

第 6 回特別講義 レポート

日時	2018 年 12 月 14 日 (金) 10:00 ~ 12:00
会場	(一財)日本科学技術連盟・東高円寺ビル 2 階講堂
テーマ	ユーザーストーリーマッピングを用いたアジャイルな要件定義ワークショップ
講師名・所属	川口 恭伸 氏 (アギレルゴコンサルティング株式会社 アジャイルコーチ)
司会	永田 敦 氏 (サイボウズ会社/本研究会 研究コース 4 主査)
アジェンダ	1.実感駆動 : アジャイル開発をなぜ行うのか? 2.ユーザーストーリーマッピング概要 3.ユーザーストーリーマッピング体験ワークショップ
アブストラクト	アジャイル開発をなぜ行うのか? 従来型のソフトウェア開発プロセスで起こりがちな課題が「リリースしたらクレームが来ても直せない」「作ってみたけど使ってもらえない」です。社内向けにせよ、顧客企業向けにせよ、一般消費者向けにせよ、使われないプロダクトを作っても雇用を確保できません。ソフトウェア開発はだいぶ軽量になってきましたので、ぼちぼちエンドユーザに目を向けて、素早く試せる開発の余地が大きいはずです。 エンドユーザからの情報の吸い上げからリリースまでを一貫して行うための合意形成としてユーザーストーリーマッピングを試してみませんか?
講義の要約	
川口 恭伸 氏 ・株式会社 QUICK にてデータ管理、業務用アプリケーション開発、金融機関向けプロダクト開発、開発インフラ整備などを経て、2012 年 4 月より現職。 ・Innovation Sprint 2011 実行委員長、Scrum Gathering Tokyo 実行委員。	

1. 実感駆動：アジャイル開発をなぜ行うのか？

・アジャイルの認識として、一般的には、イテレーティブな短い期間でモノをリリースし、要件定義も軽量、少人数のチームでモノを作る手法が有名だが、以前は、どうすれば要件定義をして開発できるのか、プロダクトオーナーがどうやって要件定義を実践できるかを指南するものが少なかった。

・会社でソフトウェア開発してみたら

要件定義→開発・テスト→リリース→保守の流れになるが、リリースされてからユーザーが本気になり、使い勝手が悪いなどとクレームが上がることが多い。

⇒改修しろと言われても、要件定義した人も開発した人もいない、要件定義の段階でドキュメントにて合意したはずなのにとのジレンマがある。

⇒ユーザーはドキュメントに対しては“まじめ”に、動くソフトウェアに対しては“本気（マジ）”になる。

⇒リリースによって混乱は必ず起きるもの。ユーザーは新たなシステムに対して、たとえ生産性が良くなったとしても最初は必ず混乱する。

⇒リリース→混乱→クレーム→対策→リリースのサイクルを素早く繰り返すことで、ユーザーの要求は満たされ、変化にすぐに対応してくれる会社にお金を出したいと考える。一方で、クレームを上げてもすぐ対応されないと分かると、ユーザーもメーカーも“まじめ”になり、次に繋がらない“あきらめ”の領域へ遷移してしまう。

⇒“要件”の実態として、まず全体の構想から予算制約と実装不可の判断で取り除かれたものがリリースされる。リリースされても実装した機能の6割は利用されず、残りが真のユーザー要件として利用されるが、非常に無駄が多い。

・どうしたらいいのか？

⇒小さく・なるべく早く。要件定義→開発・テスト→リリースを繰り返し、動くソフトウェアをユーザーが使うことで本気（マジ）になってくる。早い段階で、本当に必要な要件が分かってくる。

⇒野球に例えるなら、塁に出る→送りバント→タイムリー→ホームランの流れで、小さな成功から大きな成功へ。

⇒車に例えるなら、タイヤ→シャーシ→ボディー→車では完成するまでユーザーは本気になれない。

スケートボード→キックボード→自転車→バイク→車のように、ユーザーが最小限でも試せる環境があれば、早い段階から本気になれる。

・アジャイル

⇒2001年にアジャイルソフトウェア開発宣言が出たが何のためのマニフェストか。

⇒アジャイル開発手法の方が明らかに儲かる から。

ユーザーにとって動くソフトウェアがあればこそ、使って初めて分かること・得られる真の要求に気づくことができる。そしてユーザーに協調して対応する組織にこそ対価を払いたいと考える。圧倒的に優位である。

⇒アジャイル開発手法のスクラム(Scrum)に影響を与えたものとして、野中郁次郎氏の「The New New Product Development Game」がある。70~80年代の日本の新製品開発方式を解説しており、工程に分けて設計・製造に入るやり方では無く、全員が一か所に集まって話をし作っていたというものである。

また、リーン生産方式(Lean。MIT)もアジャイルに強く影響を与えている。無駄を減らす開発手法で、このリーンはトヨタ生産方式を一般化、体系化したものである。

・スクラム

スクラムのロジックについては以下を公開している。

10分でスクラム(<https://www.slideshare.net/kawaguti/20110118-scrum-10-mins>)。

・アジャイル失敗談

⇒[アジャイル成功の先のレイオフ] XP導入によって以前より効率的にソフトウェアをリリースできるようになったが、当面のソフトウェア開発/保守で必要でなくなった人をレイオフした結果、現場意欲が落ち、先のデリバリを考えていなかったために倒産した。

⇒[必要とされていなかった製品] CEOはアジャイルとUXに乗り気で、時間を掛けてユーザー調査や製品企画を進めていた。そしてアジャイル開発も非常にうまくいったものの、2年半かけて製品をリリースしたとき、世の中が既に変わっており誰も欲しくなかった。

⇒先が見えずプロダクトバックログを積みなくて価値が出なかった事例も、プロダクトバックログを積み上げて出しても使われなかった事例も、いずれもアジャイルをやって起こり解決できなかった問題である。

ジェフ・パットン氏がユーザーストーリーマッピングを立ち上げるきっかけになり、より軽量でよりスピーディーに市場に投入できる開発手法を目指している。

・バックログはどこからくるのか

⇒手順としては、既存のシステムを使っている人が抱える課題に対して、解決のアイデアを持ち寄る。その解決のアイデアから実機能に細分化し、漏斗に掛けてプロダクトバックログのデリバリの順番に落としてゆく。アイデアだけで使われないものや、作るのに手が掛かるものを弾き、適度なレベルのものをデリバリオーダーに落としてゆく。実際には、この手順をユーザーストーリーマッピング(後述)などの手法を使って行う。

2. ユーザーストーリーマッピング概要

・付箋の使い方

ルール：お互いが何かの専門性や必要な能力を持ったチームメンバーとして認めた上で、例え新人であってもチームに貢献することができる。協調してアイデア出しをするのがアジャイルの文化である。

⇒大きな文字で書く、1単語ないし1フレーズで（英語だと3単語ぐらい。関係者に伝わればOK。風景写真のように記憶を呼び出す鍵にする）、箇条書きにしない、すぐに書く・手分けして書く

⇒カード(付箋)→会話→確認→構築→結果のループを回す。

⇒全員が参加し、発言と同時にアップデートを重ねる。共通理解を築くところから始める。

・ユーザーストーリーマップ

⇒起源は、音楽プロデューサーのシステム構築にジェフ・パットン氏に関った際のエピソードがある。音楽のスタートアップを行うプロデューサーであるゲイリー氏が、曲作りから販売までの一連の流れを支援させるシステムを構築しようと優秀な開発者を多く雇った。ただ音楽に詳しくないため求める機能が多く、優先順位も決めなかった為に開発者は混乱し、多額の費用と工数を掛けてもシステムは完成しなかった。ジェフ・パットン氏の支援を受け、ゲイリー氏が考える対象ユーザーと機能を付箋に書き出し可視化した。ストーリーごとに並べ換え、特に大事なものに絞り取捨選択していくことで開発を進め、使えるシステムが完成した。

⇒横軸にユーザーが行う大まかなアクティビティをバックボーン(背骨)として描き、そのアクティビティのなかで発生するステップを書き出す。縦軸にステップ毎の詳細を書き出していく。

⇒多くのタイプのユーザーを扱うマップの場合は、ユーザータイプ毎のアクティビティから、その詳細動作や機能をステップとして書き出す。また全ての関連するユーザーがどのように動き、そしてどのポイントをソフトウェア(システム)が関与するかの全体像をナラティブフロー(物語の流れ)として明らかにしておく。

⇒左右の軸：時系列は、左が過去、右が未来。

上下の軸：優先度は、上が高い、下が低い。粒度は、上が抽象、下が詳細。

⇒優先度が高いものを確実にリリースしたい場合は、軸の上側にあるアクティビティ(作るもの)を減らし納期に間に合う確率を上げる。ウォーターフォールであれば、納期間際になってから実装できない要求が判明し取捨選択が始まるが、ユーザーストーリーマッピングでは、要求を洗い出すところで実装できるかどうかを検討することができる。

⇒横方向にウォークスルーを行う。ユーザーに参加してもらい、最後にできたマップをウォークスルーでチェックしてもらうことでソフトウェアの組み上げ方が判断できる。

3. ユーザーストーリーマッピング体験ワークショップ

・ワークショップ

- ⇒朝起きてから家を出るまでにやったことを洗い出す（付箋に書き出す）。3分間。
- ⇒3～4人のチームで、なるべくしゃべらずに1つのリストに纏める。
- ⇒似たものを近く、違うものを遠くに配置する。左右の軸：時系列。上下の軸：優先順位。
- ⇒家を出るまでに5分しか無いとした場合にやるものを残す。
マスキングテープを引いて、5分内でやるものを上側、やらないものを下側へ整理する。
- ⇒本当に5分でできるかを再考する。上側のやるべきことをなるべく減らす。
- ⇒何か忘れていたものが無いか再考する。

～質疑応答～

・同一優先度が出て外せない場合はどうすれば良いか？

- ⇒期限までに全ての要求が実現できないという現状を知ることが重要で、それが分かれば対策は練れる。
ROI やリスク分析や作り上の理由など様々な要因はあるが、方策がなければリリースできず、会社としては食えないことは言える。それでも落とすことは何かを全員で考えれば、合意が得られるアイデアがでてくる。

・アジャイル開発の導入に向け社内コンセンサスを得るために、適用効果など決め手はないか？

- ⇒何よりもスピードを重視している GAFMA といった企業が儲かっている事実を伝えると良い。