

Part 1

GitHub Copilot 導入の背景および社内検証

日立製作所のシステム開発における 生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

生成 AI を活用することで、システム開発の効率を飛躍的に向上させたい。2022 年 11 月の ChatGPT 登場以降、このように考える企業は増えているのではないのでしょうか。実際に開発現場で生成 AI の活用を成功させている事例が登場している一方で、具体的にどのように活用すればいいのかわからない、生成 AI を提供してもその活用がなかなか浸透しない、という問題に直面しているケースも少なくないようです。

本ホワイトペーパーでは、日立製作所（以下、日立）における GitHub Copilot の活用を中心に、システム開発現場での生成 AI 活用について、次の 3 つのパートに分けてご紹介します。

Part 1 GitHub Copilot 導入の背景および社内検証

Part 2 システム開発プロセスへの生成 AI 活用
～「日立アプリケーションフレームワーク Justware」との連携～

Part 3 生成 AI のナレッジ共有と人材育成の取り組み
～「生成 AI 実務者コミュニティ」の活動を中心に～

CONTENTS

1-1	GitHub を導入した経緯と理由	2
1-2	開発プロジェクトにおける生成 AI 活用の取り組み	2
1-3	GitHub Copilot の社内評価の概要	3
1-4	初期導入フェーズでのアンケート結果（第 1 回・第 2 回）	4
1-5	本格展開後のアンケート結果（第 3 回）	7
1-6	社内評価のまとめ	9

Part 1 GitHub Copilot 導入の背景および社内検証

1-1 GitHub を導入した経緯と理由

日立は以前より、固定の開発ツールを使用するのではなく、時代の変化やプロジェクトの特性に応じて最適なツールを柔軟に選択することを重視してきました。ソースコード管理ツールに関してもこの方針を適用してきました。

このような中、日立グループ内に蓄積された幅広いナレッジ・アセットを積極的に共有するためのしくみである「Lumada Solution Hub」の一部として、2021 年に主要なソースコード管理ツールに GitHub を導入し、社内向け提供サービスとして本格展開することを決定します。

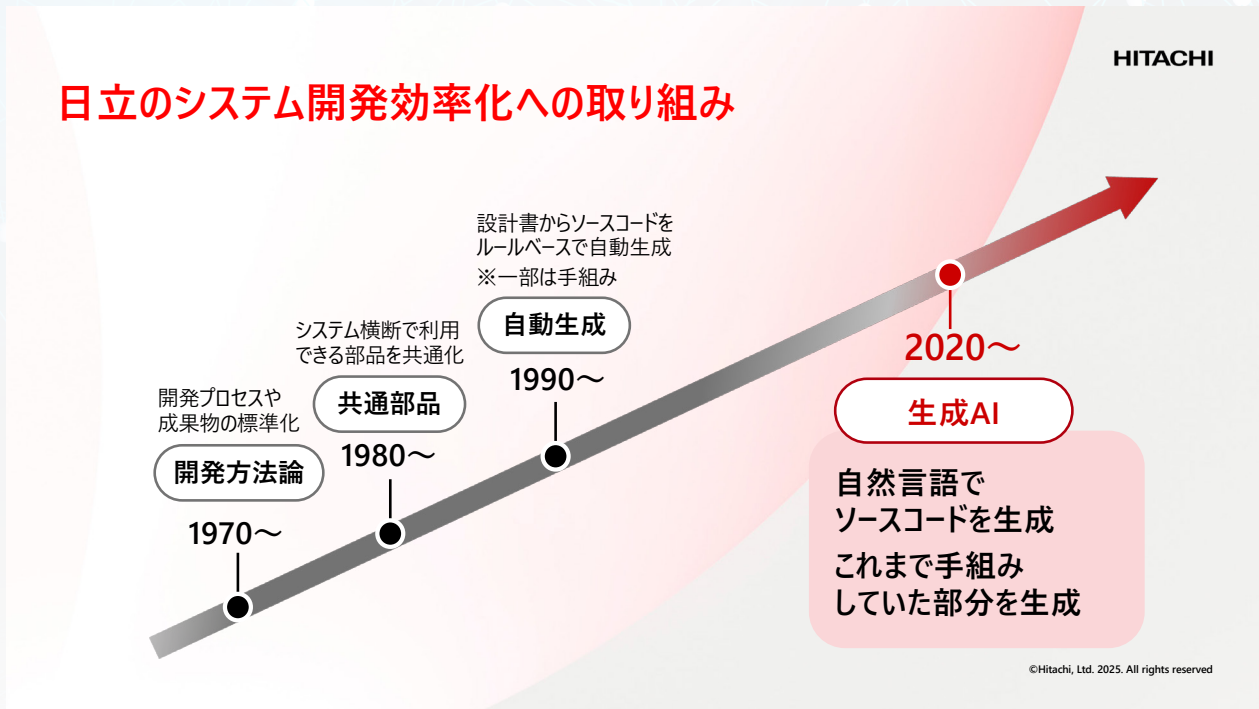
GitHub を選んだ最大の理由は、グローバル スタンダードなソースコード管理ツールであり、世界的に見ても多くのユーザーが利用していることを評価したからです。また日立グループ内のプロジェクトでも GitHub を採用するケースが増えており、これなら会社全体のプラットフォームになる可能性があるとも考えました。

1-2 開発プロジェクトにおける 生成 AI 活用の取り組み

その一方で、生成 AI 活用の取り組みも着々と進められてきました。ここでも基本的なスタンスは、全社標準の生成 AI を決めるのではなく、各プロジェクトの必要に応じて最適なものを選択する、というものです。

日立ではこれまでもシステム開発を効率化するために、さまざまな取り組みを行ってきました。まず 1970 年代には、開発プロセスや成果物を標準化するための「開発方法論 (HIPACE (ハイペース))」を確立し、1980 年代にはシステム横断で再利用できる「部品の共通化」を推進。さらに 1990 年代には、設計書からソースコードをルールベースで「自動生成」するしくみ (後の、日立アプリケーションフレームワーク Justware) も構築しています。

Part 1 GitHub Copilot 導入の背景および社内検証



その後もローコードツールを利用した開発手法などが試行されました。しかし日立の開発案件の中には金融業界や物流業界の複雑なロジックを実装したものが多く、人による高度な判断が必要になるため、さらなる効率化がなかなか進まない、といった状況でした。

このような中で登場したのが生成 AI です。生成 AI であれば自然言語の解釈が可能になり、人が判断しなければならなかったロジックのコーディングも、自動化かつ効率化できるのではないかと考えられました。またその上流の要件定義や基本設計でも、生成 AI なら「人間的な表現」が使えるはずです。そこで 2023 年ごろから、本格的に生成 AI をシステム開発で活用する、という取り組みが始まりました。

その一環として、日立では他社が提供するさまざまな生成 AI をシステム開発に活用することを進めてきました。その中には Azure OpenAI も含まれています。こうした流れの中で、GitHub Copilot の社内評価を実施することになりました。

1-3 GitHub Copilot の社内評価の概要

GitHub Copilot の社内評価は 2023 年にスタートしました。評価対象として GitHub Copilot を選んだのには、大きく 2 つの理由があります。

1 つは、2021 年に GitHub Enterprise の展開を始めており、GitHub Copilot の導入を進める運用体制ができていたこと。

Part 1 GitHub Copilot 導入の背景および社内検証

もう 1 つは、生成 AI を活用する上でのセキュリティや権利侵害リスクに関する懸念に対して、マイクロソフトや GitHub 社の協力によりガイドラインや社内規則の整備や、利用者をサポートする体制を構築することができたことです。

生成 AI 活用では、社内やお客様の機密情報が漏洩するセキュリティ上のリスクはもちろんのこと、第三者の著作権を侵害するリスクも考えなければなりません。生成 AI がインターネット上の情報を学習した結果、他者の著作物を流用した成果物を生成する危険性があるからです。

マイクロソフトと GitHub 社はこのような問題に対して、提案コードの出力にあたっては多重フィルタリングをかける、などの対応を行うことで、著作権を侵害しないように配慮しています。また 2023 年 9 月には「Copilot および Azure OpenAI の出力結果を使用することで生じる可能性のある知的財産権侵害リスクについてマイクロソフトが責任を負う」という「Copilot Copyright Commitment」も発表しています。これらの発表を踏まえ、日立としても十分な評価を行った上で、安心して利用できると判断し、採用を決定しました。

この社内評価ではまず 200 名の開発者を集め、GitHub Copilot を試行してもらいました。具体的な使い方は、コード作成に必要なナレッジを生成 AI に問い合わせる（チャット）、生成 AI にコード提案をしてもらう、といったものがメインとなりました。試行期間で 2 回のアンケート調査を実施し、GitHub Copilot の高い効果と継続利用の期待を確認できたことから、2024 年 4 月より対象を日立グループに拡大し本格的な展開を始めました。

日立では試行期間の 2 回と本格展開後の 1 回の合計 3 回にわたって GitHub Copilot の効果を確認するアンケート調査を実施しています。アンケートの手法としては、開発者の生産性を多面的に評価するためにビクトリア大学と GitHub、Microsoft Research が提唱している、「SPACE フレームワーク」を採用しています。

1-4 初期導入フェーズでのアンケート結果 (第 1 回・第 2 回)

第 1 回と第 2 回のアンケート調査は、2023 年度に日立内の開発者 200 名にて試験利用を行った上で実施。第 1 回は 2023 年 10 月に実施し、回答者数は 62 名でした。第 2 回は 2024 年 1 月に実施し、回答者数は 71 名でした。

このアンケートでは、回答者の業務内容や開発/プログラミング経験などの回答者属性、GitHub Copilot を使用したコーディングで利用した言語、GitHub Copilot を使用したことで得られた生産性向上効果、満足度などを聞いています。

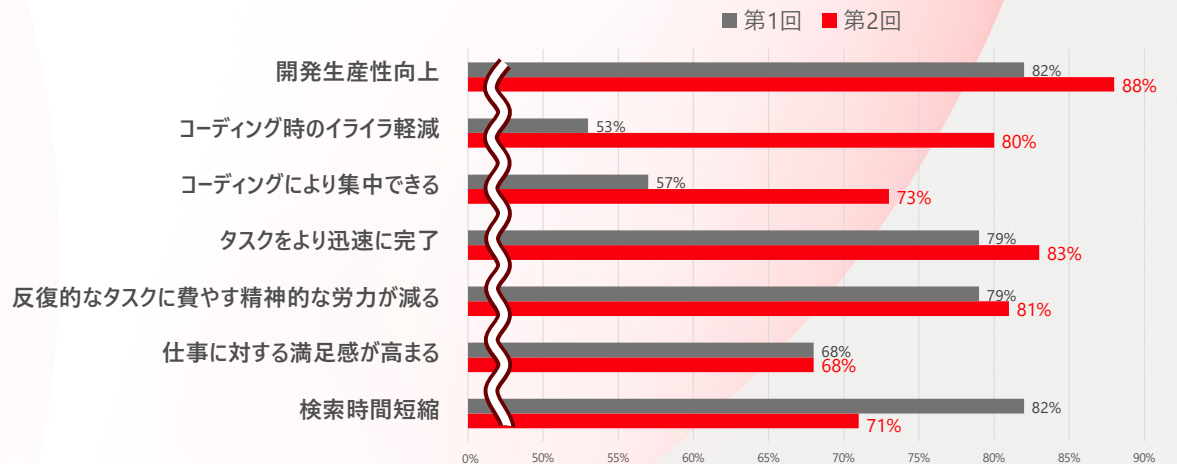
回答者の属性としては、1 日のうち 3 時間程度までをコーディングに費やしている設計/開発者やアプリケーションエンジニアが多く、主要な開発言語は Python や Java、JavaScript、TypeScript、C#、C++、C となっていました。

生産性、満足度/幸福度に関する「とてもそう思う」「そう思う」の回答の合計割合は下記の図の通りとなっています。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 1 GitHub Copilot 導入の背景および社内検証

初期導入フェーズでの社内アンケート結果 (第1回・第2回)



全体的な評価向上。特に「コーディング時のイライラ軽減」「より集中できる」に変化あり。

©Hitachi, Ltd. 2025. All rights reserved

全体的に第1回アンケート調査に比べて、第2回では生産性と満足度/幸福度ともに向上したと言えます。特に「コーディング時のイライラ軽減」や「コーディングにより集中できる」の評価が上がっており、利用者が GitHub Copilot に慣れてきたと考えられます。

さらに第1回の自由回答の設問に対しては、以下のような回答が寄せられました。

設 問	GitHub Copilot を使用することで改善されたこと、助かったことを教えてください。
回答 1	コード作成の効率化 テストコードや関数インターフェイスのテンプレート作成、エラー処理の自動挿入などにより、コーディングの効率が大幅に向上。定型的なコード記述や API サーバーの構築にも役立った。
回答 2	調査 / 検索の時間短縮 ブラウザでの検索時間が削減され、エディター内で調査が完結することで、作業効率が向上した。特に新しい技術の導入や API 調査において効果があった。
回答 3	コメント作成の簡略化 煩わしいコメント記入作業が楽になり、Javadoc などのメンテナンス作業の手助けにもなった。コメントの挿入機能が非常に助かった。
回答 4	作業効率の改善 Copilot Chat の活用により、作業効率が大幅に改善された。キー入力回数が劇的に減り、コーディングスピードが上がった。
回答 5	その他の利点 ライブラリの使用やサンプルコードの出力、ユニットテストの作成など、さまざまな面で業務効率化に貢献した。特に DB などの手続きが多い処理を迅速に開発できる点が助かった。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 1 GitHub Copilot 導入の背景および社内検証

第2回の自由回答の設問に対しては、以下のような回答が寄せられました。

設問 1	実業務に GitHub Copilot/GitHub Copilot Chat を利用した場合に、 開発効率や生産性向上につながると考えられる点を教えてください。
回答 1	コード生成の効率化 GitHub Copilot はコードの自動生成や補完を行うことで、コーディング時間を短縮し、開発効率を向上させる。特に定型的なコードやルーチンワーク的なソースコードの自動補完が便利である。
回答 2	検索と調査の手間削減 GitHub Copilot Chat は、OSS の API の使い方や調査に役立ち、検索して調査する手間を大幅に削減する。これにより、集中力を維持しながら効率的に作業を進めることができる。
回答 3	初心者サポート GitHub Copilot はプログラミング初心者のサポートとして、ポカ ミスの早期発見や実装方法の提案、テストコードの作成補助が可能であり、開発効率や生産性向上につながる。
回答 4	コードの理解と解説 GitHub Copilot は既存コードの理解や解説に役立ち、込み入った書き方のコードを解説することで効率的なコードリーディングを可能にする。
回答 5	テストコードの生成 GitHub Copilot は単体テストや回帰テストなどの定型的なテストコードを書くのに便利であり、テストコードの導入に弾みを付けることができる。

設問 2	GitHub Copilot/GitHub Copilot Chat を使用することで改善されたこと、 助かったことを教えてください。
回答 1	コーディング時間の短縮 多くのコメントがコーディング時間の短縮に言及しており、特に反復的な作業や定型的なコードの生成が効率化された。
回答 2	コード品質の向上 提案されたコードやコメントにより、読みやすいコードが作成できた。バグも軽減され、コード品質が向上した。
回答 3	検索時間の削減 API やライブラリの利用方法の確認、技術的課題の調査など、検索に費やす時間が減少した。
回答 4	テストケースの自動生成 テストケースの生成が自動化され、テスト作成のハードルが低くなった。
回答 5	その他の効率化 設計書の理解、ドキュメント作成、コメントの提案など、コーディング以外の作業でも効率化が図れた。

第2回アンケートの自由回答で興味深いのは、以下のようなコメントもあったことです。

「GitHub Copilot が COBOL のような古い言語を理解するのに役立つ」

「(既に別言語ではスキルのあるエンジニアが) 新しい言語でアプリケーション構築を進めるのを助けてくれる」

Part 1 GitHub Copilot 導入の背景および社内検証

GitHub Copilot は、開発エンジニアのスキルを拡大するための「支援者」としても、大きな貢献を果たせることがわかります。また COBOL アプリケーションのモダナイズを行う場合にも、既存コードを解読するための手助けになりそうです。

1-5 本格展開後のアンケート結果 (第 3 回)

第 3 回アンケート調査は 2025 年 1 月に実施。日立グループの関連会社も含めて 1,800 名まで利用拡大し、回答者数は一気に増え、376 名となりました。

担当業務は設計/開発が圧倒的に多くなり、1 日の中でコーディングに費やす時間は第 1 回/第 2 回に比べて分散しました。開発言語も、最も多いのは Python でしたが、それ以下は C、C++、Java、C# が続いており、多様化が見られました。

まず生産性に関する回答では、「とてもそう思う」「そう思う」の合計割合が、以下のようになりました。

設 問	回答率
開発の生産性が上がっていると感じますか	77%
検索にかかる時間が短縮されますか	77%
タスクをより迅速に完了できるようになりますか	78%

開発者の満足度や幸福度に関しては、以下のようになりました。

設 問	回答率
コーディング時のイライラは軽減されますか	57%
反復的なタスクに費やす精神的労力が減りますか	72%
開発 (コーディング) により集中できていると感じますか	67%
仕事に対する満足感が高まりますか	65%

継続利用の意向に関しては、以下のようになりました。

設 問	回答率
今後の業務でも使いたいと思いますか	91%
同僚やプロジェクトのメンバーに薦めたいと思いますか	86%

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 1 GitHub Copilot 導入の背景および社内検証

「作業の効率化および簡略化が期待できる開発工程を選択してください」という設問への回答は、第2回、第3回のアンケート結果はほぼ変わらず、以下のような割合になりました。

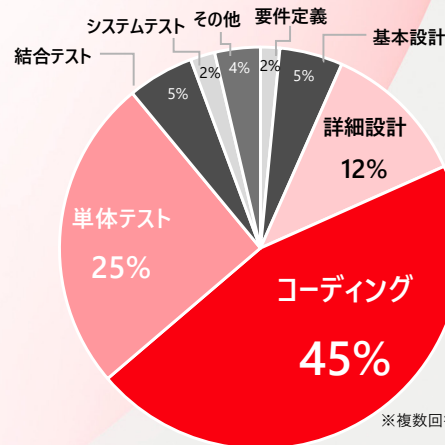
また、「1日あたり何パーセントの時間を節約できましたか」という設問に対して、平均で約20%という回答でした。

**Q: GitHub Copilot / GitHub Copilot Chatを使用することで
作業の効率化・簡略化が期待できる開発工程を選択してください。**

(第2回、第3回の社内アンケート集計結果)

平均して1日あたり約**20%**
の時間が**節約**できたと回答

※節約時間はユーザーの主観によるものです。



※複数回答を集計した結果です。

開発工程の中、コーディングが最も期待されており、次に単体テストと詳細設計が続く

©Hitachi, Ltd. 2025. All rights reserved

自由回答の設問に対しては、以下のような回答が寄せられました。

設 問	生産性の観点で GitHub Copilot を使用することで改善されたこと、助かったことを教えてください。
回答 1	検索時間の短縮 GitHub Copilot の使用により、Web 検索に費やす時間が減少し、必要な情報を迅速に取得できるようになった。
回答 2	コーディングの効率化 GitHub Copilot はコードの補完や提案を行い、コーディングの効率を向上させる。特に、繰り返しの多い作業や複雑なコードの作成が容易になった。
回答 3	コードの理解と修正 GitHub Copilot は他人のコードの理解を助けてくれるので、コードの修正やリファクタリングを効率的に行うことができた。
回答 4	テストコードの作成 GitHub Copilot でテストコードの自動生成を行い、テストの効率が上がった。特に、単体テストやモックアップの作成が容易だった。
回答 5	その他の改善点 GitHub Copilot はコメントの作成やコードレビューの支援、プロトタイプの作成など、さまざまな面でコーディングを効率化してくれた。

Part 1 GitHub Copilot 導入の背景および社内検証

1-6 社内評価のまとめ

生産性、満足度/幸福度、継続利用の意向に関する「とてもそう思う」「そう思う」の回答の合計割合をまとめると、以下のようになりました。

生産性、満足度/幸福度、継続利用の意向に関する 「とてもそう思う」「そう思う」の回答割合

HITACHI

	設 問	第 1 回	第 2 回	第 3 回
開発生産性	開発の生産性が上がっていると感じますか	82%	88%	77%
	検索にかかる時間が短縮されますか	82%	71%	77%
	タスクをより迅速に完了できるようになりますか	79%	83%	78%
開発者の 満足度/ 幸福度	コーディング時のイライラは軽減されますか	53%	80%	57%
	反復的なタスクに費やす精神的労力が減りますか	79%	81%	72%
	開発（コーディング）により集中できていると感じますか	57%	73%	67%
	仕事に対する満足感が高まりますか	68%	68%	65%
継続利用 の意向	今後の業務でも使いたいと思いますか	87%	90%	91%
	同僚やプロジェクトのメンバーに薦めたいと思いますか	—	86%	86%

©Hitachi, Ltd. 2025. All rights reserved

全 3 回のアンケート結果を通して、生産性や満足度/幸福度に関する設問では約 60-90% の高い評価が得られました。継続利用の意向に関する設問についても、約 90% の回答者が継続利用したいと答えています。第 3 回アンケートで数値が低下している理由としては、第 1、2 回目とは異なり社内有志だけでなく、日立グループ内のプロジェクト単位、部署単位での利用があり、管理者やコーディングを行わないメンバーも対象となったためと考えられます。

なお、日立は 2024 年 4 月から GitHub Copilot の本格社内展開をスタートしており、2025 年 4 月の時点で日立グループのユーザー数は 2,500 名を超えています。これを 5,000 名にまで増やすことを目指しています。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 1 GitHub Copilot 導入の背景および社内検証

■ 監修



五十嵐 聡



溝江 彰人



斎藤 岳

日立製作所 アプリケーションサービス事業部 テクノロジートランスフォーメーション本部

生成AIシステムエンジニアリング部
部長・GenAIアンバサダー

LSH適用推進部
担当部長

LSH適用推進部
部長・GenAIアンバサダー

■ お問い合わせ

日本マイクロソフト株式会社

108-0075 東京都港区港南 2-16-3 品川グランドセントラルタワー

株式会社日立製作所

「システム開発への生成 AI 適用支援サービス」 サイト

<https://www.hitachi.co.jp/products/it/appsvdiv/service/genai/>

- ・ Microsoft、Azure、Azure OpenAI は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
- ・ GitHub、GitHub Copilot は、GitHub, Inc. の米国およびその他の国における登録商標または商標です。
- ・ その他記載の会社名、製品名などは、それぞれの会社の登録商標または商標です。
- ・ 本ホワイトペーパーの内容は執筆当時 (2025 年 4 月) のものであり、変更が生じている可能性があることをご了承ください。
- ・ 本ホワイトペーパーは情報提供のみを目的としております。明示的または暗示的を問わず、本ドキュメントにいかなる保証を与えるものではありません。