

顧客志向の反復型プロセス

—リーン・スタートアップとアジャイルの組織的仕組み—

平井直樹

Customer-oriented iterative process:
Organizational structure of lean startup and agile

HIRAI, Naoki

本研究は、顧客のニーズに合ったプロダクトやサービスを開発する「仮説→検証→学習」の反復型のプロセスとして、リーン・スタートアップとアジャイルに焦点を当てる。リーン・スタートアップは革新的なプロダクトやサービスを生み出す新規事業の開発方法であり、アジャイルは顧客の要望に対しフィードバックをもとに改善していくソフトウェアの開発方法である。

変化が激しく不確実性の高い現代では、顧客のニーズは絶えず変化しており、そのニーズに合わないプロダクトやサービスを開発してしまうといったリスクを最小限に抑えたとともに、顧客にとってより良いものを作る方法が求められる。反復型プロセスは、機能横断的に部門などの組織の垣根を超えたチームが必要であり、発見と解決のサイクルを繰り返しながら成功や失敗から学び、改善していく組織が必要となる。顧客価値の実現や創造には、失敗を許容できるような文化や理解があり、市場ニーズに沿うように継続的に修正していく組織的な仕組みが必要となる。

キーワード：反復型プロセス (Iteration Process), リーン・スタートアップ (Lean Startup), アジャイル (Agile), スクラム (Scrum)

1. はじめに

先進的な企業は顧客価値の創造のため変革を迫られており、ビジネススピードは劇的に速くなっている (川口・下村, 2018)。こうしたビジネスの変革として、近年デジタルトランスフォーメーションに注目が集まっている。新たなデジタルサービスは、変革のスピードの速さから、事前に何を作るのかが不明であり、ビジネス規模を予測することは難しく、そのため、どれくらいの作業工数がかかるのかを把握することも難しい。さらに、頻繁な変更や修正が行われ、どこまで進めれば完成するのかという基準も不明確である。Gage (2012) によると、スタートアップ全体の失敗率は75%にも達するという。

こうしたどのようなプロダクトやサービスを

創り出せばよいのかわからない不確実性や急激な変化に対応するには、ユーザーの要望を見つけ出し、ユーザーにとって価値のあるものを作ろうとする顧客中心の考え方が重要であり、リスクを削減するために少しずつ作って検証するといったプロセスが不可欠である (渡辺・ブラウン・小俣, 2019)。

そのためには、仮説とその検証、学習を繰り返す反復型のプロセスが有効であり、ソフトウェア開発ではアジャイルが、新規事業開発ではリーン・スタートアップが注目されている。特にスタートアップをはじめとするベンチャー企業は、創業して日が浅く、人数も少なく、会社の制度も確立していないことが多い。そのため、既存の企業で導入されているような計画と管理を主体とした手法の導入が難しく、そのような状況でいかに製品開発を成功させるかが重

要となる。

リーン・スタートアップは、「Build (構築) → Measure (計測) → Learn (学習)」のフィードバックループを原則とする。顧客に気を配り、当初のビジョンをどうするかさまざまな事象を調整しながら、検証による学びを通して画期的な新製品を開発する方法である (Ries, 2011)。主に事業を立ち上げたばかりのスタートアップに採用されるものであり、検証による学びから、事業を継続するか、方向転換するべきかを検討していく。

アジャイルは、ソフトウェアの仕様について厳密な決定をせず、開発プロセスを進める中で仕様を擦り合わせ、ある程度決まった部分だけを先に開発、リリースすることで、ユーザーのフィードバックをもらい、次の開発に生かしていくソフトウェア開発の手法である。アジャイルは、先進的なソフトウェア開発企業のほか、ネットビジネスを中心とするスタートアップやベンチャー企業で採用されることが多い。

このリーン・スタートアップとアジャイルは、同じような反復型のプロセスであり、顧客を中心にプロダクトやサービスを作るという点から類似しており、たとえば、日本ではリクルート (高槻, 2012) やクックパッド (村田, 2013) が組み合わせて取り入れている。しかし、根底には、新規事業開発とソフトウェア開発という大きな前提の違いがあり、その仮説検証の目的や価値の基準が異なっている。

本研究では、こうした状況を踏まえ、リーン・スタートアップとアジャイルについて、比較、整理を行うとともに、反復プロセスがどのように行われているのか、また、顧客のどこに価値を置いており、そのフィードバックによる学習はどのようなものか、そしてそれを可能とする組織はいかなる特徴を持っているのかを明らかにする。

2. 反復型プロセス

前提として反復型プロセスについて確認す

る。これまでの工業化を前提とした、計画と管理とその遂行を中心とした方法は規模の効率を生み出したが、デジタル化には顧客志向のビジネスモデルと価値のある共創サービスプロセスが必要となる (Kuula, Haapasalo, and Tolonen, 2018)。顧客などの人間を中心とした考え方で発想され、アイデアを検証していくプロセスとして、本稿で取り上げるアジャイルやリーン・スタートアップのほか、顧客開発モデル (Blank, 2003)、デザイン思考 (Brown and Katz, 2011) などが注目されている。

こうしたプロセスに共通していることは、プロトタイプング手法と呼ばれる仮説と検証を繰り返す反復型のプロセスである。このプロセスの目的は、製品やサービス、ソフトウェアのアイデアに関する仮説を素早く検証することであり、失敗したときのリソースの損失を少なくするとともにより優れたアイデアの発見が期待できる (渡壁・野城, 2019)。

反復型プロセスのうち顧客の課題を探し出すことに焦点をあてたものとして、Norman and Draper (1986) の人間中心設計 (Human Centered Design) に端を発するスタンフォード大学や IDEO を中心としたデザイン思考 (Brown and Katz, 2011) があげられる。デザイン思考は、サービスやモノづくりを行う際、その最初のアイデア創出に重点が置かれている。アイデアを創出する主体は、開発者ではなく顧客である。顧客の観察やリサーチ、顧客体験を通じて、本質的な課題を抽出し、その課題解決のアイデアのプロトタイプを作成、評価、検証することにより、アイデアを改善していくプロセスを繰り返す (川口・下村, 2018)。デザイン思考は、この反復活動を通じて新たな顧客価値を生み出そうとするものである。

新規事業において、顧客を開拓していくビジネスモデルとして、顧客開発モデル (Blank, 2003) がある。顧客開発モデルは、アジャイルとともに、リーン・スタートアップの前提となったモデルである。最初から多くのリソースを

投入して製品やサービスを開発、販売するのではなく、顧客に主眼を置き、ニーズの存在や収益の有無を実証することで顧客を開拓、リソースを投入し、事業を拡大していくビジネスモデルである（川口・下村，2018）。

ソフトウェアにおいて、顧客を中心とした反復型の開発がアジャイルである。アジャイルのプロセスは、ソフトウェアに対する顧客のニーズを理解することは難しく、その技術の移り変わりも早いいため、そのような複雑なソフトウェアを前もって完全に設計することは無理があるという考え方である。顧客にとってより良いものをつくるため、顧客との共創を前提とした発見と解決のサイクルを繰り返す組織的な学習を行い、成功や失敗から学び、カイゼンしていく開発手法である（平井，2018）。

スタートアップ企業による新規事業開発の方法であるリーン・スタートアップも顧客を中心としている。いかに他社に先駆けて顧客の新たなニーズを理解し、ニーズに合致した製品やサービスを提供するかが重要視され、そのためには多くの時間や多額のコストをかけるのではなく、低コストで素早く変化を繰り返していく方法である（田村・和田，2015）。

こうした漸進的でプロトタイプ的な手法は、製造業を中心に昔から存在した。工業製品の開発では、試作としてプロトタイプを作成することが重要である。さらに、製品開発では顧客価値のための設計情報に対する要求が高くなっており、実際のプロトタイプを作成する前のシミュレーション力が重要となっている（藤本・朴，2015）。

従来の手続き型の手法に対し、こうした反復型のプロセスに注目が集まっている背景には、ビジネス環境の変化がある。急速にグローバル化が進み、市場も急激な変化を遂げており、VUCA と呼ばれるような不確実性や不透明性が増した環境となっている。従来 of 市場のニーズに応えるために長い時間をかけて調査や分析を行い、製品・サービスを開発していくような

方法は過去のものとなり、市場の不確実性の攻略と顧客への価値提供スピードに大きくシフトしている（田村・和田，2015）。

また、近年になってこうした手法が、製造業以外にも取り入れられ始めた理由として、技術の進化やデジタル化の進展により、プロトタイプの作成コストが大幅に減ったことや、クラウドソーシングやアーリーステージに投資を行うエンジェルによる企業の資金調達先の多様化（Blank，2013）などが挙げられる。

3. アジャイル開発

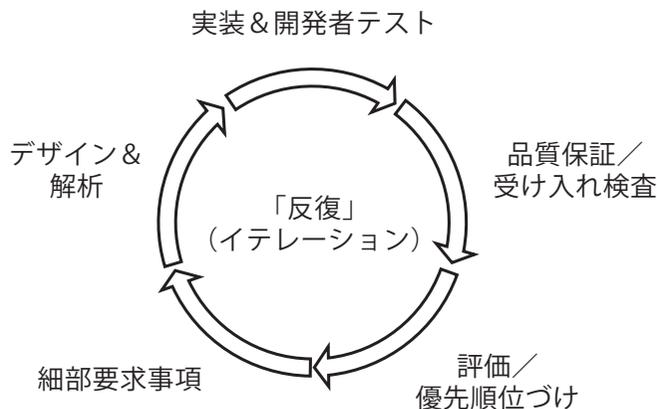
本研究が対象とするアジャイルについてその特徴を確認する。アジャイルは、正確には特定の手法を指しているのではなく、スクラム（Scrum）やXP（extreme programming）と呼ばれる、幾つかの軽量のソフトウェア開発プロセスの総称であり、ウォーターフォール・モデルと呼ばれる多くの手順に従って進んでいく重量級な開発手法と比較される（妹尾，2001）（表1）。

アジャイルは、ソフトウェアの仕様について厳密な決定をせず、開発プロセスを進める中で仕様を擦り合わせ、ある程度決まった部分だけを先に開発していく手法である。これまでソフトウェア開発は、1970～1980年代の市場の変化や成長が比較的安定していた時代を中心に、計画とその遂行、管理を前提としたウォーターフォール・モデルと呼ばれる定型化、標準化した開発プロセスが合致していた（Cusumano，1991，2004）。しかし、この方法では変化の速い市場に対応できず、試行錯誤を経て魅力的な機能を持つようなイノベティブなソフトウェアは期待することができない（妹尾，2001）。アジャイルは、激しい変化を伴う競争環境に対応しようとするものであり、特に解決すべき問題が複雑であったり、解決方法が不明であったりするなか、イノベーションを起こすために、より良いプロダクトを作ることを追求するものである（平井，2020）。

表 1 アジャイルと従来の手法の違い

項目	アジャイル	従来の手法 (ウォーターフォール)
市場環境	顧客の要望と解決方法が頻繁に変わる (変化が多い)。	市場環境は安定的で予測しやすい (変化が少ない)。
予算や納期	製品仕様は変更の可能性がある。創造的なブレイクスルーと市場投入までの時間が重視される。	詳細な仕様と作業計画が予測可能。定められた範囲で計画通りに進めることが重視される。
問題解決の見通し	問題は複雑で解決方法は未知、または開発終了までの道筋もはっきりと定まっていない。	類似の事例があり、問題は明らかとなっており、その解決方法や手順も判明している。
顧客の意思決定	要求仕様はあるが、開発過程が進むにつれ、顧客の望むものがよりはっきりする (目的が変わっていく)。	顧客からの要求仕様は当初から明確となっている。後になっても変化しない (最終目的は変更されない)。
顧客の関与とフィードバック	緊密な共同作業と素早いフィードバックが可能。	常に顧客との共同作業が可能なわけではない (進捗の報告が重要)。
ステークホルダー内の関係	部門にとらわれない横断的な共同作業が極めて重要。上層部からある程度独立している。	部門内で完結することが多い。上層部の影響が強い。
価値の基準	漸進型の開発が価値を持つ。作業は分解でき、短期の反復でできるものを作り、顧客もそれを利用、確認できる。	計画通り、予算通りに製品を開発することに価値がある。顧客は製品が完成するまで確認はできない。
一時的なミスのもたらす影響	1 回の反復の規模が小さいため、大きな失敗にはならない。	規模が大きく、完成付近になってから判明することも多いため、致命的。
開発成果からの学び	次の反復に学びを生かす (成功、失敗に関わらず学習を重ねていく)。	開発工程の後戻りができないため、次の開発まで学習の成果が生かせない。

出所：Rigby, Sutherland, and Takeuchi (2016), 平井 (2019, 2020) をもとに作成。



出所：James and Walter (2010), 平井 (2018) をもとに作成。

図 1 アジャイル開発手法 (スクラム)

3.1 反復プロセス

アジャイルのうち、代表的なスクラムと呼ばれる手法の反復プロセスが図1である。

アジャイルは、開発プロジェクトを進めていくなか、何度も反復（イテレーション）を繰り返す。反復の中では、顧客からの要求に基づくソフトウェアを作成し、その検査とフィードバックをもとに新たな要求の設計や解析といった作業をPDCAサイクルのように繰り返す。

アジャイルではすべての機能を一度に取り込むのではなく、幾つかの機能を選択して開発を行う。さらに、その開発作業も1週間から1か月程度の非常に短い期間で反復して行い、それぞれの反復期間の終了ごとにソフトウェアを本番稼働させることを目指している。完成してからソフトウェアを動かすのではなく、ある程度動くソフトウェアを成長させながら作成する、反復、且つ漸進型が大きな特徴である。顧客の要望をリスト化し、その中から優先度の高いものをできるだけ早く作成し、少しでもできあがったソフトウェアを顧客に確認してもらうことで、早期にフィードバックを得ることを目的としている（平井，2020）。

近年はマーケティングや人事といった、プロダクトの開発以外の分野にもアジャイルは取り入れられ始めており、組織構造そのものをアジャイルに適応させようとする企業も現れ始めている。こうした企業では、たとえば制度の変更を行う際、法律のような緻密性や完璧性を優先するのではなく、6～7割程度の簡素な形でビジネスの現場に導入し、ビジネスの実態に合わせて運用しながら改良している（松丘，2018）。

3.2 顧客価値

アジャイルは、ビジネスや市場に合わせて臨機応変に製品の仕様を変更していくことが可能となる。また、最低限の仕様のみを決め、最速でビジネスを展開、ニーズに対応していくことも可能となり、特にスタートアップやベンチャー企業と相性がよいと考えられ、リーン・ス

タートアップと組み合わせて導入されることも多い。

アジャイルの本質は、単に素早く開発を行うことではない。アジャイルは、反復活動を通じたフィードバックといった顧客の参画の度合いが強く、顧客との対話や交流、人と人とのコミュニケーションやコラボレーション、共創を重視している（平井，2019）。製品開発当初の計画に従うよりも顧客の要望やビジネスの変化への対応を価値としており、最後まで少しでも顧客のために改善していくことがアジャイルの目的である。

3.3 学習

アジャイルにとって重要なことは、失敗から学び、成長できるかである。反復活動を小さく短く繰り返すことは、大きな失敗を避け、リスクを最小化するという観点もあるが、それ以上に小さな失敗を繰り返すことで、多くを学んでいくことが求められる。これは単なる試行錯誤ではなく、失敗をふりかえり、学習できる環境が必要となる。また、失敗を分析するような反省会ではなく、改善のための集まりの場を用意し、開発プロセスや技術についてのほか、チーム自体もふりかえりの対象となる。心理的な安全のもと、チームメンバーが自由に発言し、共感をもって意見を聞く（渡辺・ブラウン・小俣，2019）。失敗が許され、その失敗を糧に同じような失敗を繰り返さないようにメンバーが成長していくことで、最終的に高いパフォーマンスや難易度を達成していくのである（西村・永瀬・吉羽，2013）。

新しいことを体験し、学び、それを繰り返す努力をしながらチームは成長していくのであり、こうした組織的な学びの仕組み、つまり学習する組織がアジャイル、特にスクラムの本質である（平井，2019）。アジャイルは、チームが最終的な結果を改善するために素早いサイクルで学習し、変化に迅速に適應できるようにすることが重要であり、チームや企業の境界を超

え、顧客と会話をし、共創の場の中でイノベーションを作り出していくのである。

3.4 組織

こうしたアジャイルの手法を可能とするのは、開発チームの自己組織化と機能横断的である。自己組織化とは、状況に応じてミッションを実現するための選択を自分たちが決定し、行動できることである。また、機能横断的とは、外部に頼らず仕事ができるチーム、すなわち開発をやり遂げるために必要なあらゆるスキルを持つチームになっていることである（松丘, 2018；市谷, 2019）。

自己組織化された組織では、能動的に目的を実現するために自ら必要な情報を収集し、意思決定をする必要がある。そのためには、問題が発生しても自分たちで解決したり、作業の指示を必要としたりしないようなスキルも持ち合わせた優秀なメンバーが必要となる（西村・永瀬・吉羽, 2013）。

アジャイルで求められていることは、自身の作業範囲を固定せず、全員で協力していくチームである。重要なことは、一人ひとりの能力よりも、チームとしてさまざまな状況に対応できるかどうかである。ソフトウェア開発は、単なるルーチンワークではなく、問題解決や試行錯誤を伴い、創意工夫や無から有を作り出す力が必要であり（Cusumano, 2004）、そのような点を含む限り、難しい作業であり、お互いの経験や得意分野を持ち寄り、協力して助け合わなければならない。それぞれが持つスキルや経験、考え方、性格、得意不得意な分野などを理解し、協力し合って進めていくことで、学び、成長していき、様々な問題を解決していくのである（西村・永瀬・吉羽, 2013）。

4. リーン・スタートアップ

リーン・スタートアップの手法は、成長性の高いハイテク・スタートアップを生み出すために考案された。新規ベンチャーが、早めに失敗

して顧客から学習し、当初のアイデアを修正して、開発サイクルの反復による改善を重ねることで、起業の成功確率を高めようとするものである（Blank, 2013）。

リーン・スタートアップは、不確実性の高い事業、つまり、顧客課題やニーズ、提供価値が不確かな場合に、また、急速に変化する市場に製品・サービスを投入にする場合に適している。従来の経営手法では、顧客やマーケットが判明していることを前提としているが、顧客もマーケットも不明確なスタートアップにはそうした手法は通用せず、固有のマネジメント手法が必要となる（長谷川, 2019）。

表2はリーン・スタートアップと従来の起業手法の違いである。

リーン・スタートアップは、Ries (2011) によって提唱され、顧客開発モデル（Blank, 2003）を中心に、トヨタ生産方式を参考にしたリーン生産方式（Womack and Jones, 2003）のほか、デザイン思考、アジャイル開発など、これまでのマネジメントや製品開発の手法をベースにしており、イノベーションを継続的に生み出そうとするアプローチである。リーン・スタートアップの「リーン」は、価値を生まない生産工程の無駄を徹底して省くことを意味し、リーン生産方式に由来している。トヨタ生産方式は、「ジャスト・イン・タイム」や「自動化」によるムダの徹底的な排除に主眼があり、リーン生産方式はトヨタ生産方式を整理、体系化し、一般化したものであり、同じくムダを排除するところに主眼がある。

「スタートアップ」とは、製品そのものではなくそれを作り出す組織のことを指す（Ries, 2011）が、無駄がないことを意味する「リーン」や生まれたばかりの企業をイメージさせる「スタートアップ」という言葉を使うことで、その適用範囲が狭まってしまう可能性がある（菊池, 2013）。しかし、リーン・スタートアップのアプローチは、スタートアップ企業のみならず、GE グループ、Telefonica といった大企業

表2 リーン・スタートアップと従来の起業手法の違い

	リーン・スタートアップ	従来の起業手法
戦略	ビジネスモデル：仮説重視	事業計画：管理・実行重視
新製品開発プロセス	顧客開発：実際に仮説検証	製品マネジメント：計画駆動の下、製品準備
エンジニアリング	アジャイル開発：修正を重ねていく漸進的な開発	ウォーターフォール開発（一部反復型）：事前に仕様を固めてから開発
組織	顧客対応チームとアジャイル開発チーム：学習意欲、柔軟性、スピードを重視	職能別組織：経験と実行能力を重視
財務報告	重要な指標：顧客獲得コスト、顧客の生涯価値、離反数、クチコミ効果	会計：損益計算書、貸借対照表、キャッシュフロー計算書
失敗	予想される事態：アイデアを練り直し、軌道修正もする	例外的な事態：幹部を更迭して立て直しを図る
スピード	迅速：妥当なデータを基に事業を運営	計画通りのスピード：完全なデータを基に事業を運営

出所：Blank（2013）をもとに筆者作成。

の新規サービスの開発でも取り入れられており、消費者向けに留まらず、産業向けサービスやハードウェアの領域にも拡大している（田村・和田，2015）。

4.1 反復プロセス

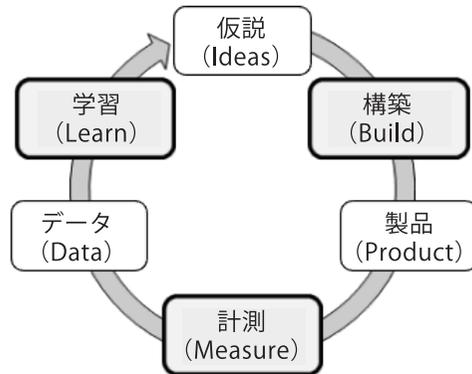
リーン・スタートアップは、反復型のプロセスのように、仮説を構築し、製品を実装、その検証を通じて軌道修正していく、「Build（構築）→ Measure（計測）→ Learn（学習）」のフィードバックループを迅速に繰り返していく。

図2がリーン・スタートアップのフィードバックループである。

新規事業開発には、必ず最初に仮説となるアイデアが存在する。そのアイデアをもとにプロダクトをつくり、ユーザーに利用してもらい、その反応を計測する。一方で、従来の新規事業開発では、競合他社にアイデアを知られないよう、また、市場機会を見透かされないよう、機密を保持してじっくりと計画を練り、準備万端のもとで試作品のテストを行うことがある。し

かし、それでは時間もかかるうえ、顧客の反応の確かめないうちの開発となり、顧客のニーズとかけ離れてしまう。周到に準備して発表するより、頻繁に顧客のフィードバックを貰う方が有益であり、好ましい結果に繋がる（Blank, 2013）。その際、MVP（Minimum Viable Product）と呼ばれる仮説の検証に必要となる最小限のプロダクトやサービスを素早く開発し、それをもとにユーザーの反応を確かめながら、アイデアをブラッシュアップしていく。

リーン・スタートアップの考え方は、潜在的なリスクを予見し、開発プロセス全体から無駄をなくすことである。サンクコストの大きさから方向転換や撤退を先延ばしにしてしまわないよう（川口・下村，2018）、傷が浅いうちに事業を転換（ピボット）することで、大きな失敗を犯して事業継続自体が不可能になるというリスクの最小化を図る（平田・田隈，2014）。そのため、顧客の反応や事業の拡大の可能性から、製品やサービスに対して大きな方向転換や撤退を判断する場合がある（川口・下村，



出所：Ries（2011）をもとに筆者作成。

図 2 リーン・スタートアップの構築 - 測定 - 学習のフィードバックループ

2018)。

こうした検証のために作られる MVP は、ムダを無くすために必要以上に作り込む必要はなく、最低限の機能でなければならない。MVP は、プログラミングにより稼働確認ができるようなものが必要な場合もあれば、実際に現場に出て構想についてユーザーの反応をみるような場合もある (Blank, 2013)。ユーザーからのフィードバックを得て、アイデア仮説の検証を行うことが目的であり、完璧な製品やサービスは必要ない (長谷川, 2019)。

特に最初のアイデアは、粗削りで発想も良くなく、多くは間違っているため、いかにこのフィードバックループを早くたくさん回し、仮説を潰していくかが重要となる (河合, 2013)。顧客が製品やサービスを使うようになったとき本当に価値を提供できるか、そして顧客がその製品やサービスをどう捉えるかが重要な仮説となる (Ries, 2011)。

4.2 顧客価値

リーン・スタートアップは、顧客との対話と MVP を通じて繰り返し仮説を検証することで、顧客課題とサービスを発見、修正していく (田村・和田, 2015)。そのためには、顧客と開発チームが共同で密にコミュニケーションを取りながら MVP をつくりあげていくような、合意

形成型の実証実験 (Proof of Concept) の仕組みが不可欠となる (平原・小林, 2015)。

一方で、MVP に対する顧客のフィードバックを重視しすぎること、顧客の要望をかなえるための商品の段階的、漸進的な改良が目的になりやすく、画期的でイノベーションに溢れた製品の開発から遠ざかってしまう可能性がある (Felin, Gambardella, Stern, and Zenger, 2020)。Apple では、顧客の要望に応えるのではなく、生み出したい将来について共通のビジョンを共有する主要な従業員の判断に頼っているという (Felin, Gambardella, Stern, and Zenger, 2020)。

特にスタートアップが生み出そうとするものは、今までにないような革新的な製品やサービスである。そのため、誰が顧客なのか、そしてその顧客がどこに価値の基準をおくのかが不明であり、何をもって品質の評価をすべきなのかもわからない (Ries, 2011)。潜在的な顧客を探し出さなければならないが、そこで対象となるのがアーリー・アダプターである。実験対象であるアーリー・アダプターと製品やサービスのターゲット市場は異なるが、アーリー・アダプターは強い顧客課題を重視し、早い段階で購入し、評価してくれる (田村・和田, 2015)。しかしながら、質の悪いプロトタイプではアーリー・アダプターには受け入れられず、悪印象

を与えてしまう可能性もある。MVP を通じて製品を日々改良するにしても、それが顧客にとって意味のある改良でなければならず (Ries, 2011), そのため、リーン・スタートアップは、何が顧客にとっての価値を生み出すのかということを重視し、それ以外のものはすべて無駄と考えるのである。

4.3 学習

リーン・スタートアップは、開発成果物よりも顧客からの「学習」を重視する。開発と製造の進捗やその品質ではなく、市場や顧客について有効に学んだかどうかを評価する (田村・和田, 2015)。

MVP を通じた仮説の検証において、期待通りの成果が出ないことで失敗とみなしてしまったり、既存の製品・サービスへの理解を過去の知識に頼ってしまうことで革新的なサービスに向けた大胆な仮説を避けてしまうことがあるが、そうした方法では学習する機会を失ってしまう (田村・和田, 2015)。計画通りに開発を進めることで重役に評価されるのではなく、作り上げたものを顧客に評価してもらわなければならない (Ries, 2011)。成功のための仮説検証を行うのではなく、リスクの高いものを優先し、それが顧客によって反証されることが重要となる。

リーン・スタートアップのフィードバックサイクルでは、アイデアの検証から学び、リスクを軽減することが目的である。何を作るかではなく、何を作らないかという視点のもと、失敗を前提としたリスクが高い仮説を構築することが重要であり、検証による学びを通して、画期的な新製品を開発していくのである。

4.4 組織

こうしたリーン・スタートアップを可能とするのは、自身でイノベーションを起こし、変化変革を起こしていく機能横断的な組織である。

リーン・スタートアップが対象とするスター

トアップは、不確実な状態で、かつ新しい製品や事業を生み出そうとするあらゆる組織を指す。Ries (2011) は、このスタートアップについて、組織として先行き不透明という状況に即した新たな経営方法が必要であることを指摘しており、フィードバックループを順調に回すことができるような社内の仕組みの整備、調整の必要性を述べている。しかし、大企業の新規事業の立ち上げのような場合、既存のプロセスや考え方が障壁となる可能性がある。

たとえば、韓国の Samsung では、新規事業にリーン・スタートアップを導入したものの、既存の組織文化や慣習に縛られてしまう問題に直面したという。新規事業に参画したメンバーが既存組織からスピノフができておらず元の組織と並行で作業をしたり、既存組織から低いパフォーマンス評価を受けてしまうなどの問題が発生し、リーン・スタートアップを円滑に進めることができず、画期的な製品を生み出すための障害になったという (Hwang and Shin, 2019)。

リーン・スタートアップでは、機能ごとに部門として成績を上げるのではなく、スモールチームによる企画・設計者・開発者などの役割を完全に分けない、機能横断的な役割を志向する (田村・和田, 2015)。Anthony (2012) によると、多くの成功したベンチャー企業は、その業界で長く経験を積み、実績のある年配の人材を関与させているという。企業は社員が起業家精神のもと実験を行える環境を整備する必要があり、リーダーは、チームが新しい文化やテストが可能なシステムを用意することで、イノベーションが進められるようにしなければならないのである (Ries, 2011)。

5. リーン・スタートアップとアジャイルの比較

本章では、アジャイルとリーン・スタートアップの特徴について、反復プロセス、顧客価値、学習、組織から比較、検討する。

表 3 リーン・スタートアップとアジャイル

	リーン・スタートアップ	アジャイル
対象	新規事業（ビジネスモデル全体）	ソフトウェア（製品／サービス）
反復（サイクル）の目的	何を作るか？（なぜ作るか？） 無駄の排除。MVP により、アイデアを作り込む前に検証，失敗し，大きなリスクを排除する。	どのように作るか？ 仕様変更への対応。少しずつ実装，試してもらうことで変化する顧客の要望に合わせる。小さなミスを重ねる。
顧客価値	顧客がいない or アーリーアダプター。 作るものが顧客の望むものかわからない（強い顧客課題を重視）	顧客が望むものはある程度はつきりしている（ゴールを共有・共創関係）
失敗による学習効果	仮説の見直し，方針転換（ピボット）。チームとしての学習。	失敗のふりかえりによる自己の成長，さらにチームとしての成長。
組織	仕事をして検証による学びを得る機能横断的なチーム。 自身がイノベーションを起こし，変化変革を起こしていく組織。	能動的に動ける自己組織化と必要なスキルを持つ機能横断的なチーム。 一人ひとりが成長し，全員で協力していく組織。

出所：Ries（2011），Blank（2013），平井（2020）をもとに筆者作成。

リーン・スタートアップとアジャイルを比較したものが表 3 である。

5.1 反復プロセス

リーン・スタートアップもアジャイルも，仮説検証型の反復プロセスとして，漸進的でプロトタイプ的な部分を重視している

リーン・スタートアップはフィードバックループの反復型プロセスにおいて，新規事業の無駄をなくすため，リスクの取捨選択のために実験を繰り返す。反復対象となるものは，試作品（ベータ版）としてのものであり，製品やサービスの利用を実際に開始するわけではない。潜在的なリスクを予見し，開発プロセス全体から無駄をなくすことであり，仮説の検証や事業の方針転換（ピボット）を行うかどうかの判断基準である。

一方で，アジャイルの反復型プロセスは，顧客の要望を少しずつ実装し，実際に製品やサービスを利用してもらうことで，早期にフィードバックを貰い，改善することを目的とする。あ

る程度動くソフトウェアを成長させながら作成する，反復，且つ漸進型であり，少しでも早くできあがったソフトウェアを顧客に確認してもらうのである。

既存の手法では，成果物を重視したり，計画通りに進んだかどうかの定量的なプロジェクトの成功や進捗が重視されてきた。プロダクトやサービスの開発において，アジャイル，リーン・スタートアップに共通していることは，こうした従来型のアプローチが犯してしまいがちな顧客が求める商品とは違うものを作ってしまうことを防ごうとするものであり，検証による学習の量や質を重視しており，実際の製品やサービスを顧客に提示，評価してもらう現物主義と顧客志向である（菊池，2013）。

5.2 顧客価値

リーン・スタートアップもアジャイルも，顧客からのフィードバックを強く重要視しているが，顧客の課題を発見しようとするのに対し，顧客の要望や変化へ対応しようとするとい

う違いがある。

アジャイルは、フィードバックといった顧客の参画の度合いが強く、ゴールを共有しており、そのため、人と人とのコミュニケーションやコラボレーション、共創を重視している。さらに、顧客からの要求一覧（バックログ）に対し、優先事項をつけて開発を行い、顧客にとって価値があるものを優先して開発していく。顧客からのフィードバックや市場の変化に応えるような開発中の設計の変更は、成功に結び付くようなものであれば、必ずしも悪いものではない。製品開発当初の計画に従うよりもユーザーの要望やビジネスの変化への対応を価値としているのである。

一方、リーン・スタートアップは、顧客自体が最初は存在せず、アーリー・アダプターを対象に顧客課題を探っていかなければならない。しかし、こうしたアーリー・アダプターは、必ずしもマーケットの主要顧客であるとは限らず、また、事業をはじめたときについてくれる顧客とも限らない。リーン・スタートアップにも、アジャイルのような開発項目の一覧は存在するが、優先事項は最もリスクが高く、さらにその検証によって学習ができるものであり、仮説の見直しや方針転換（ピボット）するかどうかを重要視する（田村・和田，2015）。

リーン・スタートアップは、何が顧客にとっての価値を生み出すのか、顧客課題とサービスを発見しようとしていくものであり、顧客が実際に利用するための製品やサービスのリリースに価値を置くアジャイルとは異なる。変化していく顧客のニーズに製品を対応させるのがアジャイルであり、リーン・スタートアップは事業の意思決定、リスクの取捨選択を行うのである。

5.3 学 習

リーン・スタートアップもアジャイルも、失敗から学習していくことを重要視しているが、学習の結果、アイデア仮説の見直しや事業その

ものの方針転換を行うのか、それとも顧客要望の変更に対応し改善していくのかという違いがある。

リーン・スタートアップは、新規事業を開発する際、事前に計画を立てることが難しいため、何を作らないかという視点のもと、アイデアの検証から学ぶことで、リスクを軽減することを目的とする。開発したものが顧客が望むものなのかが不明なことから、作ったものが受け入れられるかどうか、失敗を重ねることで学習していくことにフォーカスしている。仮説の見直しや方針転換（ピボット）できるように市場や顧客について有効に学んだかどうか重要であり、こうした検証による学びを通して、画期的な新製品を開発していく。

一方で、アジャイルでは、課題の整理や次の反復に向けた計画を立てる「ふりかえり」を実施し、カイゼンし続けることを目的とする。変化に迅速に適応できるよう、小さな失敗を繰り返す、多くを学んでいくのであり、「反復」を通じてフィードバックをもらい、カイゼンしていく。新しいことを体験し、学び、それを繰り返す努力をしながらメンバーが成長していくのである。

リーン・スタートアップは、市場や顧客について学ぶことで、自ら変化を起こし、世界に変革を起こそうとするものであり（田村・和田，2015）、アジャイルは、学びからメンバーが成長することで、最終的に高いパフォーマンスや難易度を達成しようとするのである。

5.4 組 織

アジャイルでは、学習により一人ひとりが成長することを重要視しているが、状況に応じてミッションを実現するための選択を自分たちが決定し、行動できる能動的な自己組織化が求められる。また、外部に頼らず開発のために必要なあらゆるスキルを持つような機能横断的なチームに成長していくことが必要である。こうした取り組みはソフトウェア分野に限定され

ず、自動車などの他のプロダクトやマーケティングや人事などにも適用されはじめており、組織構造そのものをアジャイルに適應させようとする企業も現れ始めている。

リーン・スタートアップでは、スペシャリスト一人ひとりの能力向上は目的に含まれず、機能横断的に仕事をして検証による学びを得るチームを重視している (Ries, 2011)。リーン・スタートアップは、起業家が事業を拡大できるようなビジネスモデルを探すことをサポートする方法論であるが、この方法を採用し、フィードバックループに従うからといって、起業家の精神が強化されるわけではなく、企業の経営特性と文化から導き出されたルールと原則によってサポートされる必要がある (Hwang and Shin, 2019)。

リーン・スタートアップもアジャイルもこのような機能横断的、すなわちクロスファンクショナルチームのような特徴を有しているが、新しい製品やサービスの開発や、そうした新規事業への参画は、現業と兼務であったり、関連業務をアウトソースしてしまう場合があり、機能横断的な対応を阻害してしまう。また、仮説検証による失敗から学習するという反復型のプロセスが機能しない企業では、失敗を許容しない組織文化があると考えられ (岩崎, 2016)、大胆な仮説や変更を避けてしまうことに繋がる。開発と企画が分断されてしまうなど、部門や組織、立場が異なるメンバーによるひとつのチームとして対応ができなければ、機能横断的な対応はできない。

こうした問題に対して、たとえば、リクルートでは、リーン・スタートアップの専任組織を社内を作り、企画と開発を一体化させ、新サービスの開発や軌道修正を早く進めることができる体制を構築している (高槻, 2012)。リーン・スタートアップもアジャイルも、チームが行動できる環境を整える必要があり、それを行うのは組織の上層部の仕事でもある。環境の変化や実験結果をもとに社内のケイパビリティを構築

し (Collis, 2016)、反復型プロセスにより、計画を練り直すような選択を重ねていくことで組織は学習し、成長していくのである。

6. おわりに

本研究では、顧客を中心とした反復型のプロセスであるリーン・スタートアップとアジャイルを比較、整理してきた。「仮説→検証→学習」の反復型のプロセスは、変化が激しく不確実性の高い現代では顧客にとってより良いものを作る方法でもあり、顧客のニーズを間違えたプロダクトやサービスを開発してしまうといったリスクを最小限に抑える。リーン・スタートアップは、世の中を変えるような大きな変革を起こそうとする新規事業のリスクをいかに減らすかという、失敗から学ぶことによる事業の意思決定、リスクの取捨選択のための反復型プロセスである。一方で、アジャイルは、顧客の要望に対し、フィードバックをもとに改善していくとともに、変化に迅速に適應できるように学習、成長していこうとする反復型プロセスである。

こうしたプロセスを可能とするのは、機能横断的に部門や組織の垣根を超えたチームであり、発見と解決のサイクルの繰り返しを通じた成功や失敗から学び、改善していく組織である。顧客価値の実現や創造には、失敗がつきものであり、そうした取り組みを失敗として終わらせず、顧客や市場ニーズに沿うように継続的に修正していく組織的な仕組みが必要なのである。大企業やスタートアップに限らず、そうした失敗を許容できるような文化や理解がある企業こそが、顧客志向のビジネスモデルを可能とするのである。

参考文献

- Anthony, Scott D. (2012) "The Dangers of the Minimal Viable Product," *Harvard business review*, Harvard Business Publishing.
- Blank, Steven G. (2003) *The four steps to the epiphany: successful strategies for products that win*, Lulu Enterprises Incorporated. (渡邊 哲・

- 堤 孝志訳 (2009) 『アントレプレナーの教科書』 翔泳社)
- Blank, Steven G. (2013) "Why the Lean Start-Up Changes Everything," *Harvard Business Review*, May 2013, Harvard Business Publishing. (有賀裕子訳 (2013) 「リーン・スタートアップ：大企業での活かし方 GE も活用する事業開発の新たな手法」『Harvard business review』 Vol.38, No.8, pp.40-51, ダイヤモンド社)
- Brown, Tim. and Katz, Barry. (2011) "Change by Design," *Journal of Product Innovation Management*, Vol.28, Issue.3, pp.381-383, Wiley.
- Collis, David. (2016) "Lean Strategy," *Harvard Business Review*, March 2016 issue, pp.62-68, Harvard Business School Publishing. (高橋由香理訳 (2016) 「機敏で優れたイノベーションを生むために リーンスタートアップに戦略を融合させる」『Harvard Business Review』 Vol.41, No.8, pp.68-78, ダイヤモンド社)
- Cusumano, Michael A. (1991) *Japan's software factories: a challenge to U. S. management*, New York; Tokyo: Oxford University Press. (富沢宏之・藤井留美訳 (1993) 『日本のソフトウェア戦略：アメリカ式経営への挑戦』三田出版会)
- Cusumano, Michael A. (2004) *The business of software: what every manager, programmer and entrepreneur must know to thrive and survive in good times and bad*, Simon and Schuster. (サイコムインターナショナル訳 (2004) 『ソフトウェア企業の競争戦略』ダイヤモンド社)
- Felin, Teppo., Gambardella, Alfonso., Stern, Scott., and Zenger, Todd. (2020) "Lean startup and the business model: Experimentation revisited," *Long Range Planning*, Vol.53, Issue 4, International Journal of Strategic Management.
- Gage, Deborah. (2012) "The Venture Capital Secret: 3 Out of 4 Start-Ups Fail," *Wall Street Journal*, Sept. 20, 2012, Dow Jones & Company, Inc.
- Hwang, Seonho. and Shin, Juneseuk. (2019) "Using Lean Startup to Power Organizational Transformation: Creating an internal division that implemented concepts from Lean Startup helped a consumer electronics firm foster an entrepreneurial mindset among employees," *Research Technology Management*, Vol.62, Issue 5, pp.40-49, Industrial Research Institute.
- Kuula, Seppo., Haapasalo, Harri, and Tolonen, Arto. (2018) "Cost-efficient co-creation of knowledge intensive business services," *Service Business*, Vol.12, pp.779-808, Springer Nature.
- Norman, Donald A. and Draper, Stephen W. (1986) *User Centered System Design: New Perspectives on Human-computer Interaction*, Taylor & Francis.
- Ries, Eric. (2011) *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, Currency. (井口耕二訳 (2012) 『リーン・スタートアップ』日経BP)
- Rigby, Darrell K., Sutherland, Jeff., and Takeuchi, Hirotaka. (2016) "Embracing Agile," *Harvard Business Review*, May 2016 Issue, Harvard Business School Publishing. (倉田幸信訳 (2016) 「アジャイル開発を経営に活かす6つの原則 臨機応変のマネジメントで生産性を劇的に高める」『ハーバード・ビジネス・レビュー』2016年9月号, ダイヤモンド社)
- Womack, James P. and Jones, Daniel T. (2003) *Lean thinking: banish waste and create wealth in your corporation*, Free Press. (稲垣公夫訳 (2008) 『リーン・シンキング』日経BP社)
- 市谷聡啓 (2019) 「正しいものを正しくつくる—プロダクトをつくるとはどういうことなのか、あるいはアジャイルのその先について—」株式会社BNN 新社.
- 岩崎健一郎 (2016) 「IoT 時代におけるリーンスタートアップ: UnlimitedHand の事例から」『SEC journal』 Vol.11, No.4, pp.18-21, 情報処理推進機構技術本部ソフトウェア・エンジニアリング・センター.
- 河合太郎 (2013) 「未知のニーズを探り当てる超上流開発 (第3回) ヤフーがリーンスタートアップ仮説検証を繰り返しアプリ作成」『日経コンピュータ』 No.840, pp.104-107, 日経BP社.
- 川口真一・下村剛士 (2018) 「プラットフォーム型ビジネスを支えるシステム運営」『Unisys technology review』 Vol.38, No.1, pp.37-47, 日本ユニシス.
- 菊池隆裕 (2013) 「広がるリーンスタートアップ 大企業の新商品開発に威力」『日経コンピュータ』 No.838, pp.62-67, 日経BP社.
- 妹尾 大 (2001) 「ソフトウェア開発の新潮流—状況論的リーダーシップの胎動」『組織科学』 Vol.35, pp.65-80, 組織学会.
- 高槻 芳 (2012) 「IT 部門を強くするリーンスタートアップ」『日経コンピュータ』 (812), pp.56-65, 日経BP社.
- 田村昌也・和田 洋 (2015) 「新規事業のイノベー

- ションを加速するリーン・スタートアップ適用とアジャイル開発: ビジネス開発とソフトウェア開発両面からのアプローチ実践のポイント」『Provision』No.87, pp.20-25, 日本アイビーエム, 西村直人・永瀬美穂・吉羽龍太郎 (2013) 『SCRUM BOOT CAMP THE BOOK』翔泳社.
- 長谷川克也 (2019) 『スタートアップ入門』東京大学出版会.
- 平井直樹 (2018) 「ソフトウェア開発プロセスにおける分業構造と知識労働—日本の受託ソフトウェア開発の組織問題—」博士論文, 立教大学大学院ビジネスデザイン研究科.
- 平井直樹 (2019) 「アジャイルの導入と本質—開発プロセスから学習する組織活動へ—」『立教 DBA ジャーナル』No.10, pp.35-44, 立教大学大学院ビジネスデザイン研究科.
- 平井直樹 (2020) 「ソフトウェア開発におけるアジャイルに向けた共創の課題 —顧客の構造問題の観点から—」『現代ビジネス研究所紀要』2019 年度, 昭和女子大学.
- 平田 隆・田隈広紀 (2014) 「P2M に基づく価値蓄積型製品開発手法の提案」『国際 P2M 学会誌』Vol.9, No.1, pp.83-94, 一般社団法人国際 P2M 学会.
- 平原誠也・小林賢治 (2015) 「eXtreme Design Studio を拠点とした顧客との共創による新ビジネスの開発手法」『Toshiba review』Vol.70, No.10, pp.52-55, 東芝研究開発統括部.
- 藤本隆弘・朴 英元 (2015) 『ケースで解明 IT を活かすものづくり』日本経済新聞出版社.
- 松丘啓司 (2018) 「アジャイルな人事変革の必要性—ビジネスのイノベーションを加速するために—」『経営センサー』9月号, 2018, No.205, 株式会社東レ経営研究所.
- 村田賢太 (2013) 「クックパッドのサービス開発を支える Ruby の技術」『赤門マネジメント・レビュー』Vol.12, No.12, pp.841-850, 特定非営利活動法人グローバルビジネスリサーチセンター.
- 渡壁 祐・野城智也 (2019) 「Ideation におけるプロトタイピングに関する一考察」『年次学術大会講演要旨集』No.34, pp.542-545, 研究・イノベーション学会.
- 渡辺 隆・ブラウンザック・小俣剛貴 (2019) 「Pivotal 流イノベーションを起こす開発チームの作り方」『日経 systems』No.319, pp.36-43, 日経 BP 社.

インターネット資料

- James, Michael. and Walter, Luke. (2010), "Scrum Reference Card", (http://scrumreferencecard.com/ScrumReferenceCard_v1_3_1-jp.pdf).