

SISTEM OTOMATISASI PENGELOLAAN KULIAH PRAKTIKUM PEMROGRAMAN BERBASIS WEB

Didit Satya Bahari¹, Wijanarto²

^{1,2}Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
Jl. Nakula 1 No 5-11 Semarang 50131 (024) 3569196
E-mail : didit.sb@gmail.com¹, wijanarto.udinus@gmail.com²

Abstrak

Dengan meningkatnya popularitas World Wide Web dan Internet memiliki mempengaruhi pembelajaran dengan di bantu komputer yang kini berubah menjadi pembelajaran berbasis web. Pembelajaran berbasis web dapat terjadi di mana saja, kapan saja, melalui komputer manapun dan tanpa tentu kehadiran tutor manusia. Sistem manajemen berbasis web yang dapat memainkan peran dosen untuk mengajar dan latihan program semakin diperlukan. Pemberian tugas melalui web yang dapat dibuka dimanapun dan penilaian otomatis atau assessment terhadap tugas source code telah menjadi suatu kebutuhan penting untuk manajemen tugas-tugas, mengevaluasi dan melatih mahasiswa memprogram dengan dibantu oleh komputer. Sebuah sistem manajemen perkuliahan memungkinkan dosen untuk mengelola kelas mereka, tugas, kegiatan, kuis dan tes, referensi, dan dapat diakses melalui lingkungan online. Sistem otomatisasi pengelolaan kuliah praktikum pemrograman berbasis web yang diberi nama d2hwebmaster ini membantu mempermudah baik mahasiswa maupun dosen dengan dapat diaksesnya secara online, otomatisasi soal yang akan aktif dengan batas waktu tertentu untuk pengumpulan jawaban, otomatisasi penilaian yang digunakan untuk membantu pengajar ketika melakukan penilaian pemrograman, menyediakan feedback yang cepat kepada mahasiswa, membantu dosen dalam merancang dan menerapkan strategi pembelajaran yang efisien. Dengan diterapkannya sistem pengelolaan ini diharapkan mampu membantu dosen dalam mengatur kelas praktikumnya dan memudahkan mahasiswa mengakses tugas-tugas praktikum.

Kata Kunci: Sistem, Otomatisasi, Praktikum, Pemrograman, Berbasis Web

Abstract

With the increasing popularity of the World Wide Web and the Internet has influence on learning with computer aids that is now turned into a web-based learning . Web-based learning can happen anywhere , anytime , through any computer and without necessarily the presence of a human tutor . Web-based management system that can play the role of a lecturer to teach and exercise programs are increasingly necessary. Work assignments via the web that can be opened anywhere and automated assessment or assessment of the task source code has become a necessity for managing tasks , evaluate and train students to be assisted by a computer program . A lecture management system allows faculty to manage their classes , assignments , activities , quizzes and tests , references , and can be accessed through the online environment . Automation systems lab course management web-based programming , named d2hwebmaster helps facilitate both students and lecturers with online inaccessibility , automation questions that will be active with a certain time limit for collecting answers , automation of assessment are used to help teachers when assessing programming , providing rapid feedback to students , assist faculty in designing and implementing efficient learning strategies . With the implementation of the management system is expected to help lecturer in the class to organize and facilitate student to access the lab tasks.

Keywords: Systems, Automation, Practical, Programming, Web-Based

1. PENDAHULUAN

Pemrograman bukanlah subjek mudah bagi pemula untuk belajar[1,2,3]. Tujuan praktik kuliah pemrograman adalah belajar untuk membaca dan menulis program[4]. Tantangan lain dalam mengajar pemrograman adalah tentang penanganan tugas, yang meliputi penerimaan, penilaian dan memberikan *feedback* kepada pengiriman mahasiswa. Saat ini, siswa menyerahkan baik softcopy dan hardcopy *source code* mereka untuk penilaian, pengajar kemudian melihat *source code*, mengkompilasi dan mengeksekusi mereka. Proses ini biasanya memakan waktu.

Pembelajaran berbasis web dapat terjadi di mana saja, kapan saja, melalui komputer manapun dan tanpa tentu kehadiran tutor manusia. Sistem manajemen berbasis web yang dapat memainkan peran dosen untuk mengajar dan latihan program semakin diperlukan. Pemberian tugas melalui web yang dapat dibuka dimanapun dan penilaian otomatis[5] atau *assessment* terhadap tugas *source code* telah menjadi suatu kebutuhan penting untuk manajemen tugas, mengevaluasi dan melatih mahasiswa memprogram dengan dibantu oleh komputer.

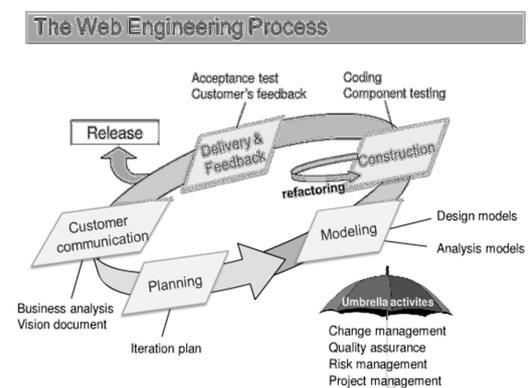
Sebuah sistem manajemen perkuliahan[6,7,8] memungkinkan dosen untuk mengelola kelas mereka, tugas, kegiatan, kuis dan tes, referensi, dan dapat diakses melalui lingkungan online. Siswa dapat *log on* dan bekerja kapan saja, di mana saja [9]. Sistem [10,11,12] manajemen perkuliahan sebagai "perangkat lunak berbasis internet yang mengelola pendaftaran siswa, melacak kinerja siswa, dan menciptakan dan mendistribusikan isi perkuliahan."

Dengan merancang sistem pengelolaan seperti yang disebutkan tadi dapat membuat perkuliahan lebih terarah dan terorganisir lebih baik serta mengurangi beban kerja dosen untuk kuliah pemrograman. Ditambah dengan diintegrasikannya kemampuan untuk menilai secara otomatis pengajar dapat langsung mendapatkan nilai dari tugas-tugas mahasiswa, dapat mengamati kemampuan belajar dan tingkat keberhasilan dari mahasiswa. Sedangkan untuk mahasiswa dapat memperoleh *feedback* yang lebih cepat dimana *feedback* salah satu hal terpenting dalam proses belajar mengajar.

Inti dari rancangan sistem yang dibuat adalah pada proses pengelolaan pada tugas *source code* yang akan diolah oleh sistem ini hingga menghasilkan nilai tugas dari mahasiswa.

2. METODE

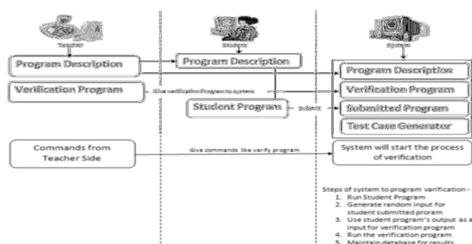
Metode yang di pakai dalam pengembangan sistem adalah *web engineering*[13], metode ini memberikan ide bagi pengembang maupun user tentang cara sistem akan berfungsi dan yang akan dikembangkan, seperti Gambar 1 di bawah ini :



Gambar 1. Metode Web Engineering

Seperti terlihat pada Gambar 1, komunikasi dalam hal ini terutama terkonsentrasi dalam 2 hal, analisa bisnis dan perumusan. Analisa bisnis akan mendefinisikan hal-hal apa saja yang akan termuat di dalam aplikasi web. Perumusan adalah pengumpulan informasi tentang hal-hal yang akan dimuat dalam web yang melibatkan semua calon pengguna. Tahap penggabungan requirement (kebutuhan) dan informasi dari user dan perencanaan teknis serta menanggapi respon (tanggapan) dari user. Pada bagian analisis meliputi, Analisis Modeling yang terdiri dari, Analisis Isi, Analisis Interaksi, Analisis Fungsional, Analisis Konfigurasi.

Sementara Design Modeling terdiri dari, Desain Antarmuka, Desain Estetika, Desain isi, Desain Arsitektur, untuk di konstruksi menjadi code yang dapat melakukan refactoring. Setelah melalui tahap pengujian, maka terakhir melakukan Web Engineering Delivery and Evaluation. Dibawah ini di tunjukan arsitektur sistem yang dibangun pada Gambar 2.



Gambar 2. Arsitektur Sistem

Terdapat terdapat 3 aktor utama yang bekerja pada sistem ini : dosen, siswa, dan sistem. Pertama, seorang dosen akan menyediakan soal pemrograman (yang disajikan secara deskriptif kepada siswa) dan inputan untuk verifikasi program. Setelah itu ketika siswa masuk kedalam sistem, siswa tersebut dapat mencoba menjawab soal yang ada.

Jawaban yang dikumpulkan oleh siswa berupa *source code* kemudian akan dinilai oleh sistem dengan cara membandingkannya dengan verifikasi program yang telah ditentukan sebelumnya oleh dosen. Informasi dari sistem, seperti kesalahan umum atau kesalahan program, jumlah input, jumlah run time akan disimpan didalam database, dimana nantinya dapat membantu dosen untuk mengevaluasi kinerja siswa dan keseluruhan kelas.

3. HASIL DAN PEMBAHASAN

3.1 Analisis

Sistem otomatisasi pengelolaan kuliah praktikum yang diberi nama *d2hwebmaster* akan digunakan untuk mendukung kuliah tatap muka khususnya kuliah praktikum pemrograman. Dari hasil komunikasi dengan pengguna didapatkan sistem kuliah praktikum pemrograman yang saat ini berjalan :

- a. Penugasan yang diberikan oleh dosen diberikan saat kuliah pemrograman sedang berlangsung sehingga bila dosen atau asisten akan memberikan tambahan tugas harus menghubungi perwakilan setiap kelas dan perwakilan mahasiswa harus menyampaikannya kepada mahasiswa yang lain.
- b. Setiap tugas diberikan batas waktu pengumpulannya dan dosen atau asisten nanti setelah batas waktu tugas telah selesai harus melakukan pengoreksian satu persatu masing-masing tugas mahasiswa.
- c. Pengumpulan tugas oleh mahasiswa dilakukan pada saat jam kuliah sedang berlangsung atau dikumpulkan dimeja dosen jika tugasnya berupa kuis.
- d. Pengumpulan tugas juga dapat dilakukan melalui *upload* ke sistem

- web yang berjalan secara lokal di laboratorium.
- e. Pengoreksian dilakukan secara manual satu persatu dengan mendatangi setiap mahasiswa dan nantinya mahasiswa akan disuruh merun program dan diberikan pertanyaan mengenai sebagian program untuk mengecek kebenaran dari program yang mereka buat.
 - f. Dari cara pengoreksian terkadang menghabiskan waktu dalam satu sesi praktikum sehingga mengurangi waktu untuk menyampaikan materi dan terkadang jika mahasiswa terlalu banyak dapat terjadi overtime untuk pengoreksian.

Selanjutnya dari sistem yang sudah berjalan sebelumnya maka yang dibutuhkan user, dalam hal ini jika dikelompokkan terdapat 3 yang memakai d2hwebmaster diantaranya dosen, mahasiswa, dan admin. Dosen memerlukan suatu sistem yang mencakup :

- a. Sistem yang dapat diakses dimana saja
- b. Sistem yang dapat menampilkan dan mnegotomatisasi tugas-tugas yang harus dikerjakan oleh mahasiswa
- c. Sistem yang dapat mengelola hasil *source code* jawaban mahasiswa dari tugas yang diberikan
- d. Sistem yang dapat menilai secara otomatis hasil *source code* yang sudah dikumpulkan dengan cara diupload oleh mahasiswa

Sedangkan bagi mahasiswa memerlukan sistem yang mencakup :

- a. Sistem yang dapat diakses dimana saja dan kapan saja
- b. Sistem yang dapat menampilkan tugas-tugas praktikum dari dosen

- c. Sistem yang dapat mengumpulkan dan mengelola hasil *source code* jawaban
- d. Sistem yang mampu memberikan *feedback* lebih cepat

3.2 Kelompok Pengguna

d2hwebmaster memiliki 3 kelompok pengguna antara lain:

1. Administrator

Memiliki hak untuk memasukkan / menanggihkan pengguna lain ke dalam sistem, untuk memodifikasi data pengguna, untuk membuat / memodifikasi kelompok, untuk mendaftarkan pengguna sebagai kelompok dosen kedalam sistem, untuk mengatur / memodifikasi *privilege* dari setiap kelompok dalam sistem, untuk melihat pengguna yang sering salah login, dan melihat error yang terjadi pada sisem d2hwebmaster.

2. Dosen

Memiliki hak untuk memberikan penugasan dengan membuat / memodifikasi soal-soal yang dibuat, melihat nilai mahasiswa hasil dari penilaian otomatis, untuk mendownload semua hasil jawaban mahasiswa, melihat profile setiap mahasiswa.

3. Mahasiswa

Memiliki hak untuk mengakses semua tugas yang diberikan dosen, untuk menyelesaikan latihan diusulkan dalam pelajaran tersebut, untuk mengirim dan menjawab soal-soal latihan, untuk mendownload jawaban yang telah diupload.

3.3 Registrasi Mahasiswa dan Dosen

Untuk memakai sistem d2hwebmaster mahasiswa dan dosen harus mendaftar terlebih dahulu. Cara pendaftaran antara mahasiswa dan dosen berbeda. Pada subbab ini akan dijelaskan cara pendaftaran mahasiswa dan dosen.

a. Cara registrasi mahasiswa
Mahasiswa mendaftar dengan masuk ke link registrasi yang sudah tersedia. Lalu mengisi form registrasi, setelah selesai akan dikirim link aktivasi ke email yang didaftarkan mahasiswa. Dengan membuka link aktivasi tersebut account mahasiswa akan menjadi aktif.

b. Cara registrasi dosen
Dosen disistem ini didaftarkan oleh administrator sehingga dosen harus menyerahkan data tertentu ke administrator. Data yang diperlukan administrator cukup email dan NIP. Jika sudah didaftarkan oleh administrator link aktivasi langsung terkirim ke email dosen yang didaftarkan. Dosen tinggal membuka link tersebut lalu dosen akan melengkapi data diri seperti nama, telepon, dll. Jika sudah account dosen akan aktif.

3.4 Tipe Penugasan

Tipe penugasan yang selama ini berjalan dilabdas terbagi menjadi 3 :

1. Tugas Mingguan

Tugas mingguan adalah tugas yang diberikan setiap minggunya oleh dosen sesuai materi yang diajarkan pada minggu tersebut.

Tugas mingguan memiliki batas waktu pengumpulan selama 1 minggu / sampai pada pertemuan selanjutnya. Nantinya pada sistem d2hwebmaster setiap penugasan memiliki batas waktu pengerjaan.

Untuk tugas yang masuk ke tipe ini nantinya memiliki batas waktu pengerjaan sampai sekitar 1 minggu. Selain dibatasi pada waktu, untuk tugas mingguan juga dibatasi pada jumlah upload source code jawaban. Dikarenakan mahasiswa dengan waktu yang panjang dapat benar-benar

memikirkan jawaban yang tepat sebelum mengumpulkan jawabannya.

2. Kuis

Tipe penugasan kuis memiliki batas waktu yang singkat untuk pengumpulan jawaban. Biasanya hanya sampai akhir matakuliah / hari itu juga. Untuk tipe penugasan ini hanya dibatasi pada waktu pengerjaan yang singkat. Untuk upload jawabannya tidak dibatasi. Tapi nanti sistem akan mencatat berapa kali mahasiswa tersebut mengupload jawaban, berapa kali jawaban yang diupload error, berapa kali jawaban yang diupload tidak error. Semua itu dicatat sebagai bahan pertimbangan dosen untuk menilai selain dari hasil penilaian otomatis.

3. Ujian

Tipe penugasan ini sama dengan kuis yang dibatasi oleh waktu pengerjaan yang singkat. Bedanya pada tingkat kesulitan soal. Pada tipe ini sistem juga akan melakukan pencatan berapa kali mahasiswa upload jawabannya.

Batas waktu disini nantinya akan diinput oleh dosen pada saat pembuatan soal. Batas waktu yang diinput ada dua: tanggal mulai soal dan tanggal akhir soal.

3.5 Status Soal

Status soal disini didasarkan pada batas waktu pengerjaan soal. Fungsi dari status soal ini untuk menginformasikan soal sudah dapat dikerjakan oleh mahasiswa atau belum. Selain itu berfungsi juga untuk melakukan pengecekan batas waktu pada saat mahasiswa upload jawaban. Status soal dibagi menjadi 4 antara lain:

1. Belum Aktif

Soal masih belum masuk waktu mulai pengerjaan soal (Tanggal sekarang <

Tanggal mulai soal). Soal yang masih belum aktif tidak akan ditampilkan ke mahasiswa. Hanya dosen pembuat soal yang dapat melihat soal dengan status ini. Ketika sudah masuk tanggal mulai soal sistem akan dengan otomatis mengupdate status soal dan menampilkan soal ke mahasiswa.

2. Aktif

Soal sudah masuk waktu mulai pengerjaan soal (Tanggal mulai soal \leq Tanggal sekarang \leq Tanggal akhir soal). Soal dengan status ini sudah ditampilkan ke mahasiswa dan mahasiswa sudah mulai dapat mengerjakannya sampai batas waktu tanggal berakhirnya soal.

3. Selalu aktif

Soal dengan status ini tidak memiliki batas waktu pengerjaan baik itu tanggal mulai maupun tanggal berakhirnya soal. Dengan tidak adanya batas waktu mahasiswa dapat mengerjakan terus soal dengan status ini tanpa dibatasi waktu. Soal dengan status ini juga tidak dibatasi pada jumlah jawaban yang boleh diupload.

4. Tidak aktif

Soal sudah melebihi batas waktu akhir pengerjaan soal (Tanggal sekarang $>$ Tanggal berakhirnya soal). Soal dengan status ini menjadikan mahasiswa sudah tidak bisa lagi mengumpulkan jawabannya. Tetapi mahasiswa masih bisa melihat soal tersebut.

Dengan pemberian status soal ini dosen dapat merancang penugasan yang nantinya berjalan selama satu semester karena soal akan ditampilkan secara otomatis oleh sistem jika sudah masuk waktu pengerjaan. Sehingga dosen yang sudah memiliki soal penugasan selama satu semester dapat langsung membuat semua soalnya disistem dan dosen tidak

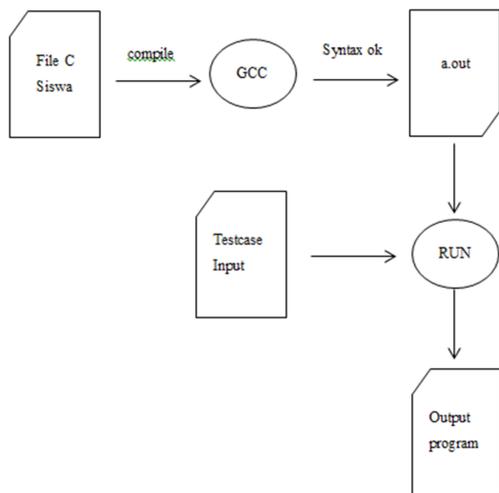
perlu memikirkan untuk membuat soal penugasan lagi setiap minggunya karena tugas sudah diotomatisasi oleh sistem berdasarkan waktunya.

3.6 Penilaian Program

Penilaian program dilakukan secara otomatis oleh sistem. Penilaian menggunakan metode *expert judgement* yaitu istilah yang merujuk secara khusus untuk teknik di mana penilaian dilakukan berdasarkan seperangkat kriteria yang spesifik dan / atau keahlian yang telah diperoleh pada area spesifik pengetahuan, atau bidang produk, disiplin tertentu (PMBOK, 2004). Peneliti memilih Wijanarto, S.Kom sebagai seorang ahli dan pengelola lab untuk menentukan cara penilaian.

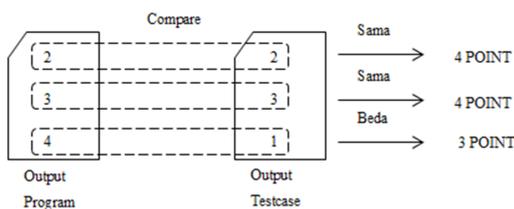
Sistem akan melakukan compile terhadap source code mahasiswa yang diupload. Jika ada error maka akan ada pemberitahuan error yang ditampilkan ke mahasiswa. Jika tidak ada error maka baru akan dilakukan verifikasi program. Verifikasi program dilakukan dengan memberikan input dan membandingkan output. Input dan output yang diberikan berasal dari file txt yang dosen upload ketika membuat soal.

Hasil dari compile source code mahasiswa tadi berbentuk file a.out yang siap diexecute. File a.out dirun dan diberikan inputan yang berasal dari pembacaan file input testcase. Output dari hasil inputan tersebut lalu disimpan dalam bentuk file output.txt yang nantinya akan dipakai penilaian. Output.txt dibaca isi filenya lalu dibandingkan dengan output testcase yang diberikan oleh dosen. Berikut gambar 3 dari penjelasan di atas,



Gambar 3. Alur Penilaian program

Penilaian akan dihitung berapa banyak kesamaan antara output program mahasiswa dengan output program dari dosen. Pada gambar 3 dapat dilihat proses penilaian. Output program mahasiswa akan dibandingkan dengan output testcase dosen. Pembacaan dilakukan perbaris. Setiap baris yang sama akan diberikan point 4, jika beda point 3. Lalu semuanya akan ditotal. Jika dicontohkan pada gambar total 11 point = 4 + 4 + 3. Perhitungan selanjutnya $(11 : 12) * 100 = 91,67$. Kenapa dibagi 12 karena jika betul semua akan didapat 12 point. Dari perhitungan tersebut didapat nilai mahasiswa 91,67.

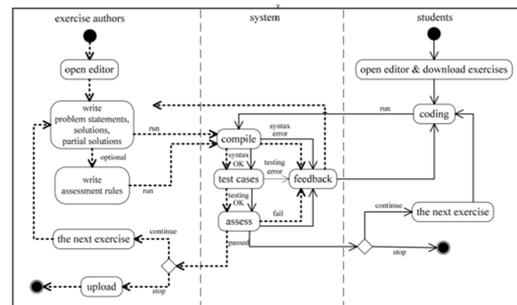


Gambar 4. Penilaian hasil jawaban

3.7 Menjawab Soal

Untuk cara menjawab, mahasiswa dapat mengupload source code jawaban yang sudah dia buat yang lalu disimpan di server untuk dilakukan penilaian. Atau

juga mahasiswa dapat menggunakan text editor sederhana yang sudah ada disistem sehingga nanti sistem akan menyimpan source code yang diketik di editor ke dalam file C. Gambar 5 menjelaskan lebih detail alur pengiriman jawaban oleh siswa.



Gambar 5. Alur pengiriman jawaban

3.8 Pembahasan Hasil

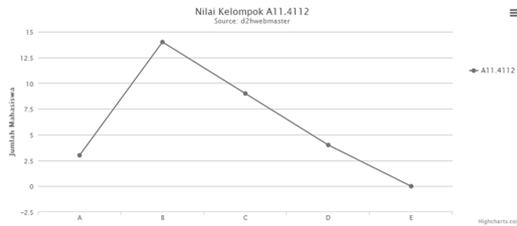
Hasil implementasi sistem di lakukan di laboratorium dasar Teknik Informatika, Fakultas Ilmu Komputer, terhadap 3 kelompok kelas yang mengikuti praktiku pemrograman, A11.4112, A11.4114 dan A11.4115. berikut gambar 5 screen shoot hasil yang di capai oleh mahasiswa yang telah mengupload jawaban dan telah di nilai otomatis oleh sistem autograder.

Grade	NIM	Nama	Nilai Angka	Nilai Huruf	Date Upload
1	A11.2012.07006	AFRIZAL ADAM MAULANA	100	A	2013-09-23 22:43:06
2	A11.2012.07027	BUDI JAELENI	100	A	2013-09-23 23:00:12
3	A11.2012.07031	ILHAM BANGUNI	100	A	2013-09-23 23:02:00
4	A11.2012.06803	AGIL AKTOR RISANI	80	B	2013-09-23 22:19:45
5	A11.2012.07001	RINTAN FERNAIDA HARYONO	80	B	2013-09-23 22:27:26
6	A11.2012.07033	ANDRI PRYADI	80	B	2013-09-23 23:03:22
7	A11.2012.07000	YULIARDI PRATOMO HABBY	75	B	2013-09-23 22:25:14
8	A11.2012.07002	DEVI YULYANTI	75	B	2013-09-23 22:31:10
9	A11.2012.07003	ALIF YUSUF FITRIYANTO	75	B	2013-09-23 22:37:24
10	A11.2012.07004	FARISAL ADI SAPUTRA	75	B	2013-09-23 22:38:47
11	A11.2012.07009	DWI PRASETYO	75	B	2013-09-23 22:46:25
12	A11.2012.07014	KANZHA ROBI ARSYADANI	75	B	2013-09-23 22:50:37
13	A11.2012.07019	ODRI GANTAR UFTIANKA	75	B	2013-09-23 22:51:33
14	A11.2012.07021	RIDHO HZRI ELRYANTO	75	B	2013-09-23 22:54:00
15	A11.2012.07023	DANA MELINA AGUSTRIA	75	B	2013-09-23 22:55:51

Gambar 6. Hasil penilaian autograder

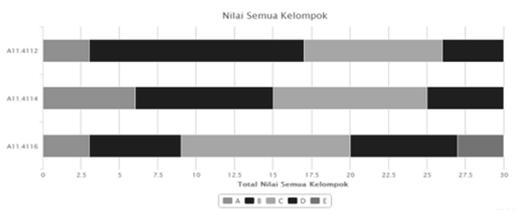
Mahasiswa juga dapat melihat dia ada peringkat berapa dalam satu kelas untuk jawaban pada soal tersebut. Seperti pada gambar 6 mahasiswa dengan NIM A11.2012.06803 berada pada peringkat 4 untuk kelas A11.4112 dan soal PDP-02. Peringkat ini didapat dengan

mengurutkan nilai terbesar ke yang terkecil, lalu diurutkan juga berdasarkan waktu menjawab dari yang lebih dulu ke yang terakhir, seperti terlihat dari gambar 7 berikut.



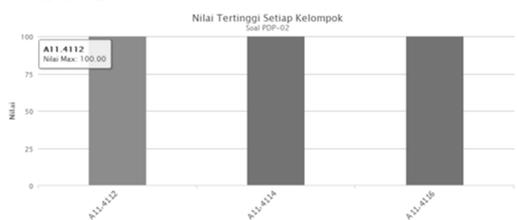
Gambar 7. Hasil ranking nilai mahasiswa

Dosen juga akan mendapatkan laporan tabel peringkat dan juga grafik yang menggambarkan nilai antar kelas. Seperti pada gambar 7 akan diketahui grafik dari kelompok A11.4112. Grafik ini menggambarkan jumlah mahasiswa yang mendapat nilai A, B, C, D dan E pada gambar 8 berikut,

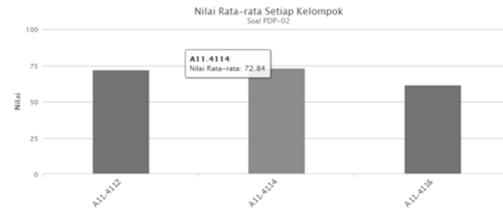


Gambar 8. Report Nilai ke dosen

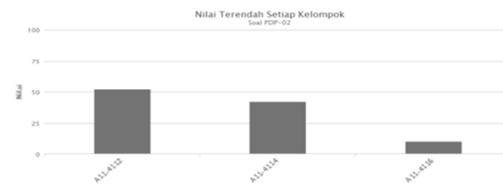
Berturut-turut pada gambar 9,10 dan 11 ditampilkan grafik nilai maksimal, rata-rata dan minimal dari semua kelompok berdasarkan penugasan. untuk dapat di evaluasi.



Gambar 9. Nilai tertinggi berdasarkan penugasan



Gambar 10. Nilai rata-rata berdasarkan penugasan



Gambar 11. Nilai terendah berdasarkan penugasan

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Dari hasil analisa dan pembahasan di atas maka dapat di simpulkn sementara, bahwa, Sistem pengelolaan kuliah praktikum pemrograman membantu mempermudah baik mahasiswa maupun dosen dengan dapat diaksesnya sistem secara online sehingga dapat digunakan darimana saja dan kapan saja. Otomatisasi pada soal yang dibuat dosen berjalan sesuai dengan tujuan. Soal akan aktif dengan batas waktu tertentu untuk pengumpulan jawaban. Otomatisasi penilaian juga berjalan sesuai dengan perhitungan yang benar. Sistem tidak hanya membantu mahasiswa dengan menyediakan *feedback* yang cepat mengenai jawabannya tapi juga membantu dosen dalam merancang dan menerapkan strategi pembelajaran yang efisien. Dosen juga dapat memonitor status belajar mahasiswanya.

4.2 Saran

Berdasarkan simpulan yang ada dan beberapa kekurangan yang di temukan dalam penelitian ini,, maka dapat disarankan, Otomatisasi pada penilaian

dapat dikembangkan lagi agar bisa juga mendeteksi output program untuk tipe data tertentu, membuat batasan *source code* harus menggunakan *looping* atau rekursif dan ditambah untuk mendeteksi plagiarisme. Dapat ditambah untuk dapat digunakan oleh guest yang tanpa perlu mendaftar dapat langsung mengerjakan semua soal yang dibuat oleh semua dosen. Dapat juga dikembangkan untuk tidak hanya melakukan penilaian terhadap *source code* bahasa C tapi juga bahasa pemrograman lainnya.

DAFTAR PUSTAKA

- [1] Kelleher, C. and R. Pausch. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys*, vol. 37 no. 2, June 2005, pages 83-137
- [2] Lahtinen, Essi. (2005). A Study of the Difficulties of Novice Programmers. Monte de Caparica, Portugal.
- [3] Choy, Sheung-On dan Sin-Chun Ng. (2004). An interactive learning environment for teaching and learning of computer programming. *Proceedings of the IEEE International Conference on Advanced Learning Technologies*.
- [4] Helminen, Juha. (2009). Quick Introduction to Programming with an Integrated Code Editor, Automatic Assessment and Visual Debugging Tool – Work in Progress. 9th Koli Calling International Conference on Computing Education Research.
- [5] Parasuraman, Raja. (2000). A Model for Types and Levels of Human Interaction with Automation. *IEEE*.
- [6] Morgan, Glenda. (2003). Faculty Use of Course Management Systems. *EDUCAUSE Center for Applied Research*.
- [7] Moura, Janine G. (2007). A web-based learning management system with automatic assessment resources. *IEEE*.
- [8] Butz, C.J. (2005). A Web-based Intelligent Tutoring System for Computer Programming. Department of Computer Science University of Regina.
- [9] Ullman, C. & Rabinowitz, M. (2004). Course management systems and the reinvention of instruction, *T.H.E. Journal*.
- [10] Indrajit. (2001). Analisis dan Perancangan Sistem Berorientasi Object. Bandung, Informatika.
- [11] Jogiyanto HM. (2005). Analisis & Desain Sistem Informasi : Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis. Yogyakarta. Andi.
- [12] Robert G Murdick, dkk. (1991). Sistem Informasi Untuk Manajemen Modern, Jakarta Erlangga.
- [13] Roger Pressman, David Lowe, 2008, *WEB ENGINEERING : A PRACTITIONER'S APPROACH*, McGraw-Hill Education, US America.