

開発学習を作業手順の記録操作によって支援するシステムの提案

木越 湧太* 加藤 淳† 高橋 治輝* 松村耕平*

概要. 近年, AI を活用した開発支援技術の進展により, ソースコードの自動生成や手順提案が可能となった一方で, 開発者によるソースコードの理解や学習の重要性は依然として高い. 特に, 新しい技術の習得や開発過程で得られた知見を記録として残す行為は, 再現性の確保やチーム内での共有, 技術習得に有用である. そこで本研究では, Visual Studio Code の拡張機能として開発手順を半自動的に捕捉し, 効率的な記録を支援するだけでなく, 主観的な感情も併せて記録することで記憶への定着を促進するシステムを提案する.

1 はじめに

システム開発において環境構築やコーディングは必要不可欠な過程であり, 多数のコマンド入力やソースコードの理解が必須となる. 近年では AI によるソースコード生成や開発手順の提案が多くの開発者を助けているが, その精度は必ずしも信頼できるとは限らない. AI が生成したソースコードを評価してから提案を採用する開発者も少なくないことから [2], AI が生成した開発手順やソースコードの妥当性評価という観点で開発技術の学習は今まで同様必要とされるであろう.

開発技術学習において, サンプルプロジェクトを手元で実際に開いたり, ポートフォリオ Web サイト制作のように自身の内容に即して変更してみたりする実践的な学習は, 学習内容の定着や応用という面で非常に有効である. 加藤らは, 学習過程で編集したコードを使い捨てずに保存し, Web アプリケーションとしてデプロイまで可能なチュートリアルの開発用フレームワーク DeployGround を提案している [1]. LaToza らは, 開発者がソースコードを理解する過程で学習したことを記録する手法の有効性を議論しており [3], 学習過程で行った操作を記録として残すことで学習効果の向上が見込める.

本研究では, 開発者による実践的学習において, 学習内容を感情と共に記録可能なシステムを提案する.

2 提案システム

本研究では, Visual Studio Code (VSCode) の拡張機能として開発手順の記録を支援するシステムを提案する. ユーザは VSCode 拡張機能から開発手順の概要を登録し, Web ブラウザからその内容を閲覧する. 図 1 に提案システムの使用例を示している.

2.1 記録内容の構造化

提案システムでは「ノート」という単位で記録内容を管理し, Web 上のデータベース (Firestore) に保存する. ノートの中には, 開発の目的達成のための作業単位である「タスク」を複数作成することができ, タスクは開発操作の最小単位である「手順」から構成される. 手順はソースコード編集, ファイル操作, ターミナルコマンド実行の 3 種類に分類されており, Web アプリケーションにおいても上記のデータモデルや分類を元に構造化されたドキュメントとして表示される.

図 1 右図中の「Java 入門」はノート名を表しており, その中に「Hello World を表示する」というタスクが追加されていることが確認できる. さらにその中には「ファイル『sample.java』の作成」や「コンソール表示」という手順が保存されており, 作業の経過が構造化されていることがわかる.

2.2 システム要件

提案システムにおける VSCode 拡張機能では, ソースコード編集, ファイル操作, ターミナルコマンド実行をエディタ上で半自動的に捕捉し, 捕捉した内容をユーザに提示する. ユーザは提示された内容を確認し, その手順に関する説明を入力して登録ボタンを押下することで, 記録内容がデータベースに格納される. 各手順種別における捕捉方法や提示内容に関しては後述する.

また手順登録の際に, 図 2 手順実行の主観的な感情を記録させることができる. 記録された感情は Web アプリケーションにおいて手順ごとにタグとして表示される.

ソースコード編集記録

VSCode のエディタ上にマウスフォーカスが当たった際に対象のエディタの変更監視を開始し, 監視停止というボタン押下をユーザから受け取った際に, エディタ上で変更された内容を差分表示と要約表示の 2 種類をユーザに提示する. 閲覧用 Web アプリ



図 1. 提案システムの使用例. 左図は VSCode 拡張機能における記録操作の動作例. 右図は閲覧用 Web アプリケーションの表示例.

ケーションにおいても差分表示と要約表示を採用しており、閲覧したユーザがファイルのどの部分にコードを記述すればよいか明確になるように実装している。

図 1 左では、ソースコード編集記録の様子を表しており、VSCode 上に差分表示と全体表示がユーザに提示されていることが確認できる。また図 1 右下部において、左図で記録した内容を差分表示で閲覧できることがわかる。

ファイル操作記録

VSCode 上で行われたファイル・ディレクトリの新規作成、削除、名称変更を検知して、ファイル変更による差分を示したファイル構造木をユーザに提示する。同時にファイル変更の内容を手順名の入力欄に自動挿入し、ユーザはファイル変更の目的の入力を任意で行う。閲覧用 Web アプリケーションにおけるファイル操作記録内容の表示では、差分を示したファイル構造木を表示しどの階層でどのファイルを操作したのが明確になるように実装している。

図 1 の右中央において、ファイルの新規作成を差分を明記したファイル構造木で閲覧できることがわかる。

ターミナルコマンド実行記録

VSCode のステータスバーに「コマンドを記録する」というボタンを配置しており、押下することで VSCode のターミナルで実行された直前のコマンドとその実行結果が入力欄に自動挿入される。閲覧用 Web アプリケーションにおけるコマンド実行記録の表示では、実行したコマンドと実行結果が表示され、コマンドの種類や応答の種類が明確になるように実装している。

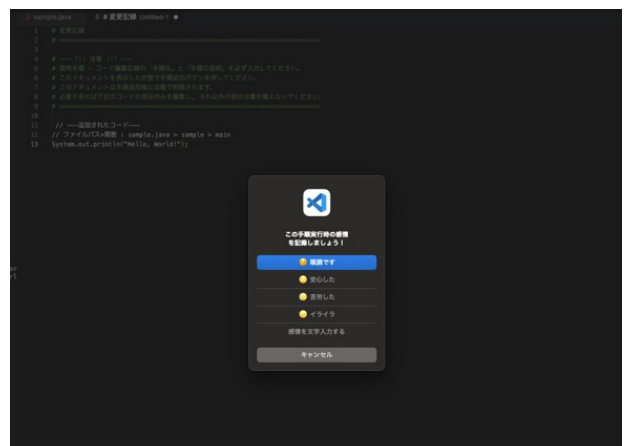


図 2. 手順登録時の感情記録の表示

3 おわりに

本研究では、実践的な開発学習において作業手順を記録する行為を効率化しながら支援するシステムを提案した。本研究での、作業手順の構造化された記録と作業実行時の主観的な感情の記録によって、開発学習へのモチベーション向上や開発者の新たなコミュニケーションの枠組の確立に貢献できると期待される。今後は、記録された感情を一連の流れとして Web アプリケーション上に表示できる、著者の感情に反応できるプロパティを豊富にするなど、ユーザのモチベーション向上体験のためのシステム修正を施し、提案システムのユーザ調査を実施する予定である。

謝辞

本研究の一部は JST CRONOS JPMJCS25K1 の支援を受けた。

参考文献

- [1] J. Kato and M. Goto. DeployGround: A Framework for Streamlined Programming from API playgrounds to Application Deployment. In *Proceedings of the 2018 IEEE Symposium on Visual Languages and Human-Centric Computing, VL/HCC '18*, pp. 259–263, Los Alamitos, CA, USA, oct 2018. IEEE Computer Society.
- [2] J. H. Klemmer, S. A. Horstmann, N. Patnaik, C. Ludden, C. Burton, C. Powers, F. Massacci, A. Rahman, D. Votipka, H. R. Lipford, A. Rashid, A. Naiakshina, and S. Fahl. Using AI Assistants in Software Development: A Qualitative Study on Security Practices and Concerns. In *Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security, CCS '24*, p. 2726–2740, New York, NY, USA, 2024. Association for Computing Machinery.
- [3] T. D. LaToza, D. Garlan, J. D. Herbsleb, and B. A. Myers. Program comprehension as fact finding. *ESEC-FSE '07*, p. 361–370, New York, NY, USA, 2007. Association for Computing Machinery.