

## Scrum における顧客要求再考イベント導入の提案

### Proposal for introducing event of rethink the customer requirements at Scrum

研究員 : 伊藤 善博 (株式会社デンソークリエイト),  
楠本 至央 (ブラザー工業株式会社),  
荒川 健太郎 (ウイングアーク 1 s t 株式会社)  
主査 : 永田 敦 (サイボウズ株式会社)  
副主査 : 山口 鉄平 (freee 株式会社),  
荻野 恒太郎 (楽天株式会社)  
アドバイザー : 細谷 泰夫 (三菱電機株式会社)

#### 研究概要

顧客からのフィードバックを基に迅速に顧客要求に応えることができるはずのアジャイル開発でリリースされたプロダクトが顧客の満足を得られない、すなわち売上につながらないということが多く見受けられる。本研究ではその原因を「開発チームが顧客の要求を重視し、顧客の価値の軽視することにある」と仮定し、開発チームがスプリント毎に顧客の「価値」を再度検討するイベントを Scrum に導入することにより、顧客が本当の意味で満足するプロダクトをリリースする事ができるのではないかという仮説のもと、そのイベントについての検証を行った。我々はその仮説検証方法とその結果および実験サンプルは少ないまでもある程度の有効性があるという判断に至ったことについて報告する。

**Abstract** Many products released in agile development that should be able to respond quickly to customer requests based on customer feedback do not satisfy customers, that is, do not lead to sales. In this study, we assume that the cause is that the development team attaches great importance to customer requirements and neglects the value of the customer. We conducted a hypothetical verification that the introduction would allow the customer to release a truly satisfying product. We report the hypothesis test results of each researcher who participated in this study.

#### 1. はじめに

##### 1.1 背景

アジャイル開発のメリットの一つとして、開発途中でも顧客からのフィードバックを受けて対応することにより顧客の要求に最大限応えることができるという点は広く知られている。

しかしそのアジャイルのプロセスを適用して開発したプロダクトも、顧客の要求に応えるものばかりがリリースされているとは言い難い。顧客の要求を獲得できないまま方向性が定まらずに延々と開発期間だけを消化しているプロダクトや、顧客の満足を得られることなく開発終了に至るプロダクトが多いのが実情である。

例えば、顧客が運びやすいスーツケースが欲しいと述べ、それを軽量にして欲しいと頼んだとしよう。その通りに実行しても、顧客は我々の洗練されたデザインを却下し、競合する会社のもっと重いものを購入するだろう。車輪付きのものを。<sup>[1]</sup>

上記の引用文にもある通り、顧客も自身の真の要求を理解していないために初回の要求や開発中のフィードバックにそのまま応えたとしても、顧客が満足するプロダクトになる

とは限らない。アジャイル開発のフレームワークのひとつであるスクラムにはスプリントレビューやバックログリファインメントという顧客要求を整理する機会となるイベントが用意されているが、そのイベントで我々が真の要求を獲得できているかと言えば答えに窮するのが現状である。

この真の要求（ユーザーが真に考えている・欲しているモノ）に、常に近い状態にすることで円滑にスプリントが進み、ユーザー・開発双方が Win & Win な状態に近づけると我々は考える。

## 1.2 本研究の狙い

我々は 1.1 のような状況に陥る原因として下記 2 点を仮定した。

- (1) 顧客の要求を獲得する際に要求ばかりに耳を傾け、顧客がどのような状態に価値を感じるかを無視及び軽視した「要求≠価値」の状態の開発を開始してしまう。
- (2) 仮に開発序盤に「要求≒価値」を獲得出来た際も、反復的に開発活動を行うに連れ、顧客の外的及び内的要因の変化について検討せず、いつの間にか「要求≠価値」となってしまう。

これらを解消するために、我々はスクラムにおけるスプリント毎に顧客の「価値」を再検討するイベントを設けることを提案する。このイベントはスクラムチームが徹底的に顧客の目線となり「顧客の価値」を検討する場として、User Value Design Thinking (UVDT) と名付ける。

スクラムにはバックログリファインメントというイベントが存在するが、ビジネスや開発（設計）視点で「何をどのような順番で作っていくか」を議論するイベント側面が強く、「顧客が本当に幸せになるのは何か」を検討する UVDT とは棲み分ける必要があるという結論に至った。

以降の 2 章では本研究における具体的な問題と研究内容について述べる。3 章では研究内容を適用した結果を述べる。4 章では研究結果の考察と今後の課題について述べる。

## 2. 研究内容

### 2.1 プロダクトが顧客満足を得られない問題

「要求≒価値」が日数（開発）を経ることにより「要求≠価値」となる要因には、次のことが考えられる。

- (1) 外的要因の変化（世界情勢や市場）により、顧客が満足するプロダクト品質が変化する。
- (2) 顧客自身の変化（思考や嗜好）により、顧客が満足するプロダクト品質が変化する。
- (3) 開発当初は顧客も自身が満足するプロダクト品質を分かっていなかった。
- (4) 開発側が過剰に作りすぎて、顧客要求とは異なった機能となってしまう。

顧客の考える要求は、時間が経つことで変化していくことは世の状況に応じては常であり、アジャイル（スクラム）の導入の必要性が様々な業界において叫ばれている。

しかしながら、アジャイル（スクラム）においては、スプリントレビューなど管理的なイベントは規定しているが、要求を開発するようなエンジニアリングプロセスは規定していない。アジャイルでは変わりうる要求と過密なスプリントの中、ユーザーが望むものをタイムリーに届ける必要がある。そのため、開発者であっても手早く利用でき手軽に重要な要求を見直していくことができるプロセスやツールが必要である。

### 2.2 要求とプロダクトのズレを解消する方法の検討

アジャイル、特にスクラムでは要求の獲得や優先度の見直し（プロダクトバックログを保つという意味合い）は、下記イベントで行われる。

表 1. スクラムにおけるプロダクトバックログの更新イベントと問題点

| イベント名         | 説明   | アウトプット                          | 問題点   |
|---------------|--|---------------------------------|---|
| スプリントレビュー     | スプリントの成果をレビューする場。プログラムを動作させ、目的にかなうものかを確認しフィードバックを得る。         | ステークホルダからのフィードバック事項をバックログに追加する。 | 要求全体を見渡した上ではないため、時には場当たりのフィードバックとなり、本当に今重要であり優先であるかの議論がなされにくい。                  |
| バックログリファインメント | スプリントとは非同期で、仕様・設計やリスクを考慮した議論によって、プロダクトバックログをより明確に(refine)する。 | 開発が Ready 状態となったプロダクトバックログ。     | 要求ありきの議論となり、開発チームのリソース状況などの都合が多分に加味されてしまい、ユーザーにとってそれが今必要なものなのかは説明も議論もされない傾向にある。 |

上記問題に陥らないようにするため、User Value Design Thinking(UVDT)というイベントを提案する。

## 2.3 UVDT (User Value Design Thinking)

### 2.3.1 UVDT の定義

UVDT とは要求全体を捉えなおし、今まさに顧客にとって何が最善・優先であるのかを見つめなおすイベントと定義する。これによりチームで顧客の要求変化を検知する確率を高めることができる。

UVDT はバックログリファインメントの前に実施する。これは顧客目線で要求を整理するため、開発者目線に落ちがちであるバックログリファインメントの前に実施しておく方が効果的なためである。

UVDT の結果はバックログリファインメントでプロダクトバックログに反映する。UVDT の概念図を図 1 に示す。

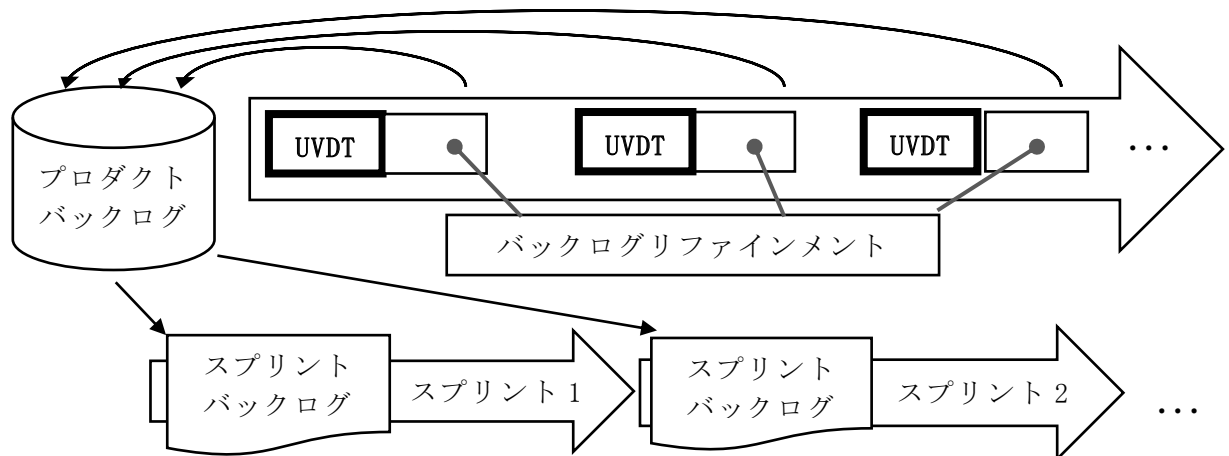


図 1. UVDT を用いたスクラムの概念図

### 2.3.2 UVDT の要素の選定

UVDT の要素として、顧客価値の検討に下記の考えを選定した。

- ① 増やしたい要素(ゲイン)、減らしたい要素(ペイン)の明確化

② ペイン、ゲインを解消した場合の顧客の状態や行動（アウトカム）を検討  
これらの要素を選定した理由は、顧客の価値を検討するのに使われている表 2 のツールから抽出した。

表 2. UVDT の要素の選定

| ツール名                            | 要素の選定箇所   |
|---------------------------------|---|
| ペルソナ <sup>[4]</sup>             | 想定する顧客（ユーザー）像を明らかにする。   |
| 顧客マップ <sup>[10]</sup>           | 顧客の背景，行動のモチベーション=ゲインやストレス=ペインを整理。                                       |
| ユーザーストーリーマッピング <sup>[6]</sup>   | 顧客にとって優先したい価値と排除したい価値を整理。   |
| カスタマージャーニーマップ <sup>[4][5]</sup> | 顧客の行動を分析し，製品やサービス利用時における感情や問題点を抽出．問題点解消の先に，顧客がどのような姿となっているべきか=アウトカムを整理。 |
| 狩野モデル <sup>[7]</sup>            | 顧客の求める品質をモデル化して，顧客満足度と優先したい価値と排除したい価値を整理。                               |

加えて，アジャイルで実行するにあたって，下記の考えも取り入れた。

- ③ 手軽に短時間で実施できる（アジリティ）。開発と企画や UX デザイナが一体<sup>[8]</sup>となり，スクラムチーム全員が参加して顧客の価値を検討する。そのため，手間がかかり長時間必要となると現場では活用されないことになってしまう。
- ④ 関係者全員で参加する。スリーアミーゴス<sup>[9]</sup>に代表されるような多角視点の導入を狙う。これにより，開発視点のみでは顧客価値が無視されがち，企画視点のみでは実現性がなくなりがちな課題を解消する。

### 2.3.3 UVDT の作成

表 2 のツールと選定における考え方を整理し，UVDT の仕組みとして下記のように定義する。

#### 2.3.3.1 UVDT のプロセス

UVDT では，下記 A～D をバックログリファインメントの前に実施する。各プロセスの具体的な成果物は施策の実施で解説する。

表 3. UVDT の実施プロセスと説明

| ID | プロセス名             | 説明  |
|----|-------------------|---|
| A  | ペルソナを確認する         | 想定するユーザー像の明確化し，全員がユーザーを把握できるようにする。  |
| B  | 顧客マップを定義・アップデートする | ユーザーのコンテキストをベースに，製品・サービスが低減すべきペイン，成長させるべきゲインを明確化する。スプリント毎に参照し，アップデートが必要な場合のみ実施する。             |
| C  | アウトカムを深堀する        | ペイン・ゲインを解消した先に，ユーザーがどうなっているべきなのか，その姿を明確にする。なぜユーザーがそうするのか，そう考えるのかを深堀して考える。ここで抽出した事項を要求として整理する。 |
| D  | 顧客価値と要求度で整理する     | C で抽出した要求を，顧客価値の大きさと要   |

|  |  |   |
|--|--|---|
|  |  | 求度（どれだけ欲しいか）の大きさを整理する．整理する手法を User Value Map と呼ぶ．詳細は後述する． |
|--|--|---|

### 2.3.3.2 User Value Map (UVM)

バックログの価値による優先順位を考えるため，縦軸を価値，横軸を要求度として図 2 のようなマップを考えた．これを User Value Map (UVM) と呼ぶ．UVM は狩野モデルをベースに，当たり前品質，魅力品質をより考えやすくするために定義したマップである．

UVM では各機能が価値や要求度の相対的な位置関係で示すため，どの要求に価値があるのかわかりやすい．加えて，重要で優先すべきであるという説明にも利用できる．

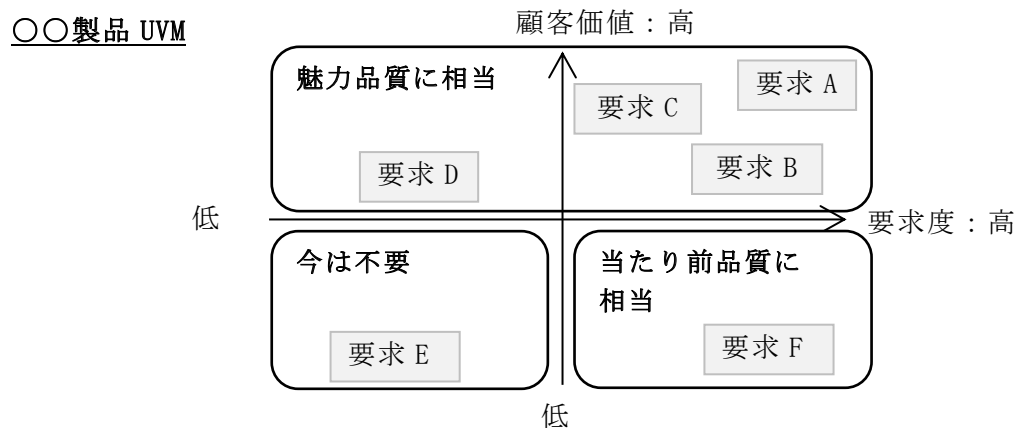


図 2. UVM

### 2.3.3.3 UVDT を短時間で実施するための仕組み (UVDT カードの利用)

ペインやゲイン，アウトカムの議論については，数回のプロセス実施の中で円滑に進めるのが難しく時間がかかることがわかった．手軽に短時間での実施を狙うため，考え方の観点・きっかけとなるツールとして，UVDT カードを用意した．これを用いることで，チームでの議論の円滑化，意識統一できるようにする（付録参照）．議論の際には，UVDT カードを見ながら考慮できていない観点がないか検討し，各自が任意のタイミングでカードを提示し意見を出し合うこととした．

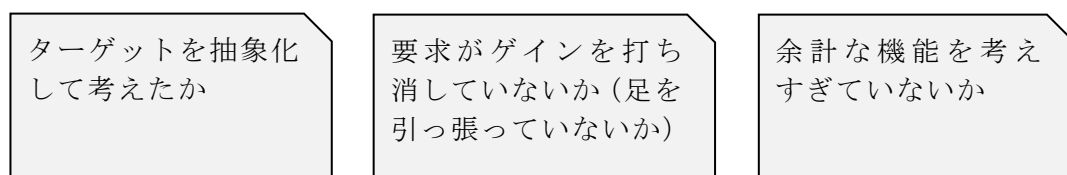


図 3. UVDT カードの一例

## 3. 施策の実験と結果

スクラムの研修などに使用される「LEGO®を使った街づくりスクラム<sup>[2]</sup>」をスクラムチームに実施してもらい，UVDT 適用前と適用後それぞれのスプリントレビュー時の指摘数および指摘内容を比較する．（付録参照）

### 3.1 施策適用対象のチーム

表 4. スクラムチームの構成

|       | 顧客  | プロダクト<br>オーナー | スクラム<br>マスター | デベロッパー |
|-------|-----|---------------|--------------|--------|
| メンバー数 | 1 人 | 1 人           | 1 人          | 2 人    |

### 3.1.1 スクラムの概要

顧客が理想とする街にある建物について要件を決め、スクラムチームはスクラムフレームワークを適用して 3 スプリントないし 2 スプリントで顧客が望む建物を作成する（付録参照）。

表 5. スクラムの流れ

| 参加者 |         | イベントおよびアクティビティ                          |
|-----|---------|---|
| 顧客  | スクラムチーム |   |
| ○   | -       | ペルソナ・ユーストリーを用い、要件を定義                    |
| ○   | ○       | 要件のヒアリング・PBI (Product Backlog Item) を作成 |
| -   | ○       | スプリントプランニング                             |
| -   | ○       | 開発スプリント                                 |
| ○   | ○       | スプリントレビュー                               |
| -   | ○       | 振り返り                                    |
| -   | ○       | バックログリファインメント                           |

### 3.1.2 施策適用前の状況

各スプリントレビューで以下のような指摘数および指摘の傾向が見られた。ここでは UVDI 適用により改善出来ると指摘の内容について抜粋して記述する。

表 6. スプリントレビューでの指摘数と指摘内容

| 指摘数合計 | 主な指摘内容  |
|-------|---|
| 13 件  | <ul style="list-style-type: none"> <li>・ 銭湯にサウナを付けてくれているが別に必要ない。</li> <li>・ 動物病院に駐車場は必要ない。</li> <li>・ カラオケ屋は 2 階建てではなく平屋が良かった。</li> </ul> |

スプリントレビューでの指摘の他に、計画した建物がスプリント内に完成しない（有効なものを期限でリリースできなかった）ということが見受けられた。

この実験を複数回実施したところ、以下のような傾向がみられた。

- ・ 顧客のいうフィードバックを鵜呑みにして実現しようとしている。
- ・ 余計な気をきかせて、言われてもないものを作ろうとする後々になり、こちらのほうがよかったという指摘をうける。
- ・ 本当に必要なのか納得しないまま、開発している。

### 3.2 施策の実施

実験対象のスクラムチームに対して、UVDI の適用実施内容を説明する。

#### ① ペルソナ定義・顧客マップの作成およびアップデート、および顧客マップの定義

- ・ 時間：5 分～10 分
- ・ 内容：ペルソナは、名前や年齢といった基本事項のほか、顧客マップにある要素を満足するよう定義。必要あれば毎回アップデートした。
- ・ 効果：ユーザー像として整理すべき事項がまとまっており、手早く定義できた。

- ・ 利用した UVDT カード：「ターゲットを抽象化して考えたか」、「ゲイン・ペインを横展開したか」

## ② アウトカムの深堀

- ・ 時間：10 分
- ・ 内容：①の結果から、ゲインを向上させる、ペインを低減させる機能の整理と、利用したユーザーの姿・価値を明記した。
- ・ 効果：ユーザーの利用時の姿・感情および価値を考えることで、全員で価値共有することができた。
- ・ 利用した UVDT カード：「余計な機能を考えすぎていないか」、「ペインを低減できているか」「ゲインを引き上げられているか」

## ③ UVM による価値整理

- ・ 時間：10 分
- ・ 内容：②で抽出した機能を、UVM にマッピングした。
- ・ 効果：相対的にユーザーが何を望んでいるのか価値のまとまりや力の置き所がわかり、今本当にユーザーにとって欲しいものが何なのか、その妥当性を皆で議論・納得することができた。
- ・ 利用した UVDT カード：「全体俯瞰せず、顧客の言いなりになっていないか」、「あって当たり前ではないか」、「ゲインを打ち消していないか」

## 3.3 施策適用後の結果

施策を適用した結果、下記のような傾向となり、UVDT が有効に機能していることがわかった。

- ・ 顧客のフィードバックに対して、真の顧客の価値に対する指摘かを検討できた。
- ・ 顧客が不要な機能の作りこみに対する指摘が減少した。
- ・ 顧客の価値から優先度を上げ、開発すべき要求から開発することができた。

### 3.3.1 施策適用前後の数値

施策前と施策後の指摘数を表 7 で比較する。適用後の指摘では、特に不要な機能の作りこみに対する指摘が減少していることが分かった。すべての指摘は付録に記載する。

表 7. 施策前と施策後の指摘数と指摘内容

|     | 指摘数  | 主な指摘内容  |
|-----|------|---|
| 施策前 | 13 件 | <ul style="list-style-type: none"> <li>・ 銭湯にサウナを付けてくれているが別に必要ない。</li> <li>・ 動物病院に駐車場は必要ない。</li> <li>・ カラオケ屋は 2 階建てではなく平屋が良かった。</li> </ul>   |
| 施策後 | 8 件  | <p>「屋根が傾いている」「強度が心配」「ブランクに見えない」など技術的な指摘はあったが、施策前に見られる顧客価値と開発のズレによる指摘はなかった。</p> <p>加えて、下記のような顧客価値につながる意見が目立った。</p> <ul style="list-style-type: none"> <li>・ お願いしていなかったが学校の屋上にプールがあるのは嬉しい。</li> <li>・ 家の裏の畑にある裏口は想定していなかったが作ってくれてよかった。</li> <li>・ 道路に街灯を並べてくれ夜道も安心できる。</li> </ul> |

表内の意見につながった箇所はいずれも UVDT のイベントを経て作成したものである。

### 3.3.2 適用後のチームの変化

適用前に比べて下記のような効果、傾向があった。

- ・顧客の価値に基づいてのリファインメントが行われた。機能や工数などのプロダクト都合の思考による変化しない現象は減少した。
- ・顧客の価値が明確になり、開発者も納得感を持って対応を進めることができた。
- ・スプリントの指摘において、ゲイン、ペインを意識した指摘となり、自分たちの都合ではなく、顧客の価値に基づいた指摘ができていた。

スプリントレビュー時に顧客の価値を再度検討するだけでなく、UVDTにより顧客の価値が明確となっているため、スプリント参加者全員が納得感を持って開発を進めていくことができた。

#### 4. 考察

アジャイル開発において、UVDTを適用することにより、非適用時より要求とプロダクトのズレが小さくなっており、特に下記が有効であることがわかった。

- ・実施する具体的なプロセスとタイミング
- ・短時間で実施できる仕組み（UVDTカード）
- ・顧客価値の大きさが見える化（UVM）

UVDTで顧客価値が大きく、バックログリファインメントで開発スピードの速いものから開発できるため、顧客が望んでいる機能を優先的に開発できる。そのため、プロダクトの価値を最大化できる。

アジャイル開発においては、顧客の価値をプロダクト全体で見つめなおすことが重要であるが、その具体的方法としてUVDTは効果的であるという検証結果を得られた。

#### 5. 今後の課題

今回はLEGO®による仮想的な開発の事例としたが、実際のアジャイル開発のプロセスに適用し、UVDTの有効性を検証していきたい。また、「顧客の価値が最大化されているか」を計測する方法も検討を進めていきたい。

#### 6. 参考文献

- [1] Pressman, R. S. : Object Oriented SELECTION, Dean Leffingwell, 2014. (邦訳) 藤井拓, 株式会社オーグス総研 訳, アジャイルソフトウェア要求, 翔泳社, 2014 .
- [2] LEGO4SCRUM : SCRUM SIMURATION WITH LEGO, <https://www.lego4scrum.com/>.
- [3] 小田ビンチ, 坂元勲, 田村大 (監修), まんがでわかるデザイン思考 (日本語) 単行本, 2017.
- [4] マッピングエクスペリエンス-カスタマージャーニー, サービスブループリント, その他ダイアグラムから価値を創る, James Kalbach, 武舎宏幸 (翻訳), 武舎るみ (翻訳), オライリージャパン, 2018
- [5] UXデザインの教科書, 安藤昌也, 丸善出版, 2016
- [6] ユーザーストーリーマッピング, Jeff Patton(著), 川口 恭伸(監修), 長尾 高弘(翻訳), オライリージャパン, 2015
- [7] 魅力的品質と当り前品質, 品質 14 巻(1984) 2 号, 狩野 紀昭, 瀬楽 信彦, 高橋 文夫, 辻 新一
- [8] LEAN UX, ジェフ・ゴーセルフ (著), ジョシュ・セイデン (著), 坂田 一倫 (監修), エリック・リース (編集), 児島 修 (翻訳), オライリージャパン, 2017
- [9] Agile Glossary, Agile Alliance(<https://www.agilealliance.org/agile101/agile-glossary/>)
- [10] Mobiusloop, Monius, (<https://www.mobiusloop.com/>)