目次

1. 序章
   1. 本章の構成
   2. 研究の背景
   3. 研究の目的
   4. 研究の方法
   5. プロジェクトマネジメントとの関連
   6. 本論文の構成
2. ソフトウェア開発について

　2.1　本章の構成

2.2　成功事例・失敗事例・導入例

2.3　プロジェクトマネジメントとの関連

第3章　GitHubについて

　3.1　本章の構成

　3.2　バージョン管理システムについて

　3.3　バージョン管理システムの必要性

　3.3　GitHubの機能説明

　3.4　プロジェクトマネジメントとの関連

第4章　マネジメントについて

　4.1　本章の構成

　4.2　プロジェクトマネジメントプロセス群と知識エリア

　4.3　リスク・マネジメントについて

　4.4　タイム・マネジメントについて

　4.5　監視・コントロール・プロセス群

第5章　EVMについて

　5.1　EVMの仕組み・必要なデータ

　5.2　成功事例

　5.3　失敗事例

　5.4導入例

　5.5プロジェクトマネジメントとの関連

第6章　機能提案および構築について

6.1　本章の構成

6.2　機能提案の仕組み

6.3　機能提案に必要なデータの定義

6.4　機能提案に必要なデータの抽出方法

6.5　運用

6.6　検証

第7章　まとめ

　7.1　考察

　7.2　結果

**第1章**

**序章**

* 1. 本章の構成

第1章では，本論文の序論について述べる．研究背景，目的，方法，プロジェクト

ネジメントとの関係，本論文の構成について述べる．

* 1. 研究の背景

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が一般的になった理由は， プロジェクトを運営していく上でいろいろな場面で役立つということである． 開発者どうしのコミュニケーション，リリース管理，バグ管理，コードの安定性の確保，安心して新機能を実験できる環境，各開発者の権限の管理など，あらゆる場面でバージョン管理が利用することができるのである．バージョン管理システムは，これらの内容をひとまとめにして管理する．その中心となる機能が，変更管理である．これは，プロジェクト内のファイルが変更されるたびに，その変更についてのメタデータ（更新日や更新者など）を収集する仕組みである．そして，あとからその変更の内容を再現できるようにするのである．つまり，変更が発生した単位で情報を管理する仕組みといえるのである．

バージョン管理システムはソースコード管理システムなどとも呼ばれ，大規模な開発を行う際には必須と言っても過言ではないのである．また，大規模な開発だけでなく小規模な開発や個人による開発においても，ファイルの変更履歴の記録やバックアップといった用途で活用されており，プロジェクト内のさまざまなファイルの変更履歴を管理するためのテクノロジーや習慣を組み合わせたものである．バージョン管理システムを提供するサービス（以下，バージョン管理サービス）もSubversionやCVS（Concurrent Versions System），Git，Mercurial，Visual SourceSafeなどと数多くある．

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WBS（Work Breakdown Structure）やEVM（Earned Value Management）のようなマネジメントの活動がバージョン管理サービスでサポートされることが望ましいのである．

プロジェクトマネジメントにおいて，プロジェクトの進捗状況や進捗に係わるリスクを把握し，コントロールするためにEVM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．GitHub（バージョン管理システム，Gitを提供するサービス．ギットハブ）のような既存のバージョン管理サービス上では，EVMを見ることはできないである．

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ジャーは，計画時のコストのデータと計画時の時間のデータを進行中の進捗時のコストのデータと進捗時の時間の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のである．

* 1. 研究の目的

バージョン管理サービスのGitHubのデータから自動的にEVMを描くシステムを開発する．このシステムは，GitHubのIssueを利用する．Issue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のことである．1つのタスクに1つのIssueが作成され，管理される．

このシステムにより，プロジェクトマネジャーを含むメンバー全員がプロジェクトの進捗状況を手軽に把握できるようになる．その結果，作業の遅延などへの対応（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．

* 1. 研究の方法

本研究では，バージョン管理サービスのGitHubを調査し，EVMに必要なデータとして，計画時のコストのデータと計画時の時間のデータ，進行中の進捗時のコストのデータ，進捗時の時間のデータをIssueに記述する仕様を決定する．そして，Issueに記述されたデータを抽出するプログラムを作成し，Issueから抽出されたデータをもとにEVMを描画するシステムを構築する．その後，過去のプロジェクトのデータを構築したEVMを描画するシステムに適用し，システムの運用とシステムの検証によるシステムテストを行う．

* 1. プロジェクトマネジメントとの関連

　本研究では，バージョン管理サービスのGitHubのデータの中からEVMに必要なデータを抽出し，そのデータをもとにEVMを自動的に作成できるシステムを構築する．このシステムにより，プロジェクトマネジャーを含むプロジェクトメンバー全員がプロジェクトの進捗状況を手軽に把握できるようになる．その結果，作業の遅延などへの対応が可能になり，リスク・マネジメントに貢献することができる．また，スケジュールの管理などへの対応も可能であり，タイム・マネジメントにも貢献することができる．

* 1. 本論文の構成

　第1章は，序論，第2章から第4章に関しては，各項目の基礎知識，問題点，必要性などをまとめていく．第5章から第6章までは実験の方法，考察，結果を記し結論をまとめていく．

　第1章では，研究の背景，目的，方法，プロジェクトマネジメントとの関連を記述していく．第2章では，ソフトウェア開発について調査する．ソフトウェア開発の歴史，成功事例，失敗事例，失敗事例の原因などを調査する．第3章ではGitHubの基礎知識を述べる．バージョン管理システムについての調査，バージョン管理システムの種類，成功事例などを述べる．第4章では，マネジメントについて述べる．本研究ではリスクマネジメントとタイム・マネジメントについて述べる．

第5章ではEVMについて調査する．第6章では，システム構築に必要な情報をまとめる．第7章では検証結果から，考察を行う．

* 1. 参考文献

[1] PMI：「PMBOK Guide 第4版」，PMI，pp．5-6，2008．

[2] CVS-Subversionを使ったバージョン管理,

http://sourceforge.jp/magazine/08/09/09/1038233

[3] バージョン管理 第3章 技術的な問題,

http://producingoss.com/ja/vc.html

第2章

ソフトウェア開発プロジェクト

2.1　本章の構成

　本章では，本研究はバージョン管理システムを中心におく開発体制になる前のソフトウェア開発プロジェクトについて調査する．ソフトウェア開発プロジェクトは以前ではどのように行われていたのかを調査し，ソフトウェア開発プロジェクトの歴史を知る．また成功事例，失敗事例，失敗事例の原因などを調査することによって，バージョン管理システムの必要性を証拠付ける．

2.2　ソフトウェア開発プロジェクトについて

　ここでは，ソフトウェア開発の歴史を記載する

ソフトウェアは今や世界的な必需品となり，ソフトウェアに従事する人口の世界の1，2位は米国と日本が占めている．両国とも過去多くのソフトウェアアプリケーションの開発に成功を収め，また同時に少なからざる失敗も重ねてきた．

ソフトウェアはすでに主要な産業に成長しており，20世紀の第4四半期においては，他のいかなる産業よりも急速に成長してきた．

ソフトウェアの開発は，企業を設立する際の初期投資に多くの資金を必要としないために，数々の発展途上国においてソフトウェアの世界市場への参入が始まっている．ソフトウェア産業のリーダは，1990年代の終わりの時点では，米国，日本，ドイツ，フランス，英国などであろうが，多くの国がソフトウェア開発のリーダの立場を狙っている．21世紀において　ソフトウェア開発の成功のために必要なもの

・ソフトウェアスケジュールを

2.3　ソフトウェア開発プロジェクトの成功事例

参考文献

[1] ソフトウェアの成功と失敗,

[2] 米国が世界のソフトウェア市場で強いの理由

http://www.atmarkit.co.jp/news/200711/09/dev.html