

# eポートフォリオシステムの開発

## Development of e-portfolio system

1012005 川口 拓郎 Takuro Kawaguchi

### 1 背景

公立はこだて未来大学では2種類の授業形態が用いられている。1つは座学などに代表される受動的な学習であり、もう1つはシステム情報科学実習や高度ICT演習などに代表される能動的な学習だ。受動的な学習とは、教師が学生に一方的な知識伝達を行い、学生たちの評価を試験などの結果によって行うことである。能動的な学習とは、学生が主となり、問題解決に向けて活動・学習を行い、それらの評価は、学生たちの問題解決に向けた意欲や活動記録などによって行うことであり、総称してPBL(Project Based Learning)という。PBLは公立はこだて未来大学だけでなく、他の大学でも授業として取り入れている事例がある。例えば、会津大学では「ベンチャー体験工房:PBL 初級編」という講座を行っている[?]。PBLでは学生が課題や問題の解決に向けた自主的な活動を評価し、ITに関する知識をより実践的な手法から学ぶことが出来ると言える。

また、PBLでは学習の過程で多くの成果物を残すことができる。例えば、受動的な学習では、教員から課された課題の解答が成果物として該当する。能動的な学習では、問題解決時に得られた知識や技能だけでなく、それらを会得するまでのプロセスも成果物として残すことが出来る。

公立はこだて未来大学における両授業形態の学習支援であるが、受動的な学習では、Moodleと呼ばれるCMS(Course Management System)によって学習支援がされている。例えば、教員がMoodle上で作成された小テストは指定された期限内ならば、学生はいつでも小テストを行うことが出来る。その上、学生一人一人の小テストはMoodle上で自動で採点されるため、教員は学生全員の採点をする手間を省くことが出来る。その一方で、能動的な学習では、十分な学習支援がされていないのが現状である。Moodleで十分な支援が行えない理由

は、問題を解決するプロセスを記録・蓄積する仕組みが整っていないからである。これらの記録・蓄積方法は各プロジェクトチーム及び学習者個人によって様々であり、一元管理がなされていない。

### 2 課題の設定

前述した通り、本学ではプロジェクト学習や高度ICT演習をはじめとしたPBLが盛んである。そこでは、学生が主体となり、問題提起から解決まで行い能動的に学習している。しかし、これらの学習は座学とは違い、成果を残すまでのプロセスも成果物とするので、学内に存在する既存のシステムでは適切な評価をすることができない。そこで私たちは多様化する問題とその解決方法に対する教員からの適切なフィードバックや、解決方法として作成された成果物を管理できるシステムがあれば、更に本学の学習体系を成長させていくことができるのではないかと考えた。そこで本プロジェクトでは、本学で用いられていない能動的な学習を支援するためのeポートフォリオシステムを開発することにした。eポートフォリオとは、学習者自らの学習の記録を電子的にまとめたものである。eポートフォリオを能動的な学習に用いることで、学生は自らの学習をまとめることができ、指導者も学生が作成したeポートフォリオを用いた適切な評価がしやすい。よって、eポートフォリオの作成を支援するシステムの開発は能動的な学習における学生や指導者の支援をすることにつながる。既存のeポートフォリオシステムとしてMaharaやSakaiがある。この2つのeポートフォリオシステムを調査した結果、成果物の共有のしやすさの点ではSakai・Maharaともに優れていることがわかった。しかし、学習の振り返りはMaharaよりSakaiのほうが優れていることがわかった。一方、UI(操作性)や学外への公開の2点にではMaharaのほうが優れていることがわかった。しかし、どちらのシステムもカスタマイズしにくいことがわかった。このこと

から、既存の e ポートフォリオシステムをカスタマイズして使用することは難しいということがわかった。そこで、本プロジェクトでは既存の e ポートフォリオシステムを利用し、カスタマイズして使うのではなく、既存の e ポートフォリオシステムが持つ主な機能を持った e ポートフォリオシステムを新規開発することにした。まず、e ポートフォリオのシステムに必要な機能について考える。PBL は能動的学習であるから、学生が自ら問題提起から解決まで行えること、ある問題への解決方法として作成された成果物を管理できること、教員が学生の成果物へフィードバックできることが必要である。私たちが開発する e ポートフォリオシステムに必要な機能として、学生による問題提起機能、学生による問題解決機能、成果物の振り返り機能、教員による評価とコメント機能があげられる。そしてそれらを備えた e ポートフォリオシステムを新たに開発することを課題として設定した。加えて、e ポートフォリオシステムを開発し、実際に学生や教員に使用してもらい、フィードバックを得ることが重要だと考える。そのため、フィードバックを得ることでシステムの改善を図る PDCA サイクルを回すことも課題とする。

### 3 課題解決のプロセス

まず、課題解決のプロセスとしてはまず e ポートフォリオシステムに必要な機能の洗い出しを行う。次に技術検証として洗い出した機能を持つような技術検証用の試作システムの設計を行い、そのシステムの実装を行う。そのシステムに使用した技術について記述する。ソースコードのバージョン管理には Git を用いた。リモートリポジトリは GitHub でホストし、GitHub が持つ課題管理機能やプルリクエスト機能も利用した。どちらもメンバーに経験者が多かったのが採用の理由である。未経験の者は各自プロジェクト時間中に学習した。その際、理解の確認のため練習用のリポジトリでプルリクエストを行うなどした。

また、バージョン管理において、Scott Chacon の GitHub Flow より以下を原則とした。

- master ブランチのものは常にデプロイ可能であること
- 新しい何かに取り組む際は、それを説明している名前をつけたブランチを master から作成し、そこで

行うこと

- ブランチをマージされたい時やフィードバックや助言がほしいときは、プルリクエストを行うこと
- プルリクエストのマージはレビューを受けて了承を得てから行うこと

プルリクエストのレビューは、プロジェクトメンバー並びに担当教員によって行われた。これらの取り組みにより、プロジェクトメンバーが集合していないときでも開発を潤滑に行うことが出来た。そして、残されたログを活用して活動の振り返りを行うことも容易であった。

プログラミング言語には Java を用いた。これは本学の学部 2 年の情報システムコースの必修講義で、メンバーが全員ある程度学んだからである。

単体テストには JUnit を用いた。これは Eclipse や NetBeans などの主要な IDE がサポートしているからである。

プロジェクトの管理には Apache Maven を用いた。これは使用するプログラミング言語が Java だからである。Maven がもつモジュール機能を利用して、モデルとビューを異なるソースツリーに分けた。これはモデルとビューならびにビューモデルの分離を促すためである。また、Maven がもつ依存性の解決機能を用いた。これはインターネット上のライブラリ等をビルド時に自動的に取得するので、外部のライブラリを簡単に利用できるからである。

Web フレームワークである Vaadin を用いた。これはプロジェクトメンバー全員が本学の情報システムコースに所属しているため UI デザインにおける知識量の無さが弱みと考えられる我々にとって、Vaadin が持つ、Java の知識のみで Web アプリケーションを作ることが出来るという特徴が適していると考えたからである。

データベースには Couchbase Server を用いた。これは JSON を値に取る Key-Value ストアとして利用できるもので、モデルを示すクラスのインスタンスを容易に格納出来る点が簡単で、開発の効率が上がると考えたからである。また、アジャイル開発を用いる本プロジェクトにおいて、モデルを示すクラスに開発途中で変更を加える事があると考えられるが、そういった変更に対応出来る点も理由である。

実装はまず、Vaadin の日本語の資料が少ないため、公

式サイトの英語のドキュメントを日本語に翻訳することから始めた。ドキュメントは巨大なので、重要と感じた節のみ翻訳を行った。翻訳は Google ドライブのドキュメントの同時編集機能を用いて、複数人同時に行った。こうした取り組みは、後述のビューの実装を効率的に行うことを目的としていた。

それと同時に、GitHub でホストされたりモトリボジトリを作成し、Maven でプロジェクトを作成し、モデルの実装を行った。実際に実装したのは、ユーザを示すモデル、ユーザの行動を示すモデル、ユーザの投稿を示すモデル、ユーザが投稿した画像を示すモデルである。これらのモデルをデータベースへ追加、読込、変更、削除するメソッドも必要なものだけ実装した。そして、それらに対応する単体テストを作成して、不具合を発見して修正し、品質を上げた。また、それら进行操作するコマンドラインインターフェースの実装も一部行った。これは、後述のビューを実装する前に、動作を試験したり、テスト用データを容易に作成することを目的としていた。

Vaadin のドキュメントの翻訳をある程度行ってから、それを参考にしてビューの実装を行った。ユーザの行動を一覧、追加する画面や、ユーザを登録する画面、並びにサインインを行う画面を、Vaadin と先述のモデルのメソッドを利用して実装した。

デプロイは、本学に置かれたサーバーで Apache Tomcat を用いて、中間成果発表会に間に合わせるように行った。Tomcat により、Web ユーザーインターフェースから容易にデプロイするプロジェクトを更新することが出来た。その試作システムの実装を通して e ポートフォリオシステムに必要な機能を再度確認・修正し、e ポートフォリオシステムの設計・実装を行なっていく。そして、実際に e ポートフォリオシステムを開発し、ステークホルダーである大場先生へのヒアリングを通して、実装した機能についての修正を行なっていく。完成したら本学の学生・教員・職員に使用してもらい、その使用した意見を評価としてフィードバックを得る。そのフィードバックを分析し、評価が良好でない場合や、実装すべき機能などが合った場合はそのフィードバックをもとにシステムの改善・機能追加をする。このように、PDCA サイクルを回すことによってシステムを改良し、課題を解決していく。

## 4 成果

前期では、成果物として e ポートフォリオシステムの技術検証用の試作システムを構築した。これにより、e ポートフォリオシステムの基本的な機能が構築できるか技術検証を行い、本プロジェクト開発における問題点の発見を行った。まず、ユースケース図を作成することでシステムの設計について学んだ。次に、ペーパープロトタイプを用いての画面設計の方法やサーバー運用等についての知識を得た。実装の過程では、Web アプリケーションフレームワークである Vaadin が Java 言語のみでの開発が可能であることから、Java の基礎力や応用力を養えた。また、プロジェクトのソースコード等の管理のため分散型バージョン管理システムである Git を学習することで、個々のソースコード変更履歴を記録・追跡することを可能にし、分散アジャイル開発のプロセスを学んだ。また、Vaadin の日本語の文献が少なかったため、Vaadin の公式マニュアルを日本語訳した。前期の段階では、e ポートフォリオシステムを開発する段階には至っていないため、e ポートフォリオシステムのマニュアル、利用者からのフィードバックを成果とすることはできなかった。

後期では、成果物として e ポートフォリオシステムを構築し、名称を miite と決定した。miite は現段階では一部の機能に対して問題点があることが分かり、実際に本学の学生や教員に利用していただける状態には至っていない。だが、今後この問題点に対しての修正を行い miite を本学の学生や教員に利用していただける状態まで実装を行い、成果物として完成させていきたい。この miite の構築にあたって、システムの開発のために必要な技術をいくつか学ぶことができた。まず、画面設計ではペーパープロトタイプを用いての UI、UX を考慮したシステム作りについて学んだ。次に、どのようなシステムを作るかを決めそれに対しての実装を行い、定期的に行ったヒアリングから実装したシステムに対しての評価をいただくことで、これに対しての改善点を発見し修正していくという PDCA のサイクルを繰り返したことからチームでの開発における流れを学んだ。実装においては、HTML や CSS、JavaScript、Java 言語等を主に用いて進めていき、これらの基礎力、応用力を身に付け、テンプレートエンジンとして用いた Mustache という技

術についても学んだ。Java 言語での実装においては軽量 Web アプリケーションフレームワークである Spark を用い、この技術についても学ぶことができた。データベースには Couchbase Server を用い、データベースとの連携を行ってシステムを構築する技術を習得した。

## 5 今後の課題

前期の活動では、メンバーのタスクの進捗状況を管理しながら実装するべきということが課題として挙げられる。なぜなら、技術検証と技術習得に予定よりも多くの時間がかかり、技術検証が十分に行えなかったためである。これを活かし後期の活動では実装と評価を繰り返し、システムをよりよいものにするために、トラブルが起きたときもスケジュールの遅れに配慮し作業する必要があると判明した。後期の活動では、データベースを設置する場所が課題に挙げられる。現在私達がシステムで使用しているデータベースは学内に1つ、学外用にさくらVPSを1つ借りている。学内で開発を行う時は学内のデータベースに接続し、学外で開発を行う際はさくらVPSに接続するように設定している。これは、学内ネットワークのセキュリティの問題でポートの80にアクセス出来ないようになっているためである。学内のデータベースにも学外からではアクセス出来ないのが現状である。これによって学内からアクセスした場合と学外からアクセスした場合で表示されるデータやデータベースの設定が異なるものになってしまうことが課題である。

## 参考文献

- [1] 東京学芸大学 森本研究室, 「教育分野における e ポートフォリオとは」, <http://draco.u-gakugei.ac.jp/eportfolio/>, (last access 2014-07-20)