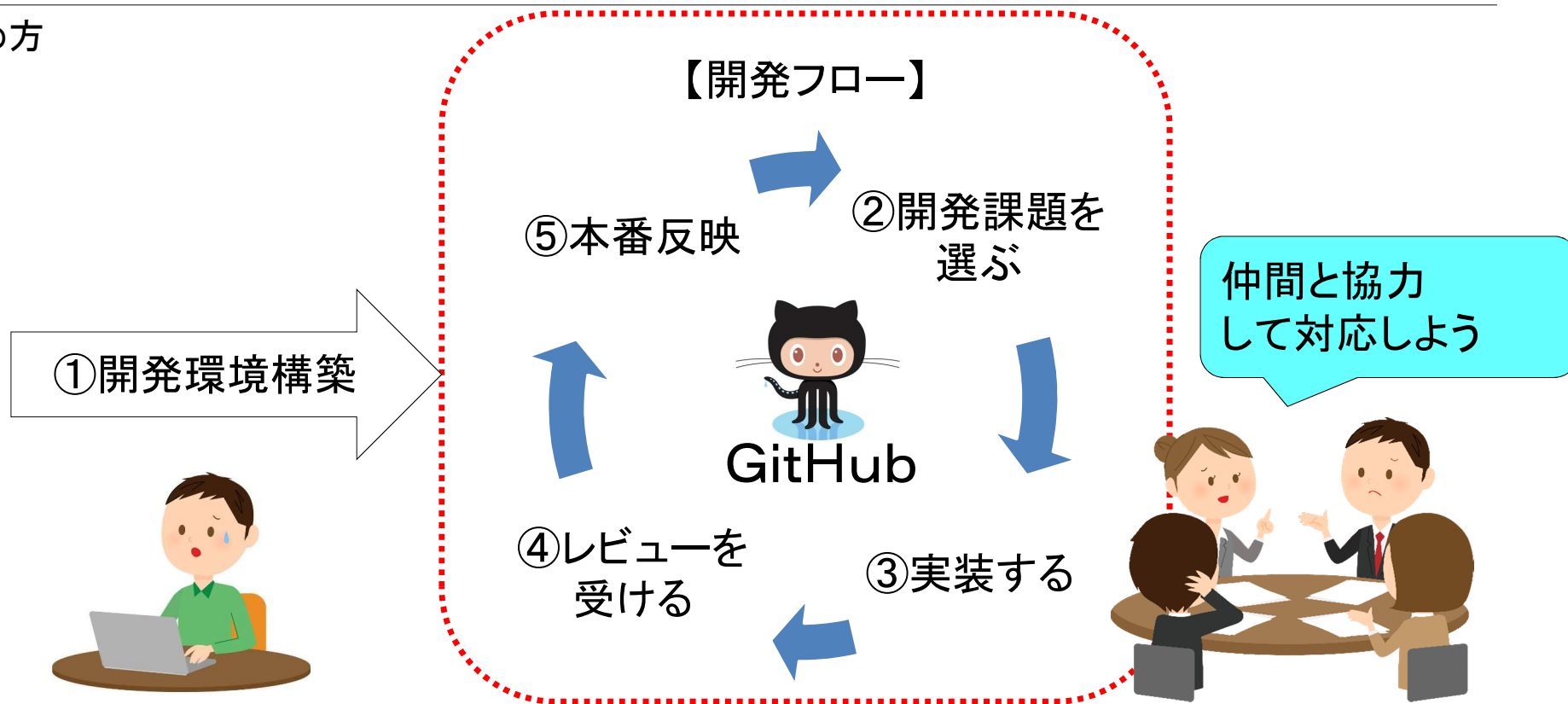


# 開発環境作った後の進め方

～Git超入門～

開発環境が準備できたら、開発フローに参加してください。

勉強会の進め方



# 今日のゴール(目指すところ)


---

開発環境を構築できた人が、Gitの仕組みを理解して、実装に着手できるようになる。



---

# 【理論編】 Git超入門



# GitHubはソフトウェア開発の標準プラットフォームです。

GitHubとは

## コードを共有するためのホスティングサービス



要するに、作ったプログラムを置いておく場所。

でもそれだけでなく...

チームが高品質のコードを生み出すための、**コラボレーション機能**がある。

- Pull Request (レビュー依頼)
- 特定ユーザへのコメントでコミュニケーション
- **ソーシャルコーディング**

⇒ 誰もが平等に参加できる。世界を広げてオープンに。

ユーザ数  
4000万以上



資料: GitHub実践入門、技術評論社、2014

# GitHubを活用することで開発現場の様々な課題が解決できます。

## 開発現場の課題

- ①現場によって開発の進め方が異なっているため、新規参入には多くの**学習コスト**が必要。  
⇒GitHubに慣れていれば、GitHubを使っている現場に新規参入しやすくなる。
- ②退職して居なくなった人や、書いた本人しか知らないコードが本番環境に残っている。  
⇒GitHubにレビュー結果が残っているので、経緯を追いやすい。
- ③書き間違いや思い違いなど、**簡単なコードのミス**により不具合が多発する。  
⇒GitHubでレビューを受けるので、単純ミスは減るはず。

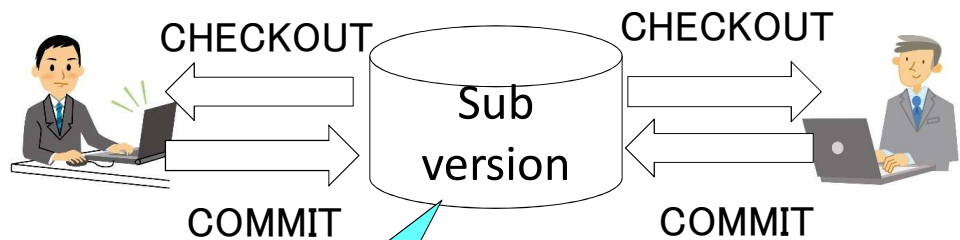


資料: GitHub実践入門、技術評論社、2014

# Gitを活用するにはルールを決める必要があります。

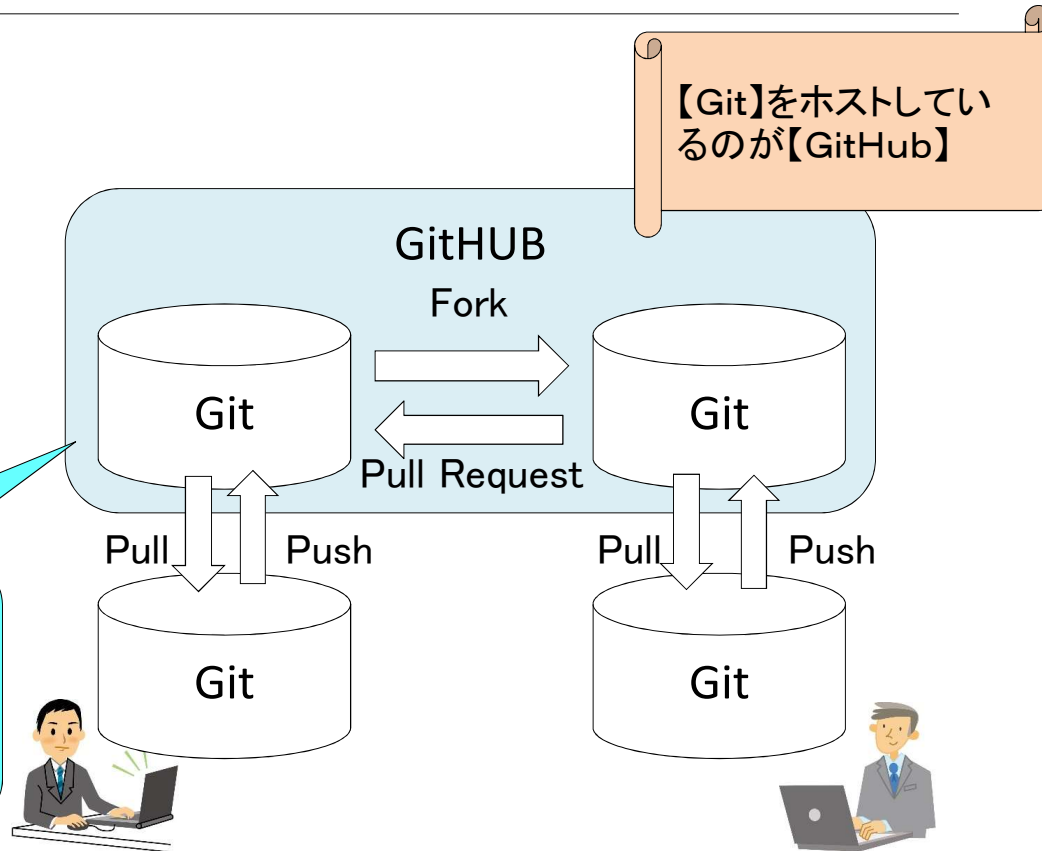
## Gitの長所と短所

### ①集中型



マスタが1つなので  
わかりやすいが、編集  
の順番に要注意

ソースが分散している  
ので編集しやすいが、  
どれがマスターかわ  
からなくなる

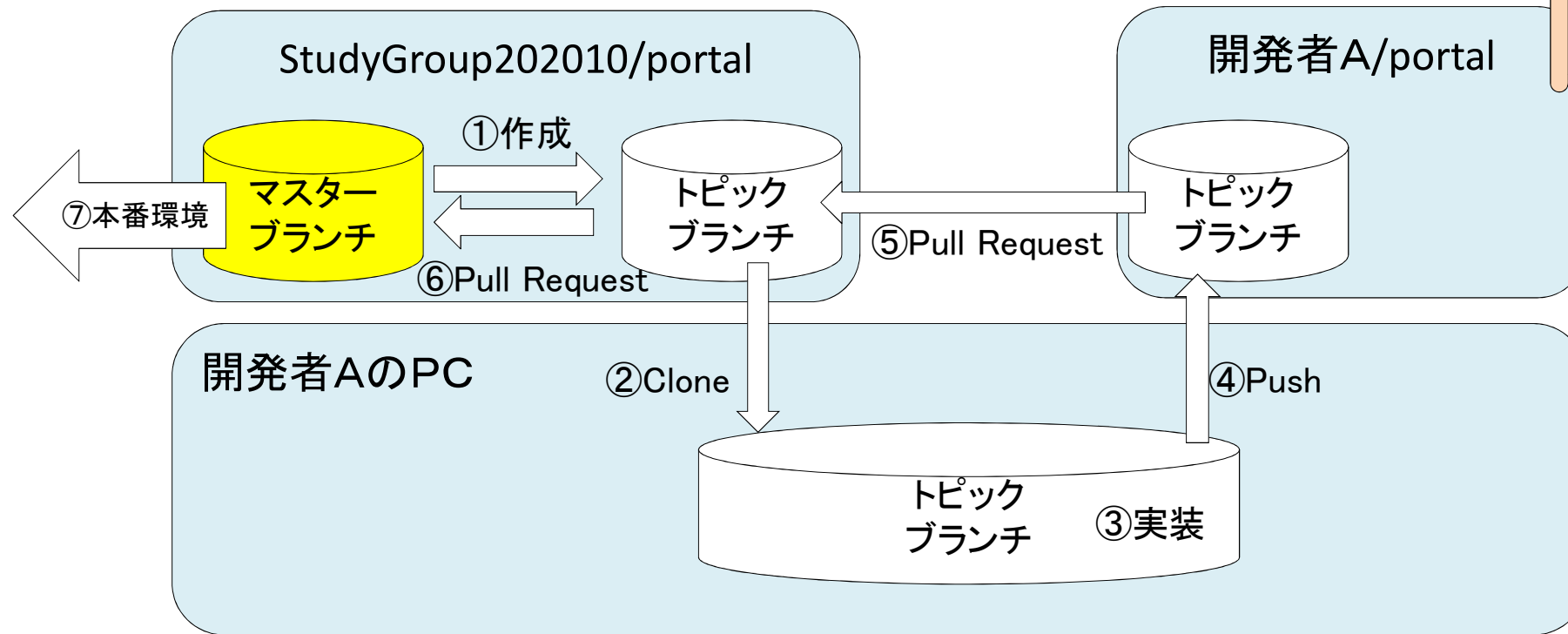


資料: GitHub実践入門、技術評論社、2014

勉強会ではGitHubが提案している**ブランチ運用ルール**に従って開発をします。

ブランチ運用ルール (GitHubFlow)

【ブランチ】  
ブランチとはマスターの  
コピーのこと



資料: GitHub実践入門、技術評論社、2014



# ブランチ運用ルールを理解して、開発を進めましょう。

## 開発フローの流れ

Issue対応の途中でギブアップしてもOKですので安心して下さい。

### ②開発課題を選ぶ

⇒GitHubリポジトリのIssueを選んでコメントをする。  
※最初は練習用のIssue【#52】を選んでください。

### ③実装する。

⇒トピックブランチを自PCにクローンして実装する。

### ④レビューを受ける

⇒実装したらブランチをCommit、PushしてPullRequestを投げます。


### ⑤本番反映

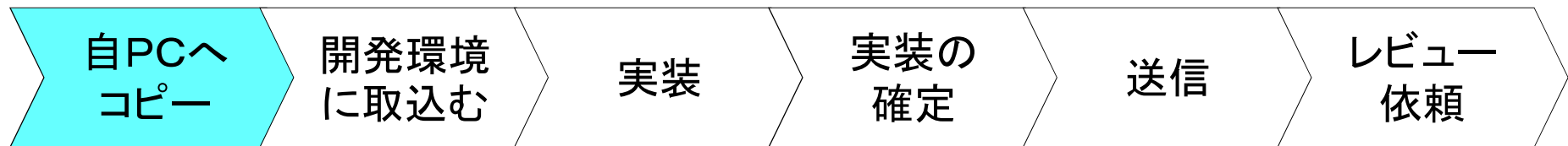
⇒ブランチがMergeされたら自動でデプロイされます。



---

# 【実践編】 実際の操作方法

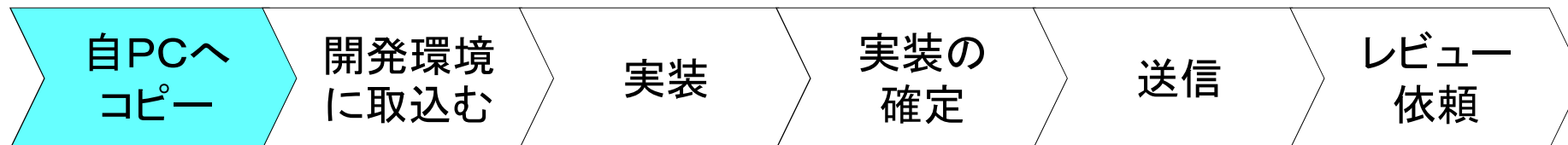




## 自PCへのトピックブランチクローン方法

1. デスクトップにフォルダを作り、その中でマウスを右クリック。  
「Gitクローン(複製)」を選択。





## 自PCへのトピックブランチクローン方法

StudyGroup202010/portalのサイトを指定する

202012\_#52 6 branches 0 tags

This branch is 13 commits behind main.

suzuki-takashi Merge pull request #121 from StudyGroup202010/202011\_#84

design Add files via upload

environment Merge branch 'main' into 202010\_1\_cor

src Merge pull request #121 from StudyGro

HELP.md Issue#24対応

LICENSE Initial co

README.md Merge br

Clone

HTTPS SSH GitHub CLI

<https://github.com/StudyGroup202010/portal>

Use Git or checkout with SVN using the web URL.

Open with GitHub Desktop

Download ZIP

Git クローン - TortoiseGit

すでに存在するリポジトリをクローン

URL: <https://github.com/StudyGroup202010/portal.git>

ディレクトリ C:\Users\suzuk\Desktop\新しいフォルダー\portal

☐ 深さ 1 ☐ 再帰的 ☐ Bareリポジトリにクローン ☐ チェックアウトしない

☒ ブランチ 202012\_#52 ☐ オリジナル名 ☐ LFS

☐ Putty 鍵のロード(K)

SVNのリポジトリから

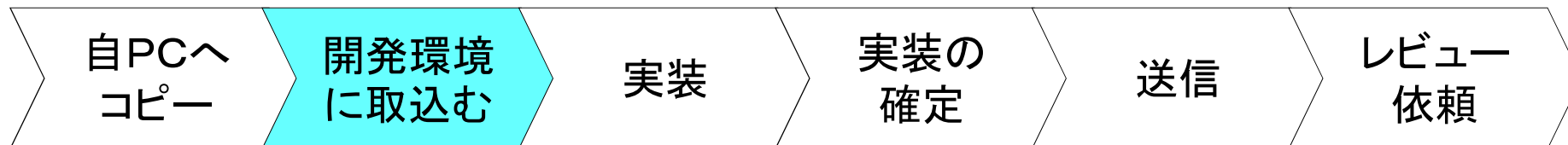
☐ SVN リポジトリから(S)

☐ Trunk: trunk ☐ タグ(G): tags ☐ ブランチ(H): branches

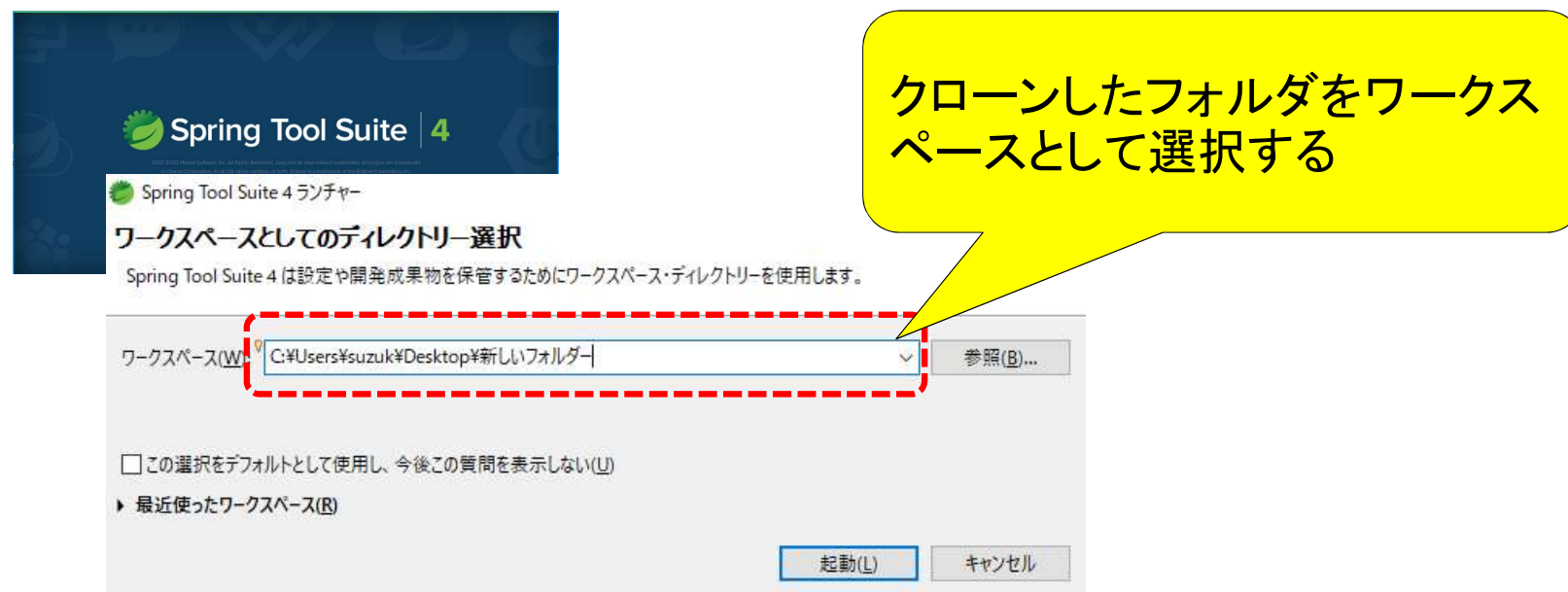
☐ From: 0 ☐ ユーザー名(N):

OK キャンセル ヘルプ

トピックブランチをクローンする  
例: 202012\_#52



## STS4のワークスペース選択方法



自PCへ  
コピー

開発環境  
に取込む

実装

実装の  
確定

送信

レビュー  
依頼

## STS4へのインポート方法

新しいフォルダー - Spring Tool Suite 4

ファイル(F) 編集(E) ソース(S) リファクタリング(T) ナビゲート(N) インポート

パッケージ・エクスプローラー

ワークスペースにプロジェクトがありません。  
プロジェクトを追加するには:

- Java プロジェクトの作成
- 新しい Spring スターター・プロジェクトを作成する
- Spring 入門コンテンツのインポート
- プロジェクトの作成...
- プロジェクトのインポート...**

インポート

既存の Maven プロジェクトをインポート

インポート・ウィザードの選択(S):

- 一般
- EJB
- Git
- Gradle
- Java EE
- Maven
  - Materialize Maven Binary Project
  - Maven リポジトリ・ヘインストールまたはアーティファクトをデプロイ
  - SCM から Maven プロジェクトを具体化
  - 既存 Maven プロジェクト
  - TextMate
  - Web
  - Web サービス
  - XMI

既存Mavenプロジェクト

Maven プロジェクトのインポート

Maven プロジェクトの選択

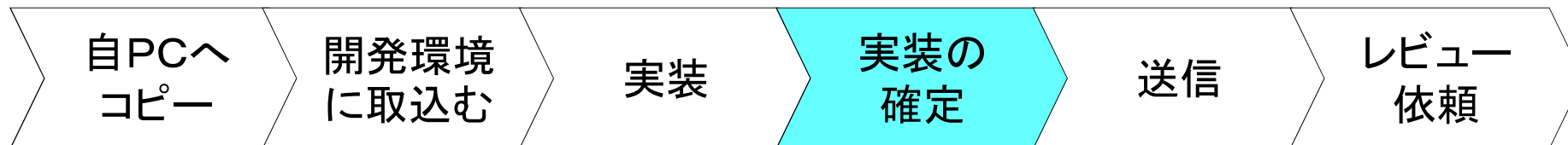
ルート・ディレクトリ(R): C:\Users\suzuk\Desktop\新しいフォルダー\portal

プロジェクト(P): /pom.xml com.portal:Portal-1.0.0.jar

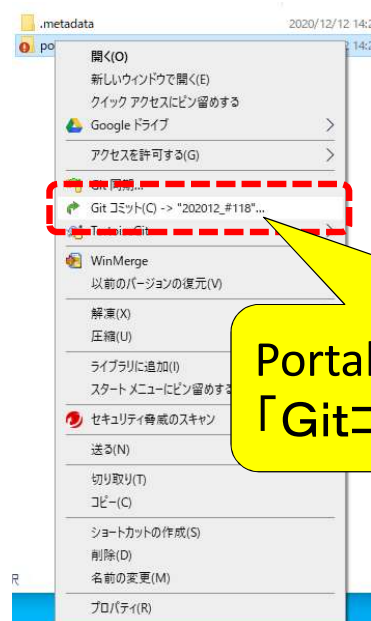
すべて選択(A) 選択をすべて解除(D)

クローンしたportalフォルダ

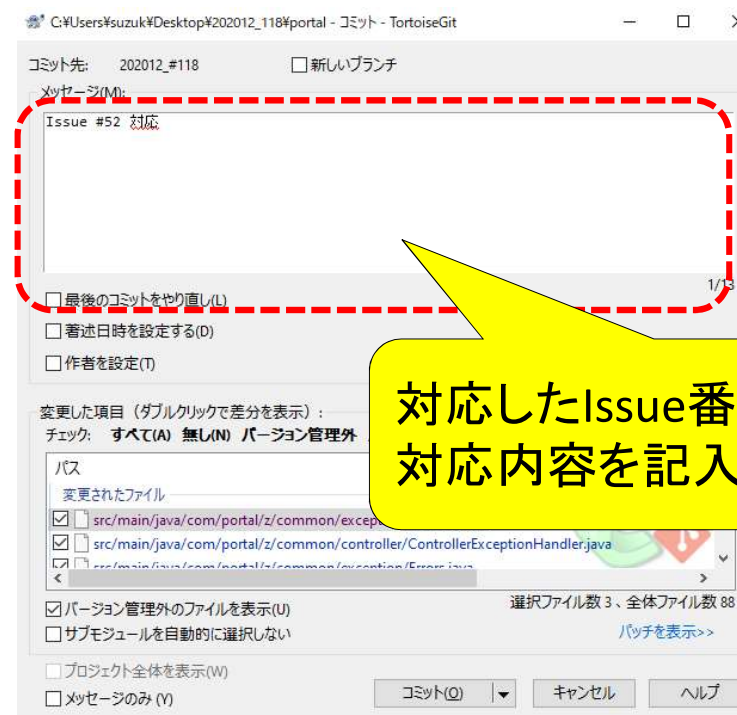
「環境構築手順」の9番の手順です。



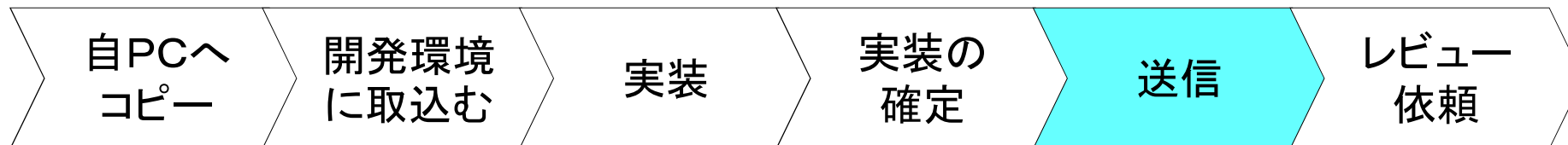
## Commit方法



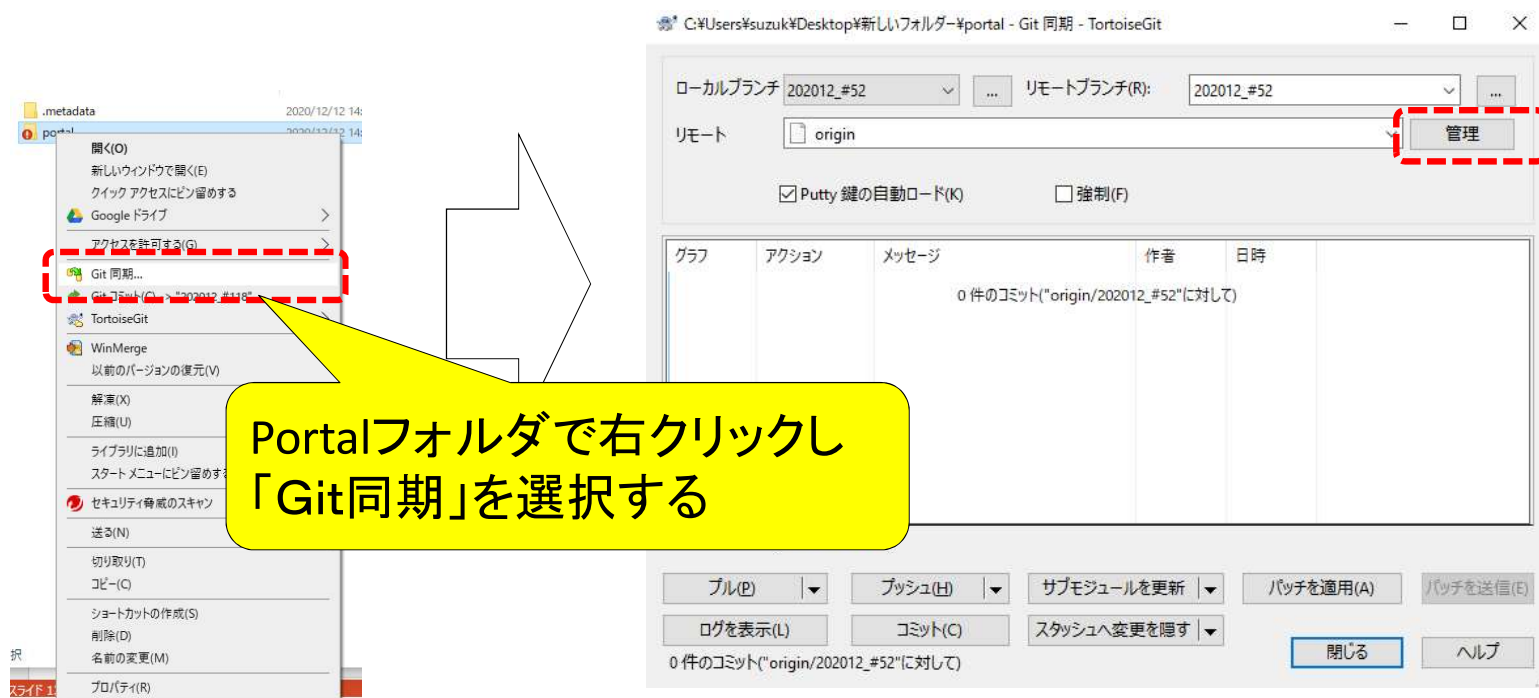
Portalフォルダで右クリックし  
「Gitコミット」を選択する



対応したIssue番号を記載し、  
対応内容を記入する



## Pushする方法





自PCへ  
コピー

開発環境  
に取込む

実装

実装の  
確定

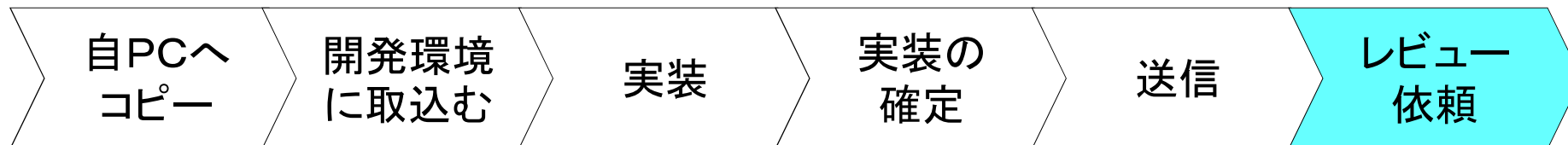
送信

レビュー  
依頼

## Pushする方法

自分のアカウント/portalのサイトのSSHのアドレスを指定する

環境構築手順1「ツール参考」で作った公開鍵のファイルを指定



## PullRequestする方法

自分のアカウント/portalに送信された

suzuki-takashi / portal  
forked from StudyGroup202010/portal

<> Code Pull requests Actions Projects Wiki Security Insights Settings

202101\_#52 had recent pushes less than a minute ago

Compare & pull request

クリックする

202101\_#52 11 branches 0 tags

Go to file

Add file

Code

This branch is even w

Pull request Compare

送信したブランチ

suzuki-takashi

StudyGroup202010/... 6dfbdb8 13 days ago 232 commits

design

last month

environment

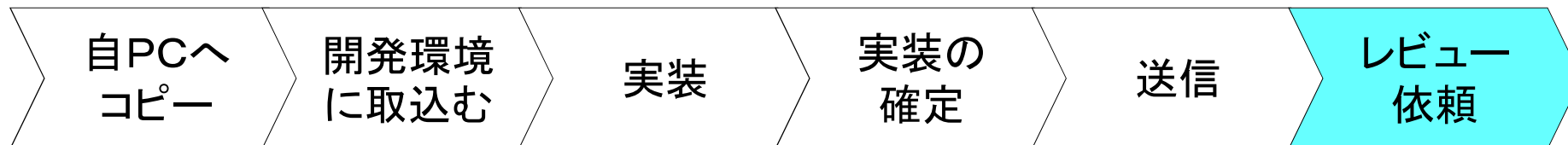
Merge branch 'main' into 202010\_1\_commit

3 months ago

src

Issue #96対応

13 days ago



## PullRequestする方法

The screenshot shows the GitHub interface for creating a pull request. Annotations include:

- PullRequest送信先** (Pull Request destination): Points to the 'base repository: StudyGroup202010/portal' dropdown.
- 自分のGitHubサイト** (My GitHub site): Points to the 'head repository: suzuki-takashi/portal' dropdown.
- クリックすると PullRequestが飛びます** (Clicking will send the Pull Request): Points to the 'Create pull request' button.

Other visible text on the page includes: 'Open a pull request', 'Create a new pull request by comparing', 'base repository: StudyGroup202010/portal', 'base: 202101\_#52', 'head repository: suzuki-takashi/portal', 'compare: 202101\_#52', 'Able to merge. These branches can be automatically merged.', 'Issue #52対応', 'Write', 'Preview', 'Leave a comment', 'New Video support! Upload MP4 and MOV file types. Attach files by dragging & dropping, selecting or pasting them.', and 'Allow edits by maintainers'.

# 今日のゴール(目指すところ)

---

開発環境を構築できた人が、Gitの仕組みを理解して、実装に着手できるようになる。



ご清聴ありがとうございます