

# プロジェクトホスティングサービスのための EVM 自動描画システムの開発

プロジェクトマネジメントコース 矢吹研究室 0842062 工藤 亮

## 1. 背景

ソフトウェア開発はバージョン管理システムを用いて行われるのが一般的である。バージョン管理システムを提供するサービス（以下、プロジェクトホスティングサービス）も数多くある。

ソフトウェア開発プロジェクトの中心にバージョン管理システムをおくことができる。そのような開発体制においても、一般的なプロジェクトと同様に、PMBOK で提示されているようなマネジメントの活動がプロジェクトホスティングサービスでサポートされることが望ましい。

プロジェクトマネジメント（以下、PM）において、プロジェクトの進捗状況や進捗に係わるリスクを把握するために EVM（Earned Value Management）が用いられる。EVM は、プロジェクトを計画通りに進行させ、進捗状況に対応した判断を行うのに必要であり重要である。しかし、GitHub（プロジェクトホスティングサービス、Git を提供するサービス、ギットハブ）のような既存のプロジェクトホスティングサービス上では、EVM を見ることはできない。そのため、バージョン管理システムを用いるプロジェクトマネージャーは、計画時のデータと進行中の進捗データを取り出して、別々に処理することによって EVM を描かなければならない。

## 2. 目的

GitHub 上で自動的に EVM を描画するシステムを開発する。このシステムにより、プロジェクト関係者がプロジェクトの進捗状況を手軽に把握できる。その結果、作業の遅延の対応（リスク・マネジメント）やスケジュール管理（タイム・マネジメント）が効率よく行えるようになることも期待できる。

## 3. 手法

(1) Issues の内容の、「タスク名」「タスク期限日」「計画予定時間数」「時給金額」「材料費」「直接労働時間数」「進捗測定基準度」「使用材料費」を記述する。Issues とは、タスクを管理するためのシステムおよびそこで管理されるタスクである。1 つのタスクに対して 1 つの Issues が作成され、管理される。

(2) (1)の記述内容は、JSON 形式で{"time":20}のよ

うに記述する。

(3) ブックマークレットを利用して、GitHub のページ上に EVM を描画する。このブックマークレットは、ブラウザのアドレス欄から GitHub のプロジェクト名を抽出し、Google Chart API（グラフ描画ツール）のための URL を作成、その URL で描かれるグラフを、ページ上に挿入する。

## 4. 結果

開発したシステムによって GitHub 上で EVM を描いた様子が図である。このように、EVM のために必要な情報を Issues に記述しておくことによって、図 1 のような GitHub 上に EVM を描画することができる。

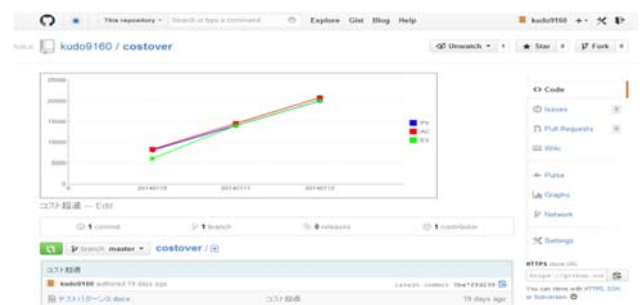


図 1.GitHub のページ上に描かれた EVM

## 5. 結論

ソフトウェア開発においてよく利用されているサービス GitHub 上で、プロジェクトマネジメントのための重要なツールである EVM を利用可能にするシステムを開発した。本システムは、GitHub でタスクを管理するのに用いられる Issues というしくみを利用している。Issues でのタスクの記述方法を統一するだけで利用可能であることが、本システムの大きな利点であろう。本システムによって、GitHub 上でのソフトウェア開発において、標準的なプロジェクトマネジメント手法を導入しやすくなることが期待される。

## 参考文献

[1] Project Management Institute,Inc. プロジェクトマネジメント知識体系ガイド. 第 4 版. Project Management Institute,Inc, 2008, 181p-183p.

# ドライブルートアシストのためのツイート収集システムの開発

プロジェクトマネジメントコース 矢吹研究室 0942013 伊藤 貴文

## 1. 背景

自動車体験を向上させる情報技術は大きく2つに分けられる。

第一に運転支援技術である。これはいくつか種類がある。車載型運転支援技術（ACC）は、前方を走行する車両との車間距離を一定に保ち、必要に応じてドライバーへの警告を行う技術である[1]。衝突被害軽減ブレーキは、前方の障害物等を検知し運転者へ警告する（衝突が避けられない場合にはブレーキの補助操作を行う）技術である[2]。レーンキープアシストは道路の白線等の走行環境を検知し、車両が走行車線を維持するよう、ハンドル操作を支援する技術で、例として日産のインテリジェントクルーズコントロールやスバルのアイサイトやトヨタのレーンキープアシストとして実現されている[3]。

第二にコミュニケーション支援技術である。これは運転者が運転中に情報を取得したり発信したりするのをサポートする技術で、例としてカーナビゲーションシステムやハンズフリー通話技術などがある。

これらの2つの技術の発展はめざましいものがある。例えばコミュニケーション支援技術においては、複数の自動車の運転中の急停止のデータを、それが発生した位置情報とともに収集することによって、道路や標識などの異常を発見できるようになっている[4]。このような技術は有用ではあるが、導入するためには、その技術を採用しているメーカーの新車を購入するしかないという欠点があった。

## 2. 目的

既存の自動車でも簡単に利用できる、運転支援・ナビゲーションシステムを作成する。このシステムは Twitter を利用して周囲の情報を収集し、運転社に通知する。

利用者の現在位置や地図上で指定した位置の周辺のツイートを収集・表示するシステムはすでに開発されている。しかしそのようなシステムを運転時に利用しようとする、ルートとは無関係なツイートが表示されてしまうという問題があった。

本研究では、収集するツイートの対象を、自動車が通行する予定の経路上のみに制限することで、この問題を解決する。

## 3. 手法

経路上の情報を Twitter で収集するカーナビを開発する。経路は Google Maps のディレクションサービスを用いて決定する。Twitter の情報は、Twitter API を用いて取得する。2つのサービスを組み合わせることで、目的のカーナビを実装する。

## 4. 結果

目的のカーナビのベースとなる、経路上のツイート収集システムが成果物である。図1は現在地から目的地までの経路上の Twitter のつぶやきを収集し、表示するアプリケーションのプロトタイプ画面である。経路上に複数ある赤い円の範囲のツイートを収集し、表示出来るようになっている。

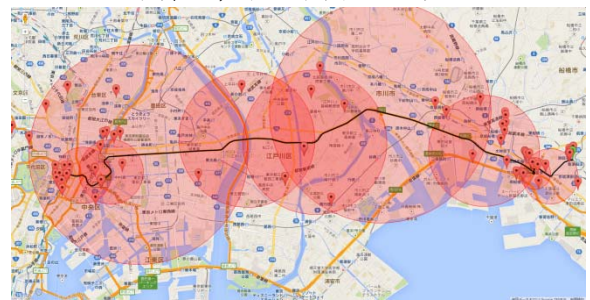


図1. 経路上のツイート収集アプリケーション

## 5. 結論

Google Maps のディレクションサービスと Twitter の位置を指定して検索するサービスを組み合わせることによって、収集するツイートの対象を、自動車が通行する経路上のみに制限するシステムを開発し、その動作を確認した。

このシステムによって、既存の自動車でも経路上の交通状況が分かるようになり、運転が楽になることや社会全体の渋滞などの交通障害が緩和されることが期待される。

## 参考文献

- [1] 日産インテリジェントクルーズコントロール  
<http://www.nissan-global.com/JP/TECHNOLOGY/OVERVIEW/icc.html>
- [2] スバル・アイサイト  
<http://www.subaru.jp/eyesight/digest/>
- [3] トヨタ・レーンキープアシスト  
[http://www.toyota.co.jp/jpn/tech/safety/technology/technology\\_file/active/lka.html](http://www.toyota.co.jp/jpn/tech/safety/technology/technology_file/active/lka.html)
- [4] Honda インターナビの急ブレーキ多発地点  
<http://www.honda.co.jp/news/2013/c130329a.html>

# チケットを活用するオープンソースソフトウェア開発の実態調査

プロジェクトマネジメントコース 矢吹研究室 0942038 久保 孝樹

## 1. 研究背景

ウェブアプリケーションなどのソフトウェア開発プロジェクトでは、開発中に変化していく環境や顧客の要求に柔軟に対応していかなければならぬため、開発プロセスもそのような事柄を考慮したものが求められている。従来のウォーターフォール型では、作業を各工程に分割して実行する。原則として前工程が終了しなければ次工程に進めることができない。顧客の要求の変化や環境への変化に対応していくためには、手戻りをしなくてはならない。そのため、要求や環境が変化すると、必然的にプロジェクトは失敗してしまう。このような問題の解決策として、アジャイル型の開発プロセスが注目されている。アジャイル型の開発プロセスでは、環境や顧客の要求の変化に柔軟に対応することが出来る。アジャイル型の開発プロセスでは、環境の変化、顧客の要求やバグの修正など、様々なタスクが発生する。それらに柔軟に対応するため、チケットと呼ばれるツールが活用されている。チケットは、報告者、担当者、優先度、マイルストーン、種類、状況、コンポーネント、解決法、その他詳細をひとまとめにして Web 上で共有するための仕組みである。Web 上で管理されるため、プロジェクトメンバはいつでもチケットを参照、更新することが出来る。チケットはソフトウェア開発において重要なツールであり、これによって、プロジェクトの範囲や進捗の管理、プロジェクトメンバの管理が効率的に行えると考えられる。

## 2. 研究目的

チケットがどのようにソフトウェア開発プロジェクトで使われ、役立っているのかを調査する。そのために、ソフトウェア開発プロジェクトにおいて、最もよく使われているバージョン管理サービスである GitHub を利用し、GitHub におけるチケット (Issue と呼ばれる) の使われ方を調査する。具体的には、GitHub で公開されているソフトウェア開発プロジェクトにおいて、Issue がどのように使われているのかを調査し、チケットがどのように使われるのかを知りたいと考えた。

## 3. 研究方法

チケットのデータを GitHub 内のスター数のラ

ンキング上位の 36 件のプロジェクトから API を用いて取得し、チケット数の時間変化を調べ、それらの違いからソフトウェア開発プロジェクトの開発形態を調査する。

## 4. 結果・考察

図の 4 つのグラフは、GitHub 上のプロジェクトにおける Issue の時間変化の典型例で、以下のような特徴がある。

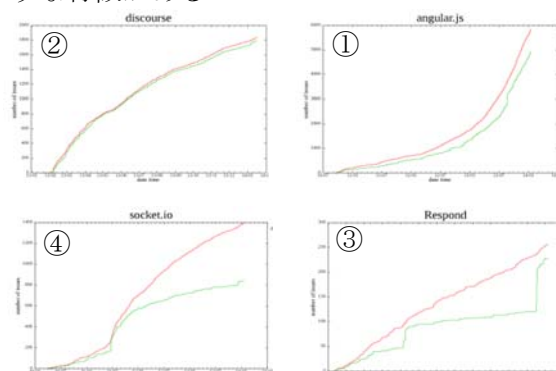


図 Issue 数の時間変化

- ① チケットの増加率が時間とともに減少する (全体の約 30%)
- ② チケットの増加率が時間とともに増加する (全体の約 40%)
- ③ チケットの消化が発行に追いついていない (全体の約 20%)
- ④ チケットの消化が停滞し急激に消化される (全体の約 10%)

Issue は、①では開発のタスク管理に、②ではバグ報告や機能追加などに使われていると考えられる。これらは順調に進捗されているプロジェクトであろう。③と④はタスクの管理、進捗の管理がうまく行えていないプロジェクトとなっていると思われる。このように、チケットの時間変化を調べることによって、オープンソースのソフトウェア開発プロジェクトを分類し、その実態を推測できるようになることがわかる。

## 参考文献

- [1] 小川明彦, 酒井誠. チケット駆動開発. 翔泳社, 2012-8-23.
- [2] 濱野純, 入門 Git, 秀和システム, 2009-9-25.
- [3] 大塚弘記. 詳解 GitHub. WEB+DB PRESS. 技術評論社, 2012-7-25, Vol.69. p.18-52.



# プロジェクトマネジメントを学ぶゲームの開発と運用実験

プロジェクトマネジメントコース 矢吹研究室 0942083 Htet Myet Mun Win

## 1. 背景

ゲームの考え方やデザイン・メカニクスなどの要素をゲーム以外の社会的な活動やサービスなどに利用することを表す「ゲーミフィケーション」が注目されている。その成功事例には以下のようなものがある。

- MyBarackObama.com : 選挙活動
- Badgeville : ビジネスモデル

このようにゲーミフィケーションは多岐に渡って利用されており、プロジェクトマネジメント（以下 PM）もゲーミフィケーションによってより良くなることが期待できる。

## 2. 目的

本研究では PM にゲーミフィケーションの概念を導入することを試みる。具体的な方法としては、学生など PM をあまり知らない人を対象とした PM の学習を、ゲーミフィケーションを利用して行えるようにする方法を提案する。これによって PM についての理解を深めることが期待できる。

## 3. 手法

- (1) PM を学習するゲームを提案し、細かな仕様を PM の要素と関連させて制作する。
- (2) (1)を複数回運用し、類似プロジェクトの類推見積もりの制度上昇と実コストの低減に効果があるかを検証する。

## 4. 結果

PM の学習をするための方法は座学だけではなく、実際にプロジェクトを行う形式もあり、インターネット上ではプロジェクトをシミュレーションするものが存在していた。

ゲーミフィケーションやシリアスゲームの実用例から、目標を可視化し、世界観を作ることによってユーザに受け入れ易くさせ、反復効果を狙い学習効果を高めるのが良いという結果が得られた。

そこで、魔王に支配された世界を救うために、規定のターン内に支配下の都市を解放していき、魔王を倒すというプロジェクトを行うゲームを制作した。このゲームはゲームマスターとプレイヤーが 1 人ずつ、トランプとサイコロとメモ用紙があれば行うことができる。ゲームの流れとしてはトランプの数字札が 3 枚配られ、それをメンバとして、能力を強化したり、人員を追加したりして各都市を解放していく。道中、メンバが怪我をしたり、敵が強化されたりするトラブルが発生する。また、敵との戦闘で都市に損害が発生する仕組み

を作り、1 都市で一定の損害が出ると修復を行わなければ都市解放ができない仕組みを作った。これらを数字としてまとめ、定期的に教えることにより、意思決定をさせる。

このゲームに含まれる PM の要素は以下の通り。

- 統合マネジメント：全体を管理する
- スcopeマネジメント：全体の都市数をタスクとして見積もる
- タイムマネジメント・コストマネジメント：目標を設定し、それを EVM で監視することによって目標を達成する
- 品質マネジメント：目標を設定し、都市の損害による品質の低下をメンバや修復行動で対処する
- 人的資源マネジメント：最初に渡されるメンバを訓練し能力を強化するか、人員追加などの意思決定を行い管理していく
- コミュニケーションマネジメント：プロジェクト内で得られた情報をどう活用していくか考えて整理していく
- リスクマネジメント：起こりうるリスクを想定し、対策したり、許容をする

被験者 2 人で試した結果、被験者 1 の見積もりの値は 1 回目が 40,000 円で 2 回目が 60,000 円と 2 回目が多いのに対して、実コストは 1 回目が 70,300 円で 2 回目が 37,100 円と 2 回目の方が少ないという結果となった。被験者 2 の見積もりは 1 回目が 50,000 円で 2 回目が 40,000 円と 2 回目が少ないのに対して、実コストは 1 回目が 41,000 円で 2 回目が 36,400 円と 2 回目の方が少ないという結果となった。

2 回目には成績がよくなること、つまり、ゲームの経験によって類推見積もりの精度が上昇し、実コストが低減するようになることが期待されるが、実験結果の t 検定では、2 回の結果には差がないという帰無仮説は棄却されなかった。

## 5. 結論

本研究は、PM を学ぶゲームにより、類似プロジェクトの類推見積もりの精度上昇と実コストの低減を期待したものである。

ゲームで遊んでもらった結果は、その有効性を示唆するものではあったが、統計的検定によって有意と言えるものではなかった。被験者を増やすことでこの問題が解決されることが期待される。

## 6. 参考文献

[1] 井上明人. ゲーミフィケーション <ゲーム> はビジネスを変える. 第3版, NHK出版, 2012年4月.

# ユーザ評価データのマイニング結果に基づく

## スマートフォンアプリの特徴分析

プロジェクトマネジメントコース 矢吹研究室 0942112 増田 知之

### 1. 研究背景

世界中でスマートフォン利用者が年々増加している。スマートフォンのプラットフォームには、Androids や iOS, Symbian, Research In Motion, Bada, Microsoft などがあるが、その中でも特に、Android と iOS の利用者が多い。

Android と iOS のスマートフォン上で動作するアプリケーション（以下アプリ）は、Google が運営するアプリ配布サイトである Google play, Apple が運営するアプリ配布サイトである App store で配布されている。この2つのストアでは、2013年11月までにおよそ195万ものアプリが配布されている[1]。

これらのアプリは、それぞれのストアでランキングによって順位付けされている。ランキングで上位のアプリはユーザに好印象を与えるアプリであり、下位のアプリはユーザに悪印象を与えると考えられる。しかし、ランキング上位のアプリが必ずしもユーザに好かれているわけではない。そのようなアプリでも、ユーザからの評価値（星の数）が少ないものはある。

### 2. 研究目的

本研究の目的はユーザの評価に基づいてアプリを分類することである。アプリに対してユーザが付ける評価点（星の数）の分布は、アプリによってさまざまである。例えば、評価点1と5のみが同数あるアプリと、評価点3のみのアプリは、平均評価点は同じだが、その性質は大きく異なると思われる。アプリの性質の違いが最も際立つような評価点の解析方法を見出し、その解釈によってアプリを分類することを目指す。

### 3. 研究方法

アプリの無料・有料ランキング及び評価データを毎日18時に取得データをする。取得したデータをもとに、データマイニング手法を適用することで、アプリを分類し、その結果を考察することによってアプリの評価指標を見出す[2]。

具体的には、アプリにつけられた評価値（星）をその値（星の数）ごとに集計し、主成分分析を試みる。

### 4. 結果

iOS のトップセールアプリの評価値を、星1つの割合から星5つの割合まで、5つの数値で表現し、主成分分析を行った結果を図1に示す。第1主成分は星1つの星5つの割合への重みが大きく、第2主成分はレビューの影響度によって分散されているものであった。第1主成分スコアは評価値の平均とよく相関しているが、第2主成分スコアは評価値の平均と相関していない。

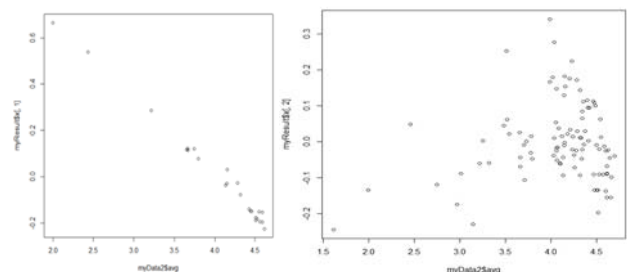


図1 平均評価点と主成分スコアの相関。  
横軸は平均評価値。縦軸は主成分スコア  
(左が第1・右が第2主成分スコア)

### 5. 結論

スマートフォンアプリの評価値を主成分分析した結果、アプリの評価は星1つと星5つを重視する見方とそれぞれの星の影響度に分けられることがわかった。評価値の平均とよく相関する第1主成分スコアはアプリ評価の新たな視点とはならないが、評価値の平均と相関しない第2主成分は、アプリ評価の新しい視点になることが期待できる。

ユーザ評価データをマイニングすることによって、個々のユーザからのアプリへの評価や、評価値の単純な平均、アプリのランキング以外にも、アプリを評価する指標があり得ることを示した。このような視点を活用することによって、開発者が、ターゲットとなるユーザからの評価を開発に生かしやすくなることが期待される。

### 参考文献

- [1] Taisyo. App Store のアプリ数、95万本に到達か?!  
<http://taisyo0.com/2013/09/25/19997.html>  
2014.1.20.
- [2] 金明哲. Rによるデータサイエンスデータ解析の基礎から最新手法まで一, 森北出版. p.320.  
2007

# オープンソース開発におけるソフトウェアテストの実態調査

プロジェクトマネジメントコース 矢吹研究室 1040060 清水 竜吾

## 1. 研究背景

これまでのソフトウェア開発の現場では、主にウォーターフォール型開発手法が採用されていた。それに代わってアジャイル型開発手法が普及してきている。アジャイル型開発手法では、テスト駆動開発手法 (Test Driven Development) がよく採用される。これは、先にプログラムに必要な機能について初めにテストを考案し、そのテストが動作する必要最低限の実装を行い、コードを洗練させ、短い工程で計画・開発を繰り返す手法である。

実際の製品として販売・配布されている多くのソフトウェアはテストを含めて、開発プロセスに関する情報は一般には知ることができなかったが、近年では開発プロセスに関する情報を一般でも見ることができるオープンソースソフトウェア (OSS) の開発が盛んである。OSS 開発には、OSS ホスティングサービスを利用しての開発されることが多い。ホスティングサービスとはソースコードやドキュメントのバージョン管理するシステムやバグ・テスト情報を記録できるシステムなどの複合システムである。

最もよく利用されているホスティングサービスの一つが GitHub である。実際、GitHub の登録リポジトリ数は、サービスを開始した 2008 年には 38,423 件だったのに対して、2012 年には 4,614,306 件と大幅に増加した。ユーザ数も同期間に 41,157 人から 2,763,437 人へと大幅に増加した。

このように多くのプロジェクトをホストする GitHub のプロジェクトを調査・解析することによって、近年のソフトウェア開発傾向を調べることができると思われる。

## 2. 研究の目的

本研究では、OSS のソフトウェア開発工程であるテスト工程に着目してソースコードの成長の様子を調査し、ソフトウェア開発プロセスを明らかにすることを目的とする。

## 3. 研究方法

GitHub でホスティングされているソフトウェアはバージョン管理が行われているため、ソフトウェアの更新履歴を知ることができる。更新履歴を用いてソフトウェアのメインコード (test ディ

レクトリー外) とテストコード (test ディレクトリー内) の行数を調査することによって、ソフトウェアの成長を視覚化し、どのように開発されているかを調査する。

## 4. 結果・考察

以下のグラフは調査した結果の典型的な例である。赤はコード合計、緑はメインコード、青はテストコードである。

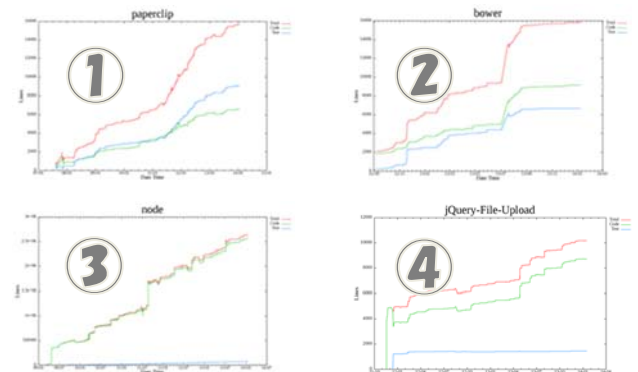


図 1.OSS におけるソースコードの行数の時間変化

調査した全 37 件のプロジェクトにおいて、コードの成長の仕方は、以下の 4 パターンに分類できた。

- ① メインコードよりテストコードのほうが多い
- ② メインコードとテストコードが共に成長する
- ③ テストコードが殆ど書かれていない
- ④ テストコードが初期状態から殆ど変化がない

このように、ソースコードの行数の時間変化に着目して、OSS 開発の実態を調査することで、ソフトウェア開発においては、テスト駆動開発がよく採用される手法だと言われているが、それを採用していないプロジェクト (パターン 3 と 4) が、調査サンプルの中では 54%にも及んでいる。このような開発体制の違いの原因と、それがプロジェクトに及ぼす影響を調査することが今後の課題である。

## 参考文献

- [1] Mint (経営情報研究会). 図解でわかる ソフトウェア開発のすべて. 日本実業出版社, 2000, 327p.
- [2] 大塚 弘記. 詳解 GitHub. WEB+DB PRESS. 技術評論社, 2012-7-25, Vol.69. p.18-52.

# オープンソースソフトウェア開発における役割分担の実態調査

プロジェクトマネジメントコース 矢吹研究室 1042067 関口 元基

## 1. 背景

オープンソースのソフトウェア開発のためには、バージョン管理システムや Wiki, バグ追跡システム, メーリングシステムがよく利用される[1][2]. これらのツールをまとめて提供するのがホスティングサイトであり, その代表的なものが GitHub である.

現在, オープンソースソフトウェア(以下 OSS) 開発は数多く世界中で行われており, その多くは GitHub で行われている. GitHub の API を使用することによりプロジェクトの活動の様子やプロジェクトメンバの活動ログを収集することができる.

GitHub の API には以下のようなイベントがある[3].

- DeleteEvent (c): デリートをを行ったイベント. プロジェクトで行われていたイベントを削除したという活動のログ.
- IssueCommentEvent (d): Issues にコメントを行ったイベント. プロジェクトメンバに限らず, 第三者も Issues にコメントしたという活動のログ.
- IssuesEvent (e): Issues を行ったイベント. プロジェクトメンバに限らず, 第三者も Issues を発行したという活動のログ.
- PullRequestReviewCommentEvent (g): プルリクエストにコメントを行ったイベント. 管理者がプルリクエストにコメントをした活動のログ.
- PushEvent (h): プッシュを行ったイベント. 変更履歴をアップロードした活動のログ.
- ForkEvent (j): フォークを行ったイベント. 自分のアカウント内に既存のリポジトリの複製をつくったという活動のログ.
- WatchEvent (i): スター(お気に入り)をしたイベント. 自分が気になっているリポジトリにスターを付けたという活動のログ.

このような GitHub の API を使い, 活動ログを取得・解析し, 結果を解釈することによって, 今まで明らかになっていなかった OSS プロジェクトにおける各メンバの役割の分担状況を明らかにすることが期待できる.

## 2. 目的

GitHub 上で公開されている OSS を調べるこ

と, プロジェクトメンバの役割分担の実態を明らかにする.

## 3. 方法

以下の手法を用いる.

- ① API を使用し, GitHub 上で行われているプロジェクトメンバ全員の活動ログを収集する.
- ② 各イベントが何回行われているかの活動ログを, プロジェクトメンバごとの一覧表にまとめる.
- ③ ②で得られた一覧表のデータを主成分分析し, 結果を解釈する.

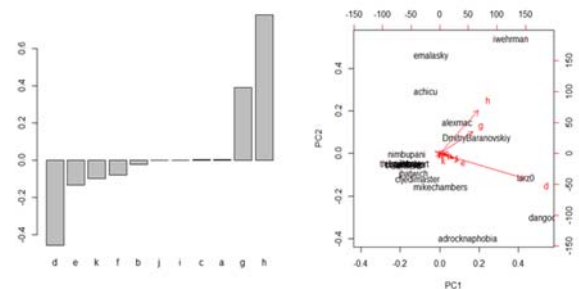


図 1 主成分(左)と主成分スコア(右). 主成分の横軸の文字は, 背景で挙げた API の括弧内の文字と対応している.

## 4. 結果と考察

各プロジェクトの主成分には次の傾向が見られた. PushEvent と WatchEvent の絶対値が大きく, 正負が逆であることが分かった. つまり, Push する行為とリポジトリにスターを付ける行為は別々のメンバが行っていることが多い.

プロジェクトメンバの役割分担の実態を解明することができた. このような手法を活用することによって, OSS 開発プロジェクトの実態を明らかにしたり, プロジェクトマネジメントの手法を導入したりすることが容易になると期待される.

## 5. 参考文献

- [1] Fogel Karl. オープンソースソフトウェアの育て方. ツールが一通り揃ったホスティングサイト. <http://producingoss.com/ja/web-site.html>, 2013.12.24
- [2] GitHub Social Coding. <https://github.com>, 2014.1.25
- [3] GitHub Developer <http://developer.github.com/v3/activity/events/types/#gollumevent>, 2014.1.11



# バージョン管理ホスティングサービスと連携するコース管理システムの開発

PM コース 矢吹研究室 1042098 野口杏子

## 1. 研究の背景

塾や学校での授業でインターネットなどを用いた e ラーニングと呼ばれる学習管理システム (LMS) による学習が増えてきている。LMS を用いると、その中での学習者の行動を記録し把握することができる。学習者の行動である行動履歴分析することで、授業方法の改善ができ学習者の理解度の向上につながっている。さらに学習者自身も行動を確認できることで学習ポートフォリオを作ることにも可能であり、教師だけではなく学習者にも利点がある。LMS は、学習の際さまざまな活動に対応できるように多機能である。用途などに応じて機能を選択して利用することができるが、別の一般的なシステムとは連携していない。学習をする際に LMS ではないシステムを利用した時には、学習やの行動を把握することができない状況である。そのため、LMS 上で、一般的なシステムの行動も把握できる環境があると望ましい。

たとえば、ソフトウェア開発ではバージョン管理が必要不可欠である。なぜなら、バージョン管理をしっかりと行うことで、常に最新のデータや過去の変更履歴を管理することができるからである。バグを早期発見でき、大きな手戻りが発生することも防ぐことができることから重要である。

このような情勢があることから、ソフトウェア開発の教育現場でも、バージョン管理システムの導入が求められるが、既存の LMS は Subversion や Git のような一般によく使われるバージョン管理システムとは連携していない。現状 LMS 上で把握することはできない。そのため、学習者の行動を把握するには LMS とバージョン管理システムをそれぞれ個別に確認しなければならず、教師への負担が大きくなることが懸念される。

## 2. 研究目的

オープンソースの LMS である Moodle 上で、学習者の GitHub での行動履歴を確認できるシステムの開発を開発する。のシステムを利用することには、学習者と教師の両方にメリットがある。学習者は、GitHub というソフトウェア開発の現場でよく使われているツールに親しむことができる。教師は、GitHub のようなツールを独自に準備・運営することなく教育に活用でき、学習者の学習状況も適切に把握できる。

## 3. 研究方法

Moodle に標準で備わっている、Moodle 上での学習者の行動履歴を表示する機能を拡張し、GitHub での行動履歴も表示できるようにする。

学習者が自分の GitHub アカウント名を Moodle に登録しておけば、教師が Moodle 上で学習者の行動履歴を閲覧しようとしたときに、システムが GitHub の API を用いて GitHub からその学習者の行動履歴を取得し、整形して画面に表示するようにする。

## 4. 結果と結論

図 1 は、Moodle 上で表示された GitHub の行動履歴である。学習者が、いつ・GitHub 上のどのプロジェクトで・何をしたのかが表示されている。教師は、GitHub を利用する学習者の学習状況を、GitHub にアクセスせずに確認できる

LMS である Moodle 上で、GitHub という外部のサービスの行動履歴を確認するシステムを開発し、その動作を確認した。本手法を応用すれば、教育の現場に GitHub 以外のサービスを導入しても、学習者の学習状況を、教師は詳細に把握できるようになるだろう。



図 1. Moodle の画面に表示された GitHub での行動履歴

## 参考文献

- [1] Moodle を利用した授業展開と利用履歴の解析. 2005-01-20. [http://crf.flib.u-fukui.ac.jp/dspace/bitstream/10461/5038/4/vol.37\\_p13-20.pdf](http://crf.flib.u-fukui.ac.jp/dspace/bitstream/10461/5038/4/vol.37_p13-20.pdf).
- [2] IPA. 非ウォーターフォール型開発の普及要因と適用領域の拡大に関する調査. 2012-06-24. [http://sec.ipa.go.jp/reports/20120611/reports\\_20120611-2.pdf](http://sec.ipa.go.jp/reports/20120611/reports_20120611-2.pdf).
- [3] 植野真臣. 知識社会における e ラーニング. 初版, 培風館, 2007.