Webプログラミングを用いたアプリケーション開発

1. はじめに

近年、スマートフォンやタブレット端末をはじめとする携帯可能な小型端末が広く普及している。これらは無線技術を使い簡単にインターネットにアクセスすることができる。これらの端末のアプリケーションを開発するにはOSごとに異なる言語を使用する必要がある。例えば、Android OSならJava言語を、IOSならSwift言語を用いて開発する。つまり、対象ユーザがAndroid及びIOSを使用している場合、異なる言語で同じ機能のアプリケーションを開発する必要がある。しかし、本研究では時間の制約上2つの言語を習得することは難しい。そこで、webプログラミングを用いてアプリケーションを開発することにした。Webアプリケーションとはブラウザ上で動作するプログラムのことである。このアプリケーションは異なるOS上でもブラウザがあれば正常に動作する。つまり、webプログラミングを用いて作られたアプリケーションは様々なOS上で動作可能という特徴を持つ。

2.　リサーチクエスチョン

本研究の目的はwebプログラミングを用いて鹿児島工業高等専門学校の授業評価アンケートを作成することである。鹿児島工業高等専門学校はマークシート方式で授業評価アンケートを実施していた。この方式は紙資源を多く使用する。しかし、web上で授業評価アンケートを実施することで紙資源を使わずにアンケートを実施することができる。

3.先行研究

web上でアンケートを実施するフレームワークとしては，Google者の「Google フォーム」や有限会社ディアイピィの「DIPSurvey-Free」などがある．これらは，ユーザーが簡単にwebアンケートを作成することを第一に考えたサービスである．そのため，アンケートはGUIを用いて作成する．具体的には，あらかじめ用意されていたテンプレートから目的に合った物を選択し，それらを連結してパズル感覚でアンケートを作成していく．



図3.1：Googleフォームのアンケート作成画面

4. ソフトウェアの開発

4.1　作製したシステムと既存のシステムの差

本研究では先行研究で見られたWEBアンケート作成フレームワークを使用しない．なぜなら，インターフェースの種類やデータの集計方法に大きな制約があるからである．そこで，本研究ではxeeシステムを用いて授業評価アンケートを作成した．xeeシステムでは，Material UIを用いて自由にインターフェースを作ることができる．また，データの集計，配布方法もプログラミングを用いて自由に行うことができる．

4.2　XEEシステムについて

xeeシステムとは林（2016）が発表したオンライン経済実験の基盤システムである．xeeシステムはシステム基幹部分と，その配下にある実験群から構成される．xeeシステム基幹部分はErlang VM上で動作するElixirという言語で書かれたwebアプリケーションの基盤であるPhoenix Frameworkを利用して設計されている．そして，実験群はFacebook社が開発したReact.jsとGoogle Material Designが提供しているUIパーツであるMaterial UI，基幹部分と連携するためのElixirで構成される．



図4.1：XEE.JPのトップページ

4.3　xeeシステムに使われた技術

　Xeeシステムは様々なweb技術を組み合わせて構築されている．本セクションではxeeシステムに使用されたweb技術を紹介する．

4.3.1　Erlang

Erlangは1987年ごろにスウェーデンの電話会社Ericssonで開発された関数型言語である．Erlangのランタイムシステムには，平行性，分散性，耐障害性のサポートが組み込まれている．そのため今では，電気通信，銀行業務，電子商取引など広い分野のシステムに組み込まれている．



図4.1：Erlangのロゴ

4.3.2　Elixir

ElixirはErlang VM上で動作する関数型言語である．これはErlangの使いづらさ（文字列処理が苦手）などを改変し，Ruby風に書けるようにした言語である．



図4.2：Elixirのロゴ

4.3.3　Phoenix Framework

Phoenix FrameworkはElixirで書かれたwebフレームワークのことである．これはRailsやDjangoのようなフレームワークに影響を受けており，MVCパターンにのっとり開発をすることができる．



図4.3：Phoenix Frameworkのロゴ

4.3.4　React

ReactはFacebook社が開発したJavaScriptライブラリである．これはwebアプリケーションのview部分を作ることに特化している．ReactはバーチャルDOMという概念を採用しており，JavaScriptでよりも高速に要素を描画することができる．

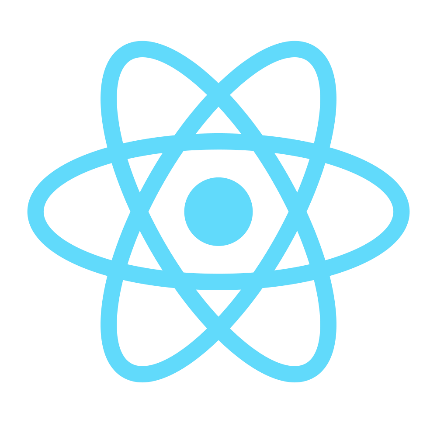


図4.4：Reactのロゴ

4.3.5　webpack

webpackはモダンなJavaScriptアプリケーションのモジュールビルダである．これは，複数のファイル依存関係を反映して一つのファイルにまとめる働きをする．

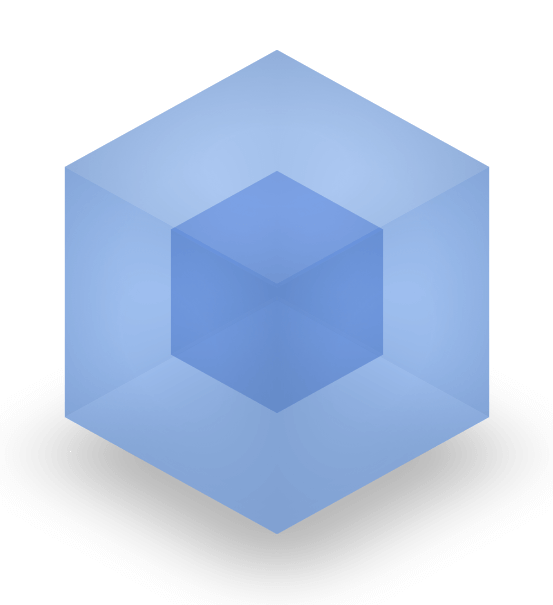


図4.5：webpackのロゴ

5. 授業評価アンケートの開発

5.1　作製した授業評価アンケートの概要

本研究で作製した授業評価アンケートは参加者と司会役の2種類の人が必用である．参加者とは授業を評価する学生たちのことであり，３０～４５人ほどを想定している．また，司会者とはクラスごとに授業評価アンケートを行う人である．司会者は参加者の進行具合に合わせてアンケートシステムを操作する．参加者が評価を終えて，司会者がアンケートシステムを操作すると，データが自動で集計される．そして司会者が集計結果ををcvs形式でダウンロードする ．

5.1　ファイル構成

今回作成した授業評価アンケートのファイル構成を以下に示す．このプログラムのルートディレクトリはcourse\_evaluation\_questionnaireである．

~\xee\apps\course\_evaluation\_questionnaire

│ .babelrc

│ .gitignore

│ LICENSE

│ mix.exs

│ package.json

│ README.md

│ webpack.config.js

│ webpack.production.config.js

│

├─components

│ EvaluationAxis.js

│ Subjects.js

│

├─config

│ config.exs

│

├─host

│ actions.js

│ App.js

│ DownloadButton.js

│ index.js

│ PageButtons.js

│ reducer.js

│ saga.js

│

├─lib

│ │ allais\_paradox.ex

│ │

│ └─allais\_paradox

│ actions.ex

│ host.ex

│ main.ex

│ participant.ex

│

├─participant

│ actions.js

│ App.js

│ Description.js

│ EndQuestion.js

│ Evaluation.js

│ Experiment.js

│ Experiment1.js

│ Experiment2.js

│ index.js

│ Pages.js

│ reducer.js

│ Result.js

│ saga.js

│ Waiting.js

│

└─util

index.js

図5.1：授業評価アンケートのファイル構成

ルートディレクトリ直下には本プログラムの各種設定ファイルを置いている．これから，各フォルダの役割を示す．

・components

このフォルダにはhostとparticipantのファイルからアクセスする配列が格納されている．この配列の内容は「評価軸」と「教科名」である．

・config

　このフォルダにはサーバーサイドの設定ファイルが格納される．

・host

　このフォルダには教師画面を構成するJavaScriptファイルが格納される．

・lib

　このフォルダにはサーバーサイドの挙動を示したElixirファイルが格納される．

・participant

　このフォルダには学生画面を構成するJavaScriptファイルが格納される．

・util

　このフォルダには外部ライブラリを使用する際に記述しなければならない定型文を記述したファイルが格納される．

5.2　操作手順

本アプリケーションは教師と学生がインタラクティブにつながることで成り立っている．つまり，教師がアプリケーションを操作したら，即座に学生側の表示も変更される．同様に，学生がアプリケーションを操作したら即座に教師側の画面にも反映される．つまり，教師画面と学生画面を分けて説明することは難しい．そこで，本節では教師画面と学生画面を同時に説明する．

5.2.1　待機画面

はじめに，待機画面の説明を行う．ここで，教師は学生をシステムにログインさせる．学生はXEEのトップページからログインする．ログインしたら待機画面に移動する．すべての学生がログインしたことを確認したら，教師は「次へボタン」をクリックして説明画面に移動する．



図5.2：教師の待機画面



図5.3：学生の待機画面

5.2.2　説明画面

次に説明画面の説明を行う．ここで，学生は授業評価アンケートの説明を読み，アンケートの回答方法を理解する．すべての学生が説明を読み終えたことを確認したら，教師は「次へボタン」をクリックして「実験」画面に遷移する．



図5.4：教師の説明画面



図5.5：学生の説明画面

5.2.3　実験画面

次に，実験画面の説明を行う．ここで，学生は複数の質問に回答する．質問は「学年学科の確認」と「授業評価アンケート」の２つに分類される．今回開発した授業評価アンケートでは，「楽しさ」「わかりやすさ」「テストの難しさ」の３つで授業を評価した．この評価はcourse\_evaluation\_questionnaire/components/EvaluationAxis.js内で静的に定義されている１次元配列のEvaluationAxisを編集することで変更可能である．この変更には，評価軸の追加も含まれる．EvaluationAxisに評価軸の要素を追加することで，自動的にアンケート画面に反映される．評価軸と同様に，質問する教科も外部ファイル内の１次配列から読み込まれている．質問される教科はcourse\_evaluation\_questionnaire/components/Subjects.js内で静的に定義されている．これも，評価軸と同様に，配列の要素を追加，削除，変更すると，それらがアンケート画面に反映される．

実験画面で学生は，まず学年学科の確認を行う．これはラジオボタンを用いて実装されている．ここでは，回答ミスを予防するために学年と学科のそれぞれで１つ以上の項目は選択できないように設定している．学年学科を選択したら「次へ」ボタンをクリックする．すると，次の授業評価アンケート画面に遷移する．

授業評価アンケート画面は，以下のような外観になっている．この画面は評価される教科が上から順に並んでいる．学生は，教科ごとのスライダーを動かし１～５の間で授業を評価する．全ての評価軸で授業を評価すると「内容確認画面」に遷移する．ここで「確定」ボタンをクリックすることで，この授業の評価を確定させる．この一連の評価を全ての教科で行うと，ページ最下部にある「NEXT」ボタンがクリック可能になる．NEXTボタンをクリックすると，自分が評価したデータがXEEシステムに送信され，アンケートが終了する．



図5.6：教師の実験画面



図5.7：学生の学年学科確認画面



図5.8：学生のアンケート画面(未回答)



図5.9：学生のアンケート画面(回答済み)

5.2.1　終了画面

最後に，終了画面の説明をする．名前の通りここはアンケートの終了を通知する画面であるため，学生側には終了画面が表示される．一方で，教師画面では，全学生のアンケート結果をcsv形式で保存することが出来る．

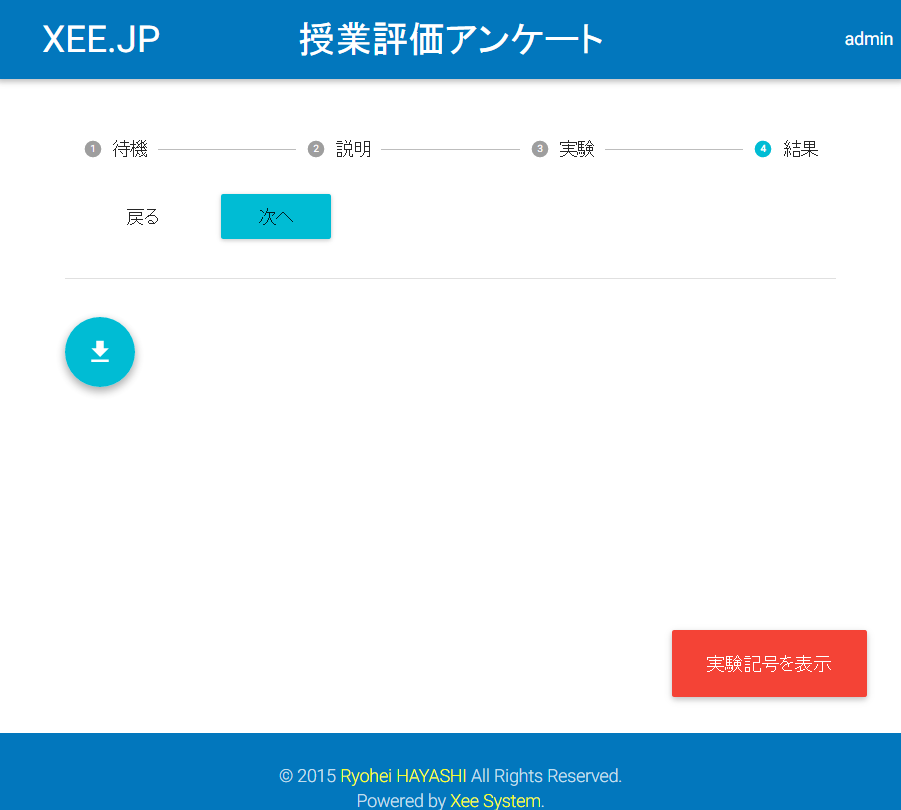


図5.10：教師の結果画面



図5.11：学生の結果画面

6. 考察

本節では５段階評価の不正確性について述べ，それに変わる集団の意見集約方法について考察する．溝（２０１６）は５段階評価システムから得られる調査結果は，合理的かどうかを判断できない．なぜなら，５段階評価システムは個人の回答において評価対象の推移性を保証していないからであると述べている．推移性とは，全ての選択肢において順位付け可能なことである．Sen（１９７０）は推移性を満たさない選択は全ての選択肢が最適となり得るから，合理的な選択であるか否かを判断しがたいと述べている．つまり，５段階評価システムから得られた回答は合理的な選択であるか不明である．

このことを統計的に確かめるために，ボルダルールを利用した多元的評価システムの開発（溝，２０１６）で収集されたデータを再集計した．そのデータとは溝（２０１６）が２０１６年１０月３１?１１月１日にかけて鹿児島工業高等専門学校３年生５クラス（合計１６９名）を対象に行った５段階評価システムと多元的評価システムの比較実験のデータである．この論文では，多元的評価システムは評価対象と評価軸にボルダルールを適用して点数算出を行っていた．しかし，これでは真の序列を再現できなかった．そこで，評価軸の重み付けにボルダルールではなく，被験者がAグループとBグループに何倍差があったかという質問で得られた整数を使用した．すると，真の序列を再現することに成功した．

これは，２つのことを示唆している．

１つ目は，５段階評価よりもボルダルールのほうが個人の意見を正確に集約可能な点である．なぜなら，ボルダルールは全ての比較対象を順位付けて比較したためである．順位付け可能であるということは推移性を満たしているということである．推移性を満たしていない５段階評価に比べて，推移性を満たしているボルダルールを用いたほうが合理的な回答を期待できる．

２つ目は，その状況に適したスコアリングルールを採用することで結果をより良いものに導ける点である．

スコアリングルールとは，1位に点，2位に点，3位に点のように選択肢に得点（スコア）を付けて，その総点和で総合順位を決める集約ルールのことである．なお，選択肢の数をmとすれば

が成り立つものとする（坂井，2013）．

溝（２０１６）では評価軸のポイント計算においてボルダルールを適用した．その結果，正しい序列は得られなかった．一方，本説で再集計する際には被験者がAグループとBグループの間に感じた点数差を用いたスコアリングルールを適用した．その結果，正しい序列を得ることができた．これは，ボルダルールが要素間の距離を考慮しないために発生したと考えられる．ボルダルールでは，最下位から順に１点，２点，３点と得点を付与していき１位には評価対象数だけの得点が与えられる．つまり，評価対象の序列の要素間にそれぞれ異なる距離が設定されていても，それらを無視してしまうということである．例えば，A,B,Cという選手がマラソンで競っているとしよう．AとBはほぼ同着でゴールし１位がAで２位がBであった．一方Cはトップ争いから脱落し，かなり遅れてゴールした．この場合，１位と２位にほとんど差がなく，２位と３位には大きな差がある．それにもかかわらず，ボルダルールで集計すると，Aが３点，Bが２点，Cが１点である．この場合，AとBにはほとんど点数差がなく，BとCには大きな差があると考えるのが自然である．このように，５段階評価よりもボルダルールが優れているからと言って，常にボルダルールを適用すればいいとは限らない．評価対象の特性に合わせて，最適な評価システムを採用することが重要である．そうすることで，より正確に人々の意見を集約することが出来る．

本節をまとめると，以下の２つに集約される．

５段階評価システムは推移性を満たしていないため，回答者からの合理的な回答を期待できない．

５段階評価システムよりもボルダルールのほうが回答者からの合理的な回答を期待できる．しかし，ボルダルールも万能ではない．質問対象の性質に合ったスコアリングルールを適用することが重要である．

7. 今後の課題

今回はXEEシステムを使用したため，制約の多いアンケートになった．今後は，試験監督なしで，いつでもどこでも利用できる授業評価アンケートシステムを構築したい．

8. まとめ

本研究では，モダンなWEB技術を使い授業評価アンケートを構築した．これはXEEシステムをベースに作られている．ユーザーインターフェースにはmaterialｰUIを使用した．その結果，わかりやすく見た目に美しいアンケートシステムが完成した．

参考文献

林、2016

溝、2016

坂井，2013

Borda, J. C.: Memoire sur les elections au scrutin, Histoire de l’Academie Royal des Sciences, pp. 657664, 1781.

Sen, K. A.: Collective choice and social welfare, Oliver and Boyd, 1970

<http://www.geocities.jp/m_hiroi/func/abcerl01.html>

謝辞

本研究を行うにあたり，指導教員の岸田一也准教授から，丁寧かつ熱心なご指導を賜りました，ここに感謝の意を表します．