

Pengembangan Sistem Pengelolaan Nilai Kurikulum 2013 Berbasis Web di Madrasah Aliyah Nurul Mujtahidin Ponorogo

Siti Nur Wahidah Abroriyah¹, Tri Astoto Kurniawan²

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹snw.abroriyah@gmail.com, ²triak@ub.ac.id

Abstrak

Madrasah Aliyah Nurul Mujtahidin adalah salah satu lembaga pendidikan tingkat menengah atas yang berada di bawah pengawasan Kementerian Agama Republik Indonesia. Sejak 2016 Madrasah Aliyah Nurul Mujtahidin telah menerapkan Kurikulum 2013 sebagai kurikulum dasar dalam kegiatan pembelajarannya. Penerapan Kurikulum 2013 memengaruhi seluruh aktivitas dalam proses pembelajaran, salah satunya adalah pengelolaan nilai. Guru wali kelas menggunakan *Microsoft Excel* untuk mengumpulkan nilai dari guru pengampu mata pelajaran. Dokumen yang berisi nilai yang sering berpindah dari satu orang ke orang lain menghasilkan data yang rawan kesalahan. Selain itu, dokumen terpisah berisi nilai mata pelajaran yang berbeda juga menyulitkan wali kelas dalam melakukan rekapitulasi nilai akhir peserta didik. Sistem pengelolaan nilai Kurikulum 2013 berbasis web dikembangkan untuk mengatasi masalah ini. Pengembangan sistem ini dilakukan dengan menerapkan model pengembangan perangkat lunak *waterfall* dan menggunakan pendekatan berorientasi objek. Sistem ini telah diuji dengan menggunakan pengujian unit, pengujian integrasi, dan pengujian validasi. Pengujian *white-box* digunakan dalam pengujian unit dan pengujian integrasi sedangkan pengujian validasi dilakukan dengan menggunakan pengujian *black-box*. Pengujian unit dan integrasi menunjukkan hasil valid untuk 3 sampel unit dan 3 sampel integrasi. Hasil pengujian validasi menunjukkan sistem 100% valid untuk 52 kebutuhan fungsional.

Kata kunci: *pengelolaan nilai, Kurikulum 2013, pengembangan perangkat lunak, model waterfall, pendekatan berorientasi objek*

Abstract

Madrasah Aliyah Nurul Mujtahidin is one of a senior education institutions lie under the supervision of Ministry of Religion of the Republic of Indonesia. Since 2016 Madrasah Aliyah Nurul Mujtahidin has been applying Kurikulum 2013 as the basic curricula in its learning processes. The application of Kurikulum 2013 influences all learning activities, one of which is the scoring management. Homeroom teachers use Microsoft Excel to collect scores from subject teachers. Documents containing scores which are transferred from one person to another frequently result in error-prone data. In addition, a heaps of separate documents containing different subject scores also results in difficulties in recapitulating students' final scores. A web-based Kurikulum 2013 scoring management system was developed to overcome these problems. The development of this system was done by applying waterfall software development model and used object oriented approach. This system has been tested by using unit, integration, and validation testing. White-box testing was used in unit and integration testing whereas validation testing was done by using black-box testing. Unit and integration testing give forth valid results for 3 unit samples and 3 integration samples. The result of validation testing shows that system is 100% valid for 52 functional requirements.

Keywords: *scoring management, Kurikulum 2013, software development, waterfall model, object oriented approach*

1. PENDAHULUAN

Kurikulum ialah sebuah rencana yang digunakan sebagai pedoman atau pegangan

selama aktivitas belajar-mengajar (Sukmadinata, 2013). Kurikulum 2013 adalah kurikulum pendidikan yang saat ini diterapkan dalam dunia pendidikan formal tingkat dasar hingga

menengah atas di Indonesia (Kementerian Pendidikan dan Kebudayaan, 2012). Kurikulum ini sudah secara luas diterapkan baik itu di sekolah-sekolah dibawah naungan Kementerian Pendidikan dan Kebudayaan (Kemendikbud) maupun madrasah-madrasah dibawah naungan Kementerian Agama (Kemenag) (Keputusan Menteri Agama, 2014).

Madrasah Aliyah (MA) Nurul Mujtahidin Ponorogo adalah salah satu madrasah dibawah naungan Kemenag yang menerapkan Kurikulum 2013. Penerapan Kurikulum 2013 di madrasah ini sudah berlangsung selama 3 tahun, yaitu sejak tahun ajaran 2016/2017 hingga 2018/2019 yang mempengaruhi proses pengelolaan nilai di madrasah.

Pengelolaan dan pelaporan nilai di MA Nurul Mujtahidin dilakukan dengan menggunakan aplikasi *Microsoft Excel*. Wali kelas sebagai pihak yang bertugas menyusun dokumen rapor harus meminta data nilai dari guru mata pelajaran dan guru bimbingan dan konseling (BK) sebagai pihak yang menilai perkembangan peserta didik. Wali kelas membuat *file* data nilai siswa (DNS) yang berisi daftar peserta didik dalam kelas. *File* DNS yang sudah diisi nilai oleh guru mata pelajaran kemudian dikembalikan ke wali kelas melalui media *flash disk* atau *compact disk* (CD).

Proses pengelolaan nilai ini masih memiliki banyak masalah. Perpindahan *file* DNS tanpa adanya pembatasan hak akses mengakibatkan pihak selain guru mata pelajaran yang bersangkutan dapat melakukan perubahan pada *file* DNS. Hal tersebut tentunya berpengaruh terhadap data nilai yang dihasilkan karena data nilai menjadi rentan kesalahan dan memiliki tingkat integritas data yang rendah .

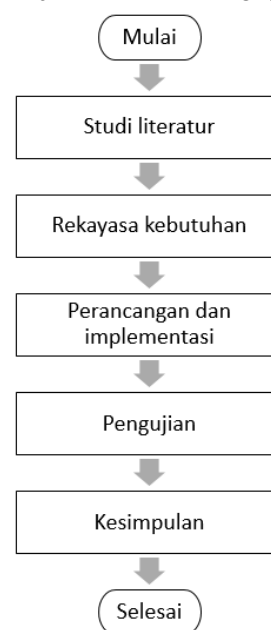
Masalah lain yang timbul adalah ketika guru mata pelajaran melakukan kesalahan ketika memasukkan nilai. Kesalahan bisa berupa format yang tidak sesuai dengan permintaan wali kelas, nilai yang tidak lengkap, atau keterlambatan dalam mengumpulkan *file* DNS kepada wali kelas yang menyulitkan wali kelas dalam melakukan rekapitulasi nilai dan menimbulkan keterlambatan penyusunan rapor.

MA Nurul Mujtahidin memerlukan mekanisme untuk mengatasi permasalahan tersebut yaitu berupa sistem pengelolaan nilai dan penyusunan rapor berdasarkan Kurikulum 2013. Sistem pengelolaan nilai Kurikulum 2013 dibuat untuk mengelola nilai dan menyusun rapor sehingga lebih mudah bagi guru untuk melakukan evaluasi hasil belajar peserta didik

(Merindasari, et al., 2015). Sistem dikembangkan dengan menerapkan model pengembangan perangkat lunak *waterfall* dan pendekatan berorientasi objek. Model *waterfall* dipilih karena kebutuhan perangkat lunak telah dapat ditemukan dan didefinisikan pada tahap awal pengembangan sistem (Pressman & Maxim, 2015). Pendekatan berorientasi objek dipilih karena model berorientasi objek dibuat berdasarkan objek-objek *real-world* sehingga lebih mudah untuk dipahami (Burleson, 2008).

2. METODOLOGI PENELITIAN

Penelitian ini dilaksanakan dalam beberapa langkah. Penelitian dimulai dengan aktivitas studi literatur yang diikuti dengan rekayasa kebutuhan. Aktivitas selanjutnya adalah perancangan dan implementasi. Pengujian dilakukan terhadap artefak yang dihasilkan yang dilanjutkan dengan penarikan kesimpulan. Gambar 1 menunjukkan metodologi penelitian.



Gambar 1. Metodologi penelitian

Studi literatur dilakukan untuk mendalami pemahaman mengenai masalah. Studi literatur juga dilakukan untuk memperoleh teori-teori yang digunakan untuk menemukan solusi permasalahan.

Rekayasa kebutuhan berisi identifikasi sistem guna memperoleh spesifikasi perangkat lunak. Aktivitas rekayasa kebutuhan terdiri dari 2 tahap yaitu elisitasi dan analisis kebutuhan serta spesifikasi kebutuhan (Marsic, 2012).

Perancangan dilaksanakan dengan menerapkan konsep *object oriented design* (OOD) dan menggunakan *Unified Modelling*

Language (UML) untuk memodelkan perangkat lunak. Perancangan perangkat lunak berisi proses representasi perangkat lunak menjadi bentuk rancangan-rancangan (Sommerville, 2011). Perancangan terdiri dari 4 jenis yaitu perancangan arsitektur (*architectural design*), perancangan komponen (*component design*), perancangan data (*data design*), dan perancangan antarmuka (*interface design*) (Pressman & Maxim, 2015). Implementasi kode program dibuat berdasarkan hasil rancangan arsitektur dan komponen menggunakan *framework* CodeIgniter pada bahasa pemrograman PHP. Implementasi basis data dilakukan menggunakan MySQL dalam MariaDB berdasarkan rancangan data. Implementasi antarmuka pengguna dilakukan dengan menggunakan HTML, Javascript, dan Bootstrap.

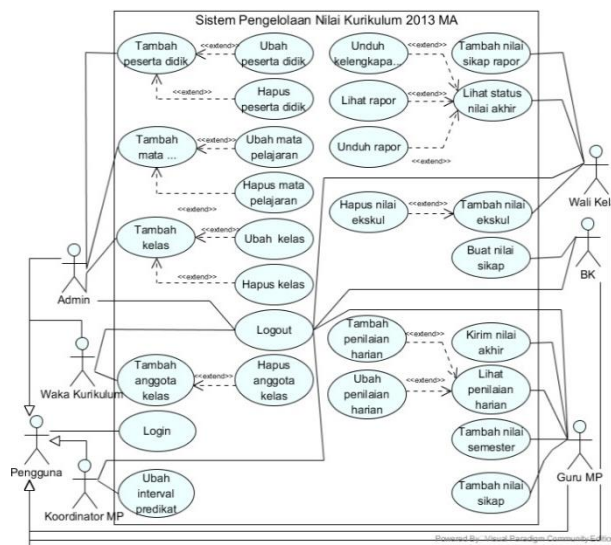
Pengujian perangkat lunak adalah serangkaian tindakan yang dirancang untuk menemukan kesalahan perangkat lunak (Myers, 2004). Pengujian dilakukan untuk menemukan situasi-situasi dimana perangkat lunak berjalan dengan tidak benar, tidak diinginkan atau tidak sesuai dengan spesifikasinya (Sommerville, 2011). Pengujian dilakukan pada 3 level yaitu pengujian unit (*unit testing*), pengujian integrasi (*integration testing*) dan pengujian sistem (*system testing*) (Marsic, 2012). Pengujian *white-box* digunakan dalam pengujian unit dan integrasi sedangkan pengujian validasi dilakukan dengan menggunakan pengujian *black-box*. Pengujian sistem dilakukan dengan menggunakan pengujian validasi.

Kesimpulan ditarik dari hasil pengujian perangkat lunak yang telah dikembangkan sebagai hasil akhir dari keseluruhan proses penelitian. Kesimpulan ini digunakan sebagai saran masukan dan bahan pertimbangan bagi penelitian selanjutnya.

3. REKAYASA KEBUTUHAN

Rekayasa kebutuhan perangkat lunak dalam penelitian ini dilakukan dengan beberapa cara. Cara pertama adalah dengan melakukan pengamatan dokumen Panduan Penilaian Kurikulum 2013 yang diterbitkan oleh Kemendikbud untuk mengetahui proses pengelolaan nilai yang benar (Pendidik dan Satuan Pendidikan Sekolah Menengah Atas, 2017). Cara kedua adalah melalui wawancara dengan Bapak Mujiono, S.Pd selaku Kepala Madrasah Aliyah Nurul Mujtahidin Ponorogo.

Wawancara dilakukan untuk mengetahui proses pengelolaan nilai dan penyusunan rapor Kurikulum 2013 yang saat ini diterapkan di madrasah. Hasil wawancara menyatakan bahwa pihak madrasah menggunakan aplikasi Pengolah Nilai Kurikulum 2013 berbasis *Microsoft Excel* yang dikembangkan oleh Kemenag Jawa Timur. Aplikasi ini sudah digunakan sejak semester genap tahun ajaran 2016/2017. Penggalan kebutuhan dilakukan dengan melakukan pengamatan terhadap aplikasi tersebut.

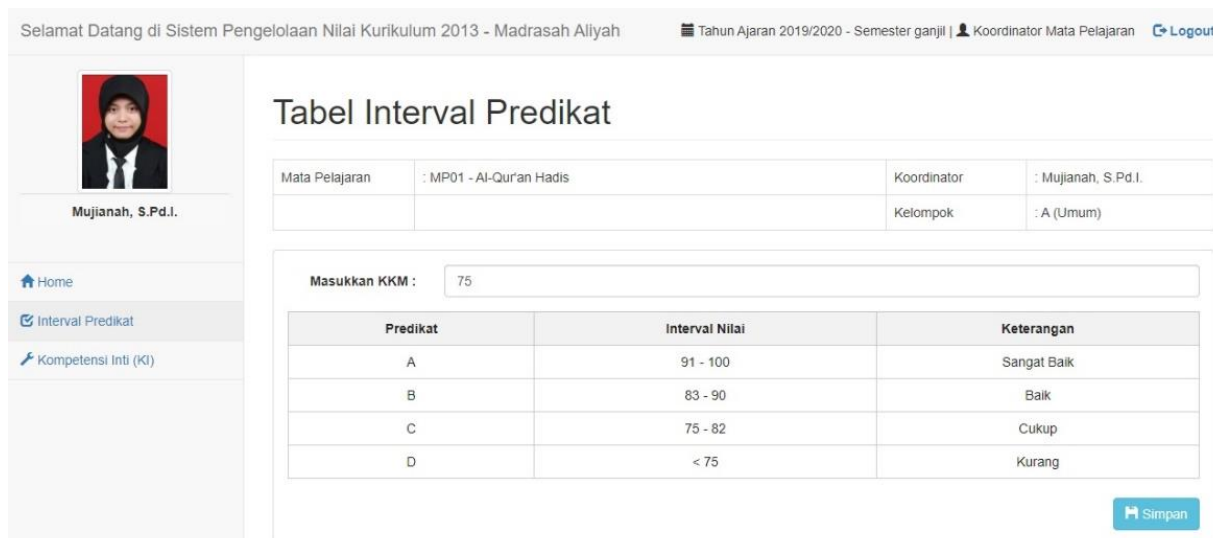


Gambar 2. Use case diagram (parsial)

Elisitasi dan analisis kebutuhan yang telah dilakukan menghasilkan 52 definisi kebutuhan fungsional perangkat lunak. Sejumlah 7 aktor diperoleh dalam proses identifikasi aktor yang meliputi Pengguna, Admin, Waka kurikulum, Koordinator MP, BK, Guru MP, dan Wali kelas. Spesifikasi untuk setiap kebutuhan fungsional dirumuskan sehingga diperoleh daftar kebutuhan fungsional yang rinci. Kebutuhan-kebutuhan tersebut kemudian dimodelkan menggunakan *usecase diagram* dan *use case scenario*. *Use case diagram* adalah sebuah model yang secara grafis menggambarkan perangkat lunak sebagai koleksi dari satu atau lebih *use case*, aktor, dan hubungan diantara keduanya (Whitten & Bentley, 2007). *Use case scenario* menjelaskan mengenai bagaimana aktor berinteraksi dengan perangkat lunak ketika *use case* tersebut sedang berjalan (Whitten & Bentley, 2007). Gambar 2 menunjukkan bagian dari *use case diagram*.

4. PERANCANGAN DAN IMPLEMENTASI

Aktivitas perancangan perangkat lunak



Gambar 5. Implementasi antarmuka halaman interval predikat mata pelajaran

yang saling berkomunikasi dalam sistem. Pembuatan model *class diagram* dilakukan untuk menemukan daftar klas dan relasi yang muncul sebagai komponen penyusun sistem. Gambar 3 menunjukkan gambaran *class diagram*. Perangkat lunak ini dikembangkan menggunakan *framework* CodeIgniter sehingga pemodelan *class diagram* dilakukan dengan mengacu pada arsitektur *framework*. *Framework* CodeIgniter menggunakan konsep klas hanya pada komponen *controller* (kontrol) dan *model/entity* (model/entitas). Sementara itu *view* direpresentasikan menggunakan HTML sebagai halaman web.

Perangkat lunak tersusun atas 27 klas-klas entitas yang saling berhubungan satu sama lain. Klas *CI_Model* adalah klas umum yang menurunkan klas-klas entitas lain. Perangkat lunak ini juga tersusun atas 24 klas-klas kontrol yang merupakan turunan dari klas *CI_Controller*.

Perancangan komponen dilakukan dengan membuat rancangan algoritme proses dalam setiap klas yang disajikan dalam bentuk *pseudocode*. Perancangan data dilakukan pada level konsep menggunakan *Conceptual Data Model* (CDM) yang kemudian dipetakan menjadi *Physical Data Model* (PDM). Gambar 4 menunjukkan rancangan CDM sistem. Rancangan antarmuka yang dibuat adalah rancangan antarmuka pengguna dalam bentuk *Graphical User Interface* (GUI).

Rancangan-rancangan yang dihasilkan kemudian digunakan sebagai dasar dalam implementasi. *Framework* CodeIgniter yang menggunakan bahasa pemrograman PHP digunakan dalam implementasi kode program.

Implementasi kode program dilakukan dengan mengacu pada hasil rancangan arsitektur dan komponen. Implementasi basis data dilakukan menggunakan MySQL dalam MariaDB yang mengacu pada rancangan PDM. Implementasi antarmuka pengguna dilakukan dengan menggunakan HTML, JavaScript, dan CSS dalam Bootstrap. Gambar 5 menunjukkan hasil implementasi antarmuka halaman interval predikat.

5. PENGUJIAN

Pengujian pada penelitian ini dilakukan dengan strategi pengujian unit, integrasi dan validasi. Pengujian unit dan integrasi dilakukan dengan menerapkan pengujian *white-box* sedangkan pengujian validasi dilakukan dengan menerapkan pengujian *black-box*.

Pengujian unit dilakukan pada klas sebagai unit terkecil dari sistem. Pengujian unit dilakukan dengan menggunakan *basis path testing*. Pengujian unit dilakukan pada 3 buah sampel yaitu *method* *set_interval_predikat()* dari klas *Mata_pelajaran*, *method* *add_nilai_harian()* pada klas *Kompetensi_inti*, dan *method* *add_ekskul_diikuti()* pada klas *Peserta_didik*.

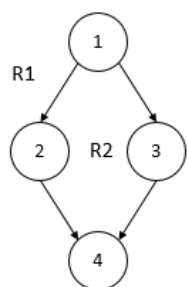
Tabel 1 menunjukkan *pseudocode* untuk *method* *set_interval_predikat()* pada klas *Mata_pelajaran*. Flow graph untuk *pseudocode* tersebut dapat dilihat pada Gambar 6. Tabel 2 menunjukkan hasil pengujian *method* *set_interval_predikat()* pada klas *Mata_pelajaran*.

Tabel 1. Pseudocode method set_interval_predikat() pada kelas Mata_pelajaran

```

Start
attribut interval_predikat = param1
data = array('mata_pelajaran' = nilai atribut
             kode,
             'tahun_ajaran' = param2,
             'batas_nilai_a' = batas_nilai_a
             dari objek param1,
             'batas_nilai_b' = batas_nilai_b
             dari objek param1,
             'kkm' = kkm dari objek param1)
query replace dengan data = data
if (replace success)
    return true
endif

return false
End
    
```



Gambar 6. Flow graph method set_interval_predikat() pada kelas Mata_pelajaran

Cyclomatic complexity (V(G))

$$V(G) = R = 2$$

$$V(G) = E - N + 2 = 4 - 4 + 2 = 2$$

$$V(G) = P + 1 = 1 + 1 = 2$$

Independent path

(i) Jalur 1 : 1 → 2 → 4

(ii) Jalur 2 : 1 → 3 → 4

Pengujian integrasi dilakukan pada kelas-kelas yang saling berinteraksi. Pengujian integrasi dilakukan dengan menerapkan *basis path testing* pada 3 buah sampel yaitu *method* *simpan_interval_predikat()* pada kelas *Interval_predikat_kontrol()* dengan hasil valid, *method* *buat_penilaian_harian()* pada kelas *Penilaian_harian_kontrol* dengan hasil valid, dan *method* *tambah_nilai_ekskul()* pada kelas *Nilai_ekskul_kontrol* dengan hasil valid.

Tabel 2. Hasil pengujian unit *method* *set_interval_predikat()* pada kelas Mata_pelajaran

Tahap	Prosedur Uji	Hasil yang diharapkan	Hasil yang didapatkan	Status
1	Klas driver menjalankan <i>method</i> <i>set_interval_predikat()</i> dengan nilai : mata_pelajaran = MP01 tahun_ajaran = 2019/2020 batas_nilai_a = 80 batas_nilai_b = 76 kkm = 72 replace = TRUE	Meng embal ikan nilai TRU E	Meng embal ikan nilai TRU E	Valid
2	Klas driver menjalankan <i>method</i> <i>set_interval_predikat()</i> dengan nilai : mata_pelajaran = MP01 tahun_ajaran = 2019/2020 batas_nilai_a = 80 batas_nilai_b = 76 kkm = 72 replace = FALSE	Meng embal ikan nilai FALS E	Meng embal ikan nilai FALS E	Valid

Pengujian validasi dilakukan pada 52 kebutuhan yang sudah didapatkan. Pengujian validasi dilaksanakan dengan memanfaatkan teknik *scenario based testing*. Hasil pengujian validasi sistem menunjukkan bahwa sistem valid untuk 52 kebutuhan.

6. KESIMPULAN DAN SARAN

Tahap rekayasa kebutuhan menghasilkan 7 aktor dan 52 kebutuhan fungsional. Tujuh aktor yang berinteraksi dengan sistem adalah Pengguna, Admin, Waka kurikulum (wakil kepala bidang kurikulum), Koordinator MP (koordinator mata pelajaran), BK (bimbingan dan konseling), Guru MP (guru pengampu mata pelajaran), dan Wali kelas. Beberapa kebutuhan fungsional utama sistem antara lain ubah interval predikat, tambah penilaian harian, tambah nilai sikap, dan tambah nilai ekskul.

Tahap perancangan menghasilkan beberapa model perangkat lunak. Hasil dari perancangan arsitektur berupa *sequence diagram* dan *class diagram*. *Class diagram* menunjukkan bahwa sistem terdiri dari 27 kelas entitas dan 24 kelas kontrol. Hasil dari perancangan komponen berupa rancangan *algoritme* dalam bentuk *pseudocode*. Hasil dari perancangan data berupa *conceptual data model* (CDM) dan *physical data model* (PDM). Hasil

dari perancangan antarmuka berupa rancangan antarmuka pengguna.

Pengujian dilakukan melalui pengujian unit, pengujian integrasi serta pengujian validasi. Pengujian *white-box* digunakan dalam melakukan menguji unit dan integrasi sedangkan pengujian validasi dilakukan dengan menerapkan pengujian *black-box*. Pengujian unit dilaksanakan pada 3 buah sampel yang menunjukkan hasil valid. Pengujian integrasi dilaksanakan dengan menguji buah sampel yang menunjukkan hasil valid. Pengujian validasi dilakukan untuk 52 kebutuhan dengan hasil valid untuk semua kebutuhan.

Saran yang diberikan untuk penelitian selanjutnya adalah mengembangkan sistem secara lebih lanjut dengan menambah fungsi pemberian notifikasi dan *messaging* untuk mempermudah komunikasi antar pengguna dalam sistem.

7. DAFTAR REFERENSI

- Burleson, D., 2008. *Advantages and Disadvantages of Object Oriented Approach*. [Online] Tersedia di: www.dba-oracle.com/t_object_oriented_approach.htm [Diakses 20 May 2019].
- Fowler, M. & Scott, K., 2000. *UML Distilled, A Brief Guide to the Standard Object Modelling Language*. 2nd ed. Canada: Addison Wesley.
- Kementerian Pendidikan dan Kebudayaan, 2012. *Dokumen Kurikulum 2013*. Jakarta: Kementerian Pendidikan dan Kebudayaan.
- Keputusan Menteri Agama, 2014. *Keputusan Menteri Agama Republik Indonesia Nomor 207 Tahun 2014 Tentang Kurikulum Madrasah*. Jakarta: Kementerian Agama Republik Indonesia.
- Larman, C., 2004. *Applying UML and Patterns : An Introduction to Object-Oriented Analysis and Design and Iterative Development*. 3rd ed. London: Addison Wesley Professional..
- Marsic, I., 2012. *Software Engineering*. New Jersey: Rutgers University.
- Merindasari, E., Widyaningtyas, T. & Arifin, M. Z., 2015. *Sistem Informasi Penilaian Akademik Siswa Kurikulum 2013 Berbasis Web Di SMAN 1 Trenggalek*, Malang: Universitas Negeri Malang.
- Myers, G. J., 2004. *The Art of Software Testing*. 2 ed. New Jersey: John Wiley & Sons, Inc.
- Pendidik dan Satuan Pendidikan Sekolah Menengah Atas, 2017. *Panduan Penilaian*. Jakarta: Direktorat Pembinaan Sekolah Menengah Atas, Direktorat Jenderal Pendidikan Dasar dan Menengah, Kementerian Pendidikan dan Kebudayaan.
- Pressman, R. S. & Maxim, B. R., 2015. *Software engineering : a practitioner's approach*. 8 ed. New York: McGraw-Hill.
- Sommerville, I., 2011. *Software Engineering*. 9 ed. London: Addison-Wesley.
- Sukmadinata, N. S., 2013. *Pengembangan Kurikulum Teori dan Praktik*. Bandung: Remaja Rosdakarya.
- Whitten, J. L. & Bentley, L. D., 2007. *System Analysis and Design Method*. 7 ed. New York: McGraw-Hill/Irwin.