

## GitHub Flow とは

Git のブランチ機能を利用する開発戦略の一つです。

Git-flow を採択して、もっと効率的に開発するソースコードを管理することができます。そして、共通のルールがあるので、コラボレーションをしながら発生する問題を最小限にすることができます。

最近、多くのプロジェクトで Git-flow 戦略が採択されています。

### ●リポジトリ

ファイルやディレクトリの状態を記録する場所で、保存された状態は、内容の変更履歴として格納されています。

変更履歴を管理したいファイルやディレクトリをリポジトリの管理下に置くことで、そのディレクトリ内のファイルやディレクトリの変更履歴を記録することができます。

Git のリポジトリには主に 2 種類あります。

#### 【リモートリポジトリ】

専用のサーバに配置して複数人で共有するためのリポジトリのことです。

リモートリポジトリからアップロードされた他の人のファイルを閲覧できる。

#### 【ローカルリポジトリ】

ユーザー一人ひとりが利用するために、自分の手元のマシン上に管理するリポジトリのことです。ローカルリポジトリで編集した内容をリモートリポジトリにアップロードできる。

### ●ブランチ

履歴の流れを分岐して記録していくためのものです。

分岐したブランチは他のブランチの影響を受けないため、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ます。

また、分岐したブランチは他のブランチと合流（マージ）することで、一つのブランチにまとめ直すことができます。

チームのメンバーは、他のメンバーの作業の影響を受けないように、メインのブランチから自分の作業専用のブランチを作成します。

そして作業の終わったメンバーは、メインのブランチに自分のブランチの変更を取り込んでいきます。

このようにすることで、他のメンバーの作業による影響を受けることなく、自分の作業に

取り組むことができます。

また、作業単位で履歴を残すことで、問題が発生した場合に原因となる変更箇所の対策や調査を行うことが容易になります。

大規模プロジェクトの場合は、チームメイトに確認してもらうため Pull Request（プルリクエスト）して確認してもらい、OK であれば main ブランチへマージすることができる。

## ● main ブランチ

リポジトリに最初のコミットを行うと、Git は main という名前のブランチを作成します。そのため、以後のコミットはブランチを切り替えるまで main ブランチに追加されていきます。