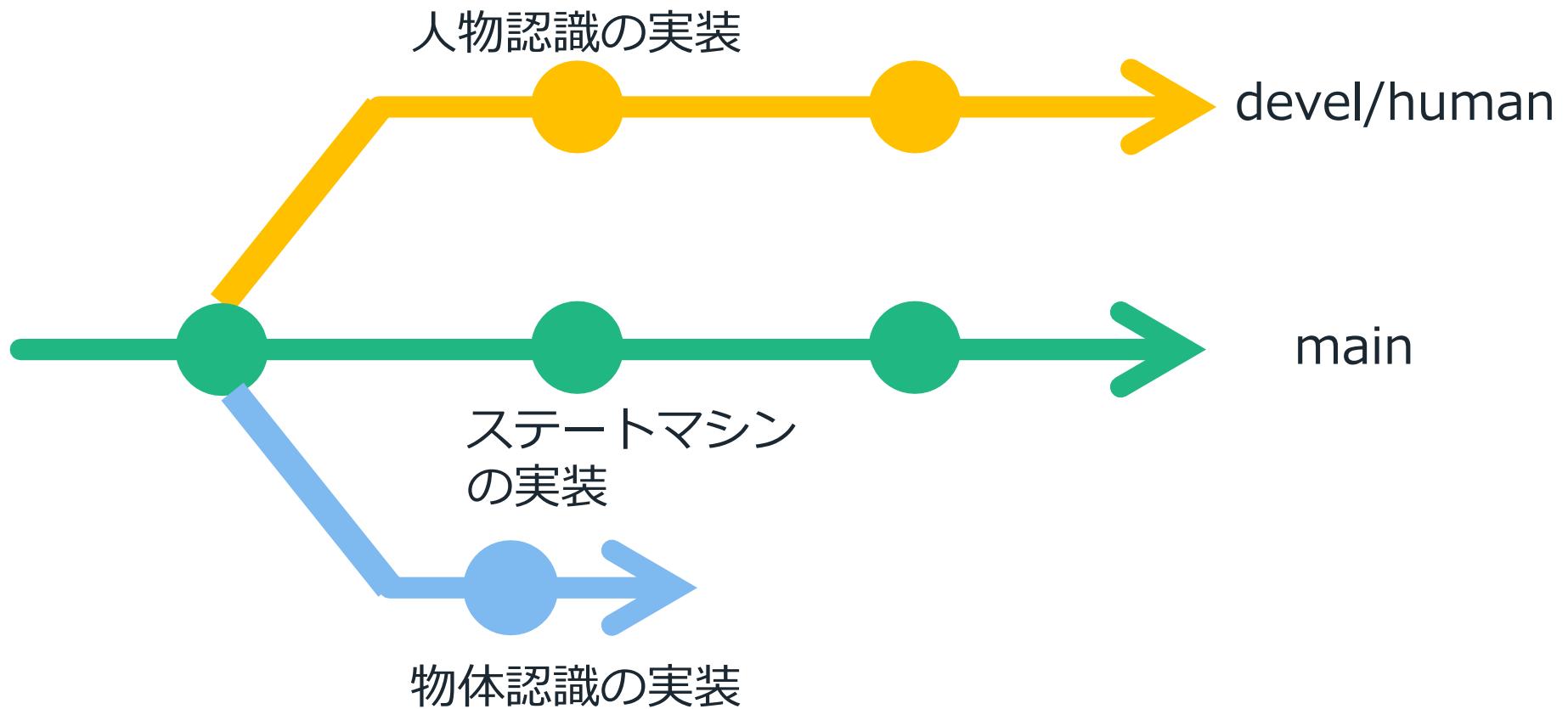
コミット(commit)

ファイルやディレクトリの追加・変更を
リポジトリに記録する操作



ブランチ(branch)

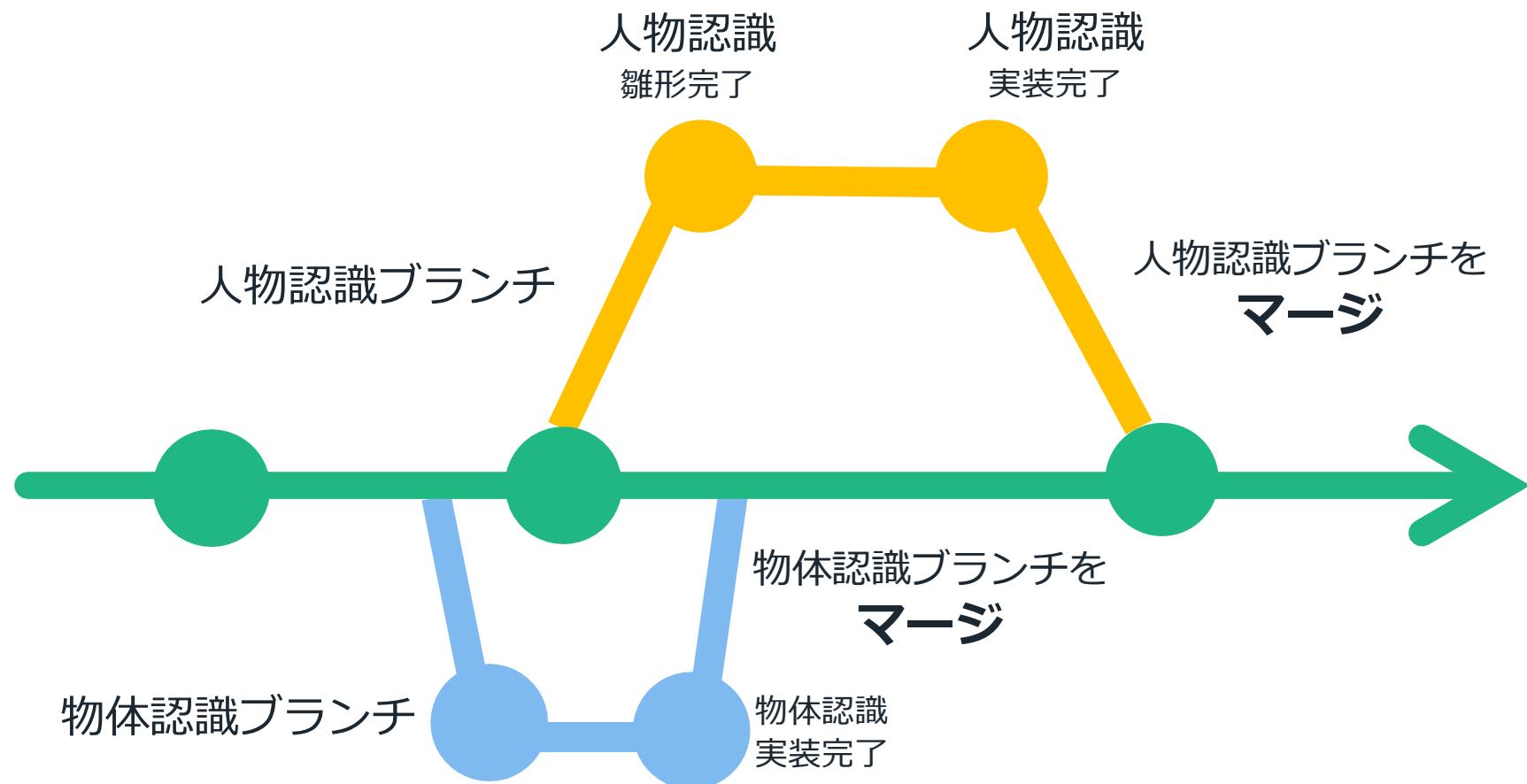
親のコミットから履歴を分岐する



同じリポジトリ中で複数の変更を同時に進めていくことができる

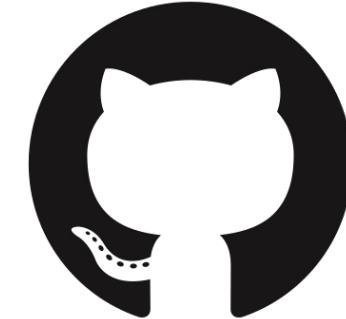
マージ(merge)

別々に進んだ履歴（ブランチ）を1つに統合する操作



GitHub

Gitで管理されたファイルを
インターネットにアップロードして
複数人での取り扱いを便利にする
プラットフォーム

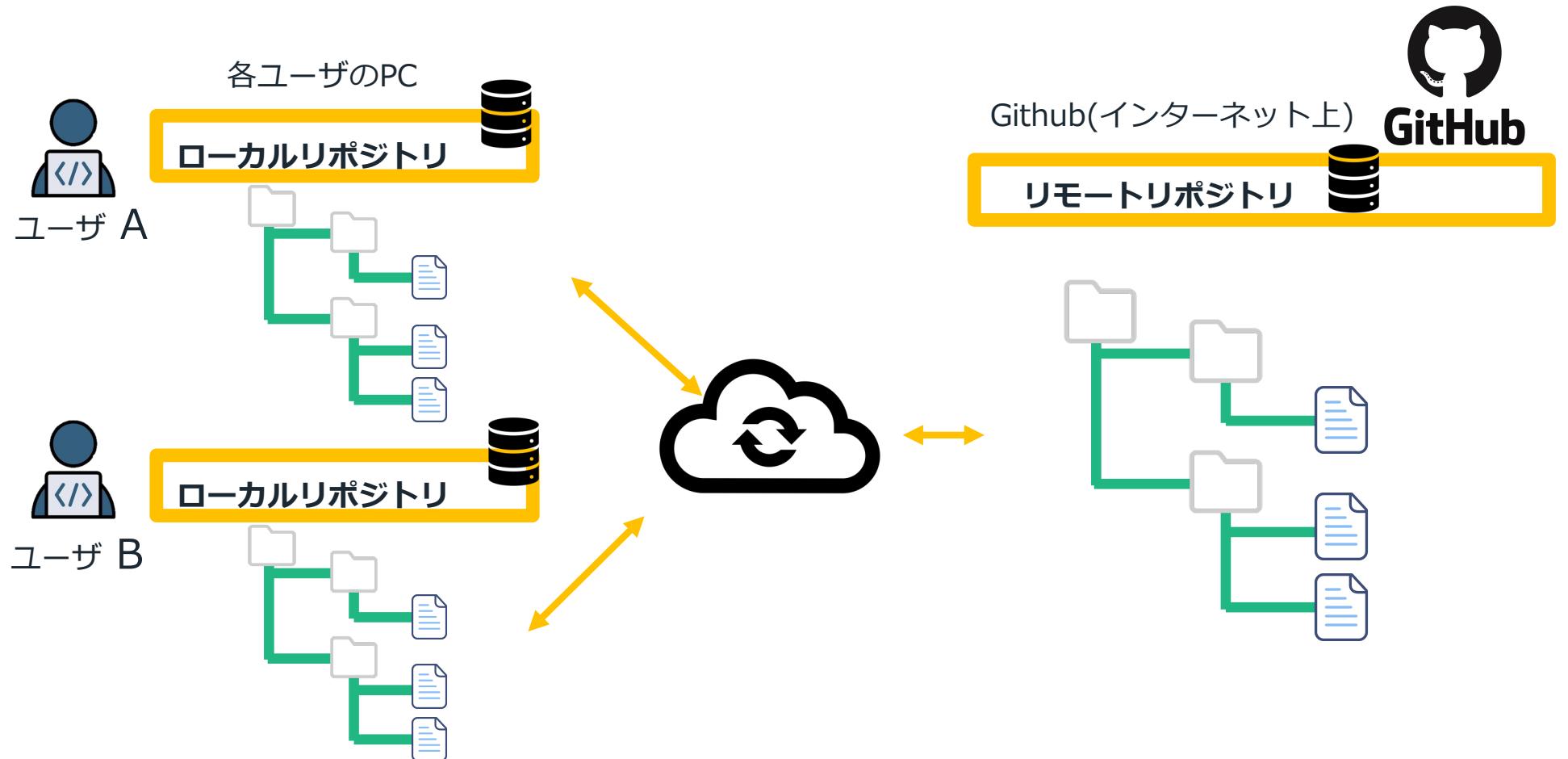


GitHub



The screenshot shows the GitHub homepage with a dark background. At the top center is the GitHub logo (a white cat silhouette). Below it, the text "開発者のための プラットフォーム" (A platform for developers) is displayed. A paragraph of Japanese text explains what GitHub is: "GitHubは、ユーザのみなさんからヒントを得て作成された開発プラットフォームです。オープンソースプロジェクトやビジネスユースまで、GitHub上にソースコードをホスティングすることで数百万人もの他の開発者と一緒にコードのレビューを行ったり、プロジェクトの管理をしながら、ソフトウェアの開発を行うことができます。" At the bottom, there is a green button with the text "GitHubに登録する" (Sign up for GitHub).

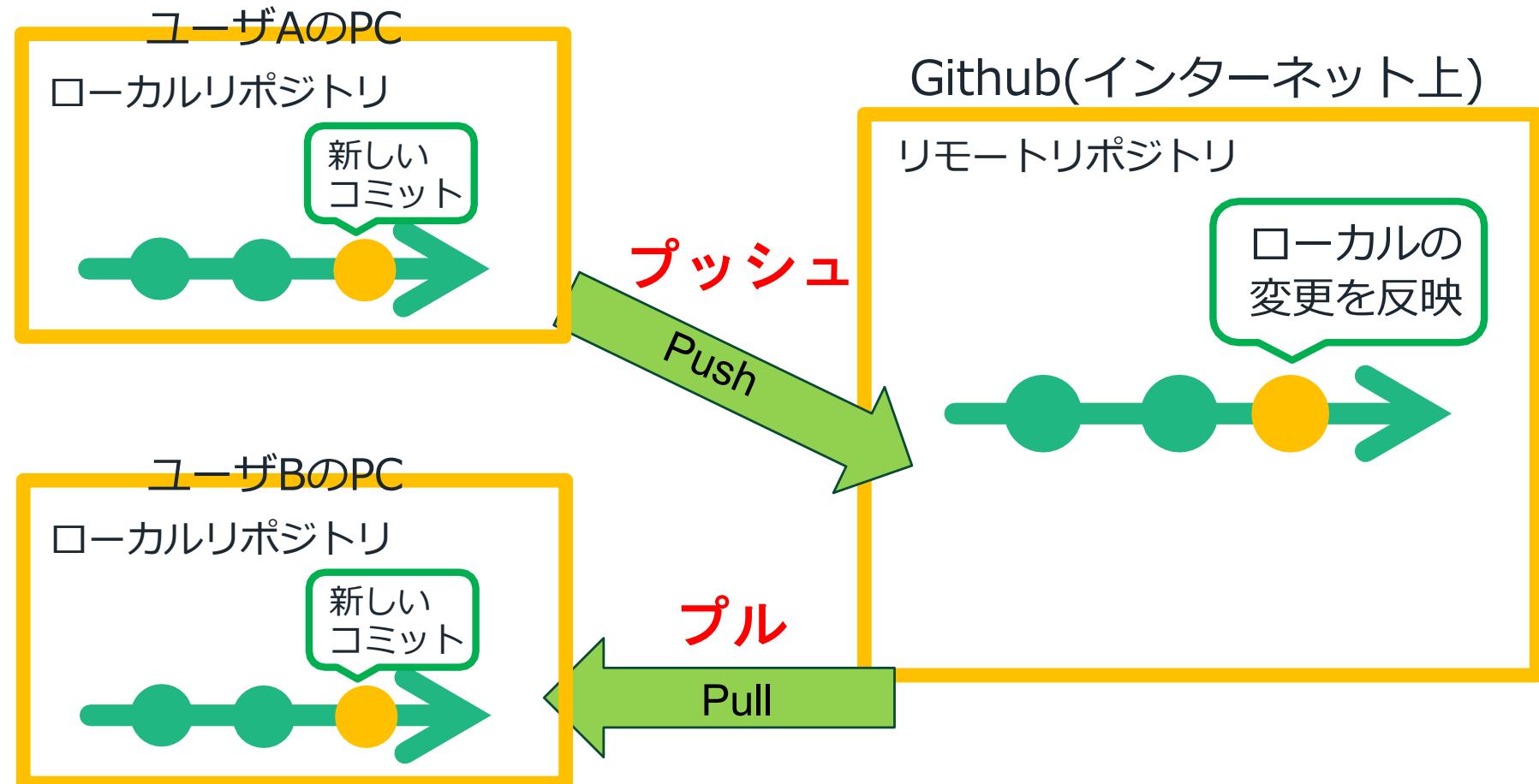
リモート・ローカル/リポジトリ



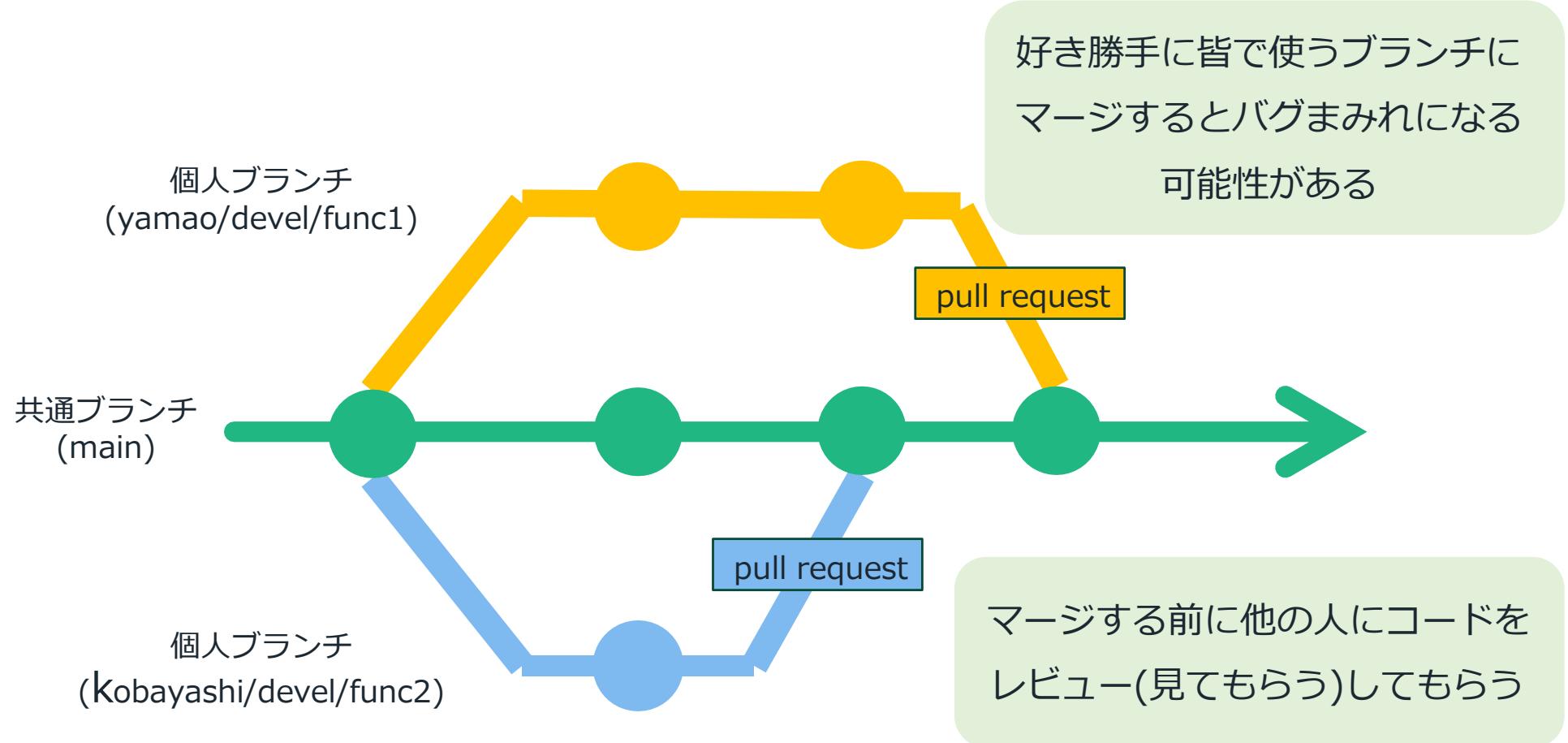
リモート・ローカル/リポジトリの利点

- ・チームでの共同作業が容易
ネットにアップロードすれば、ほかの人と容易に
コードの共有が可能になる
- ・バックアップの役割
自分のPCが壊れてもネットから
もう一度引っ張ってくれれば問題ない

プッシュ・プル

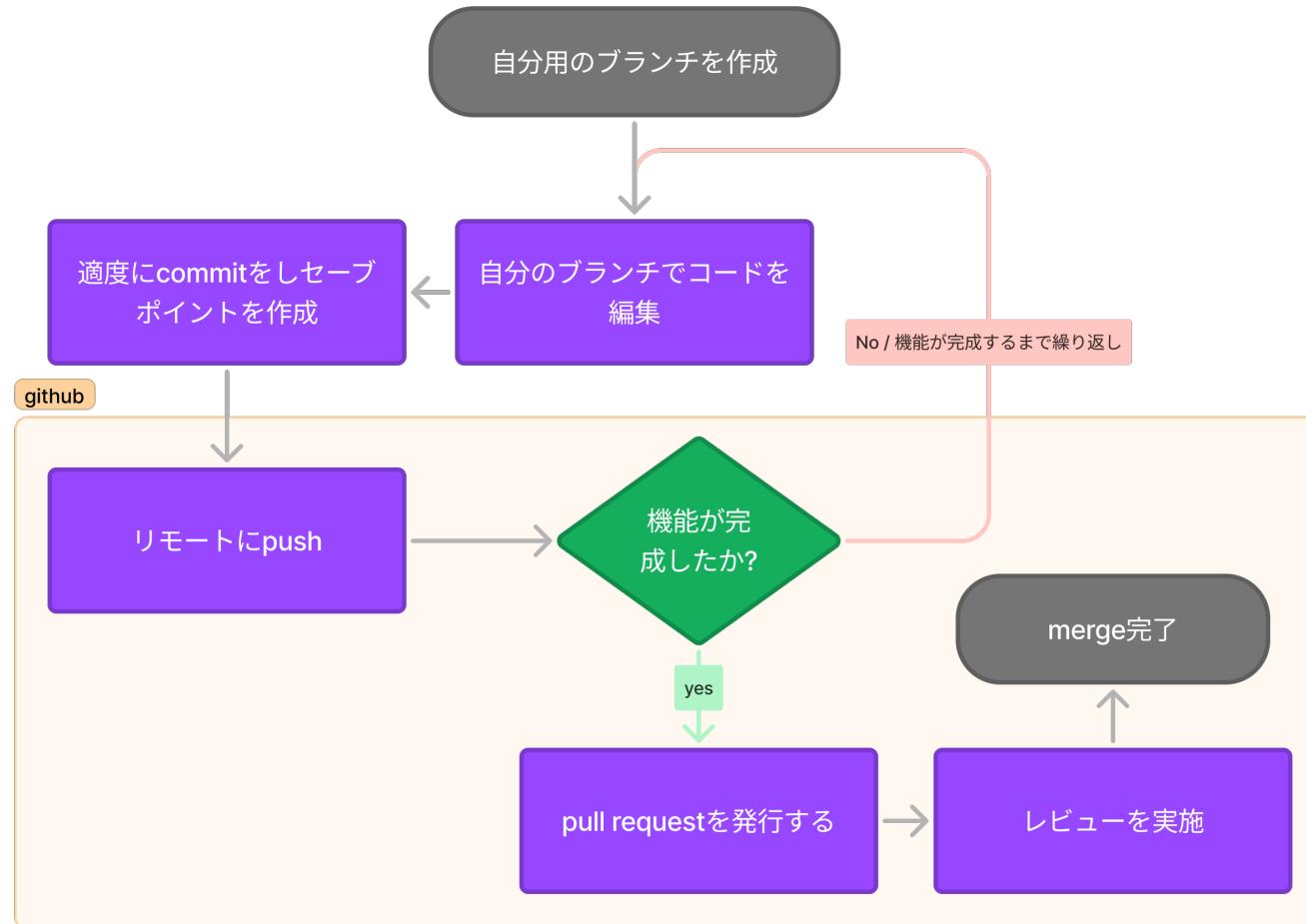


pull request



フローチャート

ブランチ作成からマージまで



Gitを扱う際の注意点

命名規則について

※ 命名規則はチームによって大きく異なる

- ・ ブランチ名

自分の名前/開発力テゴリ/開発内容 としてください

(例) yamao-devel/add_human_detection ,

yamao-hotfix/bug_fix

yamao-check/mask_filter , yamao-fix/order_bug

- ・ コミットメッセージ

開発内容が分かりやすいメッセージにしてください

(例) 人物認識の機能を追加 , マスク処理の微修正

心がけてほしいこと

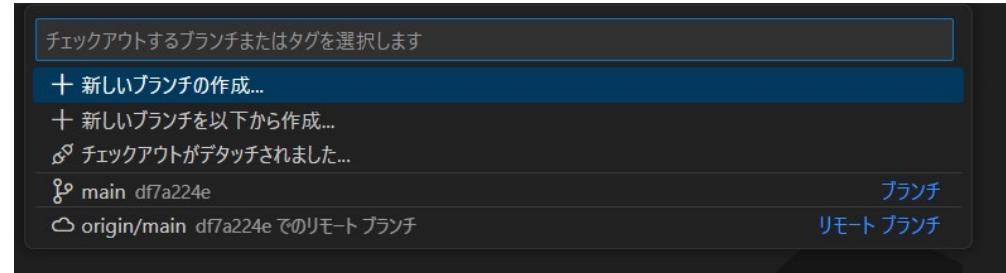
- ・プログラムに何かしらの追加をする場合は必ずブランチを切る
競合やソースコードの損失を防ぎ、バージョン管理が正しく行う
大会やデモ等で使用するブランチに直書きすると本番で動かなく
なり、事故につながる
- ・こまめにコミット、プッシュする癖をつける
こまめなコミットを積み上げることで問題発生時の切り分けが
すぐにできたり、他のユーザとの競合を防げたりする

自分のブランチを作成してみよう

VSCode

左下の“main”が現在参照している
ブランチを指しています

左下の“main”をクリックすると
メニューが表示されます



ターミナルから

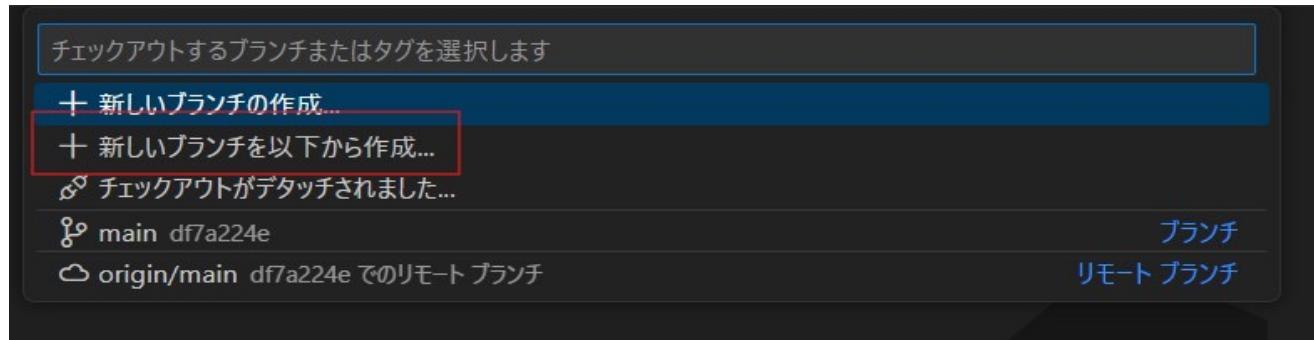
```
git checkout -b { 作成するブランチ名 }
```

例 (mainからyamao/devel/demoを作成)

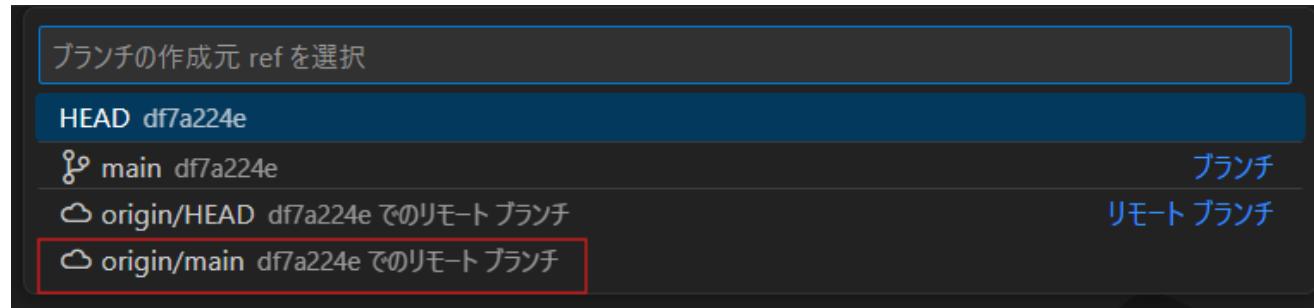
```
git checkout -b yamao/devel/demo main
```

自分のブランチを作成してみよう

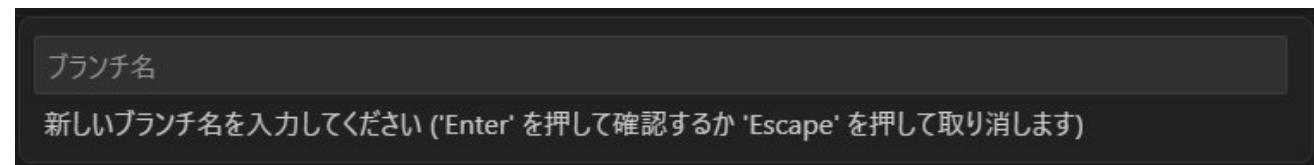
VSCode



新しいブランチを
以下から作成をクリック



リモートブランチ
の"origin/main"を選択

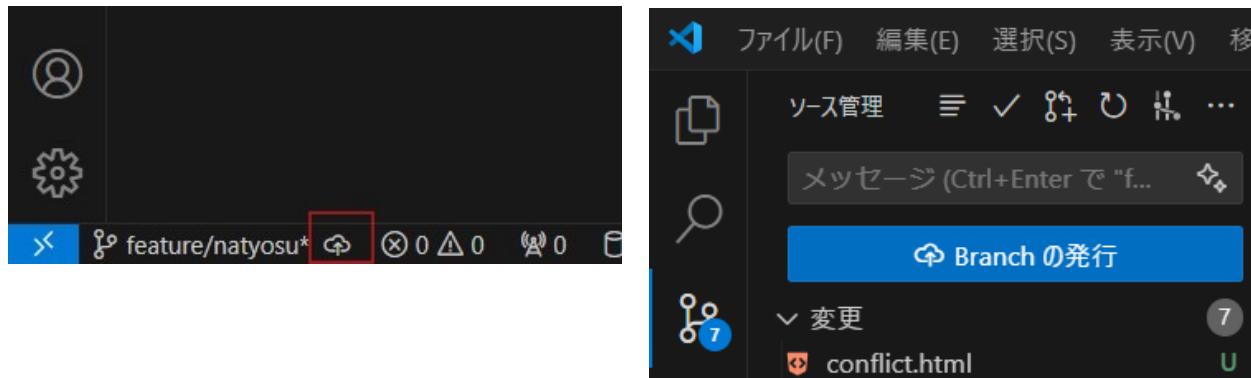


ブランチ名を入力する

自分の名前/devel/機能名

自分のブランチを作成してみよう

リモートに登録

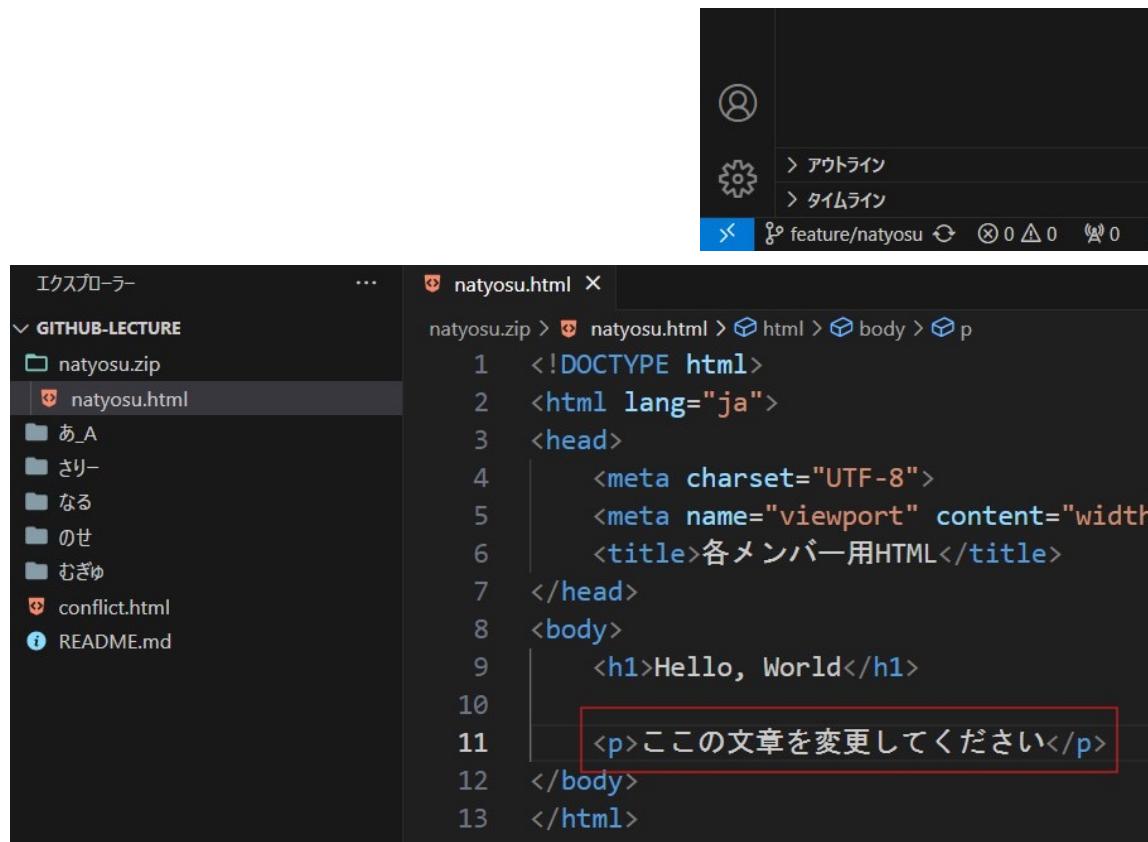


左下の雲に矢印が刺さっているマークをクリックするか、サイドバーのgit管理からbranchの発行をクリック

ターミナルから

```
git push -u origin yamao/devel/demo
```

コミットを作成してみよう

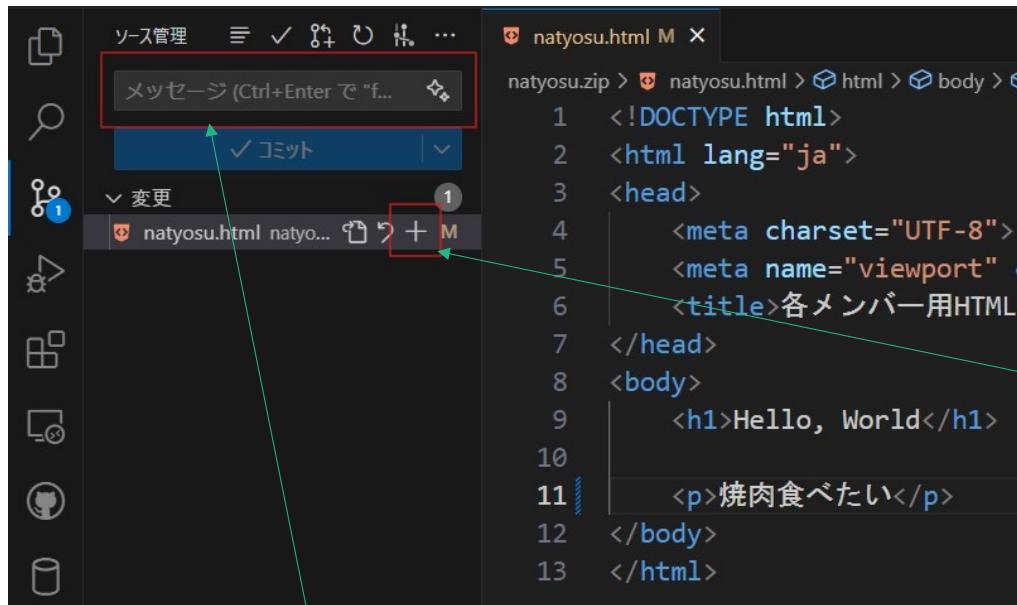


```
natyosu.zip > natyosu.html > html > body > p
1  <!DOCTYPE html>
2  <html lang="ja">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width
6          <title>各メンバー用HTML</title>
7  </head>
8  <body>
9      <h1>Hello, World</h1>
10
11     <p>ここの文章を変更してください</p>
12
13  </body>
14  </html>
```

左下で自分のブランチに移動(参照)しているかを確認する。

自分のブランチに移動できていれば、自分の名前のフォルダ内のhtmlファイルを編集して保存してください。

コミットを作成してみよう



The screenshot shows a code editor interface with a dark theme. On the left is a sidebar with various icons. In the center, there's a file viewer for 'natyosu.html' which contains the following HTML code:

```
1 <!DOCTYPE html>
2 <html lang="ja">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" c
6   <title>各メンバー用HTML<
7 </head>
8 <body>
9   <h1>Hello, World</h1>
10
11  <p>焼肉食べたい</p>
12 </body>
13 </html>
```

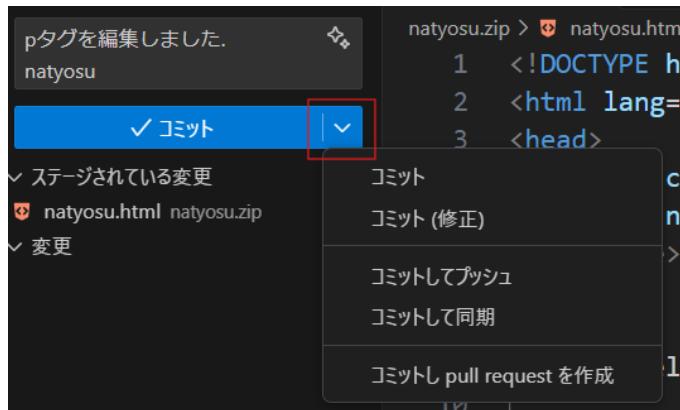
A red box highlights the 'メッセージ (Ctrl+Enter で "f...")' input field at the top of the editor. Another red box highlights the '✓コミット' button in the bottom right corner of the editor window. A green arrow points from the explanatory text below to the '✓コミット' button.

保存をしたら画像の様になつ
ているはずです。

コミットを作成すると
きは、セーブするファイルを選択(ステージする)
してコメントを書いて
セーブ(コミット)をしま
す。

セーブのコメントを書く, ex: pタグを編集しました.

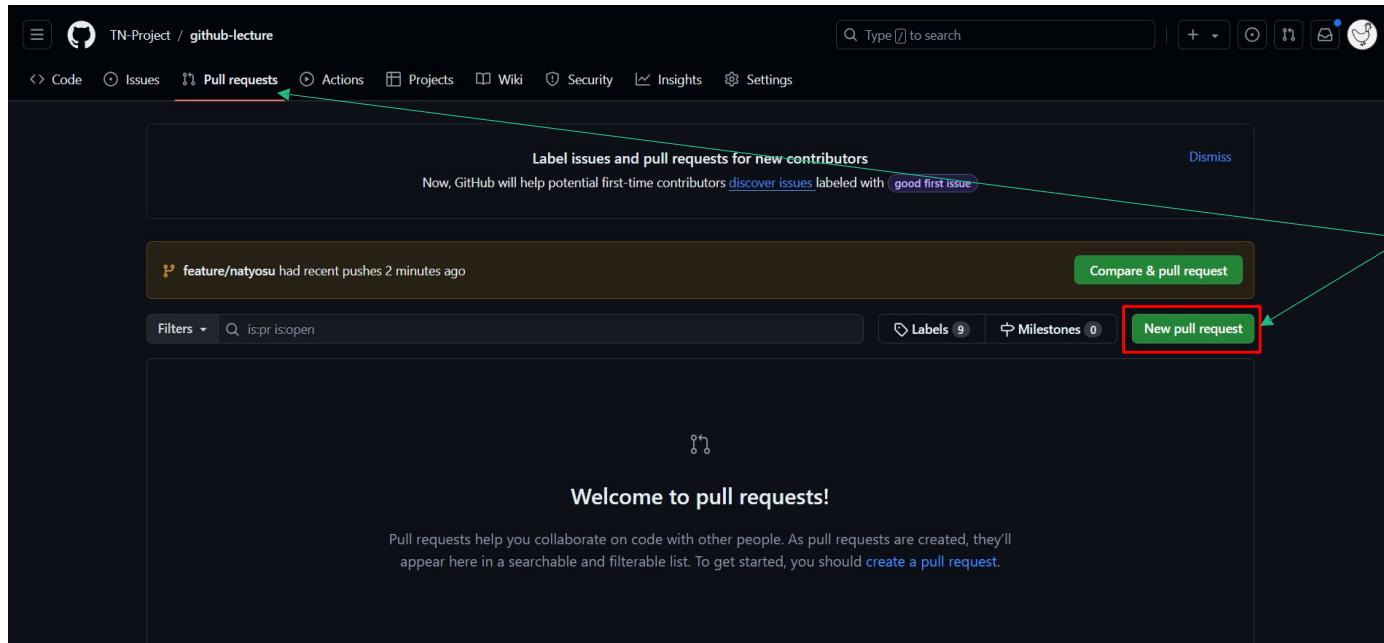
コミットを作成してみよう



逆さ山をクリックし，“コミットしてプッシュ”をクリックすることでコミットができます。

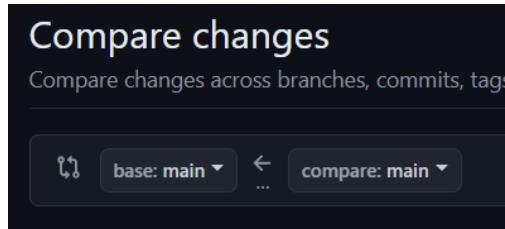
同様に2, 3個コミットを作成してみましょう

Pull requestを作成してみよう

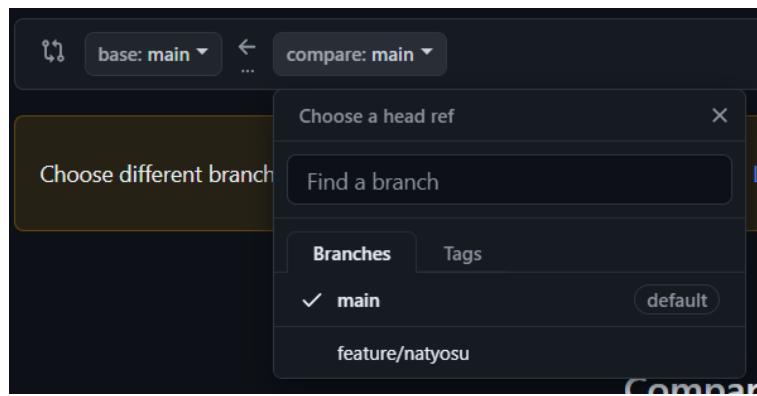


Githubのリポジトリページで
pull requestタブに移動して
New pull requestをクリック
してください。

Pull requestを作成してみよう

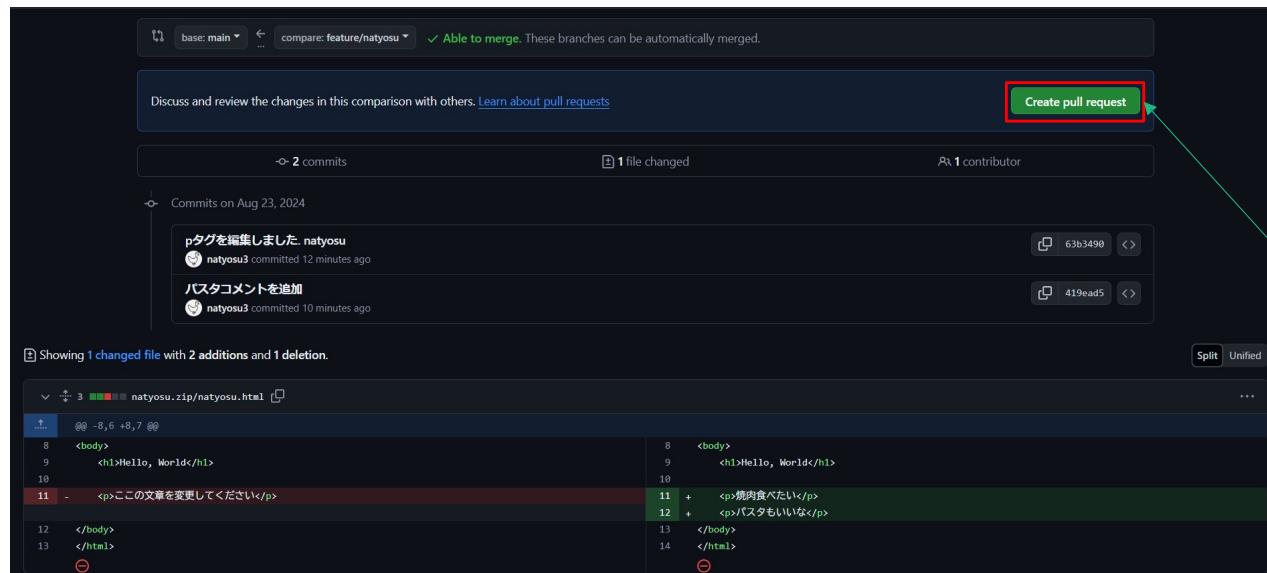


これはどのブランチをどこの
ブランチにmerge(結合)した
いのかを選択します。



左側が結合させる先, 右側が
結
合させるブランチ

Pull requestを作成してみよう



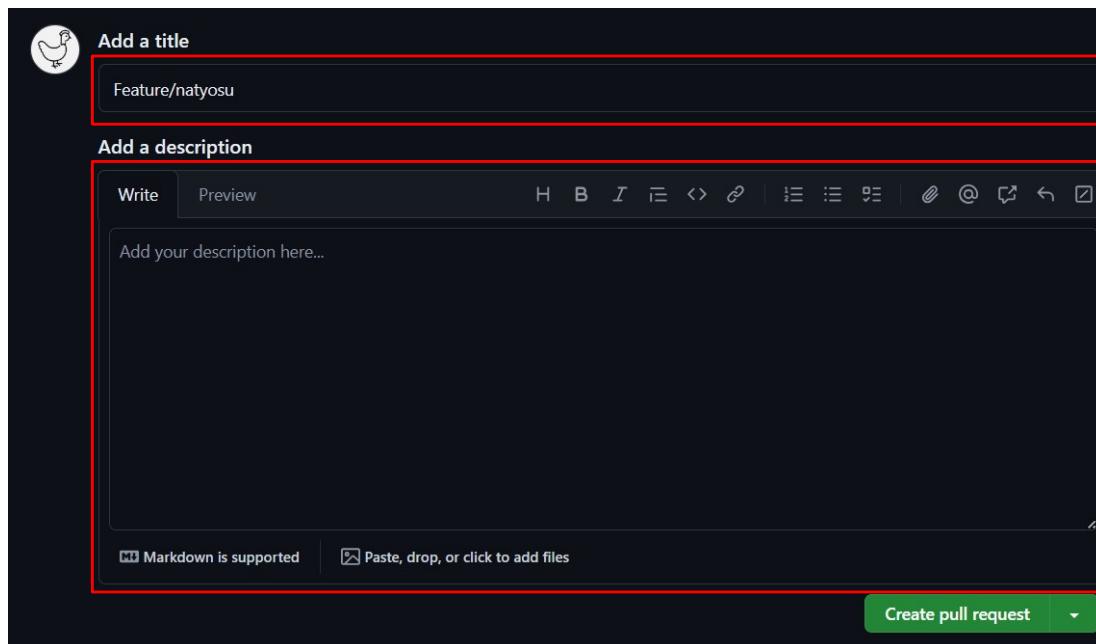
The screenshot shows a GitHub pull request comparison interface. At the top, it says "base: main" and "compare: feature/natyosu". It indicates "Able to merge. These branches can be automatically merged." Below this, there's a summary bar with "2 commits", "1 file changed", and "1 contributor". The commit history shows two commits from "natyosu3" on August 23, 2024. The first commit adds a tag and the second adds a comment. The "Create pull request" button is highlighted with a red box and a green arrow points to it from the text on the right.

選択すると、今までのコミット履歴と差分が表示されます。

問題無ければ、create pull requestをクリックします。

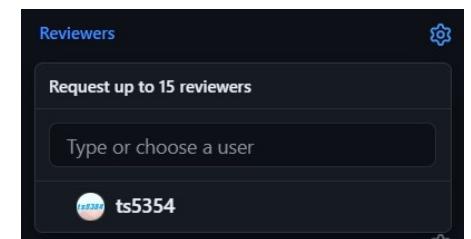
Pull requestを作成してみよう

Pull requestのタイトルを入力して、簡単なpull requestに関するコメントを書きます。



タイトル例: 機能Aを実装しました

Reviewersにレビューをして欲しい人を選択します。



最後にCreate pull requestを押しpull requestを作成できます。

レビュー&マージをする

natyosuブランチの編集が完了しました. #1

[Open](#) natyosu3 wants to merge 2 commits into `main` from `feature/natyosu`

Conversation 0 Commits 2 Checks 0 Files changed 1

 natyosu3 commented 4 minutes ago
aaa

 natyosu3 added 2 commits 28 minutes ago

- p タグを編集しました. natyosu
- パスタコメントを追加

Require approval from specific reviewers before merging
[Rulesets](#) ensure specific people approve pull requests before they're merged.

This branch has no conflicts with the base branch
Merging can be performed automatically.

Merge pull request

You can also open this in GitHub Desktop or view command line instructions.

Conversation 0 Commits 2 Checks 0 Files changed 1

Changes from all commits ▾ File filter ▾ Conversations ▾ Jump to ▾ Review ▾

natyosu.zip/natyosu.html

...	@@ -8,6 +8,7 @@	...	
8	<body>	8	<body>
9	<h1>Hello, World</h1>	9	<h1>Hello, World</h1>
10		10	
11	- <p>この文章を変更してください</p>	11	+ <p>焼肉食べたい</p>
		12	+ <p>パスタもいいな</p>
12	</body>	13	</body>
13	</html>	14	</html>

レビューが問題無ければ、
Merge pull requestをクリックして mergeを実行できます。