

Pembelajaran Rekayasa Perangkat Lunak dengan Metode *Problem-Based Learning* di Politeknik Negeri Batam

Nur Zahрати Janah *

* Politeknik Negeri Batam

Program Studi Teknik Informatika

Parkway Street, Batam Centre, Batam 29461, Indonesia

E-mail: nur.zahrati@polibatam.ac.id

Abstrak

Mata kuliah Rekayasa Perangkat Lunak (RPL) merupakan salah satu mata kuliah inti yang dibutuhkan dalam bidang Teknik Informatika. Pembelajaran RPL memiliki banyak tantangan karena mahasiswa diharapkan dapat menciptakan sendiri berbagai artefak dalam proses pengembangan perangkat lunak, tidak hanya sekedar menghafal materi saja. Masalah yang sering dihadapi adalah mahasiswa tidak bisa menyelaraskan hasil pekerjaan mereka mulai dari tahap awal siklus hidup pengembangan perangkat lunak, sampai ke tahap akhir implementasinya. Ketersediaan buku yang membahas tahap-tahap RPL sampai implementasinya yang dilengkapi dengan contoh kasus yang mudah dipahami mahasiswa pun masih belum ada. Oleh karena itu, dalam artikel ini diusulkan suatu metode pengajaran untuk mata kuliah RPL dengan menerapkan *problem-based learning*. Usulan yang dibuat mencakup susunan materi; permasalahan, proyek dan tim; sesi di kelas; penggunaan kaskas; dan metode penilaian. Pembuatan usulan ini dilakukan dengan mempertimbangkan luaran pembelajaran RPL di masa lalu, serta keterkaitannya dengan mata kuliah lain seperti Proyek Akhir I (PA I) dan Pemrograman Berorientasi Objek (PBO).

Kata kunci: **Rekayasa perangkat lunak, orientasi objek, *problem-based learning*.**

Abstract

Software Engineering (SE) is one of the core courses in Informatics Engineering. Teaching SE faces many challenges because the objective of this course is that the students are able to create many artifacts along the software development process, not merely recalling information or theory that has been taught. It is often found that students cannot synchronize their work starting from the early phase of software development until the last stage of implementation. The unavailability of reference books that explain software engineering steps, including the implementation, accompanied by relevant case study that is easy to be understood by students is also a hindering factor. Therefore, this paper will propose a teaching method for SE course which implements problem-based learning. The proposed method includes material composition; problem, project, and team; sessions in the class; tools utilization; and assessment methods. This method considers the outcomes of SE course in the past, and its tight relation with other courses, such as Final Project I and Object Oriented Programming.

Keywords : Software engineering, object orientation, problem-based learning.

1 Pendahuluan

Mata kuliah Rekayasa Perangkat Lunak (RPL) merupakan salah satu mata kuliah inti yang dibutuhkan dalam bidang Teknik Informatika. Pembelajaran RPL memiliki banyak tantangan karena mahasiswa diharapkan dapat menciptakan sendiri berbagai artefak

dalam proses pengembangan perangkat lunak, tidak hanya sekedar menghafal materi saja. Masalah yang sering dihadapi adalah mahasiswa tidak bisa menyelaraskan hasil pekerjaan mereka mulai dari tahap awal siklus hidup pengembangan perangkat lunak, sampai ke tahap akhir implementasinya.

Saat ini di pasaran telah banyak beredar buku yang berisi mengenai pengembangan perangkat lunak, baik

menggunakan bahasa populer seperti [1] dan [2] maupun buku teks yang banyak dipakai dalam perkuliahan seperti [3], bahkan buku teks asing yang telah dialihbahasakan dengan bahasa Indonesia seperti [4]. Demikian pula buku-buku yang membahas mengenai pemodelan menggunakan UML, seperti yang dibahas dalam [5], [6] dan [7].

Namun demikian belum banyak buku yang membahas mengenai proses pengembangan perangkat lunak dari awal sampai akhir dengan menggunakan metode berorientasi objek dan pemodelan UML serta pengimplementasiannya menggunakan bahasa pemrograman tertentu. Memang telah ada buku yang membahas mengenai hal tersebut, misalnya seperti [8], namun dalam buku tersebut tidak menggambarkan secara lengkap proses pengembangan perangkat lunaknya, hanya menitikberatkan pada tahap implementasi saja.

Oleh karenanya, di dalam penelitian ini akan diusulkan sebuah susunan materi bahan ajar dan metode pembelajarannya yang diharapkan dapat membantu mahasiswa menghubungkan antara mata kuliah Rekayasa Perangkat Lunak dan Pemrograman Berorientasi Objek. Dengan begitu, Proyek Akhir I akan mudah bagi mereka.

2 Rekayasa Perangkat Lunak

Menurut IEEE, definisi Rekayasa Perangkat Lunak (RPL) adalah [3]:

- (1) Pendekatan yang sistematis, berdisiplin, dan dapat dikuantifikasi untuk pengembangan, pengoperasian, dan pemeliharaan perangkat lunak, yaitu penerapan prinsip rekayasa terhadap perangkat lunak.
- (2) Studi terhadap pendekatan (1).

Roques [6] menyajikan materi rekayasa perangkat lunak dengan membedah beberapa studi kasus, diantaranya: automatic teller machine, flight booking system, coin operated pay phone, dan training request. Dari setiap studi kasus tersebut, penulis berusaha menunjukkan bagaimana mengidentifikasi aktor, use case, membuat diagram use case, skenario use case, membuat diagram sequence, dan diagram kelasnya. Pressman [3] juga melakukan pendekatan studi kasus dalam menyajikan materinya. Kasus pembuatan aplikasi SafeHome menjadi contoh sepanjang pembahasan materi-materi dalam buku ini. Sayangnya, tidak disajikan contoh pemodelan sistem SafeHome yang komprehensif beserta implementasinya.

Sedangkan Nugroho [9], menyajikan materi rekayasa perangkat lunak dengan metode Unified Sytem Development Process (USDP). Dalam bukunya dijelaskan pengertian Unified Modeling Language (UML) dan USDP, membuat model use case,

merumuskan spesifikasi kebutuhan pengguna, analisis dan perancangan sistem menggunakan metode USDP, dan diakhiri dengan implementasi serta pengujian. Nugroho dalam bukunya yang lain [8] memulai penjelasannya dengan diagram kelas, kemudian mengimplementasikannya dengan bahasa pemrograman Java.

Pembelajaran RPL dengan menggunakan studi kasus telah dibahas dalam [10] dan [11]. Penggunaan studi kasus dapat membantu mahasiswa dalam memahami konsep RPL dengan permasalahan-permasalahan dalam studi kasus yang harus mereka selesaikan. Permasalahan dibuat semirip mungkin dengan kasus yang mungkin dihadapi di dunia professional.

Fernandez-Samaca dan Ramirez [12] mengusulkan skema *problem-based learning* yang dapat diterapkan pada mata kuliah bidang rekayasa. Skema yang diusulkannya meliputi permasalahan yang diberikan kepada mahasiswa, proyek dan tim, sesi tutorial, penilaian, dan sumber daya yang diperlukan. Sistem *problem-based learning* yang mereka usulkan telah berhasil meningkatkan kinerja mahasiswa dan daya pemahamannya jika dibandingkan dengan metode tradisional.

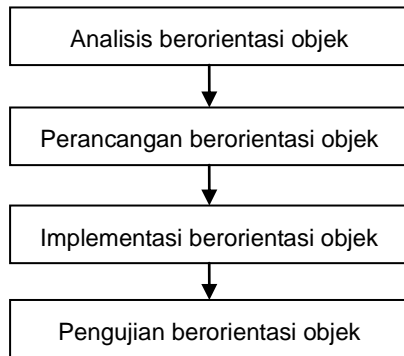
Desai dan Joshi [13] mengusulkan pembelajaran berbasis aktivitas untuk RPL. Metode yang diusulkan menggunakan berbagai strategi untuk membuat mahasiswa lebih tertarik dan meningkatkan keterlibatan mahasiswa dalam proses siklus hidup perangkat lunak dengan cara mengintegrasikan mata kuliah RPL dengan proyek kecil (*mini project*), sesi interaktif di kelas, sesi pemecahan masalah, permainan dan bermain peran, kuis, dan penggunaan kaskas RPL serta studi kasus yang relevan. Hasil dari pendekatan ini tidak hanya meningkatkan kualitas pemahaman mahasiswa terhadap mata kuliah yang diajarkan, tetapi juga meningkatkan kemampuan komunikasi, kerjasama tim, pemecahan masalah, dan kemampuan analitik.

3 Pembelajaran Pengembangan Perangkat Lunak di Politeknik Negeri Batam

Menurut silabus Teknik Informatika Politeknik Negeri Batam [14], tujuan pembelajaran mata kuliah Rekayasa Perangkat Lunak (IF-1413) adalah menjelaskan dan menerapkan prinsip dasar rekayasa perangkat lunak baik secara prosedural secara singkat maupun berorientasi objek secara detil yang berkaitan dengan kegiatan analisis kebutuhan, analisis perangkat lunak, perancangan, implementasi, pengujian dan kualitas perangkat lunak. Porsi materi pengembangan perangkat lunak berorientasi objek lebih diutamakan daripada cara prosedural yang sekarang mulai ditinggalkan.

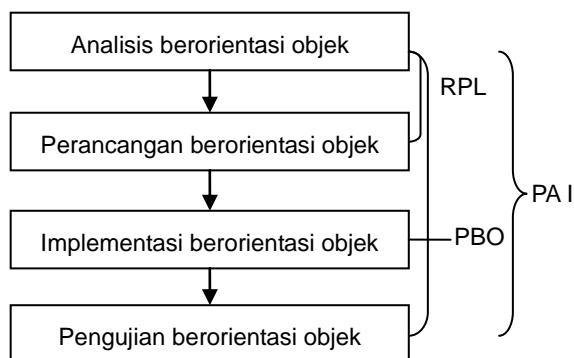
Metode berorientasi objek (OO) adalah cara

mengorganisasi perangkat lunak sebagai kumpulan objek-objek yang mencakup data dan operasi yang diberlakukan terhadapnya. Metode OO banyak dipilih karena dapat meningkatkan produktivitas dalam proses pengembangan perangkat lunak [15]. Metode OO seperti dijelaskan pada Gambar 1 meliputi rangkaian aktivitas analisis, perancangan, pemrograman, dan pengujian berorientasi objek.



Gambar 1: Proses pengembangan perangkat lunak berorientasi objek

Proses analisis dan perancangan berorientasi objek saat ini diajarkan pada mata kuliah RPL. Mata kuliah tersebut juga mencakup petunjuk teknis pengujian perangkat lunak. Sementara itu, mata kuliah IF-1217 Pemrograman Berorientasi Objek (PBO) memberikan pemahaman konsep pemrograman berorientasi objek dan implementasinya dalam pembuatan program dengan bahasa pemrograman Java. Untuk mempraktekkan kemampuan teknis pembuatan perangkat lunak sederhana, mahasiswa diwajibkan mengambil mata kuliah IF-1412 Proyek Akhir I (PA I). Dalam PA I diharapkan mahasiswa mampu menerapkan praktik pengembangan perangkat lunak yang baik sesuai yang telah diajarkan pada mata kuliah tahun pertama, terutama RPL dan PBO [14].



Gambar 2: Analisis cakupan materi RPL, PBO dan PA I dalam proses pengembangan perangkat lunak

Mata kuliah PA I memberikan pengalaman kepada mahasiswa mengenai proses pengembangan perangkat lunak dari awal sampai akhir dengan menggunakan metode berorientasi objek dan pemodelan UML serta pengimplementasiannya menggunakan bahasa

pemrograman tertentu. Dari evaluasi pelaksanaan PA I, diketahui bahwa ternyata mahasiswa belum dapat memahami konsep pengembangan perangkat lunak secara komprehensif. Dari hasil laporan PA I juga dapat disimpulkan bahwa terjadi ketidaksiharasan dalam proses siklus hidup perangkat lunak, mulai dari analisis kebutuhan, perancangan, implementasi, sampai pengujian. Hal ini dikarenakan sebagian proses pengembangan perangkat lunak disampaikan pada satu mata kuliah, sementara bagian yang lain disampaikan pada mata kuliah yang lain.

Selain itu, dilihat dari waktu penyampaian mata kuliah, ketiga mata kuliah tersebut disampaikan pada waktu yang sama, yakni semester 2. Hal ini berkontribusi pada pemahaman mahasiswa yang tidak komprehensif mengenai proses pengembangan perangkat lunak. Paruh semester pertama ketika PA I menargetkan terselesaikannya analisis dan perancangan dengan menggunakan metode berorientasi objek, materi yang berkaitan di RPL baru akan disampaikan di paruh kedua semester. Karena itulah komponen analisis dan pembahasan tidak dapat tersinkronisasi dengan proses implementasi dan pengujian.

Dari sisi keterkaitan dengan mata kuliah PBO, tidak dilihat ada masalah berarti karena tahapan implementasi berorientasi objek dilakukan pada paruh kedua semester. Pada masa tersebut, mahasiswa telah mendapatkan pengetahuan yang cukup dari mata kuliah PBO untuk menjalankan pemrograman aplikasi yang telah mereka rancang pada paruh semester sebelumnya.

Oleh karena itu, susunan bahan ajar yang dibuat harus mengintegrasikan materi Rekayasa Perangkat Lunak dan Pemrograman Berorientasi Objek sesuai dengan urutan proses pengembangan perangkat lunak yang ada. Dibutuhkan juga suatu studi kasus pengembangan aplikasi sederhana yang dimasukkan sebagai contoh untuk masing-masing tahap pengembangan perangkat lunak. Dengan contoh yang beruntun tersebut diharapkan mahasiswa lebih memahami materi dengan terintegrasi.

4 Metode yang Diusulkan

Untuk mencapai pemahaman yang integral, materi disusun dengan mempertimbangkan urutan proses pengembangan perangkat lunak serta keselarasan antar mata kuliah yang saling berkaitan.

A. Susunan Materi

Komposisi materi tetap mengikuti panduan silabus yang telah ada [14], hanya saja urutan penyampaian dan pengelompokan materi disesuaikan dengan kebutuhan pada mata kuliah PA I. Susunan materi secara singkat dijabarkan sebagai berikut.

1. Pendahuluan

2. Beberapa model proses dan paradigma pengembangan perangkat lunak
3. Analisis kebutuhan
4. Pendekatan dan prinsip berorientasi objek
5. Analisis dan perancangan berorientasi objek
6. Analisis dan perancangan prosedural
7. Implementasi dari UML ke Java
8. Pengujian
9. Pemeliharaan, metrik, dan aktivitas penjaminan kualitas perangkat lunak

Penyampaian materi analisis dan perancangan prosedural seharusnya disampaikan sebelum penjelasan mengenai metode berorientasi objek. Akan tetapi, karena kebutuhan PA I untuk menyelesaikan pembuatan analisis dan perancangan berorientasi objek pada paruh pertama semester, maka analisis dan perancangan prosedural digeser ke belakang. Hal ini tidak mempengaruhi tahap pemahaman mahasiswa karena tidak ada prasyarat untuk harus memahami metode prosedural terlebih dahulu sebelum memahami metode berorientasi objek. Lagipula, pengajaran RPL di Politeknik Negeri Batam lebih menekankan pada penggunaan metode berorientasi objek karena metode tersebut lebih luas penggunaannya.

Referensi utama dari materi-materi yang disampaikan berasal dari buku [3] karena cakupan materinya yang komprehensif, dilengkapi dengan [16] untuk memberikan referensi dalam pemodelan berorientasi objek yang lebih mendetail.

B. Permasalahan, Proyek, dan Tim

Pembelajaran RPL diintegrasikan dengan mata kuliah PA I dan PBO dalam hal permasalahan yang harus dipecahkan oleh mahasiswa. Pada awal pertemuan PA I, mahasiswa dibagi menjadi kelompok beranggotakan 3 orang. Masing-masing kelompok menentukan judul aplikasi untuk dikerjakan sebagai topik PA I selama semester tersebut. Penentuan judul dilakukan secara bebas oleh mahasiswa, dosen hanya memberikan ketentuan bahwa aplikasi haruslah berbasis *desktop*, mengakses basis data, terdapat operasi *insert*, *update* dan *delete*, serta diimplementasikan menggunakan bahasa pemrograman Java.

Judul aplikasi tersebut yang akan menjadi studi kasus pada setiap topik bahasan RPL setiap minggunya. Misalnya, pada minggu ketiga dibahas mengenai Analisis Kebutuhan. Maka, mahasiswa secara berkelompok akan melakukan analisis kebutuhan terhadap judul-judul aplikasi PA I mereka. Begitu pula dengan materi-materi yang lain. Pada sebagian besar

kesempatan, mahasiswa tetap bekerja sesuai dengan kelompok mereka di PA I.

C. Sesi di Kelas

Kegiatan pembelajaran di dalam kelas tidak hanya berisi ceramah konvensional mengenai teori-teori yang mahasiswa perlu ketahui dalam RPL, tetapi lebih didominasi dengan aktivitas pemecahan masalah, penyampaian koreksi dan umpan balik dalam diskusi, serta bermain peran. Kombinasi kegiatan ini diharapkan dapat meningkatkan partisipasi mahasiswa dalam proses pembelajaran sehingga tingkat penyerapan materi menjadi lebih baik.

D. Penggunaan Kakas

Pada fase pemodelan, penggunaan kakas dalam RPL mulai dilakukan. Mahasiswa menggunakan UML untuk memodelkan aplikasi sesuai dengan judul PA I mereka. Kakas untuk membuat aneka diagram UML yang digunakan pada perkuliahan adalah StarUML dan Visual Paradigm. Microsoft Visio juga dapat digunakan untuk membuat diagram-diagram seperti ERD, perancangan basis data, antarmuka, dan lain-lain. Pada fase implementasi, mahasiswa disarankan menggunakan Netbeans sesuai dengan kakas yang digunakan pada mata kuliah PBO. Penggunaan kakas lain dimungkinkan sesuai dengan kebutuhan dan kreativitas mahasiswa.

Pembiasaan penggunaan kakas ini diharapkan meningkatkan keterampilan teknis mahasiswa dalam berbagai aktivitas dalam RPL. Penggunaan kakas juga membantu menjaga konsistensi model yang dibuat dari tahap satu ke tahap yang lain.

E. Metode Penilaian

Komponen penilaian yang diterapkan adalah sebagai berikut.

Kuis dan tugas	: 40%
Ujian Tengah Semester	: 30%
Ujian Akhir Semester	: 30%

Kuis dan tugas mendapatkan bobot terbesar dalam komponen nilai akhir. Tugas diberikan di setiap akhir pemberian materi baru untuk memperdalam pemahaman mahasiswa. Tidak semua tugas yang diberikan setiap minggu dinilai, tetapi sebagian dibahas di kelas agar mahasiswa mendapatkan umpan balik terhadap hasil pekerjaannya. Pemberian umpan balik menjadi sangat penting karena dalam mata kuliah RPL ini diharapkan mahasiswa dapat menciptakan sendiri berbagai artefak dalam berbagai tahap siklus hidup pengembangan perangkat lunak. Dengan umpan balik, mahasiswa mengetahui kesalahannya juga kesalahan yang sering dilakukan oleh orang lain. Dengan demikian diharapkan kesalahan itu tidak dilakukan kembali di masa

selanjutnya. Sementara itu, kuis direncanakan untuk dilakukan sebanyak dua kali dalam satu semester.

Pelaksanaan Ujian Tengah Semester (UTS) dan Ujian Akhir Semester (UAS) mengikuti prosedur pelaksanaan dan evaluasi proses belajar mengajar Politeknik Negeri Batam.

5 Kesimpulan dan Saran

Pada artikel ini telah diusulkan penerapan metode *problem-based learning* untuk pembelajaran mata kuliah RPL di Politeknik Negeri Batam. Pembelajaran RPL diintegrasikan dengan PA I dari sisi pengambilan topik aplikasi PA I sebagai studi kasus sepanjang pembahasan materi-materi RPL. Diskusi kelompok yang dilakukan dalam RPL pun sebagian besar dilakukan dengan anggota kelompok yang sama dengan anggota kelompok pada PA I untuk menjamin keselarasan pengerjaan proyek dari tahap ke tahap serta kesesuaiannya dengan teori yang didapat dari RPL. Untuk mendukung hal tersebut, dilakukan pula penyesuaian urutan penyampaian materi dimana metode berorientasi objek disampaikan pada paruh pertama semester, yang seharusnya disampaikan pada paruh kedua semester.

Penerapan metode pengajaran yang diusulkan ini dilakukan pada semester genap tahun ajaran 2013/2014 yang sedang berjalan. Perlu dilakukan evaluasi mengenai efektivitas dan peningkatan kualitas pemahaman mahasiswa pada akhir semester tersebut. Evaluasi perlu dilakukan tidak hanya pada mata kuliah RPL, tetapi juga mata kuliah lain yang berkaitan, yakni PA I dan juga PBO.

Daftar Pustaka

- [1] B. D. McLaughlin, G. Pollice, and D. West, *Head-First Object Oriented Analysis and Design*. Sebastopol, CA, USA: O'Reilly Media Inc., 2007.
- [2] D. Pilone and R. Miles, *Head First Software Development*. Sebastopol, CA, USA: O'Reilly Media Inc., 2008.
- [3] R. S. Pressman, *Software Engineering: A Practitioner's Approach*. New York, USA: McGraw-Hill Education, 2005.
- [4] I. Sommerville, *Software Engineering (Rekayasa Perangkat Lunak)*. Jakarta: Erlangga, 2003.
- [5] P. Graessle, H. Baumann, and P. Baumann, *UML 2.0 in Action: A project-based tutorial: A detailed and practical walk-through showing how to apply UML to real world development projects*. Birmingham, UK: Packt Publishing, 2005.
- [6] P. Roques, *UML in Practice: The Art of Modeling Software Systems Demonstrated through Worked Examples and Solutions*. England: John Wiley & Sons, Ltd, 2001.
- [7] D. Rosenberg and M. Stephens, *Use Case Driven Modeling with UML: Theory and Practice*. Berkeley, CA, USA: Apress, 2007.
- [8] A. Nugroho, *Rekayasa Perangkat Lunak Menggunakan UML dan Java*. Yogyakarta: Penerbit ANDI, 2009.
- [9] A. Nugroho, *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP*. Yogyakarta: Penerbit ANDI, 2010.
- [10] T. Hilburn and M. Towhidnejad, "A case for software engineering," *Softw. Eng. Educ. ...*, 2007.
- [11] Y. Jia, "Improving software engineering courses with case study approach," in *2010 5th International Conference on Computer Science & Education*, 2010, pp. 1633–1636.
- [12] L. Fernandez-Samaca and J. M. Ramirez, "An approach to applying Project-Based Learning in engineering courses," in *ANDESCON, 2010 IEEE*, 2010.
- [13] P. Desai and G. H. Joshi, "Activity based teaching learning in software engineering - An experience," in *2012 IEEE International Conference on Engineering Education: Innovative Practices and Future Trends (AICERA)*, 2012, pp. 1–6.
- [14] (Jurusan Teknik Informatika Politeknik Negeri Batam), "Silabus Teknik Informatika Politeknik Negeri Batam." 2011.
- [15] R. A.S. and M. Shalahuddin, *Modul Pembelajaran Rekayasa Perangkat Lunak (Terstruktur dan Berorientasi Objek)*. Bandung: Modula, 2011.
- [16] B. Oestereich, *Developing Software with UML: Object-oriented analysis and design in practice (Addison Wesley Object Technology Series)*, 2nd edition. Addison-Wesley, 2002.