

初学者向け GitHub flow 学習支援

玉田 春昭¹

概要: 今日の開発では、git などを用いたバージョン管理は必須であり、それに加えて、Git flow, GitHub flow などの開発フローを利用した開発も一般的になりつつある。しかし初学者にとって、開発フローの理解と実践は一般的に難しい。プログラムの内容以外に考えなければならない事柄が導入される上に、その必要性の実感が持てないためである。そこで本稿では、開発フローの 1 つである GitHub flow の定着を目的にした支援システムの構築を目指す。

Learning support system of GitHub flow for novice developers

1. はじめに

昨今、GitHub などのソーシャルコーディングプラットフォーム (SCP; Social Coding Platform) を用いた開発が一般的になりつつある。そのような状況において、SCP を用いた開発フローの教育の確立が必要である。SCP を用いた開発フローの実施は、一般に初学者には難しい。初学者がこれまでに経験したプログラム開発とは根本的に異なっているためである。例えば、従来の開発での関心ごとの大半はプログラムそのものであった。一方、SCP を用いた開発フローでは、バージョン管理や、Pull request, Issue など、プログラムの管理に関することも考えなければならない。加えて、プログラムの規模も演習レベルのものに比べて格段に大きく、多人数開発を前提としている点なども大きく異なっている。

しかしながら、SCP を用いた開発フローへの理解がなければ、実際の開発に携わることが困難である。初学者が実際の実験に携わる際には、一通りの学習後に、指導者が初学者に寄り添い、適宜アドバイスを与えるような開発形態が考えられる。このような形態で開発を進めた場合、指導者の開発時間が大きく削られることになり、チーム全体の開発能力の低下に繋がる恐れがある。そこで本稿では、初学者を対象に開発フローで行うべき次の行動推薦を行い、外発的動機付けを行うためのポイント、トロフィー制度、さらに、自身の行動がどの程度開発フローに沿っているかを確認できる手法を組み合わせる。これにより、初学者が

自発的かつ自律的に開発フローを学びながら実際の実験に携わることが期待できる。加えて、初学者が指導者に助けをを求める頻度を削減しつつも、指導者が後から初学者にアドバイスするための情報を提供し、初学者の教育にも貢献できると期待できる。

2. 準備

2.1 本研究で対象とする初学者の技術レベル

本稿の想定する初学者は、基礎レベルのプログラムは問題なく作成できるものとする。加えて、バージョン管理の存在を知っているものとする。言い換えれば、バージョン管理を実践できていない、もしくは、実践した経験はあるものの定着には至らなかった者を対象とする。

2.2 開発フロー

世の中の開発で利用されている開発フローには、Git flow^{*1}や GitLab flow^{*2}などがある。その中でもシンプルな GitHub flow^{*3} (以降 GHF) を対象とする。ただし、利用する基盤技術 (バージョン管理ツールや SCP など) は同じであるため、他の開発フローへの応用も可能である。

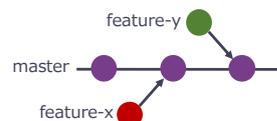


図 1 GitHub flow の模式図

¹ 京都産業大学
Kyoto Sangyo University.

^{*1} <https://nvie.com/posts/a-successful-git-branching-model/>

^{*2} <https://about.gitlab.com/2014/09/29/gitlab-flow/>

^{*3} <http://scottchacon.com/2011/08/31/github-flow.html>

本稿で対象とする GHF の手順を図 1 に示す。図 1 では、feature-x や feature-y ブランチを作成し、そのブランチで各機能 (x や y) を実装している。そして、機能が作成できれば、GitHub に push し、Pull Request を作成する。Pull Request は、レビューが行われたのちに master にマージされる。このように、GHF では機能の追加はブランチで行い、マージは必ず Pull Request 経由で行う。

3. 提案手法

本稿では次の 3 つの観点で GHF を支援する。

- モチベーション支援
- インプロセス支援
- 振り返り支援

3.1 モチベーション支援

初学者に外発的動機付けを与えるため、ポイント・トロフィー制度を設ける [1]。コミットや push など GHF のイベントの実施回数に応じてポイント、トロフィー制度を設ける。ポイントとは、行動に応じて与えられる点数のことを指し、この点数を貯めることで、何らかの報酬を得られるものとする。トロフィーとは一定の基準を満たすと得られる称号であり、PlayStation におけるトロフィー*4、Game Center におけるアチーブメント*5に相当するものである。

3.2 インプロセス支援

初学者が GHF を行うとき、次に何をすべきかがわからなくなることが考えられる。そこで、熟練者の行動を予め収集しておき、その行動履歴を元に初学者に次に何をすれば良いかを推薦することを考える [2]。例えば、コンパイル、テストの成功後のコミットの実施や、コミット後の push の実施などである。推薦のトリガには、git の hook 機構や GitHub での WebHook、また、シェルの hook 機構を用いることで可能であろう*6。

なお推薦には、ブランチの作成と移動、コミット、Pull Request の作成などの行動の推薦と、コミットや Pull Request のコメント内容の推薦が考えられる。コメント内容の推薦では、コミットメッセージの自動生成 [3] や Pull Request で書くべき内容の示唆*7などを積極的に導入する。

3.3 振り返り支援

インプロセス支援により、初学者が自律的に GHF を用

いた開発に参加できると期待できる。しかし、初学者であるため、GHF に沿わない行動を行う場合も考えられる。GHF の定着のためには機械的なフィードバックだけでなく、学習の進捗に合わせた適切なフィードバックが必要である。そのため指導者を交えて、学習者が行なった GHF の内容を振り返る必要がある。その振り返りにおけるフィードバックの議論のきっかけとして、開発フローの遵守度 [4] を示すことを考える。

4. まとめと議論したい内容

4.1 議論したい内容

- モチベーション支援、インプロセス支援、振り返り支援の 3 つの観点で十分であるか。
- モチベーション支援について
 - ポイント制で貯めたポイントはどのような用途に利用できるか。
- インプロセス支援について
 - 初学者の学習度合いに合わせた支援は、どのようなものが考えられるか。
- 振り返り支援について
 - 他に振り返りに役立つ指標はあるか。

4.2 まとめ

本稿では、GHF の実践を支援するシステムを通じて、初学者の GHF の定着を狙う。そのためのシステムの要件として、モチベーション支援、インプロセス支援、振り返り支援の 3 つの観点での支援内容を述べた。この後は実際にシステムを構築し、初学者に利用してもらうことで、提案手法の効果を確認する。

謝辞 本研究の一部は JSPS 科研費 17K00196, 17K00500, 17H00731 の助成を受けました。

参考文献

- [1] 西尾泰介, 柳川龍太郎, 玉田春昭. 初学者を対象とした github flow 支援ポット. ソフトウェアシンポジウム 2018 ワークショップ討論テーマ (4) ソーシャルコーディングのための実証的ソフトウェア工学, September 2018.
- [2] Akira Inokuchi, Haruaki Tamada, Hideaki Hata, and Masateru Tsunoda. Toward obliging bots for supporting next actions. In *Proc. of the 4th International Conference on Applied Computing & Information Technology (ACIT 2016)*, December 2016.
- [3] Luis Fernando Cortés-Coy, Mario Linares-Vásquez, Jairo Aponte, and Denys Poshyvanyk. On automatically generating commit messages via summarization of source code changes. In *Proc. of the 14th International Working Conference on Source Code Analysis and Manipulation*, September 2014.
- [4] 井上拓海, 小島遥一郎, 藤原賢二, 井垣宏. 版管理システム利用時のソフトウェア開発フロー遵守状況可視化手法の検討. 信学技法, No. SS2017-55, January 2018.

*4 <https://manuals.playstation.net/document/jp/ps3/current/trophy/trophy.html>

*5 https://developer.apple.com/library/archive/documentation/NetworkingInternet/Conceptual/GameKit_Guide/Achievements/Achievements.html

*6 <https://note.hibariya.org/articles/20170219/shell-postexec.html>

*7 <https://github.blog/2015-01-21-how-to-write-the-perfect-pull-request/>