

「グループワークによるソフトウェア開発におけるインスペクション支援ツールの研究」

松下永寿[†] 松浦佐江子[‡]

芝浦工業大学 システム工学部 電子情報システム学科^{†‡}

1. はじめに

我々はグループのコミュニケーション支援を目的としたグループワーク支援システムを研究開発し、「情報実験Ⅱ」というグループワークでソフトウェア開発を行う授業で 2003 年度より利用している。利用者の経験からグループワークには仕様書を作成する過程において「メンバー間の仕様書の理解に相違が生じる」ことが最終的に出来上がるソフトウェアの質に影響を与えるので深刻な問題であると感じた。そこで、成果物に対して間違いや誤りを指摘するインスペクション (inspection: 検査・点検) [6]がグループメンバーの仕様書の理解向上に有効だと考え、インスペクションを支援するツールを提案する。ここで述べるインスペクションとは単なる誤字脱字などを発見し指摘をして訂正する作業とは違い、中間成果物を検査・点検する作業のことである。

2. 情報実験Ⅱの概要

本科目は、学部 3 年生を対象とした週 2 時限連続の専門科目である。プログラム関連の科目の学習を基に、半期で開発可能な規模ならびに複雑度をもったソフトウェアをグループで開発する。2004 年度に開発したソフトウェアは「芝浦工業大学大宮校舎会議室予約システム」、「自動販売機の制御ロジック」と「自動販売機のシミュレータ」の 3 つで、履修者はこのうち 1 つを選択し、教員がグループを編成した。各グループには、作業の補正を行うアドバイザーが 1 人ずつついて、教員 2 名と TA 4 名がアドバイザーを担当する。

開発プロセスを要求分析、システム分析、システム設計、実装、テストのフェーズに分け、各フェーズにおいて中間成果物を作成しながら開発を進めて、最終的なレポートとして作成する。

3. グループワーク支援システム

本システムは、[1][2]で共同開発された WEB システムで、機能としてアドレス帳・作業計画書・個人作業報告書・掲示板・アップローダがある。作業計画書は、作業計画を立ててグループのメンバーがその計画に基づき行動することで、メンバーの作業状況や自分のすべきことの把握など円滑なグループワークによるソフトウェア開発をサポートする狙いがある。作業計画書は、各フェーズで実際に行う作業を作業項目として登録することで作成される。本システムでは、新たに作業項目を作るとそれに対応した掲示板やアップローダが生成され、作業の中で作成される成果物の共有手段としてアップローダがあり、掲示板を使うことで成果物に関する問題を解決できるような手段を提供している。

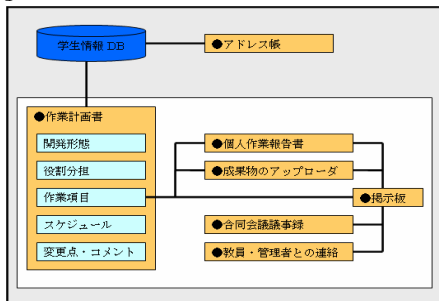


図 1 グループワーク支援システム

4. グループワークによるソフトウェア開発の問題点

我々は 2003 年度の情報実験Ⅱの経験から、本実験の問題点を以下のように考える。

問題点 1. どのような作業計画を立てて良いかわからない

学生は、作業すべき項目として作業項目を挙げて作業計画を立てる。しかし、学生のほとんどがソフトウェア開発プロセスに基づいたソフトウェア開発を行うことが初めてで、作業計画立案時に作業目的や担当者などの詳細を決定できない。2003 年度では作業内容がよく分からない作業項目が多く存在し、実際の作業と作業計画で立てた作業項目の作業との隔たりがあり、また一部のアップローダへ投稿するファイルの偏りなどという問題があった。

問題点 2. 仕様書の理解不足が最終的なソフトウェアの質に影響を与える

各フェーズで作成された成果物をまとめた仕様書を見てソフトウェアを実装するのだが、プログラム作成者が仕様について理解不足な点があり、作ったものを作り直すことがあった。また、仕様が完全に煮詰まってないまま実装に入ったこともあり、再度仕様を検討しなくてはならなくなったこともあった。

5. 開発プロセスにおけるタスクの設定

問題点 1 を解決するために、まず各フェーズを方向付け・定義・検査・まとめというカテゴリをフェーズの目標を達成するために必要な作業として用意した。このカテゴリ毎にタスクというフェーズの中でやるべき作業として用意した。例えばシステム分析フェーズでは、オブジェクトを抽出することが方向付けのタスクにあたり、定義ではオブジェクトの動作を記したシーケンス図の作成等で、検査では定義で作成したシーケンス図のインスペクションが、まとめはシステム仕様書の作成がタスクに該当する。タスクを用意することで、学生が作業計画を立てるときの指針ができ、問題を解決できると考えた。

6. インスペクション

「検査」という作業があるが、ここでインスペクションを行う。最終的に出来上がってくるソフトウェアの品質を保証するには、成果物の確認作業を行うことで実現でき、「定義」で確定した事項について「検査」でインスペクションを行うことで整合性を保つことが出来る。インスペクションは、複数人で対象のドキュメントを検査し、その際に役割が存在する。ドキュメントを作成しインスペクションの依頼をする人が「依頼者」、依頼を受け検査する人が「評価者」となる。

6. 1. インスペクションの流れ

インスペクションの手順は以下ようになる。

1. 依頼者が成果物を作成する
2. 依頼者が評価者に作った成果物を評価者にインスペクションしてもらうように依頼を出す
3. 評価者が依頼された成果物にインスペクションを行い、指摘事項を挙げる
4. 依頼者は、評価者から指摘された項目に関して意見がある場合は互いに検討し、指摘事項の中から成果物に反映すべき項目を決める
5. 依頼者はもとの成果物に 4 で確定した指摘事項を反映させた成果物を作成する
6. 評価者は依頼者が新たに作成した成果物に対して、指摘事項が反映されているか確かめる

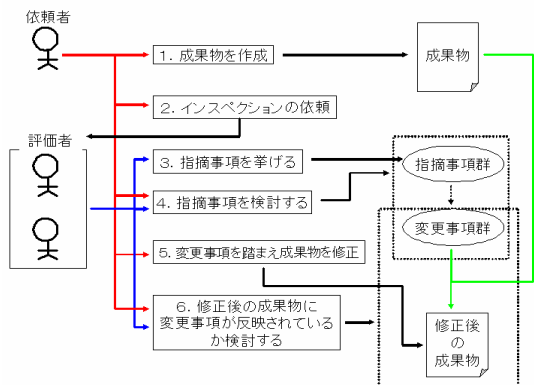


図2 インспекションの手順

6. 2. インспекション支援ツール

本システムでインспекションを行おうとすると、以下のような問題が出てくると考えた。

- ・ 本システムの掲示板を使ってインспекションを行う場合、議論している最中は気にならないかもしれないが、後で議論内容を確認したいとき、議論内容が整理されてない場合は読み取るのに苦労する。
- ・ 履修者の多くが今回初めてソフトウェア開発プロセスに基づいたソフトウェア開発を行うため、インспекションを行うときの流れや役割分担や詳細な指摘方法が分からず混乱が起こる。

以上の問題を回避するために、6. 1で記した流れ、またインспекション初心者が必要な詳細な指摘を行えるようにサポートするツールを開発、適用した。

7. 適用と評価

タスクを用意したこととインспекションを行ったことに関する評価をするために、2004年度の情報実験Ⅱにおいて適用実験を行った。ここでは、学生のアンケートと本システムから得られた学生の利用状況から分析結果を述べる。(履修者85名、内回答者71名)

7. 1. タスクについて

タスクを設けることが作業計画に役に立ったかという質問を行ったところ、63人が「役に立った」と9人が「役に立たなかった」という回答を得られた。それぞれの回答について理由を募った。「役に立った」と答えた人は、「タスクが与えられていることで、フェーズでやるべき作業を理解できた」という回答があり、また作業項目名が明確になることで「アップローダのファイルの参照がしやすかった」という意見もあった。「役に立たなかった」と答えた人は、「タスクの中にはどのような作業をやればよいか分からないものがあった」という意見があり、また実験前にグループワークソフトウェア開発に関する講義をしてほしいという意見もあった。

本システムの学生の利用状況から、2003年度では作業項目数が105だったのに対し、2004年度は1371と13倍以上になった。作業項目毎の掲示板の投稿数は、2003年度は202、2004年度が1485であった。アップローダの投稿数は、2003年度は948、2004年度が2583であった。

7. 2. インспекションについて

フェーズ毎のインспекション方法とその方法を選んだ理由についてアンケートを行った(表1)。対面で行っている理由として「意思疎通が容易」が多かった。掲示板を使った理由として「場所・時間的制約のため」が多く、少数ではあるが「ログを後でみることで次に作る仕様書のミスがなくなる」があり、ネットワークを介した議論の場として掲示板が活かされた。支援ツールを利用した人は「指摘事項がフォーマットされているので指摘が容易」という意見があり、インспекション初心者の学生にとって本ツールが役立ったと感じた。

表1 フェーズ毎のインспекション方法のアンケート結果

	対面	掲示板	支援ツール	その他
要求分析	60人	40人	0人	1人
システム分析	53人	43人	6人	1人
システム設計	49人	30人	8人	6人

本システムの学生の利用状況から学生のインспекションの実態を分析してみる。インспекションの作業項目の割合の平均は約30%で、一番高いグループで約44%、一番低いグループで約11%であった。

表2 各グループのインспекションの実態

グループ	A1	A2	A3	B1	B2	B3	C1	C2	C3
作業項目数	125	153	113	151	158	125	265	161	120
インспекションの作業項目数	44	68	36	61	48	28	29	45	51
掲示板から抽出した指摘事項数	151	210	14	56	48	39	14	41	114
本ツールを使っている作業項目数	10	0	6	7	7	0	0	2	1

8. まとめ

関連研究について、本研究と同様に、「インспекション」に着目してグループワークによるプログラミング授業における支援を行っている[5]では、教員と学生間のインспекションを支援している。[5]では2、3人のグループが対象で、本実験が9〜10人のグループが対象であった。本研究では学習者の学習意欲を促せるようなものが支援を目的として、学習者間のインспекション支援というアプローチ方法をとった。

タスクについて、作業項目の数と掲示板の投稿数が2003年度と比較して増加している。このことからタスクを設けたことは作業項目でやるべき作業が明確になり、グループでのソフトウェア開発が初めての学生が作業計画を立てるときに役に立ったのではないかと考える。

インспекションについて、アンケートから我々が狙いとしているインспекションのログを残してそれを利活用しているといった意見が少数であった。その理由として考えられることは、本科目で作成対象としているソフトウェアがログを残すほどのものではないか、また企業で一般に販売されているソフトウェアは面識のない開発者間でフェーズ毎に仕様書を受け渡して開発を行ったり、ユーザサポートのため開発を継続しなければならず、仕様決定までの過程におけるログの重要度は高く、学生の場合はそこまで作成過程の重要性が高くないのが要因だと考える。ログを残すことで仕様を決定するまでのグループメンバーの理解の向上、また前に立ち戻って現在作っているソフトウェア全体を見直して地盤を固めた品質の高いソフトウェアにしたいという狙いがあったが、学生間のインспекションということなので指摘する側の知識の問題があることや、そこまで学生の意識が向いていないのが現状ではないだろうか。しかし、平均約30%の作業項目がインспекションに割り当てられていたことから、作成した成果物のほとんどにインспекションを行ったことが分かり、またインспекションの後に再び成果物を作成するといった計画を立てているグループもあることから、インспекションの重要性は理解してもらえたと考える。本ツールについては、残念ながら表1、表2からもほとんど使われなかったことが分かる。しかし、少数であるが「指摘事項がフォーマットされているので指摘が容易」という意見もあったため、少なからず本ツールの有用性を示せたのではないかと考える。

今後の課題として、ログを残してもらえるようにするにはどのようにしたらよいか考えなければならない。その先のステップとして本ツールがあると考えます。

謝辞 本システムの利用およびアンケートに協力いただいた2004年度受講生に謝意を表す。

参考文献

- [1] 吉沼、松浦：グループワークによるソフトウェア開発教育のための進捗状況支援システム、情報処理学会第6回全国大会、pp.237-238、2004
- [2] 吉田、松浦：グループプログラミング授業における仕様書作成支援ツールの研究、情報処理学会第6回全国大会、pp.237-238、2004
- [3] 岡本敏夫：インターネット時代の教育情報工学Ⅰ、森北出版
- [4] 林、松浦：グループワークにおけるソフトウェア開発テンプレートを用いた進捗状況把握支援システム
- [5] 樋山、中野：ソフトウェア設計・開発グループ演習教育のためのコミュニケーション支援システム、情報処理学会論文誌、Vol.42、No.11、pp.2550-2561、2001
- [6] Tom Gilb, Dorothy Graham:ソフトウェアインспекション、共立出版