

教育アプリケーション構築を題材にした コンピュータサイエンス実験の試み

赤澤紀子[†] 及川永寿[†] 原克彬[†] 中山泰一[†] 角田博保[†]

コンピュータサイエンスに対応した応用的な実験として、データベース、Webプラットフォームを利用したインタラクティブな教育アプリケーションの設計構築の実験を行った。多くの学生にとって、自ら必要な機能を考え、実装することは初めての体験であることから、学生にとって馴染みのある、LMSの機能を学生の目線で設計し構築することを課題とした。この実験を通して、アプリケーションを利用するユーザーを想定しながらシステムを設計、構築することの難しさや楽しさを体感することができ、今後のシステムの設計・構築に役立つ実験になったと考える。また今回の実験で得ることのできた学生が考えるLMSに必要な機能を紹介する。

Trial of the computer science practice about the education application implementation

Noriko Akazawa[†] Hisatoshi Oikawa[†] Katsuaki Hara[†]
Yasuichi Nakayama[†] and Hiroyasu Kakuda[†]

As an experiment of the computer science, our students designed and built of interactive education application. The theme of this experiment is to design functions of LMS that a student thinks, and to build them. The students experienced the difficulty and pleasure of designing and building. We think that this experience helps the students to design and build systems in future. Moreover we indicate functions of LMS that students thought.

1. はじめに

電気通信大学情報理工学部情報・通信工学科コンピュータサイエンスコースでは、コンピュータサイエンスの基礎を2年次までに身に付け、3年次には、一年を通じてコンピュータサイエンス実験を履修する。このコンピュータサイエンス実験では、学生が複数の課題に取り組み、的確な問題分析とシステムの設計、実現、評価ができること、明晰な報告書が作成できることを目標としている。

これまで学生は、与えられた仕様に沿ってプログラミングを行う実験などを履修してきた。そこで本論文では、学生に大まかな課題を指定し、学生自身が課題から問題解決を行い、設計構築を行う実験を実施した。

近年、データベースとWebプラットフォームを利用したインタラクティブなアプリケーションは、インターネットショッピングやSNSなど多種多様になっている。教育においても、商用アプリケーションやオープンソースを利用したアプリケーションが活用されており、オープンソースを利用したアプリケーションについての研究も盛んに行われている[1][2]。普段から学生は、データベースとWebプラットフォームを利用したアプリケーションとして、MoodleやWebClassなどのLMS(Learning Management System)を講義等で使用している。そのため、学生にとってLMSは、機能イメージがつかみ易く、設計構築し易い題材であると

考えられる。そこで本実験の課題として、LMSの設計構築を取り上げた。

本論文では、実際に学生が独自のLMSを設計し、構築した成果から、学生自身が大まかな課題から問題解決を行い、設計構築することが可能であることを示した。

本論文では、まず2章で本実験の目的、実験課題および実験の流れについて述べ、3章で学生の成果物について述べ、次に4章で実験に参加した学生の実験に対する評価を述べ、さらに学生評価の考察を述べる。最後に5章で本論文のまとめと今後の課題を述べる。

2. 実験概要

2.1 コンピュータサイエンス実験とは

コンピュータサイエンス実験は、2コマ×7回を1ラウンドとしている。ラウンドごとに複数の課題が開設されており、学生は、選択的に履修する。

そのうち、“教育アプリケーション構築を題材にしたコンピュータサイエンス実験”(以降、本実験)は、2ラウンド分実施した。

本実験は、1ラウンド目は、9人が履修し個人で設計構築を行った(以降、個人ラウンド)。2ラウンド目は、8人が履修し、4人ずつのグループで設計構築を行った(以降、グループラウンド)。

2.2 本実験の目的と背景

これまで、学生はすでに用意された環境下で、与えられた仕様に沿って実験を行い、まとめ考察を行うことを行っ

[†] 電気通信大学大学院情報理工学研究科情報・通信工学専攻
Graduate School of Informatics and Engineering
Department of Communication Engineering and Informatics
The University of Electro-Communications

てきた。しかし、コンピュータサイエンス実験は、コンピュータサイエンスに対応した応用的な実験を行うことも目的としている。そこで、本実験では、大まかな課題を提示し、学生自身が課題に対する問題解析を行い、システム設計構築を行うことにより、的確な問題解析とシステムの設計、実現する力を養うことを目的とした。本論文では、その結果について示し考察を行った。

また、大まかな課題から問題解析を行うことは困難が予想される。解決策として、学生にとって身近な課題を提示することにより、システムの利用者を想像し易く、使用手順の理解が容易になり、実験への取り組み意欲を継続できると考えた。本論文では、実験の成果物やアンケートから、学生にとって身近な LMS の構築を課題とした実験の考察を行った。

2.3 実施環境

本実験では、サーバサイドのプラットフォームとして Node.js[3]を、DB(データベース)は MongoDB[4]を使用した(図 1)。また、学生は、あらかじめ用意されたサーバ上の Node.js と MongoDB に、リモートアクセスすることによりプログラミングを行った。

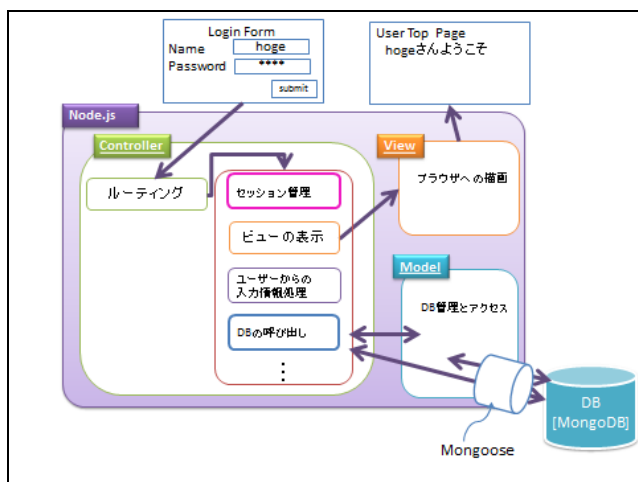


図 1 システム概要

2.3.1 Node.js

サーバサイドプラットフォームを構築するためには、Apache など様々なサーバソフトがあるが、JavaScript の実行プラットフォームである Node.js は、大量の同時接続ユーザーに対して高速にレスポンスできることなど、近年注目を集めている。Node.js の特徴を以下に示す。

・ MVC モデル

“Model” (アプリケーションが扱うデータや手続きを行う)、“View” (ユーザーへの出力、及び入力を行う)、“Controller” (View と Model の制御を行う)というように、明確に機能を分離することで、開発作業の分業が容易になり、また、互いに仕様変更の影響を受けにく

い。

・ 非同期 I/O

一つのスレッドで大量の接続を処理できる。これにより、一台のサーバで数万～数十万という接続を取り扱うことも可能である。また、非同期 I/O を扱うためには、高度な知識と、コードの中で非同期と同期処理を意識しならぬプログラミングが要求されることが多い。しかし、Node.js は、非同期 I/O を前提にしており、比較的容易に非同期 I/O を扱うことができる。

・ JavaScript

JavaScript はクライアントのブラウザで動く言語として有名であるが、Node.js は、JavaScript をサーバサイドのプログラムとして実行することを可能にしている。

これらの特徴から、本実験では Node.js を使用することとした。また、本実験では、学生に必要な最小限の実行環境を提供し、実行環境の充実が学生の判断に任せることとしたため、必要に応じてモジュールを追加できる Node.js は本実験に適していると言える。

2.3.2 MongoDB

一般的に Web サイトを構築するためには、DB(データベース)の存在が不可欠である。Node.js は非同期という性質上、DB のアクセスも非同期に行う。Web のように大量の処理を同時に実行する場合、コネクションプーリングを使用して、DB へのアクセスの性能低下を防ぐ方法がとられている。Node.js でも、コネクションプーリングは有効な手法であるが、これでは、非同期のメリットが活きていない。そこで、DB も非同期処理が可能であることが望ましい。また、Node.js は、JavaScript の実行プラットフォームであることから、言語使用が厳格でないため、オブジェクトを明確に定義するのではなく、ダイナミックなオブジェクトを扱う DB との親和性が高い。これらのことから、ドキュメント指向型の NoSQL データベースである MongoDB を使用することとした。また、Node.js からのアクセスは、非同期環境において機能することを目的とした MongoDB のオブジェクトモデリングツール Mongoose[5]を使用することとした。

2.4 実験開始前の学生の状態

実験終了後のアンケートによると、実験開始前に、LMS について知っていた学生は半数程度であった。学生は普段から、Moodle や WebClass などを利用しているが、これらを LMS と認識して使用していなかったようであった。MVC モデルについては 40%程度の学生が知っており、Node.js については知らなかった生徒が多数を占めていた(図 2)。また、使用経験については、JavaScript は 35%、DB は 20%程度であった(図 3)。さらに、グループで設計

構築を行った後半のラウンドの学生は、全員がグループでの設計構築経験がなかった。

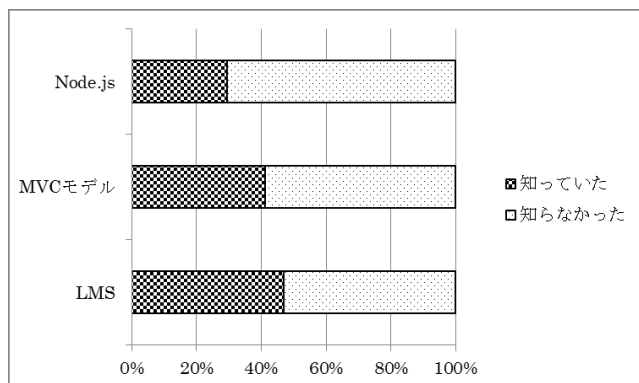


図 2 知識習得状況

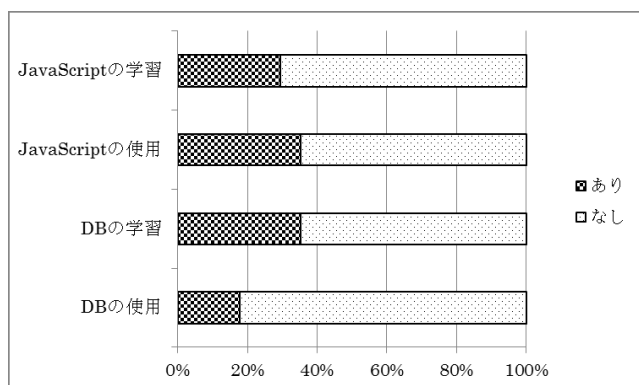


図 3 スキル習得状況

2.5 実験の本課題

本実験では、本課題として“学生の目線から考えた LMS の作成（独自 LMS の構築）”を出題した。機能は、実装できそうか否かで決めるのではなく、この機能があったら授業や自己学習で役立つかどうかの観点で考えることや、既存の LMS システムにはない機能を考えることを推奨した。

2.6 実験のながれ

本実験は準備演習と本課題“独自 LMS の構築”の二段階に分けて実施した。

2.6.1 準備演習

実験の 1~3 回は、本課題の準備として必要最小限の講義と技術を身に付けるための課題を出題した。実験 1 回目は、Node.js の基本を習得し、実験第 2 回目 3 回目は、DB(MongoDB, Mongoose)の使い方とセッション管理を習得した。

2.6.2 本課題“独自 LMS の構築”

実験の 4~7 回は、本課題を実施するため、以下のステップで実験を進めた。

(1) LMS に必要と思われる機能を考える。

独自の LMS を構築する時に、実装したい必要だと思う機能を 3 つ以上考えることを課題とした。ここでは初めに、既存の LMS の機能の例を学生に紹介したが、考える機能は、例からだけでなく、独自の機能を考えることを推奨した。また、実装できそうか否かで決めるのではなく、この機能があったら授業や自己学習で役立つかどうかの観点で考えることも推奨した。

(2) 設計

(1)で挙げた機能のうち少なくとも 1 つの機能について、先生用の画面と動作仕様、生徒用の画面と操作仕様を決定する。

(3) 実装

(2)で設計した機能を実装する。

また、グループラウンドのメンバー（4 人で 1 グループを形成）は次のように実験に取り組んだ。(1)については、個人でアイデアを 3 つ以上出してからグループ内で話し合いを行うこととした。(2)(3)については、グループ内で 4 つ（1 機能×4 人）以上の機能を決めて、それぞれの機能の設計構築は個人で行うこととした。学生には、グループ内でコミュニケーションを取り合って、設計構築を行うことを推奨した。さらに、実験終了後にグループ内でメンバーの貢献度評価を行うこととした。

3. 実験の成果物のまとめ

実験 4~7 回を通して学生が構築した成果物について報告する。学生が構築した機能数を図 4 に示す。学生一人あたり 0~4 つの機能を構築した。グループラウンドの学生は、ひとり 1 機能の構築に挑戦していた。これは、グループで実装する機能の担当を決めてから、個々に実装を行ったためと考えられる。また、個人ラウンドの学生は、2 つ以上の機能を実装した。

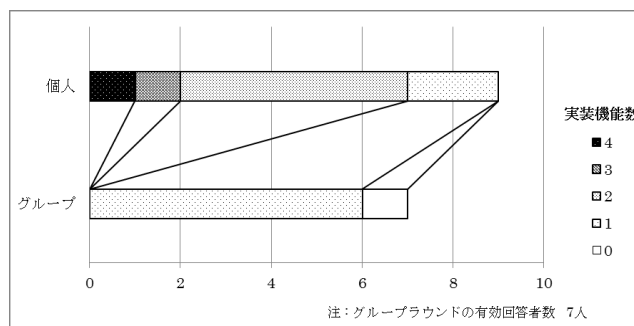


図 4 学生当たりの構築機能数

学生が実装した機能は次の通りである。掲示板やアンケート、チャットといった LMS の定番機能の実装が目立った。講義では説明をしなかった Web Socket を利用したチャ

ットなど、意欲作もあった。

- ・ 掲示板
- ・ アンケート機能
- ・ チャット
- ・ レポート提出機能
- ・ カレンダー機能
- ・ テスト機能
- ・ 資料配布機能
- ・ メッセージ送受信機能
- ・ 板書のリアルタイム教示機能
- ・ 連絡事項掲示機能
- ・ 単語帳
- ・ パスワードを忘れた方へ機能
- ・ プロフィール変更機能
- ・ 生徒一覧機能
- ・ 学生の追加、削除・変更

成果物の一例を以下に示す。図 5 図 6 図 7 は課題提出機能の例である。学生は、学生提出ページでファイルを選択して送信を行う（図 5）。送信後、学生用課題提出ページにて、課題が提出済みであることを確認できる（図 6）。教員（管理者）は、管理者ページから、学生が送信したファイルを確認することが出来る（図 7）。図 8 図 9 は、チャット機能の例である。コメントを入力して送信すると（図 8）、日時、ユーザー名とコメントが表示される（図 9）。

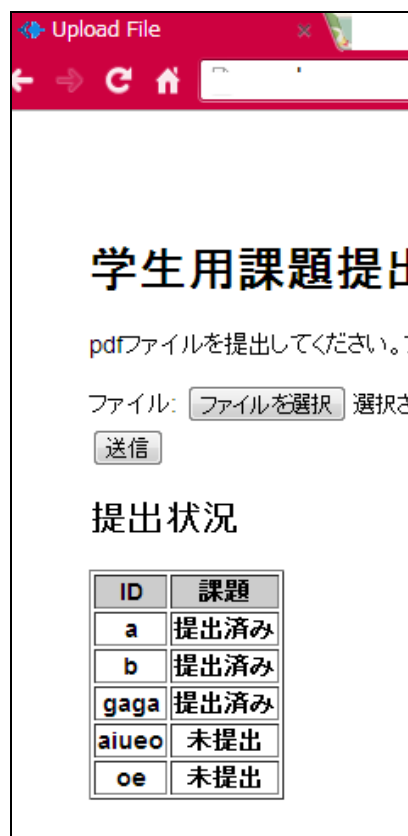


図 6 課題送信後のページ

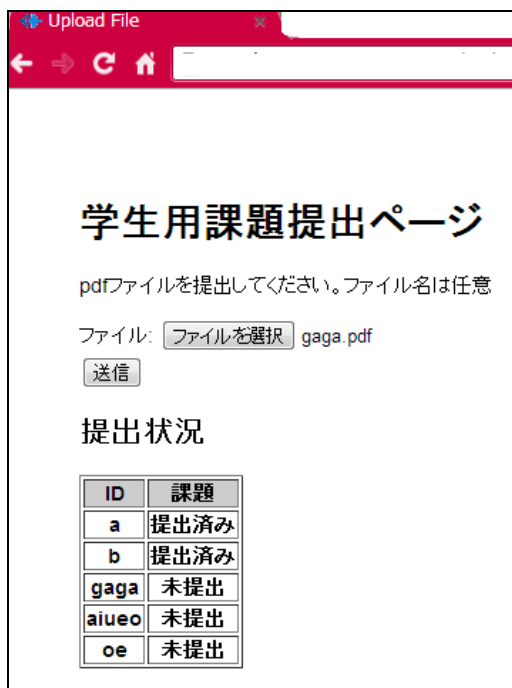


図 5 学生課題提出 ファイルの指定

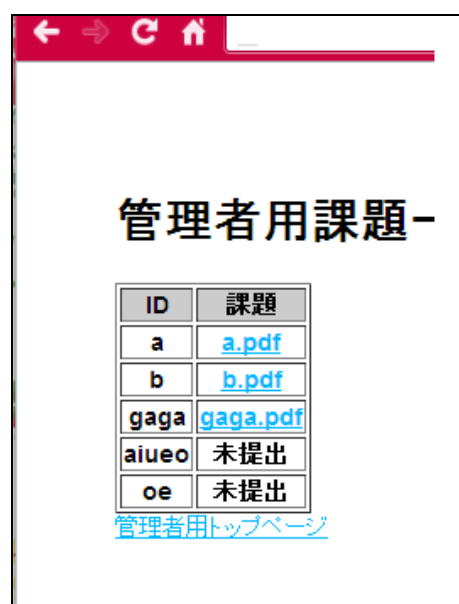


図 7 管理者用課題参照ページ



図 8 チャット 入力



図 9 送信後のページ

また、実装には至らなかったが学生が考えた LMS に必要な機能としては、既存の LMS には見られない以下のような機能もあった。

- ・ ギャラリー

美術や家庭科、情報、数学などでの授業で生徒たちがつくったアート作品を写真で e ポートフォリオ化する。

- ・ 他者の動向情報通知

同じ課題に挑戦している学生の進捗情報「××さんが問題 2 をやって 15 分経過しています」「○○さんが問題 1 を解きました」などを表示して、自分の進捗状況と比較する。

4. 本実験に関する学生による評価とその考察

実験終了時にアンケートを実施し、学生による実験の評価を行った。

本実験の“準備 (Node.js の講義)”“準備 (DB の講義)”“設計”“構築”の 4 項目についての難易度を「簡単だった」「やや簡単だった」「やや難しかった」「難しかった」の 4 を段階で評価した。結果を図 10 に示す。講義は難しく、設計実装は簡単と感じた学生が多いことがわかる。これは、Node.js や MongoDB のみでなく、JavaScript や HTML についても不慣れな学生が多かったことや、必要最小限のみを講義した為、説明不足もあったと考えられる。しかし、準備段階で知識を習得した後の設計と構築は、やや簡単と感じられるようになっている。

図 11 は、“準備 (Node.js の講義)”“準備 (DB の講義)”“設計”“構築”の 4 項目について興味関心を「面白かった」「やや面白かった」「やや面白くなかった」「面白くなかった」の 4 段階で評価した結果である。難しく感じた準備段階の DB や Node.js も新しい知識を吸収できた点で、面白かったと感じたようである。設計については、自分で自由に考えて設計できた点が面白かったなどの意見があった。構築については、設計した機能が考えた通りに動作したことが面白かったようである。時間があれば、もっと機能を構築したかったとの意見もあった。実験全体に関しても、面白かったとの意見が多数を占めていた。

また、本実験の課題が LMS であったことに対しては、自分たちがよく利用しているので、機能を考え易かったとの意見が大半であった。

さらに設計構築する際にユーザーを意識したかについては、半数程度の学生が考えながら設計構築を行うことが出来たと回答している (図 12)。考えなかった学生は、自分のスキルで機能を実装できるか否かが気になりユーザーについては二の次になってしまったとの意見が大半であった。

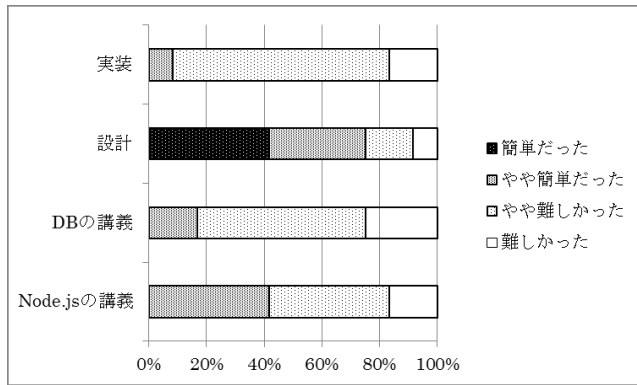


図 10 実験の難易度

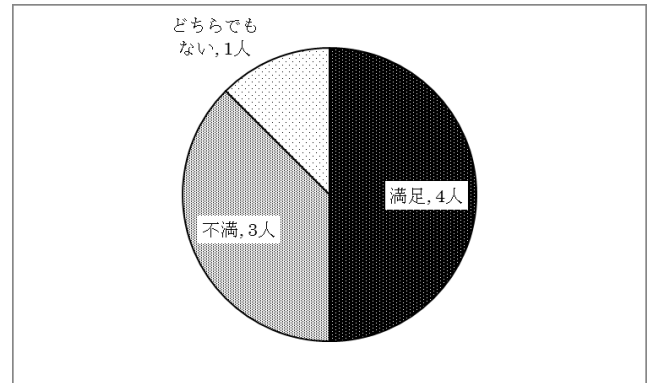


図 13 グループ活動の満足度

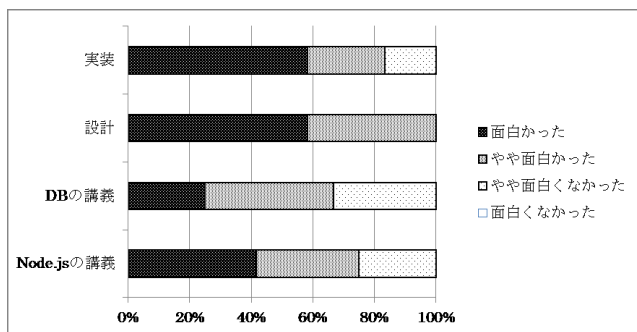


図 11 実験の興味関心

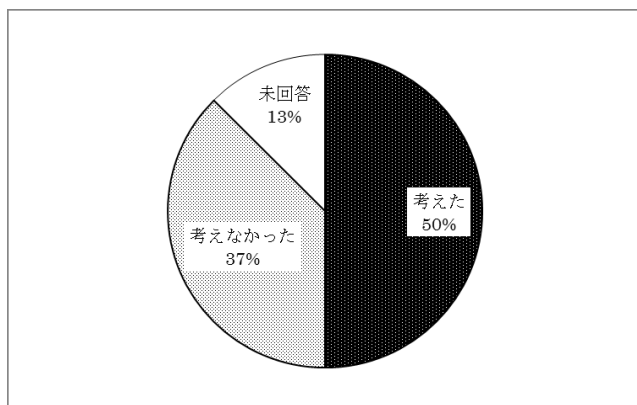


図 12 ユーザーを考慮に入れた設計構築

グループラウンドの学生には、今回行ったグループで行う設計構築に関して個人で作成する場合と比較して満足度を「満足」「不満」「どちらでもない」の3段階の評価をもらった(図13)。全員が満足とはならず、満足は半数にとどまった。これは、グループでまとまって設計構築が行えたグループは満足と答え、結果的に個人で作業をしたグループが不満と答える結果となった。満足、不満どちらの学生もコミュニケーションをとることの難しさを挙げていた。同じグループになって初めて話をする学生同士もいたようで、リーダーシップをとれる学生がいたかどうか、グループでの作業を円滑に進めることに繋がり、満足度の違いに影響があったと考えられる。

さらに、本実験をまたやってみたいか否かについて質問をした本実験の満足度では、またやってみたい学生が大半を占めた(図14)。条件付きでやってみたいと答えた学生は、現状のスキルで実現可能な機能は限られているので、今よりもスキルを身に付けてから、やってみたいとのことであった。

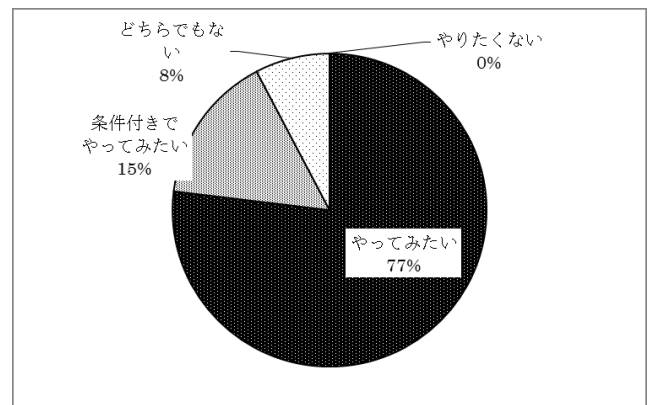


図 14 本実験の満足度

この実験に取り組み習得できたと思うことについてもアンケートを行った。Node.jsやDBなどのスキルを身に付けることができたとの回答もあったが、実験に対する意欲が継続できたとの回答や、グループ作業でのコミュニケーションの重要性についての回答や、アプリケーションを設計構築することの難しさや楽しさを習得できたことがわかる回答を得た。以下に回答の一部を示す。

- ・自分の考えを形にするという達成感が味わえた。また、この実験は楽しく感じられ、真剣に取り組むことができた。
- ・グループ作業での連絡の重要性を感じた。
- ・“使う人のことを考えて作成する”というプログラマーとして初歩の大切な考えをしっかりと身につけることができた。
- ・データベースのデータ管理に少し苦勞したが、苦勞し

た分、自分でコードを見直し1つ1つ理解できた。自分の考えを形にするという達成感が味わえた。

- 学習支援システムを学生の目線から考え、どのような機能があれば便利か考えるのが面白かった。
- 実装する際の設計から開発までの流れを身に付けることができた。
- 一番始めに計画したこと全てを実装することが出来なかったが、成長出来たと思う。

5. 考察および今後の課題

本実験では、学生に大まかな課題を提示し、学生自身が課題の問題解決を行い、道筋を立ててアプリケーションの設計構築を行った。課題は、学生にとって身近に感じられる「LMSの構築」とした。第3章で述べたように、ほぼすべての学生がLMSシステムを構築することができた。

また、第4章で述べたように、学生にとって身近な課題は、システムのイメージが容易であり、実験への意欲の継続を保つことができたと考えられる。また、実験終了後に行った学生へのアンケートから、学生がアプリケーション構築の難しさや楽しさを体験できたことが分かった。

これらのことから、学生自身が問題解決を行い、設計構築を行う実験はコンピュータサイエンスの応用実験として有効あると考えられる。本実験の経験が、学生の今後のシステム設計構築に役立つ経験になったのではないかと考える。

しかし、本実験は、主題を問題解決と設計構築に置いていたにも関わらず、準備段階で必要なスキルの習得に苦戦する学生も多く、準備演習が難しすぎたとの意見があった。さらに設計構築の段階でも、Node.jsやMongoDBに苦戦した学生もいた。これは、今まで、学生は、CやJava中心のプログラミングを行ってきており、JavaScriptに不慣れであったことや、準備段階の講義と演習が、必要最小限の開発方法のみであったことも原因していると考えられる。しかし、アンケート結果からも、Node.jsやMongoDBを学習し使用したことは、とても良い経験になったことも分かった。

また、実際のシステム開発は、通常、グループで行うものであるため、本実験では、グループでの設計構築も行った。しかし、協同の成果が十分に現れなかった。学生からのアンケート結果や実験の様子から、コミュニケーション不足が要因の1つであると考えられる。

今後、本論文のように短期間でシステム設計実装の実験を行う場合、より習得が容易な言語で実装することや、開発方法の講義の内容など精査する必要がある。また、大半の学生が不慣れな実行環境で実験を行う場合は、指導員の人数を増強し、学生が苦手意識を持たないようにする工夫が必要ではないかと考える。また、グループで設計構築を

行う場合、グループ内でのコミュニケーションが円滑にできるよう、事前に顔合わせ等を行い学生間の垣根を低くすることや、技術不足によって必要以上に作業が中断しないための工夫が必要であると考えられる。

参考文献

- 1) 川井敦, 吉田利信: 電気通信大学における情報基礎教育への学習支援システムの導入, 情報処理学会コンピュータと教育研究会報告, 2011-CE-111-12 (2011).
- 2) 小川賀代, 小村道昭: 大学力を高めるeポートフォリオ, 東京電機大出版局 (2012).
- 3) Node.js,
<http://nodejs.org/>
- 4) MongoDB,
<http://www.mongodb.org/>
- 5) Mongoose - Developers Guide,
<http://muddy-dixon.appspot.com/ja/mongoosejs/index.html>