

公立はこだて未来大学 2017 年度 システム情報科学実習
グループ報告書

Future University Hakodate 2017 System Information Science Practice
Group Report

プロジェクト名

数学学習環境のデザインと実現

Project Name

Design and Implementation of a Learning Environment for Mathematics

グループ名

実装班

Group Name

Implementation Group

プロジェクト番号/Project No.

3-B

プロジェクトリーダー/Project Leader

1013223 乗田 拳斗 Kent Norita

グループリーダ/Group Leader

1015165 南部 優太 Yuta Nambu

グループメンバ/Group Member

1015068 石村 秀翔 Shuto Ishimura

1015141 後藤 春奈 Haruna Goto

1015165 南部 優太 Yuta Nambu

1015179 外子澤 考輔 Kouske Unezawa

指導教員

高村博之 美馬義亮 富永敦子

Advisor

Hiroyuki Takamura Yoshiaki Mima Atsuko Tominaga

提出日

2018 年 01 月 19 日

Date of Submission

January 19, 2018

概要

公立はこだて未来大学のカリキュラムポリシーでは、学生が基礎的な能力を身につけるためのものとして、「学部共通専門科目群」を定めている。その「学部共通専門科目群」において、数理思考を学ぶための科目として「解析学 I,II」が存在している。「解析学 I,II」は、2 年次以降の科目の基礎となる重要な講義である。しかし、多くの学生が定理や公式などを暗記し、意味もわからず計算しているだけであり、数理思考を学ぶことができていない。このままでは、本学の学生は単位を取得できたとしても、2 年次以降の講義についていけない。この問題を解決するために、本プロジェクトは「解析学 I,II」の自学自習支援を行うこととした。我々は、前期に問題発見プロセスとして「プロジェクト内勉強会」と「解析学 I 勉強会」を行なった。そして、その活動からプロジェクトメンバーは、教科書を「読む」こと、すなわち用語の定義に立ち戻り、それらをもとに理解を進めることの重要性に気づいた。そこで、本学の解析学履修者に対して、教科書を「読む」ことを促し、支援する e-Learning サービス「ModoLuca」を作成した。これは対話形式で学習者に教科書を読むように誘導するサービスである。ModoLuca の評価実験として、解析学 II 履修者を対象に勉強会を行った。その結果、学習者に教科書の使い方や、教科書を読むことの重要性を伝えることができた。これは、それらの活動において ModoLuca のシステムの実装を行なった、実装班の報告書である。

キーワード 解析学, 学習支援, e-Learning サービス, Modoluca

(※文責: 南部 優太)

Abstract

The Hakodate Future University curriculum policy sets out "Undergraduate Common Specialized Subject Group" as a means for students to acquire basic skills. "Analytics I, II" exists in "Undergraduate Common Specialized Subject Group" to learn mathematical thinking. "Analytics I, II" is an important lecture which is the basis of subjects after the second year. However, many students only memorize theorems and formulas and calculate without knowing its meaning. In other words, they have not been able to learn mathematical thinking. If this goes on, they can not keep up with lectures after the second year. In order to solve this problem, this project decided to support self-study in "Analytics I, II". In the previous term, we conducted "Project Internal Study Meeting" and "Analysis I Study Meeting" as problem finding process. From that activity, we are noticed the importance of "Reading" textbooks. "Reading" is to return to the definition of the term and to promote understanding based on them. Therefore, we created an e-Learning service "ModoLuca" that supports "Reading" textbooks for students in "Analytics I, II". This service Interactively guides students to read textbooks . As an evaluation experiment of ModoLuca, we held a study meeting for students in Analytics II. As a result, we were able to convey the importance of learning how to use textbooks and reading textbooks. This is a report of the Implementation Group who implemented the ModoLuca system in those activities.

Keyword analytics, learning support, e-learning service, ModoLuca

(※: Yuta Nambu)

目次

第 1 章	はじめに	1
1.1	背景	1
1.2	目的	2
1.3	昨年度の活動	3
第 2 章	前期の活動	5
2.1	プロジェクト内学習会	5
2.2	解析学 I 勉強会	6
2.3	中間発表会	9
2.3.1	中間発表会の概要	9
2.3.2	中間発表会の結果と考察	10
第 3 章	後期の活動	13
3.1	e-Learning サービスの提案	13
3.1.1	価値仮説の作成	13
3.1.2	プロトタイプ of 作成	14
3.1.3	設問の作成	15
3.2	システムの実装	15
3.2.1	システムアーキテクチャの考案	15
3.2.2	開発環境の選定	17
3.2.3	モバイルウェブアプリの要件定義	18
3.2.4	Bot サービスの要件定義	20
3.2.5	データベースの要件定義	21
3.2.6	会話モデルの設計	23
3.2.7	実装とそのスケジュール	26
3.3	e-Learning サービス「ModoLuca」	28
3.4	解析学 II 勉強会	31
3.4.1	解析学 II 勉強会の概要	31
3.4.2	解析学 II 勉強会の結果と考察	32
3.4.3	解析学 II 勉強会後のアップデート	33
3.5	成果発表会	33
3.5.1	成果発表会の概要	33
3.5.2	成果発表会の結果と考察	34
第 4 章	活動のまとめと今後の展望	39
4.1	活動のまとめ	39
4.2	今後の展望	39
第 5 章	プロジェクト内のインターワーキング	41

5.1	石村秀翔	41
5.2	後藤春奈	41
5.3	南部優太	42
5.4	刈子澤考輔	42
付録 A	課題解決のための技術（新規習得）	45
付録 B	課題解決のための技術（講義）	47
付録 C	相互評価	49
C.1	石村による相互評価	49
C.2	後藤による相互評価	49
C.3	南部による相互評価	50
C.4	刈子澤による相互評価	50
参考文献		53

第 1 章 はじめに

本報告書は、プロジェクト「数学学習環境のデザインと実現」における「実装班」のグループ報告書である。実装班とは、e-Learning サービスを作成するにあたって、システムの実装を行うグループである。他に、e-Learning サービスに組み込むコンテンツを作成する「コンテンツ班」、e-Learning サービスの評価実験を行う「検証班」がある。本報告書は実装班の報告書であるため、プロジェクトにおける e-Learning サービスの実装に焦点を当てるものである。したがって、e-Learning コンテンツに組み込んだ数学的な設問や、評価実験についての詳細は本報告書では記述を行わない。

本章では、本プロジェクトが活動を行う背景・目的について述べる。また、昨年度の数学学習支援の問題点を指摘し、その課題について言及する。

(※文責: 南部 優太)

1.1 背景

公立はこだて未来大学(以下、「本学」とする)では、情報系大学として数学を重視している。本学のカリキュラムポリシーでは、学生が各コースで専門的に学ぶために必要となる基礎的な能力を身につけるものとして、「学部共通専門科目群」を定めている。基礎的な能力とは、計算論的思考、数理思考、日本語による読解力・作文力、英語の語彙力・読解力・作文力などを指す。「学部共通専門科目群」には、基礎的な能力の1つである数理思考を学生が学ぶために、数学科目が含まれている。よって、本学の学生は、1年次には数学科目として「線型代数学 I,II」、「解析学 I,II」、「数学総合演習 I,II」を履修する。また、2年次以降には「応用数学 I,II」、「確率・統計学」、「確率論」、「力学基礎」、「システム数学基礎」、「システムと微分方程式」などの科目を履修し、情報科学を学ぶために必要となる知識を学ぶ。

2年次以降の数学科目は、カリキュラムポリシーの位置付け通り、学習者が1年次の数学科目の内容を十分に理解していることを前提としている。例えば、「応用数学 I」や「システムと微分方程式」を学ぶ学生には、1年次の科目である「解析学 I,II」の内容を十分に理解していることが求められている。逆にいうと、「解析学 I,II」の単位を取得し進級してきた本プロジェクトのメンバーは、必要な前提知識を十分に持つはずであるため、「応用数学 I」や「システムと微分方程式」などの講義についていけるはずである。

しかし、本プロジェクトメンバーには2年次以降の科目を履修する際に、前提知識が不十分であったために講義についていけなかった者がいた。具体的には、

- 講義「応用数学 I」を履修する際に、微分の理解が不十分であったために偏微分を理解することができない
- 講義「システムと微分方程式」を履修する際に、積分ができずに授業に全くついていけない

ということを経験したプロジェクトメンバーがいた。これは、そのプロジェクトメンバーが1年次に「解析学 I,II」への取り組みが不十分であったために、カリキュラムポリシーが指す数理思考を十分に育めなかったことが1つの要因だと考えられる。

つまり、いかに「解析学 I,II」の単位を取得できたとしても、定理や公式などを暗記し、意味もわからず計算しているだけでは、2 年次以降の講義にはついていけない。このような問題は、本プロジェクトメンバーに限らず、本学の学生全体に起こりうることである。そのため、本学の学生に対して数学への理解を深め、数理思考を学んでもらえるような取り組みが必要である。したがって、本学における解析学の学習環境を整備し、学習者を支援することが必要である。

(※文責: 南部 優太)

1.2 目的

本プロジェクトの目的は、「数学を対象として学ぶ動機や目的意識を高めるための学習環境を情報技術やデザイン技術をもちいて整備する」ことである。具体的には、e-Learning サービスを作成し、本学の学生の解析学自習環境の整備・支援を行うことである。e-Learning とは、「教育・学習に ICT (In-formation Communication Technology) を用いて、その効率や効果を高めるもの」(林, 2012) と定義されている。

前節で述べたように、解析学履修者には学習支援が必要である。そこで、e-Learning サービスによって解析学の自習環境の整備・支援を行うことには理由がある。

自習環境を整備・支援する理由は、自習時間に何をどのように学べば良いのかわからない学生が存在するからである。学生にとって、大学科目の内容を講義の時間だけで理解することは難しい。なぜなら、大学の科目は講義の時間だけで、内容を全て理解できるように設計されていないからである。その証拠として、文部科学省 (2012) によると大学の科目は、講義時間の倍以上、学生が自習するように設計しなければならないと定めている。つまり、学生が大学科目の内容を理解するためには、自学自習が不可欠である。

しかし、学生が自習を行う際には、講義の時間とは異なり学習の方針を指導してくれる人はいない。講義の時間であれば、学生は「読むべき教科書のページを、指導教員が指示する」、「重要な部分には、指導教員が注意を促す」など学習に関する指導を受けることができる。対して自習を行う際には、そのような指導者は存在しない。そのため、自習時間に何をどのように学べば良いのかわからない学生が存在するはずである。したがって、そのような学生のために、自習環境を整備・支援する必要がある。

e-Learning サービスを作成する理由は、e-Learning によって、学生の学習の機会が増えるからである。e-Learning のメリットは、松田 (2006) によると、「いつでもどこでも学べる」ことが挙げられている。つまり、学習者は時間や場所の制約を受けずに学習できるようになる。そのため、学習者の繰り返し学習する機会が増えるため、学習時間の増加が期待できる。また、学習時間が増えることに伴って、学習者の講義内容への理解が促進されることも期待できる。

さらに、本学は e-Learning を導入しやすい環境が整っている。本学の学生は 1 人 1 台コンピュータを保持しているため、e-Learning を利用する環境が整っている。さらに、e-Learning を用いる講義が存在しているため、本学の学生は e-Learning への抵抗感が少なくスムーズに操作できると考えられる。よって、本学において e-Learning を導入することは適当であると考えられる。

以上のことから我々は、e-Learning サービスを構築し、本学の学生の数学自習環境の整備・支援を行うことをプロジェクトの目的とした。

(※文責: 南部 優太)

1.3 昨年度の活動

本プロジェクトは、2015年度から引き継いで行われてきたプロジェクトである。ここでは、とりわけて昨年度、つまり2016年度の活動を振り返る。2016年度の活動を振り返る理由とは、2016年度のe-Learningサービスの問題点を見つけることで、今年度の活動で改善すべき内容を発見できると考えたからである。2016年度のプロジェクトでは、本学1年生に対して数学用語の理解の補助を目標として活動を行った。これを目標とした理由は、プロジェクトメンバーが数学を学習した際に、教科書に存在しない用語を使うなど用語を正しく理解していない言動が多く見られたためであった。「数学用語を理解する」という状態は、「定義の意味、定理の証明や使い方を説明でき、それらを正しく使える状態」と定義していた。

成果として数学用語の理解を補助する学習支援サイト「ますますたでい2016」を作成した。これは、2015年度の成果物の「ますますたでい2015」を踏襲し改善したものであった。ますますたでい2016では、設問に対して誤答をしてしまった場合には、理解を補助するグラフなどを表示することにより、数学用語を正しく理解しないまま学習を進めてしまうという事態を防いでいた。また、チェックテストという予備問題を用意することで、設問を解くための前提知識を、学習者にあらかじめ習得してもらっていた。

2016年度の活動を調べていく過程で、次の3つの問題点を発見した。

1つ目の問題点は、ますますたでい2016は使用できる場所が限られていたということである。ますますたでい2016は学内ネットワークのみからアクセスできるものであり、大学の外では利用することができないウェブサイトであった。つまり、e-Learningサービスのメリットであった「いつでもどこでも使える」という部分が失われていた。

2つ目の問題点は、ますますたでい2016に用意されていた設問の数が、3問という少ない数であったことである。この問題点は、2016年度の最終報告書においても言及されており、「多くの範囲で使えると嬉しい」、「もっと多くの問題演習ができればいいと思った」というようなユーザーの声に対応することを、今後の展望として挙げていた。

3つ目の問題点は、学習者1人で学習できるサービスではなかった点である。ますますたでい2016は勉強会のように、学習者と支援者が同一の空間にいることを前提として作られたサービスである。そのため、学習者がますますたでい2016を使用している際、わからない数学用語が現れると、学習者が支援者に質問し、支援者がそれに回答するという形をとっていた。つまり、ますますたでい2016は学習者が1人で学習することを対象としていないサービスであった。よって、学習者がこのサービスを利用する機会は限定されていた。

以上の3つの問題点は、サービスのコンセプトや目標に依存するものではなく、サービスの形態によって発生した問題点である。よって、今年度のプロジェクトの目標に関わらず、実装班として上記の3つの問題点は改善したいと考えた。

(※文責: 南部 優太)

第 2 章 前期の活動

前期の活動では、「プロジェクト内学習会」「解析学 I 勉強会」「中間発表」を行った。2.1 節では、数学学習における問題点を探るために行ったプロジェクト内学習会について述べる。2.2 節では、プロジェクト内学習会で立てた仮説が、1 年生に対して仮説が成り立つのかを調べた解析学 I 勉強会について述べる。2.3 節では、各プロジェクトが前期の時点での進捗を報告しあい、共有する場である中間発表会について述べる。

なお、プロジェクト内学習会の詳細については、コンテンツ班の報告書に書かれている。また、解析学 I 勉強会の詳細については、検証班の報告書に書かれている。

(※文責: 後藤 春奈)

2.1 プロジェクト内学習会

本学の学生の学習支援を行うために、我々は 2 つのを行う必要があった。1 つ目は、プロジェクトメンバーの数学学習における問題点を把握することである。2 つ目は、数学への理解を深めることである。

これらを達成するためにプロジェクト内学習会を行った。プロジェクト内学習会では、メンバーが数学の問題を解くことで、学習習慣の問題点を見つけつつ、数学への理解を深めた。内容を表 2.1 にまとめた。表 2.1 に記載されている教科書とは、上見鍊太郎ら (2015) の書籍『微分』のことである。以降の教科書という言葉は、書籍『微分』を指す。

表 2.1 プロジェクト内学習会の形式や問題など

実施した回数	7 回
実施した問題	<ul style="list-style-type: none"> ・ $\sin(x)$ が連続であることを示せ ・ 『微分』 p23 練習問題 1.6-1 (1) ・ 『微分』 p23 練習問題 1.6-2 の 3 問及びここから派生した問題
勉強会の形式	まず全員が教科書の設問を解いてくる。その後、一人が、プロジェクトメンバーと教員の前で解答の説明をする。その説明を受けて、他のメンバーはディスカッションを行う。
解答の作成方法	解答のみを書くのではなく、次の 2 つを明記した。 <ul style="list-style-type: none"> ・ 解答を作成するにあたり自分が考えた内容 ・ 解答を書くにあたり参考にした教科書のページやインターネットのページ

プロジェクト内学習会を通して、学習習慣に関する以下の 2 つの仮説を立てた。

- プロジェクトメンバーが数学の教科書を「読め」ていないということは、本学の1年生も同様に教科書を「読め」ていない可能性がある
- プロジェクトメンバーが問題を解く際にインターネットを利用しているということは、本学の1年生も同様にインターネットを利用している可能性がある

ここでの「読む」とは、一般的な動詞の読むとは異なり、プロジェクト独自で定義した一連の手順である。その手順とは「問題を解く過程でわからない数学用語があれば、教科書の該当ページに戻る。そして、その数学用語に関する記述の記号や言葉の意味を理解する。もし、その該当ページにわからない数学用語があれば、さらにその数学用語のページに戻って理解を深めていく」というものである。

また、問題を解く際にインターネットを利用しているプロジェクトメンバーがいることがわかった。設問を解き進めることができなくなった時、教科書は読みにくく難しいという理由から教科書を使わず、インターネット検索を行っていた。インターネット上には正しくない情報もある。それにもかかわらず、情報の真偽を判定できずに正しい情報だと思い込み、自身の解答へと流用していた。その結果、プロジェクトメンバーは不正確な情報を含む解答を、正しい解答のつもりで提出してしまっていた。

(※文責: 後藤 春奈)

2.2 解析学 I 勉強会

前節の仮説は、あくまでプロジェクトメンバーの学習習慣から言えることである。仮説が本学の1年生に対しても成り立つかどうかは検証する必要がある。その検証のために実施したのが解析学 I 勉強会である。

解析学 I 勉強会は、我々が用意した設問を、参加者に解いてもらう問題演習形式で行われた。また、解析学 I 勉強会が参加者にとってどのような効果があるのかを調べるために、勉強会の前後でアンケートも行った。勉強会当日において、プロジェクトメンバーは、チューターという役割を担うこととした。チューターの役割は、参加者からの数学に関する質問に対して、教科書を用いて説明を行うことである。また、参加者が数学の問題を解く様子を観察し、気になった行動をメモするという役割があった。

解析学 I 勉強会の参加者は合計 42 名であった。そのうち、アンケートに回答した参加者は 41 名であった。

まず、勉強会の前後のアンケートの結果から、参加者が教科書を読んでいるかを検証する。勉強会前アンケートにて「課題を解く際に、何を参考にしますか (複数回答可)」という質問についての回答の結果が図 2.1 である。

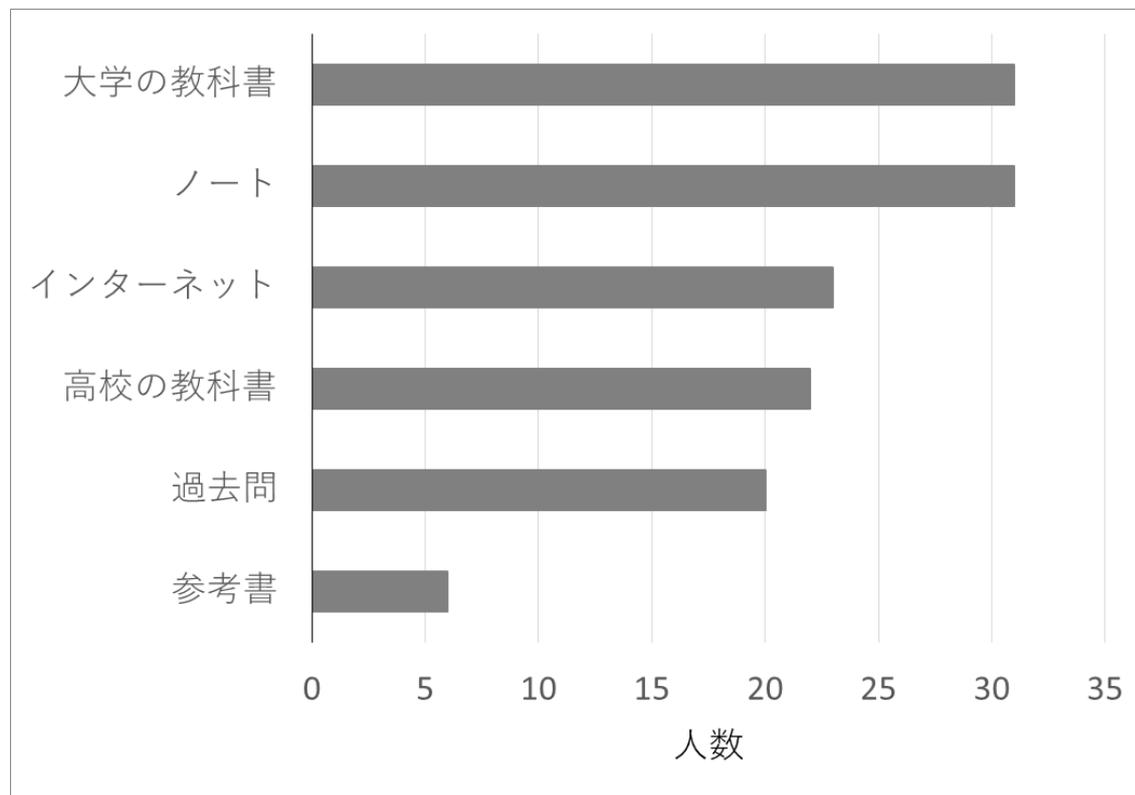


図 2.1 解析学 I 勉強会の参加者が課題を解くときに参考にするもの

「大学の教科書」と回答した勉強会参加者は 41 名中 31 名いたことから、解析学 I 履修者は「課題に困った際に大学の教科書を利用した方が良い」と思っていることが読み取れる。また、インターネットを参考にする学生が 41 名中 23 名と半分以上いたこともわかった。これは、「問題を解く過程で困った場合、教科書を利用するのではなく、インターネット検索を行なっているのではないか」という仮説の、「インターネット検索を行なっているのではないか」という部分に合致する。

次に、勉強会前アンケートにて「. わからない問題に直面したときに、教科書のどのページを参照すればよいのかわかりますか」という質問についての回答の結果が図 2.2 で示す。

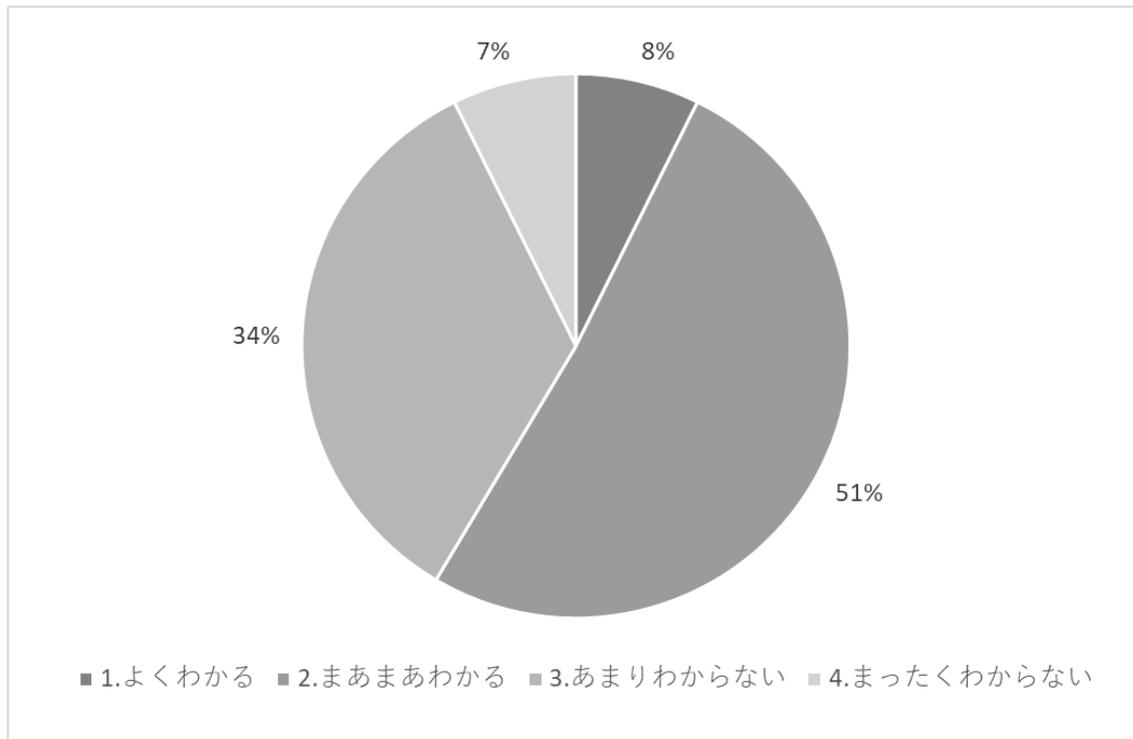


図 2.2 教科書のどのページを参照すればよいかわかるかの回答結果

「教科書」を学習リソースとして利用できているのは、「よくわかる」と「まあまあわかる」を合わせた 58.5%、つまり半数程度だということである。これより参加者は教科書を参考にしようとするものの、読むべきページがわかっていない人が半数いるということがわかる。

次に、「参加者の答案から読み取れること」と「チューターの観察の結果」について述べる。ここまでのアンケート結果は、参加者の主観的な評価である。そのため、客観的な結果が別途に求められるからだ。参加者から回収した答案は、プロジェクトメンバーが全てに目を通し以下の観点で分析した。

- 教科書に掲載されているように正しく語句を使えているか
- 記号の使い方は適切か
- 式変形に対して説明をしているか
- 結果だけでなく、途中計算や説明文が書かれているか

これらの観点は、プロジェクト内学習会で得た経験から作成したものであり、教科書を読むことができているかどうかを調べる重要な観点である。

上記の観点に照らし合わせて参加者の解答を見ると、教科書を「読む」ことができていないと思われる解答が複数存在した。具体的には、「 $x = 0 + 0$ のとき」や「 $x = 0 - 0$ のとき」といった記述である。これは、教科書には掲載されていないものであり、数学的に誤っている表現である。

チューターの観察の結果によると、「1 年生は教科書をパラパラめくるだけで、必要な情報が見つけれられないようであった」というような内容が複数のプロジェクトメンバーから報告されていた。これによって、図 2.2 では「教科書のどのページを参照すればよいかわかる」と答えた 58.5% の参加者は本当はわかっていない可能性が生まれた。つまり、参加者は教科書を読んでいるつもり

であるが、プロジェクトメンバーから見ると教科書を「読め」ていないと推測される。

また、チューターの観察の結果には、教科書を「読む」ことが出来ていなかったという記述だけでなく、参加者の勉強の仕方についても記述があった。その中で特筆すべき事項は、「勉強会参加者の多くはスマホを用いていた」ということである。つまり、参加者はインターネット検索をする際にスマートフォンを利用していることがわかった。

解析学 I 勉強会の結果と考察から 2 つのことがわかった。1 つ目は、1 年生もプロジェクトメンバーと同様に教科書を読めていなかったことである。2 つ目は、1 年生はスマートフォンを使用して、インターネットを参考にして学習しているということである。そこで、1 年生にとって触れやすい e-Learning サービスを用いて、教科書を「読む」支援を行えたらいいのではないかと考えた。さらに、スマートフォンに対応することによって 1 年生にとってより使いやすいものになるのではないかと考えた。

(※文責: 後藤 春奈)

2.3 中間発表会

中間発表会とは、各プロジェクトが前期の時点での進捗を報告しあい、共有する場である。主にポスターセッション形式で行われるが、スライド発表形式で行うプロジェクトも存在する。ここでは、その中間発表会に向けておこなわれた準備と、中間発表会の結果および考察について述べる。

(※文責: 石村 秀翔)

2.3.1 中間発表会の概要

中間発表会を行うにあたって、本プロジェクトでは 2 つの班に分かれて活動を行なった。1 つは、発表にて使用するスライドを作成する「スライド班」である。もう 1 つは、ポスターセッションにおいて使用するポスターを作成する「ポスター班」である。

まず、スライド班はこれまでの活動を振り返り、骨組みとなるストーリーを考えた。その後、それを元にスライドを作成した。スライド作成の過程では、何度もプロジェクトメンバーや指導教員からレビューを頂いた。そして、レビューをもとに修正を行い、完成させた。

ポスター班は、スライド発表では話すことができない内容も含めて、スライドより詳細な内容を含むポスターを作成した。作成の過程では、スライド班と同様に何度もプロジェクトメンバーや指導教員からレビューを頂き、修正を行った。

中間発表会当日は、前半と後半にメンバーを分け各 3 回ずつ、計 6 回の発表を行うこととした。また、中間発表会では発表評価シートを聴講者に対して配布し、記入して頂くことになっていた。発表評価シートとは発表技術と発表内容の 2 つの項目について、聴講者に評価をして頂くためのものである。それぞれについて、「1(非常に悪い) から 10(非常に優秀)」の 10 段階評価と、自由記述による評価を設けた。

(※文責: 石村 秀翔)

2.3.2 中間発表会の結果と考察

本プロジェクトの中間発表では、前述した「プロジェクト内学習会」、「解析学 I 勉強会」でわかったことについて発表した。また後期には、教科書を「読む」ことを促し、「読む」ことを支援する e-Learning サービスを作成することを示唆した。

中間発表会当日は、学生と教員を含む 76 名の評価を頂いた。ただし、評価シートの一部に記入がなされていないものが存在していたため、その項目に関しては無効とした。以下、評価結果について記述する。

発表技術についての数値評価は、平均 8.21（有効回答数 $n = 72$ 、標準偏差 $SD = 1.23$ ）であった。

発表内容についての数値評価は、平均 7.73（有効回答数 $n = 66$ 、標準偏差 $SD = 1.42$ ）であった。

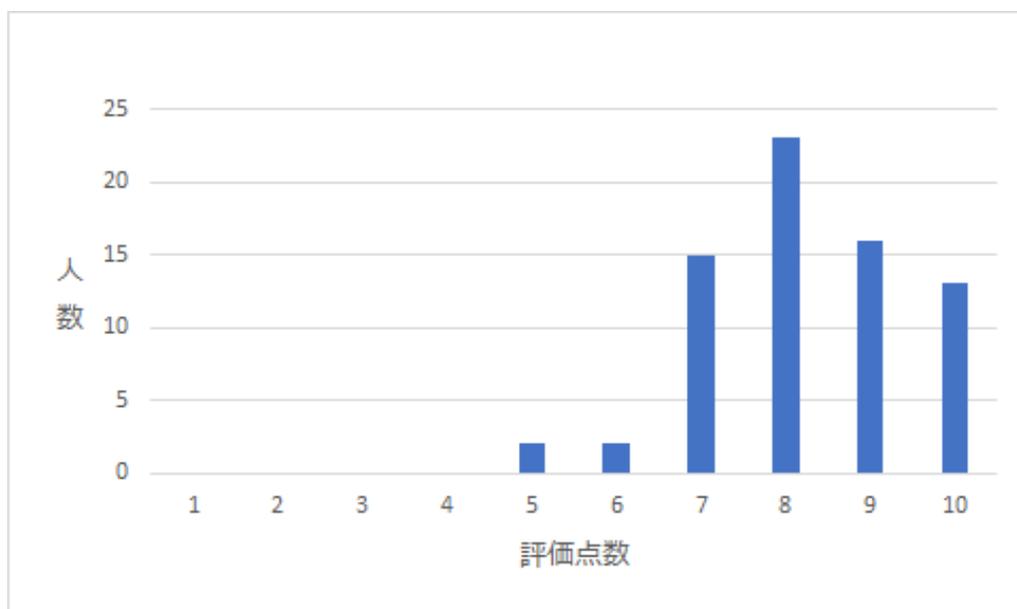


図 2.3 発表技術の度数分布表

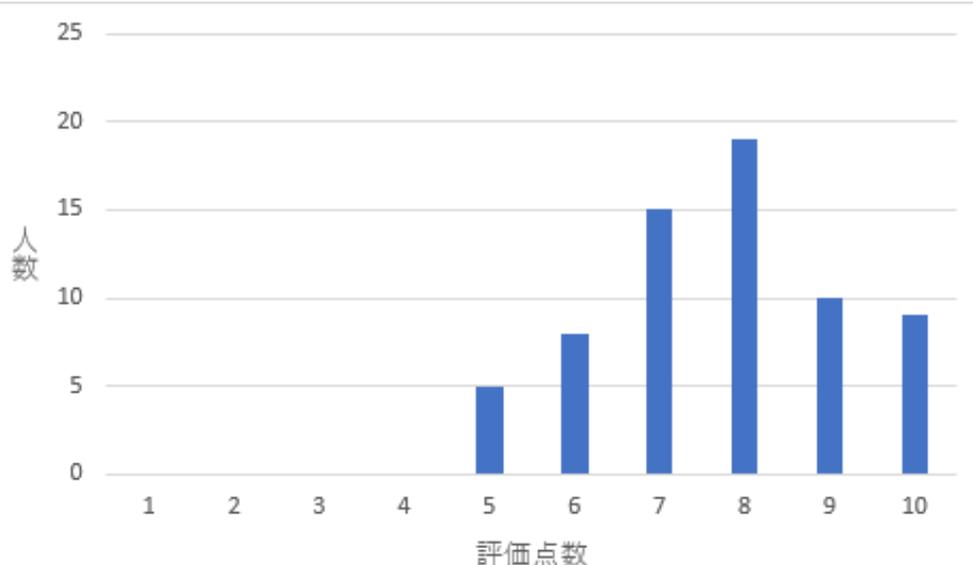


図 2.4 発表内容の度数分布表

自由記述で得られたコメントは以下に箇条書きでまとめる。コメントは、評価シートに多く書かれていたものを一部抜粋した。

発表技術

- 声の大きさや速さ、抑揚などがちょうどよくて聞き取りやすかった
- スライドが見やすかったので内容が理解しやすかった
- スライドを読んでいる印象を受けた
- 質問に答える声が小さく、聞き取りづらかった
- ジェスチャーを交えていたため、印象に残りやすかった

発表内容

- 前年度のプロジェクト内容をブラッシュアップし、より良いものを作ろうとする意図を感じた
- 課題の確認は良いが、具体的にどのようなサポートを行うのか分からなかった
- 目的が明確でよかったと感じた
- 『読む』ということについての定義づけをもっとしっかりするべきだと感じた
- 数学の補助教材を e-Learning で作るのは、難しいのではないだろうか

発表評価シートに基づいて我々の活動を振り返る。発表評価シートによると、発表技術と発表内容の平均点はそれぞれ「8.21」、「7.73」だった。技術面、内容面ともにおよそ 8 割の点数が得ることができた。以下では、評価の詳細について述べる。

まず、我々の発表の高評価を受けた点について述べる。発表技術の評価では「スライドが分かりやすかった」という意見が挙がった。学習会や勉強会で得られたデータからどんなことが言えるのか示すなど、論理的な構成になっていたことが理由として挙げられる。最終発表会で用いるスライドでも、「データ」→「論拠」→「論証」の流れで話を構成することを念頭に置きたい。また、発表内容の評価では「しっかり現状を調べて補助できるように考えてある」という意見が挙がった。今

後も、「学習者が教科書を“読んでいない”」という問題点を軸に、学習者のサポート方法を考えていく必要がある。

次に、反省すべき点について述べる。発表技術に関しては「スライドを読み上げている印象を受ける」という指摘が目立った。この指摘に関しては、話を展開する順序を把握し切れていなかったなど、発表の練習不足が原因として考えられる。練習不足になってしまった理由として、スライドが完成したのが発表会の前日だったこと、発表を担当するメンバーが決定したのが発表会の2日前だったことが挙げられる。最終発表会では、早い段階で準備に取りかかり、発表練習の時間を十分に設ける必要がある。さらに、発表内容に関して「これから何を作っていくのか具体的なビジョンが見えない」という指摘も目立った。この指摘に関しては、我々が学習会や勉強会を通して発見した“数学学習の問題点”をどのように解決していくのか、具体的な説明が欠けていたためだと考えられる。後期はWebコンテンツによって学習者をサポートする予定である。それにもかかわらず、具体的なサポート方法についてはプロジェクト内でも議論できていなかった。今後、学習者をサポートする方法について、プロジェクト内でよく議論する必要がある。

(※文責: 石村 秀翔)

第 3 章 後期の活動

2 章では前期の活動を通して、本学の 1 年生は教科書を「読む」ことができていなかったという問題を発見した。この章では後期の活動について述べる。後期の活動では、2 章で発見した問題を解決するために e-Learning サービス「ModoLuca」を作成した。次に、ModoLuca が教科書を「読む」ことを支援することが出来るのかどうかを調査するために ModoLuca の評価実験を行った。

(※文責: 知子澤 考輔)

3.1 e-Learning サービスの提案

ここでは、e-Learning サービスを作成するために行った活動について述べる。3.1.1 項では、作成する ModoLuca が持つべき価値を考案し、提案する活動について述べる。3.1.2 項では、考案した価値に基づいて作成したプロトタイプについて述べる。3.1.3 項では、プロトタイプに必要な設問の作成活動について述べる。

(※文責: 知子澤 考輔)

3.1.1 価値仮説の作成

後期の活動は「教科書を『読む』ことを促し、『読む』ことを支援する e-Learning サービス」の作成を行うことであった。しかし、これだけでは e-Learning サービスの売りがないので e-Learning サービスを作り始めることはできない。そのため、プロジェクトメンバー全員で価値仮説の考案を行った。価値仮説とは、システムのコンセプトを決めるとともに、システムの価値を訴える仮説であり、ユーザー、欲望、課題、製品の特徴の 4 項目で構成される。

- ユーザー (A) : システムの対象とする利用者のこと
- 欲望 (B) : ユーザーが望んでいるであろう事
- 課題 (C) : 欲求を満たすための障壁となっていること
- 製品の特徴 (D) : 課題を解決できるような、システムが満たすべき特徴

以上の 4 項目を組み合わせた文章を価値仮説と定義した。プロジェクトメンバー全員が出した価値仮説を元に議論を行った。その結果、決定した価値仮説は次の内容となった。

- ユーザー : 本学の解析学履修者
- 欲望 : 教科書を用いて解析学の知識を身に付けたい
- 課題 : 教科書を用いた数学用語の調べ方や、数学用語の数学的な概念を理解する重要性を知らない
- 製品の特徴 : よって我々が発見した「読む」という学習方法を学習者に提供することは価値がある

この価値仮説は、解析学 I 勉強会の結果に基づいて作成した。解析学 I 勉強会で 1 年生は教科書を自発的に開くが、「読む」ことができていないことが判明した。また、解析学 I 勉強会を通して 1 年生は教科書を参考にしていくと回答している人が増加した。このことから、1 年生は教科書を用いて解析学の知識を身に付けたいだろうと考えた。この欲求に我々の「読む」という学習方法を提供することは価値があることだと考えた。

(※文責: 卯子澤 考輔)

3.1.2 プロトタイプの作成

決定した価値仮説をもとに、プロトタイプを作成した。プロジェクトメンバーを 3 チームに分けて、1 チームにつき 1 つのプロトタイプを作成した。3 チームのうち 2 チームの内容は似ていたため、統合して合計 2 つのアイデアが生まれた。1 つ目のアイデアは 2016 年度の「ますますたでい 2016」を踏襲する形となるサービスである。2 つ目のアイデアは、対話形式で、ある数学用語が載っている教科書のページを学習者に教えることにより、学習者に「読む」ことを誘導するサービスである。どちらの案を採用するか議論した結果、2 つ目のアイデアを採用することが決定した。決定した理由は 2016 年度までのサービスとは違うという新規性と、コンテンツとして組み込まれる数学の設問作成の手間が小さいからである。

このサービスの根幹をなす機能はチャット機能である。チャット機能とは、学習者がわからない数学用語をテキストとして入力すると、「その A という単語は教科書の x ページに載っていますよ」というように対話形式で教えてくれる機能のことである。その後、A という単語を学習者が教科書を用いて調べ、理解したかどうかを確認する質問を投げかける。もし、その質問に対して誤った回答をしてしまった場合には、「A という単語を理解していないようですね。A という単語を理解するためには、B や C という単語を理解する必要があります。わからない単語はありますか？」と質問を学習者に投げかける。そこで学習者が B と答えた場合、「その B という単語は教科書の y ページに載っていますよ」と教え、A のときと同じように、B という単語を学習者が教科書を用いて調べ、理解したかどうかを確認する質問を行う。この質問に誤った回答をした場合、先ほどと同様に「B という単語を理解していないようですね。B という単語を理解するためには、D や E という単語を理解する必要があります。わからない単語はありますか？」と質問を投げかける。

この対話を繰り返し、学習者がわかるようになるまで、教科書を遡って調べることを促す。この繰り返しの仕組みは、我々が提唱する「読む」という学習方法である。また、この繰り返しによって学習者が十分な理解を得たと判断した場合のみ、模範解答を学習者に見せる。

このサービスには、「スマートフォンとの親和性の高さ」、「失敗からの学び」、「設問作成コストの低さ」の 3 つのメリットがある。

「スマートフォンとの親和性の高さ」とは、対話形式とスマートフォンの相性の良さを指している。スマートフォン向けにサービスを作成するときにつきまとう課題は、描画領域が狭いことである。スマートフォンはパソコンに比べると画面が小さいため、一度に表示できる情報量が少ない。そこで、情報を表示できる量を増やす単純な方法として、ページを縦に長く作成しユーザーにスクロールを促すという方法がある。長いスクロールを要求するページは操作性が悪い。しかし、Skype や LINE のようなチャットアプリでは、長いスクロールが当たり前となっている。近年では、これらのアプリケーションを利用する大学生は多い。よって、新規に作成する e-Learning サービスも対話形式のチャットアプリにすれば、学習者が負担無く e-Learning サービスを利用

きると考えた。

「失敗からの学び」とは、e-Learning サービスを使うユーザーが失敗からの学びをすることが期待できるということを指している。2016 年度の成果物は、設問 1 つに対してチェックテストと呼ばれる小問を複数用意することで学習者に予備知識を与えておき、その設問を間違えないようするという形式をとっていた。しかし、今年度のプロジェクトメンバーは「間違える」という行為から多くのことを学んできたため、学習者にも積極的に間違えて欲しいと考えた。そこで、このサービスは学習者に行き詰まるまで 1 人で設問を解いてもらい、どうしても次に進めなくなったときだけ質問してもらおうというコンセプトになっている。

「設問作成コストの低さ」とは、1 つの設問を作成することにかかるコストが低いということを示している。2016 年度の e-Learning サービス「ますますたでい 2016」は、先ほど述べたように 1 つの設問に対して複数の小問を用意していた。それによって、問題作成コストがかさみ 3 問しか作成することができなかった。しかし、ここで提案したサービスは、あくまで学習者に教科書のページ数を教えて、理解度チェックとなる質問をするだけなので、問題作成コストが 2016 年度ほど高くない。よって、2016 年度より多くの設問を用意できることが期待できる。

以上の理由により、対話形式で教科書を「読む」ことを促すサービスの実装が決定した。

(※文責: 知子澤 考輔)

3.1.3 設問の作成

3.1.2 項で決定したプロトタイプの実装を実現するためには、システムの実装とサービスにコンテンツとして組み込まれる数学の設問が必要である。システムの実装は実装班が行い、設問の作成はコンテンツ班が行った。設問作成の詳細はコンテンツ班のグループ報告書を参照していただくこととする。

(※文責: 知子澤 考輔)

3.2 システムの実装

この節では、e-Learning サービスの実装の過程を述べる。3.2.1 項では、実装するにあたって必要となる、データフローやモジュールなどシステムアーキテクチャの定義について述べる。次の 3.2.2 項からは、各モジュールの要件定義について述べる。最後に 3.2.5 項では、実際の実装過程とそのスケジュールを述べる。

(※文責: 南部 優太)

3.2.1 システムアーキテクチャの考案

3.1 節で決定したプロトタイプはアイデア段階のものであり、具体的な実装方法が提案されていない。そこで、決定したプロトタイプをシステムとして実現するにあたって、システムアーキテクチャを考案する必要があった。我々が作成するシステムは、以下の 5 つの機能を実現できるものでなければならない。

1. スマートフォンによってアクセスできること
2. 数学の設問選択ができること
3. 模範解答を表示できること
4. 対話形式の応答ができること
5. 数学用語と関連データを保持しておき、適宜に対話形式で出力できること

以上の5つを満たすシステム構成として、アプリケーション、Bot サービス、データベースという3つのモジュールが必要であると結論づけた。アプリケーションは、学習者がスマートフォンからアクセスするモジュールそのものである。また、このアプリケーション内で、設問の表示や模範解答の表示を行いたいと考えた。アプリケーションの候補としては、ネイティブアプリケーションやWebアプリケーションなどが挙げられる。Bot サービスは、問い合わせに対して自動で応答するサービスのことである。これは、製作しようとしているe-Learningサービスの根幹を担うものであり、必ず必要なものである。データベースは、数学用語の関連データを保持しておくためのものである。上述したBotサービスがこのデータベースにアクセスし、データを受け取り自動応答にそれらのデータを含めることで、数学用語の関連データを出力することを実現する。以上の内容をデータフロー図で表すと図3.1になる。

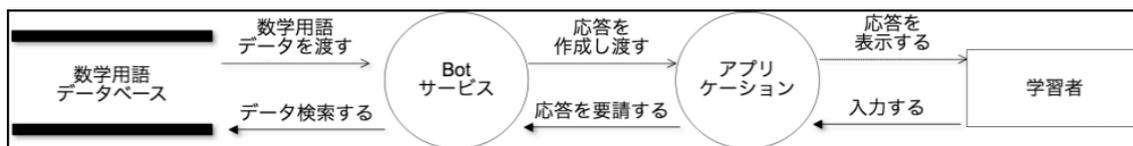


図 3.1 システム構成案のデータフロー図

次に、3つのモジュールをどのような形態で構築するかを決定した。その結果、アプリケーションはモバイルウェブアプリで、Bot サービスは既存のサービスで、データベースはNOSQL形式で構築することとした。以下にその詳細を示す。

アプリケーションは、モバイルウェブアプリとして実装することとした。アプリケーションは選択肢として、ネイティブアプリとモバイルウェブアプリという2つがあった。ここでモバイルウェブアプリを採用した理由は、手軽さが重要であると考えたからである。今回のシステムのコンセプトには、「手軽にe-Learningサービスを利用してもらうことで、教科書を用いて学習することへの敷居を下げたい」というねらいが存在する。ここで、もし実装するものがネイティブアプリである場合、アクセシビリティが悪い。なぜなら、使用する前にApple StoreやGoogle Play Storeを経由したアプリケーションのインストールが必要となるからである。また、スマートフォンの各種OSに対応する必要があり、クロスプラットフォームでの開発が求められる。よって、インターネットブラウザさえ所持していれば、誰でも閲覧することのできるモバイルウェブアプリを採用した。

また、モバイルウェブアプリは、HTML、CSS、JavaScriptという基本的な構成で作成することとした。このようにした理由は、サーバーサイドプログラムの知識を持つメンバーが少ないためである。つまり、Node.jsなどを用いた開発は避け、基本的な最小構成を取ることがベストであると考えられた。また実装を行う際には、一からプログラムを書くのではなく、既存のフレームワークを利用するのが良いと考えた。短い工期で開発を行うためには、既存のサービスやフレームワー

クを利用することは妥当であると考えられる。そうしたとき、比較的古くから存在する、HTML、CSS、JavaScript は学習リソースやフレームワークの量が豊富であり、短期開発に向いていると判断した。

Bot サービスは、既存のサービスを利用することとした。Bot サービスを一から製作することは、技術と工期という2つの側面から不可能であると考えられた。そこで、すでに提供されている Bot 開発サービスについて情報収集を行った。その結果、LINE 株式会社が提供する「LINE Bot」や GitHub 社が提供する「Hubot」、Microsoft 社が提供する「Bot framework」などが存在していた。拡張性や作成難度などを評価し、Microsoft 社の Bot framework を採用する見込みとした。

データベースは、NOSQL 形式のものとした。プロトタイプのアイデア段階であったように、データベースには、数学用語とその関連データが格納される。その際、数学用語と関連データを用意するのは本プロジェクトのコンテンツ班の役割であった。工期の都合上、システムの開発とそのコンテンツの作成は、並列での作業となった。このような分業形態において予想されるリスクは、我々実装班が必要としているデータと、コンテンツ班が用意したデータに齟齬が生じることである。ここで、データベースに RDB 形式のものを採用した場合を考える。RDB 形式のものはスキーマが変更できないため、途中で仕様変更があった場合、テーブルごと作り直しになる。しかし、NOSQL 形式ならこの問題に柔軟に対応できる。なぜなら、NOSQL はスキーマレスでありカラムを自由に定義することができる。よって格納の形式を JSON 形式にすることによって、突然の仕様変更があったとき、少なくともデータベースの内部を作り直す必要はない。

(※文責: 南部 優太)

3.2.2 開発環境の選定

次に、3つのモジュールの開発環境を選定した。その結果、サーバーの管理・Bot サービスの開発・データベースの構築という3つを「Microsoft Azure」で行うことにした。Microsoft Azure とは、Microsoft 社が提供しているクラウドサービスである。OS、サーバーや記憶装置、ネットワーク機器などのインフラをインターネット経由で提供している。

Microsoft Azure を利用した主な理由は3つある。

1つ目は、サーバー管理の容易さである。我々がモバイルウェブアプリを作成するにあたって、サーバーサイドの開発が困難になると予想された。理由は前述したように、今回のプロジェクトメンバーにおいて、サーバー管理に詳しいメンバーは少なく、難航することが見込まれたからである。また、2016年度の活動履歴を調べたところ、2016年度のプロジェクトメンバーは、ローカルの機体をウェブサーバーとして利用できるよう、開発・運用・保守を行っていた。その結果、多くの手間と時間を要しているようであった。しかし、Microsoft Azure のクラウド環境を利用すると、サーバーのセットアップに時間をかけなくとも済む。なぜなら、サーバー環境は事前に構築されているため、我々は作成した HTML 等のプログラムを wwwroot ディレクトリに配置するだけで良いからである。

2つ目は、サーバーを学内ネットワークから分離することができるからである。2016年度に利用したサーバーは、担当指導教員の研究室に設置されていた。また、学内のネットワークのみからアクセス可能な状態にあった。これには理由がある。2016年度に作成されたシステム「ますますたでい 2016」は、教科書の内容を一部含むものであった。よって、際限なく外部に公開してしまうと、著作権法に抵触する恐れがあった。そこで、2016年度は学内のネットワークのみからアクセスさせることで、教科書を確実に所持している人のみが閲覧できるようにしていた。

この著作権法の問題は、今年度においても発生する。なぜなら、我々が想定しているシステムも教科書の内容を部分的に含むからである。しかし、我々のコンセプトを踏まえると、学内ネットワークのみからアクセスできなくするのは、好ましくない。なぜなら、今年度は場所を選ばずに利用できるシステムを目標の1つとして掲げているからである。そこで、我々は Microsoft Azure のクラウドサーバーにウェブページをアップロードし、ログイン機能をつけることで対応することとした。クラウドのウェブページをアップロードすることで、学内ネットワークから分断することができる。また、我々がユーザーをあらかじめ登録しておき、ログインをした場合のみウェブページを閲覧できるようにすることで、著作権の保護を行うこととした。

3つ目は、Bot サービスとデータベースの両方を提供しているからである。Bot サービスとデータベースは連携が求められる。開発の際に、Bot サービスを開発するプログラミング言語とデータベースを操作するプログラミング言語が異なると O/R マッピングなどに手間がかかる。しかし、Microsoft Azure では、Bot 開発サービスとして Bot framework と、NOSQL 形式のデータベースとして CosmosDB が提供されている。どちらも Microsoft Azure によって提供されているため、親和性が高く、連携が容易であると考えられた。

以上の理由から、Microsoft Azure を利用することは、技術的コスト・時間的コストという両方の側面からみて、適切であると考えられた。

(※文責: 南部 優太)

3.2.3 モバイルウェブアプリの要件定義

まず、モバイルウェブアプリが持つべき機能の書き出しを行った。書き出した機能は、主に以下の6つである。

1. ログイン機能

ログイン機能とは、我々が承認した人のみがシステムを利用できるようにするための仕組みである。この機能は、3.3 節で述べたように、著作権上の観点から設けたものである。この機能の実装には、簡便さを考慮して Google が運営している「Firebase」を用いることとした。Firebase は、スマートフォンアプリにおける汎用的な機能をクラウドから提供するサービスの1つである。

2. 基本情報表示機能

基本情報とは、システムコンセプトや使い方、本プロジェクトの情報などである。これらの情報は頻繁に読むものではないため、通常時は隠しておき学習者がメニューボタンを押したときのみ、見られるようなページ構成とする。

3. 設問表示機能

我々が提案するサービスでは、複数の数学の設問を用意し、その中から好きなものを学習者に選んでもらい学習してもらう。そのため、設問表示機能が必要である。各設問が持つべき情報は次のものとした。

- 設問番号：設問を識別するためのユニークな数字
- タイトル：設問の出題意図を簡易に示すテキスト情報
- サムネイル：アイキャッチの役割を果たす画像
- 単元：教科書『微分』の、どの単元に位置するかを示すテキスト情報

- 重要語句：この設問を学ぶ際に必要となる数学用語を集めたテキスト情報
- 問題文：教科書『微分』から引用した設問そのもののテキスト情報
- ねらい：この設問を解くことで、学習者にできるようになってほしいことを示すテキスト情報

4. 対話 UI 表示機能

対話形式で数学を学ぶというアイデアを実装するにあたって、対話を行うためのインターフェースが必要になる。そこで、我々は Microsoft 社の「Microsoft Bot framework」が持つ、埋め込みタグを用いることにした。Microsoft 社が提供する iframe タグを埋め込むだけで、UI 部分をウェブページに埋め込むことができる。対話に適した UI は、対話という機能の性質上から動的であり、短い工期で作成することは難しい。しかし、「Microsoft Bot framework」を利用することで容易に実装できると考えた。

5. メモ機能

この機能は、学習者が数学用語などをメモする機能である。我々が提案するサービスでは、学習者が数学用語について質問すると、教科書の該当ページと関連用語を教えてくれる機能が根幹をなす。しかし、対話形式の性質上、1 度に複数の質問を行うことはできない。よって、学習者が質問したいことを忘れないためにメモ機能を実装することとした。これは、簡易なテキストボックスにて実装する。

6. 模範解答表示機能

Bot サービスとの対話の内容によって、状態が変わる解答表示機能が必要である。学習者は、対話形式でわからない数学用語を質問し学習する。その過程で、ある特定の単語（キーワード）について学習した場合のみ模範解答を表示したい。また、模範解答を分割しておき、キーワードと模範解答を多対一対応させておくことで、複数のキーワードを学習しなければ模範解答を全て見ることはできない仕組みとしたい。つまり、Bot サービスと協調しながら、ウェブページの内容を動的に変更する必要がある。そこで、その仕組みを実装する手段として、Cookie を用いることとした。学習者がキーワードを学習したとき、Bot サービスは Cookie に記録を行う。そして、ウェブページは読み込まれるたびに Cookie から情報を取得し動的に模範解答の表示範囲を変える

次に、レイアウトの作成や画面遷移図の作成を行った。UI は主にスマートフォン向けに作成することとした。そのため、ページ遷移が少なくなるように設計し、文字数が多くなりすぎることや、文字が小さくなりすぎること避けることにした。レイアウトと画面遷移図 (図 3.2) の作成は、Goodpatch 社が提供する「Prott」というサービスを用いて行った、図 3.2 がそこで作成した画面遷移図である。また、デスクトップ向けに、スマートフォン環境を再現する iframe タグ埋め込んだ HTML ページも作成する。これらのウェブページは、「Microsoft Azure」にて提供されるクラウドサーバーにアップロードする。クラウドサーバーサービス「App Service」における設定内容は以下の表 3.1 にまとめた。

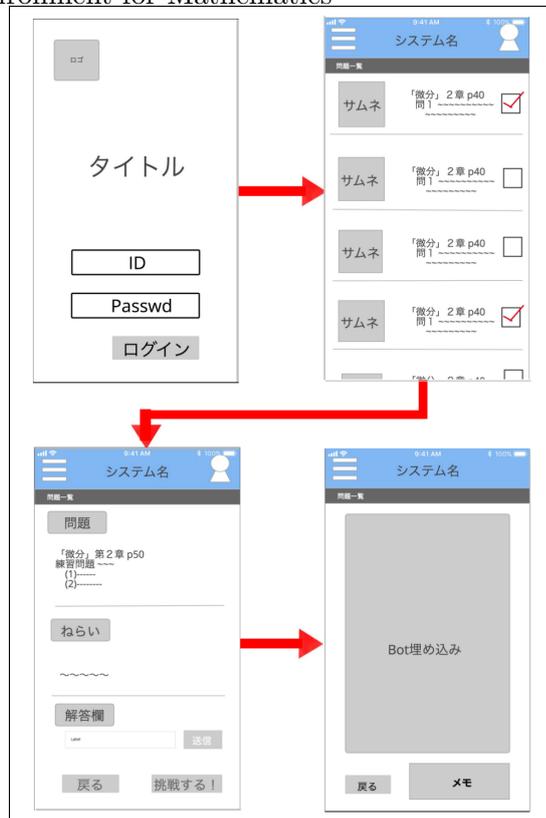


図 3.2 Prott で作った画面遷移図

表 3.1 Azure クラウドサーバーの構成

App Service プラン	F1FreeJapanEast (Free: 0 S)
OS 名	Windows Server 2016
場所	Japan East
URL	https://mathpromobile.azurewebsites.net

(※文責: 南部 優太)

3.2.4 Bot サービスの要件定義

Bot サービスの実装内容は、3.4 節で作成した会話モデルのフロー図をプログラムとして作成することがメインである。「Microsoft Bot framework」の開発環境は、「Visual Studio 2017」、開発言語は「C#」である。「Microsoft Bot framework」を用いた Bot 開発では、待機状態と返答内容を作る必要がある。待機状態というのは、状態遷移図における分岐にあたるものである。つまり、相手から受け取った情報を処理して、どのような返答を返すことが適切か判定するものである。「Microsoft Bot framework」はユーザーの入力を POST リクエストによって受け取り、その返答を Http レスポンスで返している。つまり、Bot サービス自体は POST リクエスト受け付けるだけの WebAPI であり、状態を保持する機能は持たない。しかし、「Microsoft Bot framework」には Bot ディレクトリという機能があり、簡易なデータであれば Cookie のように、辞書データ型で情報を保持することができる、これによって、Bot サービスの状態を保持することができる。よって、状態とユーザーの入力という 2 つの情報から返答を変えることができる。例えば、状態 1 のと

きに「こんにちは」という入力があったら「こんにちは」と返し、状態2のときに「こんにちは」という入力があったら「どの数学用語がわかりませんか?・・・」と返答を切り替えることができる。返答内容というのは、相手にどのような返答をするかという部分である。先ほどの例で言えば、「こんにちは」や「どの数学用語がわかりませんか?・・・」というものが返答に当たる。返答を作成すること自体は、専用の関数がフレームワークによって提供されており、難しいことではない。難しいことは、返答する内容をデータベースに問い合わせることである。求められる機能は、「学習者がつまずきそうなポイントをケアする提案する機能」と、「数学用語を質問されたときに返答する機能」である。これらの機能はいずれもデータベースへの問い合わせが必要となる。両機能の実装案は以下の通りである。

- 「学習者がつまずきそうなポイントを提案する機能」の実装案

学習者がつまずきそうなポイントというものは、設問に依存する。例えば、極限の問題では「極限值がわからない」などがこれに該当し、微分の設問を解いている過程では「導関数がわからない」などが該当する。よって提案する数学用語は設問によって異なる必要がある。データベースには設問ごとに「学習者がつまずきそうなポイント」を保存している。よって、設問番号をキーとしてデータベースに問い合わせ、その結果を返答に含めている。

- 「数学用語を質問されたときに返答する機能」の実装案

学習者から数学用語を質問されたとき、最初にその単語名でデータベースに問い合わせを行う。そこで、合致するデータがなかった場合、その旨を返答として返す。合致するデータが存在した場合、その数学用語が掲載されている教科書のページと簡易な質問がデータベースに保存されているため、それを含めて返答する。学習者が質問に対して間違ったときは、その数学用語を学ぶために必要な他の数学用語を提示する。

また、学習者のニーズを汲み取る手段として、学習者が Bot サービスに質問した数学用語を保存しておくことにした。学習者が数学用語を質問したときに、その単語がデータベースに保存されていなければ学習者に支援を行うことはできない。つまり、学習者が質問したにも関わらず支援できない数学用語が存在するということである。それらの「未対応数学用語」を集積することで、学習者にはニーズがあるにも関わらず我々が用意することができなかった数学用語を知ることができる。よって、データベースに保存されていない数学用語を学習者から入力されたときには、データベースに保存することにした。

(※文責: 南部 優太)

3.2.5 データベースの要件定義

データベースには、つまずきポイントと数学用語を格納する。つまずきポイントとは、学習者が設問を解く過程でつまずくであろうと予想されるポイントのことである。データベースは、前項で述べたように、Bot サービスが返答を作成する際に参照するために用いる。機能としては、学習者に「今、どこでつまずいてしまっていますか?次の中から選んでください。」と質問し、学習者に選択してもらったら「そこで詰まるということは、この数学用語を教科書で調べた方が良いでしょう」というようにアシストしたい。そこで、学習者がどんなところでつまずくかをあらかじめ保存しておく必要がある。これが、つまずきポイントをあらかじめデータベースに保存しておく理由である。当然、そのあとに提示する数学用語も格納しておく必要がある。また、つまずきポイントはコンテンツ班が数学の問題をあらかじめ解いておき選定した。

Bot サービスが参照するデータを考慮して、データベースに格納する形式を考えた。データベースは、スキーマレスである方が柔軟であると考え、NOSQL 形式の「CosmosDB」とした。「CosmosDB」は、「App Service」や「Microsoft Bot framework」同様、Microsoft Azure で提供されるデータベースである。保存形式は、データの操作のしやすさを考慮して、JSON 形式とした。内部的には、オブジェクト指向に基づきとあるクラスを JSON 形式に変換した。そのクラスは、WordData というクラスであり、そのメンバ変数は以下の表 3.2 に示した。

表 3.2 WordData クラスのメンバ変数

変数名	データ型	内容
qid	string	設問番号を格納する。「つまずきポイント」を検索する際に用いる。
wid	int	単語ごとに一意に振られる番号である。
name	string	「つまずきポイント」や数学用語そのものを格納する。
page	string	数学用語が掲載されている教科書のページを格納する。
question	string	教科書を用いて数学用語を調べたかどうか調べるための質問を格納する。
backwords	string[]	その数学用語を学ぶにあたって事前に必要となる数学用語を格納しておく。

「つまずきポイント」の格納例

```
{
  "qid" : "1",
  "wid" : 0,
  "name" : "Leibniz の公式って何？",
  "page" : null,
  "question" : null,
  "answer" : null,
  "backWords" : [
    "Leibniz の公式"
  ],
}
```

数学用語の格納例

```
{
  "qid" : "0",
  "wid" : 2,
  "name" : "極限值",
  "page" : "p5",
  "question" : "ある実数  $\alpha$  に  $a_n$  が限りなく近づくとは、数直線上で点  $a_n$  と  $\alpha$  との距離がいくらでも小さくなることを意味している式番号は？",
  "answer" : "1.6",
  "backWords" : [
    "数列の極限"
  ],
}
```

また、WordDate クラスをデータベースに格納するプログラムと読み出すプログラムが必要である。よって、C#にて、それらのプログラムを作成することも実装作業に含まれた。ただし、「Microsoft Bot framework」と「CosmosDB」はいずれも Microsoft 社に提供されており、どちらも C#にて開発を行える。また、「CosmosDB」に接続するための学習リソースは Microsoft の公式サイトに掲載されているため、実装難度は高くないと考えられた。

(※文責: 南部 優太)

3.2.6 会話モデルの設計

前述したように、作成予定の Bot サービスは対話的に学習支援を行う。対話的に支援を行うため、Bot サービスは学習者への質問や、学習者の発言に対して応答を行う。そのため、Bot サービスの会話の流れをあらかじめ作成しておく必要がある。Bot サービスと学習者の間でなされる会話の流れを構造化したものが会話モデルである。例えば、

1. 最初は Bot サービス側から挨拶をする。
2. もし応答があれば、今の気分について質問する。
3. 「気分が良い」という応答だったならば・・・

というような流れが会話モデルの1つの例である。我々は、e-Learning サービスが満たすべき機能を考慮し、会話モデルを作成した。図 3.3 は、会話モデルを状態遷移図として簡易的に表したものである。ここで、会話モデルの各状態について説明する。

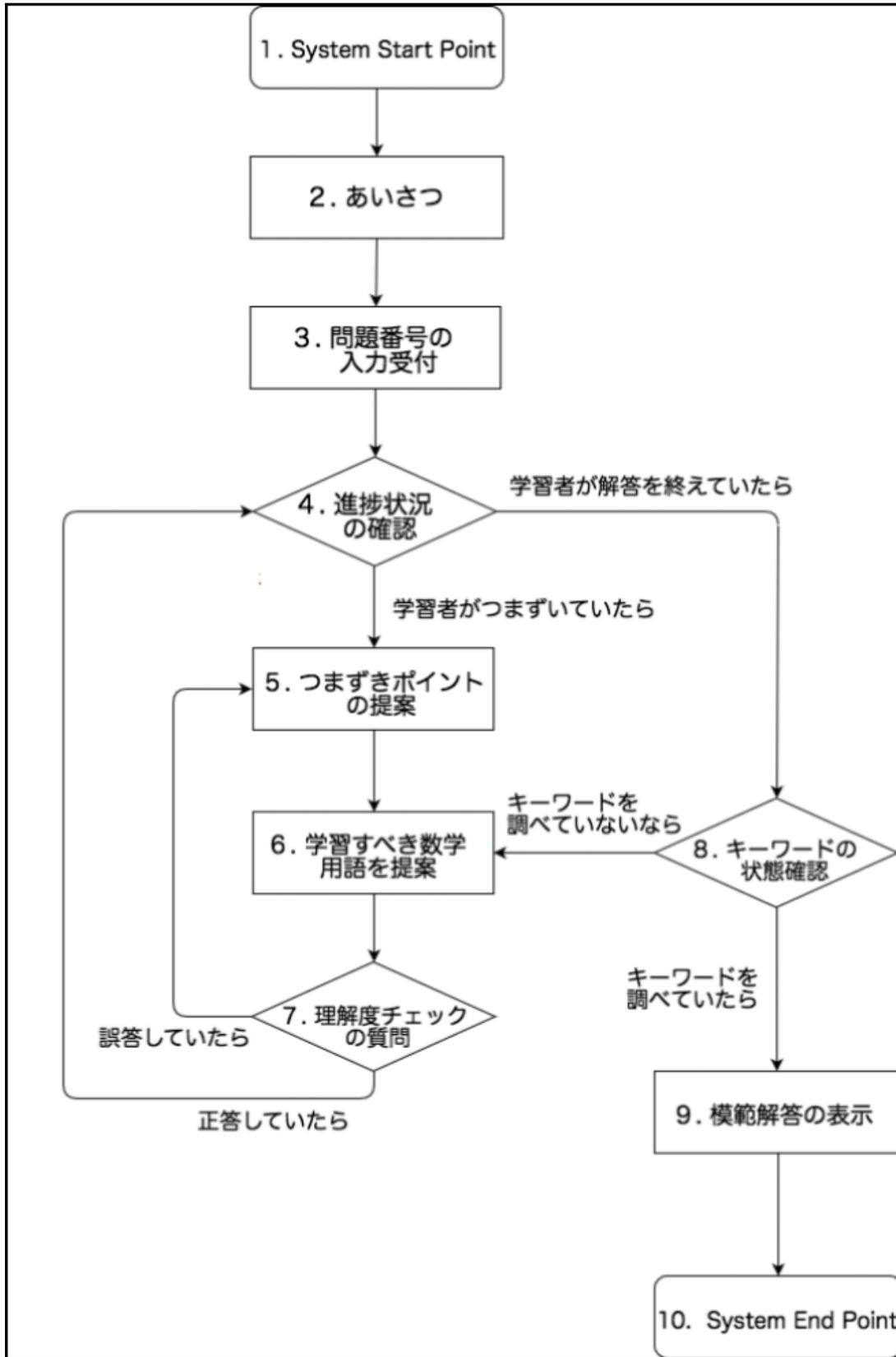


図 3.3 会話モデルの状態遷移図

1. 「System Start Point」について

System Start Point とは文字通り、会話の初期状態のことである。この状態で、学習者から何らかの入力があった場合、次の状態 2 に進む。

2. 「あいさつ」について

この状態では、あいさつをして学習者と最初の対話を行う。次に、複数ある数学の設問のうち、何番の問題を選んだのかを入力するように学習者に促し、次の状態に進む。

3. 「問題番号の入力受付」について

学習者が入力した問題番号を保存する。そして、行き詰まるまで1人で設問を解くように学習者に促し、次の状態に進む。ここで入力された番号によって、状態5で表示される選択肢が変化する。

4. 「進捗状況の確認」について

ここでは、学習者に対して、設問を解き終わったのか、あるいはどこかでつまづいたのかを質問し回答を受け付ける。ここで、解き終わっていた場合には、状態8に進む。どこかでつまづいていた場合には、状態5に進む。

5. 「つまづきポイントの提案」について

ここでは、学習者がつまづきそうなポイントをいくつか選択式で提示する。ここで、提示するつまづきポイントは、問題によって異なるものである。そのため、状態3で保存した問題番号から、学習者が現在どの設問に挑戦しているのかを判断し、その問題に適したつまづきポイントを提示する。そして学習者には、そのつまづきポイントの中から、自分がつまづいている状況に最も近いものを選択してもらう。そして、状態6に進む。

6. 「学習すべき数学用語を提案」

ここでは、状態5で学習者に選んでもらったつまづきポイントに合わせて、適切な数学用語の理解を促す。促す方法とは、ある数学用語が掲載されている教科書上のページ数を教えることである。例えば、極限値を求める問題で、学習者がつまづきポイント「極限値の意味がわからない」を選択した場合には、「極限値」という単語が掲載されている教科書上のページを伝える。そして、次の状態7に進む。

7. 「理解度チェックの質問」について

ここでは、状態6で提示した数学用語を、学習者が教科書を用いて調べたかを確認する簡単な質問をする。例えば、先ほど述べた「極限値」についてであれば、「ある実数 α に a_n が限りなく近づくと、数直線上で点 a_n と α との距離がいくらでも小さくなることを意味している式番号は？」というような質問を行う。この質問に対して、学習者が正答できれば状態4に戻る。学習者が正答できなかった場合は、その数学用語について理解できていないと判断し、その数学用語を理解するために必要な別の数学用語を再度提示する。「極限値」の例でいうと、「数列の極限」という単語を提示して状態5に戻る。

8. 「キーワードの状態確認」について

この状態は、状態4において学習者が設問を解き終わっていた場合に遷移する状態である。ここでは、学習者が本当にこの設問を解き終えることができているのかを調べるためのチェックを行う。具体的には、設問ごとに設定されているキーワードを学習者をデータベースに問い合わせたかチェックする。例えば、先ほどの極限値を求める問題であれば、極限値の定義を知っていることは、問題を解く上で必須である。そのため、キーワードを極限値とする。そして、学習者が「極限値」という言葉をデータベースに問い合わせたかどうか、履歴に残っている情報からその状態を確認することで判別できる。ここで、学習者がキーワードを調べていれば、状態9に進む。学習者が調べていなかった場合には、状態9で正答できなかった場合と同様に「数列の極限」という単語を提示して、状態5に戻る。

9. 「模範解答の表示」について

状態 8 において、学習者が質問に正答することができていた場合、模範解答を見せても良いと判断して模範解答を表示する。そして、状態 10 に進む。

10. 「System End Point」について

この状態は文字通り、会話の終了状態を表す。この状態にたどり着いた学習者には、別の設問を解くことを勧めたり、復習をすることを勧めたりする。

(※文責: 南部 優太)

3.2.7 実装とそのスケジュール

実装は、2017 年 10 月 11 日 (水) から、2017 年 11 月 10 日 (金) までの約 5 週間で行われた。実装の範囲にはすでに述べた、会話モデルの作成や、3.5.1 項から 3.5.3 項の内容を考案することも含んでいる。手順については以下に、1 週ごとにまとめて掲載する。

第 1 週 (10/1~10/17)

- モバイルウェブアプリ
 - ・機能要件を考案
 - ・必要なウェブページの考案
 - ・レイアウトを考案
 - ・JavaScript について学習
- Bot サービス
 - ・サービスの構成要素をグループ内で共有
 - ・開発環境の構築
 - ・会話モデルの作成
 - ・会話モデルに基づいた、C 言語でのアルゴリズムの考案
- データベース
 - ・データベースに保存すべき内容の考案

第 2 週 (10/18~10/24)

- モバイルウェブアプリ
 - ・基本的なモバイルウェブアプリの実装 (ウェブアプリとして動作やページ遷移のみを行うもの、テキスト等は全て仮データの状態)
 - ・ログイン機能の作成開始
 - ・git を導入
 - ・TA による git 勉強会の開催
 - ・「コンセプト」や「使い方」などの、テキスト部分の文章作成
 - ・メモ機能の実装

- Bot サービス
 - ・選択肢を含んだ返答の実装
 - ・会話モデルに基づき、C#にてアルゴリズムを実装
 - ・Bot サービスから Cookie にデータを保存する方法を模索
- データベース
 - ・「CosmosDB」の環境構築
 - ・格納するクラスの作成
 - ・データを格納するプログラムの作成
 - ・テスト用デモデータを格納

第3週 (10/25~10/31)

- モバイルウェブアプリ
 - ・レイアウトの調整
 - ・模範解答表示機能の作成
 - ・コンテンツ班から1問分の実データ受け取り、動作テストを実施
- Bot サービス
 - ・コンテンツ班から1問分の実データ受け取り、動作テストを実施
- データベース
 - ・データ格納プログラムの変更
 - ・コンテンツ班から1問分の実データ受け取り、動作テストを実施

第4週 (11/01~11/07)

- モバイルウェブアプリ
 - ・動作テストを受けての修正
- Bot サービス
 - ・動作テストを受けての修正
- データベース
 - ・動作テストを受けての修正

第5週 (11/08~11/10)

- モバイルウェブアプリ
 - ・動作テストを受けての修正
 - ・コンテンツ班から4問分の実データ受け取り実装

- ・問題数を増やしたためテストを実施
- ・ログイン機能の完成

- Bot サービス
 - ・動作テストを受けての修正
 - ・コンテンツ班から4問分の実データ受け取り実装
 - ・問題数を増やしたためテストを実施

- データベース
 - ・動作テストを受けての修正
 - ・コンテンツ班から4問分の実データ受け取り実装
 - ・問題数を増やしたためテストを実施

(※文責: 南部 優太)

3.3 e-Learning サービス「ModoLuca」

3.2 節で述べた実装過程を経て作成されたものが、e-Learning サービス「ModoLuca」である。ModoLuca の目的は、学習者に教科書を読むことを促し、支援することである。学習者が ModoLuca を利用する環境は、教科書・ノート・ペンという通常の自習環境に、副教材としてスマートフォンから ModoLuca にアクセスしてもらうことを想定している。ModoLuca は、学習者が数学の設問を解く中で、どこかでつまづいてしまい手が止まってしまったときに、教科書の利用を促し教科書を”読む”ことを支援することで、学習者につまづいている状態から抜け出してもらうためのサービスである。ここでは、ModoLuca の使用方法を説明することで、サービスの概要の説明とする。

まず、ModoLuca にインターネットブラウザでアクセスした学習者は、問題一覧画面から解きたい問題を自由を選ぶ。選択を行うと、その問題の詳細が記述されたページに遷移する。図 3.4 が、問題の詳細表示画面である。学習者が問題詳細を読んで、その問題を解くことを決めたならば、「この問題 [1] に挑戦する」ボタンを押す。ボタンのテキストに含まれている、角括弧に囲まれた「1」という番号は問題番号であり、後に利用するため学習者が覚えておく必要がある。

「この問題 [1] に挑戦する」ボタンを押すと、チャット画面に遷移する。チャット画面にて、「こんにちは」という挨拶の文章を学習者が送信すると、図 3.5 のように、応答が返ってくる。ここで、解きたい問題番号を入力するように促されるので、先ほどの問題番号「1」を入力する。すると図 3.6 のように、「行き詰まるまで一人で問題を解いてみてね!」という言葉とともに、学習者に問題を一人で解くことを促す。

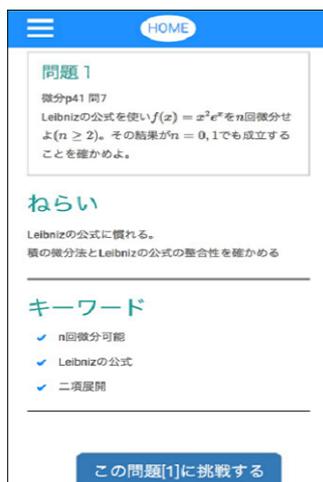


図 3.4 問題詳細画面

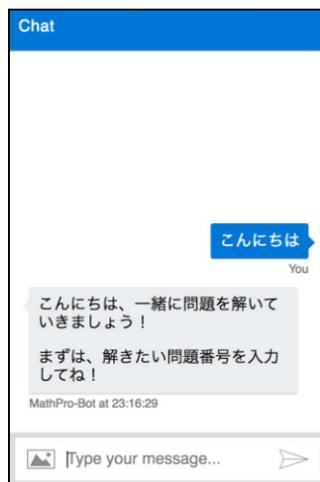


図 3.5 あいさつとその応答

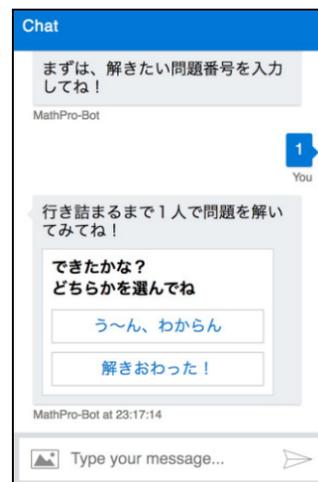


図 3.6 問題番号入力とその応答

学習者が問題を解く過程で行き詰まった場合、「う～ん、わからん」を選択してもらう。そうすると、ModoLuca は教科書を読むことを支援する状態に遷移する。まず、ModoLuca は図 3.7 のように、学習者が陥りがちな状況を指す「つまずきポイント」をいくつか提案し、学習者に選んでもらう。そうすると ModoLuca は、図 3.8 のように、学習者がそのつまずきから抜け出すために学習すべき数学用語を提案し、教科書を用いて調べるように促す。また、学習者が本当に教科書を用いて調べたかどうかを確認するために、理解度チェックの質問を行い、学習者からの回答を受け付けます。



図 3.7 つまずきポイントの提案画面



図 3.8 数学用語の提案と質問

もし学習者がこの質問に正答できた場合、図 3.9 に示したように、学習者一人で問題を解くことを再度促し、1 度支援を終了する。数学の問題を解く上で、学習者がまた別の理由でつまずいた場合には、「う～ん、わからん」をもう一度選択してもらうことで、つまずきポイントを選ぶところから支援を受けることができる。もし学習者が先ほどの理解度チェックの質問に誤答して

しまった場合、図 3.10 に示したように、その単語を理解するために必要な別の数学用語を提案する。学習者はそれらの数学用語の中から、理解が不十分であると思うものを選択する。そうすると、ModoLuca はその数学用語を教科書を用いて調べるように促し、理解度チェックの質問を行う。そして、この質問に学習者が正答できた場合、ModoLuca は支援を一度終了する。この質問にも、誤答してしまった場合、ModoLuca は先程と同様に別の数学用語を提案する。

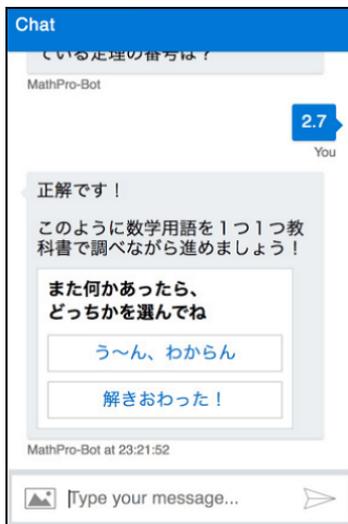


図 3.9 質問に正答した場合の応答

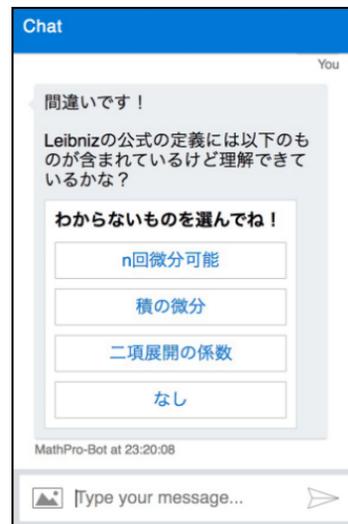


図 3.10 質問に誤答した場合の応答

このように ModoLuca は、学習者がある数学用語を理解できていないようならば、教科書の前のページに戻って、その用語を理解するために必要な数学用語を学ぶように促す。教科書の前のページに戻って学習する力、つまり「もどる力」を身につけて欲しいという思いから、このサービスには「ModoLuca」という名前がつけられている。学習者が問題を解き終わった場合には、図 3.9 に示されている「また何かあったら、どっちかを選んでね」という応答に対して「解き終わった！」ボタンを選択する。そうすると、模範解答を掲載しているページの URL が表示されるため、そのリンクにしたがってページ遷移をすると、模範解答を閲覧することができる。ただし、学習者が模範解答の全体を閲覧できるとは限らない。各設問には複数のキーワードが設定されている。キーワードとは、ある設問の模範解答を学習者に見せるにあたって、少なくとも理解しておいて欲しい数学用語群のことである。キーワードは、本プロジェクトのコンテンツ班が定めたものであり、設問ごとに数学用語とその個数が異なっている。学習者が ModoLuca を用いてキーワードを調べた場合、そのキーワードに対応する模範解答の一部を、学習者は閲覧することができる。つまり、模範解答は複数個に分割されており、我々が定めるキーワード全てを ModoLuca によって調べない限り、学習者は模範解答の全体を閲覧することができない。この仕組みによって、ModoLuca の支援を受けずに模範解答のみを閲覧するという行為を防いでいる。

(※文責: 南部 優太)

3.4 解析学Ⅱ勉強会

ModoLuca が本当に教科書を「読む」ことを促し、支援できているのかを調べるために、評価実験として解析学Ⅱ勉強会を実施した。この活動の概要については、3.4.1 項、3.4.2 項で述べるが、詳細については、検証班のグループ報告書を参照いただくこととする。

(※文責: 宍子澤 考輔)

3.4.1 解析学Ⅱ勉強会の概要

解析学Ⅱ勉強会は、ModoLuca が本学の 1 年生に対して教科書の利用を促すことができるのかを調査するために行った。解析学Ⅱ勉強会は、2017 年 11 月 10 日 18 時 10 分より 494 及び 495 教室にて行った。当日は、解析学Ⅱ履修者 32 名が参加した。評価実験のスケジュールを表 4.1 で示す。なお参加者が教科書や授業時にとっている各自のノートおよびインターネットでの検索、友人への相談することについては制限しなかった。

表 3.3 解析学Ⅱ勉強会のスケジュール

内容	時刻表	所要時間 (分)	詳細
受付	17:45 ~		入り口付近にて受付
挨拶	18:15 ~ 18:20	5	自己紹介及び企画の趣旨説明
問題解答 1	18:20 ~ 18:30	10	ModoLuca を使用せず問題を解く
ModoLuca の説明	18:30 ~ 18:35	5	ModoLuca の使い方を説明する
問題解答 2	18:35 ~ 19:35	60	ModoLuca を使用して問題を解く
挨拶	19:35 ~ 19:40	5	終了の挨拶
解答スキャン、アンケート	19:40 ~		終わり次第解散。解答のスキャンを行う

我々プロジェクトメンバーは、この解析学Ⅱ勉強会にチューターとして参加した。チューターは参加者に ModoLuca の利用方法を教える。もし、ModoLuca を正しく使えていない、あるいは使っていない参加者がいた場合には、ModoLuca の利用方法を再度教えて、利用を促した。ModoLuca を利用しても、問題を解き進めることができない参加者に対しては、チューターがヒント教えることとした。

解析学Ⅱ勉強会の参加者には勉強会の前後アンケートと、解析学Ⅱ勉強会から 2 週間後にフォローアップアンケートに回答していただいた。アンケートの詳細内容は検証班の報告書を参照いただくこととする。

(※文責: 宍子澤 考輔)

3.4.2 解析学 II 勉強会の結果と考察

解析学 II 勉強会当日は、ModoLuca のバグやエラーによって進行不可能になることはなかった。解析学 II 勉強会前後のアンケートやチューターの観察をまとめた結果より、ModoLuca は参加者に教科書を「読む」ことを促し、支援することが出来た。解析学 II 勉強会の結果、このシステムには 4 つの問題があることが発見した。

1 つ目は、ModoLuca が参加者の意図していた挙動をすることができない状況が起きたということである。この問題の発見は、解析学 II 勉強会后アンケートの自由記述の回答に書かれていたからである。実際のアンケートの回答は次のようであった。

- 質問しようを使ったあとだと上半分の画面を使わないとスクロールがすぐにできなくメモしようにいけない
- 全角数字に対応してくれない時がある
- 細かな疑問に適切な答えが返ってこない点
- 計算ミスで行き詰まった時の解決策があったら、嬉しいです。
- たまに返してくれない
- 選択枝外の対応語句を増やして欲しい

このことから、半角、全角が区別されるような入力の柔軟性が低いシステムを実装したことが、原因だと考えられる。また、データベースに数学用語を問い合わせる際、名前が完全に一致したものを取り出す仕様であったために、学習者の入力に 1 文字でも誤字があると、Bot が応答しなかった。

2 つ目は、数学用語を質問する機能があまり使われていなかったことである。これは、この機能が参加者にあまり周知されていなかったことが原因だと考える。さらに、上述したように 1 文字でも誤字なく質問しなければいけないため、参加者がタイプミスをしてしまうと応答が正しくできない。実際に起きたタイプミスの中には、「Liebniz の公式」のようなタイプミスがあった。正しくは、「Leibniz の公式」である。さらに、データベースに格納されている数学用語が少なかったため、我々が想定していない質問には対応できなかった。ModoLuca が計算方法を教えることを想定していなかったが、「1C0」といった式計算について Modoluca に質問した参加者がいた。

3 つ目は、ModoLuca の UI が参加者に対して親切でなかった。ModoLuca は説明書や、メンバーからの説明がないと操作ができないサービスとなってしまう。さらに、端末によっては画面の表示が崩れてしまい、Bot の応答が読みにくいという声が参加者から上がった。

4 つ目は、理解度チェックのために行う質問は、簡単すぎてあまり理解度チェックの役目を果たせていなかったということである。これは我々が学習者を観察してわかったことである。質問が簡単になった原因は、Bot 応答には、MathJax の数式を表示できなかったため、数学に関する込み入った質問ができなかったからである。また、質問に対する正答が「1」や「2.1」のようなシンプルなものにしないでいけなかった。なぜなら、複雑な正答を許してしまうと、Bot が正誤判定できなくなってしまうおそれがあったからである。

(※文責: 卯子澤 考輔)

3.4.3 解析学Ⅱ勉強会後のアップデート

3.4.2 項の 4 つの問題点を踏まえ解析学Ⅱ勉強会后、ModoLuca のアップデートを行った。アップデートでは問題 2 問の追加と、2 点のシステムの改善を行った。システムの改善については、解析学Ⅱ勉強会后アンケートで、ModoLuca の問題点・改善点を勉強会の参加者に自由記述で聞いていた。この自由記述の中から、改善するものを実装班で吟味した結果、以下の 2 点をアップデートした。

- 全角・半角入力を区別することの改善。

ModoLuca では、定理番号の入力などで、英数字を入力する機会が多数ある。その入力の際に、全角と半角を区別する仕様になっていた。つまり、ある質問に対する正答が半角の「1」である場合に、全角の「1」を使って回答してしまうと間違いであると判定してしまっていた。よって、この問題をに関しては、入力された文字列のうち英数字のみを一度半角に変換する処理を加えることで対応した。

- Modoluca の質問に間違えてしまったあと、戻ることができないことの改善。

ModoLuca では、理解度チェックのために学習者に対して質問を行う。もし、その質問に間違えてしまうと、次の状態に遷移してしまい、もう一度質問に挑戦することができなかった。これによって、先ほど述べた全角・半角問題などによって、ModoLuca から「間違い」という判定を受けた場合、本当は理解しているにも関わらず「間違い」ルートに遷移せざるを得ない状況となる。だが実際には、もう一度質問に答え直すという仕様が存在している。この仕様が十分に学習者に伝わらなかったためこの問題が発生した。したがって、ModoLuca の使い方の説明ページに追記をして対応した。

(※文責: 卯子澤 考輔)

3.5 成果発表会

本章では、プロジェクト学習において最終的な成果を発表する場である、成果発表会について報告する。まず、成果発表会のために行った準備や当日のことを成果発表会の概要として記載する。次に、成果発表会の結果と考察について述べる。

(※文責: 後藤 春奈)

3.5.1 成果発表会の概要

成果発表会への準備をするにあたって、前期の中間発表会と同様にスライド班とポスター班に分かれて活動を行った。

スライド班は、成果発表会当日に発表者が使用するスライドの作成を行った。スライドの内容は、今年度の活動をまとめたものであり、ModoLuca のデモ発表なども用意した。中間発表同様、プロジェクトメンバーや担当教員からレビューをいただいて、たびたび修正作業を行った。また、発表者は発表練習を行い、その結果も踏まえたスライドの修正が行われた。

ポスター班は、発表で使用されるポスターの作成を行った。作られたポスターは、メインポスター 1 枚、サブポスター 1 枚の計 2 枚であった。メインポスターはこのプロジェクトの 1 年間の活動に

ついて詳細に説明するものであった。サブポスターは作成した ModoLuca のシステムのメリットと、ModoLuca の会話モデルを説明したポスターであった。

スライドとポスターの作成は、当初想定していたスケジュールから大幅に遅れた。そのため、人員を再配置し、リスケジュールを行った。その結果、ポスターとスライドは成果発表会当日直前に完成した。

成果発表会では、メンバーを前半と後半に分けた。前半と後半それぞれ 3 回ずつの、計 6 回の発表を行った。その発表の際、聴講者に対して発表評価シートを配布し、記入してもらった。発表評価シートは中間発表会同様に、発表技術と発表内容に関する項目のほかに、「ModoLuca を使いたいと思ったか」「ModoLuca に追加してほしい機能・改善してほしい機能はあるか」「数学を学習するときどのようなサポートを受けたいか」という項目を別途に追加した。

(※文責: 後藤 春奈)

3.5.2 成果発表会の結果と考察

成果発表会当日は、学生や教員のほか、外部の人を含む 96 名の評価をいただいた。項目ごとに記入がなされているものを有効票、記入がなされていないものを無効票とした。発表技術についての評価の平均は 8.59(有効票 $n = 93$ 、標準偏差 $SD = 1.10$) であった。発表内容についての評価の平均は 8.50(有効票 $n = 50$ 、標準偏差 $SD = 1.15$) であった。「ModoLuca を使用したいと思うのか」の平均は 4.26(有効票 $n = 50$ 、標準偏差 $SD = 1.02$) であった。これら 3 つの度数分布を、以下の図 3.11、図 3.12、図 3.13 にまとめた。

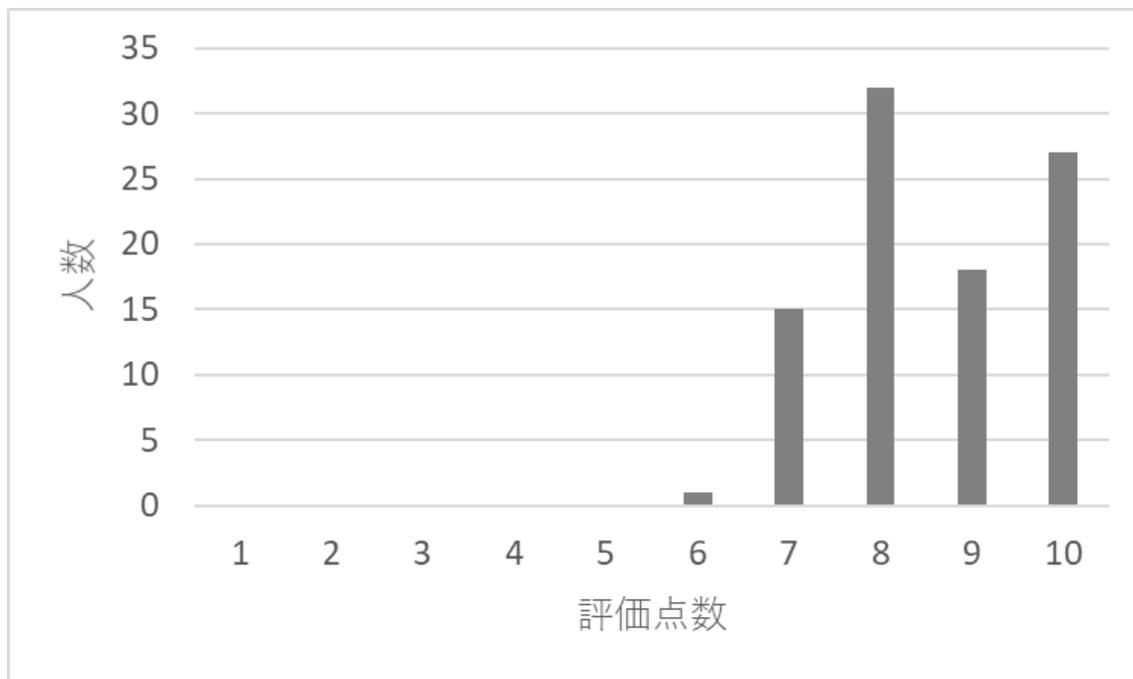


図 3.11 発表技術についての度数分布

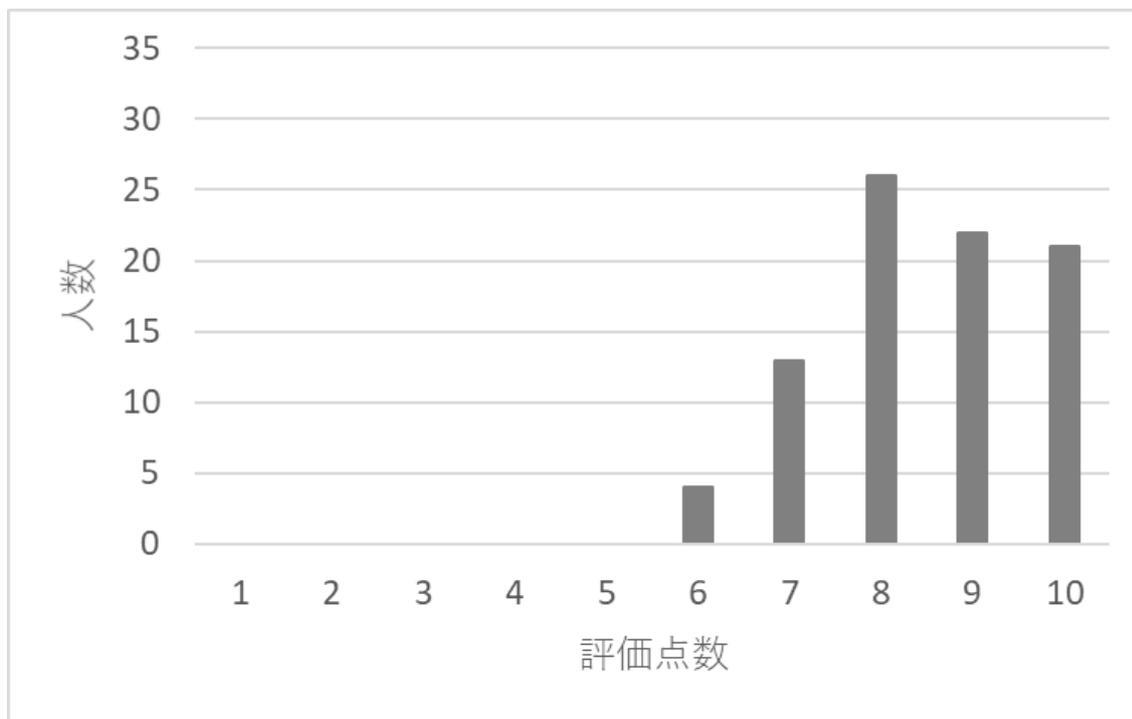


図 3.12 発表内容についての度数分布

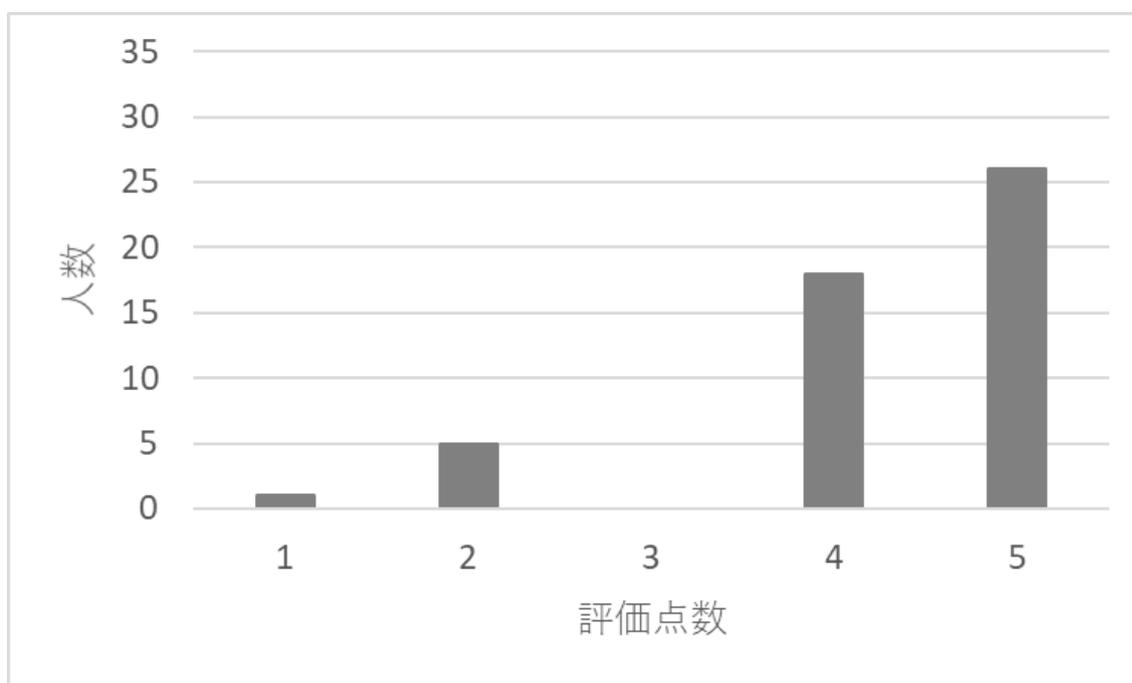


図 3.13 ModoLuca を使用したいと思うかについての度数分布

また自由記述で得られたコメントは以下の表 3.4 にまとめた。なお、このコメントは、評価シートから引用したものである。

表 3.4 評価シートに記入された自由記述

	デモがありわかりやすかった
	デモなどがあるのは分かりやすいと思う。アンケート結果、図の使い方が分かりやすく効果的だったと思う。
発表技術	アプリの良さがはっきり分かる発表でした。
	発話が良いですね。資料がテキスト中心でした。視覚情報を有効に使うとさらに良くなるのでは？
	アプリのデモが見辛かった。ディスプレイが小さい。実機を使い、自分で操作して見たい。
	アイデアがすごいと思った。高校にもほしいです。
	教科書を「読む」を促せるのは素晴らしいと思いました。
発表内容	MoDoLuCa が必要なくなる過程、その後へのビジョンがほしい
	支援方法や内容はとても良いけど、人力で解説を作るので、増やすのが大変そう。
	アンケートの母数が少ないように感じた。

中間発表と比較すると、発表技術、発表内容ともに平均点が上昇している。発表技術の平均点は 8.21 から 8.59 に上昇した。また、発表内容の平均点は 7.73 から 8.50 へと大きく上昇した。中間発表の時点では、e-Learning サービスの内容を具体的に決めることができていなかった。そのため「どのようなサポートをするのかが分からなかった」というコメントがあった。それに対して成果発表では、e-Learning サービス ModoLuca の内容について発表したことにより点数が上昇したと考えられる。

自由記述において「デモが見づらかった」というコメントが多くあった。ModoLuca は、スマートフォンに縦画面で表示されることを想定し、画面が設計されている。しかし、デモを行ったときのモニターは横向きに設置されていた。そのため、ModoLuca がモニターに十分に表示されていなかったことに原因があると考えられる。モニターへの表示を十分にできなかった原因は、発表会の前日に使用する機材に接続してチェックすることができなかったからであった。

次に、我々が独自に追加した質問項目の考察について述べる。まずこの項目は、回答数があまり多くなかった。これは発表評価シートの裏面に質問項目を印刷したため、気づかれにくかったことに原因があると考えられる。これより「裏面に続きます」といった文言を強調し、目立つように書く必要があると考えられる。

「ModoLuca を使用したいと思いますか」という質問については 5 段階評価のうち、平均点 4.26 と高い評価であった。このことから、ModoLuca は利用者に使いたいと思われるものであるといえる。

「ModoLuca に追加してほしい機能はありますか」という質問では「他の科目にも対応してほしい」といった意見があった。そのほかに「類似問題を増やしてほしい」という意見もあった。ModoLuca のコンセプトは「読む」を促し、「読む」を支援することである。そのため、最終的にはユーザーが ModoLuca を使用しなくても教科書を「読む」ことができるのが理想である。利用者が正しい学習方法を身につけることができれば、類似問題の実装や他の科目は不要であると考えられる。

「数学を学習する際にどのようなサポートをうけたいですか」という質問では、「わからないところを教えてもらいたい」や「つまづいた場所の解説が欲しい」などの意見があった。これは、

ModoLuca がサポートすることができるものなので、ニーズを満たしていると考え。一方で「類似問題が欲しい」「おすすめの参考書を教えてほしい」という意見もあった。しかし、この意見は ModoLuca のニーズを満たしていない。ModoLuca によって「読む」という学習方法を身につけることができれば、類似問題や参考書は不要であると考え。

(※文責: 後藤 春奈)

第 4 章 活動のまとめと今後の展望

ここでは、プロジェクト学習における活動のまとめと展望について述べる。4.1 節ではこのプロジェクト活動で行ってきたことを前期と後期に分けて述べる。4.2 節では今後の展望について述べる。

(※文責: 知子澤 考輔)

4.1 活動のまとめ

前期は、プロジェクト内学習会で「プロジェクトメンバーが教科書を『読め』ていないということは、本学の 1 年生も同様に教科書を『読め』ていない可能性がある」という仮説を立てた。この仮説が成り立つかを検証するために解析学 I 勉強会を行った。解析学 I 勉強会によって次の 2 つがわかった。1 つ目は、1 年生は「問題を解く過程で困った場合、教科書を利用するのではなく、インターネット検索を行なっている」ことである。2 つ目は、1 年生はインターネット検索する際にスマートフォンを利用していることである。この 2 つの発見をうけて、我々は教科書を「読む」ことを促し、「読む」ことを支援するスマートフォン向け e-Learning サービスを作成することを決定した。中間発表ではここまでの内容を発表し評価を受けた。

後期は、e-Learning サービス「ModoLuca」を提案し、実際に作成した。この ModoLuca が利用者に対して「読む」ことを促し、「読む」ことを支援することが出来るかどうかを検証するために解析学 II 勉強会を行った。その結果、勉強会の参加者に「読む」ことを促すことが出来た。しかし、「読む」ことを促すだけでは学習者は問題を解けるようにはならなかった。

(※文責: 知子澤 考輔)

4.2 今後の展望

解析学 II 勉強会の結果と考察から、次の 3 つのシステムの改善を検討している。

- ModoLuca の質問機能の強化
- ModoLuca の UI の改善
- ModoLuca への入力 of 柔軟性向上

ModoLuca には学習者が教科書を読んだかどうか調べるための質問がある。しかし、ModoLuca が行う質問が効果的であったとは考えにくい。その質問の正答は、定理番号などの教科書に書かれてある内容であった。そのため解答が容易に発見できてしまい、学習者が教科書を読んだかどうかを客観的に評価することができない。また、質問が容易すぎるために学習者にとっては、ただのわずらわしい機能となっている可能性がコンテンツ班から指摘された。よって、もっと効果的な質問を考案することが 1 つの展望としてあげられる。

次に、Bot Frame の UI の改善が必要である。ModoLuca の対話を行う UI 部分は、開発を容易にするために Microsoft 社が提供している iframe タグを埋め込んで実装している。よって、表示されるフレームは Microsoft 社が管理するサーバー上のウェブページが表示される。そのため、我々が UI 変更のためにプログラムを改変することはできない。ただし、そのウェブページのソースコードは公開されているため、自身でサーバーを用意して開発をすることで UI の改善を行うことができる。改善を行うためには、Node.js というサーバーサイドプログラムの知識が必須である。

最後に、ModoLuca の入力処理をさらに柔軟にする必要がある。「ModoLuca」は学習者に対してテキスト入力を求めるが、その入力に対する応答の柔軟性が低い。例えば、「 n 次導関数」という用語について質問をしたときに、誤って「 N 次導関数」と入力するだけでも現行のシステムでは、ねらった動作をしない。柔軟性を上げる方法の 1 つとして、自然言語処理を導入することが挙げられる。Microsoft 社では、自然言語処理を容易に導入することのできる「LUIS」というサービスがある。これによって、「微分がわからない」と一字一句全てが一致するように入力しなくとも「微分って何？」や「微分とはなんですか？」のような入力にも対応できるようになる。

アンケートによる学習者の主観的評価により、ModoLuca によって学習者に「読む」ことを促すことが出来たと考えられる。なので、さらに次の支援が必要であると考ええる。その次の支援として考えられるのは、具体的な計算方法が理解できていない場合に支援をすることである。解析学Ⅱ勉強会においてはチューターは学習者に対して具体的な計算方法を教えるという支援を行っていた。この支援をチューターの代わりに ModoLuca が行うということである。この支援を行うためには ModoLuca だけでなく、2016 年度の「ますますたでい 2016」や、メタ学習ラボと連携することが出来ないかどうかを検討する必要がある。

(※文責: 卯子澤 考輔)

第 5 章 プロジェクト内のインターワーキング

本章ではプロジェクトメンバーが各自で内省したものを記述する。プロジェクトメンバーが割り当てられた課題を解決する際にどのように解決したのか、メンバーが連携して行った作業の内容について述べる。

(※文責: 知子澤 考輔)

5.1 石村秀翔

私は、プロジェクトを通して 3 つのことを学んだ。

1 つ目は、新たな環境を使う際の学習方法である。後期では実装班に所属し、ModoLuca の制作、その中でもログイン機能の制作に携わった。機能を実現するために Firebase を組み込んだり、システムを共有するために Sourcetree を利用したりと新しい環境ばかりだった。その際に、TA やメンバーに聞いたりチュートリアルサイトを見たりして運用することができた。ただ、その際も周りに頼ってばかりだったので、自分で専門書やサイトを調べて使えるようにならなければいけないと思った。

2 つ目は、成果発表の重要さである。発表担当となったが私用に練習会に参加することができず、結果として発表者内では一番最悪な発表をしてしまったと反省している。ある程度、議事録や今までの成果物を見て振り返ったものの、まだまだ勉強不足のため質疑応答ではたじろいでしまった。今後同じような事があったら、しっかり今までやってきたことを見返してどんなことでも答えられるような万全な体制にすることが重要だと思った。

3 つ目は、グループ活動のルールである。これは、前期の活動でもメンバーから言及されるほど連絡を怠ったり業務を遂行できなかった。そのため、後期ではなるべくないように努めたが自分の意志が弱いがために幾度も同じことを繰り返した。やり方云々よりも自分の意志が弱いと認識できたので、今後は意識の改善に努めていこうと思う。

(※文責: 石村 秀翔)

5.2 後藤春奈

私は、実装班の 1 人として活動を行った。しかし、実装班として満足のいく活動をすることはできなかった。

私は当初、ModoLuca の自然言語処理の実装に携わる予定であったが、技術の習得に時間がかかり、予定通りの進捗を生み出せなかった。そのため、リーダーが 1 人で作業を行った方が早いという結論になり、リーダー 1 人に作業を任せてしまうことになった。これは、私の実力不足も原因の 1 つだと思うが、与えられた仕事に対する責任感が弱かったのが大きな原因だと考えている。何があっても与えられた仕事をこなし、成果を出す。そういった意志が強ければリーダーから割り振られた仕事をこなすことができたのではないかと思う。

実装班としては十分に活動することはできなかったが、成果発表会では上記の反省を生かし、責

任感を持って活動できたと思う。

私は成果発表会で発表者を担当することになった。しかし、プレゼンが苦手で、気心のしれたプロジェクトメンバーの前ですら、緊張してしまい、早口になったり、聴講者を意識して発表をすることができなかった。もし本番もこのような発表をしてしまったら、プロジェクト全体の成果に悪影響を与えてしまう。そうならないように、自分のプレゼンに対するレビューをいただき、それをもとに必死に練習を重ねた。その結果、発表当日は自信をもって発表することができた。発表に関する評価でも、良い評価をもらうことができた。発表者として、責任感のある活動ができたのではないかと思っている。

(※文責: 後藤 春奈)

5.3 南部優太

私は実装班のリーダーとして活動を行なった。主な業務として、他班（コンテンツ班, 検証班）との情報共有や、担当指導教員への報告、そして実装班の作業円滑化が求められていた。他班との情報共有の観点では、よく連絡が取れていたように思う。コンテンツ班のリーダーとは、何度も打ち合わせを行い、互いに認識に齟齬がないかよく確認しあった。担当指導教員への報告についても、自主的にプロジェクトリーダーと指導教員の打ち合わせに参加し、進捗の報告ができていたため、大きな問題はなかったように思う。

自身の活動を振り返って、反省する点は2つある。1つ目は、コンテンツ班の作業遅延への対応である。e-Learning サービスを作成する際に、実装班とコンテンツ班で分業して活動していた。実装班がほぼスケジュール通りにタスクを終えていたが、コンテンツ班は活動に2週間ほどの遅延が起きてしまい活動が思うように進んでいなかった。コンテンツ班から実装班へと必要なデータが予定通りに渡されなかったために、実装班の活動までもが停滞してしまった。このとき私は、コンテンツ班に対して毅然とした態度で接することができず、コンテンツ班の遅延をある程度、許容してしまった。その結果、コンテンツ班の遅延によって、実装班の活動にも遅延が生まれてしまい、システムの完成が完成予定日から1週ほど遅れてしまった。

2つ目の反省点は、他のメンバーが実装作業を行うための環境を用意し、作業の円滑化を行うのがリーダーの業務であったのに、リーダーが自らがほとんどの作業を行ってしまったことである。システムを実装するにあたって、私はWBSの作成やスケジュールの調整を行っていた。そして、他のメンバーには、それぞれタスクを割り振っていた。しかし、私が他のメンバーの進捗状況と活動の成果を芳しくないと判断したために、ほとんどの作業を私がやってしまうという結果になった。リーダーの業務は、他のメンバーの進捗が順調でないときにアシストし、そのメンバーが課題を達成できるようサポートすることである。そのような場面で、リーダーが自らそのタスクを行ってしまうことは、リーダーとしてはふさわしい行動でなかったと言える。

(※文責: 南部 優太)

5.4 卯子澤考輔

私自身の活動を振り返って反省する点は、実装班の作業のほとんどをリーダーである南部君に負担させてしまったことである。リーダーはスケジュールの調整を行い、作業を割り振ってくれた。

ここで私は ModoLuca の開発環境を整備し、TopPage や問題選択のページを実装するという作業が割り振られていた。この作業には HTML、CSS、JavaScript の知識が必要である。これらの知識が乏しかった私はこれらの知識の習得から開発作業が始まった。HTML は一年次に情報機器概論で触れたのと卒業研究配属決定時の HTML でしか作成をしたことがなかったので div タグなど詳しいことがわからずその勉強から始めた。CSS は今回の作成で始めて触れた内容でどのように処理をされていくのかがわからなかったため、初歩から勉強を始めた。JavaScript も CSS と同じく、初歩から勉強を始めた。その結果進捗は当初のスケジュールから大幅に遅延し、リスケジュールをする必要が出てきた。その結果開発の作業のほとんどを南部君に負担させてしまった。

リーダーに作業が集中して負担してしまうことを回避するためには夏休み中に必要な知識の学習を進めておく必要があったと考えている。なぜなら前期が終わるときに後期から e-Learning サービスを作成することが決定していたので必要な知識は何かを事前に調べ、学ぶことができたはずだからである。このことを反省し、先を見通した計画を自分で立て、目標達成に必要な知識や技術を洗い出し、学習することが今後の課題であると考えた。

リスケジュールして私が行ったことは、HTML の文章の作成と CSS の修正であった。HTML の作業は文章を書き込んでいくだけの作業なのである程度こなすことが出来た。

CSS は TA の方々に試験運用してもらった際に色覚的に違和感があった部分を主に修正した。修正する CSS には既成の物を流用して作成している部分もあった。流用していることが原因で色覚的に違和感があったものを修正するのに非常に時間がかかった。CSS を修正するために Google で HTML を開き、検証の画面でソースコードを調べ原因になる部分を探した。探した後、検証画面でソースコードを実際書き換えて修正できるかを調べた。この検証の部分でソースコードを書き換えすることができることを知るまでは既存の CSS ソースコードを一から見てどこに原因があるのかを探そうとしていたので非常に時間がかかった。検証で調べることが出来る事を知ったことにより修正時間を大幅に短縮することができた上、CSS の処理がどうなっているのかを大まかであるが知ることができた。

(※文責: 知子澤 考輔)

付録 A 課題解決のための技術（新規習得）

TeX

: 各種の報告書の作成に用いた。

HTML

: モバイルウェブアプリの作成に用いた。

CSS

: モバイルウェブアプリの装飾に用いた。

Java Script

: モバイルウェブアプリの動的な挙動の作成に用いた。

C#

: Bot サービスの構築とデータベースの操作に用いた。

(※文責: 知子澤 考輔)

付録 B 課題解決のための技術（講義）

解析学 I・II

：プロジェクトメンバーの学び、そして「ModoLuca」のコンテンツ部分の作成に活用した。

科学技術リテラシ

：報告書等の文章作成の際に活用した。

プロジェクトマネジメント

：タイムマネジメントやスコープマネジメントなどを行う際に活用した。

ヒューマンインタフェース

：モバイルウェブアプリの作成の際に活用した。

企業実習

：「Microsoft Azure」や.NET 環境での開発を行う際に活用した。

システム管理方法論

：サーバーにおいてコマンド操作を行う際に活用した。

Communication1～4

：発表会のときに使用したスライドの作成やデータのまとめる際に活用した。

（※文責: 後藤 春奈）

付録 C 相互評価

C.1 石村による相互評価

後藤

: Modolucano データベーススキーマ作成を頑張っていた。時には私が悩んでいた時も嫌な顔一つせず相談に乗ってくれた。また、グループの中ではムードメーカー的存在でありながらも客観的に全体を見渡し、的確に足りない箇所を突いていたようにも思える。最終発表も、私とは比べ物にならないくらい練習を重ねて臨んだ印象を受けた。

南部

: 実装班のリーダーとして、業務管理やスケジュール管理を徹底し無理なく仕事ができるよう一番尽力していた。また、ModoLuca 作成時でも、連結テストや製品テストの不具合対処や勉強会当日の不具合対応、サービス提供時の管理など率先して様々な役割を担っていたため一番尽力していた人だと思う。

卯子澤

: 仕事に対して早く的確に対処していた印象があった。私が悩んでいるとき一番力になってくれた人でもあった。また、数学に対してもメンバー内で抜き出ている印象も受けた。最終発表も私と一緒に担当したが、ふがいなかった部分をカバーしてくれた。グループ内では一番助けられた存在であると思う。

C.2 後藤による相互評価

石村

: プロジェクト時間内で、積極的に発言をしていた。質問などもちゃんとしていて、積極的にプロジェクト活動に参加しようとしているイメージだった。

前期に比べて報連相を徹底していた。しかし、締め切りが守れないことが何度かあったので、そこを改善してもらいたいと思った。

南部

: グループリーダーとして、責任感を持って班員を引っ張っていた。進捗管理などがしっかりしていて、班員が活動しやすいよう土台を築いてくれた。常に状況を冷静に判断し、班員が無理なく動けるようなスケジュールを常に考えていた。議事録などもしっかり記録し、いつでも実装班の活動を振り返れる状態を作ってくれていた。

ModoLuca の作成の際にも、多くの仕事を担当していた。ModoLuca の作成の際に、最も尽力した人だと思う。

卯子澤

: 仕事が早く、常に締め切りよりも早く仕事を終わらせていた。特に、ModoLuca のアルゴリズムの作成の際は、かなり短い時間で完成させていた。また、自分のタスクにどれくらいの時間がかかるのかを把握していたため、スケジュール管理が上手だったように感じた。

周りが見えなくなり、一人で突っ走ってしまうことがあったので、そこを改善してもらい

たいと思った。

C.3 南部による相互評価

石村

: 前期に比べると後期は、仕事を全うしようとする意思が見られ、仕事が間に合わなさそうな時は事前に連絡するなど、ホウレンソウと呼ばれる行為がよくなるようになったと感じました。ただ、依然として締め切りに間に合わないことが多く、「次の?時までには間に合わせます」と自分で宣言するも、その時刻にも間に合わせることができない状態にはリーダーとしては困るばかりでした。そればかりか、自分の仕事を投げ出して音信不通になることもありました。やはり、仕事に対する責任感に問題があると思います。

後藤

: このグループの中では、タスクと雰囲気という両方の面でバランスを取る役割を担って来ていました。現状を見つめ、課題達成のためにこのグループに足りないものに気づくことのできるメンバーだと思います。ただ、仕事のクオリティや態度に自信のなさや甘えが目立ちます。まず、自分に何ができて、何ができないのかを自覚することが必要だと思います。その上で、私はこれでは負けないというものを作るのが良いと感じました。

卯子澤

: 積極的かつ、早急に仕事に取り掛かってくれるメンバーでした。遅い時間や、締め切りギリギリになっても嫌な顔をせず取り組んでくれました。また、数学に関しては他のメンバーと比べて、抜きでた力を持っていた印象でした。与えられた仕事を全うすることに対しては十分な力を発揮しますが、自分で仕事を創出することは苦手であると感じました。

C.4 卯子澤による相互評価

石村

: ページのログイン機能の作成に尽力してくれた。連絡がたびたびつかなくなることや、タスクが遅れてしまうことが正直目立ちました。前期のときに Slack の通知が来ないといわれていたが、通知が来ないから見れないのではなく、通知が来なくても見に行けばいいのではないかと思います。

後藤

: ページの内容修正や、ModoLuca のデータベースへの入力をやってくれた。使い方ページなど私が作成した文章の校閲に協力いただいた。初見で読んだときにどう思うのかという視点から意見をくれたことが修正するに当たって非常に大きな影響があったと思う。成果発表の練習では早口であることを気にしていたが、当日の発表では言葉と言葉の間に二拍呼吸を入れることを意識していたように見えました。早いという評価がありましたが、練習のときと比べると格段にゆっくり話すことができていたと思います。

南部

: ModoLuca の実装作業が無事に終わることができたのはあなたが尽力してくれたからだと思います。実装においてメンバー内のタスクを柔軟に変化させ、メンバーの采配をしてくれました。Wiki に実装班のページを作り、進捗満足度や、次回のタスクを作成し次にメン

Design and Implementation of a Learning Environment for Mathematics

バーが何をやればいいのかを明確にしてくれました。進捗満足度が100%にならなかったとき、どうすればより進捗満足度が上がるのか書いてくれると1メンバーとしてはより士気が向上したと思います。

参考文献

- [1] 林雄介 (2012) 「実用的 e-Learning 環境の構築と運用」特集号の発刊にあたって. 教育システム情報学会誌, 29, pp.5-6.
- [2] 松田典親 (2006) e-Learning の教材作成 -入学前の数学自習教育-. 奈良文化女子短期大学紀要, 37, pp.27-34.
- [3] 文部科学省 (2012) 大学設置基準. http://www.mext.go.jp/b_menu/shingi/chousa/koutou/053/gijiroku/_icsFiles/afieldfile/2012/10/30/1325943_02_3_1.pdf (2018.01.18 アクセス).
- [4] 齋藤順一, 山方竜二 (2007) 講義の補助を目的とした e-Learning システムの開発と活用方法. 高等教育ジャーナル-高等教育と生涯学習-, 15, pp.61-66.