

# Project-Based Learning を支援するコース管理システム

明石 敬<sup>†</sup> 松澤 芳昭<sup>†</sup> 大岩 元<sup>††</sup>

Project-Based Learning(PBL) を支援するコース管理システムである Collasys を開発した。Collasys はプロジェクトの活動を支援する機能と PBL のコース運営を支援する機能を持ち、特にプロジェクト・成果物のレビュー・改善作業を考慮した。システムを構築するに当たって、支援の対象となるプロジェクトとは何であるかを明らかにするためにプロジェクトの再帰構造を用いた概念モデルを作成した。この概念モデルは汎用的なものであり、教育活動をプロジェクトと捉えることで、教育活動も同一のモデルで表現できる事がわかった。本稿では、プロジェクトの概念モデルとそのモデルを用いて実装された PBL を支援するコース管理システムである Collasys について報告する。

## A Course Management System Supporting Project-Based Learning

KEI AKASHI,<sup>†</sup> YOSHIAKI MATSUZAWA<sup>†</sup> and HAJIME OHIWA<sup>††</sup>

I have developed Collasys that is Course Management System for Project-Based Learning(PBL). Collasys has functions to support activities of projects and course management of PBL, and considered the review and the improvement work of projects and products. I have defined a conceptual model of project which has a recursive structure to understand the project. This conceptual model is general and it can express educational activities by considering educational activities to be projects. In this report, this conceptual model of project and Collasys that is Course Management System for PBL are explained.

### 1. はじめに

近年、問題発見解決能力の育成手法として Project-Based Learning(PBL) が注目されている。PBL は、学生がグループを組みプロジェクト形式で問題発見・解決を行う教育である。従来の講義形式のように指導者からの知識伝達を行うのではなく、学生が主体となってプロジェクトの遂行を通して学んでいくのが特徴であり、実践的なスキルを身につけることができる。

筆者らは 2005 年秋から PBL 型のソフトウェア開発教育であるコラボレイティブ・マネジメント型情報教育を実施している<sup>1)</sup>。

この教育では、一般的なプロジェクトと同様に、学習者らが実施するプロジェクトにおいて成果物やプロセスの質の向上のために、レビュー・改善作業

を反復して行っている。また、毎週行われる授業の時間には、各プロジェクトの週の成果を発表し、プロジェクト間で相互レビューを行っている。

このような活動を効果的に行うためにはレビューやレビューに対する議論の記録・共有が不可欠である。それを実現するためにはシステムの支援が必要である。

PBL の支援する環境として、Microsoft Project に代表されるプロジェクト管理システム、Moodle や WebCT に代表されるコース管理システムが挙げられる。また、PBL 専用の支援環境として、海外では Hadj らによって開発された NetPro が国内では西森らによって開発された Project Board などがある。これらのシステムは 4 節で述べるように一長一短であり、2 節で述べる筆者らが行っている PBL 型のソフトウェア開発教育の支援を行う上では、十分とは言えない。

こうした背景から、学習者のプロジェクト支援システム Collasys を開発した。Collasys で採用した概念モデルは汎用的なものであり、そのモデルを利用することでコース管理機能も実現することができた。

<sup>†</sup> 慶應義塾大学 政策・メディア研究科  
Graduate School of Media and Governance, Keio University

<sup>††</sup> 慶應義塾大学 環境情報学部  
Faculty of Environmental Information, Keio University

本稿では、我々の行っている PBL 型のソフトウェア開発教育、関連研究、支援システムの基礎となる概念モデル、支援システムの概要を述べる。

## 2. コラボレイティブ・マネジメント型情報教育

### 2.1 教育の概要

コラボレイティブ・マネジメント型情報教育は、2005 年秋学期より、慶應義塾大学藤沢キャンパスで実施されている、産学連携のソフトウェア開発教育である。この教育では、3-4 名の大学学部生のソフトウェア開発プロジェクトに、Project Manager(以下、PM)として IT 企業の若手社員がプロジェクトに参加し、プロジェクトマネジメントを行う。学生のソフトウェア開発の実践的な知識・スキルの育成と、IT 企業の若手社員のプロジェクトマネジメント能力の育成がねらいである。ソフトウェア開発プロジェクトは 4-5 プロジェクトが実施され、それぞれのプロジェクトが 1 学期間を通して異なる目標を設定し、ソフトウェア開発を行う。

### 2.2 授業における活動

1 週間に 2 コマある授業時間には、一般的な知識伝達型の講義は行われず、以下の活動を行っている。

#### ● 進捗報告会

進捗報告会では、プロジェクトの週の活動の成果とそのプロセスを発表し、相互評価を行う。進捗報告会を行うことで、プロジェクトに発生した問題や得られた知見をプロジェクト間で共有することができる。

#### ● PM ミーティング

PM ミーティングでは、コーディネータ、オブザーバ、各プロジェクトの PM が集まり、それぞれのプロジェクトの PM からの視点での週の活動の報告とプロジェクトマネジメント手法についての討論が行われる。それぞれのプロジェクトについて報告・討論を行うことで、プロジェクトマネジメント手法の共有ができる。

#### ● 勉強会

勉強会は、必要に応じて学生から希望のあったテーマを取り上げ、講義形式の授業が行われる。講師は、コーディネータが務めることもあるが、多数を学生自身が務め、コーディネータや研究室の大学院生が講義資料などのレビューを行う。

### 2.3 プロジェクトにおける活動

自分たちで設定した目標を達成するために、複数人で作業分担を行い、成果物を作成する。プロジェクトを実施する上での条件として「人に使ってもらえるソフトウェアを作る」が設定されている。人に使ってもらえるソフトウェアを作成するためには、高い品質が求められる。そのため作成した全ての成果物はプロジェクトメンバーでレビューが行われ、改善作業を行う。

## 3. 支援環境に対する要求

2 節で述べた教育の実施経験から支援環境に対する要求をまとめる。

### 3.1 レビュー・改善作業を支援する

プロジェクト内では成果物を目標とする品質にするまでプロジェクト内でレビューを行う。レビューの効果として成果物の品質の向上だけでなく教育効果の向上が見込まれる<sup>4)</sup>。

円滑にレビュー・改善作業を行うためには、成果物とレビューの履歴の記録が必要である。また、複数人でレビュー・改善作業を行う場合は、途中からレビューに参加したレビュアーを考慮し、レビューとレビューに対する議論の記録も必要である。

### 3.2 プロジェクト外からのレビューを支援する

前述したとおり、筆者らが実施しているコラボレイティブ・マネジメント型情報教育の授業時間にはプロジェクトの報告を行い、他のプロジェクトのメンバーやコーディネータ、オブザーバはプロジェクトのレビューを行っている。これらのレビューも記録・共有することで翌週からのプロジェクトの活動が円滑に行えると考えられる。また、レビュアーも前回のレビューを参照することができれば、レビューをどのように反映したのかをプロジェクトに質問することができ、授業での議論が活発になると考えられる。

### 3.3 ユーザの役割を管理する

Hadj らは、PBL を「誰がプロジェクトを定義し計画するか」「誰がプロジェクトをコントロールするか」の 2 軸で PBL の分類した。表 1 に示す表が PBL の分類である。筆者らが実施しているコラボレイティブ・マネジメント型情報教育は、プロジェクトの定義・計画とコントロールを社会人 PM(学習者)が行っていることから「4. Working on projects」に当てはまる。

これら 4 つの種類の PBL は、プロジェクトの定

表 1 Ways of using PBL

	Def- inition and Plan- ning done by teachers	Def- inition and Plan- ning done by students
Done and con- trolled by stu- dents	3. Learning by projects	4. Working on projects
Done by stu- dents and con- trolled by teach- ers	1. Learning from teachers	2. Learning project management

義・計画・コントロールを指導者が行うか学習者が行うかの違いであり、実際に実施される活動に違いはない。従って、ユーザの役割に応じて実行できる機能を管理することで4つの種類のPBLに対応することが可能であると考えられる。

また、PM、メンバ以外にもクライアントやオプザバーバ、技術的な支援を行うサポータなど様々な役割がある。これらも役割に応じて実行できる機能を管理しなければならない。

### 3.4 プロジェクトの階層化 (WBS) による管理を支援する

大きい規模のプロジェクトになると作業を小さな粒度に分解しプロジェクトメンバで役割分担を行う必要がある。プロジェクトの階層化を行うことで管理しやすい粒度まで作業を分解することができる。WBSはプロジェクト管理の一般的な手法であり、筆者らの実施する教育でもほとんどのプロジェクトで採用されている手法である。WBSをプロジェクトメンバ内で簡単に共有する仕組みがあれば、プロジェクトは円滑に進むと考えられる。

## 4. 関連研究

関連研究としてコース支援システム、プロジェクト管理システム、PBL専用の支援環境を挙げこれらを比較する。

### 4.1 コース管理システム

コース管理システムとは、「高等教育機関における一学期分の講義など、ひとまとまりの教育プロセス (=コース) において、講義時間だけでなく、課外時間での教育・学習活動も含め、トータルに支援することにより、教育効果及び学習成果を最大にするためのシステム」である<sup>5)</sup>。商用ではWebCTやblackboardなどが有名であり、近年では、オープンソースのmoodleやsakaiなどが注目されている。

多くのコース管理システムは、グループワークを支援する機能としてフォーラム (掲示板) やチャットなどの機能が予め用意されている。しかし、一般的な講義形式の授業を支援するのが目的であり、表1で示したPBLの分類における「1.Learning from teachers」の支援が中心である。筆者らが実施している教育は前述した通りPBLの分類における「4.Working on projects」であるため、コース管理システムのみでは支援環境として不十分である。

### 4.2 プロジェクト管理システム

プロジェクト管理システムは、プロジェクトを実施する上で必要となる情報を管理するためのシステムである。WBSとガントチャートによるスケジュール管理や進捗管理が主に利用される機能である。また、Microsoft Projectのように他のツールと連携して成果物の管理も行うシステムもある。

プロジェクト管理システムは単一のプロジェクトを行うことを想定している。そのため、コース内で実施される一つのプロジェクトを管理することには向いているが、コラボレイティブ・マネジメント型情報教育で実施されている進捗報告会のような他のプロジェクトに対してレビューをする、といった教育活動を支援する仕組みは整えられていない。

### 4.3 PBL 支援環境

海外ではHadjらによって開発されたNetPro<sup>2)</sup>、国内では西森らによって開発されたProject Board<sup>3)</sup>などがPBL支援環境として挙げられる。

NetProは遠隔でのPBLを支援するためのシステムである。学生が遠隔でプロジェクトを行うための機能が備わっており、成果物の共有・編集、スケジュールの共有、プロジェクト間でのピアレビュー機能がある。

これら2つの支援環境で共通する点はプロジェクトの成果や進捗状況をクラス内で共有することである。また、NetProはプロジェクト間でのピアレビューを行うことから筆者らの求める支援環境に近い。

しかし、NetProもProject Boardもプロジェクト内で成果物をレビューし改善するといった活動は考慮されておらず、システム上で複数人でレビューを行うといった仕組みは整えられていない。そのため、筆者らが要求する支援環境としては不十分であると考えられる。

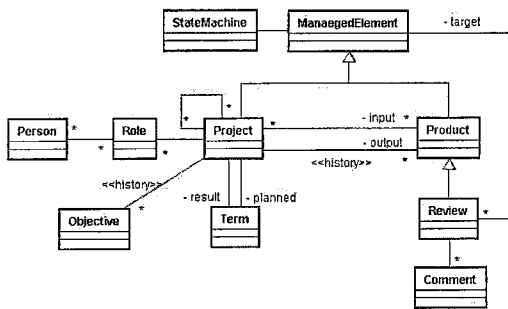


図 1 本研究で提案する概念モデル

## 5. プロジェクトと教育活動のモデリング

プロジェクトが何であるかを明らかにするためにプロジェクトの概念モデルを作成した。作成したモデルは汎用的なモデルであり、教育活動もプロジェクトである、と捉えることで教育活動についても適用することができた。本節では、プロジェクトの定義について述べ、作成したプロジェクトの概念モデルについて述べる。その後作成した概念モデルの教育活動への適用について述べる。

概念モデルの表記には UML のクラス図を用いる。また、多重度の表記がない場合は 1 とする。関連に表示されているステレオタイプ「history」は履歴管理されていることを表す。

### 5.1 プロジェクトの定義

PMBOK によれば、プロジェクトとは「独自のプロダクト、サービス、所産を創造するために実施される有期性の業務」である<sup>6)</sup>。筆者らはプロジェクトを限られたリソースで決められた期限までに目標とする成果を挙げることでありと捉えている。

### 5.2 プロジェクトの概念モデル

図 1 に示すモデルが本研究で提案する概念モデルである。モデルの各要素についてそれぞれ述べる。

#### 5.2.1 プロジェクト

プロジェクトを表現するのが Project である。Project は、後述する Person が Objective に沿って予定された Term の中で Product を作成する活動である。「設計」「実装」「テスト」「クラス設計」「DB設計」などが Project のインスタンスである。

プロジェクトは、プロジェクトの目標を達成し、成果物を作成するために、階層的に要素分解し、それぞれサブプロジェクトとすることができる<sup>6)</sup>。そ

れぞれのサブプロジェクトにも目標と成果物が存在し、下位レベルになるにつれ詳細で具体的な目標、成果物が設定される。提案する概念モデルでは、この特性を利用し、Project は親プロジェクトからサブプロジェクトへの参照を持つ再帰構造にした。

#### 5.2.2 成果物

プロジェクトを実施した結果得られる成果物を表現するのが Product である。Product は、プロジェクト活動の中で後述する Review を受けとり、作業によって更新されていく。複数の作業、レビュアーが協調作業を行う場合、成果物の履歴を記録しておかねばならない。よって、Product は履歴を保持するようにした。「クラス図」「テスト結果報告書」「ソースコード」「仕様書」などが Product のインスタンスである。

また、他の Project で作成された Product をリソースと見て Project の input 要素とすることができる。例えば、「進捗報告準備」という Project で作成した「進捗報告資料」という Product を「進捗報告」という Project に関連付けることができる。

#### 5.2.3 参加者と役割

プロジェクトに参加し何らかの活動を行うのが Person である。Person は参加するプロジェクトに対して役割を持っており Role として表現される。Role には、「PM」や「作業者」、「クライアント」、「コーディネータ」などが挙げられる。

本来、Person から ManagedElement や Comment に対して所有を表す関連が引かれるが、図が煩雑になるのを避け関連の線は引いていない。

#### 5.2.4 状態管理

プロジェクトと成果物を抽象化した概念を ManagedElement とし StateMachine を持たせた。StateMachine は状態機械のことである。状態機械とは、外部からのイベントを受け取りと現在の状態に応じたアクションを実行し、次の状態へと遷移するモデルのことである。このモデルを採用することによって、プロジェクトや成果物、レビューの状態を共有でき、作業の進捗状況やどのレビューを反映させるべきかといった情報を把握することができる。

#### 5.2.5 レビューとコメント

Project や Product を抽象化した ManagedElement に、Review をつけることができる。Review は複数関連付けることができ、それぞれの Review について Comment を用いて議論をすることがで

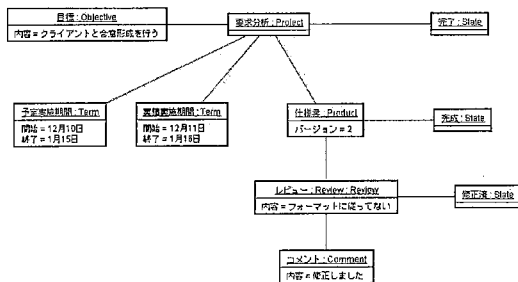


図 2 概念モデルの振る舞いの例

きる。

### 5.2.6 目標と実施期間

プロジェクトの目標を表現するのが Objective である。目標はプロジェクトの実行中に変更されていくものである<sup>6)</sup>。よって、Objective も履歴を保持するようにした。

プロジェクトの実施期間を表すのが Term である。実施期間には、予定の実施期間と実績の実施期間がある。

### 5.3 概念モデルの振る舞い

5.2 節で示したプロジェクトの概念モデルを「要求分析」プロジェクトを表現する。モデルを適用させたインスタンス図が図 2 である。インスタンス図では、図の煩雑さを回避するために StateMachine が管理している状態を State とし、直接 Project や Product から参照を引いた。

図 2 は以下のようなプロセスを経ている。

- (1) 要求分析プロジェクトを定義した
- (2) 仕様書を作成した
- (3) 仕様書に対してレビューをした
- (4) 仕様書を修正し、プロジェクトが完了した

### 5.4 教育活動への適用

筆者らは教育活動もプロジェクトの一つであると捉えている。教育活動でも、限られたリソース（人材）で一定期間内（授業期間内）で成果（教育成果）を上げなければならない。そのため、プロジェクトの概念モデルを教育活動の概念モデルにも利用できると考えた。

#### 5.4.1 コース

コースをプロジェクトの概念モデルを用いて表現したものが図 3 である。コースにはサブプロジェクトとして、授業がある。授業のサブプロジェクトには、進捗報告会や PM ミーティング、勉強会がある。

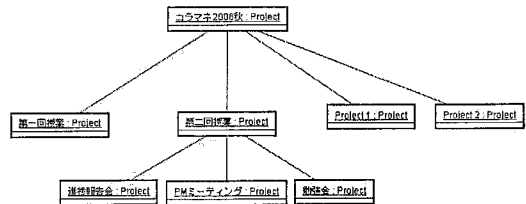


図 3 コース

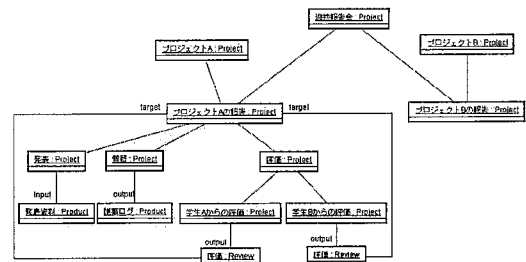


図 4 進捗報告会

また、コースには、コラボレイティブ・マネジメント型情報教育で実施されるソフトウェア開発プロジェクトもサブプロジェクトとしてある。

#### 5.4.2 進捗報告会

「進捗報告会」をプロジェクトの概念モデルを用いて表現したものが図 4 である。「進捗報告会」のサブプロジェクトには、各プロジェクトの「報告」がある。さらに、各プロジェクトのサブプロジェクトには、「発表」、「質疑」、「評価」がある。

「評価」は、さらに進捗報告会参加者 1 人 1 人の個人プロジェクトへと分解される。図 4 で示した「学生 A からの評価」が個人のプロジェクトに当たる。個人のプロジェクトには、成果物として Review 型の「評価」があり、発表したプロジェクトをレビュー対象として参照する。

#### 5.5 プロジェクトとコースのネットワーク構造

プロジェクトとコースは階層化を行うことが可能であるが、下位レベルのサブプロジェクトが他のプロジェクトと同一のサブプロジェクトになることがある。このことからプロジェクトのモデルは、木構造のモデルではなく、ネットワーク構造を持っていると言える。ここでは、進捗報告会を例にし説明する。

進捗報告会をさらにサブプロジェクトとして分解すると各プロジェクトの進捗報告で構成される。さらに各プロジェクトの進捗報告は、発表と質疑、プロジェクトへの評価によって構成される。

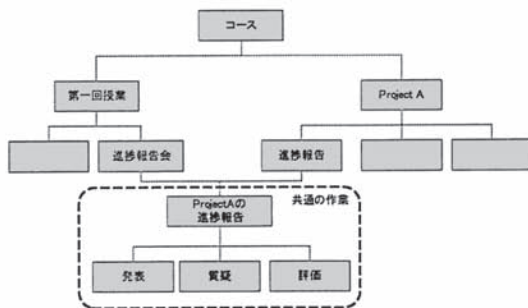


図 5 コースとプロジェクトのネットワーク構造

ここで、「進捗報告会」のサブプロジェクトである「ProjectA の進捗報告」は、「ProjectA」のサブプロジェクトとして捉えることもできる。

このような活動を図5のようなネットワーク構造を用いて表現する。こうすることで進捗報告がプロジェクト側から見てもコース側から見ても存在し、かつ同一のものとして扱うことができる。

### 5.6 同じ要素を持つプロジェクトの管理（プロジェクトテンプレートの導入）

コース管理には、一般的なプロジェクト管理システムとは異なり、同じ要素を持つプロジェクトの管理が必要である。同じ要素を持つプロジェクトの例として進捗報告会でのプロジェクトの報告を挙げられる。進捗報告会では、プロジェクト毎に報告を行う。この報告は、発表を行うプロジェクトが異なるだけで、そのプロセス、求められる成果の種類などは変わりはない。

また、発表に対して行う評価も個人ごとに行うプロジェクトと考えることができる。

これら同じ要素を持つプロジェクトを管理するためにプロジェクトテンプレートを導入する。プロジェクトテンプレートは、プロジェクトのメタな概念である。兄弟のようなプロジェクトは、プロジェクトテンプレートを使用して、「プロジェクトごと」「個人ごと」「指定された数ごと」に生成することができる。

### 5.7 PBL を支援するコース管理システムの構築

#### 5.7.1 システムの概要

5 節で述べたモデルを利用して PBL を支援するコース管理システム Collasys の実装を行った。Collasys は Jakarta Struts を利用した Web アプリケーションとして実装され、成果物などのデータの記録機構として Subversion を利用している。Subversion

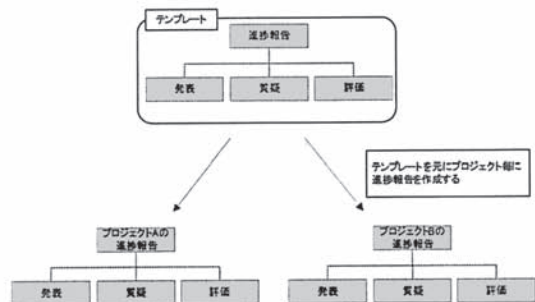


図 6 テンプレートを用いたプロジェクトの生成

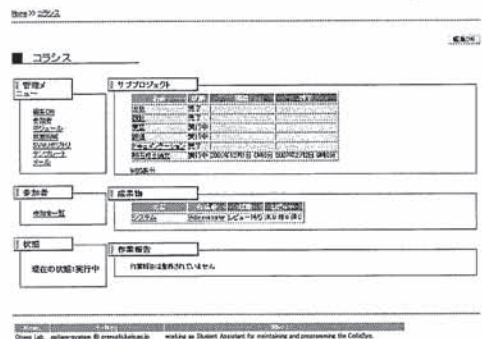


図 7 プロジェクト管理画面

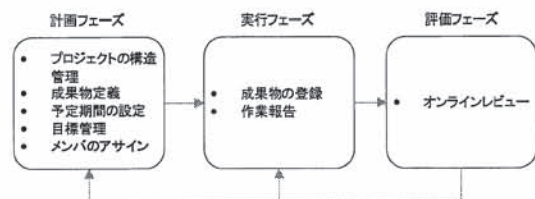


図 8 Collasys によるプロジェクト支援

を利用することで成果物のバージョン管理が実現される。

#### 5.7.2 プロジェクトの支援

本節では、Collasys のプロジェクトの支援機能について説明する。図 7 がプロジェクト管理画面である。

図 8 は、Collasys を利用したプロジェクト支援の流れを示している。

ここでは、5.3 節で説明したプロジェクトの流れに沿って説明する。

##### (1) プロジェクトの定義

「ソフトウェア開発プロジェクト」というプロジェクトのサブプロジェクトとして「要求分析」を作成

し、要求分析を行うメンバをアサインし、役割を設定する。目標や実施期間を入力する。成果物の名前を「仕様書」として設定する。

### (2) 成果物の作成

要求分析を行うメンバがプロジェクトの状態を「実行中」に変更し、要求分析プロジェクトを開始する。

1日の作業が終わったら Collasys 上から作業報告を行う。作業時間を入力し、作業内容を記述する。作業報告の内容は設定されたプロジェクトのメールアドレスへ送信される。

仕様書が出来上がったら、成果物画面から Web ブラウザを利用して Collasys に登録する。登録された成果物はバージョン管理される。

### (3) 成果物に対するレビュー

仕様書を Collasys からダウンロードし、レビューを行う。レビュー項目ごとに掲示板のスレッドが作成され、このスレッドを利用して、レビュアーと作成者でレビューや修正方針に関する議論を行う。また、レビューも状態管理されており、はじめは「未修正」となっている。

設定次第でレビューを追加することで作成者にメール通知をすることもできる。

### (4) 改善

作成者は、成果物のレビュー一覧を確認し、「未修正」のレビューについて対処していきコメントを書き込む。レビューに対処した成果物を Collasys に再度登録を行い、成果物情報を更新する。レビュアーは、レビュー内容がきちんと成果物に反映されているかを確認し、反映されていたらレビューの状態を「修正済み」と変更する。

レビュー・改善を反復して行い、成果物の品質が目標とする品質となったらプロジェクトの状態を完了に変更し、プロジェクトを完了させる。

## 5.8 コースの管理

本節では Collasys のコース管理機能について述べる。図 9 がコース管理画面のイメージである。プロジェクト管理画面とは異なり、画面の中央に授業ごとにブロックを配置した。コースを管理するコーディネータは授業のブロックから進捗報告会、課題といった活動を新規作成することができる。

コース管理の「コースの作成」「学習者の管理」「コースの状態管理」はプロジェクト管理機能で達成されている。ここでは、進捗報告会、課題の支援

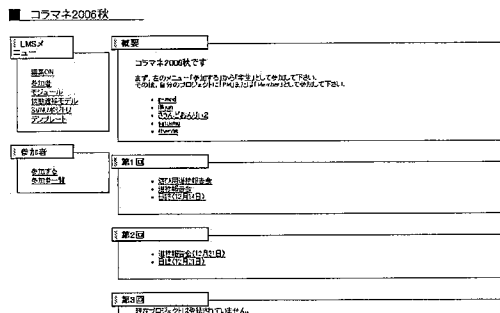


図 9 コース管理画面イメージ

について述べる。

### 5.8.1 進捗報告会

進捗報告会では、授業で行われる進捗報告会の進行管理とレビューの支援を行う。

#### (1) 発表順を決める

はじめに司会者が各プロジェクトの発表順を決める。発表順は、ランダムに並び替えを行うか、または、手動で並び替えを行うことができる。

#### (2) 発表/質疑の開始/終了の記録

発表が始まると司会者は発表開始ボタンを押す。このボタンを押すことで発表開始時間が記録される。発表が終わると発表終了ボタンを押す。このボタンを押すことで発表終了時間が記録される。質疑についても同様である。

#### (3) レビューの開始/終了

司会者は発表が開始されると「レビュー中にする」ボタンを押す。このボタンを押すことによって発表に対してレビューをすることができるようになる。「レビューを終了する」ボタンでレビューを締め切ることが出来る。

#### (4) レビューをする

設定されたレビュー項目に従いレビューを行うことができる。レビュー画面のイメージが図 10 である。記述形式のレビューの場合、テキストエリアにレビュー内容を記述し、項目のボタンを押す。例えば、不足と対案として「発表内容が構造化されておらずわかりにくい。きちんと発表の設計をした方がよい。」というレビューを行いたい場合、テキストエリアに「発表内容が構造化されておらずわかりにくい。きちんと発表の設計をした方がよい」と入力し、その後、「不足と対案」ボタンを押す。こうすることでレビューが登録される。また、発表を聞いていてさらにレビューを追加する場合も同様の操作を行う

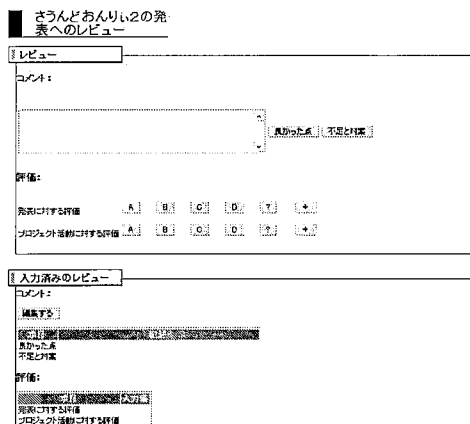


図 10 進捗報告会レビュー画面

ことでレビュー内容が追加される。レビュー内容は、編集可能であり、「入力済みレビュー」ブロックから編集を行うことができる。

選択形式の場合は、評価ボタンを押す。ボタンは何度も押すことができ、押すたびに評価が更新される。また、「+」ボタンを押すことで「B+」などの微調整を行うことができる。

#### (5) レビュー一覧を閲覧する

登録されたレビューの一覧を確認することができる。進捗報告会の司会者やコーディネータは、誰がどのようなレビューを行ったかわかるが、学生たちには評価者の名前は隠される。登録されたレビューは CSV 形式のファイルとしてダウンロードすることができる。

#### (6) プロジェクトと進捗報告へのリンク

進捗報告会でを行った進捗報告は、進捗報告を行ったプロジェクトにリンクされ、プロジェクト管理画面からプロジェクトの進捗報告プロジェクトの画面へと画面遷移することが可能である。

進捗報告プロジェクトの画面では、進捗報告会で得られた評価を閲覧することができる。また、得られた評価について議論と対処したかどうかをチェックできるように、成果物のレビューと同様にそれぞれの評価について状態管理ができるスレッド式の掲示板を用意した。

### 5.8.2 課題

授業の感想やアンケートや個人評価など設定された設問に対し学生が答える活動を支援する。課題には設問を複数設けることができ、回答形式も記述形

式、選択形式を選ぶことができる。

コーディネータやオブザーバは学生が入力した回答を一覧画面で確認することができる。

## 6. おわりに

本稿は、PBL を支援するコース管理システムの基礎となるプロジェクトの概念モデルとそのモデルを利用して実装したシステム Collasys について述べた。

本稿で提案した概念モデルを利用することで PBL のみならず一般的な演習課題や学生同士での相互評価を行う演習などの成果物を作成し評価を行う活動を実現することができると考えられる。

今後は、モデルを利用して実装した Collasys の本格的な運用と勉強会や PM ミーティングといった活動の支援を行う予定である。

## 参考文献

- 1) 松澤芳昭, 大岩元. コラボレイティブ・マネジメント型情報教育によるソフトウェア技術者の育成と評価方法の提案. 情報教育シンポジウム, Vol. 2006, No. 8, pp. 247-252, 2006.
- 2) 佐藤 匡正. ソフトウェア・デザイン・レビューの実践技法? その効果的進め方と応用. ソフトリサーチセンター, 1992.
- 3) 梶田将司. コース管理システムの発展と我が国の高等教育機関への波及. メディア教育研究, Vol. 1, No. 1, 2004.
- 4) Batatia, Hadj. A model for an innovative project-based learning management system for engineering education. *Computer-aided learning in Engineering Education, CALIE2001, Tunis, 08/11/01-10/11/01*, 2001.
- 5) 西森年寿, 加藤浩, 望月俊男, 八重樫文, 久松慎一, 尾澤重知. 高等教育におけるグループ課題探求型学習活動を支援するシステムの開発と実践. 日本教育工学会論文誌, Vol. 29, No. 3, pp. 289-297, 2005.
- 6) Project Management Institute. プロジェクトマネジメント知識体系ガイド. Project Management Inst, 2004.