エージェント・シフト：ソフトウェアエンジニアリングにおける AI の現状と未来の探求

Gemini Deep Research

Part I: 自律的ソフトウェアエンジニアリングの基盤

本章では、AI ソフトウェアエージェントの基本概念を確立し、それが従来の AI ツールからいかにして大きな飛躍を遂げたかを定義する。中心的なケーススタディとして Devin の登場を取り上げ、急成長するオープンソースの動向と比較対照する。

Section 1: アシスタントからエージェントへ：パラダイムシフト

1.1 AI ソフトウェアエージェントの定義：生成 AI を超えて

AI ソフトウェアエージェントとは、特定の目的を達成するために自律的に意思決定を行い、周囲の環境を認識・理解しながら推論を繰り返すことで、最適な行動を選択し、タスクを実行する AI システムである[1]。これは、主にコンテンツを生成するものの、独立した行動を取らない受動的な指示追従型の生成 AI（初期の ChatGPT など）とは根本的に異なる[4]。

AI エージェントの核となる特徴は以下の通りである。

- **自律性 (Autonomy):** 人間の継続的な介入なしにタスクを遂行する能力。ゴールは人間が設定するが、その実行プロセスはエージェント自身が管理する[3]。
- **能動性 (Proactivity):** 指示を待つだけでなく、自ら状況を分析し、目標達成のために行動を起こす能力[4]。
- **推論 (Reasoning):** 収集した情報と自身の知識に基づき、論理的な結論を導き出し、次の行動を計画する能力[2]。

- **計画 (Planning):** 高度で複雑な目標を、実行可能な一連のサブタスクに分解し、戦略的な行動計画を立案する能力[2]。
- **学習と適応 (Learning and Adaptation):** 過去の行動の結果から学び、パフォーマンスを向上させ、変化する環境に柔軟に対応する能力[4]。

これらの特徴により、AI エージェントは単なるツールではなく、「デジタルのチームメイト」として機能することが期待されている[9]。

1.2 コアアーキテクチャ：モデル、ツール、オーケストレーション、メモリ

現代の AI エージェントの能力は、単一の技術ではなく、複数の要素が連携して機能する「エージェント・スタック」によって実現される。その主要な構成要素は以下の通りである。

- **モデル (The Brain):** 主に大規模言語モデル（LLM）が担う、エージェントの「頭脳」。自然言語の理解、推論、計画立案といった中核的な認知機能を担当する[7]。
- **ツール (The Hands):** エージェントが外部環境と対話し、変化をもたらすための手段。これには、コードエディタ、ターミナル（シェル）、ウェブブラウザ、API、データベースなどが含まれる。ツールを使用する能力こそが、エージェントに具体的な行動力を与える[7]。
- **オーケストレーション (The Nervous System):** ワークフロー全体を管理する論理層。ユーザーから与えられた高レベルの目標を達成可能なステップに分解し、どのツールをいつ使用するかを決定し、その結果を処理して次の行動を計画する。この計画、行動、振り返りの連続的なループが、エージェントの自律的な振る舞いを支える[2]。
- **メモリ (The Context):** 過去の対話、観察、行動の履歴を保持する仕組み。これにより、エージェントは長時間の複雑なタスクにおいても文脈を維持し、過去の成功や失敗から学習することが可能になる[2]。

「アシスタント」から「エージェント」への移行は、単なる漸進的な改善ではなく、根本的なアーキテクチャの転換を意味する。その核心的なイノベーションは、より優れた LLM の存在そのものではなく、LLM に「エージェンシー（行為主体性）」、すなわちツールを用いて計画を遂行する能力を与える**オーケストレーション層**にある。初期の AI ツールは、単一のプロンプトに応答するステートレスな存在だった[5]。しかし、エージェントは、モデル（頭脳）とツール（手足）を分離し、それらをオーケストレーショ

ン層が結びつけることで、状態を保持しながら反復的なタスクを実行できる[7]。Devin のようなエージェントが、単なるコードの提案ではなく、リポジトリのクローンからデプロイまでの一連のタスクを完遂できるのは、このオーケストレーションエンジンが複数のステップ、ツール呼び出し、フィードバックを管理しているからに他ならない[10]。したがって、今後の AI エージェント市場における競争優位性は、LLM の性能だけでなく、いかに効果的で信頼性の高いオーケストレーションフレームワークを構築できるかによって大きく左右されるだろう。これは、基盤モデル開発とは異なる、新たな価値と専門性が求められるレイヤーの出現を示唆している。

1.3 エージェント・ループ：AI エージェントの知覚、推論、行動のサイクル

AI エージェントの動的な振る舞いは、一般的に OODA ループ（Observe-Orient-Decide-Act）に類似した、継続的なサイクルによって定義される[2]。NVIDIA は、このプロセスを以下の 4 つの段階でモデル化している[2]。

1. **知覚 (Perceive):** センサー、データベース、API など多様なソースから能動的にデータを収集・統合し、状況を広範に理解する。
2. **推論 (Reason):** 高度な目標をより小さなタスクに分解し、行動計画を立案する。
3. **行動 (Act):** 計画に基づき、適切なツール（コードエディタ、API 呼び出しなど）を使用して環境に働きかける。
4. **学習・振り返り (Learn/Reflect):** 行動の結果を評価し、そのフィードバックを基に次の計画を修正・最適化する[7]。

この反復的なプロセスこそが、エージェントが数千もの意思決定を要する複雑なマルチステップタスクを自律的に遂行することを可能にしている[10]。

Section 2: Devin という触媒：初の自律型エンジニアへのディープダイブ

2.1 Devin の能力の分解：計画からデプロイまで

Cognition 社が開発した Devin は、「世界初の完全自律型 AI ソフトウェアエンジニア」として、この分野のベンチマークを確立した [10]。その能力は、従来のコード生成ツールを遥かに凌駕する。

- **エンドツーエンドの開発とデプロイ:** 要件定義からアプリケーションの構築、デプロイまでを自律的に実行可能 [15]。
- **未知の技術の学習:** ブログ記事などのドキュメントを読むだけで、未知の技術やライブラリの使い方を習得し、タスクに適用できる [10]。
- **自律的なバグ修正:** オープンソースリポジトリの Issue を読み込み、問題を再現し、コードを修正して解決策を提案する [10]。
- **実世界タスクの遂行:** フリーランスプラットフォーム「Upwork」に掲載された実際の開発案件を、要件を伝えるだけで完遂した実績を持つ [15]。

2.2 技術アーキテクチャとワークフローの分析

Devin の内部アーキテクチャの詳細は非公開であるが、公開情報からそのワークフローを推測することは可能である。

- **サンドボックス化された開発環境:** Devin は、人間の開発者が必要とするツール一式（シェル、コードエディタ、ブラウザ）を備えた、隔離されたコンピューティング環境内で動作する [10]。
- **長期的推論と計画:** 複雑なエンジニアリングタスクを数千の個別の意思決定に分解する、高度な長期的推論・計画能力を特徴とする [10]。
- **リアルタイムな協調:** ユーザーに対して進捗をリアルタイムで報告し、必要に応じて設計上の選択肢についてフィードバックを求め、共同で作業を進める能力を持つ [10]。これは、完全な自律性ではなく、人間がループに関与する（Human-in-the-Loop）高度な協調設計がなされていることを示唆する。
- **広範な技術スタックへの対応:** React を用いたフロントエンド、バックエンド API、データベーススキーマの設計といったフルスタック開発から、CI/CD パイプラインの管理、オープンソースへの貢献まで、多岐にわたる言語とフレームワークに対応している [16]。

Devin の最も重要な革新は、その卓越したコーディング能力そのものよりも、**完全なサンドボックス型開発環境と協調的なユーザーインターフェースの統合**にあるのかもしれない。これにより、Devin は単なるツールではなく、「チームメイト」としての体験

を提供する。従来のコード生成ツールは、既存の IDE やチャットウィンドウ内で機能する受動的な存在だった[17]。対照的に、Devin は自身専用のシェル、エディタ、ブラウザを持ち、リポジトリのクローン、依存関係のインストール、Netlify へのデプロイといった、人間が行う一連の作業を自律的に実行できる[10]。さらに、ユーザーと対話し、フィードバックを受け入れながら作業を進めるという協調的なアプローチ[10]は、完全に自律的なエージェントが誤った方向に進むリスクを軽減し、ユーザーの信頼を醸成する上で極めて重要な HCI（ヒューマン・コンピュータ・インタラクション）要素である。したがって、Devin の成功は、AI の能力だけでなく、優れたプロダクトデザインの勝利でもある。今後の競合製品は、単にタスク実行能力を模倣するだけでなく、このシームレスで統合された協調的なユーザーエクスペリエンスを再現する必要に迫られるだろう。

2.3 市場へのインパクト：誇大広告と現実の分離

Devin の発表は、技術コミュニティに大きな興奮と同時に、特にジュニアレベルのエンジニアの将来に対する深刻な懸念をもたらした[15]。開発元である Cognition 社は、発表後すぐに約 40 億ドルという驚異的な評価額を達成し、投資家からの強い関心と市場の期待を証明した[19]。

さらに、金融大手のゴールドマン・サックスが Devin を導入した最初の主要銀行となり、これを人間と AI エージェントが協働する「ハイブリッドな労働力」戦略の一環と位置づけたことは、エンタープライズ領域での本格的な導入に向けた強力なシグナルとなった[19]。

Section 3: オープンソースの逆襲：Devika と OpenDevin/OpenHands

3.1 オープンソース代替品の台頭

Devin のようなプロプライエタリなシステムへの対抗として、その能力を再現し、民主

化することを目的としたオープンソースプロジェクトが急速に台頭した。その代表格が **Devika** [20] と

OpenDevin [22] である。これらのプロジェクトは、Devin の登場に直接的な影響を受け、その代替となることを明確な目標として掲げている [20]。

特筆すべきは、OpenDevin プロジェクトがその後 **OpenHands** へと名称を変更し、発展を続けている点である [25]。

3.2 オープンソースアーキテクチャの比較分析

- **Devika:** そのアーキテクチャは GitHub 上で明確に文書化されている [28]。最大の特徴は、プランナー、リサーチャー、コーダー、ランナーといった専門化されたサブエージェント群を、エージェントコアが統括するモジュラー型のシステムを採用している点である [28]。対応する LLM も幅広く、Claude 3、GPT-4、そして Ollama を介したローカルモデルまでサポートしている [20]。
- **OpenDevin/OpenHands:** こちらも Devin 同様、サンドボックス化された Docker 環境、コマンドシェル、コードエディタ、ブラウザ UI を備えたアーキテクチャを持つ [23]。安定したエージェントフレームワークの構築に重点を置き、CodeAct や SWE-Agent といった様々なエージェント実装を積極的に研究している [30]。さらに、プロジェクト独自にファインチューニングしたモデル「CodeQwen1.5-7B-OpenDevin」を公開するなど、フルスタックでのオープンソース開発への強いコミットメントを示している [32]。

3.3 エージェント AI を加速させるコミュニティの役割

これらのプロジェクトは、世界中の開発者、研究者、そして熱心なユーザーによるコミュニティ主導で開発が進められている [23]。このオープンな協調体制により、様々な LLM、エージェントアーキテクチャ、評価手法の実験が迅速に行われ、単一の企業では達成し得ないスピードでイノベーションが加速する可能性がある。

また、オープンソースであることの最大の利点の一つは**透明性**である。ユーザーはシステムの内部構造を理解し、必要に応じて修正し、そして信頼して使用することができ

る。これは、内部がブラックボックス化されたプロプライエタリなソリューションに対する明確な優位点となる[24]。

Table 1: 主要な AI ソフトウェアエージェントの比較分析

特徴	Devin (Cognition)	Devika (stitionai)	OpenHands (All-Hands-AI)
中核目標	世界初の完全自律型 AI ソフトウェアエンジニア	Devin のオープンソース代替	Devin の再現・強化・革新
アーキテクチャ哲学	プロプライエタリな統合システム	モジュラー型サブエージェント	拡張可能なエージェントフレームワーク
主要コンポーネント	サンドボックス環境, ブラウザ, コードエディタ	プランナー, リサーチャー, コーダー, ランナー	サンドボックス環境, ブラウザ, コードエディタ, CodeAct
対応 LLM	非公開（プロプライエタリ）	Claude 3, GPT-4, Gemini, ローカル LLM	設定可能, CodeQwen などカスタムモデルも
開発状況	プライベート・アーリーアクセス	活発な公開開発	活発な公開開発
協調モデル	ユーザーフィードバック、リアルタイム UI	チャットインターフェース	リアルタイムな協調 UI

出典: [10]

エージェント AI におけるオープンソースの動きは、単に無料のクローンを生み出して

いるだけではない。それは、**研究と人材育成のための並行エコシステム**を醸成している。これらのプラットフォームは、新しいエージェント理論を検証するための生きた実験室として機能し、次世代の AI エンジニアのための訓練場となっている。Devika や OpenHands の GitHub リポジトリは、単なるコードの保管場所ではなく、Issue やプルリクエスト、議論が活発に行われるコミュニティのハブである[20]。これらのプロジェクトは、ベンチマークでの評価向上や新しいアーキテクチャの探求といった、研究指向の目標を明確に掲げている[20]。大学や個人の研究者は、高価なプロプライエタリシステムへのアクセスを必要とせずに、これらのオープンプラットフォームを利用して最先端の AI 研究を実践できる。これは、研究への参入障壁を劇的に下げる。同時に、これらのプロジェクトに貢献する開発者たちは、「エージェント・スタック」に関する実践的な経験を積んでおり、エンタープライズ環境に直接応用可能なスキルを持つ人材プールを形成している。したがって、技術リーダーにとって、オープンソースエコシステムの価値は二重である。一つは社内ツールを構築するための潜在的なプラットフォームとして、もう一つはトップクラスの AI エンジニアリング人材を採用するための豊かな供給源としてである。

Part II: 現実の測定：パフォーマンス、ベンチマーク、そして市場

本章では、理論から実践へと移行し、エージェントのパフォーマンスがどのように測定されているかを批判的に検証し、競争および金融の現状を概観する。

Section 4: 実証の場：SWE-bench への批判的視点

4.1 SWE-bench の方法論の理解

SWE-bench は、AI が実世界のソフトウェアエンジニアリングタスクを解決する能力を評価するための、事実上の標準（デファクトスタンダード）となっている[33]。

- **データセット:** Django や scikit-learn といった著名な Python リポジトリから収集

された、2,294 件の実際の GitHub Issue に基づいている [36]。

- **評価指標:** 中核となる評価指標は**「fail-to-pass test」**である。これは、エージェントが生成したコードパッチを適用することで、当初は失敗していた特定のユニットテストが成功（パス）するようになるか、というものである [37]。これにより、修正が実際に問題を解決したかを客観的かつ堅牢に測定できる。

4.2 リーダーボードのパフォーマンス分析：誰が、なぜリードしているのか

- **初期のパフォーマンス:** 当初、人間の支援なし（unassisted）での解決率は 1.96% と極めて低く、これらのタスクの難易度の高さを示していた [37]。
- **Devin のブレークスルー:** Devin が報告した **13.86%**（unassisted）というスコアは、それまでの全てのモデルを大幅に上回る画期的な成果だった [10]。
- **現在の競争状況:** SWE-bench-Live のリーダーボードを見ると、トップ争いは熾烈である。**OpenHands + Claude 3.7 Sonnet** や **SWE-agent + Claude 3.7 Sonnet** といった組み合わせが**約 17.67%**の解決率を達成している [33]。
- **成功の要因:** これらの結果は、パフォーマンスが基盤となる LLM（例：Claude 3.7 vs GPT-4.1）の能力だけでなく、エージェントを制御する**スキャフォールディング**（足場となるソフトウェア、例：OpenHands vs SWE-agent）の設計にも大きく依存することを示している [33]。

4.3 限界と AI エンジニアリングベンチマークの未来

SWE-bench は標準としての地位を確立しているが、いくつかの限界も指摘されている。

- **スコープの限定:** Python 言語に限定されており、データセットの大部分が Django リポジトリに偏り、タスクの多くがバグ修正である [35]。
- **新たなベンチマークの登場:** このような限界に対応するため、Amazon から **SWE-PolyBench** のような新しいベンチマークが登場している。SWE-PolyBench は、複数言語をカバーするだけでなく、ファイルレベルの特定精度（file-level localization）やコード構文木の分析といった、より詳細なメトリクスを導入している。これにより、タスクの完了だけでなく、エージェントの**コード理解能力**その

ものを測定することを目指している[35]。

- **解釈の複雑さ:** Lite 版や Verified 版といった異なるデータセットのサブセットが存在するため、報告されるスコアを単純に直接比較することは困難である[33]。

Table 2: SWE-bench リーダーボードサマリー（Lite セット）

順位	エージェント/システム	基盤 LLM	解決率 (%)	日付
1	OpenHands	Claude 3.7 Sonnet	17.67	2025/04/30
2	SWE-agent	Claude 3.7 Sonnet	17.67	2025/04/30
3	SWE-agent	GPT 4.1	16.33	2025/04/30
4	SWE-agent	DeepSeek V3	15.33	2025/04/30
5	Agentless	DeepSeek V3	13.33	2025/04/30
6	OpenHands	DeepSeek V3	13.00	2025/04/30
7	Agentless	GPT 4.1	12.00	2025/04/30
8	Agentless	GPT 4o	11.67	2025/04/30

出典: [33]

現在の SWE-bench におけるトップスコア（20%未満）は、AI エージェントが驚くべき能力を持つ一方で、人間のエンジニアを完全に代替するには程遠いことを示してい

る。主要なボトルネックは、単純なコード生成から、より高度な**複雑な推論、コードベースのナビゲーション、そして戦略的な問題解決**へと移行している。SWE-bench のタスクは、大規模で複雑なコードベースを理解し、ファイル間の微妙な相互作用を把握することを要求する[36]。にもかかわらず、最高の性能を持つエージェントでさえ、80%以上のタスクで失敗している[33]。SWE-PolyBench で「ファイルレベルの特定精度」といった新しい評価指標が導入されたことは[35]、研究コミュニティが問題の本質を「AI は正しいコードを書けるか？」から「AI はそもそもコードを書くべき正しい場所を見つけられるか？」へと再認識している証拠である。これは、より高次の認知能力を問うものである。したがって、次のイノベーションの波は、LLM のコーディング構文の精度向上よりも、エージェントの「実行機能」、すなわち検索、アーキテクチャに関する推論、そして一貫した計画を立案する能力の強化に焦点を当てることになるだろう。組織は、単にコード生成を指示するだけでなく、エージェントの高レベルな推論プロセスを導くためのツール投資とエンジニア教育に注力すべきである。

Section 5: エコシステムマップ：主要プレイヤーと投資動向

5.1 巨大企業の戦略：Microsoft, Google, OpenAI

AI エージェント市場の大部分は、少数の巨大テクノロジー企業によって支配されている[39]。

- **Microsoft:** Azure AI プラットフォーム、GitHub Copilot、Microsoft Copilot Studio を擁し、既存の広範なエコシステムと深く統合された、エンタープライズ向けのツール群を提供している[40]。
- **Google:** Gemini モデル、Vertex AI プラットフォームを核に、「Gems」や「Deep Research」といった特化型のエージェントサービスを展開している[41]。
- **OpenAI:** ChatGPT の成功を基盤に、開発者が自社の強力なモデル上で容易にエージェントライクなアプリケーションを構築できる Assistants API を提供している[41]。

5.2 スタートアップのフロンティア：Cognition, Composio　などの革新者

ベンチャーキャピタルに支えられたスタートアップ企業が、この分野のイノベーションを牽引している。

- **Cognition:** Devin の開発元。Peter Thiel などの著名な投資家から支援を受けている [19]。
- **Composio:** 2500 万ドルを調達したインフラストラクチャ・スタートアップ。エージェントが経験から学習し、時間と共に行動を改善するための「共有学習レイヤー」という、現状の技術に欠けている重要な要素の開発に注力している [44]。
- **Delve:** 3200 万ドルを調達し、コンプライアンスという複雑な特定領域に特化した AI エージェントを構築している [45]。
- **その他の動向:** xAI（100 億ドル）や Levelpath （5500 万ドル）といった AI ネイティブ企業への巨額の資金調達は、エージェント AI 分野への莫大な資本流入を示している [46]。

5.3 研究ランドスケープのマッピング：主要な大学研究室

自律型エージェント研究の最前線に立つ学術機関も重要な役割を担っている。

- **スタンフォード大学:** 自律エージェントラボ（人間のような学習に焦点）およびスタンフォード知的システム研究所が中心的な役割を果たしている [47]。
- **カリフォルニア大学バークレー校:** BAIR (Berkeley Artificial Intelligence Research) ラボが有名 [48]。
- **テキサス大学オースティン校:** Peter Stone 教授が率いる学習エージェント研究グループが存在する [49]。
- **その他の主要機関:** MIT、カーネギーメロン大学、トロント大学、バージニア大学なども、この分野における主要な研究拠点である [48]。

AI エージェント市場は、大きく二つの戦略に分岐しつつある。一つは、巨大テクノロジー企業による**プラットフォーム戦略**であり、統合されたエコシステムを提供することを目指している。もう一つは、スタートアップによる**特化型・インフラストラクチャ戦略**であり、コンプライアンスのような特定の垂直市場の問題や、エージェントの学習能力といった基盤的な課題をターゲットにしている。Microsoft 、Google、

OpenAI は、Azure AI、Vertex AI、Assistants API といった広範なプラットフォームを構築し、エージェント AI の「OS」となることを目指している[41]。一方で、Delve（コンプライアンス）[45] や Levelpath（調達）[46] のようなスタートアップは、汎用エージェントではなく、特定の高価値なビジネス課題を解決するためにエージェント技術を応用している。さらに、Composio[44] のような企業は、プラットフォームに依存せず、あらゆるエージェントの性能を向上させるためのインフラストラクチャ、すなわち「共有学習レイヤー」という水平的なソリューションに取り組んでいる。この構造は、クラウドコンピューティングの黎明期に見られた、少数の主要プラットフォームプロバイダー（AWS, Azure, GCP）と、専門化された SaaS 企業およびインフラツールプロバイダーからなる活気あるエコシステムの共存という構図を彷彿とさせる。したがって、技術リーダーが取るべき戦略は単一の選択ではなく、ポートフォリオアプローチであるべきだ。すなわち、中核的な能力は主要プラットフォームを活用し、特定の高 ROI タスクには特化型エージェントを評価し、そして自社のエージェント・スタック全体を強化しうるインフラストラクチャ・スタートアップの動向を注視することが求められる。

Part III: エンジニアリングとビジネスへの変革的インパクト

本章では、AI エージェントがソフトウェアエンジニアの日常業務に与える実践的な影響を分析し、役割、セキュリティ、知的財産に関する広範な意味合いを探る。

Section 6: AI 拡張 SDLC：エンジニアリングワークフローの再定義

6.1 構想からコードへ：エージェント駆動の設計とプロトタイピング

AI エージェントは、開発の最も初期段階を加速させることができる。高レベルの要件からユーザーストーリーを生成し、アプリケーション設計を提案し、大まかなアーキテクチャ図を作成することが可能である[14]。また、プロトタイプを迅速に構築することで、より速い実験とイテレーションを可能にする[5]。

6.2 インナーループの自動化：コーディング、デバッグ、テスト

- **コード生成:** エージェントは、単純なオートコンプリートを超え、関数、クラス、さらにはアプリケーション全体を生成する能力を持つ[15]。
- **デバッグとバグ修正:** エージェントは、バグを自律的に再現し、コードを分析して根本原因を特定し、修正を実装することができる。これは SWE-bench で試される中核的な能力である[9]。
- **自動テスト:** AI はユニットテストやインテグレーションテストを生成し、さらにはバグが含まれる可能性が高いコード領域を予測してテスト活動を集中させることもできる[17]。

6.3 アウターループ革命：CI/CD, DevOps, そして自律的オペレーション

エージェントは、CI/CD パイプラインの管理、デプロイの自動化、ロールバックの実行、アプリケーションのパフォーマンス監視などを通じて、DevOps を変革している[4]。本番環境のメトリクスを分析してパフォーマンスと品質に関する洞察を提供し、ライフサイクル全体を合理化する[51]。

さらに、脆弱性スキャナと統合してセキュリティ問題を自動的に検出し修正することで、パイプライン内でのセキュリティ（DevSecOps）を強化することも可能である[4]。

AI エージェントによる生産性向上の最大の効果は、単なる「インナーループ」（コーディング）の自動化よりも、「アウターループ」（DevOps、CI/CD、監視）の自動化から得られる可能性が高い。インナーループであるコーディングは創造的で曖昧なタスクを多く含み、エージェントは支援はできるものの、斬新な問題や複雑なアーキテクチャ設計には依然として苦戦する[51]。一方で、テストの実行、コードのデプロイ、エラー監視、デプロイ失敗時のロールバックといったアウターループのタスクは、高度に構造化され、ルールベースで定義されている[4]。これらは、計画に従ってツールを使用するというエージェントの能力が最も活かせる領域である。これらのプロセスを自動化することは、エンジニアリング組織全体で膨大な時間を節約し、ヒューマンエラーを削減するため、効率性の向上が飛躍的に大きくなる。一人の開発者のコーディング速度が10%向上することよりも、チーム全体のデプロイとテストの時間が 50%削減されるこ

との方が、組織全体へのインパクトは遥かに大きい。したがって、組織は AI エージェントの導入を、まず DevOps や SRE（Site Reliability Engineering）機能から優先的に検討すべきである。そこが、最も迅速かつ実質的な ROI を実現できる領域だからだ。

Section 7: 新たなエンジニアの役割：戦略家、指揮者、そして監督者

7.1 雇用の代替 vs 役割の変革

AI がデバッグや単純なコーディングといったエントリーレベルのタスクを担えるようになるにつれ、特にジュニアレベルのエンジニアの雇用が代替されることへの大きな懸念が存在する [15]。

しかし、専門家や業界レポートの大勢を占める見方は、大量の雇用代替ではなく、**役割の変革**である。未来は人間と AI の協調によって形作られる [9]。Gartner は、2027 年までにエンジニアの 80%がスキルアップを必要とすると予測している [52]。実際、「生成 AI エンジニア」や「AI 拡張ソフトウェアエンジニア」といった新たな職種が生まれ、AI スキルを持つエンジニアへの需要は急増している [59]。

7.2 AI 拡張エンジニアに不可欠なスキル

エンジニアに求められるスキルの焦点は、低レベルのコーディング作業から、より高次の能力へと移行する。

- **戦略的な問題解決とシステムアーキテクチャ:** 「何を」「なぜ」作るのかを定義し、「どのように」作るのかはエージェントに委ねる [58]。
- **AI の指揮とプロンプトエンジニアリング:** AI エージェントを効果的に指導し、指示し、管理する能力が中核的なコンピテンシーとなる [18]。
- **批判的な評価と監督:** AI が生成したコードやソリューションの品質とセキュリティをレビューし、検証し、保証する責任が極めて重要になる [51]。
- **ドメイン専門知識:** 金融やヘルスケアといった特定の業界知識とエンジニアリング

スキルを組み合わせ、専門的な文脈で AI を導く能力[58]。

7.3 「AI ファースト」なエンジニアリング文化の台頭

この変化は、エンジニアが AI エージェントを単なるツールではなく、協働者として捉える文化的なシフトを要求する[60]。チームは、人間とエージェントの協調、コードレビュー、品質保証に関する新しいワークフローとベストプラクティスを開発する必要がある[53]。

ソフトウェアエンジニアの価値は、「書かれたコードの行数」から「**定義された問題と検証されたソリューションの質**」へとシフトしている。最も価値のあるエンジニアとは、AI エージェントを自身の戦略的思考の「戦力増幅器」として最も効果的に活用できる人物となるだろう。AI エージェントは、明確な指示に基づいた機械的なコーディング作業を得意としつつある[17]。これにより、純粋なコーディングスキルのコモディティ化が進行する。しかし、エージェントには真の文脈認識、ビジネス理解、創造性が欠けており、解決すべき問題そのものを定義することはできない[52]。エンジニアの役割は、高次のスキル、すなわちアーキテクチャ設計、問題解決、プロダクト思考へと移行する[58]。プロセスは次のように変化する：人間が複雑な問題を定義し、それを AI 向けの明確な目標に分解する。AI がその目標を実行し、人間がその出力を検証する。したがって、エンジニアのキャリア開発とトレーニングは、コーディング能力のみに焦点を当てるのではなく、システム設計、プロダクトマネジメントの原則、そして AI エージェントのチームを効果的に管理する「技術」を重視するように再構築されなければならない。エンジニアは、AI からなるチームの「テックリード」となるのである。

Section 8: 新たな脅威：エージェント AI 時代のセキュリティ

8.1 エージェント特有の脆弱性の分析

AI エージェントは、LLM が持つすべてのリスク（プロンプトインジェクション、デー

タ汚染など）を継承するだけでなく、それらに基づいて**行動する能力**を持つため、リスクを増幅させる[62]。

- **プロンプトインジェクション／目標操作:** 攻撃者は、エージェントを騙してセキュリティ機能を無効にしたり、データを外部に送信させたりといった、悪意のあるアクションを実行させるような入力を巧妙に作り出すことができる[62]。
- **サプライチェーンの脆弱性:** エージェントが、タイポスクワッティング（タイプミスを狙った悪意のあるパッケージ）によって不正な依存関係をインストールするよう仕向けられたり、CI/CD パイプラインを改ざんして侵害されたコードを注入させられたりする可能性がある[61]。
- **サンドボックス回避とラテラルムーブメント:** エージェントのサンドボックス環境（例：コンテナ）に脆弱性があった場合、エージェントがそこから脱出してホストシステムにアクセスしたり、内部ネットワークを横断的に移動（ラテラルムーブメント）したりする危険性がある[61]。
- **データ漏洩と IP 侵害:** コードベース、ログ、環境変数に広範なアクセス権を持つエージェントは、企業の独自コードや機密データを盗み出すための格好の標的となりうる[61]。
- **安全でない認証情報管理:** エージェントは機能するために API キーやその他の秘密情報を必要とすることが多い。エージェント自身が侵害された場合、これらの秘密情報も漏洩する[61]。
- **過剰な権限を持つ連携:** GitHub や AWS の管理者権限など、エージェントに過剰な権限を付与することは、巨大なセキュリティリスクを生み出す[61]。

8.2 安全な AI エージェント展開のための戦略

これらの新たなリスクを軽減するためには、多層的な防御アプローチが不可欠である。

- **最小権限の原則:** エージェントには、タスク実行に必要な絶対最小限の権限のみを付与すべきである。まず読み取り専用アクセスから始め、必要な場合にのみ、限定的な時間で書き込み権限を昇格させる[8]。
- **エフェメラル（短命）なランタイム:** エージェントが実行されるサンドボックス環境は、タスク完了後に必ず破棄し、機密データや認証情報が永続化するのを防ぐ[61]。
- **重要なアクションに対する人間による必須の承認（Human-in-the-Loop）:** コードのマージやデプロイといったシステムに重大な変更を加えるすべてのアクション

は、人間のレビューと承認を**必須**としなければならない[8]。

- **厳格なロギングと監査:** エージェントの不透明な活動は大きなリスクである。監査やインシデント対応のために、すべてのプロンプト、推論ステップ、アクションを不変の形で記録する必要がある[61]。
- **入出力のサニタイズ:** エージェントへのすべての入力と、エージェントからのすべての出力（特に実行されるコードやコマンド）は、厳格に検証され、無害化（サニタイズ）されなければならない[62]。

Table 3: AI エージェントのセキュリティ脆弱性と緩和戦略

脆弱性カテゴリ	説明	攻撃ベクトルの例	緩和戦略
プロンプトインジェクション/目標操作	攻撃者がプロンプトを操作し、エージェントに意図しない有害な行動を取らせる。	悪意のあるウェブページを読み込ませ、会話履歴を外部に送信させる。	厳格な入力検証、明確なシステムプロンプト、人間による監視。
サプライチェーン攻撃	エージェントを騙して、悪意のある依存関係をインストールさせたり、ビルドプロセスを改ざんさせたりする。	`npm install bad-pkg` のようなコマンドを実行させるプロンプト。	依存関係スキャン、署名検証、CI/CD パイプラインのアクセス制御。
サンドボックス回避	エージェントが実行される隔離環境から脱出し、ホストシステムやネットワークにアクセスする。	コンテナランタイムの脆弱性を利用したエスケープ。	最新のセキュアなランタイムの使用、ネットワークポリシーの強化。
データ漏洩	エージェントがアクセスできる機密情報（コード、API キー、	広範なファイルシステムアクセス権を持つエージェントにデ	最小権限の原則、データ損失防止(DLP)ソリューション、アク

	個人情報）を盗み出す。	ータ送信を指示。	セスログの監査。
安全でない認証情報管理	エージェントが使用する API キーやトークンが漏洩し、不正利用される。	侵害されたエージェントの環境変数から API キーを窃取。	シークレット管理ツールの使用、短命なトークン、エフェメラルなランタイム。
過剰な権限を持つ連携	連携するツール（GitHub, AWS）に対して、必要以上の権限をエージェントに付与する。	GitHub リポジトリへの管理者権限を持つエージェントが、保護ブランチのルールを無効化する。	最小権限の原則、スコープを限定したアクセストークンの使用。
不十分な監査	エージェントの行動ログが不十分で、インシデント発生時に追跡や原因究明が困難。	エージェントの意思決定プロセスが記録されず、「なぜその行動を取ったか」が不明。	すべてのプロンプト、推論、アクションの不変なロギング。

出典: [8]

AI エージェントの導入は、従来のセキュリティモデルを根本から覆す。静的なコードやインフラを保護するだけでなく、今や、システムへの特権アクセスを持つ**動的で自律的な意思決定主体**そのものを保護しなければならない。従来のアプリケーションセキュリティは、コード自体の脆弱性（SQL インジェクションなど）に焦点を当てていた[63]。しかし、AI エージェントは静的なコードではなく、コードを

書き、コマンドを**実行し**、外部システムに**アクセスする**プロセスである[61]。攻撃対象領域（アタックサーフェス）は、もはやアプリケーションのエンドポイントだけでなく、エージェントの「思考」、すなわちプロンプトインターフェース、意思決定ロジック、そして利用可能なツール群そのものに拡大する。「目標操作」[62]や「プロンプト誘導型攻撃」[61]といった脆弱性は、従来のソフトウェアのバグではなく、エージェントの認知プロセスを悪用するものである。したがって、セキュリティ対策は「コードの保護」から「

エージェントの統治（ガバナンス）」へと進化しなければならない。これには、ラン

タイム監視、行動分析、厳格なアクセス制御（最小権限）、そしてシステムに変更を加えるあらゆるアクションに対する人間による必須の監督といった、新たな管理策が求められる。これは、AppSec から「AgentSec」へのパラダイムシフトである。

Section 9: 法的なグレーゾーン：知的財産と AI 生成コード

9.1 人間による創作性の要件：AI 時代の著作権

現在の米国著作権法は「独創的な著作物」を保護の対象としており、裁判所および米国著作権局は、その成立に**人間の著作者**の存在を要求すると解釈している[64]。十分な人間の創造的インプットなしに AI のみによって生成されたコンテンツは、一般的に著作権保護の対象外と見なされ、パブリックドメイン（公有）に属する可能性がある[65]。

ここでの法的な鍵となるのは、人間の関与の度合いである。米国著作権局のガイダンスによれば、「重要なのは、人間がその著作物の表現に対してどの程度の創造的コントロールを及ぼしたか」である[64]。単にプロンプトを提供するだけでは、創造的コントロールが不十分と判断される可能性が高い。

9.2 知的資産を保護するためのリスク緩和戦略

- **著作権保護の欠如:** AI が生成したコードに著作権が認められない場合、競合他社がそのコードを自由に利用、複製、改変できることになり、重大なビジネスリスクとなる[65]。
- **人間による創作性の証明:** 企業は、創造的なプロンプトの考案、AI が生成した要素の選択・配置、そして人間のエンジニアによる大幅な修正や改良といった、創造的プロセスにおける人間の関与を綿密に文書化する必要がある[65]。
- **代替的な IP 保護:** 著作権が利用できない場合でも、他の形態の知的財産保護に頼ることができる。
    - **営業秘密:** コードを機密情報として管理する場合。

    - **特許:** 新規性のあるアルゴリズムやプロセスに対して。
    - **契約:** ライセンス条項、秘密保持契約（NDA）、雇用契約を通じて[65]。
- **学習データに関する著作権侵害:** 別の法務リスクとして、LLM の学習に使用されたデータセットの問題がある。著作権で保護されたコードを許可なく学習に利用することは、著作権者の複製権を侵害するとの主張に基づき、訴訟が進行中である[64]。これらの訴訟の結果は、商用 AI モデルの利用の合法性やコストに大きな影響を与える可能性がある。

知的財産権に関する未解決の問題は、中核的な製品開発を AI エージェントに大きく依存する企業にとって、重大かつ定量化が困難な**偶発債務**を生み出している。この法的な不確実性は、リスクを嫌う業界での導入を遅らせるか、あるいは IP（知的財産）に関する補償を提供するツールへの選好を促す可能性がある。企業の主要な資産は、多くの場合、そのプロプライエタリなソースコードであり、著作権はこれを保護する主要なメカニズムである。しかし、人間のインプットが不十分な AI 生成コードは著作権で保護されない可能性がある[64]。もし企業が AI エージェントを用いて中核製品を開発し、競合他社にそれを模倣された場合、そのコードがパブリックドメインと見なされれば、法的な対抗手段を失うかもしれない。これは、企業の存続に関わるビジネスリスクである。さらに、学習データを巡る進行中の訴訟[64]は、将来の裁判所の判断次第で、特定の AI モデルの使用が著作権侵害と見なされる可能性を意味する。そうなれば、企業は開発パイプラインの重要な部分を破棄し、再構築することを余儀なくされるかもしれない。したがって、法務およびリスク管理チームは、AI エンジニアリングエージェントの調達と展開に深く関与しなければならない。技術リーダーの意思決定プロセスには、今や、ベンダーの IP 補償ポリシーの評価と、防御可能な IP ポートフォリオを構築するための人間による創作性を文書化する社内プロセスの整備が含まれなければならない。

Part IV: 今後の道筋：戦略、協調、そして未来展望

本章では、これまでの分析を統合し、人間と AI のインターフェースに焦点を当てながら未来志向の視点を提供し、組織と個人に対する実行可能なロードマップを提示する。

Section 10: ヒューマン・エージェント・インターフェース：HCI の次なるフロンティア

10.1 自律型システムにおける信頼と説明可能性の構築

エージェントがより自律的になるにつれて、その行動が信頼でき、理解可能であることを保証することが、普及のための鍵となる[66]。これは、単なる「ブラックボックス」な推薦から脱却し、意思決定に対する明確な根拠（説明可能 AI、XAI）を提供することを意味する[66]。これはデバッグ、検証、そして規制遵守のために不可欠である。

10.2 効果的な人間と AI の協調と監督のための設計

この分野では、AI エージェントのための HCI（ヒューマン・コンピュータ・インタラクション）という新たな研究領域が生まれている。研究の焦点は、シームレスで直感的なインタラクションパラダイムの創出にある[67]。主要な研究テーマには、人間がエージェントを効果的に監督し、フィードバックを与え、必要な時に介入できるインターフェースの設計が含まれる[66]。

研究者たちは、協調的なシステムを設計・比較するための構造化された語彙を創出するために、対話型アシスタンスやコマンド駆動型アクションといったインタラクションタイプの分類体系（タキソノミー）を開発している[70]。最終的な目標は、AI の自律性と人間のコントロールとの間で適切なバランスを見つけることである。

AI ソフトウェアエージェントの最終的な成功は、その純粋な技術的能力よりも、**ヒューマン・コンピュータ・インタラクション（HCI）設計の質**に大きく依存するだろう。90%の精度を持つが不透明で制御不能なエージェントよりも、80%の精度であっても透明性があり、信頼でき、シームレスな人間との協調を可能にするエージェントの方が、結果としてより有用である。タスクは複雑であり、SWE-bench が示すようにエージェントは完璧ではないため、エラーは必ず発生する[33]。エージェントが失敗したとき、人間の開発者は「なぜ」失敗したのかを理解し、それを修正する方法を知る必要がある。これには説明可能性と透明性が不可欠である[8]。未来が協調的なものであることは、多くの専門家が指摘するところであり、協調には効果的なコミュニケーション、共通理解、そして信頼が必要となるが、これらはすべて HCI の中核的な原則である[57]。

Devin が成功していると認識されている理由の一つは、進捗を報告しフィードバックを受け入れるという、その協調的な UI にある [10]。これは HCI の重要な特徴である。したがって、AI エージェントを開発または導入する企業は、開発者のユーザーエクスペリエンスに重点的に投資しなければならない。エージェントを管理し、観察し、協調するためのインターフェースは、エージェントの基盤となる AI モデルそのものと同じくらい重要なのである。最も成功する製品は、この人間と AI のパートナーシップを極めたものになるだろう。

Section 11: 戦略的提言と将来の軌跡

11.1 AIエンジニアリングエージェント導入のための組織的ロードマップ

- **小さく始め、高い ROI を狙う:** CI/CD パイプラインの自動化やユニットテストの生成など、ROI が測定しやすい明確に定義された領域でパイロットプロジェクトを開始する [12]。「AI ムーンショット」のような壮大な計画は失敗する可能性が高いため避けるべきである [71]。
- **スキルアップへの投資:** システムアーキテクチャ、プロンプトエンジニアリング、AI ガバナンスといった新たな必須スキルについて、エンジニアを積極的にトレーニングする [52]。
- **ガバナンスフレームワークの確立:** 広範な展開の前に、セキュリティ、IP 管理、人間による監督に関する明確なポリシーを策定する。Section 8 で概説した技術的な管理策を実装する [8]。
- **効率性と人材育成のバランス:** ジュニアレベルの職務を完全に廃止することには慎重になるべきである。ジュニアエンジニアに AI と協働させ、検証、テスト、そしてより高次のスキルの学習に焦点を当てることで、健全な人材パイプラインを維持する [56]。

11.2エンジニアがキャリアを未来に適応させるための個人ロードマップ

- **基礎を極める:** プログラミング、アルゴリズム、システム設計といった強力な基礎は、以前にも増して重要になる [73]。
- **AI ファーストの思考法を取り入れる:** AI アシスタントやエージェントを協働パートナーとして利用することを学ぶ。GitHub Copilot のようなツールに習熟し、「vibe coding」を実践する [59]。
- **高次のスキルを伸ばす:** AI が苦手とする領域、すなわち複雑な問題解決、アーキテクチャ設計、コミュニケーション、ユーザーニーズの理解に焦点を当てる [52]。
- **専門性を追求する:** MLOps、データサイエンス、セキュリティ、あるいは特定の業界ドメインといった需要の高い分野での専門知識を深める [59]。

11.3専門家の予測：2027 年以降の AI ソフトウェアエンジニアリングの姿

- **ハイブリッドな労働力が標準に:** AI エージェントと人間のエンジニアが並行して作業し、エージェントが日常的な実装や運用タスクの大部分を担うようになる [19]。
- **ドメイン特化型エージェントの台頭:** 汎用エージェントも進化するが、最大の価値は金融、ヘルスケア、コンプライアンス、セキュリティといった特定の業界やタスクに特化したエージェントから生まれる [71]。
- **コード生成から意思決定へ:** AI の役割は、コンテンツ生成から、ビジネス上の意思決定を駆動し、複雑な運用ワークフローを自動化する方向へとシフトし続ける [71]。
- **ハイプサイクルの終焉？** Gartner は、不適切な適用やプロセスの再設計の失敗により、2027 年までにエージェント AI プロジェクトの 40%以上が中止されると予測している。これにより、より現実的で価値主導のアプローチが主流となるだろう [72]。

AI エンジニアリングエージェントの導入成功は、技術的な課題というよりも、**組織的な変革管理の問題**である。ワークフロー、役割、ガバナンスを根本的に見直すことなく、単に AI エージェントを「導入」するだけの企業は、最小限の ROI しか得られず、高い失敗率に直面するだろう。Gartner は、多くのプロジェクトが「不適切に適用」され、企業が「プロセスの再設計」を怠るために失敗すると明確に警告している [72]。McKinsey は、AI 成熟度の最大の障壁が「リーダーシップの自己満足」であると指摘している [72]。成功裏の導入には、スキルアッププログラム [59]、新しいガバナンスポリシー [8]、エンジニアのための改訂されたキャリアパス [59]、そして人間と AI の協調に向けた文化的シフト [58] といった、多くの非技術的な変革が必要である。これらはすべて、単純なソフトウェアの展開ではなく、大規模な組織変革の典型的な要素である。したが

って、この技術が企業内で最終的に成功するかどうかは、この包括的な変革を推進するリーダーシップの能力によって決定される。その責任は CTO だけでなく、CHRO（人材開発担当）や COO（プロセス再設計担当）を含む経営陣全体にある。

引用文献

1. www.scsk.jp, 7 月 26, 2025 にアクセス、https://www.scsk.jp/sp/itpnavi/article/2025/07/ai_agents.html#:~:text=%E8%A7%A3%E8%AA%AC%E3%81%97%E3%81%BE%E3%81%99%E3%80%82AI%E3%82%A8%E3%83%BC%E3%82%B8%E3%82%A7%E3%83%B3%E3%83%88%E3%81%AE%E5%AE%9A%E7%BE%A9,%E3%81%99%E3%82%8BAI%E3%81%AE%E3%81%93%E3%81%A8%E3%81%A7%E3%81%99%E3%80%82
2. AI エージェントの定義。2025 年の最重要 AI 用語の概念を整理 - Laboro.AI, 7 月 26, 2025 にアクセス、 https://laboro.ai/activity/column/engineer/aiagent/
3. 自律型 AI の仕組みと活用事例は？導入のメリットや注意点を紹介, 7 月 26, 2025 にアクセス、https://www.ntt.com/business/services/xmanaged/lp/column/autonomous-ai.html
4. 自律型 AI とは：DevOps とセキュリティのための AI エージェントを理解する - GitLab, 7 月 26, 2025 にアクセス、 https://about.gitlab.com/ja-jp/topics/agentic-ai/
5. AI Agents: Transforming Software Engineering for CIOs and Leaders | Gartner, 7 月 26, 2025 にアクセス、 https://www.gartner.com/en/articles/ai-agents-transforming-software-engineering
6. AI エージェントとは？特徴や生成 AI との違い、種類や活用シーンを紹介 - AIsmiley, 7 月 26, 2025 にアクセス、 https://aismiley.co.jp/ai_news/what-is-ai-agent-introduction/
7. 【エンジニア向け】 AI エージェントの特徴と活用方法｜Gemini - Google の AI - note, 7 月 26, 2025 にアクセス、https://note.com/google_gemini/n/n3dfda224f268
8. AI Agents in Action: Scaling Impact Across the Enterprise - Klover.ai, 7 月 26, 2025 にアクセス、 https://www.klover.ai/ai-agents-in-action-scaling-impact-across-the-enterprise/
9. Devin AI: Redefining Software Development - HashStudioz Technologies, 7 月 26, 2025 にアクセス、 https://www.hashstudioz.com/blog/devin-ai-redefining-software-development/
10. Introducing Devin, the first AI software engineer - Cognition AI, 7 月 26, 2025 にアクセス、 https://cognition.ai/blog/introducing-devin
11. Deep Dive on Devin: The AI Software Engineer | Scalable Path ®, 7 月 26, 2025 にアクセス、 https://www.scalablepath.com/machine-learning/devin-ai
12. AI エージェントの仕組みをわかりやすく解説｜非エンジニアでも理解できる基礎知識, 7 月 26, 2025 にアクセス、

https://nocoderi.co.jp/2025/04/03/ai%E3%82%A8%E3%83%BC%E3%82%B8%E3%82%A7%E3%83%B3%E3%83%88%E3%81%AE%E4%BB%95%E7%B5%84%E3%81%BF%E3%82%92%E3%82%8F%E3%81%8B%E3%82%8A%E3%82%84%E3%81%99%E3%81%8F%E8%A7%A3%E8%AA%AC%EF%BD%9C%E9%9D%9E%E3%82%A8/
13. How Does Devin AI Works ? - GeeksforGeeks, 7 月 26, 2025 にアクセス、https://www.geeksforgeeks.org/artificial-intelligence/how-does-devin-ai-works/
14. Tech Trend 01: AI-augmented software development: A new era of efficiency and innovation, 7 月 26, 2025 にアクセス、https://www.ey.com/en_in/services/technology/ai-augmented-software-development-a-new-era-of-efficiency-and-innovation
15. 自律型 AI「Devin」を見て、ソフトウェア開発の未来を見た気持ちになる - Qiita, 7 月 26, 2025 にアクセス、https://qiita.com/ay5399/items/30b7347b74778a76fd7a
16. Devin 2.0: The AI Software Engineer That's Revolutionizing Development - DEV Community, 7 月 26, 2025 にアクセス、https://dev.to/aibughunter/devin-20-the-ai-software-engineer-thats-revolutionizing-development-18p4
17. AI Agents in Software Engineering: The Next Frontier of Development - Index.dev, 7 月 26, 2025 にアクセス、https://www.index.dev/blog/ai-agents-software-development
18. A Beginner's Guide to AI-Augmented Software Development - The Ninja Studio, 7 月 26, 2025 にアクセス、https://www.theninjastudio.com/blog/a-beginners-guide-to-ai-augmented-software-development
19. Goldman Sachs Becomes First Major Bank to Use AI Agent Devin, Signaling Shift Toward Hybrid Workforce | by ODSC, 7 月 26, 2025 にアクセス、https://odsc.medium.com/goldman-sachs-becomes-first-major-bank-to-use-ai-agent-devin-signaling-shift-toward-hybrid-adeb746a2d55
20. stitionai/devika: Devika is an Agentic AI Software Engineer ... - GitHub, 7 月 26, 2025 にアクセス、https://github.com/stitionai/devika
21. shreeramdrao/Devika-Agentic-AI - GitHub, 7 月 26, 2025 にアクセス、https://github.com/shreeramdrao/Devika-Agentic-AI
22. OpenDevin - AI Agent Store, 7 月 26, 2025 にアクセス、https://aiagentstore.ai/ai-agent/opendevin
23. OpenDevin - smashing.tools, 7 月 26, 2025 にアクセス、https://smashing.tools/ai/open-devin
24. Devika AI - Software Engineer Website India, 7 月 26, 2025 にアクセス、https://devikaai.co/
25. All-Hands-AI/OpenHands: OpenHands: Code Less, Make More - GitHub, 7 月 26, 2025 にアクセス、https://github.com/All-Hands-AI/OpenHands
26. OpenHands: An Open Platform for AI Software Developers as Generalist Agents - arXiv, 7 月 26, 2025 にアクセス、https://arxiv.org/abs/2407.16741
27. OpenHands (formerly OpenDevin). Installation OpenDevin Guide | by Kenji -

Medium, 7 月 26, 2025 にアクセス、 https://medium.com/@kenji-onisuka/openhands-formerly-opendevin-fde9f4b53bdb
28. Devika-Agentic-AI/ARCHITECTURE.md at master - GitHub, 7 月 26, 2025 にアクセス、 https://github.com/shreeramdrao/Devika-Agentic-AI/blob/master/ARCHITECTURE.md
29. Sahaj777/devika-ai-SDE - GitHub, 7 月 26, 2025 にアクセス、 https://github.com/Sahaj777/devika-ai-SDE
30. OpenDevin: Code Less, Make More - GitHub, 7 月 26, 2025 にアクセス、 https://github.com/AI-App/OpenDevin.OpenDevin
31. OpenHands: The Open Source Devin AI Alternative - Apidog, 7 月 26, 2025 にアクセス、 https://apidog.com/blog/openhands-the-open-source-devin-ai-alternative/
32. OpenHands/CodeQwen1.5-7B-OpenDevin - Hugging Face, 7 月 26, 2025 にアクセス、 https://huggingface.co/OpenHands/CodeQwen1.5-7B-OpenDevin
33. SWE-bench-Live Leaderboard, 7 月 26, 2025 にアクセス、 https://swe-bench-live.github.io/
34. Raising the bar on SWE-bench Verified with Claude 3.5 Sonnet - Anthropic, 7 月 26, 2025 にアクセス、 https://www.anthropic.com/research/swe-bench-sonnet
35. Amazon introduces SWE-PolyBench, a multilingual benchmark for AI Coding Agents - AWS, 7 月 26, 2025 にアクセス、 https://aws.amazon.com/blogs/devops/amazon-introduces-swe-polybench-a-multi-lingual-benchmark-for-ai-coding-agents/
36. SWE Benchmark: LLM evaluation in Software Engineering Setting | by Sulbha Jain | Medium, 7 月 26, 2025 にアクセス、 https://medium.com/@sulbha.jindal/swe-benchmark-llm-evaluation-in-software-engineering-setting-52f315b2de5a
37. SWE-bench technical report - Cognition AI, 7 月 26, 2025 にアクセス、 https://cognition.ai/blog/swe-bench-technical-report
38. SWE-bench Leaderboards, 7 月 26, 2025 にアクセス、 https://swe-bench.com/
39. www.designveloper.com, 7 月 26, 2025 にアクセス、 https://www.designveloper.com/blog/top-ai-agent-companies/#:~:text=Market%20leaders%3A%20Major%20tech%20firms,of%20the%20AI%20agent%20market.
40. おすすめの AI エージェント 13 製品を比較！ 種類や比較ポイントも解説 - Dify, 7 月 26, 2025 にアクセス、 https://dify.tdse.jp/post_column/251/
41. Top 9 AI Agent Companies for Businesses in 2025 | Comparison Guide, 7 月 26, 2025 にアクセス、 https://www.deligence.com/top-9-ai-agent-companies-for-businesses-in-2025-comparison-guide/
42. 78 Artificial Intelligence (AI) Companies to Know | Built In, 7 月 26, 2025 にアクセス、 https://builtin.com/artificial-intelligence/ai-companies-roundup
43. AI エージェント 人気サービス 12 選！調査や資料自動生成、コーディング支援などにすぐ使える AI エージェントとは？ - AI Market, 7 月 26, 2025 にアクセス、 https://ai-market.jp/services/ai-agents-services/

44. AI startup Composio raises $25 million led by Lightspeed Venture Partners, 7 月 26, 2025 にアクセス、 https://timesofindia.indiatimes.com/business/india-business/ai-startup-composio-raises-25-million-led-by-lightspeed-venture-partners/articleshow/122844348.cms
45. Delve Raises $32M Series A to Build AI Agents for Compliance | Insight Partners, 7 月 26, 2025 にアクセス、 https://www.insightpartners.com/ideas/delve-raises-32m-series-a-to-build-ai-agents-for-compliance/
46. The Week's 10 Biggest Funding Rounds: AI On Top Again, Led By xAI's Massive Raise, 7 月 26, 2025 にアクセス、 https://news.crunchbase.com/agtech-foodtech/biggest-funding-rounds-ai-xai-savvy/
47. Stanford Autonomous Agents Lab: Home, 7 月 26, 2025 にアクセス、 https://www.autonomousagents.stanford.edu/
48. Research Laboratories working in the field of autonomous navigation (SDCs) : r/SelfDrivingCars - Reddit, 7 月 26, 2025 にアクセス、 https://www.reddit.com/r/SelfDrivingCars/comments/njb9tv/research_laboratories_working_in_the_field_of/
49. AI-Lab - Learning Agents - UT Computer Science - University of Texas at Austin, 7 月 26, 2025 にアクセス、 https://www.cs.utexas.edu/~ai-lab/?larg
50. Digital Experience | AI Augmented Software Engineering - Infosys Blogs, 7 月 26, 2025 にアクセス、 https://blogs.infosys.com/digital-experience/emerging-technologies/ai-augmented-software-engineering.html
51. Is There a Future for Software Engineers? The Impact of AI [2025] - Brainhub, 7 月 26, 2025 にアクセス、 https://brainhub.eu/library/software-developer-age-of-ai
52. Will AI Make Software Engineers Obsolete? Here's the Reality, 7 月 26, 2025 にアクセス、 https://bootcamps.cs.cmu.edu/blog/will-ai-replace-software-engineers-reality-check
53. AI-Augmented Development: Transforming Software Engineering - Codewave, 7 月 26, 2025 にアクセス、 https://codewave.com/insights/ai-augmented-development-transforming-software-engineering/
54. Will AI Replace Software Developers? The Role of Prompt Engineering in AI - Vention, 7 月 26, 2025 にアクセス、 https://ventionteams.com/blog/ai-in-software-development
55. AI is already replacing jobs—software development is just the beginning - Matt Hopkins, 7 月 26, 2025 にアクセス、 https://matthopkins.com/business/ai-is-already-replacing-jobs-software-development-is-just-the-beginning/
56. Is AI closing the door on entry-level job opportunities? - The World Economic Forum, 7 月 26, 2025 にアクセス、 https://www.weforum.org/stories/2025/04/ai-jobs-international-workers-day/
57. (PDF) The Future Of Human-Ai Collaboration In Software Development: A Narrative Exploration - ResearchGate, 7 月 26, 2025 にアクセス、 https://www.researchgate.net/publication/385933635_The_Future_Of_Human-

Ai_Collaboration_In_Software_Development_A_Narrative_Exploration
58. AI agents' impact on software engineering - FutureCIO, 7 月 26, 2025 にアクセス、 https://futurecio.tech/ai-agents-impact-on-software-engineering/
59. Future of Software Engineering in an AI-Driven World - Aura Intelligence, 7 月 26, 2025 にアクセス、 https://blog.getaura.ai/future-of-software-engineering-in-an-ai-driven-world
60. Why the Future of Software Belongs to AI Agents - Quadrant Technologies, 7 月 26, 2025 にアクセス、 https://www.quadranttechnologies.com/why-the-future-of-software-belongs-to-ai-agents/
61. The Hidden Security Risks of SWE Agents like OpenAI Codex and Devin AI, 7 月 26, 2025 にアクセス、 https://www.pillar.security/blog/the-hidden-security-risks-of-swe-agents-like-openai-codex-and-devin-ai
62. Mitigating the Top 10 Vulnerabilities in AI Agents - XenonStack, 7 月 26, 2025 にアクセス、 https://www.xenonstack.com/blog/vulnerabilities-in-ai-agents
63. AI Agents Are Here. So Are the Threats. - Unit 42, 7 月 26, 2025 にアクセス、 https://unit42.paloaltonetworks.com/agentic-ai-threats/
64. Generative Artificial Intelligence and Copyright Law - Congress.gov, 7 月 26, 2025 にアクセス、 https://www.congress.gov/crs-product/LSB10922
65. Think While You Are Using AI Coding | Baker Donelson, 7 月 26, 2025 にアクセス、 https://www.bakerdonelson.com/think-while-you-are-using-ai-coding
66. Human-AI collaboration is not very collaborative yet: a taxonomy of interaction patterns in AI-assisted decision making from a systematic review - Frontiers, 7 月 26, 2025 にアクセス、 https://www.frontiersin.org/journals/computer-science/articles/10.3389/fcomp.2024.1521066/full
67. Software Engineering and Human-Computer Interaction, 7 月 26, 2025 にアクセス、 https://engineering.oregonstate.edu/EECS/research/software-engineering-and-human-computer-interaction
68. Research Engineer, Human-Centered AI | OpenAI, 7 月 26, 2025 にアクセス、 https://openai.com/careers/research-engineer-human-centered-ai/
69. AI and Human-Computer Interaction | by Md Shahriare Hossain Arafat | Medium, 7 月 26, 2025 にアクセス、 https://medium.com/@Shahriare/ai-and-human-computer-interaction-481e39f7d032
70. [2501.08774] How Developers Interact with AI: A Taxonomy of Human-AI Collaboration in Software Engineering - arXiv, 7 月 26, 2025 にアクセス、 https://arxiv.org/abs/2501.08774
71. 20 Expert Gen AI Predictions for an Ambitious 2025, 7 月 26, 2025 にアクセス、 https://drive.starcio.com/2024/12/20-expert-gen-ai-predictions-ambitious-2025/
72. AI Agents in Enterprise: Market Survey of McKinsey, PwC, Deloitte, Gartner - Klover.ai, 7 月 26, 2025 にアクセス、 https://www.klover.ai/ai-agents-in-enterprise-market-survey-mckinsey-pwc-deloitte-gartner/
73. AI Developer Roadmap: A 12-Month Learning Path to Mastery | DataCamp, 7 月 26, 2025 にアクセス、 https://www.datacamp.com/blog/ai-developer-roadmap

74. Vibe coding: Your roadmap to becoming an AI developer - The GitHub Blog, 7 月 26, 2025 にアクセス、 https://github.blog/ai-and-ml/vibe-coding-your-roadmap-to-becoming-an-ai-developer/
75. A Developer's Roadmap to Getting Started with AI in 2025 | by Madhukar Kumar - Medium, 7 月 26, 2025 にアクセス、 https://medium.com/madhukarkumar/a-developers-roadmap-to-getting-started-with-ai-in-2025-f3f000ef6770