

令和 2 年度 創造工学科 情報科学・工学系 卒業研究論文

研究テーマ

情報系科目向け課題作成・回答システムの開発

研究者：創造工学科 情報科学・工学系 5 年 16 番 高橋 雄太

提出日：2021 年 1 月 22 日（金）

研究期間：2020 年 5 月 12 日（火）～2021 年 1 月 28 日（木）

指導教員：三上 剛

目次

1. 緒言

1.1. 研究の目的

情報系科目向け課題作成・回答システムの開発を通して、Web アプリケーション開発に必要な技術と知識、リレーショナルデータベースの運用方法を習得するとともに、想定ユーザに向けた適切な機能を持つアプリケーションを作成することを目的とする。

1.2. 研究の意義

教育機関では今日の COVID-19（新型コロナウイルス感染症）の流行により、遠隔授業の実施が激増した。2020 年 5 月時点では、国立大学で 91%、国立高等専門学校で 98%が全面的な遠隔授業を実施した^[1]。

全面遠隔授業の割合は徐々に減少し、それに代わり後期授業では対面と遠隔を組み合わせた授業の実施割合が増加している。2020 年 9 月時点での調査では、国立大学で 96.5%、国立高等専門学校で 43%が対面と遠隔を併用した授業方針を固めている^[1]。

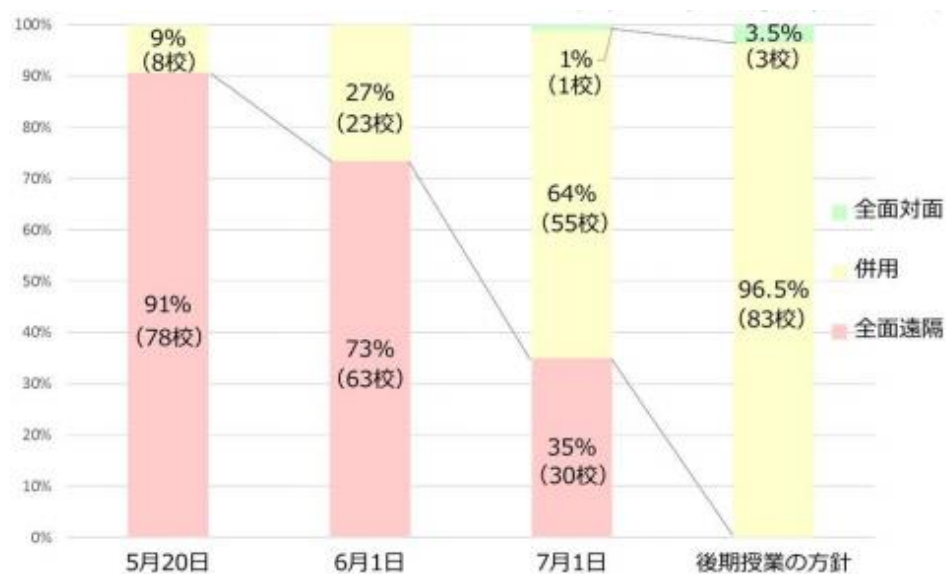


図 1.1. 国立大学の授業形態の推移（令和 2 年 9 月時点 文部科学省調査）^[1]



図 1.2. 国立高等専門学校の授業形態の推移（令和 2 年 9 月時点 文部科学省調査）^[1]

遠隔授業の実施に伴い、レポートや課題を遠隔で課す事例も増加した。本校では、学生の理解度の把握を目的として、毎回の授業で「簡単な」小テストまたは課題を実施した^[1]。本校の課題では、学習管理システム（LMS）の Bkackbord や、Microsoft の Forms などのシステムを利用していた。しかしこれらのシステムは、簡単な課題やアンケートの実施には適しているが、専門科目の課題の出題・回答に最適であるとは言い難く、科目や設問内容によっては教員による課題の作成、学生の回答に時間や技術を要するといった問題点が存在する。

本研究は、専門科目の課題の作成・回答に特化した LMS を有する e ラーニングシステムを開発することで、専門科目におけるオンライン課題の円滑な実施をサポートすることを目的としている。

2. 原理

2.1. e ラーニング

2.1.1. 概要

e ラーニング (e-Learning、イーラーニング) は、主にインターネットと電子機器 (PC、モバイル端末など) を利用した学習形態のことを指す。e ラーニングの"e"は、electronic (電子的な) の意味であり、日本語においてもアルファベットのままの表記が多い。

2.1.2. e ラーニングシステムの概要

e ラーニングシステムとは、e ラーニングを実施するための情報システムであり、おおまかには、「教材・学習材」と「学習管理システム (LMS)」から構成されている。なお、「e ラーニング」という用語単独でも、e ラーニングを実施するための情報システムを意味することもある。

e ラーニングシステムの利用者には、学習を行う者 (以下、「学習者」という) と学習をさせる者 (以下、「教師」という) が想定されており、教師用の機能と、学習者用の機能は異なる。また、多くの e ラーニングシステムには、e ラーニングシステムの「システム管理者」 (システムアドミニストレータ) が存在し、システム管理者によって、学習活動・教育活動に対する支援が行われる場合もある。

「教材・学習材」について、e ラーニングシステムの教材・学習材は、静止画や動画の映像、音声、文章などを組み合わせたマルチメディア形態のことが多い。これらは主に資料提示型の教材として活用されている。また、学習した内容を逐次確認していく「小テスト」、問題演習を行うことのできる「ドリル」、さらにそのまとめとしての「試験問題」なども教材・学習材にあげられる。教材・学習材を準備する方法として、e ラーニングシステムの提供会社が用意しているものを利用する方法や提供会社に制作を依頼する方法、教師が自ら教材を制作する方法などが挙げられる。

「学習管理システム (LMS)」について、e ラーニングシステムにおける学習管理システムは、教材・学習材の保管・蓄積、学習者への教材・学習材の適切な配信、学習者の学習履歴や小テスト・ドリル・試験問題の成績などを統合的に管理するものである。一般に LMS (Learning Management System) と呼ばれる。

2.1.3. 利点と欠点

e ラーニングを活用する主な利点と欠点を、学習者側、教師側に分けて以下に示す。

利点

● 学習者側

- 自由な時間・場所で学習に取り組むことができる
- 個々人の習熟度に合わせて学習を進めることができる

- 進捗状況やテスト結果などのフィードバックを即座に確認することができる
- 目的に応じた均一化（標準化）された授業を受けることができる

- 教師側

- 学習者と同時間、同一場所にいる必要がなく、効率的に業務が遂行できる
- クラスごとに同じ授業を繰り返し行う必要がない
- 印刷教材の量を減らすことができる
- 成績管理などの自動化が図れる
- 最新の教材を全学習者に一律に提供できる

欠点

- 学習者側

- 学習意欲の継続が難しい
- 質疑などその場での問題解決がしにくい
- 教師やほかの学習者との交流がとりにくい
- デバイスによっては学習画面や入力方法が制限される

- 教師側

- 学習者の状況をデータからしか把握できない
- 教材・学習材の作成の工数が大きく、コストがかかる
- 教材・学習材の作成技術が必要となる場合がある
- イニシャルコストがかかる

2.2. LAMP

LAMP とは、データベースを利用した Web アプリケーションを開発・運用する際のプラットフォームとなるオープンソースソフトウェアの組み合わせの略称である。具体的には OS の「Linux」、Web サーバの「Apache」、データベースの「MySQL」（または MariaDB）、そしてプログラミング言語の「PHP」、「Perl」、「Python」（のいずれか）を指し、LAMP とはこれらの頭文字を取ったものである。

いずれのソフトウェアも、データベース連動型で動的なコンテンツを含む Web アプリケーションの開発に適しており、かつ各分野でも人気の高いソフトウェアであるという共通点がある。そのため、Web アプリケーションのためのソフトウェア環境を構築するための定番的組み合わせとして多く利用されており、関連情報や事例も豊富である。

これらのソフトウェア群は、いずれもオープンソースソフトウェアであるためコストがかからないか、かかっても極めて低コストであるため、開発・運用にかかる費用を軽減することができるとともに、高いカスタマイズ性を活かせるというメリットがある。半面、保証、サポートがないなど、運用管理面でのリスクが大きく、開発者、運用管理者には相応のスキルが必要となる^[4]。

2.3. WSL

WSL とは「Windows Subsystem for Linux」の略であり、Windows で Linux を利用する仕組みである。このシステムを利用すると、従来の仮想マシンやデュアルブートセットアップのオーバーヘッドなしで、ほとんどのコマンドラインツール、ユーティリティ、アプリケーションなどを含む GNU/Linux 環境を Windows 上で直接実行することができる。

Windows OS 内で実行するアプリケーションであるため起動も早く、ファイルなどのデータ連携を直接行うこともできる、Microsoft が提供しているため Windows との相性が良く導入が容易であるといった特徴が挙げられる。

2021 年 1 月現在、WSL1 と WSL2 の 2 バージョンのアーキテクチャが存在し、各ディストリビューションはいつでもアップグレードまたはダウングレードできる。また、両ディストリビューションを並行して実行することも可能である。WSL1 と WSL2 の違いについて、WSL1 は Linux カーネルをエミュレーションしたものが存在していたのに対し、WSL2 には完全な Linux カーネルが存在している、ファイル IO パフォーマンスが向上している、起動時のメモリ使用量が少ないなどといった違いがある。

2.4. Apache HTTP Server

Apache HTTP Server (アパッチ エイチティーティーピー サーバ、以下「Apache」という。)とは、世界中で使用されている Web サーバソフトウェアの一種である。1995 年に初版がリリースされ、1996 年 4 月以来、最も人気のあるサーバであり続けている。現在も、全世界で 35.0%とトップのシェアを占め (2020 年 1 月 11 日時点) [6]、多くの人に愛用されている。

Apache の長所として、オープンソースソフトウェアであるため、無償で誰でも使うことができる、複数の OS に対応している、長い実績と繰り返してきたバージョンアップによる信頼性・安定性などがある。その反面、無保証であること、同時処理が多くなるほどメモリ消費が高くなってしまふといった短所も持ち合わせている。

2.5. MySQL

MySQL (マイ エスキューエル) とは、オープンソースソフトウェアのリレーショナルデータベース管理システム (RDBMS) である。DB-Engines によれば、Oracle に次いで 2 番目に、オープンソースソフトウェアでは 1 番人気の高いデータベース管理システムである (2020 年 1 月時点) [7]。また MySQL は、YouTube、Facebook、Twitter など、多くの人気 Web サイトでも使用されている。

MySQL の特徴として、高性能で大規模アプリケーションの運用に適している、Linux、Windows、Mac OS など複数の OS で利用可能である、オープンソースであるため低コストで運用できるなどといった点がある。

また、MySQL 専用の管理ツールが存在しており、より簡単に操作できる。具体例として、Web ブラウザ上で機能するもので phpMyAdmin や MySQL Workbench などが存在している。

2.5.1. phpMyAdmin

phpMyAdmin とは、MySQL サーバを Web ブラウザ上で管理するためのデータベース接続クライアントツールの一種である。PHP で実装されている。phpMyAdmin を用いることで、SQL 文を記述することなく、MySQL のデータベースに対して、テーブルの作成や削除、データの読み書きなどの様々な操作が行える。また、ユーザが任意の SQL 文を記述して実行することもできる。

特徴として、直感的な Web インターフェイスによる操作でほとんどの MySQL 機能をサポートしている点や複数のサーバを同時に管理できる点、様々なフォーマットにデータをインポート・エクスポートできる点、クエリを使用することで複雑な問い合わせを作成することができる点などが挙げられる。

phpMyAdmin を利用するには、MySQL が動作するサーバに Web サーバソフトウェアと PHP を導入する必要がある。

セキュリティ面について、phpMyAdmin は Web サーバに置くため、容易にアクセスされてしまう危険性がある。また、この分野では有名な管理ツールで広く普及しているため、外部の攻撃者の対象とされやすい。そのため、運用の際には、最新のバージョンに更新、厳重なパスワード管理、IPS や Web アプリケーションファイアーウォールを設置する等の対策が必要である。

2.6. PHP

PHP（ピーエイチピー、Hypertext Preprocessor を再帰的に略したもの）は The PHP Group によって開発されているオープンソースの汎用スクリプト言語である。PHP は HTML に埋め込むことができ、Web 開発に適している。簡単な例を以下に示す。

リスト 1. 初歩的な HTML 埋め込み型構文の例

```
<!DOCTYPE html>
<html>
  <head>
    <title>Example</title>
  </head>
  <body>

    <?php
```



```
        echo "Hi, I'm a PHP script!";
    ?>

</body>
</html>
```

PHP タグ `<?php ?>` で囲まれている箇所が PHP の処理系によって実行される。リスト 1 では `body` タグ内に「Hi, I'm a PHP script!」という文字列が出力される。

PHP とクライアントサイド JavaScript のようなものの大きな違いは、そのコードがどこで実行されるかにある。JavaScript は、クライアントサイドがコードを受け取り、そこで動作する。従ってクライアントは、動作しているコードの内容を確認することができる。一方、PHP はサーバーサイドでコードを実行し、クライアントサイドはその結果のみを受け取る。そのためクライアントは、結果を出力するコードがどんなものかを確認することはできない。

PHP を用いると、動的な Web サイトを作成することができる。PHP でよく使用される機能を以下に示す。

- HTML ソースコードを生成する
 - リスト 1 の例のように、HTML ソースコードを生成することができる。すべての文章を PHP で書く必要はなく、HTML の一部分のみ PHP で生成することができる。
- データベースと連携する
 - MySQL や PostgreSQL をはじめとした数多くのデータベースを標準でサポートしており、データベースと連携した Web サイトを簡単に作成することができる。
- クライアントから送信された情報を受け取る
 - HTML のフォームでユーザが入力した情報を PHP を使って受け取ることができる。受け取った情報を HTML ソースに表示したり、データベースに保存したりすることもできる。

2.7. JavaScript

JavaScript は、クロスプラットフォームで動作するオブジェクト指向のスクリプト言語であり、Web ページをインタラクティブにするために使用される（複雑なアニメーション、押下可能なボタン、ポップアップメニューなどを設けるなど）。Node.js のようなサーバ側アプリケーションを作るものにも利用されている。JavaScript には、Array、Date、そして Math といったオブジェクトからなる基本的なライブラリ、そして演算子、制御構造、文といったプログラミング言語の要素からなる主要な機能が含まれている。JavaScript の機能は、

追加のオブジェクトを補うことで、様々な目的に合わせて拡張することができる。^[8]

JavaScript に備わる機能を大まかに 3 つに分けて以下に示す。

- Web サイトに動きを付ける
 - アニメーションを追加することで動きのある Web サイトを実現できる。
- ユーザ操作で内部処理を発生させる
 - ユーザがブラウザでおこなった操作（マウス移動、フォーム入力値のリアルタイム検知、クリック、ダブルクリック、ホバー時など）をトリガーに、さまざまな処理を発動させることができる。
- 非同期通信
 - Web サイトを更新することなく表示内容を更新することができる。

JavaScript は Application Programming Interface (API) と呼ばれる機能を利用することにより、より多くの機能を実現できる。API は大きく 2 種類に分けられ、Web ブラウザに組み込まれたブラウザ API と、サードパーティ（通常は Facebook、Twitter、Google などの企業）が提供しているサードパーティ API が存在する。

また、ライブラリやフレームワークを利用することによって、開発スピードを上げ、便利な機能をより短い機能で実装することができる。

2.7.1. jQuery

jQuery（ジェイクエリー）とは、ウェブブラウザ用の JavaScript コードをより容易に記述できるようにするために設計された JavaScript ライブラリである。

jQuery の特徴として、DOM（マークアップがなされたリソース（Document）をリソース要素（Object）の木構造（Model）で表現し操作可能にする仕組み、またそのモデル）や CSS の操作・変更、イベント処理、Ajax（サーバとの非同期通信）などを容易に実現できるといった点がある。また、多数のブラウザで機能する、軽量で高速であるなどといった点も挙げられる。^[9]

2.7.2. iinkJS

iinkJS は、手書き認識を実現するための JavaScript ライブラリである。主にテキスト認識と数式認識がサポートされており、65 種類の言語、200 種類以上の数学記号に対応している。またコンテンツのインポートとエクスポートにも対応している。認識と変換はサーバ上で処理されるため、高速なインターネット接続が必要である。

MyScript が公式で公開しているデモページのスクリーンショットを図 2.1、図 2.2 に示す。

math

webdemo.myscript.com

Math

Math LaTeX MathML

$x^{2}+\left(y-\sqrt[3]{x^{2}}\right)^{2}=1$

Convert

$x^{2}+\left(y-\sqrt[3]{x^{2}}\right)^{2}=1$

MyScript.

図 2.1. MyScript の Web デモページ (Math)



図 2.2. MyScript の Web デモページ (Write)

API を使用するには、MyScript Cloud に登録しアプリケーションキーと hmac キーを取得する必要がある。MyScript Cloud へのアクセスは、月間リクエスト回数が 2,000 件までは無料で許可されるが、それ以上はリクエスト回数に応じた料金請求が発生する。

iinkJS の処理の流れについて、iinkJS のアーキテクチャ図を図 2.3 に示す。

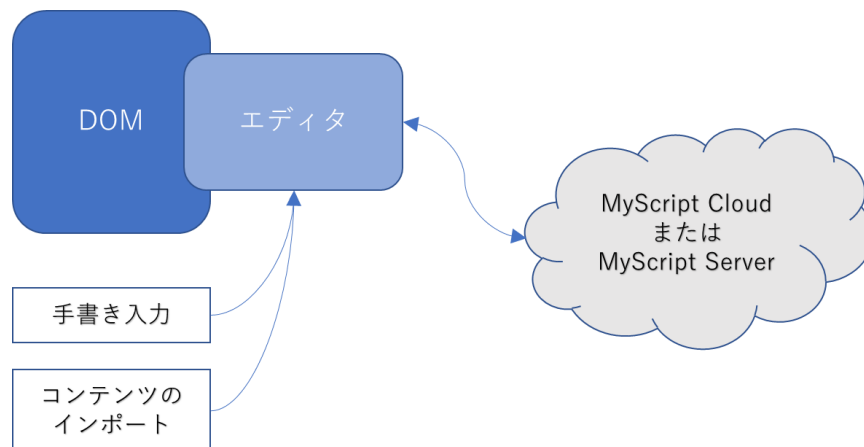


図 2.3. iinkJS のアーキテクチャ図

DOM 要素にエディタと呼ばれる要素を接続し、そのエディタ内でユーザの入力をキャプチャし、描画する。入力がキャプチャされると、iinkJS はサーバ（MyScript Cloud または MyScript Server）との通信を処理する。

2.7.3. Ace

Ace とは JavaScript で記述された埋め込み可能なコードエディタである。Ace はオープンソースであり、API を利用することで Web ページや JavaScript アプリケーションに簡単に埋め込むことができる。

Ace のエディタとしての特徴は、ネイティブなテキストエディタに匹敵する機能とパフォーマンスである。具体的な機能の一部を以下に示す。

- 110 種類以上の言語のシンタックスハイライト（構文の強調表示）
- 20 種類以上のテーマ（TextMate、Monokai など）
- 自動インデント、アウトデント
- オプションのコマンドライン
- vim および Emacs モードを含む完全にカスタマイズ可能なキーバインド
- 選択した単語や一致する括弧の強調表示
- 検索し正規表現で置換
- 行の折り返し、コードの折り畳み
- ライブ構文チェック

3. 開発環境

3.1. ハードウェア開発環境

本研究で利用したハードウェアの開発環境を以下に示す。

(1) ラップトップ PC

- ・メーカー : Dell
- ・型名 : Vostro 14-3468
- ・プロセッサ : Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz 2.70GHz
- ・実装 RAM : 8.00GB
- ・OS : Windows 10 Home

(2) iPad

- ・メーカー : Apple
- ・型名 : iPad Air (第 4 世代)
- ・CPU : Apple A14 Bionic
- ・ストレージ : 64GB
- ・OS : iPadOS 14

(3) Apple Pencil

- ・メーカー : Apple
- ・型名 : Apple Pencil (第 2 世代)

3.2. ソフトウェア開発環境

本研究で利用したソフトウェアの開発環境を以下に示す。

(1) WSL

- ・バージョン : 1
- ・Linux ディストリビューション : Ubuntu 18.04 LTS

(2) Apache

- ・バージョン : 2.4.29

(3) MySQL

- ・バージョン : 5.7.30

(4) PHP

- ・バージョン : 7.2.24

(5) phpMyAdmin

・バージョン : 4.6.6deb5

(6) Chrome

・バージョン : 87.0.4280.141 (Official Build) (64 ビット)

4. 研究概要

4.1. システムの概要

本システムを開発するにあたって、高等専門学校の情報系学科での利用を想定した。従って、システムのユーザは「教員」と「学生」に分けられる。ログイン画面は両ユーザ共通で、ログイン成功後にそれぞれのユーザに沿った初期画面を表示する。

ユーザが「教員」の場合、課題作成画面で科目と課題のタイトルを決定すると、設問作成画面に遷移する。設問作成画面で課題作成の完了を選択すると、データベースに作成した課題のデータが保存され、作成完了画面に遷移する。作成完了画面では、教員用初期画面に戻ることができる。

ユーザが「学生」の場合、課題選択画面で取り組む課題を選択し、遷移先の回答画面から課題の回答を行うことができる。課題の回答が完了した場合、データベースに回答済みのデータが保存され、回答完了画面に遷移する。回答完了画面では、学生用初期画面に戻ることができる。

作成したシステムの概念図を図 4.1 に示す。

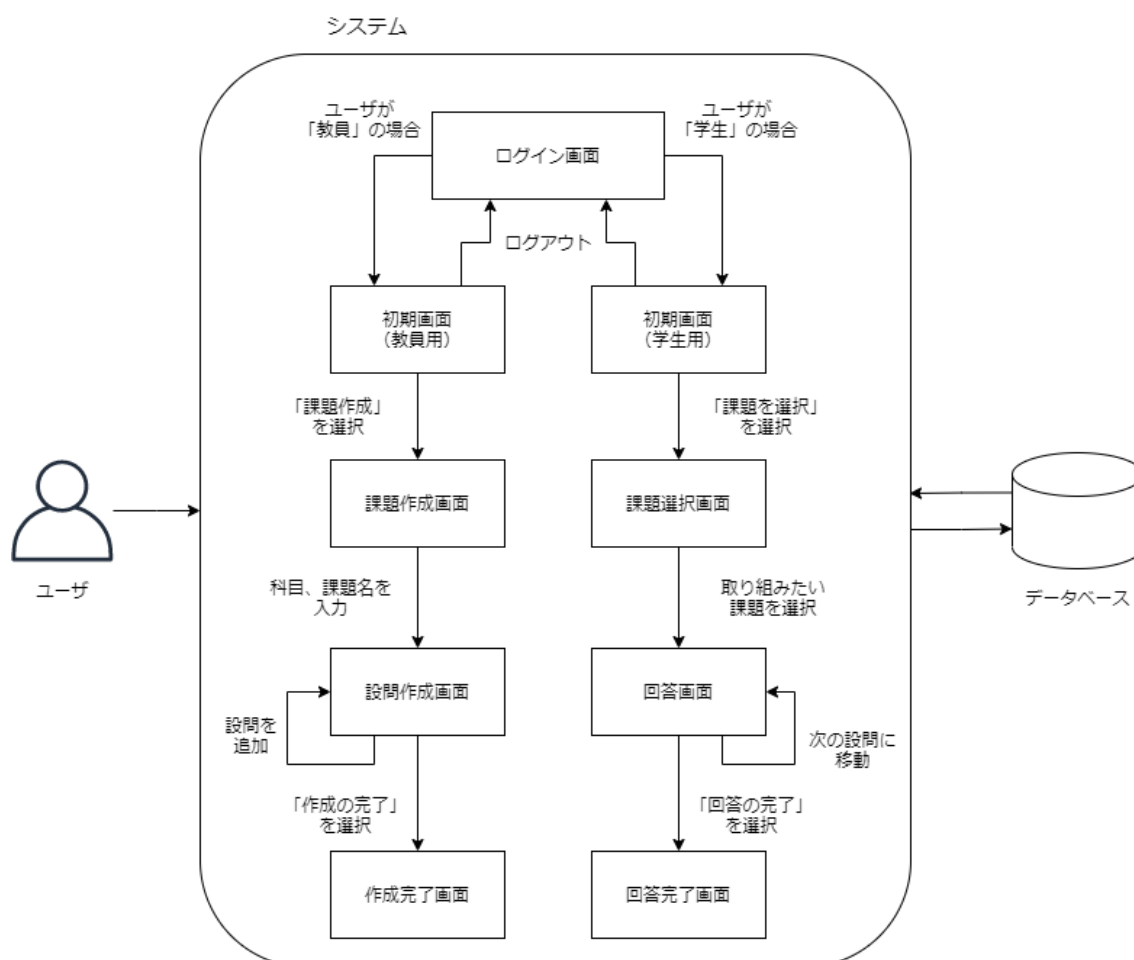


図 4.1. システムの概念図

4.2. ファイルの構成

作成したファイルの構成を以下の表 4.1、表 4.2、表 4.3 に示す。

表 4.1. HTML ファイル

ファイル名	ディレクトリ	説明

表 4.2. PHP ファイル

ファイル名	ディレクトリ	説明

表 4.3. JavaScript ファイル

ファイル名	ディレクトリ	説明

5. 研究結果

5.1. 作成した画面及び機能

5.1.1. ログイン画面

ログイン画面とは、ユーザのログイン処理を行う画面である。
作成したログイン画面のスクリーンショットを図 5.1 に示す。

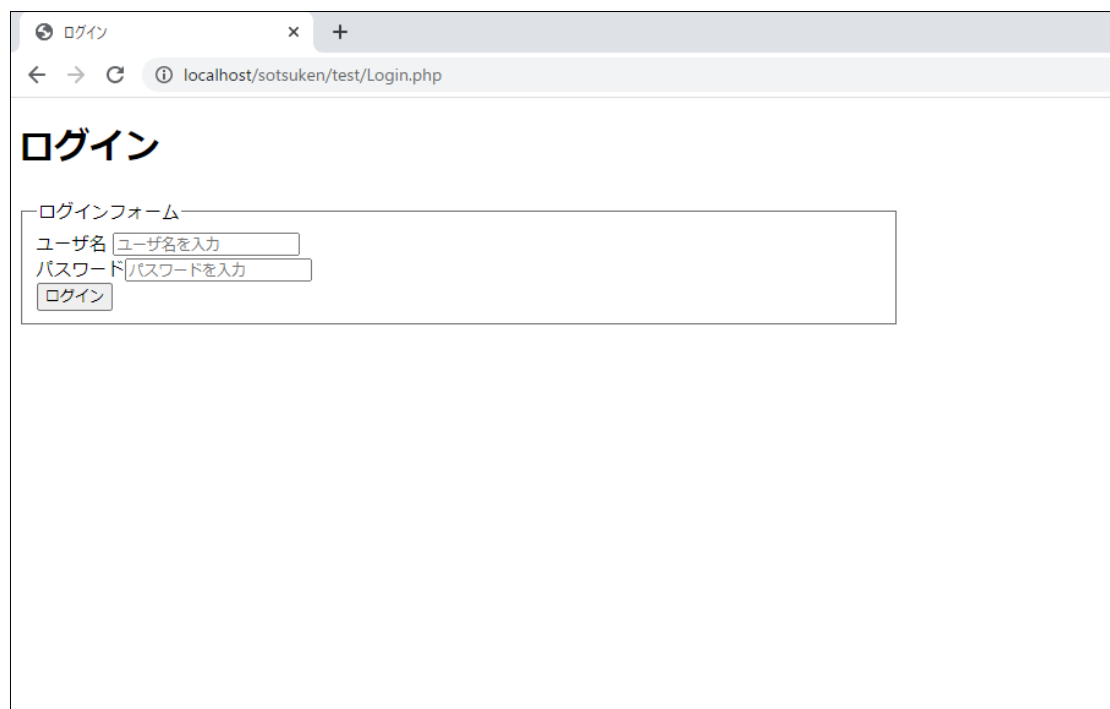


図 5.1. ログイン画面

この画面は全ユーザ共通で、ユーザ名とパスワードの入力フォーム、ログインボタンが設けられている。

正しいユーザ名とパスワードを入力しログインボタンを押下すると、データベースからそのユーザが「教員」、「学生」のどちらであるかを検索し、それに対応した初期画面に遷移する。このとき、データベースから取得したユーザ情報が PHP のセッション変数に格納される。

ユーザ名、パスワード、またはその両方が空欄や誤りであった場合、このログイン画面に遷移する。（つまり、見た目上は画面遷移が起こらない。）

データベースに保存されているパスワードは、PHP の password_hash 関数によりハッシュ化されているため、password_verify 関数を用いて、この画面で入力したパスワードがハッシュにマッチするかどうかを調べている。

5.1.2. 初期画面（教員用）

初期画面（教員用）とは、ユーザ（教員）のメイン画面であり、課題の作成を開始することができる画面である。

作成した初期画面（教員用）のスクリーンショットを図 5.2 に示す。

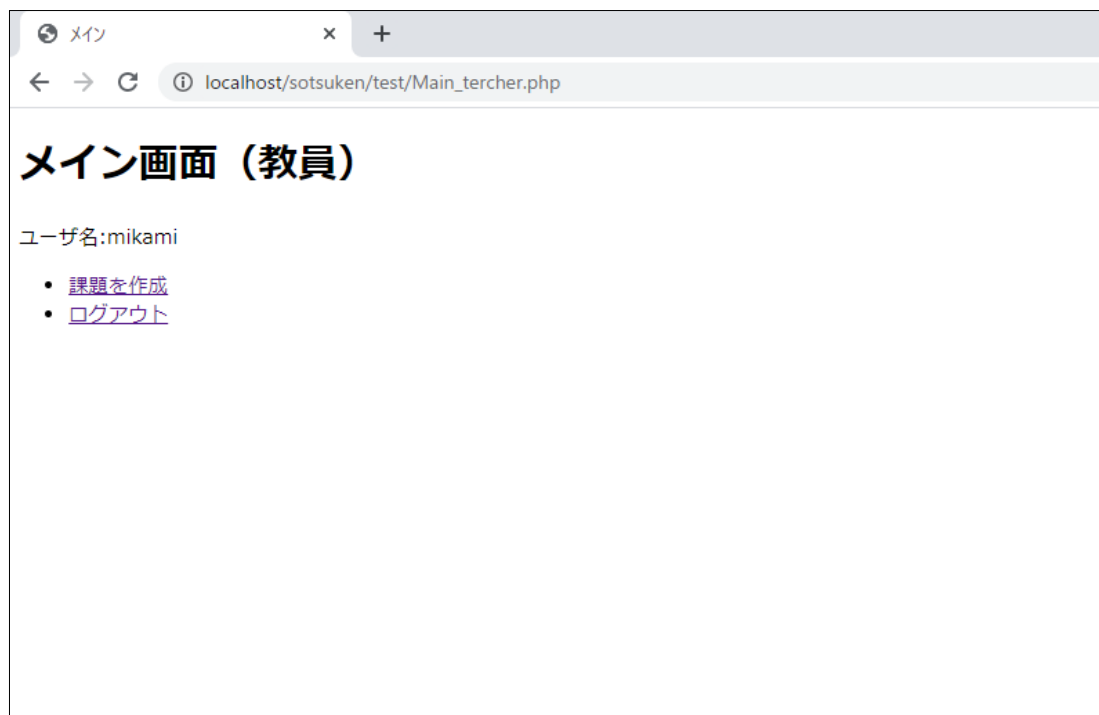


図 5.2. 初期画面（教員用）

この画面はログインユーザが教員の場合に遷移することができる。ログインユーザが学生状態で遷移を試みた場合、5.1.6 に示す初期画面（学生用）に遷移する。

この画面には課題作成リンクとログアウトリンクが設けられている。また、画面上部に現在ログインしているユーザ名が表示される。

課題作成リンクをクリックすると、課題作成画面へ遷移する。ログアウトリンクをクリックすると、PHP のセッション変数を開放することで現在ログインしているユーザからログアウトし、ログイン画面へ遷移する。

5.1.3. 課題作成画面

課題作成画面とは、ユーザ（教員）が課題を作成する際の科目と課題名を決定する画面である。

作成した課題作成画面のスクリーンショットを図 5.3 に示す。

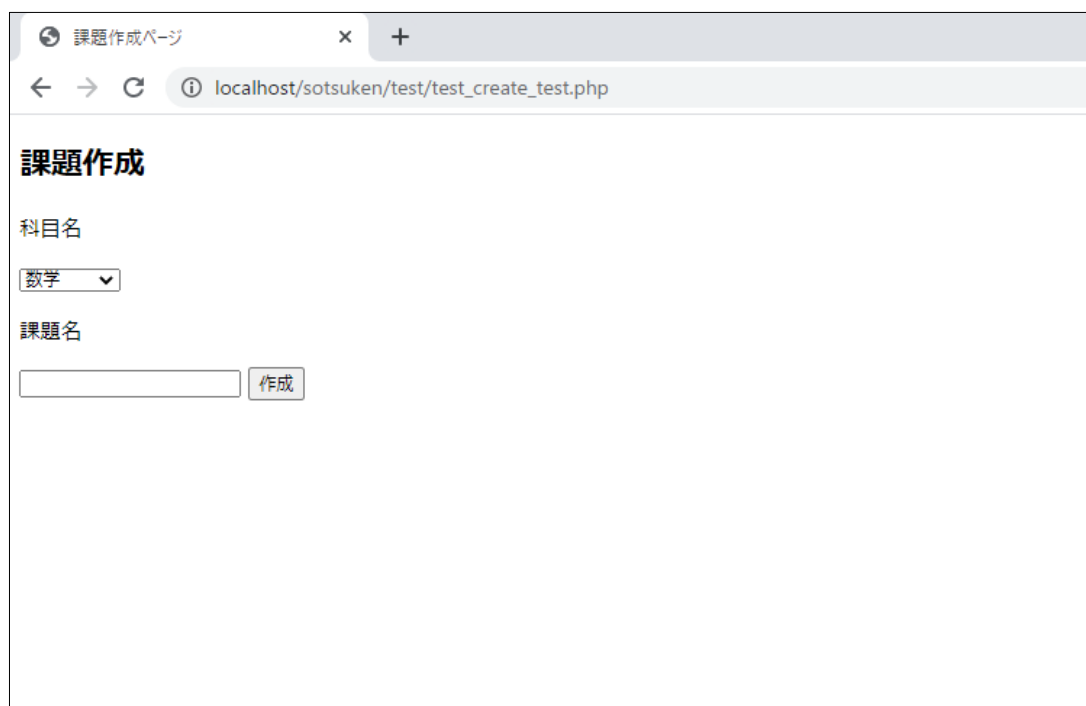


図 5.3. 課題作成画面

この画面はログインユーザが教員の場合に遷移することができる。ログインユーザが学生状態で遷移を試みた場合、5.1.6 に示す初期画面（学生用）に遷移する。

この画面には科目選択のプルダウンメニューと課題名の入力フォーム、作成ボタンが設置されている。科目名のプルダウンメニューは、データベースから現在ログインしている教員が担当する科目を取得し、取得した科目名が表示されている。

作成ボタンを押下すると、現在選択している科目名と入力した課題名がセッション変数に格納され、設問作成画面に遷移する。

5.1.4. 設問作成画面

設問作成画面とは、ユーザ（教員）が課題の設問を作成する画面である。

作成した設問作成画面のスクリーンショットを図 5.4 に示す。

設問作成

localhost/sotsuken/test/test_create_Q.php

数学課題

問1

問題文

☐エディタを使用 ☐フローチャートを使用 ☐手書き数式を使用

選択肢

1
☐

2
☐

図 5.4. 設問作成画面

この画面はログインユーザが教員の場合に遷移することができる。ログインユーザが学生状態で遷移を試みた場合、5.1.6 に示す初期画面（学生用）に遷移する。

この画面には初期状態で問題文を入力するテキストエリア、使用オプションを選択するチェックボックス、選択肢を入力するテキストボックス、「次へ」ボタン、「作成完了」ボタンが存在する。またこの画面の見出し（h1 要素）は、課題作成画面で入力した課題名となっている。

使用オプションを選択するチェックボックスは複数選択が可能で、各チェックボックスを選択することで対応するオプションを表示させることができる。「エディタを使用」のチェックボックスを選択すると、コードエディタが表示される（図 5.5）。「フローチャートを使用」のチェックボックスを選択すると、フローチャート作成フォームが表示される（図 5.6）。「手書き数式を使用」のチェックボックスを選択すると、手書き数式フォームが表示される（図 5.7）。

数学課題I

問1

問題文

☒エディタを使用 ☐フローチャートを使用 ☐手書き数式を使用

エディタ

Q言語H

1

選択肢

1

☐

2

☐

-

+

次へ

テスト作成

図 5.5. 「エディタを使用」のチェックボックスを選択した際の設問作成画面

21

数学課題I

問1

問題文

☐エディタを使用☒フローチャートを使用☐手書き数式を使用

• ☒処理

• ☐判断

• ☐ループ

開始

終了

追加

選択肢

1

☐

2

☐

-

+

次へ

テスト作成

図 5.6. 「フローチャートを使用」のチェックボックスを選択した際の設問作成画面

数学課題I

問1

問題文

☐エディタを使用 ☐フローチャートを使用 ☒手書き数式を使用

手書き数式

コピー

Convert

選択肢

1

☐

2

☐

-

+

次へ

テスト作成

図 5.7. 「手書き数式を使用」のチェックボックスを選択した際の設問作成画面

5.1.5. 作成完了画面

作成完了画面とは、ユーザ（教員）が課題の設問を全て作成完了した際に表示される画面である。

作成した作成完了画面のスクリーンショットを図 5.8 に示す。

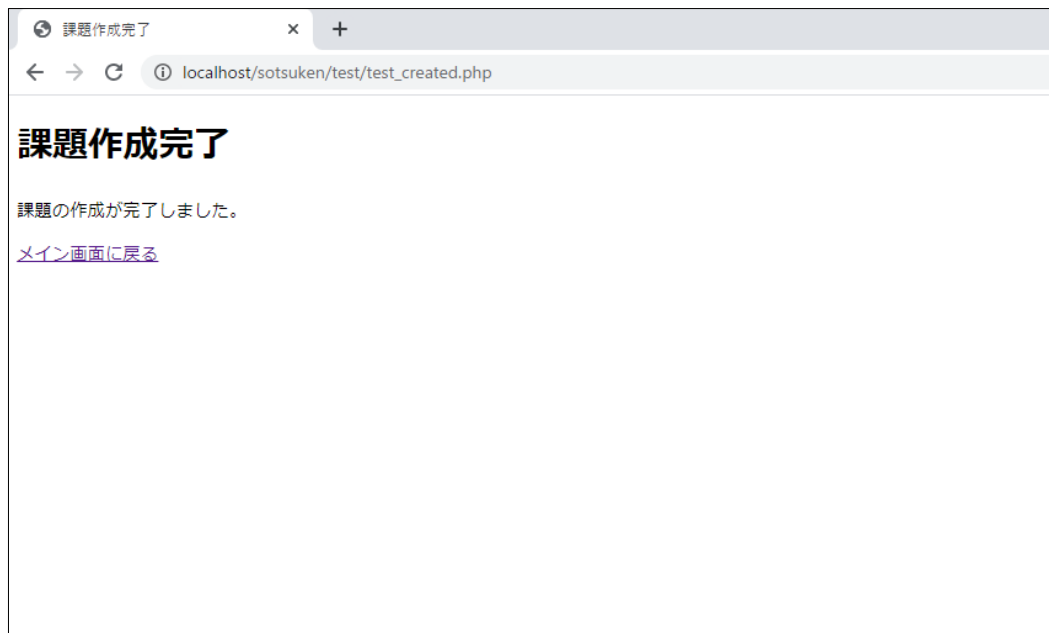


図 5.8. 作成完了画面

この画面はログインユーザが教員の場合に遷移することができる。ログインユーザが学生状態で遷移を試みた場合、5.1.6 に示す初期画面（学生用）に遷移する。

「メイン画面に戻る」のリンクをクリックすることで、初期画面（教員用）に遷移する。

5.1.6. 初期画面（学生用）

初期画面（学生用）とは、ユーザ（学生）のメイン画面であり、課題の作成を開始することができる画面である。

作成した初期画面（学生用）のスクリーンショットを図 5.9 に示す。

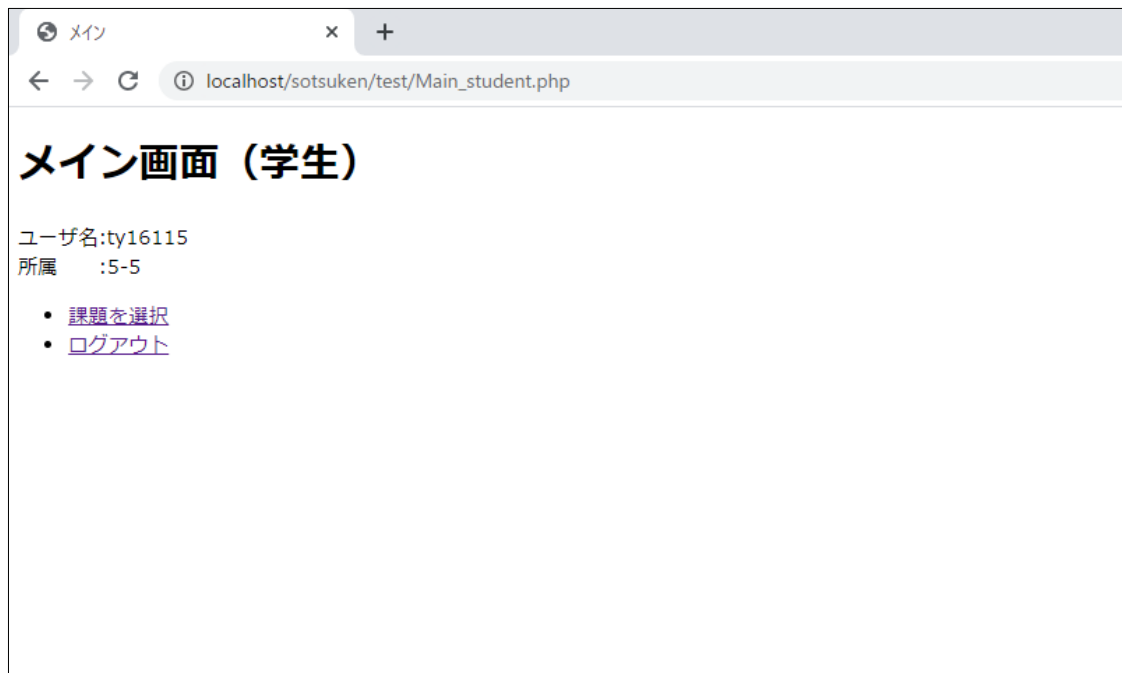


図 5.9. 初期画面（学生用）

この画面はログインユーザが学生の場合に遷移することができる。ログインユーザが教員の状態で遷移を試みた場合、5.1.2 に示す初期画面（教員用）に遷移する。

この画面には課題選択リンクとログアウトリンクが設けられている。また、画面上部に現在ログインしているユーザ名と、ユーザが所属しているクラス名が表示される。

課題選択リンクをクリックすると、課題選択画面へ遷移する。ログアウトリンクをクリックすると、PHP のセッション変数を開放することで現在ログインしているユーザからログアウトし、ログイン画面へ遷移する。

5.1.7. 課題選択画面

課題選択画面とは、ユーザ（学生）が取り組む課題を課題一覧から選択する画面である。作成した課題選択画面のスクリーンショットを図 5.10 に示す。



図 5.10. 課題選択画面

この画面はログインユーザが学生の場合に遷移することができる。ログインユーザが教員の状態で遷移を試みた場合、5.1.2 に示す初期画面（教員用）に遷移する。

この画面には、科目別に課題名のリンクが設けられている。また、画面上部に現在ログインしているユーザ名と、ユーザが所属しているクラス名が表示される。

科目名と課題名はデータベースに登録されているものの内、ログインユーザの所属しているクラスと、科目の開講クラスが一致するものを表示している。また、クラスと科目の関係は多対多の関係であるため、中間テーブル（クラス-科目テーブル）を用意し、そこに関係を登録している。

課題名のリンクをクリックすると、クリックした課題の回答画面へ遷移する。

5.1.8. 回答画面

回答画面とは、ユーザ（学生）が課題の設問の回答を入力する画面である。

作成した回答画面のスクリーンショットを図 5.11 に示す。

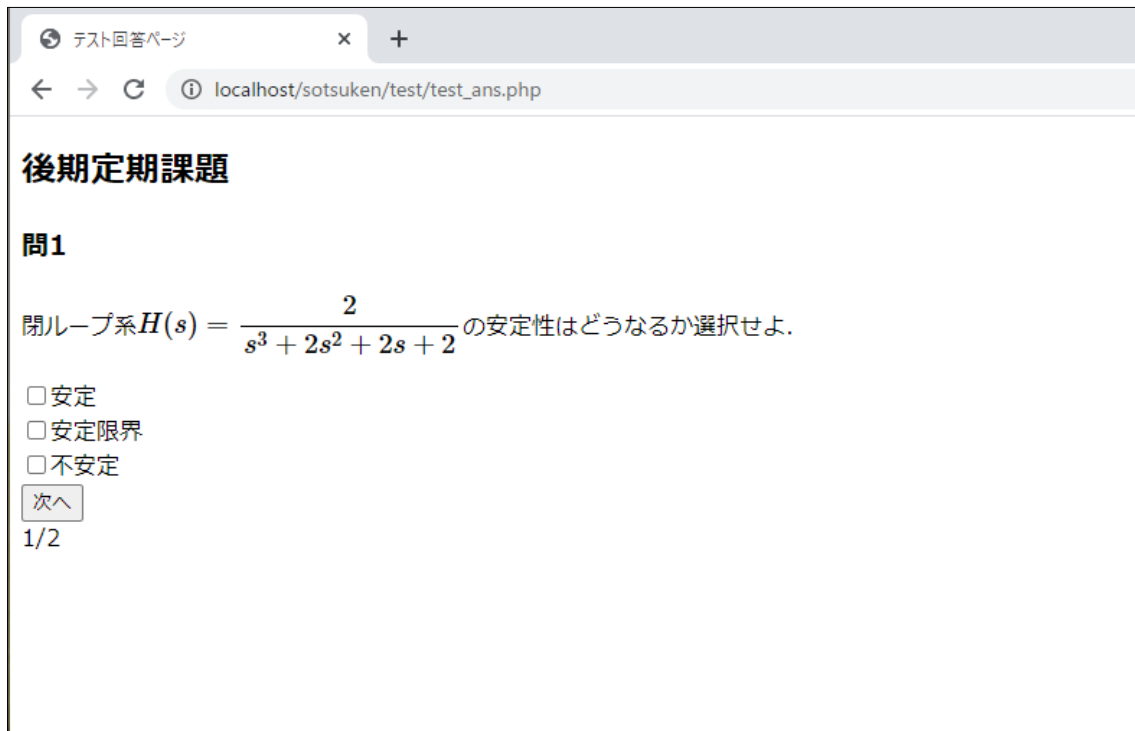


図 5.11. 回答画面

この画面はログインユーザが学生の場合に遷移することができる。ログインユーザが教員の状態で遷移を試みた場合、5.1.2 に示す初期画面（教員用）に遷移する。

この画面には、現在取り組んでいる課題名、設問番号、問題文、選択肢などの問題情報と「次へ」または「回答」ボタン、現在のページ番号が表示される。

設問の選択肢のチェックリストを選択した状態で「次へ」ボタンを押下することによって、設問に回答することができる。また、表示している設問が課題の最終設問である場合は、「次へ」ボタンが「回答」ボタンに置き換わり、「回答」ボタンを押下することによってすべての回答がデータベースに登録され、回答完了ページに遷移し課題が終了する。

5.1.9. 回答完了画面

回答完了画面とは、ユーザ（学生）が課題の全ての設問への回答が完了した際に表示される画面である。

作成した回答完了画面のスクリーンショットを図 5.12 に示す。

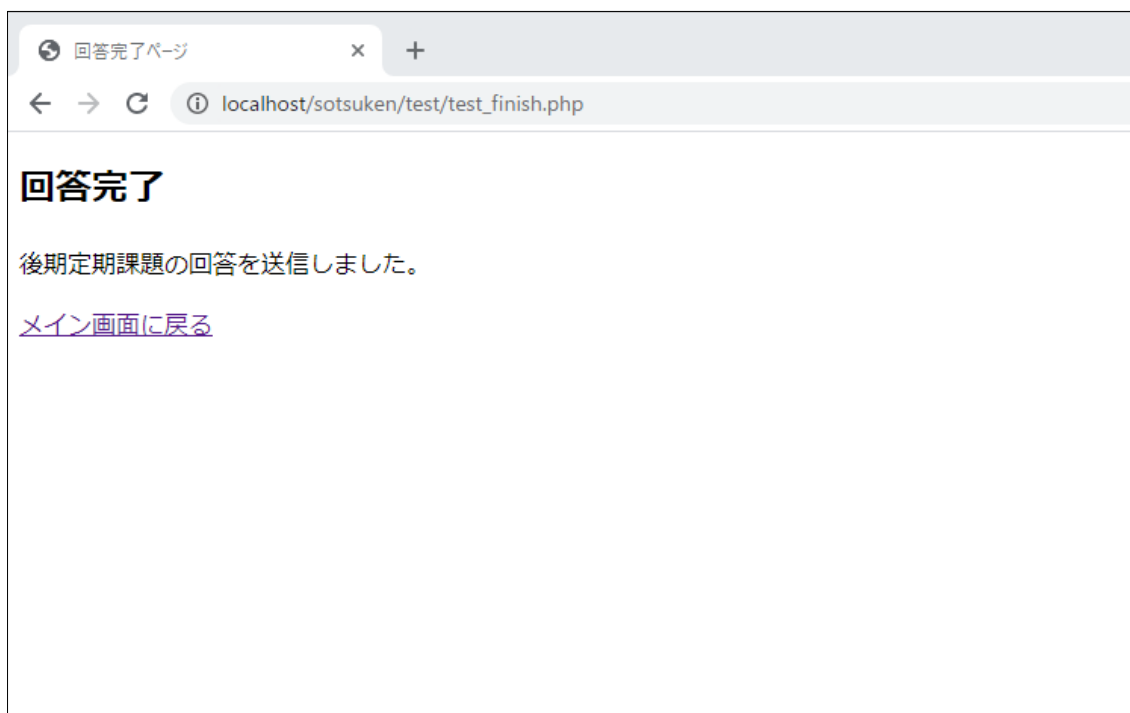


図 5.12. 回答完了画面

この画面はログインユーザが教員の場合に遷移することができる。ログインユーザが教員の状態で遷移を試みた場合、5.1.6 に示す初期画面（教員用）に遷移する。

「メイン画面に戻る」のリンクをクリックすることで、初期画面（学生用）に遷移する。

5.2. 作成したデータベース

作成したデータベースを構成するテーブルを表 5.1 に示す。また、各テーブルの仕様を表 5.2～表 5.○に示す。

表 5.1. データベースを構成するテーブル

テーブルの名称	テーブルの定義名	目的・用途	関連するテーブルの名称
ユーザ情報テーブル	userData	ユーザの情報を登録する	
役職テーブル	Role	ユーザの役職（教員、学生）を登録する	
クラステーブル	Classes	学校の学級名を登録する	
科目テーブル	SubjectTable	ユーザが開講・受講する科目を登録する	
クラス-科目テ	Classes_Subject	各学級の持つ各科目の関	

ーブル		係を登録する クラステーブルと科目テ ーブルの中間テーブル	
課題テーブル	TestTable	ユーザ（教師）が作成した 課題を登録する	
設問テーブル	QuestionTable	課題テーブルに登録され た課題の設問を登録する	
ソースコードテ ーブル	Q_Code	設問テーブルに登録され た設問に添付されている ソースコードを登録する	
プログラミング 言語テーブル	P_languages	設問を作成する際のコー ドエディタで使用するこ とができるプログラミング 言語を登録する	
フローチャート テーブル	Flowchart	設問テーブルに登録され た設問に添付されている フローチャートを登録す る	
回答選択肢テー ブル	Q_CheckBoxTable	設問テーブルに登録され た設問の選択肢を登録す る	
回答テーブル	AnswerTable	設問に対する回答を登録 する	

表 5.2. ユーザ情報テーブル

カラムの名 称	カラムの定 義名	キー 項目	データ型	長さ	条件制約、補足など
ID	ID	主 キ ー	int	11	not null、主キー制約、 オートインクリメント
ユーザ名	name	-	varchar	20	not null
パスワード	password	-	varchar	100	not null、ハッシュ化され ているため長さを長めに 確保している
ユーザの役 職 ID	role_ID	外 部 キー	int	11	not null、外部キー制約、 役職テーブルを親に持つ

所属クラス ID	class_ID	外部 キー	int	11	not null、外部キー制約、 クラステーブルを親に持 つ
-------------	----------	----------	-----	----	--------------------------------------

表 5.3. 役職テーブル

カラムの名 称	カラムの定 義名	キー 項目	データ型	長さ	条件制約、補足など
ID	ID	主キ ー	int	11	not null、主キー制約、 オートインクリメント
役職名	role_name	-	varchar	10	not null

表 5.4. クラステーブル

カラムの名 称	カラムの定 義名	キー 項目	データ型	長さ	条件制約、補足など
ID	ID	主キ ー	int	11	not null、主キー制約、 オートインクリメント
クラス名	class_name	-	varchar	20	not null

表 5.5. 科目テーブル

カラムの名 称	カラムの定 義名	キー 項目	データ型	長さ	条件制約、補足など
ID	ID	主キ ー	int	11	not null、主キー制約、 オートインクリメント
科目名	name	-	varchar	20	not null
教員 ID	teacher_ID	外部 キー	int	11	not null、外部キー制約、 ユーザテーブルを親に持 つ

表 5.6. クラス-科目テーブル

カラムの名 称	カラムの定 義名	キー 項目	データ型	長さ	条件制約、補足など
------------	-------------	----------	------	----	-----------

ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
クラス ID	class_ID	外部キー	int	11	not null、外部キー制約、クラステーブルを親に持つ
科目 ID	subject_ID	外部キー	int	11	not null、外部キー制約、科目テーブルを親に持つ

表 5.7. 課題テーブル

カラムの名称	カラムの定義名	キー項目	データ型	長さ	条件制約、補足など
ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
科目 ID	subject_ID	外部キー	int	11	not null、外部キー制約、科目テーブルを親に持つ
作成者 ID	creator_ID	外部キー	int	11	not null、外部キー制約、ユーザテーブルを親に持つ
課題名	test_name	-	varchar	30	not null

表 5.x. 設問テーブル

カラムの名称	カラムの定義名	キー項目	データ型	長さ	条件制約、補足など
ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
課題 ID	test_ID	外部キー	int	11	not null、外部キー制約、課題テーブルを親に持つ
設問番号	Q_number	-	int	11	not null
問題文	Q_text	-	text	-	not null
画像の有無	is_media	-	tinyint	1	not null、デフォルト値は 0
ソースコードの有無	is_code	-	tinyint	1	not null、デフォルト値は 0

フローチャートの有無	is_flowchart	-	tinyint	1	not null、デフォルト値は0
------------	--------------	---	---------	---	-------------------

表 5.x. ソースコードテーブル

カラムの名称	カラムの定義名	キー項目	データ型	長さ	条件制約、補足など
ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
設問 ID	Q_ID	外部キー	int	11	not null、外部キー制約、設問テーブルを親に持つ
プログラミング言語 ID	P_language_ID	外部キー	int	11	not null、外部キー制約、プログラミング言語テーブルを親に持つ
ソースコード	code	-	text	-	not null

表 5.x. プログラミング言語テーブル

カラムの名称	カラムの定義名	キー項目	データ型	長さ	条件制約、補足など
ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
言語名	language_name	-	varchar	20	not null

表 5.x. フローチャートテーブル

カラムの名称	カラムの定義名	キー項目	データ型	長さ	条件制約、補足など
ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
設問 ID	Q_ID	外部キー	int	11	not null、外部キー制約、設問テーブルを親に持つ
フローチャート本文	flowchart_text	-	text	-	not null

表 5.x. 回答選択肢テーブル

カラムの名称	カラムの定義名	キー項目	データ型	長さ	条件制約、補足など
ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
設問 ID	Q_ID	外部キー	int	11	not null、外部キー制約、設問テーブルを親に持つ
選択肢テキスト	checkbox_text	-	text	-	not null
選択肢の正誤	is_correct_Ans	-	tinyint	1	not null、デフォルト値は 0

表 5.x. 回答テーブル

カラムの名称	カラムの定義名	キー項目	データ型	長さ	条件制約、補足など
ID	ID	主キー	int	11	not null、主キー制約、オートインクリメント
設問 ID	Q_ID	外部キー	int	11	not null、外部キー制約、設問テーブルを親に持つ
ユーザ ID	user_ID	外部キー	int	11	not null、外部キー制約、ユーザテーブルを親に持つ
回答の正誤	is_correct	-	tinyint	1	not null、デフォルト値は 0

6. 評価

6.1. システムのテストケース

システムのテストケースを以下の表 6.1 に示す。

7. 今後の課題

7.1. ユーザインタフェースの改善

7.2. 回答画面の機能追加

7.3. 成績管理機能の追加

7.4. セキュリティ対策

8. 結言

本研究では、情報系科目向け課題作成・回答システムの開発を行った。課題作成システムの面では、従来の LMS にはない入力機能を追加することで、情報系科目の教員に向けたシステムを開発することができた。課題回答システムの面では、回答形式が限定されているため、情報系科目の学生の利用に特化したシステムとは言い難いものの、一般的な回答システムに近いシステムを開発することができた。

システム全体では、ユーザインタフェースの観点や機能数、システムの安定性といった点から、完成度の高いシステムを開発することはできなかった。

本研究を通じて、Web アプリケーション開発に必要な技術と知識、リレーショナルデータベースの基本的な運用方法を習得することができた。また、想定ユーザに向けた適切な機能を持つアプリケーションを作成することができた。

9. 謝辞

本研究を行うにあたり、テーマの決定や論文執筆について様々なご指導を頂いた三上剛教授に深く感謝申し上げます。また、研究を進めるにあたって助言をいただいた同研究室、三河教授研究室の皆様感謝します。

10. 参考文献

[1] コロナ対策の現状、課題、今度の方向性について – 文部科学省

URL : https://www.mext.go.jp/content/20200924-mxt_keikaku-000010097_3.pdf

最終閲覧日 : 2021 年 1 月 14 日

[2] e ラーニング - Wikipedia

URL :

<https://ja.wikipedia.org/wiki/E%E3%83%A9%E3%83%BC%E3%83%8B%E3%83%B3%E3%82%B0%E6%95%99%E6%9D%90%E3%83%BB%E5%AD%A6%E7%BF%92%E6%9D%90>

最終閲覧日 : 2021 年 1 月 15 日

[3] e ラーニングとは？意味・目的・方法・メリット・比較など徹底解説！

URL : <https://www.pro-seeds.com/trend/elearning.html>

最終閲覧日 : 2021 年 1 月 15 日

[4] LAMP とは | クラウド・データセンター用語集／IDC フロンティア

URL : <https://www.idcf.jp/words/lamp.html>

最終閲覧日 : 2021 年 1 月 9 日

[5] Linux 用 Windows サブシステムの概要 | Microsoft Docs

URL : <https://docs.microsoft.com/ja-jp/windows/wsl/>

最終閲覧日 : 2021 年 1 月 11 日

[6] Historical trends in the usage statistics of web servers, January 2021

URL : https://w3techs.com/technologies/history_overview/web_server

最終閲覧日 : 2021 年 1 月 11 日

[7] DB-Engines Ranking - popularity ranking of database management systems

URL : <https://db-engines.com/en/ranking>

最終閲覧日 : 2021 年 1 月 11 日

[8] 今さら聞けない！PHP とは【初心者向け】 | 現役エンジニアが解説 | TechAcademy
マガジン

URL : <https://techacademy.jp/magazine/6618>

最終閲覧日 : 2021 年 1 月 15 日

[9] PHP: PHP とはなんでしょう? - Manual
<https://www.php.net/manual/ja/intro-what-is.php>

最終閲覧日 :2021 年 1 月 15 日

[10] JavaScript とは ・ JavaScript Primer #jsprimer
<https://jsprimer.net/basic/introduction/>

最終閲覧日 :2021 年 1 月 15 日

[11] jQuery
<https://jquery.com/>

最終閲覧日 :2021 年 1 月 15 日

[12] Ace - The High Performance Code Editor for the Web
<https://ace.c9.io/>

最終閲覧日 :2021 年 1 月 15 日