

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.1.1 Sistem

Menurut Anggraeni (2017) menjelaskan bahwa sistem mengacu pada sekelompok entitas yang saling terkoneksi, bekerja satu sama lain untuk kepentingan tertentu yang ditetapkan sebelumnya.

Menurut Widarti et al (2024), konsep sistem mengacu pada sebuah entitas yang didalamnya terdapat elemen atau komponen yang saling berkomunikasi dan terkoneksi secara berkelanjutan dalam lingkungan sekitarnya yang berfungsi untuk meraih tercapainya suatu tujuan.

Berdasarkan uraian tersebut, kesimpulannya bahwa sistem merupakan sekelompok elemen maupun entitas yang bersatu menjadi sebuah kesatuan yang saling terhubung dan berinteraksi secara harmonis untuk tercapainya tujuan tertentu.

2.1.2 Informasi

Elisabet (2017) berpendapat bahwa yang dimaksud dari informasi yaitu kumpulan data yang telah diproses sehingga memiliki nilai dan menjadi berguna dan bermakna bagi penerima informasi, sehingga dapat membantu penerima informasi dalam pengambilan keputusan.

Menurut Athoillah dan Putri (2023), Informasi dapat diartikan sebagai data yang telah diolah atau diinterpretasikan dengan cara tertentu, sehingga hasil dari olahan tersebut memiliki nilai dan makna khususnya bagi penerima informasi.

Edhy Sutanta (2011) menjelaskan informasi merupakan hasil akhir dari proses pengolahan data yang berubah sehingga menjadi bermakna bagi penerimanya. Informasi ini bermanfaat untuk pondasi dalam pengambilan keputusan, informasi dapat memberikan efek positif baik secara langsung pada saat itu maupun di masa mendatang.

Berdasarkan uraian tersebut, kesimpulannya bahwa informasi dipahami sebagai kumpulan data yang sudah melalui proses sehingga memiliki nilai dan kegunaan bagi penerima. Informasi berperan penting dalam mengurangi ketidakpastian saat mengambil keputusan di situasi tertentu.

2.1.3 Sistem Informasi

Elisabet (2017) mengatakan bahwa pengertian dari Sistem Informasi adalah alat yang penting bagi perusahaan atau organisasi untuk mencapai tujuannya. Dengan sistem informasi yang baik, perusahaan atau organisasi dapat beroperasi dengan lebih efisien, efektif, dan mencapai tujuan jangka panjangnya.

Menurut Fendi (2020), sistem informasi bagaikan sebuah alat canggih yang dirancang untuk mengolah data menjadi informasi berharga. Informasi ini kemudian dimanfaatkan untuk membantu dalam pengambilan keputusan sehingga dapat lebih cepat, akurat dan efektif.

Berdasarkan uraian tersebut, kesimpulannya Sistem Informasi adalah komponen penting dalam organisasi yang berfungsi untuk mengintegrasikan berbagai kebutuhan yang mendukung proses operasional, manajemen, dan strategi organisasi. Sistem ini membantu organisasi dalam mencapai tujuannya dengan lebih efektif dan efisien.

2.1.4 Basis Data

Basis Data merupakan wadah atau media di mana data itu dapat dikelola yang terstruktur dan terorganisir untuk menampung informasi tentang berbagai hal yang ada di dunia nyata. Istilah "basis" dalam kata ini merujuk pada pusat atau tempat penyimpanan informasi, sedangkan "data" mengacu pada informasi itu sendiri. (Rachmadi, 2020).

Database merupakan segolongan data yang tersimpan dan terorganisir serta saling terhubung yang memungkinkan penggunaan ulang dengan mudah dan cepat. Data dalam *database* disimpan dalam berbagai format, seperti *file*, tabel, atau arsip, dan dihubungkan satu sama lain untuk membentuk struktur yang terstruktur. Tujuan utama database adalah untuk mempermudah pengelolaan data, termasuk pengaturan, pengelompokan, dan pengorganisasian data sesuai dengan kebutuhan. (Rachmadi, 2023).

Berdasarkan penjelasan tersebut dapat diambil kesimpulan bahwa Basis data merupakan gabungan antara pusat penyimpanan (basis) dan informasi (data) yang merekam fakta-fakta dari dunia nyata. Fakta-fakta ini dapat berupa berbagai hal, seperti manusia, hewan, kendaraan, dan lain sebagainya. Informasi tersebut disimpan dalam berbagai format, seperti huruf, angka, gambar, teks, atau bunyi, dan dihubungkan satu sama lain untuk membentuk struktur yang terorganisir.

2.1.5 *Application Programming Interface (API)*

Kurniawan et al (2023) berpendapat bahwa penjelasan dari API (*Application Programming Interface*) adalah sebuah dokumentasi yang meliputi antarmuka, fungsi, kelas, dan struktur yang digunakan untuk mengembangkan perangkat lunak. Konsep dari API adalah menyediakan fitur antarmuka pemrograman aplikasi yang memungkinkan pengguna lain untuk mengakses dan memanfaatkan aplikasi tanpa perlu mengubah struktur kode dasar atau basis data sistem. API juga memungkinkan komunikasi antara sistem-sistem yang berbeda meskipun memiliki perbedaan teknis.

Budiman (2022) mengatakan bahwa API merupakan sistem komunikasi yang memungkinkan interaksi antara dua atau lebih program komputer. Ini adalah bentuk antarmuka perangkat lunak yang memberikan layanan kepada program-program lain.

Dari penjelasan tersebut dapat diambil kesimpulannya bahwa API (*Application Programming Interface*) adalah sebuah sistem komunikasi yang memfasilitasi interaksi antara program-program komputer. API menyediakan antarmuka pemrograman aplikasi yang memungkinkan pengguna untuk mengakses dan memanfaatkan aplikasi tanpa perlu mengubah struktur kode dasar atau basis data sistem. Hal ini memungkinkan komunikasi antara berbagai sistem yang berbeda, meskipun memiliki perbedaan teknis.

2.1.6 *Pengertian Payment Gateway*

Menurut Baez (2022), *Payment Gateway* adalah layanan yang memproses pembayaran *online* dengan aman dengan menghubungkan toko atau platform perdagangan elektronik dengan rekening bank atau lembaga keuangan lainnya.

Menurut Yusun et al (2022), *Payment Gateway* berawal dari kata *payment* yang berarti pembayaran dan *gateway* yang berarti gerbang. Apabila dibangun, *Payment Gateway* dapat diartikan sebagai gerbang pembayaran. *Payment Gateway* merupakan sebuah media bertransaksi. Media ini umumnya disediakan pada *E-Commerce* untuk menghadirkan sistem pengecekan pembayaran yang lebih mudah secara *real-time*.

Dari penjelasan tersebut, dapat disimpulkan bahwa *Payment Gateway* adalah layanan yang memproses pembayaran secara *online* dengan aman, menghubungkan platform perdagangan elektronik atau toko *online* dengan rekening bank atau institusi keuangan lainnya. Istilah "*Payment Gateway*" merupakan gabungan dari

kata "*Payment*" yang artinya pembayaran, dan "*Gateway*" yang artinya gerbang. Jadi, *Payment Gateway* dapat diartikan sebagai gerbang pembayaran. Fungsinya adalah menyediakan sarana untuk melakukan transaksi, terutama dalam konteks *platform E-Commerce*, sehingga memfasilitasi pengecekan pembayaran secara langsung.

2.1.7 *System Development Life Cycle (SDLC)*

Menurut Dennis et al (2015), SDLC merupakan serangkaian tahapan terstruktur yang bertujuan agar dapat memastikan bahwa Sistem Informasi (SI) yang dikembangkan sesuai dengan kebutuhan dan perencanaan bisnis. SDLC ini memiliki empat tahapan yang diantaranya yaitu:

a. Perencanaan (*Planning*)

Fase perencanaan adalah fase awal yang krusial dalam proses pengembangan. Pada tahap ini, fondasi pemahaman tentang tujuan perusahaan dalam membangun sistem informasi dibangun dengan kuat. Hal ini dilakukan dengan mengambil langkah-langkah terstruktur untuk mengendalikan proyek melalui fase SDLC (Dennis et al., 2020).

b. Analisis (*Analysis*)

Fase analisis dan evaluasi sistem juga merupakan langkah penting dalam pengembangan sistem informasi. Pada tahap ini, sistem yang sedang digunakan dipelajari dan dianalisis secara mendalam. Tujuannya adalah untuk memahami sistem secara menyeluruh dan mengidentifikasi kelemahan dan kekurangannya (Dennis et al., 2020).

c. Desain (*Design*)

Fase ini merupakan fase di mana proses perancangan sistem dilakukan, perancangan berfokus pada spesifikasi rancangan yang sudah ditentukan di awal. Hasil akhir pada tahap ini dapat berupa basis data, analisis masukan dan analisis keluaran (Dennis et al., 2020).

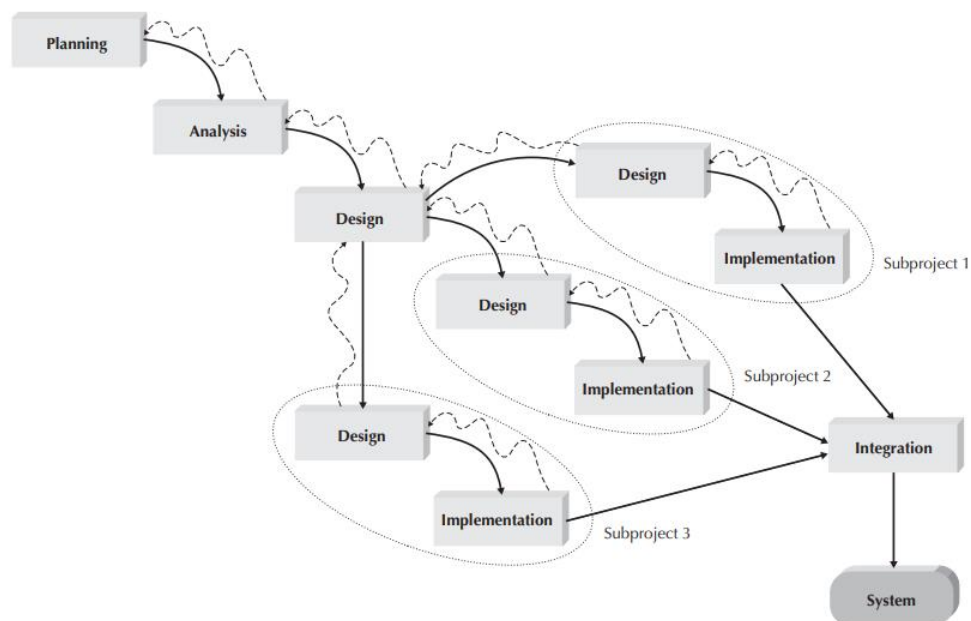
d. Implementasi (*Implementation*)

Fase ini merupakan fase yang dimulai dari pembuatan kode sistem yang dapat dioperasikan pada komputer nantinya, pembuatan kode ini menggunakan Bahasa pemrograman yang sudah ditentukan sebelumnya. Tahap ini terdiri dari beberapa langkah, pertama yaitu pembuatan sistem dengan menggunakan bahasa pemrograman lalu diselingi dengan pengujian yang dilakukan bersamaan agar tidak ada terjadinya *error* atau *bugs*.

Selanjutnya yaitu dilanjutkan dengan dilibatkannya pengguna dalam pelatihan agar pengguna dapat memahami cara menggunakan sistem (Dennis et al., 2020).

2.1.8 *Rapid Application Development (RAD)*

Menurut Dennis et al (2015), RAD merupakan pendekatan pengembangan sistem informasi (SI) yang terintegrasi pada SDLC (*Software Development Life Cycle*). RAD berfokus pada kecepatan dalam membangun aplikasi dengan menekankan pada proses analisis, desain, dan implementasi yang singkat.



Gambar 2.1 Metode Pendekatan RAD (*Rapid Application Development*)
(Sumber: Dennis, Wixom dan Tegarden, 2020)

2.1.9 *Object Oriented Analysis and Design (OOAD)*

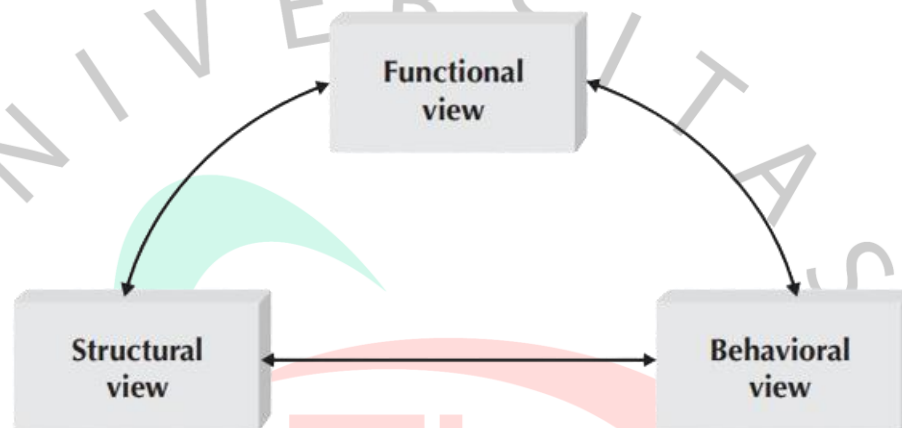
Menurut Dennis et al (2015), OOAD adalah suatu pendekatan dalam pengembangan sistem informasi yang memberikan penekanan pada pemahaman dan perancangan sistem dengan mempertimbangkan objek sebagai elemen utama.

Menurut Valacich dan George (2015), OOAD merupakan metode pendekatan sistem yang inovatif. Pendekatan ini dapat mengintegrasikan data dan aktivitas ke dalam entitas tunggal yang dikenal sebagai objek.

Berdasarkan uraian yang telah dijelaskan, dapat disimpulkan bahwa OOAD adalah pendekatan dalam pengembangan sistem informasi yang fokus pada pemahaman dan perancangan sistem dengan mempertimbangkan objek sebagai elemen sentral. Pendekatan ini dapat menghubungkan data dan aktivitas ke dalam suatu entitas yang dikenal sebagai objek.

2.1.10 Unified Modeling Language (UML)

Dennis et al (2015) menjelaskan bahwa penjelasan dari UML sendiri adalah bahasa yang pada umumnya digunakan oleh banyak industri dalam menjelaskan tentang persyaratan, perancangan arsitektur dan analisis dalam konteks pemrograman OOP (*Object Oriented Programming*) berorientasi objek. UML dapat menjadi representasi untuk menggambarkan model dari setiap aspek dalam pengembangan sistem, dimulai dari analisis sampai implementasi.



Gambar 2.2 Jenis-jenis Unified Modelling Language (UML)
(Sumber: Dennis, Wixom dan Tegarden, 2020)


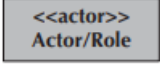

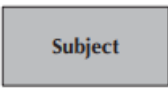

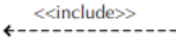
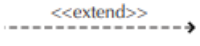
A. Model Fungsional (*Functional Model*)

Model fungsional memiliki peran penting dalam menggambarkan bagaimana sistem informasi akan berinteraksi dengan lingkungannya atau proses bisnis yang terlibat. Model ini memberikan gambaran yang jelas tentang fungsi utama sistem informasi dan bagaimana sistem tersebut akan digunakan oleh para pengguna.

- **Use Case Diagram**

Diagram ini berisi gambaran visual yang menggambarkan dasar dari bagaimana cara sistem berinteraksi dengan para aktor yang terlibat. Diagram ini membantu memvisualisasikan hubungan antara sistem dan para aktor, serta aktivitas yang dilakukan oleh para aktor dalam berinteraksi dengan sistem (Dennis et al, 2015). Berikut dibawah ini merupakan pemaparan dari gambaran setiap notasi yang ada dalam diagram use case.

Tabel 2.1 Simbol dan Notasi Use Case Diagram

Simbol	Notasi	Keterangan
 Actor/Role 	Actor	Notasi aktor ini dapat berupa orang atau sistem yang memiliki dampak dan berada diluar subjek.
 Use Case	Use Case	Notasi ini mempresentasikan bagian utama dari fungsionalitas yang ada pada sistem.
 Subject	Subject Boundary	Notasi <i>subject</i> ini mewakili adanya ruang lingkup subjek, contohnya yaitu proses bisnis atau sistem.
	Assosiation Relationship	Notasi ini berfungsi sebagai jembatan atau penghubung antara aktor dengan use case yang sedang berinteraksi.
	Include Relationship	Notasi ini berfungsi sebagai penghuban use case fungsionalitas dengan use case yang lain.
	Extend Relationship	Berfungsi untuk perluasan use case secara opsional dan menyertakan use case lain.

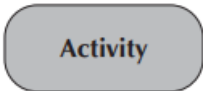
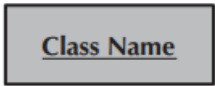



↑	Generalization Relationship	Notasi bagian ini berfungsi untuk generalisasikan aktor agar menjadi lebih spesifik.
---	--	--


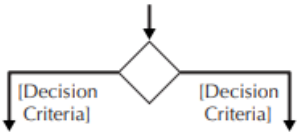
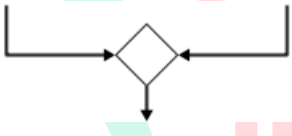
Sumber: (Dennis, Wixom dan Tegarden, 2015)

- **Activity Diagram**

Diagram aktivitas pada umumnya dimanfaatkan untuk memvisualisasikan tindakan atau aktivitas pada proses bisnis. Diagram ini memvisualisasikan urutan langkah-langkah yang dilakukan dalam suatu proses, serta hubungan antara aktivitas tertentu dengan proses secara keseluruhan (Dennis et al., 2015).

Tabel 2.2 Simbol dan Notasi Activity Diagram

Simbol	Notasi	Keterangan
	<i>Activity</i>	Menggambarkan adanya suatu aksi atau tindakan pada proses bisnis.
	<i>Object</i>	Berfungsi untuk menggambarkan objek yang terhubung kepada aliran objek lainnya.
	<i>Control Flow</i>	Mempresentasikan urutan proses yang dieksekusi.
	<i>Object Flow</i>	Notasi ini mempresentasikan aliran yang ada pada suatu objek pada suatu aktivitas ke aktivitas yang lain.
	<i>Initial Node</i>	Notasi ini menandakan awal

		dari diagram aktivitas.
	Final Activity Node	Notasi yang berfungsi untuk mengakhiri proses dari suatu serangkaian aktivitas.
	Decision Node	Notasi yang berfungsi untuk menandakan adanya suatu kondisi dalam aktivitas.
	Merge Node	Berfungsi untuk menghubungkan kembali berbagai jalur keputusan yang telah dibuat dengan menggunakan simpul keputusan.

Sumber: (Dennis, Wixom dan Tegarden, 2015)

B. Model Struktural (*Structural Model*)

Model struktural dalam Analisis dan Desain Berorientasi Objek (OOAD) menggambarkan organisasi dan interaksi antar objek yang terlibat dalam proses bisnis suatu perusahaan atau organisasi. Model ini memvisualisasikan struktur kelas atau objek dalam sistem beserta hubungan antar mereka.

- **Class Diagram**

Diagram ini merupakan visual yang menunjukkan adanya klasifikasi pada objek dan antar kelas yang konsisten yang ada pada sistem. *Class Diagram* ini mendefinisikan kelas beserta perilakunya, termasuk metode dan operasi (Dennis et al., 2015).

Tabel 2.3 Simbol dan Notasi Class Diagram

Simbol	Notasi	Keterangan
--------	--------	------------

	Class	Notasi ini menandakan adanya kelas atau objek yang ada dalam sistem.
<p>attribute name /derived attribute name</p>	Attribute	Notasi ini merepresentasikan properti pada suatu objek.
<p>operation name ()</p>	Operation	Notasi menggambarkan fungsi atau tindakan yang ada dalam objek.
	Association	Notasi yang menggambarkan hubungan antar kelas.
	Generalization	Notasi yang menunjukkan hubungan hierarki antara kelas-kelas.
	Aggregation	Notasi yang menggambarkan hubungan “whole-part” atau “has-a” antara keduanya.
	Composition	Notasi ini menunjukkan hubungan antara kelas di mana satu kelas adalah bagian integral dari kelas

lainnya.

Sumber: (Dennis, Wixom dan Tegarden, 2015)


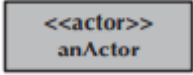


C. Model Prilaku (*Behavioral Model*)


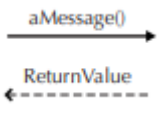


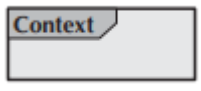
Diagram perilaku atau Behavioral UML adalah jenis diagram yang berfokus pada tingkah laku dan interaksi antara objek dalam sistem. Diagram ini membantu dalam memodelkan aspek dinamis dari sistem, termasuk cara objek berinteraksi dalam berbagai situasi.

- **Sequence Diagram**

Diagram interaksi merepresentasikan secara visual terhadap sekumpulan objek yang terlibat pada diagram use case dan pesan-pesan yang dikirim antar objek selama proses sedang berlangsung. Diagram ini merupakan model dinamis yang mengilustrasikan urutan pesan yang dikirim antara objek-objek selama interaksi. (Dennis et al., 2015).

Tabel 2.4 Simbol dan Notasi Class Diagram

Simbol	Notasi	Keterangan
 anActor	Actor	Notasi ini menunjukkan aktor yang terlibat dalam sistem, aktor ini bisa berupa orang atau sistem.
		
	Object	Notasi yang menunjukkan suatu entitas atau <i>instance</i> dari kelas tertentu yang berpartisipasi dalam interaksi.
	Lifeline	Notasi yang menunjukkan aliran dari proses dalam suatu urutan.

	Execution Occurrence	Notasi yang merujuk pada representasi visual dari eksekusi atau jalannya suatu aksi.
	Message	Notasi yang berfungsi untuk memberikan pesan dalam suatu interaksi.
	Guard Condition	Notasi yang menggambarkan kondisi tertentu Dimana kondisi tersebut harus terpenuhi.
	Object Destruction	Notasi yang mengindikasikan bahwa objek tersebut akan menghentikan keberadaannya dan ditempatkan di ujung garis hidup suatu objek.
	Frame	Notasi yang menggambarkan diagram konteks urutan.

Sumber: (Dennis, Wixom dan Tegarden, 2015)

2.1.11 Black Box Testing

Desikan dan Ramesh (2006), menjelaskan bahwa yang dimaksud dengan *Black-box Testing* ialah suatu metode yang digunakan untuk pengujian pada *software* yang dijalankan tanpa terlalu memperhatikan pada struktur internal atau implementasi kode program. Penguji hanya fokus pada spesifikasi eksternal sistem dan tidak memiliki pengetahuan tentang bagaimana *software* mencapai hasil tertentu. *Black-box Testing* menguji fungsionalitas dan kinerja *software* dari perspektif pengguna atau pelanggan, menggunakan input dan mengamati output yang dihasilkan.

2.2 Tinjauan Studi

Tinjauan studi ini dilakukan guna sebagai bahan untuk mendukung riset penelitian, jurnal yang dipilih memiliki topik yang relevan dengan topik yang dibahas oleh peneliti. Adapun beberapa referensi jurnal yang digunakan untuk literature review sebagai berikut:

1. Abas Sunarya, Andri Cahyo Purnomo, Muhamad Iqbal Wahib (2019). "Penerapan Midtrans *Payment* Pada Official Site Asosiasi Perguruan Tinggi Swasta Indonesia" (jurnal). Penelitian ini mengadopsi berbagai metode analisis seperti SWOT, elisitasi, *flowchart*, UML, dan HIPO. Fokusnya adalah untuk meningkatkan efisiensi proses pendaftaran melalui situs resmi APTISI dengan integrasi formulir *online* dan sistem pembayaran Midtrans, memungkinkan anggota untuk melakukan transaksi keanggotaan kapan saja dan di mana saja selama terhubung ke internet. Sistem ini mempermudah pendaftar untuk menyelesaikan proses pendaftaran tanpa perlu masuk ke email terlebih dahulu. Setelah menyelesaikan proses yang diperlukan, pendaftar akan menerima notifikasi email mengenai data mereka, serta konfirmasi resmi pendaftaran beserta email dari APTISI setelah menyelesaikan transaksi pembayaran dan mengirimkan formulir pembaruan data anggota.
2. Siti Zahra Salma, Asmawati, dan Adam M Tanniewa (2023) dalam Jurnal Teknologi dan Sistem Informasi (vol. 4, No 2, hal. 127-137) mengembangkan aplikasi konsultasi psikologi *online* dengan menggunakan API Midtrans sebagai *gateway* pembayaran. Penelitian ini dimotivasi oleh kekurangan aplikasi sebelumnya yang menggunakan metode pembayaran transfer bank, yang sering menimbulkan penundaan dalam layanan. Hasil penelitian menunjukkan bahwa sistem baru ini menghilangkan hambatan pembayaran, meningkatkan aksesibilitas layanan konsultasi psikologi. Uji usability sistem menunjukkan nilai rata-rata 85, menandakan tingkat usability yang baik bagi pengguna aplikasi ini.
3. Viktor Handrianus Pranatawijaya dan Hendra Yulianto (2023) dalam jurnal yang berjudul "Penerapan API (*Application Programming Interface*) MIDTRANS Sebagai *Payment Gateway* Pada Indekos Berbasis Website" membahas pemanfaatan teknologi finansial (*fintech*) untuk meningkatkan efisiensi pembayaran sewa indekos melalui gerbang pembayaran *online*

(*Payment Gateway*). Mereka menerapkan metode waterfall yang terdiri dari tahapan analisis, desain, coding, pengujian, dan pemeliharaan dalam penelitian ini. Pengujian dilakukan menggunakan metode *blackbox testing*. Hasil penelitian menunjukkan bahwa integrasi API Midtrans pada website indikos memungkinkan penyewa untuk melakukan pembayaran sewa secara *online* melalui transfer bank atau e-wallet secara *real-time*. Hal ini mengeliminasi kebutuhan untuk konfirmasi pembayaran manual, sehingga mempermudah dan mempercepat proses pembayaran bagi penyewa indikos.

4. Ikbal Fauzi dan Ines Heidiani Ikasari (2023) dengan judul "Rancang Bangun Penerapan Teknologi Aplikasi *Payment Gateway* pada Sistem Pembayaran Berbasis Web (Studi Kasus: Toko Bandar Aki)" mengembangkan solusi untuk permasalahan sistem pembayaran Toko Bandar Aki. Penelitian ini memanfaatkan *payment gateway* dan *sms notification* sebagai teknologi informasi terbaru untuk membangun sistem pembayaran web. Tujuannya adalah membantu bagian keuangan dalam manajemen pembayaran dan memberikan pengingat kepada pelanggan. Hasil penelitian menunjukkan bahwa penerapan teknologi ini dapat mengatasi permasalahan pembayaran, meningkatkan kinerja admin, dan memperbaiki pengalaman pelanggan dengan memudahkan proses pembayaran serta memberikan notifikasi yang membantu mengingatkan pelanggan akan jatuh tempo pembayaran.
5. Ayu Putu Yulia Kusuma Wardani dan Nyoman Ari Surya Darmawan (2020) dengan judul "Peran *Financial Technology* pada UMKM: Peningkatan Literasi Keuangan Berbasis *Payment Gateway*" di Jurnal Ilmiah Akuntansi dan Humanika, Vol. 10 No. 2, Mei – Agustus 2020, mengkaji peran teknologi finansial bagi UMKM di Kecamatan Buleleng dalam meningkatkan literasi keuangan dengan menggunakan *payment gateway*. Penelitian ini menggunakan metode kualitatif dengan pendekatan deskriptif, mengumpulkan data primer dan sekunder melalui wawancara, observasi, dan dokumentasi. Hasilnya menunjukkan bahwa penerapan teknologi finansial membantu meningkatkan literasi keuangan pelaku UMKM dengan mempermudah pengelolaan keuangan dan transaksi digital yang tercatat otomatis, sehingga memudahkan pengelolaan pemasukan.

6. Nurul Fauziah, Fajar Darmawan, dan Wanda Gusdya (2022) dengan judul "Integrasi *Payment Gateway* Untuk Donasi Menggunakan *Framework Flutter* (Studi Kasus Donasi HMTIF UNPAS)" di Jurnal Pasinformatik, Volume 1, Issue 1, Januari 2022, mengembangkan aplikasi donasi berbasis *mobile* yang terintegrasi dengan *payment gateway*. Penelitian ini menggunakan metodologi yang dimulai dari identifikasi masalah, pengumpulan data melalui kuesioner dan studi literatur, analisis, perancangan berbasis teknologi yang relevan, hingga implementasi dengan menggunakan *Framework Flutter*. Hasilnya adalah penciptaan aplikasi donasi yang memudahkan pembayaran dan transaksi *online*.
7. Muhammad Al Biruni, Ahmad Faisol, dan Nurlaily Vendyansyah (2023) dengan judul "Penerapan REST API Dan Integrasi Midtrans Sebagai *Payment Gateway* pada Platform Pelatihan *Online* (Studi Kasus: Bisabacakitab)" di Jurnal Mahasiswa Teknik Informatika, Vol. 7 No. 4, Agustus 2023, mengimplementasikan API dari Midtrans sebagai *payment gateway* untuk semua transaksi pada platform pelatihan *online* Bisabacakitab. Sistem ini dibangun menggunakan REST API dengan *framework* Laravel untuk mempermudah integrasi dengan Midtrans. Hasil pengujian menggunakan *Blackbox Testing* menunjukkan bahwa sistem dapat mengelola transaksi pembayaran, memonitor status pesanan, dan memberikan akses pengguna setelah pembayaran berhasil dengan tingkat keberhasilan 100%. Dengan demikian, masalah sistem Bisabacakitab berhasil diatasi dengan bantuan API dari Midtrans. Studi ini menunjukkan bahwa aplikasi Bisabacakitab berhasil memenuhi kebutuhan pelanggan, dengan hasil kuesioner menunjukkan kepuasan sebesar 89% dari 55 responden, termasuk pengguna lama Bisabacakitab.