

Tinjauan Literatur Sistematis Mengenai Pengembangan Perangkat Lunak Berorientasi Objek

Hilmiyah Maulida Issyatirrahim^{1*}, Irna Syarifatul Aen², Nabila Nur Azizah³

UIN Syarif Hidayatulla, Jl. Ir H. Juanda No.95 Ciputat Kota Tangerang Selatan 15412

^{*1}email: hilmiyahmaulida5@gmail.com

²email: irnasyarifatul@gmail.com

³email: nabilazahh@gmail.com

(Naskah diterima: 4 Mei 2024; Naskah direvisi: 25 Juni 2024; Naskah diterbitkan: 1 Desember 2024)

ABSTRAK – Di era digital ini, perangkat lunak telah menjadi bagian tak terpisahkan dari kehidupan manusia. Perkembangan teknologinya pun semakin pesat, menuntut metode pengembangan perangkat lunak yang efektif, fleksibel, dan mudah beradaptasi. Salah satu pendekatan yang banyak digunakan adalah Pengembangan Perangkat Lunak Berorientasi Objek (PBO). Penelitian ini bertujuan untuk meninjau beberapa penelitian secara sistematis mengenai pengembangan perangkat lunak berorientasi objek, dengan menggunakan metode penelitian tinjauan literatur sistematis atau Systematic Literature Review (SLR). Dalam penulisan ini maka didapatkan hasil yaitu, terdapat berbagai pendekatan yang digunakan dalam pengembangan perangkat lunak berorientasi objek, seperti analisis berorientasi objek, desain berorientasi objek dengan penggunaan Unified Modeling Language (UML), serta metode pengukuran metrik berorientasi objek. Pemrograman berorientasi objek juga memfasilitasi pembuatan sistem yang lebih mudah dipahami, dipelihara, dan diperluas. Pengembangan berorientasi objek juga tidak hanya meningkatkan kualitas produk akhir tetapi juga mempercepat siklus pengembangan secara keseluruhan. Diperlukan juga pemodelan yang cermat dan validasi menyeluruh untuk meminimalkan risiko dampak negatif pada pengembangan dan pengoperasian sistem.

Kata Kunci – Pengembangan berorientasi objek, Perangkat lunak, Systematic Literature Review.

A Systematic Literature Review Concerning Object-Oriented Software Development

ABSTRACT – In this digital era, software has become an inseparable part of human life. Technological developments are increasingly rapid, demanding software development methods that are effective, flexible and easily adaptable. One approach that is widely used is Object-Oriented Software Development (PBO). This writing aims to systematically review several studies regarding object-oriented software development, using the systematic literature review (SLR) research method. In this paper, the results obtained are that there are various approaches used in developing object-oriented software, such as object-oriented analysis, object-oriented design using the Unified Modeling Language (UML), as well as object-oriented metric measurement methods. Object-oriented programming also facilitates the creation of systems that are easier to understand, maintain, and extend. Object-oriented development also not only improves the quality of the final product but also speeds up the overall development cycle. Careful modeling and thorough validation are also required to minimize the risk of negative impacts on system development and operation.

Keywords - Object-oriented development, software, Systematic Literature Review.

1. PENDAHULUAN

Saat ini, pengembangan teknologi sistem informasi berlangsung dengan mudah, cepat, dan selalu mengikuti perkembangan terbaru, di mana informasi bisa diakses dari mana saja dan kapan saja. Kemajuan teknologi dan informasi sangat penting

dalam mendukung kemajuan yang pesat serta dalam menciptakan teknologi yang selalu mutakhir. Teknologi komputer sangat mendukung pengolahan data untuk menghasilkan sistem informasi yang memberikan kinerja optimal [1].

Rekayasa perangkat lunak saat ini berfokus pada pendekatan, penggunaan, proses, model, dan

metode dalam rangka untuk merancang dan memberikan solusi teknologi bagi kelompok orang tertentu. Baik entitas yang sedang berkembang maupun yang sudah mapan, menggunakan pendekatan ini untuk menciptakan solusi teknologi yang sesuai dengan kebutuhan spesifik kelompok tersebut. Tujuannya adalah untuk menghadirkan metodologi inovatif yang mendukung pembangunan dan penggunaan perangkat lunak dalam prosesnya [2].

Tantangan dan Praktik Terbaik dalam Pengembangan Perangkat Lunak Berorientasi Objek (OOP) telah menjadi fokus utama dalam dunia teknologi informasi. Dalam era di mana sistem perangkat lunak semakin kompleks dan tuntutan akan kualitas serta kehandalan semakin tinggi, pemahaman yang mendalam tentang tantangan yang dihadapi dan praktik terbaik yang dapat diterapkan menjadi krusial. Dalam konteks ini, penelitian sistematis dalam bentuk literature review muncul sebagai alat atau instrumen yang efektif untuk menyelidiki, mengidentifikasi, dan menyusun pemahaman kolektif tentang tantangan dan praktik terbaik yang berkembang dalam pengembangan perangkat lunak berorientasi objek [3].

Beberapa penelitian terdahulu telah dilakukan mengenai pengembangan berorientasi objek, salah satunya yang dilakukan oleh Kornelis dan Azhari [4], keduanya membahas terkait metrics dalam konteks pengembangan perangkat lunak yang merupakan ukuran atau alat yang sering digunakan untuk mengevaluasi, mengukur, dan memantau berbagai aspek dari proses pengembangan perangkat lunak, kualitas kode, dan kinerja tim.

Penelitian selanjutnya dilakukan oleh peneliti Muhammad et al. [5], yang berkaitan dengan penggunaan metode OOP itu sendiri dalam proses pengembangan perangkat lunak sederhana berupa aplikasi keuangan untuk mahasiswa. Pengembangan berorientasi objek dipilih karena pada bagian proses implementasinya dapat dilakukan dengan cepat sesuai karakteristik mahasiswa mulai dari analisis permasalahan hingga tahapan pengujian aplikasi.

Penelitian ini bertujuan untuk menyajikan hasil dari literature review sistematis yang dilakukan untuk mengeksplorasi landscape saat ini dari tantangan yang dihadapi dan praktik terbaik yang diadopsi dalam pengembangan perangkat lunak berorientasi objek. Melalui pendekatan ini, peneliti tersebut berharap dapat memberikan wawasan yang mendalam bagi para praktisi dan peneliti tentang lanskap pengembangan perangkat lunak berorientasi objek saat ini, serta menyediakan panduan praktis yang dapat membantu mengatasi tantangan yang dihadapi dan menerapkan praktik terbaik dalam konteks pengembangan perangkat lunak yang kompleks.

2. TINJAUAN PUSTAKA

2.1 Pengembangan Perangkat Lunak

Pengembangan perangkat lunak merupakan serangkaian proses kompleks yang melibatkan berbagai tahap mulai dari perencanaan, analisis, desain, implementasi, pengujian, serta pemeliharaan. Proses ini sangat penting untuk menghasilkan perangkat lunak berkualitas tinggi, yang sesuai kebutuhan pengguna, serta mudah untuk dipelihara.

Perencanaan, adalah tahap awal dalam pengembangan perangkat lunak. Pada tahap ini, tujuan dan ruang lingkup proyek perangkat lunak ditentukan. Analisis Kebutuhan, melibatkan pengumpulan data, spesifikasi perangkat lunak, dan pemodelan kebutuhan perangkat lunak. Dokumen kebutuhan dihasilkan sebagai acuan untuk tahap berikutnya. Desain pada perangkat lunak dibagi menjadi dua sub-tahap: desain arsitektur dan desain detail. Desain arsitektur menentukan struktur tinggi dari sistem, termasuk modul utama dan hubungan antar modul. Implementasi adalah tahap di mana desain sistem yang diterjemahkan ke dalam kode program. Pengujian, adalah tahap di mana perangkat lunak diuji untuk memastikan bahwa ia berfungsi sesuai dengan kebutuhan yang telah ditentukan. Pemeliharaan, adalah tahap setelah perangkat lunak dirilis, di mana perangkat lunak diperbarui dan diperbaiki seiring waktu. Ada berbagai metodologi yang digunakan dalam pengembangan perangkat lunak, seperti *Waterfall*, *Agile*, *Scrum*, dan *DevOps* [6].

2.2 Pemrograman Berorientasi Objek

Pemrograman berorientasi objek, juga dikenal sebagai pemrograman berorientasi objek (OOP), dimana pada teknik ini, pendekatan pemrograman yang menggunakan object dan class. Dengan OOP membuat sebuah program menjadi lebih mudah, dan beberapa keuntungan dalam program berorientasi objek atau OOP antara lain adalah:

- 1) *Reusability*, artinya kode yang sudah dibuat dapat digunakan kembali.
- 2) *Extensibility*, maksudnya konsep ini seorang programmer memiliki sebuah kemampuan untuk membuat method baru atau mengubah yang sudah ada sesuai keinginan tanpa harus membuat kode dari awal
- 3) *Maintability*, maksudnya manajemen kode menjadi lebih mudah dikelola ketika aplikasi dikembangkan pada skala yang besar, yang memungkinkan terjadinya kesalahan selama pembuatannya. Hal ini dapat diatasi dengan cara menerapkan pemrograman OOP karena konsep modularitas sudah ada dalam pemrograman OOP [7].

Salah satu keuntungan utama dari teknik pemrograman berorientasi objek dibandingkan dengan teknik pemrograman prosedural adalah

memungkinkan programmer untuk membuat modul yang tidak perlu dimodifikasi setiap kali sebuah jenis objek baru ditambahkan. Seorang pengembang perangkat lunak mampu menghasilkan object baru yang memperoleh banyak atribut dari entitas yang sudah ada sebelumnya. Akibatnya, ini meningkatkan kemudahan perubahan dalam program perangkat lunak berorientasi objek [8].

2.3 Tinjauan Literatur Sistematis

Tinjauan Literatur Sistematis atau *Systematic Literature Review* (SLR) melibatkan langkah-langkah yang sistematis untuk mengumpulkan dan mengevaluasi literatur yang relevan untuk mengidentifikasi mengenai apa yang telah diketahui sebelumnya tentang suatu topik tertentu, bagaimana penelitian sebelumnya dilakukan, dan di mana ada kesenjangan pengetahuan. Metode ini melibatkan proses pencarian literatur yang komprehensif, penilaian kualitas studi-studi yang relevan, serta sintesis temuan-temuan tersebut dalam suatu laporan yang sistematis dan terstruktur. SLR biasanya dilakukan dalam beberapa tahap yang terdefinisi dengan jelas, termasuk pada formulasi pertanyaan penelitian, pengembangan strategi pencarian, seleksi studi yang relevan berdasarkan kriteria inklusi dan eksklusi yang telah ditetapkan, penilaian kualitas metodologi studi-studi yang dipilih, serta analisis dan interpretasi data secara sistematis [9].

SLR bertujuan untuk memberikan ikhtisar yang komprehensif tentang bukti yang tersedia terkait dengan pertanyaan penelitian tertentu. Teknik SLR ini biasa mengikuti protokol yang telah ditentukan sebelumnya untuk memastikan objektivitas untuk mengurangi bias dalam memilih dan menilai sebuah studi [10].

3. METODE PENELITIAN

Metodologi yang digunakan dalam penelitian ini adalah tinjauan literatur sistematis. Jurnal yang membahas tentang pengembangan perangkat lunak berorientasi objek digunakan sebagai data untuk menyusun penelitian ini. Jurnal yang dipilih merupakan terbitan tahun 2019-2024. Pengumpulan data ini dimulai dengan klasifikasi, identifikasi, pengumpulan data dan analisis.

3.1 Pertanyaan Penelitian

Tahap pertama yang harus dilakukan dalam membuat SLR adalah membuat beberapa pertanyaan penelitian yang sesuai dengan topik yang sudah ditentukan. Adapun *research question* dalam penelitian ini antara lain.

RQ1: Pendekatan apa saja yang digunakan dalam pengembangan perangkat lunak berorientasi objek?

RQ2: Bagaimana efektivitas praktik terbaik dalam pengembangan perangkat lunak berorientasi objek?

RQ3: Apa saja dampak yang dihadapi dalam menerapkan pengembangan perangkat lunak berorientasi objek?

3.2 Proses Pencarian

Tahap selanjutnya setelah *research question* adalah *search process* atau tahap pencarian sumber penelitian yang relevan dengan topik yang sudah ditentukan sebelumnya. Pencarian sumber pada penelitian di sini dilakukan melalui *publish or perish* dengan *search* yang dipilih adalah *crossref* dan *google scholar* menggunakan keyword "pendekatan berorientasi objek (OOP)".

3.3 Kriteria Inklusi dan Eksklusi

Tahap ketiga yang dilakukan setelah *search process* adalah kriteria inklusi dan eksklusi, tahap ini digunakan dalam menentukan data dan sumber yang layak untuk digunakan. Adapun kriteria dalam penelitian ini, antara lain.

- 1) Sumber data jurnal dimulai dari 2019-2024.
- 2) Sumber jurnal di ambil dari aplikasi *publish or perish*.
- 3) Sumber data jurnal yang digunakan hanya berkaitan dengan pengembangan perangkat lunak menggunakan pendekatan berorientasi objek (OOP).

4. HASIL DAN PEMBAHASAN

4.1 Hasil Proses Penelusuran dan Kriteria Inklusi Eksklusi

Bagian ini adalah hasil dari proses penelusuran dan kriteria inklusi dan eksklusi yang berhasil mendapatkan 15 jurnal yang sudah melalui tahap seleksi sesuai dengan kriteria yang dimulai dari tahun 2019 sampai tahun 2024. Jurnal yang digunakan hanya membahas mengenai kegiatan pengembangan perangkat lunak berorientasi objek. Berikut adalah tabel jurnal yang telah melalui tahap seleksi:

Tabel 1. Daftar Judul Artikel Ilmiah yang Terpilih

No	Artikel Jurnal	Tahun
[11]	"Pengembangan Sistem Informasi Wedding Organizer Menggunakan Pendekatan Sistem Berorientasi Objek pada CV Pesta"	2021
[12]	"Perancangan Sistem Informasi Perawatan Mesin Menggunakan Pendekatan Analisis Berorientasi Objek"	2019
[13]	"Rancang Bangun Sistem E-Marketing dengan Pendekatan Sistem Berorientasi Objek pada PT. Khaula Prima"	2021
[14]	"Penerapan Konsep Object Oriented	2019

No	Artikel Jurnal	Tahun
	Programming untuk Aplikasi Pembuat Surat”	
[15]	“Pengembangan Aplikasi untuk Mengukur Reusability Perangkat Lunak Menggunakan Object-Oriented Metric”	2021
[16]	“Implementasi Konsep Pemrograman Berorientasi Objek pada Aplikasi Sistem Parkir Menggunakan Bahasa Pemrograman Java”	2019
[17]	“Implementasi Konsep Pemrograman Berorientasi Objek dalam Aplikasi Pembukuan Keuangan Penjual Jus Buah Menggunakan Bahasa Pemrograman Java”	2022
[18]	“Implementasi Konsep Pemrograman Berorientasi Objek Oriented Metodologi dan UML pada Pengembangan Sistem Informasi Keuangan Organisasi”	2021
[19]	“Pengembangan Sistem Informasi Administrasi Akademik dengan Rancangan Modul Program Menggunakan Bahasa Pemrograman Berorientasi Objek”	2019
[20]	“Perancangan Aplikasi Android Bank Sampah Menggunakan Metode Object Oriented di Wilayah Salatiga”	2023
[21]	“Waterfall sebagai Model Pengembangan Sistem Persediaan Apotek Berorientasi Objek”	2022
[22]	“Pemodelan Perangkat Lunak Pemilihan Calon Pendoron Darah Potensial Berbasis Object Oriented”	2022
[23]	“Implementasi Pendekatan Terpadu Berorientasi Objek untuk Mengembangkan Sistem Informasi Terpadu di PT. Sapt Karya”	2020
[24]	“Pembuatan Website Sistem Informasi Objek Wisata Menggunakan Pendekatan Object Oriented Analysis and Design (OOAD)”	2023
[25]	“Perancangan Aplikasi Perkantoran Elektronik dengan Menggunakan Metode Object Oriented Analysis Design Berbasis Web pada KJPP DAR”	2023

4.2 Hasil Analisis Jawaban dari Pertanyaan Penelitian

Selanjutnya, pada tahap ini adalah menuliskan jawaban dari hasil analisis jurnal terpilih dalam Tabel 1 berdasarkan pertanyaan penelitian yang sudah ditentukan sebelumnya.

RQ1: Pendekatan apa saja yang digunakan dalam pengembangan perangkat lunak berorientasi objek?

Pendekatan yang digunakan dalam mengembangkan perangkat lunak berorientasi objek menunjukkan bahwa berbagai pendekatan yang digunakan mencakup analisis berorientasi objek [11], [12], [13], [14], design berorientasi objek

(termasuk penggunaan unified modeling language atau UML) [15], [16], [17].

Pemrograman berorientasi objek dengan bahasa pemrograman seperti JAVA, serta *Object-Oriented Analysis and Design* (OOAD) [15], [18], [19]. Terdapat juga menggunakan metrik berorientasi objek untuk bisa mengukur kualitas dan reusabilitas perangkat lunak.

Beberapa jurnal juga menunjukkan kombinasi dengan metodologi seperti *waterfall* yang biasanya digabungkan dengan pendekatan berorientasi Objek [20], [21], [22]. serta penerapan pendekatan terpadu berorientasi objek untuk dapat mengembangkan sistem informasi yang juga efisien, modular dan mudah dirawat dan dievaluasi (*maintenance*) [23], [24], [25]. Pendekatan-pendekatan ini membantu dalam menangani kompleksitas sistem, memastikan kualitas perangkat lunak, meningkatkan reusabilitas komponen, dan mempermudah pemeliharaan sistem secara keseluruhan.

RQ2: Bagaimana efektivitas praktik terbaik dalam pengembangan perangkat lunak berorientasi objek?

Efektivitas praktik terbaik pada pengembangan perangkat lunak berorientasi objek (OOP) ini juga dapat dilihat dari berbagai manfaat yang ditawarkan. Pertama, pemrograman berorientasi objek membuat seluruh proses pengembangan menjadi lebih efisien. Pengembang juga dapat menciptakan sebuah sistem perangkat lunak yang lebih mudah untuk dapat dipahami, dipelihara, dan diperluas, sehingga menghasilkan produk perangkat lunak yang lebih berkualitas dan proses pengembangan yang lebih efisien [25]. Pemrograman berorientasi objek juga sering dipilih dikarenakan metodologi yang lama sering menimbulkan masalah, terjadinya kesulitan menerjemahkan hasil dari satu tahap pengembangan ke tahap pengembangan berikutnya. OOP juga merupakan strategi baru untuk dapat membuat program atau dapat merancang sebuah sistem dengan fokus pada objek, sehingga memudahkan transisi antar tahap pengembangan [19].

Pengembangan perangkat lunak berorientasi objek mengarah pada sistem yang lebih andal, dapat dipelihara, dan dapat disesuaikan yang secara efektif memenuhi kebutuhan pengguna [14]. Dengan pemrograman berorientasi objek, aplikasi dapat mengintegrasikan dan menghubungkan satu fungsi ke fungsi lainnya, memberikan kemampuan untuk digunakan kembali, yaitu kemampuan untuk menggunakan kembali kode ketika mengembangkan aplikasi lain. Hal ini tidak hanya meningkatkan produktivitas programmer, tetapi juga memastikan bahwa kode yang digunakan telah teruji dan berkualitas baik [16], [17].

Selain itu, dengan memisahkan logika program menjadi objek-objek terpisah, pemrograman

berorientasi objek meningkatkan kualitas kode dan membuatnya lebih mudah untuk dipelihara. Objek terpisah ini memungkinkan setiap bagian dari program untuk dikembangkan dan diuji secara independen, mengurangi kompleksitas dan meningkatkan efisiensi pengembangan [19]. Secara keseluruhan, praktik terbaik pada pemrograman berorientasi objek telah terbukti efektif dalam meningkatkan kualitas dan mutu perangkat lunak, produktivitas pengembang, dan efisiensi proses pengembangan, menjadikannya pilihan yang lebih baik daripada metodologi pengembangan perangkat lunak lama yang tradisional.

RQ3: Apa saja dampak yang dihadapi dalam menerapkan pengembangan perangkat lunak berorientasi objek?

Dampak dari menerapkan pengembangan berorientasi objek terhadap pengembang diantaranya adalah pengembang sering menghadapi kesulitan dalam menangani kerumitan sistem berorientasi objek karena hubungan yang rumit dan interaksi antara objek yang memerlukan desain dan implementasi yang cermat. Banyak juga objek yang berinteraksi satu sama lain dan pengembang perlu memahami bagaimana mereka berinteraksi dan mempengaruhi satu sama lain, hal itu dapat menimbulkan kompleksitas. Manajemen yang buruk dari pengembangan perangkat lunak berorientasi objek dapat mengakibatkan sistem yang sulit untuk dipahami, dioperasikan, dan ditingkatkan [22], [24].

Melakukan pemeriksaan dan validasi secara menyeluruh juga sangat penting untuk menjamin berfungsinya sistem dengan benar dan sesuai, karena perubahan pada suatu bagian sistem dapat menyebabkan dampak pada entitas terkait, memerlukan pengujian dan validasi menyeluruh untuk memastikan kecocokan sistem. Karena ketika memiliki banyak objek yang saling berinteraksi, membuat perubahan pada satu objek bisa merambat ke objek lainnya [25].

Abstraksi dan pemodelan juga dapat memiliki dampak yang signifikan pada pengembang karena melibatkan pemecahan sistem yang rumit menjadi komponen yang lebih kecil dan lebih mudah dikelola. Praktek abstraksi dan pemodelan sistem dapat menyederhanakan dan mengatur alur proses pengembangan perangkat lunak. Melalui abstraksi, pengembang atau developer sistem dipaksa untuk mendapatkan pemahaman menyeluruh tentang fungsionalitas sistem dan hubungan diantara berbagai elemennya [19], [22].

5. SIMPULAN

Berdasarkan tinjauan literatur sistematis tentang pengembangan perangkat lunak berorientasi objek, dapat disimpulkan bahwa dengan pendekatan OOP

memberikan kerangka kerja yang kuat untuk mengatasi kompleksitas dan meningkatkan kualitas perangkat lunak. Berbagai pendekatan seperti analisis berorientasi objek, desain dengan UML, dan penggunaan metrik berorientasi objek telah terbukti efektif dalam mengukur kualitas serta reusabilitas perangkat lunak. Praktik terbaik dalam OOP, seperti modularitas kode, reusabilitas, dan kemudahan pemeliharaan, juga telah meningkatkan efisiensi dan produktivitas pengembang dalam menghadapi proyek-proyek yang kompleks. Namun, pada pengembangan perangkat lunak berorientasi objek selalu ada tantangan yang nyata. Kompleksitas sistem berorientasi objek, interaksi antara objek yang kompleks, serta manajemen yang buruk dapat menghambat pemahaman dan pemeliharaan sistem secara efektif. Penting untuk melakukan pemodelan yang cermat dan validasi menyeluruh guna mengurangi risiko dampak negatif yang mungkin terjadi. Dengan memahami tantangan ini dan menerapkan praktik terbaik, pengembang dapat maksimal menggunakan potensi OOP untuk menghasilkan solusi perangkat lunak yang kuat, mudah dipelihara, dan dapat berkembang secara berkelanjutan.

DAFTAR PUSTAKA

- [1] S. F. Arief and Y. Sugiarti, "Literature Review: Analisis Metode Perancangan Sistem Informasi Akademik Berbasis Web," *J. Ilm. ILMU Komput.*, vol. 8, no. 2, pp. 87-93, Sep. 2022, doi: 10.35329/jiik.v8i2.229.
- [2] M. Rizky and Y. Sugiarti, "Penggunaan Metode Scrum Dalam Pengembangan Perangkat Lunak: Literature Review," *Journal of Computer Science an Engineering (JCSE)*, vol. 3, no. 1, pp. 41-48, 2021.
- [3] W. Andriyani *et al.*, *Pengembangan Sistem Berbasis Data*. Makassar: Tohar Media Workshop, 2024.
- [4] K. Letelay and A. Sn, "Evaluasi Perangkat Lunak Lunak dengan Metrics Berorientasi Objek," 2012.
- [5] M. Darwis, D. Apriani, D. Islamiyati, and R. Agam, "Pengembangan Aplikasi MyNeeds Berorientasi Objek untuk Memantau Keuangan Mahasiswa," *Syntax J. Inform.*, vol. 12, no. 02, pp. 16-26, Nov. 2023, doi: 10.35706/syji.v12i02.9926.
- [6] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 8th ed. New York: McGraw Hill, 2014.
- [7] C. Ramadhani, *Dasar Algoritma dan Struktur Data dengan Bahasa Java*, 1st ed. Yogyakarta: Andi Publisher, 2015.

- [8] M. Weisfeld, *Object-Oriented Thought Process, The (Developer's Library)*, 4th ed. Addison-Wesley Professional, 2013.
- [9] J. A. Bell, *Doing Your Research Project*, Revised. Maidenhead: Open University Press.
- [10] R. Kiteley and C. Stogdon, *Literature Reviews in Social Work*, 3rd ed. New York: SAGE Publications, 2013.
- [11] M. Siddik and A. Sirait, "Pengembangan Sistem Informasi Administrasi Akademik Dengan Rancangan Modul Program Menggunakan Bahasa Pemrograman Berorientasi Objek," *JOISIE J. Inf. Syst. Inform. Eng.*, vol. 2, no. 1, p. 51, Jan. 2019, doi: 10.35145/joisie.v2i1.251.
- [12] W. Kosasih, I. K. Sriwana, and W. J. Purnama, "Perancangan Sistem Informasi Perawatan Mesin Menggunakan Pendekatan Analisis Berorientasi Objek," *J. Ilm. Tek. Ind.*, vol. 6, no. 3, Jun. 2019, doi: 10.24912/jitiuntar.v6i3.4246.
- [13] M. Aman, "Rancang Bangun Sistem E-Marketing dengan Pendekatan Sistem Berorientasi Object pada PT. Khaula Prima," *Insan Pembang. Sist. Inf. Dan Komput. IPSIKOM*, vol. 9, no. 1, Jun. 2021, doi: 10.58217/ipsikom.v9i1.195.
- [14] Y. A. M. Aziz and M. Fajar, "Implementasi Pendekatan Berorientasi Objek Terpadu untuk Pengembangan Sistem Informasi yang Terintegrasi pada PT. Sapta Karya," *J. Ilmu Komput.*
- [15] Universitas Patria Artha and M. Rais, "Penerapan Konsep Object Oriented Programming Untuk Aplikasi Pembuat Surat," *PROtek J. Ilm. Tek. Elektro*, vol. 6, no. 2, pp. 96-101, Sep. 2019, doi: 10.33387/protk.v6i2.1242.
- [16] A. Apandi, "Pembuatan Website Sistem Informasi Objek Wisata Menggunakan Pendekatan Object Oriented Analysis and Design (OOAD)," *J. Tek. Dan Sci.*, vol. 2, no. 2, pp. 23-33, Jun. 2023, doi: 10.56127/jts.v2i2.633.
- [17] S. Rivai and T. Tukino, "Perancangan Aplikasi Perkantoran Elektronik dengan Menggunakan Metode Object Oriented Analysis Design Berbasis Web pada KJPP DAR," *Comput. Based Inf. Syst. J.*, vol. 11, no. 1, pp. 42-54, Mar. 2023, doi: 10.33884/cbis.v11i1.6865.
- [18] A. N. Iman and T. A. Kurniawan, "Pengembangan Aplikasi untuk Mengukur Reusability Perangkat Lunak menggunakan Object-Oriented Metric".
- [19] R. Efendi, A. Erlanshari, and M. R. Herliansyah, "Permodelan Perangkat Lunak Pemilihan Calon Pendonor Darah Potensial Berbasis Object Oriented," *JSAI J. Sci. Appl. Inform.*, vol. 5, no. 1, pp. 1-11, Feb. 2022, doi: 10.36085/jsai.v5i1.2254.
- [20] N. A. Ilham, "Implementasi Konsep Pemrograman Berorientasi Objek pada Aplikasi Sistem Parkir Menggunakan Bahasa Pemrograman Java," *J. Edukasi Elektro*, vol. 3, no. 2, Jan. 2020, doi: 10.21831/jee.v3i2.28293.
- [21] Z. A. W. Sugandi, Y. A. Nugraha, S. N. Anam, and I. Darmayanti, "Implementasi Konsep Pemrograman Berorientasi Objek Dalam Aplikasi Pembukuan Keuangan Penjual Jus Buah Menggunakan Bahasa Pemrograman Java," vol. 8, no. 1, 2022.
- [22] C. Cahyaningtyas and E. Sedyono, "Perancangan Aplikasi Android Bank Sampah Menggunakan Metode Object Oriented di Wilayah Salatiga," *J. Inf. Technol.*, vol. 3, no. 1, pp. 41-48, Mar. 2023, doi: 10.46229/jifotech.v3i1.667.
- [23] A. S. Putra and K. Haryono, "Implementasi Object Oriented Metodologi dan UML pada Pengembangan Sistem Informasi Keuangan Organisasi".
- [24] M. Aman and Suroso, "Pengembangan Sistem Informasi Wedding Organizer Menggunakan Pendekatan Sistem Berorientasi Objek Pada CV Pesta," *J. Janitra Inform. Dan Sist. Inf.*, vol. 1, no. 1, pp. 47-60, Apr. 2021, doi: 10.25008/janitra.v1i1.119.
- [25] M. Syarif, "Waterfall Sebagai Model Pengembangan Sistem Persediaan Apotek Berorientasi Objek," 2022.