GitHub

NEWS RELEASE

2023年10月20日
ギットハブ・ジャパン合同会社

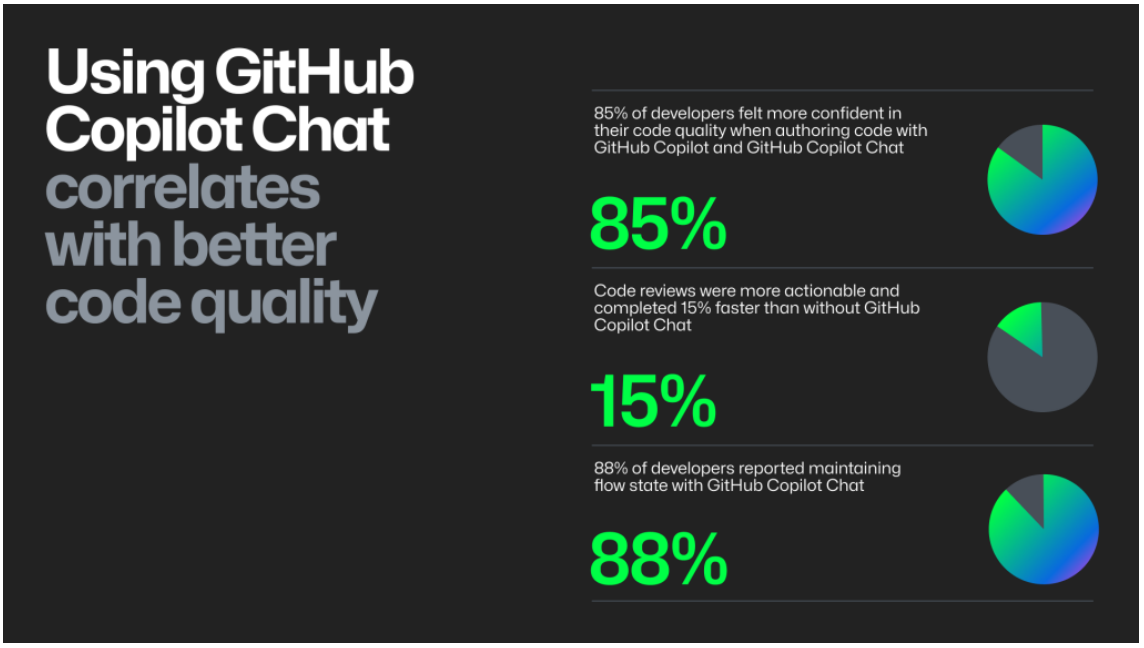
GitHub Copilotの調査結果を公開

～ ソースコードの品質に与える影響を数値化 ～

安全なソフトウェアを開発、拡張、提供するためにAIを搭載した世界最大の開発者プラットフォームを提供するGitHub, Inc.（本社：米国サンフランシスコ）は2023年10月10日(米国時間)、GitHub Copilot Chatに関する新たな調査を公開しました。自然言語の力を活用することで、調査に参加した開発者は、GitHub Copilot Chatを利用してコーディングの具体的な課題に合わせたガイダンス、ヒント、トラブルシューティング、修正、解決策を、IDEを離れることなくすべてリアルタイムで得ることができました。

GitHub Copilot Chatを有効にすることで、本機能の使用経験が無い開発者であっても、作成されたコードやレビューされたコードの品質が全体的に向上することが判明しました。

- 開発者の85%は、GitHub CopilotとGitHub Copilot Chatを使ってコードを作成すると、コードの品質に自信を持てたと感じている。
- GitHub Copilot Chatを使うことで、コードレビューをすぐに始められるようになり、完了までの時間が15%短くなった。
- 開発者の88%は、GitHub Copilot Chatを使うことで集中力が増し、イライラが減り、コーディングがさらに楽しくなったため、ワークフロー状態が維持された。



昨年のGitHub　Copilot調査では、本機能を使った開発者のコーディングのスピードが55%も速いことが明らかになりました。但し、仕事が速いということは、全体の中の一部分でしかありません。従来、何かを迅速に行うことと、何かを正しく行うことは、トレードオフの関係にあります。つまり、AIの処理速度が上がり、AIがさらに多くの開発者のためにコードを作成することに合わせて、優れたコード品質の確保することが一層重要となります。

高品質なコードとは
GitHubでは、コードの品質を測定するために、GitHub内部で使う5つの指標としてルーブリック（rubric：規定）を作成しています。このルーブリックは、学術的(1)および業界的(2)な基準にも合致しています。参加者に、優れたコードと作業速度を低下させるコードを識別する際、以下の指標を用いました。

読みやすさ
コードは、その言語のイディオムや命名パターンに従っているか？　読みにくいコードは、保守、改善、ドキュメント化を難しくします。

再利用性
コードは、再利用できるように書かれているか？　コードの再利用は、開発者コラボレーションの基本です。また、時間と労力を節約し、サイロ化を打破によって全体的な一貫性を創出します。

簡潔性
コードは、**DRY (Don't Repeat Yourself**、同じことを繰り返さない)を守っているか？　コードは、繰り返しが少なければ少ないほど、読みやすく、理解しやすく、開発しやすくなります。複雑なコードは、修正に時間を要するバグや問題を引き起こす可能性があります。

保守性
コードは、機能を明確かつ透過的に、対応中の問題に適切になるように書かれているか？　適切に保守されたコードとは、開発者が依存関係を最小限に抑えることができるコードで

す。保守性の高いコードは、開発者が簡単に検索したりコードを再利用すすることができます。

レジリエント

コードは、エラーを想定して対処できているか？　レジリエントなコードは、エラーがあってもその機能性を維持します(または、少なくとも中断を最小限に抑えます)。簡単に言うと、これはコードを確実に機能させるのに大きな役割を果たします。

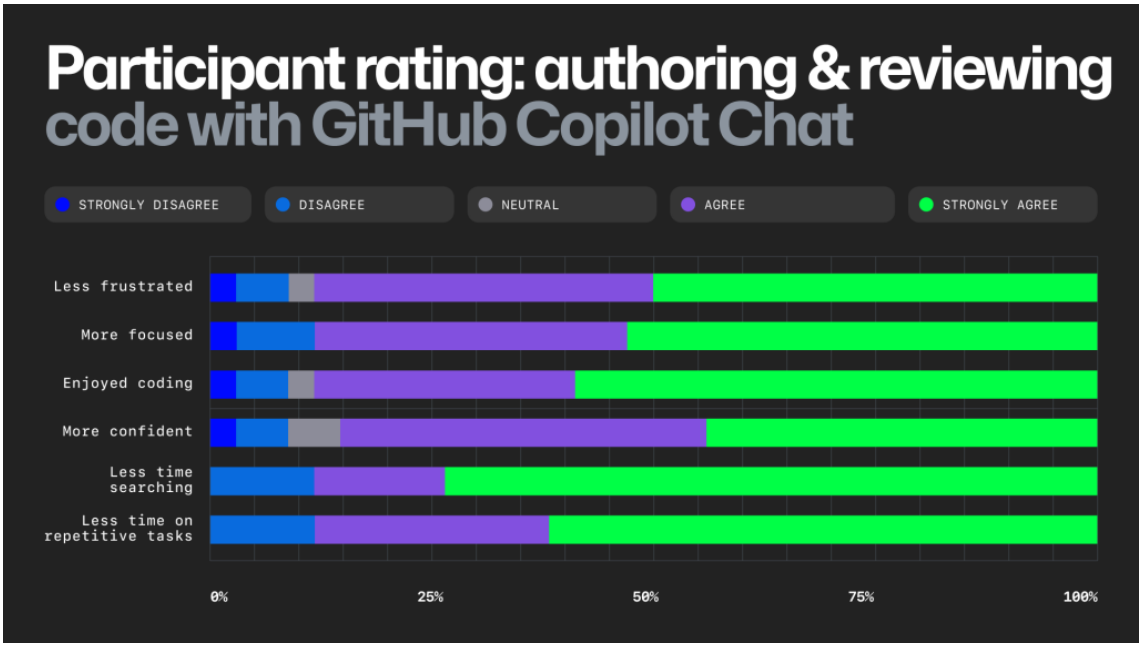
GitHub Copilotの使用とコード品質向上の相関関係

この調査では、GitHub　Copilotとそのチャットボット機能により、生成されたコードの知覚品質が向上し、コードのレビューに費やす時間が短縮され、ユニットテストに合格するコードが生成されるかどうかを調査しました。その結果、すべての指標において、開発者はGitHub Copilotを使うことでコーディングが向上したと感じています。

開発者の**85%**は、**GitHub　CopilotとGitHub　Copilot　Chat**を使ってコーディングすることで、コードの品質に自信を持てたと感じています。GitHub　Copilot　Chatは、GitHub　Copilotと対話できるチャットインターフェイスです。対応しているIDE内から直接、コーディングに関する質問をしたり、回答を受け取ったりすることができます。ドキュメントを確認したりオンラインフォーラムを検索したりしなくても、チャット画面からコーディング情報やサポートにアクセスできます。現在、オープンベータ版のGitHub　Copilot　Chatは、Visual　Studio　CodeとVisual Studioに対応しています。

GitHub Copilot Chatを有効にする方法はこちら

開発者によると、全体的にGitHub　CopilotとGitHub　Copilot　Chatを使った方が、使わない場合よりもコーディングが簡単になりエラーが少なく、読みやすくて再利用可能、簡潔で保守性が高くレジリエントなコードを作成できるため、自信が持てるようになったと回答しています。



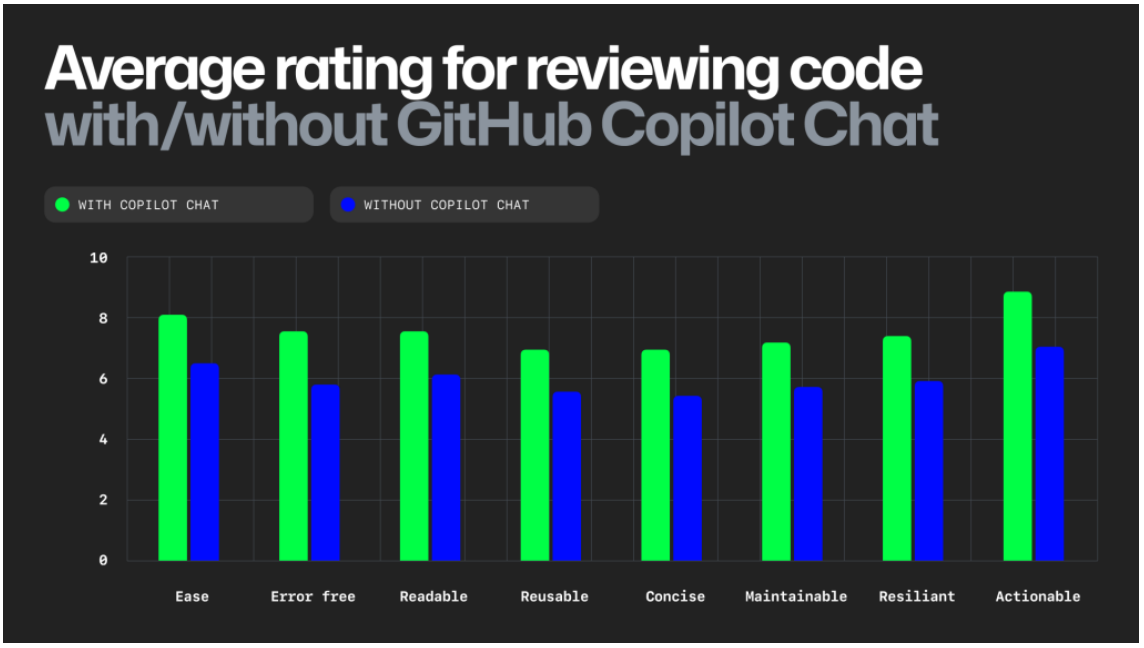
Fortune 500企業のシニアソフトウェアエンジニア(調査参加者)は、次のようにコメントしています。「コードはとてもクリーンで、見ただけで何をしているかがわかり、自分のコードベースにかなり簡単に組み込むことができました。そのため、コードの質はとてもクリーンで理解しやすいと感じました。GitHub Copilotを使ってPull Requestのレビューをしていた時、これは適切なエラー処理コードの生成にとても優れていると実感しました」

GitHub Copilot Chatを使うことで、コードレビューをすぐに始められるようになり、完了までの時間が**15%**も短縮されました**(**初めて使ったユーザーも同様**)**。ここからは、両立が可能になった品質とスピードについてご紹介します。

開発者が、GitHub Copilot Chatをコードレビューに使用するか否かでは、使用することでコードの品質が向上したと回答しています。GitHub Copilot Chatを使ったコードレビューは15%高速になり、コメントを受け入れられる割合も高くなりました。実際、参加者の約70%が、GitHub Copilot Chatを使ったレビュアーからのコメントを受け入れています。

こうした結果は、GitHub Copilot Chatがコラボレーションに与える影響を示しており、大規模な組織で大規模なエンジニアリングチームに導入を拡大することの潜在的な効果が明確に示されています。Pull Requestやコードレビューにかかる時間が短縮され、開発者はより優先度の高い変更に集中することができます。また、最初からより質の高いコードを作成することができるため、後でコードをロールバックする必要がなくなり、追加のテストも不要になります。

あるソフトウェアエンジニア(調査参加者)は、次のようにコメントしています。「実用的なコメントのひとつが、コードの概念的な理解に関するものでした。他の開発者にも私のやっていることを理解してもらいたいので、これは良いフィードバックでした」



開発者の**88%**は、**GitHub Copilot Chat**を使うことで集中力が増し、イライラが減り、コーディングがさらに楽しくなったため、フロー状態が維持されたと報告しています。昨年の調査では、GitHub Copilotを使用した開発者の60～75%が、仕事にやりがいを感じ、コーディング時のイライラが減り、より満足度の高い仕事に集中できたと回答していました。今年の調査では、参加者の88%が同様にイライラが減り、集中力が増したと感じています。その理由のひとつは、IDE内に留まることで検索にかかる時間を減らし、誰もが望む「集中したフロー状態」にある時間を増やすことができるからです。

シニアシステムバリデーションエンジニア(調査参加者)は、次のようにコメントしています。「私は10年以上この業界にいるので、普段はメモ帳やシンプルなプラットフォームでコーディングしています。GitHub Copilot Chatは、GoogleやStack Overflowにアクセスしなくても基本的なことを確認できる、本当に便利なツールでした。しかも、GitHub Copilotには、かなりきちんとした構文が備わっています」

調査の流れ

この調査の目的は、管理された環境でコードを作成する、コードをレビューする、コードレビューで提案された変更を取り入れるというプロセスをシミュレーションすることでした。そのため、参加者にはそれぞれ、コードの作成、コードのレビュー、コードレビューからの提案の確認と変更の採用することを依頼しました。

調査の参加者は、5～10年のソフトウェア開発経験者36名です。GitHub Copilot Chatを使用時と未使用時の両方で、コードの作成とレビューを実施してもらいました(参加者はGitHub Copilotの使用経験はありましたが、GitHub Copilot Chatの使用経験はありませんでした)。

参加者に、オブジェクトの作成、読み取り、削除を行うHTTPサービスのAPIエンドポイントの作成を依頼しました。GitHub Copilot Chatの使用／不使用はランダムに割り当てました。GitHub Copilot Chatを使う参加者は、予め機能についての簡単な動画を視聴してから、作業を開始しました。APIエンドポイントの作成に関する作業のPull Requestを1つ、読み取りと削除の部分についてPull Requestをもう1つ作成してもらいました。

APIエンドポイントのコードを作成した後、参加者に、GitHub Copilot Chatを使用することで記述したコードの品質にどのような影響があったかを比較してもらいました。具体的には、作業がより簡単になったかどうかを尋ねました。例えば、コードのエラーは減ったか、読みやすくなったか、再利用しやすくなったか、簡潔になったか、保守しやすくなったか、レジリエントになったか、などを確認しました。

このセッションの後、レビュー担当の開発者に、別の参加者が作成したPull Requestを2つ割り当てました。どちらのPull RequestがGitHub Copilotを使ったものかを伝えずに、Pull Requestをレビューし、どうすればコードを改善できるかについて提案するよう依頼しました。その後、GitHub Copilot Chatを使った場合と使わなかった場合のレビューの実施プロセスを評価してもらいました。レビュー担当の開発者は、上述のルーブリックを使ってコードの品質を評価し、コードが読みやすいか、再利用可能か、設計は優れているかを測定しました。

別の参加者によるコードレビュー終了後、元のコードを作成した参加者がPull Requestのコメントをレビューし、どのコメントがコードの品質向上に役立ったか、コメントはどの程度実用的であったかを判断しました。繰り返しになりますが、参加者には、どちらのPull Requestのレビューに GitHub Copilot Chatを使ったのかを伝えませんでした。

GitHub Copilot Chatの目標：より質の高いコードを、より迅速に

仕事が速いことと優れた仕事をすることは別物ですが、GitHub Copilot Chatを活用すれば、この2つを両立できます。GitHub CopilotとGitHub Copilot Chatは、開発者の集中力を高め、フロー状態を維持し、仕事に喜びを見出せるように支援することで、開発者の生活向上を目的に開発されました。この調査結果から、こうしたAIツールはそれ以上のことを成し遂げていることがわかりました。GitHubでは、開発者の皆さんがこれから開発していくものに期待を寄せています。

最後に
本調査にご協力いただいた開発者の皆さま、誠にありがとうございました。GitHub Customer Researchは、 GitHub Nextの協力とコンサルティングのもと、本調査を実施しました。

---

(1) Börstler, J., Bennin, K.E., Hooshangi, S. et al. Developers talking about code quality. Empir Software Eng 28, 128 (2023). https://doi.org/10.1007/s10664-023-10381-0
(2) Ghani, U. (2023, September 18). 5 code review best practices – Work Life by Atlassian. Work Life by Atlassian. https://www.atlassian.com/blog/add-ons/code-review-best-practices

---

GitHub Blog
英語：
https://github.blog/2023-10-10-research-quantifying-github-copilots-impact-on-code-quality/

日本語:
https://github.blog/jp/2023-10-20-research-quantifying-github-copilots-impact-on-code-quality/

GitHubに関する情報は、こちらからもご覧いただけます。
Blog: (英語) https://github.blog (日本語) https://github.blog/jp
X: (英語) @github(https://twitter.com/github)
(日本語) @GitHubJapan(https://twitter.com/githubjapan)

【GitHub について**】https://github.co.jp**
GitHubは、すべての開発者のためのグローバルホームとして、安全なソフトウェアの開発拡張、提供するための統合開発者プラットフォームです。フォーチュン100に名を連ねる企業のうち90社に所属する開発者を含む1億人以上がGitHubを利用し、3億3千万以上のリポジトリから、社会に素晴らしいものを創造し送り出しています。GitHubが提供するすべてのコラボレーション機能は、個人やチームがこれまでよりも迅速に、さらに高品質なコーディングをかつてないほどに容易にしています。

【製品／サービスに関するお問い合わせ先】
ギットハブ・ジャパン営業およびサポート窓口
Email: jp-sales@github.com