

ソフトウェア開発プロセスを 実践的に学修するための改善と効果

柿崎 淑郎^{1,a)}

概要: 本取組は、Webシステム開発を題材として、ソフトウェア開発プロセスの一連の流れを学修することを目的とした授業改善の取り組みである。ソフトウェア開発プロセスの学修は、プロセスをいくつかの段階に分けて複数の科目に分割する学修方法や、一連のプロセスを1科目で実施するがプログラミングは行わない理論中心の学修方法などがある。複数科目に分かれるとその関係性を十分に理解できなかったり、実装を行わないと理論の体得が十分でなかったりする問題がある。また、多くのプログラミング学修が個人での取り組みであるのに対して、実際のシステム開発では複数人によるチーム開発が基本である。そこで、ソフトウェア開発プロセスを要件定義から実装まで一貫して行う演習科目として、Project-Based Learningでチーム開発に取り組む形式に改善する取り組みを行った。これによって、ソフトウェア開発プロセスを1科目で密に学ぶことができるとともに、理論と実践の両面からアプローチすることにより、実学としての体得を可能としている。この取り組みによって、学生の理解度向上、意欲・興味の向上、将来像の形成などに寄与している。

Improvement and Effectiveness for Practical Study of Software Development Process

YOSHIO KAKIZAKI^{1,a)}

1. はじめに

情報系、特に情報工学系大学において、プログラミングは必修科目として定着している。しかしながら、プログラミングは何かの目的を達するための手段であり、プログラミングによって何をするかを明確にすることは重要である。本稿では、Webシステム開発を題材としてソフトウェア開発プロセスを実践する演習科目における授業改善の取り組みについて紹介する^{*1}。

2. 趣旨

本取組は東京電機大学未来科学部情報メディア学科3年前期専門選択科目である「サーバプログラミング演習」に

おいて、図1に示すソフトウェア開発プロセスに対応するように（橙部は本科目の範囲）、過年度に学んだ知識と技能を集約し、発揮することを目的とした授業改善の一環である。

本学科を卒業し就職する卒業生の多くは、システムエンジニア職に就くことが多い。本科目では、単にプログラミングをするのではなく、ソフトウェア開発プロセスに則り、要件定義、設計、実装、テスト、運用という一連の流れを一貫して実践しながら学修することに重きを置いている。このプロセスの各段階に関連する内容は、過年度における各科目で学修している。そのため、それまで別個の科目として独立し、科目間の関係性が不明瞭であった学生も、ソフトウェア開発プロセスによる一貫したシステム開発を行う過程において、学修してきた内容がどのように位置付けて、どのような役割を果たすのかを認識することができ、またそれらの復習にもなり、より意味のある知識体系の確立に寄与する。そして、就職後の将来像形成に貢献する。

¹ 東京電機大学
Tokyo Denki University

^{a)} kakizaki@mail.dendai.ac.jp

^{*1} 本取組は情報システム教育コンテスト ISECON2017 において優秀賞を受賞した。



図 1 ソフトウェア開発プロセス

特色・優位性

- 過年度に学んだ知識を総合的に活用することができる演習構成
- ソフトウェア開発プロセスの実践
- チームによる複数人でのシステム開発の体験と実践
- VPS (Virtual Private Server) 利用による実践的な演習環境

独創性

- PBL (Project-Based Learning) 学修によるアクティブラーニング化
- チュートリアル形式による演習主体の実践的学修
- 職業観の醸成と将来像形成への寄与

3. 取組内容

本科目は、専任教員 1 名 (筆者)、副手 (大学院生) 6 名で講義にあたる。受講生は概ね 80 名であり、全体を 2 グループに分け、各グループは隔週で講義を受ける。本科目は 3 年前期に開講され、2 コマ × 8 回で実施される。前半 3 回は開発環境構築と MVC モデルに基づくサーバサイド Java プログラミングのチュートリアルを実施しつつ、要件定義、基本設計、詳細設計を進める。後半 4 回はチームによる Web システム開発に充てている。最後の 1 回は最終発表会として、チームで開発した Web システムの発表とその取り組みについて発表させるとともに、その発表を聞くすべての受講生・教員・副手はルーブリックを用いた評価を実施している。

受講生は 4 名ないし 5 名のチームを作り、チームで 1 つの Web システムを開発する。そのチーム内では、プロジェクトマネージャ、アプリケーションスペシャリスト、IT アーキテクト、品質保証マネージャの役割をそれぞれに与え、その役割を自覚し、責任を持って取り組むように指導している。各チームには 1 名の副手がメンタとして専属する。受講生にとって、プログラミング科目におけるチーム開発は初めてとなる。「自分だけがわかればいい」「動けばいい」という独りよがりのプログラミングから、チーム開発によって「他の人のプログラムと連携するためのプログラムインタフェース」「他の人が修正できるプログラムコード」の重要性を気付かせることができる。さらに、役割分担によって、自らの進捗が他のメンバに影響を与えることや、提出物の期限によって、タスク管理の重要性も身につ

き、協調性を高めることができる。

本学科では 1 年次より一貫して Java によるプログラミング学習を行っている。しかしながら、サーバサイド Java については本科目が初めてとなるため、本科目で学修する必要がある。従来では通常の講義 + 演習形式で実施していたが、2016 年度よりチュートリアル形式へ変更した。本講義では講義に必要な資料は学期開始前から全て公開している。講義形式で利用していた説明資料を受講生が読めば理解できるように修正加筆するとともに、その内容を理解できたかどうかを確認することができる演習問題を多数準備している。これによって、いつでも講義に必要な知識を学ぶことができる能動的学修を可能としている。

この取り組みによって、講義時間内における教員の説明時間は大幅に減った。反面、講義時間内における受講生からの質問は増えた。質問が増えたことで、受講生がどこまでを理解し、どこに躓いているのかを把握することができるようになった。なお、頻繁に受け付ける質問については講義資料に加筆することで対応している。また、講義時間内の説明時間が減ったため、学生が手を動かしてプログラミングをする時間が増えることとなり、演習型授業の本旨を全うしている。

本科目ではソフトウェア開発プロセスに従って、チームで Web システムの開発を行う。その過程においては、要件定義書、外部設計書、内部設計書などの各種ドキュメントが作成される。これらドキュメントの役割や記載すべき内容については他科目で学修しているが、実際にドキュメントを書き上げるのは本科目が初めてとなる。受講生には最低限書かれるべき内容が箇条書きされたテンプレートのみが配布される。受講生はテンプレートと理論科目で学修した成果に基づいて、ドキュメントを作成する。このドキュメントは講義毎に作成され、提出される。提出されたドキュメントは各チーム担当副手によって添削が行われる。その後に担当教員がチェックをし、次回講義時に各チームへ返却されるとともに、メンタリングを実施する。メンタリングでは、提出されたドキュメントへのコメント、チーム活動の状況確認、当日の作業内容確認を実施する。メンタリングは各チーム担当副手と担当教員があたる。ドキュメントは複数回の修正を経て、完成度を高めていく。

本科目の特長として、実際の開発現場に近い演習環境を提供している点がある。2015 年度より、本学 PBL 教育支援

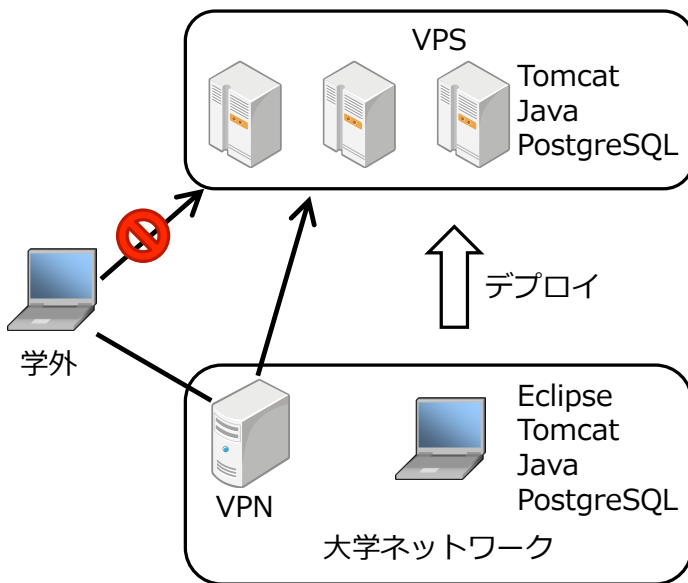


図2 VPSを利用した演習環境

プログラムの支援を受けて、VPSをチーム毎に貸与している。図2に示すように、従来では各人のパソコン上で開発したWebシステム(Eclipse上のTomcatとPostgreSQL)を動かすに留まっていたが、VPS上にデプロイすることにより、誰もが開発されたWebシステムにアクセスできるようになる。これによって、実際のWebシステムと同じ環境で動作させることができ、より実感が沸くとともに、チームメンバ以外でも開発したシステムを利用できるため、自らの作品のクオリティ向上にも寄与している。また、実際の開発現場と同様の体験をすることができ、開発のモチベーション向上、就職後の働く自分像の形成などに寄与する実践的学修ができています。なお、授業時間数の関係上、セキュリティを考慮したWebシステム開発は行っていないため、VPSへのアクセスは学内回線からのみに制限している。

2016年度からはGMOインターネット株式会社とタイアップし、VPS運用支援、最終発表会での講評などの協力体制を構築してきた。学外者が最終発表会に参加することで、受講生は気が引き締まり、しっかりとした発表を行っている。同時に、学外者、しかもWeb業界で活躍するエンジニアから講評をもらうことで、自分たちの取り組みが評価されているという達成感も得られている。

4. 評価と効果

2016年度授業評価アンケートの結果に基づいて説明する。授業時間外学修は1週間あたり平均して2時間以上の回答が9割であった。授業内容の難易度は難しかったが54%、やや難しかったが37%であり、合わせて9割が難しかったとの認識を示している。この結果から、学生はかなりの努力をして科目に取り組んでいることがわかる。前述

のように、演習時間中は教員・メンタによるチェックとメンタリング、また直面した問題解決への対応が主となるため、必然的に授業時間外学修の時間は多くなる。また、学問として新しいことに取り組むことは常に難しいことであり、それに立ち向かい乗り越えることこそが、学生の成長に繋がると考える。事実、7割強の学生が「興味・関心が高まった」と回答しており、就職後の働く自分像の形成にも寄与しているものと考えられ、本科目の難易度の設定は適切であり、また目的も達成されているといえる。これらの取り組みの正しさは、卒業生の評価からも垣間見られる。学科FD活動のために学部卒業生(現修士学生)に対して、「卒業後に役に立ったと思う科目」の調査を実施した。「サーバプログラミング演習」は2位に2倍以上の大差をつけ、圧倒的な支持を集めた。自由記述によれば「要件定義の重要性を理解できた」「タスク管理を学べた」「チーム開発の難しさを実感できた」などの意見が見られた。

就職した卒業生からは「新入社員研修で同じようなことをやっている」「実際の業務でやることを習っていたことに気付いた」などの話も寄せられている。

このように、本科目における取り組みは高い評価を受けており、特に卒業後においてその取り組みの意味が理解されるなど、高い教育効果を発揮する実践的学修であるといえる。

これらの取り組みにより、従来型授業より、以下のような変化があった。

- 受講生に公開している講義資料は充実した
- 受講生が手を動かす演習時間が増加した
- 教員の全体に対する説明時間は減少した
- 受講生からの質問が増えた
- 従来型授業と比較して受講生の理解度に変化はない
- 本科目の最終成果であるWebシステムの完成度は向上している
- 受講生の満足度は向上している

説明時間を減らしたにも関わらず、受講生の理解度は低下していない。この理由の1つとしては講義資料を充実させたことが挙げられる。あるいは、従来型授業において受講生の能力を低く見積もりすぎており、殊更に説明不要なことを説明することで、ある意味無駄な時間を浪費していたのかも知れない。また、演習時間が増加したことによって、システム開発に取り組む時間が増えたためか、最終成果物の完成度は向上している。結果的には教育効果が高まっており、受講生の満足度も上がっている。

謝辞 本取組は東京電機大学教育改善推進室「PBL教育支援プログラム」の支援を受けた。VPSの利用に当たっては永尾泰之氏をはじめGMOインターネットConoHaの支援を受けた。ここに深謝の意を表する。