

大学における演習・実験管理支援システムの開発 Development of University Practical/Experiment Management Support System

百井 美由紀†
Miyuki Momoi

早川 栄一‡
Eiichi Hayakawa

1. はじめに

インターネットが普及し、学生の PC 所有率が増加している。また、Web 上で学習できるサイトが増え、学生の勉強方法も PC を用いた学習へと変わってきた。また、大学における授業情報をインターネット上で公開^[1]するなど、Web 上で講義のサポートする機会が増えている。

大学の理系学科には、実際に自分の手を動かして行う演習や実験という科目^[2]がある。たいていの場合、複数の教授者や TA (Teaching Assistant) がいる。本学情報工学科においても、大学院生から有志を募り、TA として演習実験のサポートを行っている。

ここで、TA として講義室に行くとその日は試験だったなどのように、コミュニケーション不足のために連絡事項の伝達ができているという問題が生じた。関連研究に、複数の教員と職員が同じスペースに在籍できる環境を構築し、教職員間で活発なコミュニケーションが図れるシステム^[3]がある。しかし、問題は事前にすべてを把握できるわけではなく、講義中に突然生じるものがあるので、対応しきれないことがある。

そこで本研究では、演習や実験の進行を円滑に進めるための支援を行うことを目的としている。また、例に挙げたような現在起きている問題点の解決を目指す。

2. 問題分析

本システムを作成しはじめるにあたり、TA 経験者 6 人にアンケートをとった。その結果に基づき、全体としてサポートする部分について問題点と共に述べる。

アンケートの結果、特性によって問題を大きく三種類に分けることができる。そこで、本システムでは、次の 3 点に視点を置いたサポートを行っていくことにする。

(1) サイト構築上の問題

- ・ Web にデータを公開することに手間がかかる。
- ・ 必ずしも、十分な情報が公開されていない。

コンテンツの作成・管理、公開をサポートすることで、サイト構築を支援する。

(2) 定期的な問題

- ・ 学生の出席状況や提出済課題のチェックをしたい。
- ・ 誰が理解していないのかを把握しきれない。

出席状況の把握などのように講義を行うたびに起きる処理や課題の提出・受付管理、学生の理解状況を把握するための問題を解決する。

(3) 突発的な問題

- ・ 実験室に行くまで、その日が試験だと知らなかった。
- ・ 講義中に起きる、学生の PC トラブルをサーバ管理者以外でも解決できるような機能がほしい。

教授者・TA 側のコミュニケーション不足や PC トラブルを解決する。これにより、予期せずに起きた講義の進行を遮る問題を解決する。

3. 設計方針

演習や実験の内容や進行方法によって、欲しい機能が違う。そこで、本システムの方針は、それぞれの演習や実験に合わせたシステムを提供することである。

本システムの特徴は次のとおりである。

- (1) CMS (Contents Management System) を用いてコンテンツ収集、作成、管理、配信を一貫してサポートすることでユーザ側からコンテンツ配信を管理することができる。
- (2) クラスタライブラリを提供することで、演習・実験に合わせたモジュールの作成が容易にできる。
- (3) 課題の提出および受取り、採点、結果の公開までの一連の流れを一括してすべて Web 上で行うことができる。
- (4) 事前に定型的な処理をデータベースに登録することで、サーバにアクセスしてシェルを開くといった操作なしに、コマンドやスクリプトを実行できる。

4. システムの設計

次に、本システムの設計について述べる。

4.1 全体構成図

図 1 に本システムの全体構成図を示す。

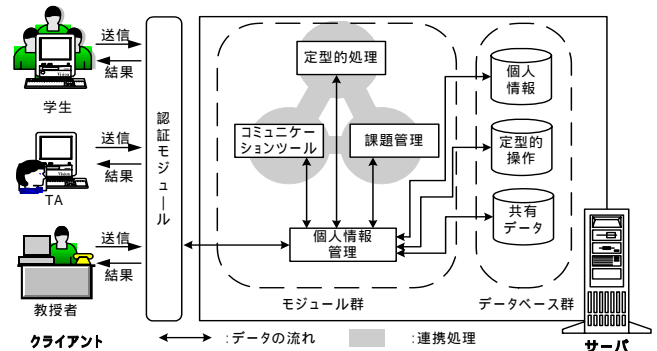


図 1. 全体構成図

クライアントからのアクセスに応じて、認証モジュールが階層の識別を行う。モジュールは、演習・実験の内容によって、適宜組換え可能である。モジュール同士は連携して動いており、データベースへアクセスする時は、個人情報管理モジュールを介して操作する。これにより、データベース内の操作を制限し、データを保護する。

4.2 フレームワーク

支援内容の大きな枠組みとして、次の四つを提供する。各枠組みは機能ごとに細分化し、クラスタライブラリとして提供することにより、管理者は演習や実験の内容に合わせたコンテンツを作成することができる。

† 拓殖大学大学院工学研究科, Graduate School of Engineering, Takushoku University

‡ 拓殖大学工学部, Takushoku University

4.2.1 コミュニケーションツール

演習や実験に関わる人のコミュニケーション不足をサポートする．提供する機能を次に挙げる．

- (1) データベースアクセス（登録・取得）
ID、権限、メッセージ配信先アドレス、
メッセージ配信予定日、メッセージ開封済の人のリスト
- (2) 掲示板のように書き込めるボード
- (3) E-mail の送信
- (4) メッセージ配信先の選択
- (5) 相手がメッセージを読んだことを確認
- (6) コンテンツ更新検知

例えば、E-mail や掲示板のように時間を共有することなく、個々のタイミングに合わせたメッセージの送信や閲覧ができる．また相手に確実に情報を伝えるために、書き込んだ内容を E-mail で通達することもできる．

4.2.2 定型的処理

定型的処理とは、同じ操作を繰り返す必要がある操作のことである．講義を行う上で必要な定型的操作をサポートする．提供する機能を次に挙げる．

- (1) データベースアクセス（登録・取得）
ID、権限、コマンドライン、共有ファイルの詳細、
ログインした学生、アクセスログ
- (2) ログインした座席の位置取得、表示
- (3) コマンド選択画面表示
- (4) 一覧のリスト表示
- (5) アクセスログの取得
- (6) WebDAV（ファイル転送用）

例えば、ログインした学生の ID と座席の位置を取得すれば、誰がどこに座っているかを画面に表示することができる．図 2 に出席管理モジュールの構成図を示す．

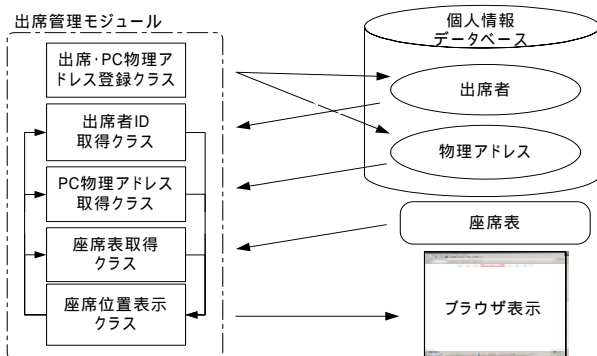


図 2. 出席管理モジュール

ログインするときに、ID 情報とログインした PC の物理アドレスを個人情報データベースに登録する．座席表には、あらかじめ PC の物理アドレスを登録しておくものとする．

管理者が座席位置表示を行うとき、データベースから出席者の ID とその学生が座っている PC のアドレスを取得する．座席位置表示クラスは、座席表の物理アドレスと取得した物理アドレスを比較し、一致した場合だけ ID を当てはめて表示する．

4.2.3 課題管理

課題の受付、提出、採点、結果までの一連の流れをサポートする．提供する機能を次に挙げる．

- (1) データベースアクセス（登録・取得）
ID・権限、コマンドライン、課題提出完了日時、
提出受付開始と終了時刻の予約、出席者リスト、

課題提出済リスト、採点結果の登録

- (2) 提出受付対象者の制限
- (3) 出席者リストと比較
- (4) 指定 ID のフォルダ情報の取得、表示
- (5) 課題提出済リストの表示
- (6) 課題ファイルの内容表示
- (7) 採点結果の表示

例えば、課題の受け付けやブラウザ上で課題の採点を行うこともできる．結果をデータベースに登録することで、グラフ作成をサポートし、Web に公開することができる．

4.2.4 個人情報の管理

個人情報は、すべてのモジュールで共有するデータであり、サーバ上にデータベースとして保持する．いろいろなモジュールから直接アクセスすると、モジュールを組み替えたときに変更箇所が増え、処理が複雑になる．

そこで、個人情報データベースにアクセスするためのモジュールを用意し、アクセス制限を行う．

4.3 データベース

用途によって分けた、三種類のデータベースをサーバ上で管理する．これにより、モジュール間のデータ共有をスムーズにする．データベースへのアクセスが必要な操作は、アクセス権限を持ったユーザだけが可能である．

4.4 モジュール間提携

4.2 で述べた枠組みは、機能ごとにモジュール化する．なかには、他のモジュールに従属しているために、単独で使えないものがある．よって、どのモジュール間でデータを共有しているのかをデータベースで管理する．どのモジュール間でデータの共有を行っているかがすぐにわかるので、演習や実験の内容に応じた組合せが自由にできる．

5. 開発環境

開発環境を決めるにあたり次の三点について検討した．

- (1) 既存の CMS を使用
- (2) データベースアクセス
- (3) モジュール化のしやすさ
- (4) 既存の HTTP サーバとの調和性

これらを満たすものとして、ベースとなる CMS ポータルツールは XOOPS^[4] を使用し、プラグインの開発には XOOPS のベース言語である PHP を用いることにした．

6. おわりに

本研究では、大学における演習・実験管理支援システムの開発を行っている．これまでに TA 経験者を対象にアンケートを行い、問題点の把握とそれに沿った設計を行った．

現在は、自作したクラスライブラリを用いて、モジュールのプロトタイプを作成している．今後は、クラスライブラリをより充実させる予定である．

参考文献

- [1] Massachusetts Institute of Technology : MIT Open Course Ware , <http://ocw.mit.edu/OcwWeb/index.htm>
- [2] 拓殖大学工学部講義要綱：拓殖大学 2004 年度版
- [3] 福田崇之：を旨とした学生参加型の教育・修学支援システム，情報処理学会「情報処理」2002 Vol.43 No.7
- [4] The XOOPS Project : <http://jp.xoops.org/>