

## SISTEM INFORMASI *LOGBOOK* PADA LABORATORIUM SISTEM INFORMASI

Siti Mukaromah<sup>1)</sup>, Agung Brastama Putra<sup>2)</sup>

E-mail : <sup>1)</sup>[sitimukaromah.si@upnjatim.ac.id](mailto:sitimukaromah.si@upnjatim.ac.id) , <sup>2)</sup>[agungbp.si@upnjatim.ac.id](mailto:agungbp.si@upnjatim.ac.id)

<sup>1,2</sup>Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional “Veteran” Jawa Timur

### Abstrak

Kegiatan yang dilaksanakan di laboratorium program studi sistem informasi membutuhkan pencatatan kegiatan secara sistematis. Hal ini dikarenakan banyak agenda kegiatan yang dilakukan di laboratorium namun belum terdapat pencatatan aktifitasnya, sehingga jika saat akreditasi membutuhkan agenda kegiatan laboratorium masih belum dapat dengan cepat menunjukkan historinya. Permasalahan tersebut diupayakan untuk ditangani dengan dibuatkan sebuah sistem informasi logbook laboratorium agar pencatatan aktifitas laboratorium dapat tercatat dan tersimpan secara sistematis. Untuk membangun sistem informasi *logbook* laboratorium menggunakan metode SDLC, mulai dari perancangan basis data, perancangan sistem dengan iconix, sampai dengan pembangunan sistem menggunakan Visual Studio .NET 2005. Hasil dari penelitian ini adalah sebuah sistem informasi logbook yang diterapkan di Laboratorium Program Studi Sistem Informasi Fakultas Ilmu Komputer Universitas Pembangunan Nasional “Veteran” Jawa Timur.

**Kata kunci:** *Iconix, Logbook, Laboratorium, UML, SDLC.*

### 1. PENDAHULUAN

Dalam sebuah perguruan tinggi atau universitas, sistem pembelajaran membutuhkan praktik langsung. Khususnya program studi (prodi) sistem informasi fakultas ilmu komputer Universitas Pembangunan Nasional “Veteran” Jawa Timur, dalam pelaksanaan Kegiatan Belajar Mengajar (KBM) dibutuhkan praktik langsung menggunakan komputer.

Dalam pelaksanaan kegiatan yang dilaksanakan di laboratorium prodi sistem informasi terdapat kendala yaitu bagaimana mengelola kegiatan laboratorium agar bisa tercatat dan sistematis. Penggunaan laboratorium agar tidak berbenturan dengan kegiatan lain. Untuk itu dibutuhkan sistem informasi yang mencatat semua kegiatan yang dilakukan di laboratorium prodi Sistem Informasi Universitas Pembangunan Nasional “Veteran” Jawa Timur serta mencatat agenda kegiatan yang tidak regular agar kegiatan tersebut tidak berbenturan jadwal dengan kegiatan yang lain.

Untuk menangani permasalahan tersebut maka diharapkan terdapat sistem informasi yang mampu menangani permasalahan di laboratorium prodi sistem informasi. Semua aktifitas perlu dicatat pada aplikasi tersebut, inputan bisa menggunakan *barcode* yang ada di Kartu Tanda Mahasiswa (KTM) dan kartu identitas dosen.

### 2. METODOLOGI

Untuk membangun sebuah sistem informasi agar sistem informasi tersebut dapat menjadi aplikasi yang *robust*, maka diperlukan pembangunan sistem yang terencana dengan baik [1]–[6]. Serta dalam membangun sebuah sistem harus sesuai dengan tujuan bisnis perusahaan agar tidak terjadi adanya *IT Productivity Paradox* [7], [8] yaitu dimana sistem informasi yang ada tidak berdampak baik pada tujuan bisnis perusahaan. Untuk itu sistem informasi yang akan dibuat harus dengan pertimbangan yang sangat matang. Pembuatan sistem informasi perlu diawasi dalam pembangunannya, baik proses

pelacakan dan kontrol proyek, manajemen risiko, jaminan kualitas, manajemen konfigurasi, tinjauan teknis, dan lainnya - diterapkan di seluruh proses [2].

Siklus hidup pengembangan sistem/*System Development Life Cycle (SDLC)* merupakan pola dalam mengembangkan/rekayasa perangkat lunak [9]–[13]. Tahapan dalam membangun sistem informasi menurut pressman [2] yaitu: *communication, planning, modeling, construction, and deployment*. *Communication* bisa berupa inisiasi project dan mengumpulkan kebutuhan. *Planning* yaitu memperkirakan selesainya proyek. *Modeling* yaitu membuat desain analisis untuk sistem informasi yang akan dibuat. *Construction* merupakan aktifitas pembuatan aplikasi (*code*) dan melakukan pengujian (*test*) setelah selesai membuat aplikasi [14]. Pengujian *code* bisa menggunakan *white box testing* ataupun *black box testing*. Tahapan terakhir yaitu *deployment* dimana sistem informasi yang telah dibuat di-*deliver* ke pengguna, serta jika ada masukan dari pengguna masih bisa disesuaikan agar dalam implementasi aplikasi tersebut tidak terdapat kendala yang tidak diharapkan.

Dalam artikel jurnal ini tahapan development yang dibahas adalah *modelling* dan *construction*. Untuk *modelling* dibagi menjadi 2 bagian, yaitu perancangan basis data dan *system modelling*. Perancangan basis data terdiri dari dua langkah, yaitu pembuatan *Conceptual Data Model (CDM)* dan *Physical Data Model (PDM)*. Untuk *system modelling* menggunakan UML dengan iconix process.

### 2.1 Perancangan Basis Data

Pembuatan basis data diawali dengan membuat *Conceptual Data Model (CDM)* dan *Physical Data Model (PDM)* [15]. *Conceptual Data Model (CDM)* merupakan konsep pembuatan entitas basis data secara logic. Tiap-tiap entitas berelasi dengan entitas yang lain. Relasi antar entitas tersebut menggambarkan hubungan antar tabel yang nantinya akan terbuat dalam basis data [16].

Langkah selanjutnya dalam rancangan basis data adalah *generate CDM ke PDM*. *PDM* merupakan gambaran entitas yang akan menjadi table dalam basis data. *PDM* tidak boleh diubah untuk menyesuaikan kebutuhan table yang ada di basis data. Jika *PDM* tidak sesuai, maka yang perlu diperbaiki adalah *CDM*-nya.

### 2.2 System Modelling

Pemodelan sistem untuk menggambarkan proses yang akan berjalan bisa menggunakan beberapa metode, antara lain terstruktur dan orientasi obyek. Pemodelan sistem menggunakan metode terstruktur bisa menggunakan system flow diagram dan DFD untuk menggambarkan aliran proses dalam entitas [17], [18]. Pemodelan sistem yang berorientasi obyek bisa menggunakan *Unified Modeling Language (UML)* [2], [9], [11], [19]–[26].

Pada artikel jurnal ini pemodelan sistem menggunakan iconix proses, yang diawali dari pembuatan *domain model, use case diagram, robustness diagram, dan sequence diagram* [9], [25]. *Domain model* menggambarkan daftar istilah yang akan digunakan pada sistem yang akan dibuat. Langkah selanjutnya yaitu memodelkan perilaku sistem dengan *Use case Diagram*. *Robustness diagram* menggambarkan class dan perilaku sistem yang terlibat dalam project. Sedangkan diagram *sequence* yaitu menggambarkan aktifitas use case secara detail.

### 2.3 Construction

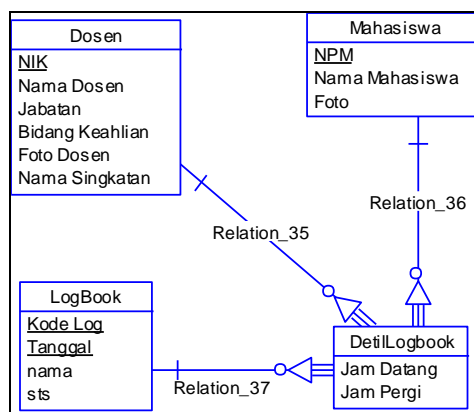
Pembuatan sistem informasi logbook ini menggunakan Visual Studio .Net 2005. Pada Visual Studio .Net 2005 sudah mendukung .Net Framework 2.0 dengan teknologi ADO.NET[27].

### 3. HASIL DAN PEMBAHASAN

Pada bab berikut akan dibahas hasil penelitian mulai dari desain basis data sampai dengan hasil dari pengembangan sistem informasi log book laboratorium.

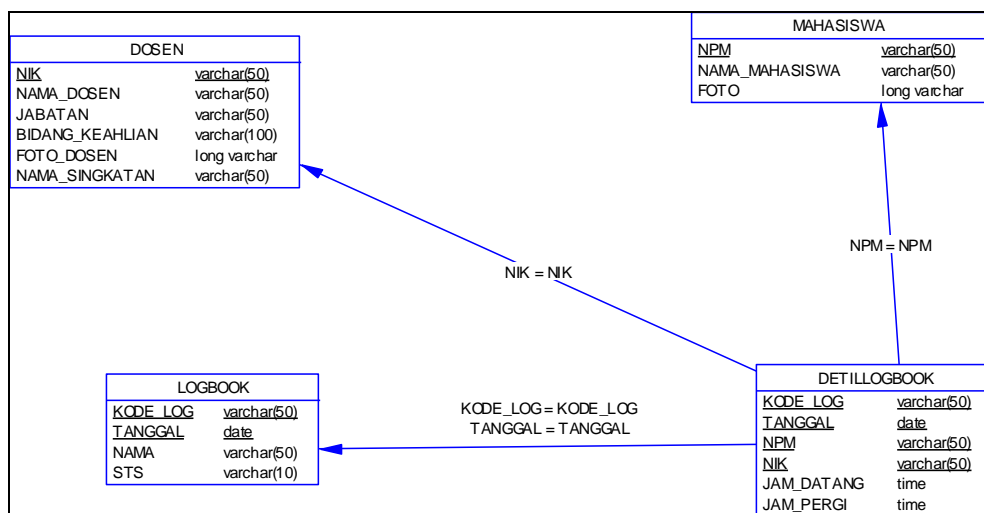
#### 3.1 Basis data Design

Dalam mendesain basis data diawali dengan membuat Conceptual Data Model (CDM). Dari gambar CDM terdapat empat entitas yang terlibat dalam pembuatan sistem informasi LogBook Laboratorium prodi Sistem Informasi, yaitu entitas Dosen untuk menampung data dosen, entitas Mahasiswa untuk menampung data mahasiswa, entitas Logbook dan detil logbook digunakan untuk mencatat aktifitas kegiatan yang ada di laboratorium. Pemodelan CDM seperti yang tergambar pada gambar 1.



Gambar 1. CDM SI LogBook

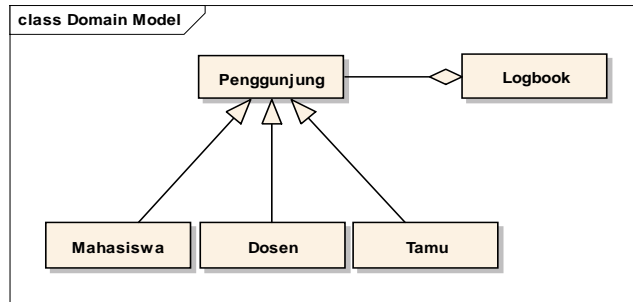
Pada gambar 2 yaitu Physical Data Model (PDM) yaitu hasil generate dari CDM dimana hasil dari PDM yang nantinya akan menjadi table yang ada di basis data yang akan digunakan untuk menyimpan data dari aplikasi logbook.



Gambar 2. PDM SI LogBook

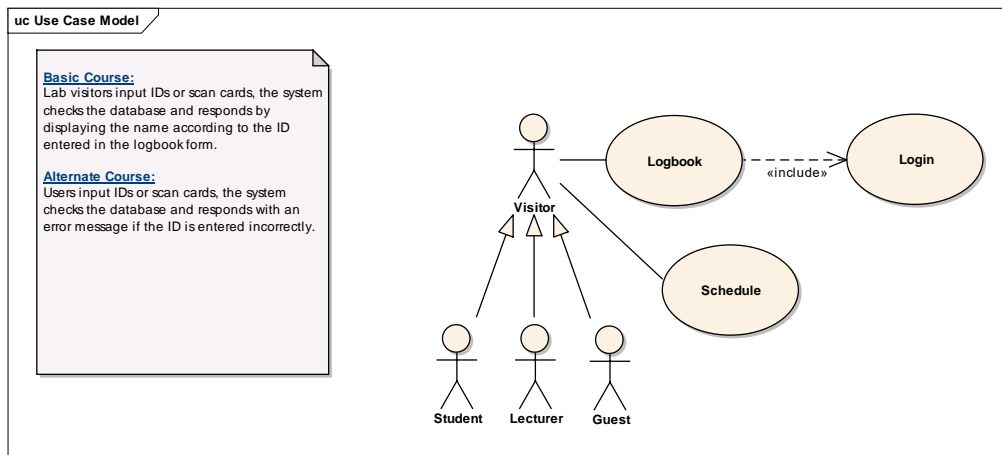
#### 3.2 System Modelling

Desain UML untuk iconix diawali dengan membuat domain model, yaitu penggambaran lingkungan yang akan ada sekitar aplikasi, yaitu pengunjung dan logbook itu sendiri. Pengunjung laboratorium dibagi menjadi tiga jenis, yaitu mahasiswa, dosen, dan tamu, tergambar pada gambar 3.



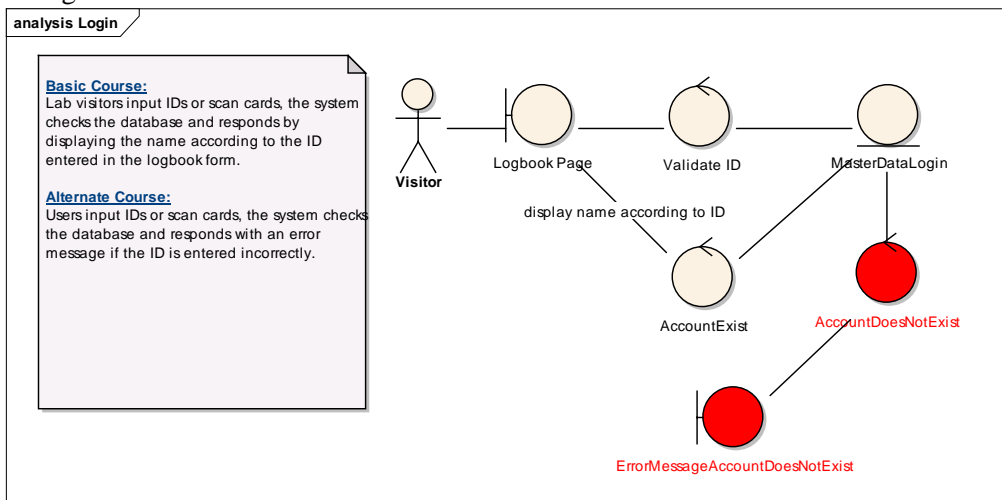
Gambar 3. Domain Model SI LogBook

Use case diagram menggambarkan sistem yang dijalankan pada aplikasi logbook, yaitu logbook itu sendiri dan jadwal penggunaan laboratorium. Untuk masuk ke logbook diharuskan untuk login terlebih dahulu. Model use case diagram digambarkan pada gambar 4.



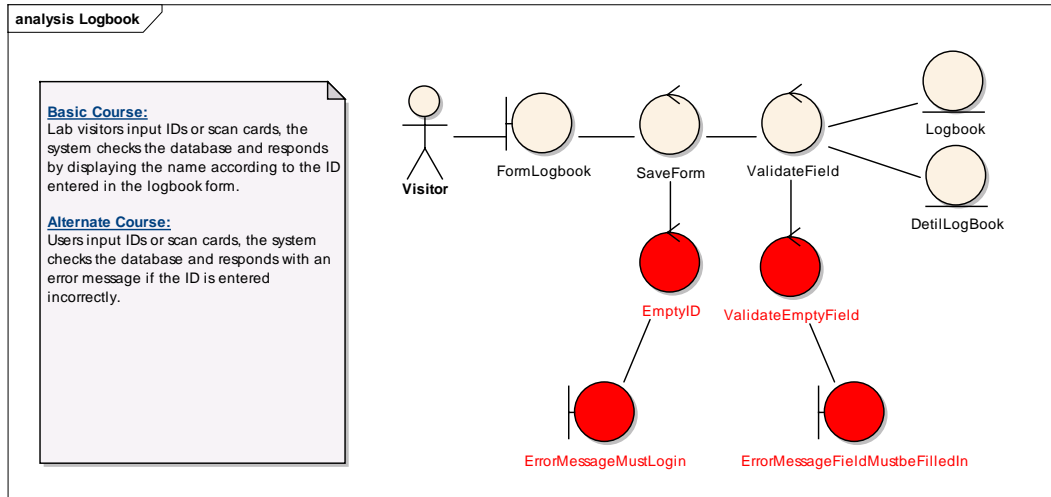
Gambar 4. Use case Diagram SI LogBook

Setelah mendesain use case diagram, selanjutnya dibuatkan robustness diagram, dimana pada diagram robustness diharapkan aplikasi yang dibangun bisa menjadi aplikasi yang kokoh, dimana semua kemungkinan error sudah dapat ditangani. Gambar 5 merupakan robustness diagram untuk menggambarkan sistem yang berjalan untuk use case login.



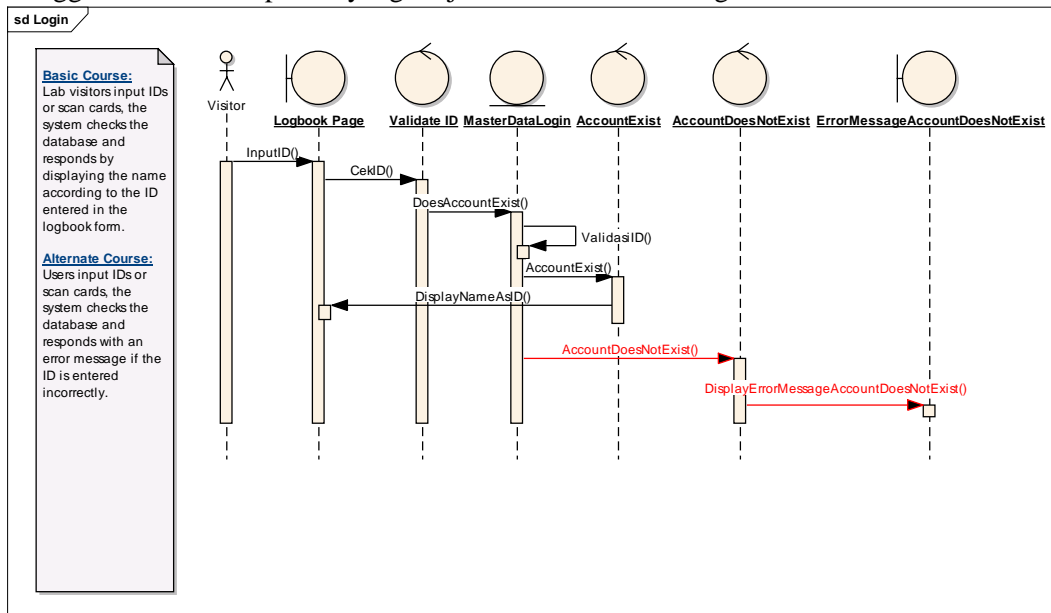
Gambar 5. Robustness Diagram Login SI LogBook

Gambar 6 merupakan robustness diagram untuk menggambarkan sistem yang berjalan untuk use case logbook. Hal ini untuk handle jika terdapat field yang kosong, maka sistem akan mengeluarkan peringatan untuk mengisi inputan yang kosong atau belum diisi.



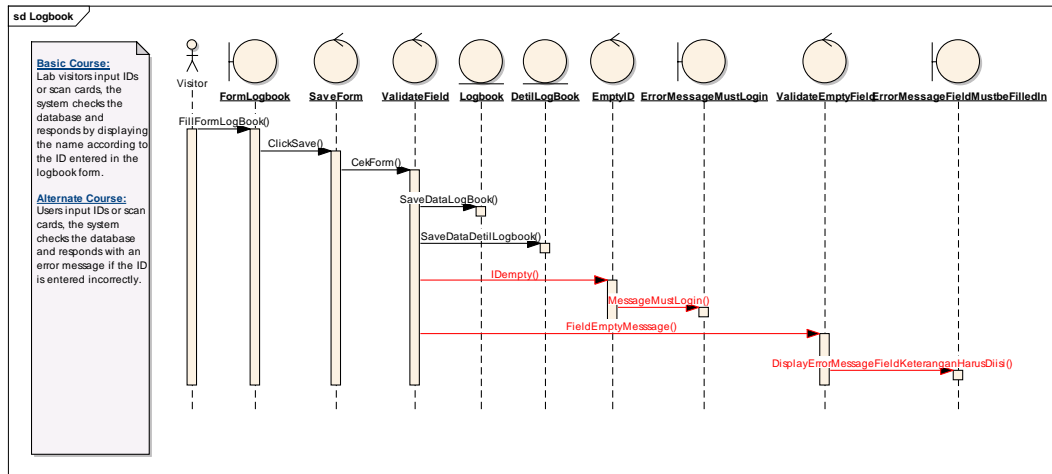
Gambar 6. Robustness Diagram Logbook SI LogBook

Setelah mendesain robustness diagram, maka langkah selanjutnya adalah mendesain sequence diagram. Gambar 7 merupakan sequence diagram untuk menggambarkan detail proses yang berjalan untuk use case login.



Gambar 7. Sequence Diagram Login SI LogBook

Gambar 8 merupakan sequence diagram untuk menggambarkan sistem yang berjalan untuk use case logbook.



Gambar 8. Sequence Diagram Logbook SI LogBook

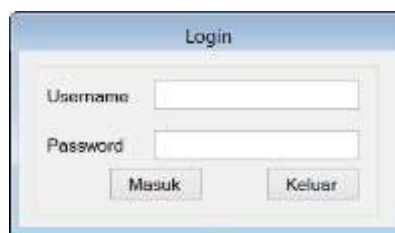
### 3.3 Construction

Setelah tahapan desain, langkah selanjutnya adalah develop aplikasi. Aplikasi logbook laboratorium dibangun dengan menggunakan visual studio .Net 2005. Gambar 9 merupakan tampilan utama aplikasi logbook. Pada halaman utama tersebut terdapat menu log Tamu, yaitu digunakan untuk mencatat tamu yang beraktifitas di laboratorium. Yang dimaksud tamu adalah selain dosen dan mahasiswa yang melakukan kegiatan di laboratorium, misal pembicara saat kuliah tamu yang diadakan di laboratorium, atau pelatih dari luar saat mengadakan pelatihan di laboratorium. Menu Log Dosen/Mahasiswa digunakan saat Dosen/Mahasiswa ada kegiatan di laboratorium. Menu Cetak data digunakan saat dibutuhkan print history aktifitas laboratorium.



Gambar 9. Tampilan utama menu Logbook

Gambar 10 adalah tampilan untuk login saat masuk logbook untuk semua aktifitas yang ada di laboratorium.



Gambar 10. Tampilan Login

Gambar 11 merupakan tampilan untuk tamu, misal pelatih atau dosen tamu selain dosen dari UPN "Veteran" Jawa Timur. Terdapat dua isian yaitu Nama tamu dan keterangan kegiatan di Laboratorium.



Gambar 11. Tampilan Log Tamu

Gambar 12 merupakan tampilan untuk log Mahasiswa dan Dosen. Terdapat 5 inputan, NPM mahasiswa, nama mahasiswa, kegiatan di Laboratorium, kegiatan di Laboratorium, dan penanggung jawab. Button simpan data untuk menyimpan inputan data tersebut sekaligus menampilkan data ke listview. Tampilan list view bisa difilter berdasarkan apakah data yang ditampilkan mahasiswa atau dosen. Untuk menu dosen tidak jauh berbeda. Inputan NPM atau NPT dosen bisa dilakukan dengan scan barcode yang ada di KTM atau kartu dosen.



Gambar 12. Tampilan Log Dosen dan Mahasiswa

Gambar 13 adalah tampilan form master untuk menyimpan data utama dosen. Diawali dengan memasukkan NPT/NIP dosen, selanjutnya Nama lengkap dosen, dan diikuti bidang keahlian masing-masing dosen. Setelah itu bisa dimasukkan juga foto dosen yang bersangkutan.



Gambar 13. GUI Master Form Dosen

#### 4. KESIMPULAN DAN SARAN

Kesimpulan dari penelitian ini adalah pembuatan sistem informasi Logbook laboratorium diawali analisis perancangan dengan Iconix Process menggunakan

pendekatan UML dan pembuatan aplikasi menggunakan Visual Studio .Net 2005 telah selesai dibuat yang selanjutnya dapat dilanjutkan implementasi di Laboratorium Sistem Informasi Fakultas Ilmu Komputer Universitas Pembangunan Nasional “Veteran” Jawa Timur.

## 5. DAFTAR RUJUKAN

- [1] N. C. H. J. Stephen Topper, “Model-Based Systems Engineering in Support of Complex Systems Development,” *JOHNS HOPKINS APL Tech. Dig.*, vol. 32, no. 1, pp. 419–432, 2013.
- [2] R. S. Pressman, *Software Engineering*, SEVENTH. McGraw-Hill, 2010.
- [3] R. K. Rainer and C. G. Cegielski, *Introduction to information systems*. John Wiley & Sons, 2011.
- [4] *Introduction to Information Systems: Supporting and Transforming Business by Rainer, R. Kelly, Prince, Brad, Cegielski, Casey G. (November 8, 2013) Paperback: Amazon.com: Books.* .
- [5] R. K. Rainer, B. Prince, and C. G. Cegielski, *Introduction to Information Systems, 5th Edition: Fifth Edition*, vol. 12. John Wiley & Sons, 2013.
- [6] R. K. Rainer, B. Prince, and C. G. Cegielski, *Introduction to information systems : supporting and transforming business.* .
- [7] S. Mukaromah and A. B. Putra, “Maturity level at university academic information system linking it goals and business goal based on COBIT 4.1,” in *MATEC Web of Conferences*, 2016, vol. 58.
- [8] R. (Ronald) Weber, *Information systems control and audit*. Prentice Hall, 1999.
- [9] D. Rosenberg and M. Stephens, *Use case driven object modeling with UML: Theory and Practice*. 2007.
- [10] U. G. Nurasiah (Sistem Informasi, Fakultas Ilmu Komputer dan Teknologi Informasi, “Perencanaan Pengembangan Sistem Informasi Pembayaran Uang Kuliah Dengan Metode SDLC Waterfall Nurasiah,” *J. Teknol. dan Rekayasa*, Vol. 19 No. 3, Desember 2014, vol. 19, no. 3, pp. 72–81, 2014.
- [11] I. Jacobson, G. Booch, and J. Rumbaugh, *The Unified Software Development Process (Paperback) (Addison-Wesley Object Technology Series)*. Addison-Wesley Professional, 1999.
- [12] A. Alshamrani and A. Bahattab, “A Comparison Between Three SDLC Models Waterfall Model , Spiral Model , and Incremental / Iterative Model,” vol. 12, no. 1, pp. 106–111, 2015.
- [13] R. Susanto, A. D. Andriana, R. Susanto, and A. D. Andriana, “PERBANDINGAN MODEL WATERFALL DAN PROTOTYPING,” vol. 14, no. 1, pp. 41–46.
- [14] G. D. Everett and R. McLeod, *Software Testing: Testing Across the Entire Software Development Life Cycle*. 2006.
- [15] R. Elmasri and S. B. Navathe, *Fundamentals of Basis data Systems Sixth Edition*. 2016.
- [16] S. B. N. Ramez Elmasri, *Fundamental of Basis data Systems SIXTH EDITION*, 6th ed. Boston,Massachusetts: Addison-Wesley, 2011.
- [17] “By James A. Hall: Accounting Information Systems Seventh (7th) Edition: J.K: Amazon.com: Books.” [Online]. Available: <https://www.amazon.com/James-Hall-Accounting-Information-Systems/dp/B004HGVY44>. [Accessed: 28-Oct-2019].
- [18] J. E. K. KENNETH E. KENDALL, *SYSTEMS ANALYSIS and DESIGN*, 8th ed. Pearson Education, Inc., publishing as Prentice Hall, 2011.
- [19] “UML Distilled: A Brief Guide to the Standard Object Modeling Language (3rd Edition): Martin Fowler: 0785342193688: Amazon.com: Books.” [Online]. Available: <https://www.amazon.com/UML-Distilled-Standard-Modeling-Language/dp/0321193687>. [Accessed: 28-Oct-2019].



- [20] J. S. Topper and N. C. Horner, “Guiding Principles Several overarching principles guide the conceptual modeling process. OVERVIEW OF THE CONCEPTUAL MODELING PROCESS,” 2013.
- [21] “Object Oriented Software Engineering: A Use Case Driven Approach: Ivar Jacobson: 9780201544350: Amazon.com: Books.” [Online]. Available: <https://www.amazon.com/Object-Oriented-Software-Engineering-Approach/dp/0201544350>. [Accessed: 28-Oct-2019].
- [22] “Driving Design with Use Cases.” [Online]. Available: [http://www8.tfe.umu.se/courses/systemteknik/Doit/UML/Driving Design with Use Cases.htm](http://www8.tfe.umu.se/courses/systemteknik/Doit/UML/Driving%20Design%20with%20Use%20Cases.htm). [Accessed: 28-Oct-2019].
- [23] “Design and development of school assets management system — UNIC | Research Portal.” [Online]. Available: <https://pure.unic.ac.cy/en/publications/design-and-development-of-school-assets-management-system>. [Accessed: 28-Oct-2019].
- [24] G. Liu, X. Wang, and J. Fang, “A UML-based Simple Function Point Estimation Method and Tool,” no. c, pp. 39–45, 2016.
- [25] D. Rosenberg, M. Stephens, and M. Collins-Cope, *Agile development with ICONIX process: People, process, and pragmatism*. A-Press, 2005.
- [26] B. Thakare, “Deriving Best Practices from Development Methodology Base ( Part 2 ),” vol. 1, no. 6, pp. 1–8, 2012.
- [27] A. Ramadhan, *Vb.net 2005*. PT Elex Media Komputindo, 2006.