

# ■ Git の運用手順

## ● 基本手順

- (1) 作業用ブランチを作成
  - (2) develop ブランチをフェッチ
  - (3) develop ブランチを作業ブランチにマージ
  - (4) 諸々の開発作業
  - (5) 作業したファイルをコミット
  - (6) 作業ブランチをサーバーにプッシュ
  - (7) GitLab の Web ページからマージリクエストを作成
- ※ 以後は (1) から繰り返す

## ● 各手順の詳細

### 【 0. develop ブランチを最新の状態に更新 】

※ 以下の手順は初回のみ

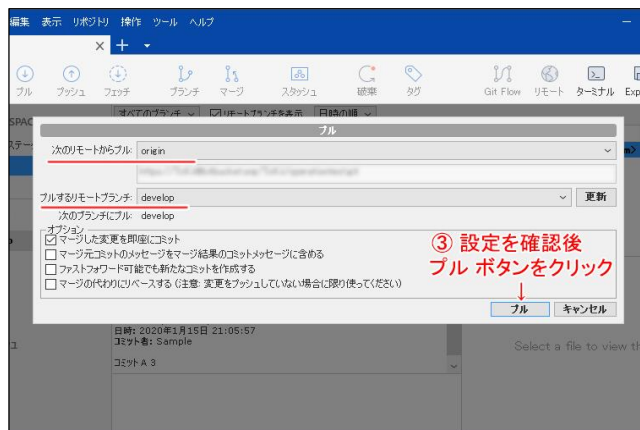
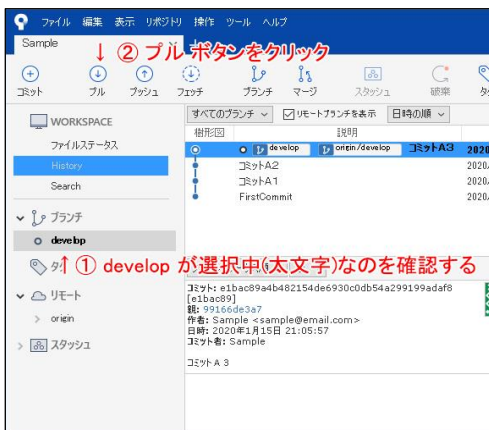
SourceTree上で develop ブランチが選択されていることを確認する (太文字になっていないときはダブルクリックして切替え)

ツールバーから『プル』ボタンをクリック

設定ダイアログが以下のようにになっているのを確認する (デフォルトの設定)

ダイアログ下部の『プル』ボタンをクリック

次のリモートからプル	origin
プルするリモートブランチ	develop



## 【1. 作業ブランチを作成】

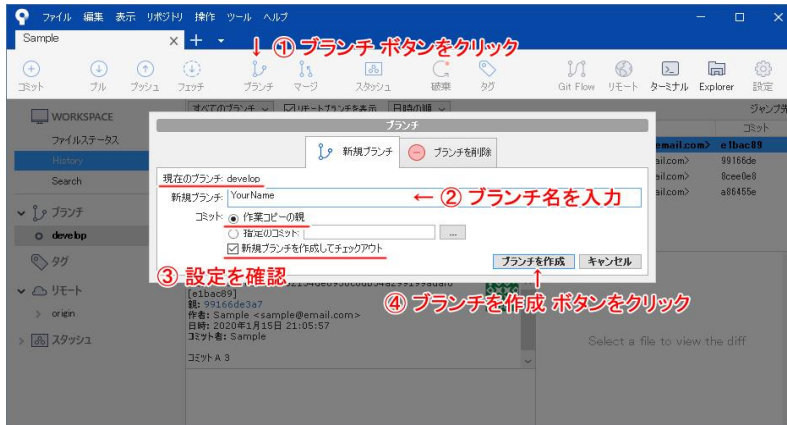
ツールバーから『ブランチ』ボタンをクリック

『新規ブランチ』の項目にブランチ名を入力（『ブランチの命名規則』を参照）

ダイアログ下部の『ブランチを作成』をクリック

指定した名前のブランチが作成され、選択状態（太文字）になっていればOK

現在のブランチ	develop (初回の場合) 前回の作業ブランチ名 (2回目以降)
新規ブランチ	「自分の名前 / 作業名」の形で入力
コミット	作業コピーの親 にチェック
新規ブランチを作成してチェックアウト	オン



### <ブランチの命名規則>

日本語は使用せず半角文字のみ使用する

「自分の名前 / 作業名」の形にする

例) マップ01を作成するなら「Nuruki/Map01」とする

## 【2. develop ブランチをフェッチ】

ツールバーの『フェッチ』ボタンをクリック

develop ブランチの横に『2 ↓』のような数字と下矢印が表示されていたら以下の手順を行う

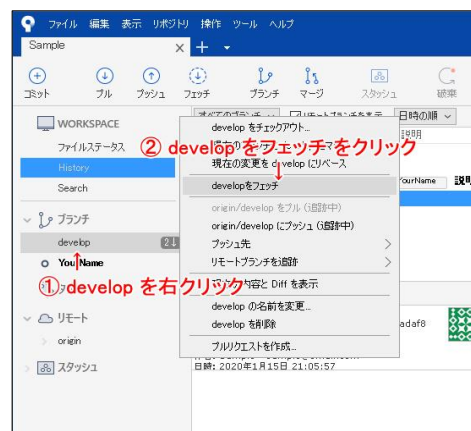
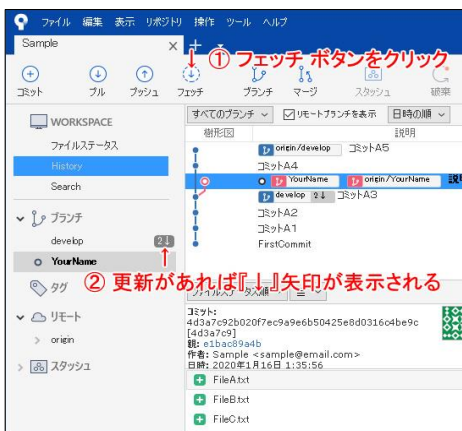
(何も表示されていないときは手順4まで飛ばしてOK)

develop ブランチの上で右クリックメニューを開き『develop をフェッチ』をクリックする

正しく成功すると develop の横にある数字と下矢印が消える

※ develop ブランチの横に上矢印と数字が表示されたときは早急に塗木にご連絡ください

誤って develop ブランチを不正に書き換えています

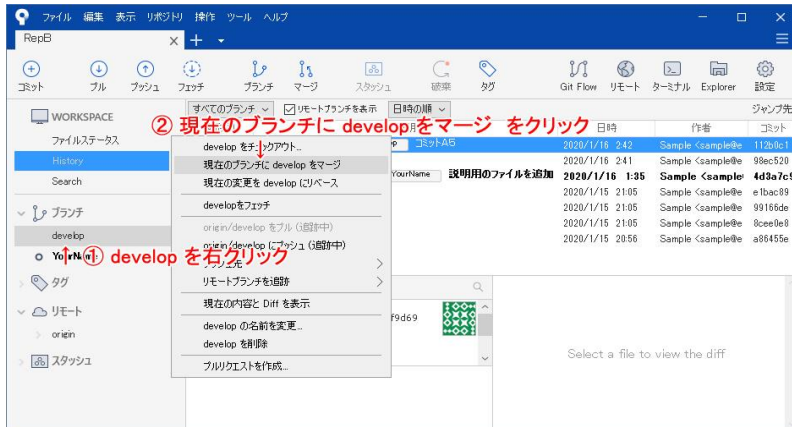


### 【 3. develop ブランチを作業ブランチにマージ 】

develop ブランチの上で右クリックメニューを開き『現在のブランチに develop をマージ』をクリックする

確認ダイアログが出たら『OK』をクリック

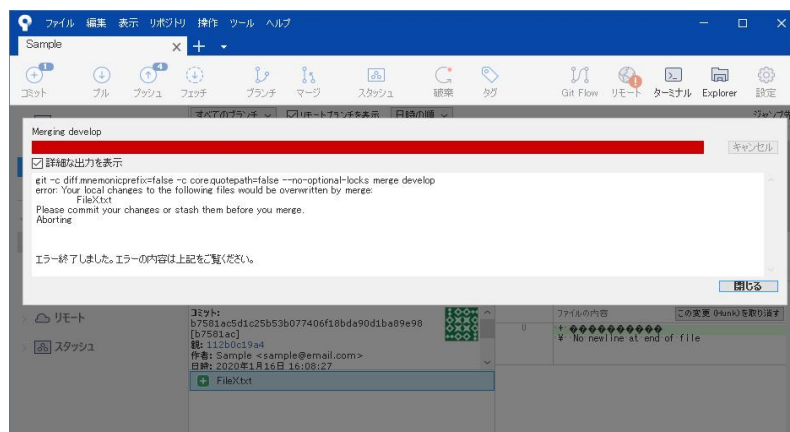
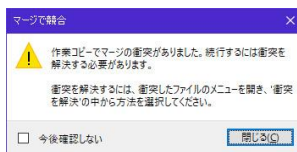
マージ後はUnity上でエラーが発生していないかなど動作確認を行う



#### <マージエラーについて>

他のメンバーと同時にファイルを編集した際にはマージエラーが発生する

対処方法が判らない場合は塗木までご連絡を



### 【 4. 諸々の開発作業 】

自身の担当作業を行う

履歴管理と確認の効率化のため一言で何をしたら説明できる程度の単位で作業を行うのが理想

## 【 5. 作業したファイルをコミット 】

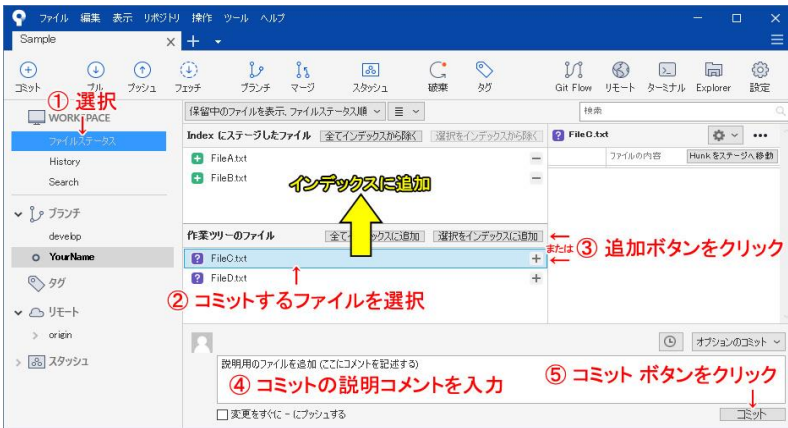
『ファイルステータス』を選択し編集ファイルの一覧を表示

『作業ツリーのファイル』からコミットするファイルを選択し『+』又は『選択をインデックスに追加』ボタンでインデックスに追加  
上の操作を繰り返してコミットする全てのファイルをインデックスに追加する

コメントウィンドウに以下のルールでコメントを記載する

- ・コメントは日本語で書く（但し、環境依存文字は避ける）
- ・改行を使用せず1行にまとめる
- ・他の人でもコメントを見るだけで編集内容が分かるようにするのが望ましい  
（1行で説明できるくらいの作業単位でコミットを作るのが良い）

『コミット』ボタンを押してコミットを作成する



～注意～ ファイルをUnityに追加した際は必ず meta ファイルもコミットする（ファイル名.拡張子.meta となっているファイル）  
自分が編集していないファイルは原則コミットしない

- └ Unity が自動で生成、書換えを行う事もあるため判断がつかない場合は担当者に相談
- エラーが残っている、ファイルが足りていないなど正しく動作しない状態ではコミットしない
- コミットするファイルの依存関係も考慮する
- └ 依存するファイルをコミットし忘れるなどすると他の人の環境でのみエラーが発生して恨まれる

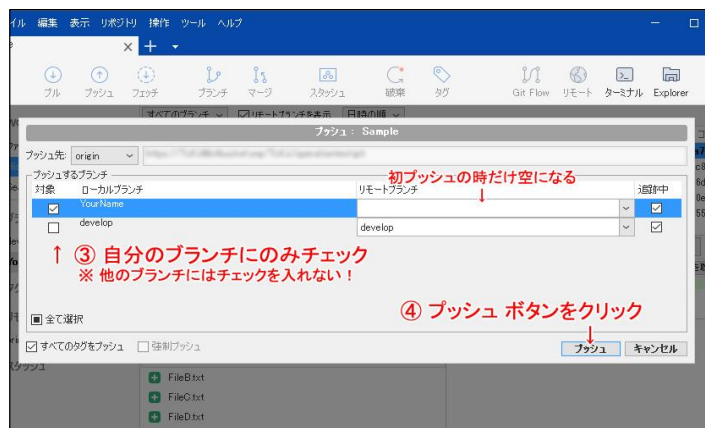
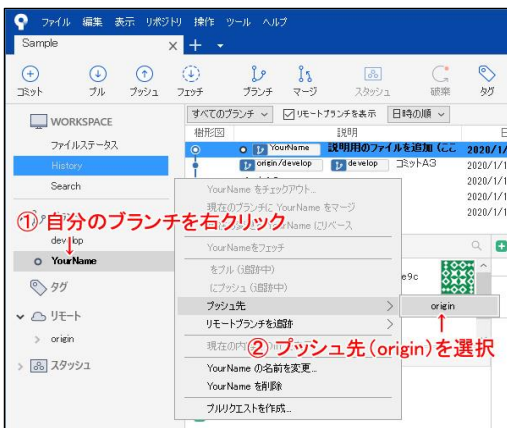
## 【 6. 作業ブランチをサーバーにプッシュ 】

作業ブランチで右クリックメニューを開き プッシュ先 → origin を選択する

開いたダイアログにて、プッシュするブランチの対象が自分のブランチにのみチェックが入っていることを確認する

（他のブランチをプッシュ対象に含めないように注意）

ダイアログ右下の『プッシュ』ボタンをクリック



## 【7. GitLab の Web ページからマージリクエストを作成】

GitLab にアクセスしてブランチページを開く

**サムライバンダリズム** [http://pti-software.synology.me:6090/samurai\\_vandalism/samurai\\_vandalism/branches](http://pti-software.synology.me:6090/samurai_vandalism/samurai_vandalism/branches)

**軌轢のイデオログ** <http://pti-software.synology.me:6090/ideologue/ideologue/branches>

手順6でプッシュしたブランチの『Merge request』ボタンをクリック

マージリクエストのページから以下の項目を設定

『Submit』ボタンを押してマージリクエストを発行

<b>Title</b>	マージリクエストのタイトル。コミットに設定したコメントでOK
<b>Description</b>	補足や連絡事項などがあれば記載する。なければ空白にする
<b>Source branch</b>	手順6でプッシュしたブランチ名になっているかを確認
<b>Target branch</b>	develop ブランチになっているのを確認
<b>Delete source branch ...</b>	チェックを入れる (マージ完了後にブランチを削除する)