

**PERANCANGAN SISTEM INFORMASI PENGELOLAAN
TAGIHAN DAN PEMBAYARAN ADMINISTRASI KEUANGAN
SISWA SMAN 15 SURABAYA BERBASIS WEB (PXP)**

***DESIGN OF A WEB-BASED INFORMATION SYSTEM
FOR STUDENT BILLING AND FINANCIAL ADMINISTRATION
PAYMENTS AT SMAN 15 SURABAYA (PXP)***

**Proposal Tugas Akhir
Mata Kuliah Tugas Akhir (CBK4BAA4)**

**Disusun oleh:
Ahmad Dandy Hasananta
1202220430**



**Universitas
Telkom**

**PROGRAM STUDI SARJANA TEKNOLOGI INFORMASI
DIREKTORAT KAMPUS SURABAYA
UNIVERSITAS TELKOM
SURABAYA
2026**

LEMBAR PENGESAHAN
TUGAS AKHIR
PERANCANGAN SISTEM INFORMASI PENGELOLAAN
TAGIHAN DAN PEMBAYARAN ADMINISTRASI KEUANGAN
SISWA SMAN 15 SURABAYA BERBASIS WEB (PXP)

Telah disetujui dan disahkan untuk memenuhi sebagian syarat memperoleh gelar
Sarjana pada Program Studi S1 Teknologi Informasi
Direktorat Kampus Surabaya
Universitas Telkom

Disusun oleh:
AHMAD DANDY HASANANTA
1202220430

Surabaya, 7 Januari 2026
Menyetujui,

1.
Philip Tobianto Daely, S.T., M.Eng., Ph.D. (Pembimbing I)
NIP. 19940002
2.
Mastuty Ayu Ningtyas, S.Kom., M.MT. (Pembimbing II)
NIP. 19940002
3.
Rizky Aditya Pratama, S.Kom., M.Kom., Ph.D. (Penguji I)
NIP. 1973XXXX
4.
Maria Indah Lestari, S.T., M.Eng. (Penguji II)
NIP. 1984XXXX

Kaprodi S1 Teknologi Informasi,

Muhammad Adib Kamali, S.T., M.Eng.
NIP. 22970007

LEMBAR PERNYATAAN ORISINALITAS

**FOTO FORMAL
4X6
BERWARNA**

Nama : Ahmad Dandy Hasananta
NIM : 1202220430
Alamat : Jl. Pagesangan Baru II No.36, Pagesangan, Kec. Jambangan, Surabaya, Jawa Timur 60233
No. Telp : 081377286361
Email : ahmaddandy@student.telkomuniversity.ac.id

Menyatakan bahwa Tugas Akhir ini merupakan karya orisinal saya sendiri. Atas pernyataan ini, saya siap menanggung risiko atau sanksi yang dijatuhkan kepada saya apabila kemudian ditemukan adanya pelanggaran terhadap kejujuran akademik atau etika keilmuan dalam karya ini, atau ditemukan bukti yang menunjukkan ketidakaslian karya ini.

Surabaya, 7 Januari 2026

Ahmad Dandy Hasananta

ABSTRAK

Sistem pengelolaan penagihan dan pembayaran administrasi siswa merupakan bagian penting dalam administrasi keuangan sekolah. Di SMAN 15 Surabaya, proses penagihan dan pencatatan pembayaran masih dilakukan secara konvensional, sehingga menimbulkan permasalahan seperti pencatatan berulang, kesulitan memantau status pembayaran, serta keterbatasan akses informasi bagi orang tua siswa. Penelitian ini bertujuan untuk merancang dan membangun sistem informasi pengelolaan tagihan dan pembayaran administrasi siswa berbasis web yang mampu mengelola berbagai jenis tagihan secara terpusat, mulai dari penetapan tarif oleh bendahara, pembuatan tagihan per siswa oleh Tata Usaha, hingga penyajian informasi tagihan dan riwayat pembayaran bagi orang tua. Metode pengembangan perangkat lunak yang digunakan adalah *Personal eXtreme Programming* (PXP), yaitu pendekatan pengembangan berbasis *agile* yang dirancang untuk pengembang tunggal dengan proses iteratif dan adaptif terhadap perubahan kebutuhan pengguna. Perancangan sistem dimodelkan menggunakan *Unified Modeling Language* (UML) dan *Entity-Relationship Diagram* (ERD), sedangkan implementasi sistem direncanakan menggunakan *framework* Laravel dan basis data MySQL. Pengujian sistem dilakukan melalui *white-box testing* dalam bentuk *unit testing* untuk memverifikasi logika dan struktur kode program, serta *black-box testing* melalui *User Acceptance Testing* (UAT) untuk memastikan kesesuaian fungsionalitas sistem dengan kebutuhan pengguna. Evaluasi penerimaan sistem dilakukan menggunakan kuesioner berbasis skala Likert sebagai bagian dari UAT.

Kata kunci: sistem informasi, penagihan, pembayaran administrasi, sekolah, *Personal eXtreme Programming*.

ABSTRACT

Student billing and administrative payment management systems are an important component of school financial administration. At SMAN 15 Surabaya, the billing and payment recording processes are still carried out conventionally, resulting in several issues such as repetitive data entry, difficulties in monitoring payment status, and limited access to information for students' parents. This study aims to design and develop a web-based information system for managing student billing and administrative payments that is capable of handling various types of charges in a centralized manner, ranging from tariff determination by the treasurer, bill generation per student by administrative staff, to the presentation of billing information and payment history for parents. The software development method employed in this study is Personal eXtreme Programming (PXP), an agile-based development approach designed for single developers with an iterative process that is adaptive to changes in user requirements. The system design is modeled using Unified Modeling Language (UML) and Entity-Relationship Diagram (ERD), while the system implementation is planned using the Laravel framework and a MySQL database. System testing is conducted through white-box testing in the form of unit testing to verify the program logic and code structure, as well as black-box testing through User Acceptance Testing (UAT) to ensure that the system's functionality meets user requirements. The evaluation of system acceptance is carried out using a Likert-scale-based questionnaire as part of the UAT process.

Keywords: information system, billing, student administration payment, school, Personal eXtreme Programming.

KATA PENGANTAR

Tuliskan Kata Pengantar Laporan TA anda di sini.

DAFTAR ISI

Lembar Pengesahan	ii
Lembar Pernyataan Orisinalitas	iii
Abstrak	iv
Abstract	v
Kata Pengantar	vi
Daftar Isi	viii
Daftar Gambar	ix
Daftar Tabel	x
1 Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Manfaat Penelitian	3
1.4.1 Manfaat Akademik / Keilmuan	3
1.4.2 Manfaat Praktis	3
1.5 Batasan Penelitian	3
1.6 Sistematika Penulisan	5
2 Tinjauan Pustaka	7
2.1 Kajian Teori	7
2.2 Penelitian Terkait	7
2.2.1 Sistem Informasi	10
2.2.2 Sistem Pembayaran	10
2.2.3 Pembayaran Kanal	11
2.2.4 Metode Pengembangan Model <i>Personal eXtreme Programming</i>	11
2.2.5 Unified Modeling Language	12
2.2.6 Black-Box Testing	13
2.2.7 White-Box Testing	13
2.2.8 Instrumen Penelitian Kuantitatif	13

3 Metodologi dan Rancangan Penelitian	14
3.1 Metodologi Penelitian	14
3.1.1 Requirements (Kebutuhan)	14
3.1.2 Planning (Perencanaan)	17
3.1.3 Iteration Initialization (Inisialisasi Iterasi)	18
3.1.4 Design (Desain)	18
3.1.5 Implementation (Implementasi)	19
3.1.6 System Testing (Pengujian Sistem)	20
3.1.7 Retrospective (Retrospektif)	21
3.2 Alat dan Bahan yang Digunakan	21
3.3 Perancangan Sistem Awal	22
3.3.1 <i>Use Case</i>	22
3.3.2 <i>Entity-Relationship Diagram</i>	24
3.3.3 <i>Activity Diagram</i>	26
3.3.4 <i>Mockup</i>	31
3.4 Pengujian Sistem (<i>Black-Box</i>)	37
3.5 Evaluasi/Pengukuran (Kuesioner Likert)	39
3.6 Jadwal Pelaksanaan	41
4 Hasil dan Analisis	42
5 Kesimpulan dan Saran	43
5.1 Kesimpulan	43
5.2 Saran untuk Penelitian Selanjutnya	43
Daftar Pustaka	44
Lampiran	46
A Simbol Use Case Diagram	47
B Simbol Entity-Relationship Diagram	48
C Simbol Activity Diagram	49

DAFTAR GAMBAR

3.1	Alur metode <i>Personal eXtreme Programming</i>	14
3.2	<i>Use Case Diagram</i>	22
3.3	<i>Entity-Relationship Diagram</i>	24
3.4	<i>Activity Diagram</i> Bendahara	26
3.5	<i>Activity Diagram</i> Tata Usaha	28
3.6	<i>Activity Diagram</i> Orang Tua	30
3.7	Desain <i>Landing Page</i>	31
3.8	Desain Beranda Pada Tata Usaha	32
3.9	Desain Data Pada Siswa Tata Usaha	32
3.10	Desain Data Pada Kelas Tata Usaha	33
3.11	Desain Tagihan Pada Tata Usaha	33
3.12	Desain Laporan Pada Tata Usaha	34
3.13	Desain Riwayat Penolakan Pada Tata Usaha	34
3.14	Desain Beranda Pada Bendahara	35
3.15	Desain Membuat Harga Pada Bendahara	35
3.16	Desain Laporan Pada Bendahara	36
3.17	Desain Beranda Pada Orang Tua	36
3.18	Desain Tagihan Pada Orang Tua	37

DAFTAR TABEL

2.1	Tabel Penelitian terdahulu	7
3.1	Kebutuhan Fungsional Sistem dalam Bentuk User Stories (PXP)	15
3.2	Kebutuhan Fungsional dan Non-Fungsional Sistem SIPANSE	18
3.3	Perangkat Keras (<i>Hardware</i>)	21
3.4	Perangkat Lunak (<i>Software</i>)	22
3.5	Contoh <i>Test Case</i> Pengujian Fungsional dengan Metode <i>Black-Box</i>	37
3.6	Kuesioner Evaluasi Sistem (Responden: TU dan Bendahara)	39
3.7	Ini adalah contoh tabel.	41
A.1	Simbol <i>Use Case Diagram</i>	47
B.1	Simbol <i>Entity-Relationship Diagram</i>	48
C.1	Simbol <i>Activity Diagram</i>	49

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Sistem pembayaran merupakan salah satu komponen penting dalam pengelolaan administrasi keuangan pada lembaga pendidikan. Sekolah tidak hanya menyelenggarakan kegiatan belajar mengajar, tetapi juga memiliki tanggung jawab dalam mengelola berbagai kewajiban administrasi siswa, seperti pembayaran SPP, iuran kegiatan, pembelian buku, maupun biaya lain yang berkaitan dengan operasional sekolah. Pengelolaan penagihan dan pembayaran yang tidak tertata dengan baik dapat menimbulkan beban tambahan bagi pihak sekolah, baik dari sisi administrasi maupun pengawasan keuangan.

Pada banyak sekolah, mekanisme penagihan dan pembayaran administrasi siswa masih dilakukan secara konvensional. Informasi tagihan umumnya disampaikan melalui pengumuman tertulis atau perantara siswa, sedangkan pembayaran dilakukan secara tunai atau melalui transfer biasa tanpa dukungan sistem informasi yang terintegrasi. Kondisi ini menyebabkan pihak Tata Usaha (TU) dan bendahara harus melakukan pencatatan serta pengecekan data pembayaran secara berulang, sehingga berpotensi menimbulkan kesalahan pencatatan, keterlambatan pembaruan status pembayaran, serta kesulitan dalam menyusun laporan keuangan secara akurat.

Selain itu, keterbatasan sistem juga berdampak pada pihak orang tua siswa. Orang tua sering kali mengalami kesulitan dalam memperoleh informasi yang jelas dan terkini mengenai jenis tagihan, besaran nominal, jadwal pembayaran, serta status pembayaran anaknya. Hal ini dapat menimbulkan kesalahpahaman antara pihak sekolah dan orang tua, khususnya terkait apakah suatu tagihan telah dibayarkan atau masih memiliki tunggakan.

SMAN 15 Surabaya saat ini masih menerapkan mekanisme penagihan dan pembayaran administrasi siswa secara konvensional. Proses penyampaian informasi tagihan dan pencatatan pembayaran dilakukan melalui jalur yang terpisah, sehingga menyulitkan pemantauan status pembayaran secara menyeluruh. Pihak sekolah memiliki keinginan untuk mulai memanfaatkan sistem berbasis web yang dapat membantu pengelolaan penagihan administrasi siswa sekaligus memudahkan orang tua dalam memantau kewajiban pembayaran. Sistem tersebut diharapkan bersifat fleksibel dan dapat mendukung berbagai metode pembayaran yang digunakan oleh sekolah, tanpa bergantung pada satu metode pembayaran tertentu.

Berdasarkan kondisi tersebut, diperlukan perancangan dan pembangunan sistem

informasi pengelolaan tagihan dan pembayaran administrasi siswa berbasis web di SMAN 15 Surabaya. Pengembangan sistem ini dilakukan menggunakan metode *Personal eXtreme Programming* (PXP), yaitu metode pengembangan perangkat lunak berbasis *agile* yang dirancang untuk pengembang tunggal dengan proses iteratif dan adaptif terhadap perubahan kebutuhan pengguna. Metode ini dipilih karena sesuai dengan kondisi pengembangan yang dilakukan dalam jangka waktu serta pengembang yang terbatas atau individu yang membutuhkan fleksibilitas serta keterlibatan pengguna secara berkelanjutan. Melalui penelitian ini diharapkan dapat dihasilkan sistem informasi yang mampu membantu sekolah dalam mengelola administrasi keuangan siswa secara lebih terstruktur, transparan, dan mudah digunakan oleh seluruh pihak terkait.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan, rumusan masalah dalam penelitian ini adalah sebagai berikut:

1. Bagaimana membangun sistem informasi pengelolaan tagihan dan pembayaran administrasi keuangan siswa pada SMAN 15 Surabaya berbasis web yang mendukung pembuatan tagihan, pencatatan pembayaran, serta laporan sesuai kebutuhan pengguna?
2. Bagaimana menguji dan mengevaluasi sistem yang dibangun untuk memastikan fungsionalitas dan kemudahan penggunaan sistem?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah sebagai berikut:

1. Membangun sistem informasi berbasis web yang membantu pengelolaan tagihan dan pembayaran administrasi keuangan siswa di SMAN 15 Surabaya, sehingga proses pembuatan tagihan, pencatatan pembayaran, dan laporan dapat dilakukan dengan terstruktur sesuai kebutuhan pengguna.
2. Melakukan pengujian dan evaluasi terhadap sistem yang dikembangkan guna memastikan sistem berfungsi dengan baik dan mudah digunakan oleh pengguna terkait.

1.4 Manfaat Penelitian

1.4.1 Manfaat Akademik / Keilmuan

Penelitian ini diharapkan memberikan manfaat akademik sebagai berikut:

1. Menambah referensi mengenai perancangan dan pembangunan sistem informasi penagihan dan pembayaran administrasi siswa berbasis web pada lingkungan sekolah menengah.
2. Menyajikan contoh penerapan analisis kebutuhan, perancangan alur bisnis, dan pemodelan data untuk sistem informasi yang mengelola tagihan per siswa dengan keterlibatan peran bendahara, TU, dan orang tua.
3. Memberikan gambaran penerapan metode *Personal eXtreme Programming* (PXP) dalam pengembangan sistem informasi administrasi sekolah dengan pendekatan iteratif dan adaptif terhadap kebutuhan pengguna.

1.4.2 Manfaat Praktis

Secara praktis, penelitian ini diharapkan memberikan manfaat sebagai berikut:

1. Membantu pihak sekolah, khususnya bendahara dan Tata Usaha (TU), dalam mengelola penagihan dan pembayaran administrasi siswa secara lebih terstruktur melalui satu sistem berbasis web.
2. Memudahkan orang tua dalam mengetahui jenis tagihan yang dimiliki anak, memantau status pembayaran, serta memperoleh informasi administrasi pembayaran yang terdokumentasi dengan baik.
3. Mengurangi beban pekerjaan pencatatan ulang dan pengecekan manual yang sebelumnya dilakukan secara terpisah, sehingga proses pengelolaan administrasi keuangan sekolah dapat berjalan lebih efisien.
4. Menyediakan rancangan sistem yang dapat dijadikan dasar pengembangan lebih lanjut apabila sekolah ingin mengintegrasikan sistem dengan kanal pembayaran yang digunakan di masa mendatang.

1.5 Batasan Penelitian

Penelitian ini memiliki beberapa batasan agar ruang lingkupnya jelas dan dapat diselesaikan dalam waktu pelaksanaan tugas akhir. Batasan-batasan tersebut adalah sebagai berikut.

1. Batasan alat (teknologi).

- Sistem yang dikembangkan berupa aplikasi berbasis web yang dibangun menggunakan *framework* Laravel, bahasa pemrograman PHP, dan basis data MySQL.
- Akses pengguna dilakukan melalui *web browser* pada komputer atau laptop. Pengembangan aplikasi mobile (Android/iOS) tidak termasuk dalam penelitian ini.
- Dukungan kanal pembayaran non-tunai dimodelkan pada tingkat alur sistem dan pencatatan data transaksi. Integrasi teknis langsung dengan dengan sistem perbankan atau penyedia layanan pembayaran eksternal tidak menjadi cakupan penelitian ini dan hanya direpresentasikan secara konseptual dalam perancangan sistem.

2. Batasan data.

- Data yang digunakan berupa data contoh (dummy) dan/atau data yang diberikan oleh SMAN 15 Surabaya untuk keperluan uji coba, seperti data siswa, data kelas, dan contoh data tagihan.
- Data yang digunakan hanya mewakili sebagian kondisi di sekolah dan tidak mencakup seluruh siswa maupun seluruh periode tahun ajaran secara lengkap.
- Penelitian ini tidak membahas proses pengelolaan data dalam skala besar atau integrasi dengan sistem data lain di luar kebutuhan aplikasi yang dikembangkan.

3. Batasan fitur.

- Fitur sistem difokuskan pada pengelolaan tagihan administrasi siswa (seperti SPP, iuran, dan biaya buku), pencatatan pembayaran, penampilan status tagihan, pengiriman notifikasi di dalam sistem, serta pencetakan *invoice*.
- Sistem tidak mencakup transaksi lain di luar administrasi sekolah, misalnya transaksi kantin atau penjualan barang harian.
- Fitur laporan disajikan dalam bentuk tampilan tabel dan ekspor data sederhana ke berkas *spreadsheet*. Analisis laporan yang lebih mendalam dengan grafik atau perhitungan statistik khusus tidak termasuk dalam ruang lingkup penelitian ini.

- Sistem yang dibangun pada penelitian ini belum terintegrasi secara langsung dengan layanan pembayaran eksternal. Pencatatan pembayaran dilakukan di dalam aplikasi sesuai alur yang telah ditentukan, berdasarkan input dari pengguna atau data simulasi yang disiapkan untuk keperluan pengujian.

4. Batasan lingkungan.

- Pengujian sistem dilakukan dalam lingkungan terbatas, misalnya pada server pengembangan atau komputer yang disiapkan untuk keperluan tugas akhir, bukan pada infrastruktur produksi yang digunakan untuk operasional penuh sekolah.
- Uji coba sistem dilakukan dengan skenario penggunaan oleh bendahara, Tata Usaha, dan orang tua dalam skala kecil untuk memastikan fungsi dasar sistem berjalan dengan baik.

5. Batasan waktu.

- Pengembangan dan pengujian sistem dibatasi oleh durasi pelaksanaan tugas akhir, sehingga uji coba jangka panjang dalam beberapa tahun ajaran tidak dilakukan.
- Penyesuaian lebih lanjut terhadap kebijakan internal sekolah yang mungkin muncul setelah masa tugas akhir selesai tidak termasuk dalam ruang lingkup penelitian ini.

1.6 Sistematika Penulisan

Sistematika penulisan laporan tugas akhir ini adalah sebagai berikut:

1. Bab 1 Pendahuluan

Bab ini menjelaskan latar belakang permasalahan, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, serta sistematika penulisan proposal secara keseluruhan.

2. Bab 2 Tinjauan Pustaka

Bab ini memaparkan teori-teori dasar yang menjadi landasan penelitian, meliputi konsep sistem, informasi, sistem informasi, sistem pembayaran, kanal pembayaran, *virtual account*, metode pengembangan perangkat lunak *Waterfall*, Unified Modeling Language (UML), Laravel, MySQL, metode *black-box testing*, serta instrumen penelitian kuantitatif dan skala Likert. Bab ini juga

menyajikan penelitian terdahulu yang relevan dan analisis posisi penelitian yang diusulkan.

3. Bab 3 Metodologi dan Rancangan Penelitian

Bab ini menjelaskan metode pengembangan sistem yang digunakan, tahapan penelitian berdasarkan model *Waterfall*, alat dan bahan yang digunakan, serta perancangan sistem awal yang meliputi *use case diagram*, *Entity-Relationship Diagram*, *activity diagram*, dan rancangan antarmuka (*mockup*). Selain itu, bab ini menampilkan jadwal pelaksanaan penelitian.

4. Daftar Pustaka dan Lampiran

Bagian ini memuat daftar referensi yang digunakan dalam penyusunan proposal serta lampiran yang mendukung, seperti diagram lengkap, rancangan antarmuka, atau dokumen pendukung lainnya.

BAB 2

TINJAUAN PUSTAKA

2.1 Kajian Teori

Bab ini menyajikan teori-teori dan konsep dasar yang digunakan sebagai landasan dalam perancangan Sistem Informasi Penagihan dan Pembayaran Administrasi Siswa SMAN 15 Surabaya berbasis web. Pembahasan meliputi pengertian sistem, informasi, sistem informasi, sistem pembayaran, serta kanal pembayaran yang digunakan dalam pengelolaan administrasi keuangan sekolah. Selain itu, membahas metode pengembangan perangkat lunak *Personal eXtreme Programming* (PXP) sebagai pendekatan pengembangan yang bersifat iteratif dan adaptif, konsep *Unified Modeling Language* (UML) untuk pemodelan sistem. Bab ini juga membahas konsep pengujian perangkat lunak yang meliputi *white-box testing* dan *black-box testing* melalui *User Acceptance Testing* (UAT), serta instrumen penelitian kuantitatif menggunakan skala Likert yang digunakan untuk mengevaluasi tingkat penerimaan pengguna terhadap sistem yang dikembangkan.

2.2 Penelitian Terkait

Berikut beberapa penelitian mengenai sistem pendukung yang relevan dengan penelitian ini.

Tabel 2.1: Tabel Penelitian terdahulu

Judul Penelitian	Masalah	Metode	Hasil / Keterbatasan
Pengembangan Sistem Informasi Pembayaran Sumbangan Pembinaan Pendidikan (SPP) Berbasis Web Menggunakan Codeigniter Studi Kasus SDIT AL-Manar Pekanbaru Hartomi, Afsari, Rahmawati, dan Lendra (2021)	Proses pembayaran SPP di SDIT AL-Manar masih dicatat di buku, laporan sulit disusun, pencarian data SPP lambat, dan belum ada sistem informasi berbasis web yang terintegrasi.	Metode pengembangan perangkat lunak Waterfall; analisis kebutuhan, perancangan, implementasi web berbasis CodeIgniter, dan basis data MySQL; pengujian menggunakan <i>black-box testing</i> .	Menghasilkan sistem informasi pembayaran SPP berbasis web yang membantu petugas keuangan mencatat pembayaran dan tunggakan, mempermudah pencarian data, serta menyajikan informasi pembayaran SPP bulanan kepada orang tua/wali.

Judul Penelitian	Masalah	Metode	Hasil / Keterbatasan
Sistem Informasi Pembayaran Biaya Sekolah Berbasis Website dengan WhatsApp <i>Gateway</i> Irawan, Lianawati, dan Wibowo (2023)	Pengelolaan pembayaran biaya sekolah dan penyampaian informasi ke orang tua masih manual; pencatatan dan pembuatan laporan kurang efisien serta belum ada notifikasi otomatis status pembayaran kepada orang tua.	Pengembangan sistem informasi menggunakan metode SDLC (model <i>waterfall</i>); analisis, desain, implementasi, pengujian; implementasi dengan PHP, MySQL dan <i>framework</i> CodeIgniter; integrasi WhatsApp <i>Gateway</i> untuk pengiriman notifikasi.	Menghasilkan sistem informasi pembayaran biaya sekolah berbasis web yang membantu pencatatan pembayaran, mempermudah pembuatan laporan, serta mengirim notifikasi pembayaran/tagihan kepada orang tua melalui WhatsApp sehingga monitoring pembayaran siswa lebih mudah.
Sistem <i>E-Wallet</i> untuk Pembayaran dalam Lingkup Sekolah Ikhsanudin, Pratama, dan Maulindar (2022)	Diperlukan alternatif pembayaran non-tunai yang praktis untuk berbagai transaksi di lingkungan sekolah (SPP, kantin, koperasi); belum banyak implementasi <i>e-wallet</i> yang spesifik dirancang untuk ekosistem transaksi di sekolah.	Pengembangan aplikasi <i>e-wallet</i> seolah “MBAYAR” dengan metode Waterfall; studi literatur dan observasi; perancangan <i>workflow</i> , <i>activity diagram</i> , <i>use case</i> dan antarmuka; implementasi prototipe <i>e-wallet</i> berbasis Android dengan server <i>backend</i> .	Menghasilkan prototipe aplikasi <i>e-wallet</i> yang mendukung pembayaran SPP, kantin, dan koperasi menggunakan <i>barcode/QR</i> , fitur isi saldo, riwayat transaksi, tabungan, dan pembatasan pengeluaran; dinilai dapat menjadi alternatif pembayaran non-tunai yang sederhana dan mudah dioperasikan di lingkungan sekolah.

Judul Penelitian	Masalah	Metode	Hasil / Keterbatasan
Rancang Bangun Website Manajemen Pembayaran SPP Sekolah Rizki, Jakak, Sari, dan Satriadi (2024)	Di SMK Nurul Huda Buay Madang pembayaran SPP masih manual menggunakan kartu pembayaran dan buku, serta informasi sekolah belum tersedia secara <i>online</i> sehingga proses administrasi dan akses informasi menjadi lambat dan kurang efektif.	Metode SDLC dengan model Waterfall: analisis kebutuhan pengguna, perancangan struktur navigasi dan UML (<i>use case, activity, class diagram, database</i>), implementasi website menggunakan PHP, Bootstrap, dan MySQL; pengujian fungsional dengan <i>black-box</i> dan uji portabilitas di berbagai browser.	Menghasilkan website informasi dan pembayaran SPP yang dapat diakses siswa dan orang tua; admin dapat mengelola data siswa, guru, wali kelas, dan pembayaran SPP. Pengujian menunjukkan sistem berjalan sesuai kebutuhan dan membantu efisiensi administrasi pembayaran di sekolah.

Berdasarkan analisis terhadap penelitian-penelitian terdahulu, dapat disimpulkan bahwa sebagian besar penelitian berfokus pada pengembangan sistem informasi pembayaran administrasi sekolah yang bertujuan untuk menggantikan proses manual menjadi terkomputerisasi. Sistem yang dikembangkan umumnya mampu membantu pencatatan pembayaran, pengelolaan data siswa, serta penyajian laporan administrasi keuangan sekolah secara lebih rapi dan terstruktur. Namun, sebagian besar penelitian terdahulu masih menekankan pengembangan sistem pada aspek fungsional dasar, seperti pencatatan transaksi dan penyajian informasi pembayaran, dengan pembahasan yang terbatas pada metode pengembangan perangkat lunak dan pengujian fungsional. Aspek keamanan data, pengelolaan hak akses pengguna, serta mekanisme pengujian internal sistem belum banyak dibahas secara mendalam, khususnya dalam konteks sistem administrasi sekolah yang melibatkan data sensitif siswa dan keuangan.

Selain itu, metode pengembangan yang digunakan pada penelitian terdahulu umumnya masih bersifat linear dan kurang menekankan proses pengembangan yang iteratif serta adaptif terhadap perubahan kebutuhan pengguna. Padahal, sistem administrasi sekolah memiliki karakteristik kebutuhan yang dapat berkembang seiring dengan kebijakan internal dan alur kerja sekolah. Berdasarkan celah tersebut, penelitian ini berfokus pada perancangan sistem informasi pengelolaan tagihan dan pembayaran administrasi siswa dengan pendekatan *Personal eXtreme Programming* (PXP), yang memungkinkan pengembangan sistem secara iteratif dan responsif oleh

pengembang tunggal. Selain menitikberatkan pada fungsionalitas sistem, penelitian ini juga memperhatikan aspek keamanan data melalui penerapan mekanisme autentikasi, otorisasi berbasis peran, validasi input, serta pengujian sistem yang mencakup *white-box testing* dan *black-box testing*. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi yang lebih komprehensif dibandingkan penelitian terdahulu, khususnya dalam mengintegrasikan aspek pengembangan sistem, keamanan data, dan penerimaan pengguna dalam konteks administrasi sekolah.

2.2.1 Sistem Informasi

Nitami, Munthe, dan Masrizal (2021) mendefinisikan sistem informasi sebagai se-kumpulan komponen informasi yang saling berhubungan saling mengumpulkan atau mendapatkan, memproses, menyimpan, dan mendistribusikan informasi Nitami et al. (2021).

Adham (2024) menegaskan bahwa sistem informasi adalah serankaian komponen yang saling berinteraksi untuk membantu dalam pengumpulan, pengolahan, penyimpanan, dan penyebaran informasi secara efektif dan efisien Adham (2024).

Berdasarkan uraian diatas, dapat disimpulkan bahwa sistem informasi adalah sebuah kombinasi komponen yang terorganisir untuk mendukung dan mencapai suatu tujuan.

2.2.2 Sistem Pembayaran

Sistem pembayaran adalah seperangkat infrastruktur yang terdiri atas lembaga, instrumen, aturan, prosedur, standar, serta mekanisme teknis yang digunakan untuk memfasilitasi pemindahan nilai uang antar pihak yang bertransaksi, baik melalui instrumen pembayaran tunai maupun non-tunai Suryanto, Muhyi, dan Kurniati (2022).

Pembayaran Tunai

Sistem pembayaran tunai merupakan sistem pembayaran yang memanfaatkan uang kartal, yaitu uang kertas dan uang logam, sebagai alat tukar dalam berbagai aktivitas ekonomi dan transaksi jual beli Rayadi dan Maradesa (2021).

Pembayaran Non-Tunai

Sistem pembayaran non-tunai merupakan sistem pembayaran yang tidak lagi menggunakan uang kartal secara langsung, tetapi memanfaatkan beragam instrumen, mulai dari warkat seperti cek dan bilyet giro hingga instrumen berbasis elektronik seperti kartu dan uang elektronik Khayyirah, Kurniawan, dan Gemilang (2022).

2.2.3 Pembayaran Kanal

Kanal pembayaran merupakan jalur layanan yang digunakan untuk memproses transaksi uang elektronik. Pada uang elektronik berbasis chip kanal ini berupa *shared delivery channel* seperti ATM, mesin EDC, dan kode QR, sedangkan pada uang elektronik berbasis server kanal pembayaran diwujudkan melalui layanan *mobile banking* dan *internet banking* Sihaloho, Ramadani, dan Rahmayanti (2020).

2.2.4 Metode Pengembangan Model *Personal eXtreme Programming*

Personal eXtreme Programming (PXP) merupakan pengembangan dari *Extreme Programming* (XP) yang disesuaikan untuk pengembang tunggal, memungkinkan pengembangan textitagile tanpa tim besar Ulfie, Marthasari, dan Nuryasin (2024). Tahapan pada metode PXP dapat dijelaskan sebagai berikut:

1. Requirement

Tahap ini melibatkan pengumpulan kebutuhan sistem dari klien atau pengguna, yang kemudian didokumentasikan untuk menjadi dasar pengembangan perangkat lunak.

2. Planning

Pada tahap ini, pengembang merencanakan iterasi pengembangan berdasarkan prioritas kebutuhan yang telah dikumpulkan, serta mendiskusikan rencana tersebut dengan klien untuk memastikan kesesuaian dengan harapan mereka.

3. Iteration Initiations

Tahap ini merupakan proses awal dalam pengembangan perangkat lunak, di mana pengembang menentukan iterasi awal dan memulai pengembangan sistem.

4. Design

Tahap ini merupakan proses bertahap pada perancangan berbagai komponen sistem, seperti membuat desain dan alur pemrograman. Sehingga dapat membantu untuk menentukan perangkat lunak dan algoritma secara keseluruhan.

5. Implementation

Pada tahap ini, rancangan yang telah disusun diubah ke dalam bentuk kode program yang dapat dijalankan oleh komputer. Setiap modul dikembangkan sesuai desain yang telah ditentukan sehingga membentuk fungsi-fungsi sistem yang saling terintegrasi.

6. System Testing

Pada tahap ini, pengujian akan dilakukan untuk memastikan bahwa perangkat lunak yang telah dibuat bekerja sesuai yang diharapkan.

7. Retrospective

Tahap ini merupakan evaluasi keseluruhan proses pengembangan perangkat lunak, termasuk pencapaian tujuan, kendala yang dihadapi, serta pelajaran yang dapat diambil untuk perbaikan di masa mendatang.

2.2.5 Unified Modeling Language

Unified Modeling Language (UML) adalah bahasa pemodelan standar yang digunakan untuk mendokumentasikan, mendefinisikan, dan membangun perangkat lunak. UML menggunakan notasi grafis berupa diagram-diagram untuk memvisualisasikan, menspesifikasikan, serta mendokumentasikan sistem perangkat lunak yang berbasis orientasi objek Nasution, Suendri, dan Harahap (2023).

Use Case Diagram

Use Case Diagram merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use Case* digunakan untuk mengetahui apa saja fungsi yang ada di dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi tersebut Endang dan Susanti (2021). Lihat Tabel A.1 pada Lampiran A.

Entity-Relationship Diagram

Entity-Relationship Diagram (ERD) merupakan sebuah konsep data model yang digunakan dalam desain *database*. ERD menggambarkan relasi antar data objek (entitas) dan atribut yang dimiliki masing-masing entitas Palinggi, Maesaroh, Permana, Huda, dan Priyono (2024). Lihat Tabel B.1 pada Lampiran B.

Activity Diagram

Activity Diagram merupakan pemodelan alur kerja dalam sistem yang menunjukkan bagaimana berbagai aktivitas sistem saling berhubungan dan berurutan. *Activity Diagram* berfokus pada proses internal sistem, seperti pengolahan data atau eksekusi perintah, menggambarkan alur kegiatan, keputusan, dan akhir proses Khasani dan Subrata (2025). Lihat Tabel C.1 pada Lampiran C.

2.2.6 Black-Box Testing

Black-box testing merupakan metode pengujian perangkat lunak yang berfokus pada pengecekan fungsi-fungsi sistem berdasarkan spesifikasi yang telah ditentukan, tanpa melihat atau memahami struktur kode program di dalamnya. Validitas pengujian dijaga dengan cara membandingkan secara konsisten antara spesifikasi sistem dan keluaran yang diharapkan dari setiap skenario uji Kartono et al. (2024).

User Acceptance Test

User Acceptance Testing (UAT) merupakan salah satu metode pengujian *alpha* yang melibatkan pengguna akhir (*end-user*) yang secara langsung berinteraksi dengan sistem. Pengujian ini bertujuan untuk memverifikasi apakah seluruh fungsi yang disediakan oleh sistem telah berjalan sesuai dengan kebutuhan dan harapan pengguna. UAT dilakukan setelah tahap pengujian sistem selesai, sebagai bentuk evaluasi untuk memastikan bahwa sistem telah memenuhi persyaratan yang ditetapkan As, Hartono, dan Muin (2024).

2.2.7 White-Box Testing

White-box testing merupakan metode pengujian perangkat lunak yang berfokus pada mekanisme internal sistem, khususnya pada alur kontrol dan alur data dalam suatu program. Pengujian ini dilakukan dengan memeriksa struktur kode sumber untuk memastikan bahwa logika program berjalan sesuai dengan yang diharapkan Sie, Musdar, dan Bahri (2022).

2.2.8 Instrumen Penelitian Kuantitatif

Instrumen penelitian kuantitatif adalah alat atau sarana yang digunakan peneliti untuk mengumpulkan data berbentuk angka, sehingga data tersebut dapat diolah dan dianalisis secara statistik Subhaktiyasa (2024).

Skala Likert

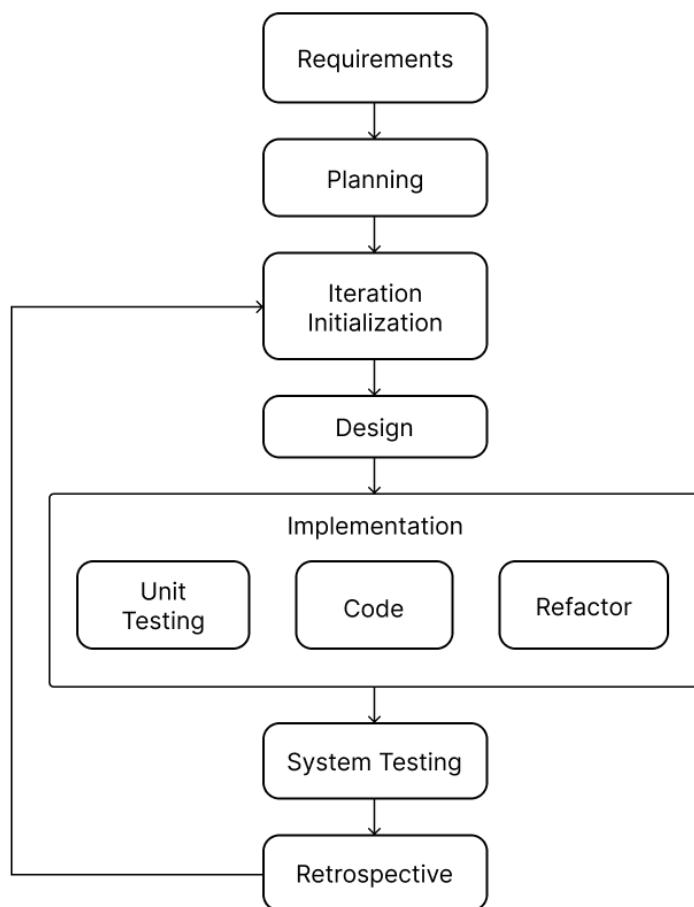
Skala Likert adalah salah satu jenis skala psikometrik yang paling sering digunakan dalam kuesioner, khususnya pada penelitian dengan metode survei. Skala ini digunakan untuk mengukur sikap, pendapat, serta persepsi individu maupun kelompok terhadap suatu peristiwa atau fenomena sosial Rahayu dan Shafina (2022).

BAB 3

METODOLOGI DAN RANCANGAN PENELITIAN

3.1 Metodologi Penelitian

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *Personal Extreme Programming* (PXP). Metode ini dipilih karena sesuai dengan karakteristik pengembangan sistem yang dilakukan oleh pengembang tunggal (*single developer*) serta membutuhkan proses pengembangan yang bersifat iteratif dan fleksibel.



Gambar 3.1: Alur metode *Personal eXtreme Programming*

3.1.1 Requirements (Kebutuhan)

User stories dituliskan dengan format: “Sebagai [jenis pengguna], saya ingin [melakukan aktivitas tertentu] agar [tujuan atau manfaat yang diperoleh]”. Pendekatan ini digunakan untuk memastikan bahwa setiap kebutuhan sistem berorientasi langsung

pada pengguna serta mudah dipahami oleh pengembang. Daftar user stories yang dihasilkan pada tahap ini disajikan pada Tabel 3.1.

Tabel 3.1: Kebutuhan Fungsional Sistem dalam Bentuk User Stories (PXP)

Iterasi	Kode US	Role	User Story
1	US-01	Tata Usaha / Bendahara	Sebagai Tata Usaha/Bendahara, saya ingin melakukan login agar dapat mengakses sistem sesuai dengan hak akses yang dimiliki.
	US-02	Orang Tua	Sebagai Orang Tua, saya ingin melakukan login agar dapat melihat informasi tagihan siswa.
2	US-03	Bendahara	Sebagai Bendahara, saya ingin mengelola master tagihan agar jenis dan nominal tagihan dapat ditentukan secara terpusat.
3	US-04	Tata Usaha	Sebagai Tata Usaha, saya ingin mengekspor template Excel data siswa agar format pengisian sesuai dengan kebutuhan sistem.
	US-05	Tata Usaha	Sebagai Tata Usaha, saya ingin mengimpor data siswa dari file Excel agar data siswa dapat dimasukkan dengan cepat tanpa input manual.
	US-06	Tata Usaha	Sebagai Tata Usaha, saya ingin mengubah atau menghapus data siswa apabila terjadi kesalahan atau perubahan.
4	US-07	Tata Usaha	Sebagai Tata Usaha, saya ingin menambahkan data kelas berdasarkan data siswa serta mengisi jurusan dan wali kelas agar informasi kelas lengkap untuk administrasi sekolah.
	US-08	Tata Usaha	Sebagai Tata Usaha, saya ingin mengubah atau menghapus data kelas apabila terjadi kesalahan atau perubahan.
5	US-09	Tata Usaha	Sebagai Tata Usaha, saya ingin membuat tagihan berdasarkan master tagihan agar penagihan sesuai ketentuan sekolah.
	US-10	Tata Usaha	Sebagai Tata Usaha, saya ingin menentukan target tagihan agar penagihan tepat sasaran.

(Dilanjutkan ke halaman berikutnya)

Iterasi	Kode US	Role	User Story
6	US-11	Tata Usaha	Sebagai Tata Usaha, saya ingin menentukan tanggal tagihan dan jatuh tempo agar orang tua mengetahui batas waktu pembayaran.
	US-12	Tata Usaha	Sebagai Tata Usaha, saya ingin melihat daftar tagihan agar dapat memantau status pembayaran.
	US-13	Tata Usaha	Sebagai Tata Usaha, saya ingin memfilter daftar tagihan agar pencarian data lebih mudah.
	US-14	Tata Usaha	Sebagai Tata Usaha, saya ingin mengubah atau menghapus tagihan apabila terjadi kesalahan.
7	US-15	Tata Usaha	Sebagai Tata Usaha, saya ingin melihat laporan pembayaran agar dapat memantau status pembayaran siswa.
	US-16	Tata Usaha	Sebagai Tata Usaha, saya ingin memfilter laporan pembayaran agar data sesuai kebutuhan.
	US-17	Tata Usaha	Sebagai Tata Usaha, saya ingin melakukan verifikasi manual pembayaran agar dapat membantu apabila terjadi kendala.
	US-18	Tata Usaha	Sebagai Tata Usaha, saya ingin menolak pembayaran agar pembayaran bermasalah tidak dianggap berhasil.
	US-19	Tata Usaha	Sebagai Tata Usaha, saya ingin melihat dan mencetak invoice pembayaran sebagai bukti pembayaran.
	US-20	Tata Usaha	Sebagai Tata Usaha, saya ingin mengekspor laporan pembayaran ke file Excel untuk keperluan arsip.
8	US-21	Tata Usaha	Sebagai Tata Usaha, saya ingin melihat daftar riwayat penolakan pembayaran.
	US-22	Tata Usaha	Sebagai Tata Usaha, saya ingin melihat detail penolakan pembayaran beserta alasannya.
	US-23	Semua Pengguna	Sebagai pengguna, saya ingin melihat daftar tagihan agar mengetahui status pembayaran.
	US-24	Semua Pengguna	Sebagai pengguna, saya ingin melihat dan mencetak invoice pembayaran.

(Dilanjutkan ke halaman berikutnya)

Iterasi	Kode US	Role	User Story
	US-25	Semua Pengguna	Sebagai pengguna, saya ingin melihat informasi tata cara pembayaran.
9	US-26	Bendahara	Sebagai Bendahara, saya ingin melihat laporan pembayaran yang telah berhasil agar dapat memantau pemasukan sekolah.
	US-27	Bendahara	Sebagai Bendahara, saya ingin memfilter laporan pembayaran agar data sesuai kebutuhan.
	US-28	Bendahara	Sebagai Bendahara, saya ingin melihat detail invoice pembayaran agar memiliki bukti pembayaran yang valid.
10	US-29	Bendahara	Sebagai Bendahara, saya ingin melihat ringkasan pembayaran dan jumlah tagihan aktif agar dapat memantau kondisi keuangan sekolah.
	US-30	Tata Usaha	Sebagai Tata Usaha, saya ingin melihat ringkasan data tagihan dan pembayaran agar dapat memantau kondisi pembayaran secara keseluruhan.
	US-31	Orang Tua	Sebagai Orang Tua, saya ingin melihat beranda yang menampilkan ringkasan status tagihan dan notifikasi terkait siswa agar dapat mengetahui kewajiban pembayaran dengan cepat.

3.1.2 Planning (Perencanaan)

Pada tahap ini mendefinisikan kebutuhan fungsional dan non-fungsional dari sistem yang akan dikembangkan. Tabel 3.2 memperlihatkan identifikasi kebutuhan fungsional dan non-fungsional dari sistem pembayaran dan administrasi pada SMAN 15 Surabaya.

Tabel 3.2: Kebutuhan Fungsional dan Non-Fungsional Sistem SIPANSE

Kebutuhan Fungsional	Kebutuhan Non-Fungsional
Autentikasi pengguna dan akses ke dalam sistem	Deskripsi Kebutuhan Non-Fungsional
Pengelolaan data master tagihan	Keamanan data sistem
Manajemen data siswa dan data kelas	Antarmuka sistem yang mudah digunakan (<i>user-friendly</i>)
Pembuatan dan pengelolaan tagihan administrasi	Konsistensi dan keandalan sistem
Proses pembayaran dan verifikasi pembayaran	–
Pengelolaan penolakan pembayaran	–
Penyajian laporan pembayaran	–
Akses informasi tagihan dan pengecekan invoice	–
Penyajian dashboard sistem	–
Pengiriman notifikasi tagihan kepada orang tua	–

3.1.3 Iteration Initialization (Inisialisasi Iterasi)

Berdasarkan kebutuhan fungsional dan non-fungsional yang telah diidentifikasi, tahap inisialisasi iterasi dilakukan dengan memfokuskan masing-masing user stories yang telah didefinisikan pada tahap requirements. Setiap iterasi dikerjakan secara bertahap dan tidak boleh melompat ke iterasi lain apabila belum selesai, hal ini bertujuan agar proses pengembangan sistem dapat berjalan dengan terstruktur.

3.1.4 Design (Desain)

Pada tahap *Design*, dilakukan perancangan awal sistem berdasarkan kebutuhan yang telah didefinisikan pada tahap *Requirements* dan *Planning*. Tahap ini bertujuan untuk menerjemahkan kebutuhan fungsional dan non-fungsional ke dalam bentuk rancangan sistem yang terstruktur sebelum proses implementasi dilakukan.

Perancangan sistem pada penelitian ini direpresentasikan dalam bentuk diagram dan rancangan visual, antara lain *Entity Relationship Diagram* (ERD) untuk memodelkan struktur basis data, *Use Case Diagram* untuk menggambarkan fungsionalitas sistem berdasarkan peran pengguna, *Activity Diagram* untuk menjelaskan alur proses utama sistem, serta *mockup* antarmuka sebagai gambaran awal tampilan sistem.

Pembahasan secara rinci mengenai rancangan sistem tersebut disajikan pada

bagian **Perancangan Sistem Awal**, yang mencakup penjelasan setiap diagram dan rancangan antarmuka yang digunakan dalam pengembangan sistem.

3.1.5 Implementation (Implementasi)

Pada tahap ini, pengembangan sistem dilakukan secara iteratif dengan mengawali proses dari penyusunan dan pelaksanaan *unit testing* sebagai acuan keberhasilan setiap modul. Pengujian ini digunakan untuk memastikan bahwa fungsionalitas yang dikembangkan telah sesuai dengan kebutuhan yang dirancang.

Berdasarkan hasil *unit testing* tersebut, proses pengkodean dilakukan hingga modul dapat memenuhi kriteria pengujian yang ditetapkan. Apabila masih ditemukan kegagalan pada pengujian, maka dilakukan perbaikan kode melalui proses *refactoring*. Setelah seluruh pengujian berhasil dan modul dinyatakan berjalan dengan baik, pengembangan dilanjutkan ke modul berikutnya.

Unit Testing (Pengujian Unit)

Unit testing merupakan tahapan pengujian yang digunakan untuk memastikan bahwa setiap fungsi pada sistem dapat berjalan sesuai dengan yang diharapkan. Pengujian ini dilakukan menggunakan fasilitas pengujian pada *framework* Laravel untuk memverifikasi apakah hasil pengujian berada pada kondisi berhasil (*passed*) atau gagal (*failed*). Hasil pengujian ini digunakan sebagai dasar evaluasi sebelum melanjutkan ke tahap pengkodean.

Code (Kode)

Pada tahap pengkodean, sistem diimplementasikan berdasarkan hasil perancangan pada subbab Perancangan Sistem Awal. Implementasi dilakukan menggunakan *framework* Laravel dengan pendekatakan arsitektur *Model-View-Controller* (MVC).

Sistem manajemen basis data yang digunakan dalam pengembangan sistem adalah MySQL, yang dipilih karena bersifat *open-source* serta memiliki kompatibilitas yang baik dengan *framework* Laravel. Proses pengkodean sistem dilakukan secara iteratif sesuai dengan prinsip *Personal Extreme Programming* (PXP), di mana pengembangan fitur dilaksanakan secara bertahap berdasarkan kebutuhan yang telah diidentifikasi.

Keamanan data juga menjadi salah satu fokus utama pada tahap implementasi sistem. Langkah-langkah keamanan yang diterapkan dalam proses pengkodean meliputi:

- Pengamanan autentikasi pengguna melalui mekanisme *password hashing* menggunakan algoritma *bcrypt*, serta fitur *forgot password* dan *reset password* berbasis email.
- Penerapan *middleware* autentikasi dan otorisasi untuk membatasi akses sistem berdasarkan peran pengguna (*role-based access control*), sehingga setiap pengguna hanya dapat mengakses fitur sesuai dengan kewenangannya.
- Validasi input untuk mencegah serangan keamanan seperti *SQL Injection* dan *Cross-Site Scripting (XSS)*, serta penerapan *Cross-Site Request Forgery (CSRF) Protection*. Selain itu, digunakan library *HTML Purifier* untuk membersihkan input berbentuk HTML.
- Pemanfaatan *Eloquent ORM* dalam pengolahan basis data untuk meningkatkan keamanan query dan meminimalkan risiko manipulasi data.
- Penerapan *audit trail* untuk mencatat aktivitas pengguna, seperti pada proses pembuatan tagihan serta verifikasi pembayaran.
- Penerapan mekanisme *backup* basis data dan pemeliharaan sistem secara terjadwal untuk menjaga integritas dan ketersediaan data.

Refactor (Refaktor)

Setelah proses *unit testing* dan pengkodean selesai dilakukan, tahap selanjutnya adalah proses *refactoring* untuk meningkatkan kualitas kode program tanpa mengubah fungsionalitas sistem. Pada tahap ini dilakukan perbaikan struktur kode serta penyesuaian penulisan kode agar lebih efisien dan mudah dipahami. Proses refactoring bertujuan untuk menjaga keterbacaan dan kemudahan pemeliharaan kode, sehingga sistem dapat dikembangkan lebih lanjut.

3.1.6 System Testing (Pengujian Sistem)

Pada metode Personal Extreme Programming (PXP), proses pengujian dilakukan secara sederhana dan bertujuan untuk memastikan bahwa sistem yang dibangun telah berfungsi sesuai dengan kebutuhan pengguna. Pada tahap ini dilakukan dua jenis pengujian, yaitu *white-box testing* dan *black-box testing*.

White-box Testing

White-box testing dilakukan dengan memeriksa struktur internal kode program untuk memastikan logika dan alur program berjalan dengan benar. Pada penelitian ini,

metode yang digunakan dalam *white-box testing* adalah *unit testing*, di mana setiap unit atau modul kecil pada sistem diuji secara terpisah. Pengujian ini berfokus pada validasi logika program, alur kontrol, serta penanganan kesalahan pada proses-proses utama sistem.

Black-box Testing

Black-box testing dilakukan tanpa memperhatikan struktur internal kode program, melainkan berfokus pada pengujian fungsionalitas sistem dari sudut pandang pengguna. Pada penelitian ini, metode *black-box testing* yang digunakan adalah *User Acceptance Testing* (UAT). Pengujian UAT dilakukan dengan melibatkan pengguna sistem, yaitu pihak Tata Usaha, Bendahara, dan Orang Tua siswa. Pengguna diminta untuk menguji sistem berdasarkan skenario penggunaan yang telah ditentukan, guna memvalidasi apakah sistem telah memenuhi kebutuhan pengguna serta mudah digunakan. Pengujian ini mencakup aspek fungsionalitas sistem, tampilan antarmuka (user interface), dan pengalaman pengguna (user experience).

3.1.7 Retrospective (Retrospektif)

Pada tahap ini dilakukan evaluasi terhadap hasil pengembangan sistem yang telah dilakukan pada setiap iterasi. Tahap ini bertujuan untuk menarik kesimpulan atas kinerja dan kualitas sistem yang dibangun. Apabila masih ditemukan kekurangan atau kesalahan, maka dilakukan perbaikan dengan kembali ke tahap *Iteration Initialization* pada iterasi yang bersangkutan untuk penyempurnaan sistem.

3.2 Alat dan Bahan yang Digunakan

Alat dan bahan yang digunakan untuk membantu proses pengembangan sistem dapat dilihat pada Tabel 3.3 perangkat keras (*hardware*) dan Tabel 3.4 perangkat lunak (*software*).

Tabel 3.3: Perangkat Keras (*Hardware*)

No	Hardware	Spesifikasi
1	Laptop	Prosesor Intel(R) Core(TM) i7-9750H CPU @ 2.60GHz / NVIDIA GeForce GTX 1660 Ti / RAM 16 GB / SSD 256 GB (<i>Internal</i>) / SSD 1 TB (<i>External</i>), sistem operasi Windows 11.

Tabel 3.4: Perangkat Lunak (*Software*)

No	Software	Spesifikasi / Versi
1	<i>Visual Studio Code</i>	v12.15.0
2	Figma	<i>Release 25 November 2025</i>

3.3 Perancangan Sistem Awal

3.3.1 Use Case



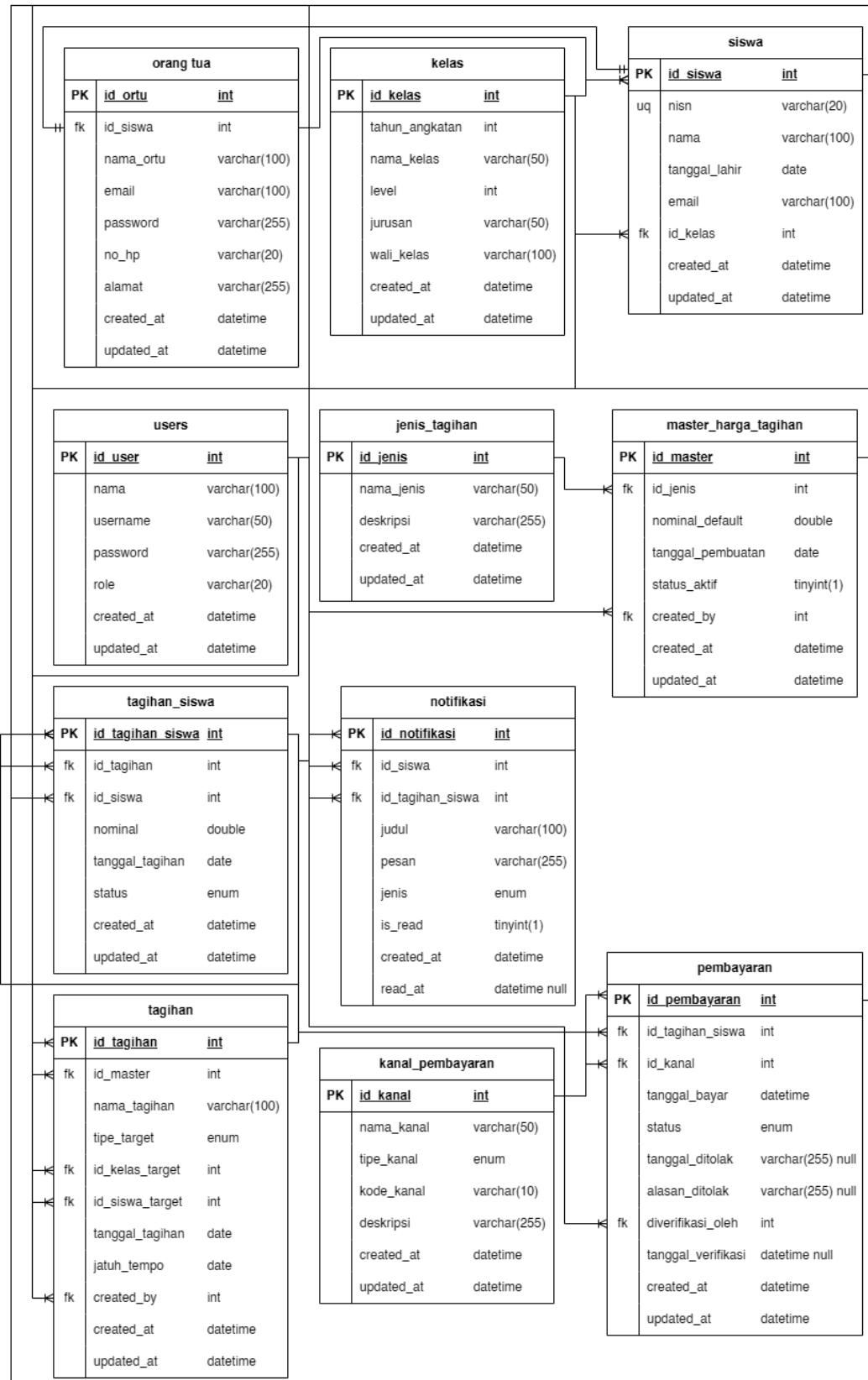
Gambar 3.2: *Use Case Diagram*

Use Case Diagram pada Gambar 3.2 memodelkan interaksi antara empat aktor utama, yaitu Bendahara, Tata Usaha, Orang Tua, dan Sistem Pembayaran Kanal. Bendahara berfokus pada pengelolaan *master* harga tagihan melalui use case *Membuat Harga Tagihan*, serta melakukan pemantauan keuangan melalui *Melihat Laporan Pembayaran* dan *Cetak Invoice*. Tata Usaha bertanggung jawab mengelola

data operasional sekolah, mulai dari *Kelola Data Siswa dan Kelas*, *Membuat Tagihan*, hingga *Melihat Daftar Tagihan Siswa*. Pada sisi pemantauan pembayaran, TU menggunakan use case *Melihat Status Pembayaran* yang dapat diperluas («extend») ke use case *Verifikasi Pembayaran* dan *Membatalkan Pembayaran* ketika diperlukan, misalnya saat ada pembayaran bermasalah atau perlu koreksi.

Orang Tua berperan sebagai pengguna eksternal yang mengakses portal untuk *Login*, melihat *Daftar Tagihan Siswa*, *Melihat Detail Tagihan*, dan memantau *Status Pembayaran*. Sistem juga menyediakan use case *Mendapatkan Notifikasi* agar orang tua menerima pemberitahuan ketika ada tagihan baru, perubahan status pembayaran, atau informasi penting lainnya, serta use case *Cetak Invoice* sebagai bukti dan panduan pembayaran. Di sisi lain, Sistem Pembayaran Kanal (misalnya bank atau *payment gateway*) direpresentasikan sebagai aktor eksternal yang berinteraksi melalui use case *Menerima Pembayaran Kanal*, yaitu proses penerimaan dan pemrosesan transaksi pembayaran yang kemudian diintegrasikan kembali ke sistem. Secara keseluruhan, diagram ini menunjukkan bahwa proses penagihan, pembayaran, dan pemantauan status dapat berjalan terstruktur dengan peran yang jelas pada setiap aktor.

3.3.2 Entity-Relationship Diagram



Gambar 3.3: Entity-Relationship Diagram

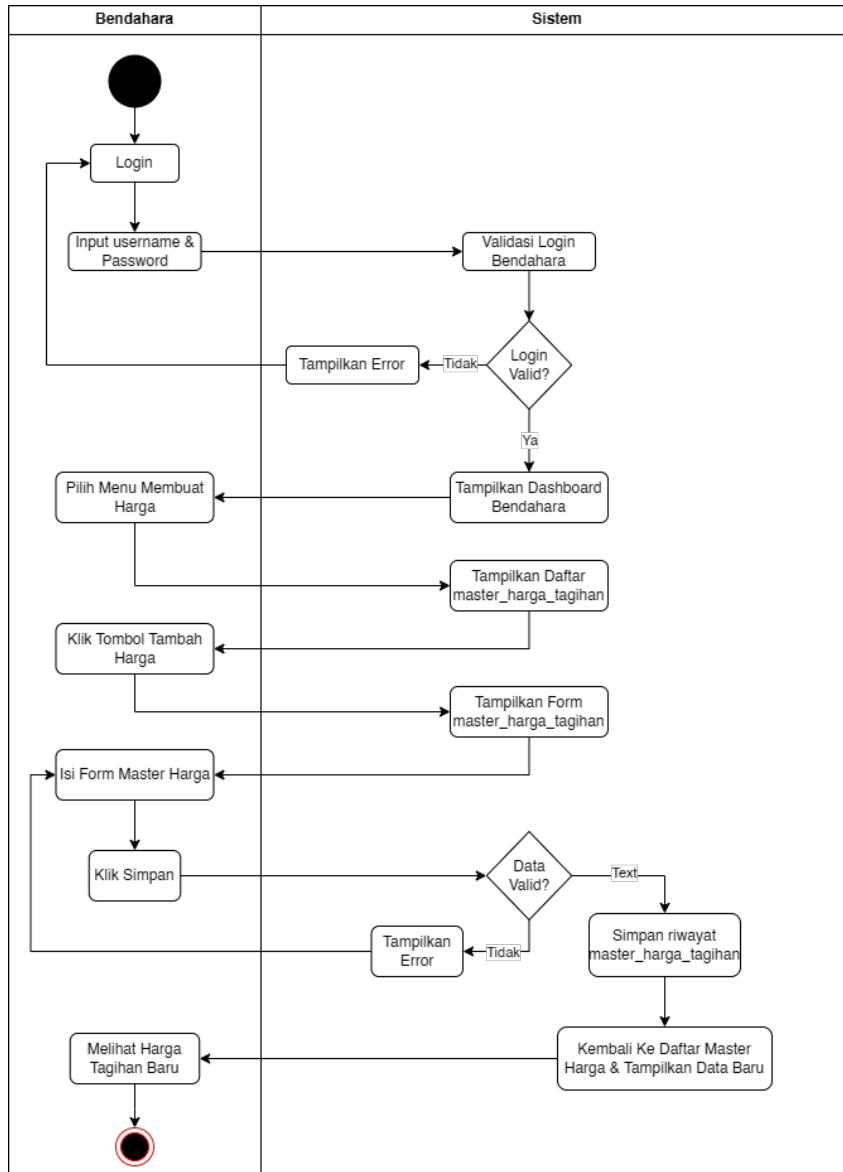
Entity Relationship Diagram (ERD) pada Gambar 3.3 memodelkan struktur basis data relasional yang digunakan untuk menyimpan dan mengelola informasi pada sistem penagihan dan pembayaran iuran siswa di SMAN 15. Perancangan ini berpusat pada entitas pengguna utama, yaitu users sebagai akun petugas internal (Tata Usaha dan Bendahara), entitas siswa yang merepresentasikan peserta didik, serta entitas orang_tua yang menjadi pemilik akun portal orang tua. Setiap siswa terhubung ke satu entitas kelas melalui relasi *one-to-many*, sehingga informasi tahun angkatan, nama kelas, level, jurusan, dan wali kelas dapat dikelola secara terstruktur. Di sisi lain, entitas orang_tua berelasi ke siswa melalui atribut id_siswa, sehingga satu siswa dapat memiliki lebih dari satu akun orang tua (misalnya ayah dan ibu), sementara orang tua melakukan login menggunakan NISN anak yang tersimpan pada entitas siswa.

Pada sisi manajemen tagihan, Bendahara mengelola referensi jenis biaya melalui entitas `jenis_tagihan` dan `master_harga_tagihan`. Entitas `jenis_tagihan` mendefinisikan kategori umum seperti SPP, kegiatan OSIS, buku, atau seragam, sedangkan `master_harga_tagihan` menyimpan master harga *default*, status aktif, serta informasi pembuat (`created_by` yang berelasi ke `users`) sebagai dasar penyusunan tagihan periodik. Petugas Tata Usaha kemudian membentuk entitas tagihan sebagai definisi tagihan per periode, yang mengacu ke master harga, menyimpan nama tagihan, tipe target (semua siswa, per kelas, atau per siswa), serta relasi opsional ke kelas (`id_kelas_target`) atau langsung ke siswa (`id_siswa_target`). Setiap definisi tagihan tersebut dirinci lebih lanjut ke level individu melalui entitas transaksional `tagihan_siswa`, yang menghubungkan satu tagihan dengan satu siswa dan menyimpan informasi nominal yang dikenakan, tanggal tagihan, jatuh tempo, serta status (misalnya “Belum Lunas” atau “Lunas”).

Proses pembayaran dimodelkan oleh entitas pembayaran yang merekam setiap transaksi yang dilakukan orang tua terhadap sebuah `tagihan_siswa`. Entitas ini menyimpan referensi ke `tagihan_siswa`, kanal pembayaran yang digunakan melalui relasi ke entitas `kanal_pembayaran` (misalnya transfer bank, *virtual account*, atau tunai), tanggal dan nominal bayar, status pembayaran, serta informasi verifikasi oleh petugas Tata Usaha (`diverifikasi_oleh` yang berelasi ke `users`). Untuk mendukung transparansi dan komunikasi, sistem juga memanfaatkan entitas notifikasi yang berelasi ke siswa dan `tagihan_siswa`, sehingga setiap pembuatan tagihan baru, perubahan status pembayaran, maupun penolakan transaksi dapat dikirimkan sebagai notifikasi yang dapat dibaca melalui portal orang tua. Seluruh entitas dilengkapi atribut waktu seperti `created_at` dan `updated_at` untuk mendukung kebutuhan audit trail dan pelacakan riwayat perubahan data dalam sistem.

3.3.3 Activity Diagram

Activity Diagram Bendahara



Gambar 3.4: Activity Diagram Bendahara

Activity Diagram pada Gambar 3.4 menggambarkan proses yang dilakukan Bendahara sekolah dalam mengelola *master* harga tagihan, yang menjadi acuan nominal bagi tagihan yang dibuat oleh Tata Usaha.

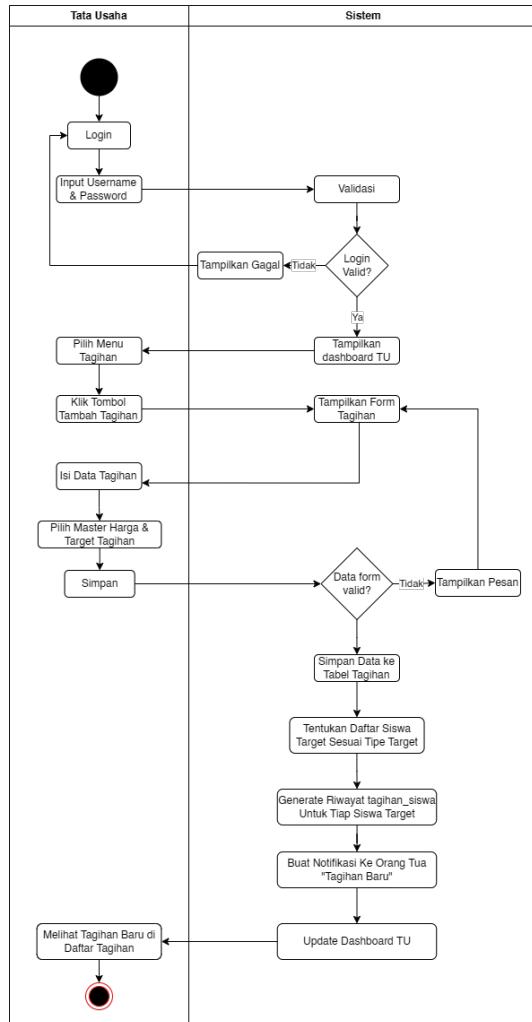
Proses dimulai ketika Bendahara melakukan login ke sistem dan memasukkan *username* serta kata sandi. Sistem memvalidasi kredensial login; jika tidak valid, sistem menampilkan pesan kesalahan dan mengembalikan Bendahara ke halaman login. Jika valid, sistem menampilkan *dashboard* Bendahara yang memuat

ringkasan jumlah jenis tagihan dan total pembayaran. Dari *dashboard* tersebut, Bendahara memilih menu *Membuat Harga* sehingga sistem menampilkan daftar `master_harga_tagihan` yang sudah tersimpan.

Untuk menambahkan harga tagihan baru, Bendahara menekan tombol *Tambah Harga*. Sistem menampilkan form `master_harga_tagihan` yang berisi pilihan jenis tagihan, nama tagihan, nominal *default*, status aktif, dan atribut lain yang diperlukan. Bendahara kemudian mengisi form master harga dan menekan tombol *Simpan*. Sistem melakukan keputusan *Data valid?*. Jika data tidak valid, sistem menampilkan pesan error dan mengembalikan Bendahara ke form untuk memperbaiki isian. Apabila data valid, sistem menyimpan entri baru ke tabel `master_harga_tagihan` sebagai riwayat master harga yang dapat digunakan pada proses pembuatan tagihan.

Setelah penyimpanan berhasil, sistem kembali menampilkan daftar `master_harga_tagihan` dan menampilkan data baru pada daftar tersebut. Aktivitas di sisi Bendahara dia-khiri dengan melihat harga tagihan baru yang muncul di daftar, yang menandakan bahwa master harga berhasil ditambahkan dan siap digunakan oleh TU pada proses pembuatan tagihan.

Activity Diagram Tata Usaha



Gambar 3.5: *Activity Diagram Tata Usaha*

Activity Diagram pada Gambar 3.5 menjelaskan alur kerja petugas Tata Usaha (TU) dalam membuat tagihan baru berdasarkan master harga tagihan dan menentukan daftar siswa yang menjadi target penagihan.

Proses diawali saat TU melakukan login ke sistem dengan memasukkan *username* dan kata sandi. Sistem melakukan validasi login; apabila tidak valid, sistem menampilkan pesan gagal dan mengembalikan TU ke halaman login. Jika valid, sistem menampilkan *dashboard* TU yang berisi ringkasan data tagihan dan pembayaran. Dari *dashboard* ini TU memilih menu *Tagihan* dan menekan tombol *Tambah Tagihan* untuk membuat tagihan baru.

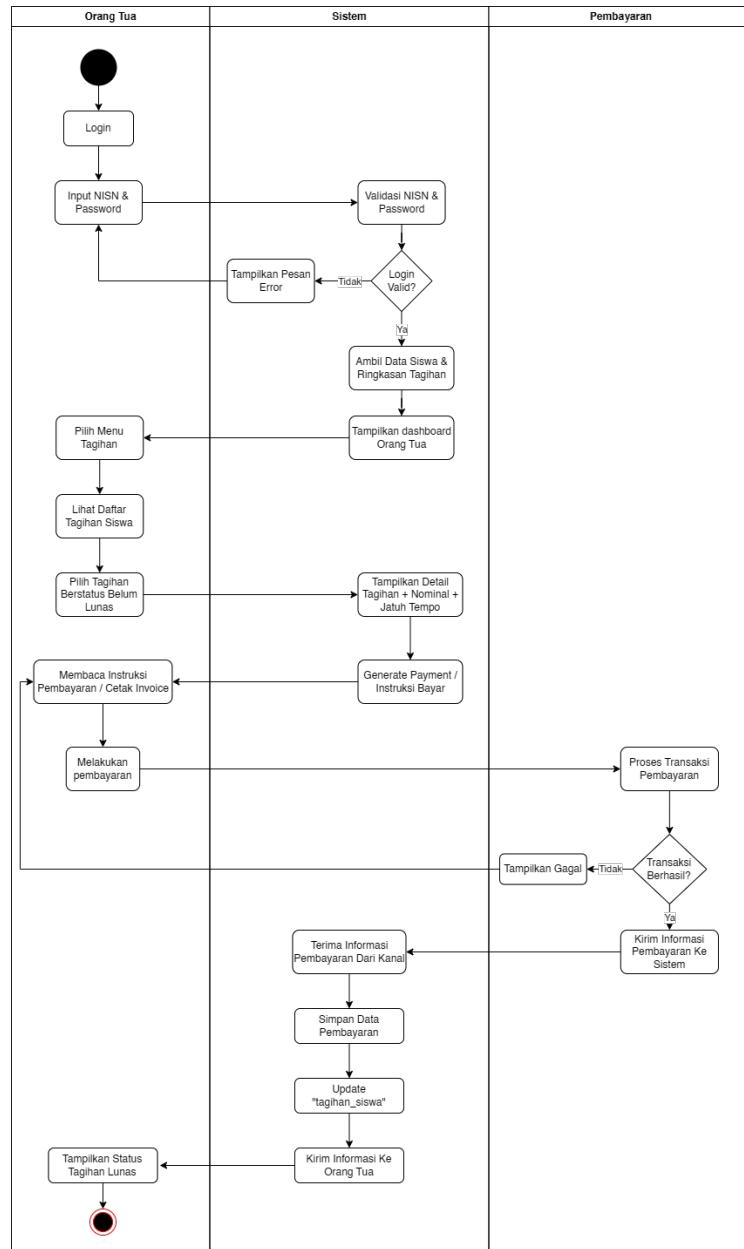
Sistem kemudian menampilkan form tagihan yang berisi pilihan master harga, tipe target (misalnya seluruh siswa, per kelas, atau per siswa), tanggal tagihan, tanggal jatuh tempo, serta informasi lain yang diperlukan. TU mengisi data tagihan

pada form tersebut berdasarkan `master_harga_tagihan` dan target yang dipilih, lalu menekan tombol *Simpan*. Setelah itu sistem melakukan keputusan *Data form valid?*. Jika data tidak valid (misalnya ada kolom yang kosong atau format tidak sesuai), sistem menampilkan pesan kesalahan dan mengembalikan TU ke form untuk memperbaiki isian. Jika data valid, sistem menyimpan data tersebut ke tabel `tagihan` sebagai definisi tagihan baru.

Berikutnya sistem menentukan daftar siswa target sesuai tipe target yang dipilih TU. Untuk setiap siswa yang termasuk dalam target, sistem menghasilkan entri `tagihan_siswa` dengan nilai nominal dan status awal *Belum Lunas*. Setelah proses pembentukan tagihan per siswa selesai, sistem membuat notifikasi kepada orang tua terkait adanya *tagihan baru* dan mengupdate *dashboard* TU, misalnya pada informasi total tagihan bulan berjalan dan daftar tagihan terbaru.

Pada sisi TU, proses ditutup dengan aktivitas melihat tagihan baru pada daftar tagihan, yang menandakan bahwa pembuatan tagihan berhasil dan siap ditindaklanjuti oleh orang tua.

Activity Diagram Orang Tua



Gambar 3.6: *Activity Diagram Orang Tua*

Activity Diagram pada Gambar 3.6 menggambarkan alur proses pembayaran tagihan sekolah dari sudut pandang orang tua, mulai dari login hingga status tagihan menjadi lunas.

Proses diawali ketika orang tua membuka portal dan melakukan *input* NISN serta kata sandi. Sistem kemudian melakukan validasi NISN dan kata sandi. Jika data login tidak valid, sistem menampilkan pesan kesalahan dan mengembalikan pengguna ke aktivitas pengisian kredensial. Apabila valid, sistem mengambil data

siswa beserta ringkasan tagihan, kemudian menampilkan *dashboard* orang tua.

Dari *dashboard*, orang tua memilih menu *Tagihan*, melihat daftar tagihan siswa, kemudian memilih salah satu tagihan yang berstatus *Belum Lunas*. Setelah tagihan dipilih, sistem menampilkan detail tagihan seperti nama tagihan, nominal, dan jatuh tempo, sekaligus menghasilkan instruksi pembayaran (misalnya nomor virtual account atau rekening tujuan) dan menampilkannya kepada orang tua dalam bentuk informasi maupun *invoice*.

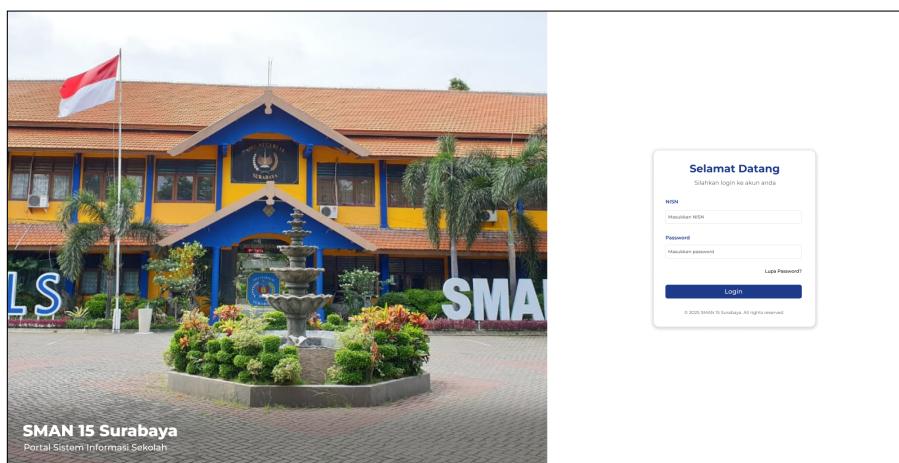
Selanjutnya orang tua melakukan pembayaran melalui kanal pembayaran (ATM, *mobile banking*, atau metode lain). Kanal pembayaran memproses transaksi dan melakukan keputusan apakah transaksi berhasil atau tidak. Jika transaksi gagal, informasi kegagalan disampaikan kepada orang tua sehingga orang tua dapat kembali ke instruksi pembayaran dan mencoba kembali. Jika transaksi berhasil, kanal pembayaran mengirimkan informasi pembayaran ke sistem, mencakup identitas tagihan, nominal, dan waktu pembayaran.

Sistem menerima informasi tersebut, menyimpan data pembayaran ke basis data, kemudian mengubah status data *tagihan_siswa* yang bersesuaian menjadi *Lunas*. Setelah itu, sistem mengirimkan informasi kepada orang tua bahwa pembayaran telah berhasil, dan pada sisi orang tua ditampilkan status tagihan yang telah berubah menjadi *Lunas*. Proses diakhiri ketika orang tua melihat bahwa tagihan yang dibayar sudah berstatus lunas.

3.3.4 Mockup

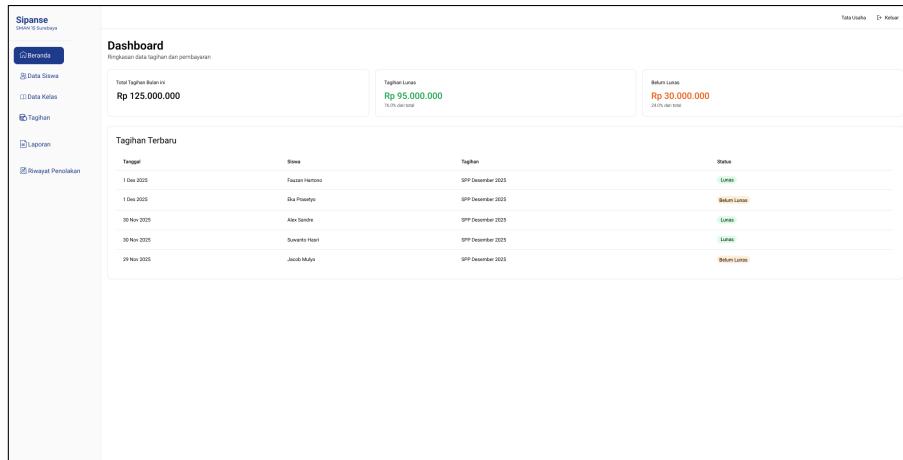
Halaman:

- *Landing Page*



Gambar 3.7: Desain *Landing Page*

- Tata Usaha - Beranda



Gambar 3.8: Desain Beranda Pada Tata Usaha

- Detail Tata Usaha - Data Siswa

Data Siswa							Template Data	+ Tambah Siswa
Ketahui data akhirnya dan informasi orang tua								
Tahun Angkatan	NIM	Kelas	Nama	Nama Orang Tua	Tanggal Lahir	Email	Action	
2024	0012345678	X - IPA 1	Iqbal Juna	Budi Santoso	15 Mei 2008	jiengjeng@gmail.com	Edit	Hapus
2024	0012345679	X - IPA 1	Yuda Muli	Michael Ronald	22 Maret 2008	semail@gmail.com	Edit	Hapus

Gambar 3.9: Desain Data Pada Siswa Tata Usaha

- Tata Usaha - Data Kelas

Tahun Ajaran	Kelas	Level	Jurusan	Wali Kelas	Jumlah Siswa	Action
2024	X-IPS-1	10	IPS	Dr. Siti Ainiyah, S.Pd	32 siswa	
2024	X-IPS-2	10	IPS	Drs. Hayyanto Kurniadi, M.Pd	30 siswa	
2023	X-IPS-1	11	IPS	Hj. Agus Bambara S.Pd	28 siswa	

Gambar 3.10: Desain Data Pada Kelas Tata Usaha

- Tata Usaha - Tagihan

Bulan	Kelas	Type Target	Nama Target	Nominal	Tanggal Tagihan	Jatuh Tempo	Status
Desember	Seleksi Kelas	Siswa	X-IPS-1	Rp 200,000	1/12/2025	20/12/2025	Lunas
		Per Kelas	X-IPS-1	Rp 120,000	5/12/2025	15/12/2025	Belum Lunas

Gambar 3.11: Desain Tagihan Pada Tata Usaha

- Tata Usaha - Laporan

Tanggal Bayar	Nama Siswa	Kelas	Nama Tagihan	Metode	Nominal	Status	Action
1/12/2023	Kahn Horsidan	X-IPK.1	SPP Bulan December 2023	Virtual Account	Rp 200.000	Belum	OK
2/12/2023	Loreta Mathela	X-IPK.2	SPP Bulan December 2023	Manual	Rp 20.000	Menunggu	X

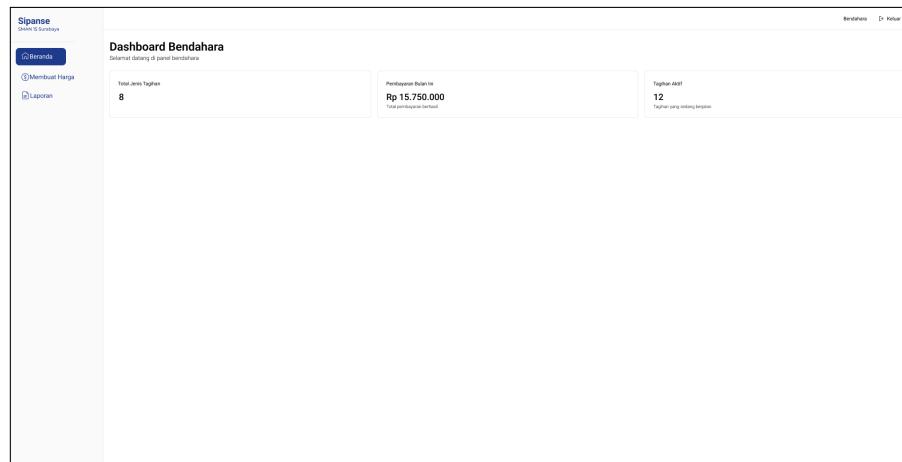
Gambar 3.12: Desain Laporan Pada Tata Usaha

- Tata Usaha - Riwayat Penolakan

Tanggal Ditolak	Nama Siswa	Kelas	Nama Tagihan	Metode	Nominal	Action
1/12/2023	Kahn Horsidan	X-IPK.1	SPP Bulan December 2023	Virtual Account	Rp 200.000	Ditolak
2/12/2023	Loreta Mathela	X-IPK.2	SPP Bulan December 2023	Cash	Rp 20.000	Ditolak

Gambar 3.13: Desain Riwayat Penolakan Pada Tata Usaha

- Bendahara - Beranda



Gambar 3.14: Desain Beranda Pada Bendahara

- Bendara - Membuat Harga

Tagihan					Bantahan	Dikirim
Ketela Tagihan sisa		Jenis Tagihan	Nama Tagihan			
Filter Jenis Tagihan	-	Can Nama Tagihan				
Tagih Periode:		Jenis Tagihan	Nama Tagihan	Nominal	Aktif	
2024-01-15		gap	SPP Siswa Kelas X	Rp 200.000	<input checked="" type="checkbox"/>	<input type="checkbox"/>
2024-01-15		Sengaja	Sengaja Lengkap	Rp 100.000	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Gambar 3.15: Desain Membuat Harga Pada Bendahara

- Bendahara - Laporan

Tanggal Bayar	Name Stew	Roles	Name Tagihan	Metode	Nominal	Status	Action
1/12/2023	Kain Herlina	X IPN 1	GPP Bulan December 2023	Virtual Account	Rp 200,000	Sudah	Detail
2/12/2023	Lonna Matilda	X IPN 2	GPP Bulan December 2023	Manual	Rp 20,000	Sudah	Detail

Gambar 3.16: Desain Laporan Pada Bendahara

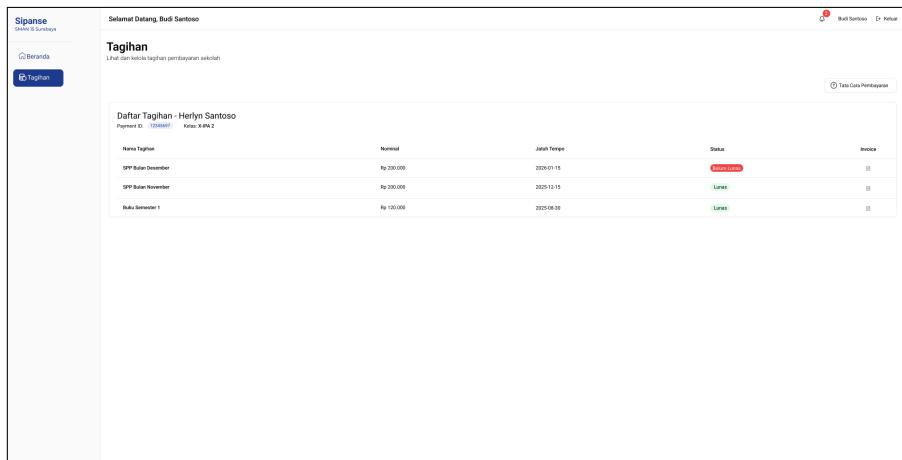
- Orang Tua - Beranda

Name	NIM/NRIC	Kelas
Budi Santoso	12021901	X IPN 2

Tagihan Belum Lunas	Tagihan Lunas
2	6
Tagihan yang belum dibayar	

Gambar 3.17: Desain Beranda Pada Orang Tua

- Orang Tua - Tagihan



Gambar 3.18: Desain Tagihan Pada Orang Tua

3.4 Pengujian Sistem (*Black-Box*)

Pengujian sistem dilakukan menggunakan metode *black-box testing*, yaitu pengujian fungsionalitas berdasarkan masukan dan keluaran yang dihasilkan tanpa melihat struktur kode program. Pengujian dilakukan pada fitur utama yang digunakan oleh orang tua dan admin/bendahara, seperti autentikasi, pengelolaan tagihan, pembuatan *invoice*/kode pembayaran, verifikasi pembayaran melalui kanal, pembaruan status secara manual, serta notifikasi pada halaman orang tua.

Tabel 3.5: Contoh *Test Case* Pengujian Fungsional dengan Metode *Black-Box*

ID	Modul/Fitur	Skenario Uji	Hasil yang Diharapkan	Status
BB-01	<i>Login</i> Orang Tua	Pengguna memasukkan NISN dan kata sandi yang valid lalu menekan tombol masuk.	Sistem menampilkan dashboard orang tua dan pengguna berhasil masuk ke sistem.	<i>Pass/Fail</i>
BB-02	<i>Login</i> Orang Tua	Pengguna memasukkan NISN/kata sandi tidak valid.	Sistem menolak login dan menampilkan pesan kesalahan tanpa masuk ke dashboard.	<i>Pass/Fail</i>

Continued on next page

Tabel 3.5: Contoh *Test Case* Pengujian Fungsional dengan Metode *Black-Box* (Continued)

BB-03	Tagihan Orang Tua	Orang tua membuka menu Tagihan dan memilih salah satu tagihan.	Sistem menampilkan daftar tagihan dan detail tagihan (nominal, jatuh tempo, status).	<i>Pass/Fail</i>
BB-04	<i>Generate Invoice/Kode Pembayaran</i>	Orang tua menekan tombol “Buat Pembayaran” pada tagihan berstatus belum lunas.	Sistem menghasilkan <i>invoice/kode pembayaran</i> dan menampilkan informasi pembayaran sesuai tagihan.	<i>Pass/Fail</i>
BB-05	Verifikasi Pembayaran (Kanal)	TU melakukan verifikasi transaksi pembayaran yang masuk dari kanal pembayaran.	Status tagihan berubah menjadi “Lunas”, riwayat pembayaran tercatat, dan tagihan tidak bisa dibuat pembayaran ulang.	<i>Pass/Fail</i>
BB-06	<i>Update Manual Status (Fallback)</i>	TU memperbarui status pembayaran secara manual ketika terjadi kendala verifikasi kanal.	Sistem menyimpan perubahan status, mencatat perubahan, dan data tagihan konsisten.	<i>Pass/Fail</i>

Continued on next page

Tabel 3.5: Contoh *Test Case* Pengujian Fungsional dengan Metode *Black-Box* (Continued)

BB-07	Notifikasi <i>Header</i> Orang Tua	Setelah tagihan dibuat, orang tua membuka dashboard.	Notifikasi pada bagian <i>header</i> halaman orang tua tampil sesuai kondisi (tagihan baru).	<i>Pass/Fail</i>
-------	------------------------------------	--	--	------------------

3.5 Evaluasi/Pengukuran (Kuesioner Likert)

Petunjuk Pengisian Kuesioner

Kuesioner ini digunakan untuk mengetahui penilaian responden (TU dan Bendahara) terhadap rancangan sistem informasi pembayaran sekolah. Responden diminta memberikan penilaian pada setiap pernyataan dengan memberi tanda (✓) pada salah satu pilihan berikut:

- **STS** = Sangat Tidak Setuju
- **TS** = Tidak Setuju
- **N** = Netral
- **S** = Setuju
- **SS** = Sangat Setuju

Skala penilaian menggunakan rentang 1–5, yaitu: STS(1), TS(2), N(3), S(4), dan SS(5)

Tabel 3.6: Kuesioner Evaluasi Sistem (Responden: TU dan Bendahara)

No	Pernyataan	STS(1)	TS(2)	N(3)	S(4)	SS(5)
1	Sistem mudah dipelajari saat pertama kali digunakan.					
2	Menu dan navigasi sistem mudah dipahami.					

Continued on next page

Tabel 3.6: Kuesioner Evaluasi Sistem (Responden: TU dan Bendahara) (Continued)

3	Proses input data (mis. siswa/tagihan) mudah dilakukan.				
4	Sistem membantu saya menyelesaikan pekerjaan lebih cepat.				
5	Informasi tagihan jelas (periode, nominal, jatuh tempo, status).				
6	Informasi status pembayaran mudah dipantau dan tidak membingungkan.				
7	Data yang ditampilkan sistem konsisten dengan data yang tersimpan.				
8	Pembuatan/penetapan tagihan siswa dapat dilakukan dengan mudah.				
9	Proses verifikasi pembayaran melalui kanal pembayaran mudah dilakukan.				
10	Perubahan status pembayaran membantu monitoring pembayaran.				
11	Pembaruan status pembayaran manual (fallback) mudah dan aman dilakukan.				
12	Sistem memudahkan pembuatan laporan pembayaran sesuai periode.				
13	Sistem memudahkan melihat daftar tunggakan (tagihan belum lunas).				

Continued on next page

Tabel 3.6: Kuesioner Evaluasi Sistem (Responden: TU dan Bendahara) (Continued)

14	Sistem membantu rekap pembayaran per siswa/kelas lebih cepat.					
15	Hak akses pengguna sesuai peran (TU/Bendahara) sudah tepat.					
16	Sistem membantu mengurangi risiko kesalahan input/perubahan data.					
17	Secara keseluruhan, saya puas terhadap rancangan/fitur sistem yang diusulkan.					

3.6 Jadwal Pelaksanaan

Jadwal atau linimasa menggunakan Gantt chart atau tabel.

Tabel 3.7: Ini adalah contoh tabel.

Aktivitas	Bulan ke-					
	1	2	3	4	5	6
Mengumpulkan data awal						
Mendesain sistem						
Mengembangkan sistem						
Menguji kinerja sistem						
Membuat laporan TA						

BAB 4

HASIL DAN ANALISIS

Bab ini berisi hasil-hasil penelitian dan analisis TA anda.

BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Subbab ini bukan berisi ringkasan pendahuluan atau ringkasan hasil saja, melainkan fokus utamanya adalah menjawab rumusan masalah dan/atau menjelaskan pencapaian kontribusi dari penelitian. Isinya harus langsung, ringkas namun padat, dan jalan ceritanya mengalir dari Bab 4 (Hasil dan Analisis).

5.2 Saran untuk Penelitian Selanjutnya

Subbab ini berisi rekomendasi-rekomendasi berdasarkan masalah atau keterbatasan yang ditemui selama pengerjaan TA dan berisi pengajuan bagaimana sebuah penelitian atau sistem dapat dikembangkan lebih lanjut oleh peneliti lain.

DAFTAR PUSTAKA

- Adham, M. F. (2024). Analisis implementasi sistem informasi: Studi literatur. *Jurnal Teknologi Sistem Informasi*, 5(1), 265.
- As, A., Hartono, N., & Muin, A. (2024, 12). Penggunaan user acceptance testing (uat) pada pengujian sistem informasi pengelolaan keuangan dan inventaris barang. *Switch : Jurnal Sains dan Teknologi Informasi*, 3, 84-100. doi: 10.62951/switch.v3i1.330
- Endang, & Susanti, M. V. (2021, September). Penerapan metode waterfall pada rancang bangun aplikasi p2db di smk dewantara cikarang menggunakan php dan mysql. *Jurnal Informatika SIMANTIK*, 6(2), 38.
- Hartomi, Z. H., Afsari, M., Rahmawati, Y., & Lendra, R. (2021, April). Pengembangan sistem informasi pembayaran sumbangan pembinaan pendidikan (spp) berbasis web menggunakan codeigniter studi kasus sdit. *Jurnal Ilmu Komputer*, 10(1), 1-7.
- Ikhsanudin, M. N., Pratama, R. E. B., & Maulindar, J. (2022, June). Sistem e-wallet untuk pembayaran dalam lingkup sekolah. -, 445-451.
- Irawan, N. A., Lianawati, Y., & Wibowo, A. (2023, February). Sistem informasi pembayaran biaya sekolah berbasis website dengan whatsapp gateway. *Jurnal Penelitian Rumpun Ilmu Teknik (JUPRIT)*, 2(1).
- Kartono, F. K., Nursaadah, S., Nugroho, M. R., Tama, D. A., Mashudi, F. A., Wicaksono, A., & Nasir, M. (2024, Dec.). Pengujian black box testing pada sistem website osha snack: Pendekatan teknik boundary value analysis. *Jurnal Kridatama Sains dan Teknologi*, 6(02), 754–766. Retrieved from <https://jurnal.umnu.ac.id/index.php/kst/article/view/1407> doi: 10.53863/kst.v6i02.1407
- Khasani, R. N. K., & Subrata, J. (2025, February). Perancangan sistem informasi pemesanan di camellia cafe rsu islam harapan anda kota tegal berbasis website. *Jurnal Mahasiswa Teknik Informatika*, 9(1), 438.
- Khayyirah, S., Kurniawan, R. A., & Gemilang, S. G. (2022, Jun.). Analisis penggunaan alat pembayaran non tunai pada tingkat mahasiswa universitas islam negeri mataram tahun 2021/2022. *SOCIETY*, 13(1), 7–17. Retrieved from <https://ftkjurnal-uinmataram.id/index.php/society/article/view/5297> doi: 10.20414/society.v13i1.5297
- Nasution, M. A., Suendri, & Harahap, A. M. (2023). Customer service information system using dynamic priority scheduling algorithm at pt sumatra sistem integrasi. *Journal of Information System and Technology Research*, 2, 27.

- Nitami, A., Munthe, A. A., & Masrizal. (2021). Sistem informasi reservasi hotel rantauprapat berbasis web dengan framework codeigniter. *Journal of Student Development Information System (JoSDIS)*, 1(1), 26.
- Palinggi, O., Maesaroh, S., Permana, M. B., Huda, D. F., & Priyono, K. A. (2024, May). Entity-relationship diagram technique in database. *Collabits Journal*, 1(2), 102.
- Rahayu, W. I., & Shafina, M. R. (2022). Aplikasi analisis kelayakan sistem untuk pengukuran usability dengan menerapkan metode use questionnaire. *Jurnal Teknik Informatika*, 14(3), 152-160.
- Rayadi, R., & Maradesa, D. (2021, July). Evaluasi sistem pembayaran non tunai pada pt. angkasa pura i (persero) bandar udara internasional sam ratulangi manado. *Jurnal EMBA*, 9(3), 193–200.
- Rizki, U., Jakak, P. M., Sari, p. C., & Satriadi, D. (2024, January). Rancang bangun website menejemen pembayaran sumbangan pembinaan pendidikan (spp) sekolah. *Jurnal Teknologi Informasi dan Pendidikan*, 1(1), 1-6.
- Sie, J. B. L., Musdar, I. A., & Bahri, S. (2022). Pengujian white box testing terhadap website room menggunakan teknik basis path. *JURNAL ILMU KOMPUTER KHARISMA*, 17(2).
- Sihaloho, J. E., Ramadani, A., & Rahmayanti, S. (2020, April). Implementasi sistem pembayaran quick response indonesia standard bagi perkembangan umkm di medan. *Jurnal Manajemen Bisnis*, 17(2), 287–297.
- Subhaktiyasa, P. G. (2024). Evaluasi validitas dan reliabilitas instrumen penelitian kuantitatif: Sebuah studi pustaka. *Journal of Education Research*, 5(4), 2299-5609.
- Suryanto, Muhyi, H. A., & Kurniati, P. S. (2022, April). Use of digital payment in micro, small and medium business. *Jurnal Pemikiran dan Penelitian Administrasi Bisnis dan Kewirausahaan*, 7(1), 55-65. doi: <https://doi.org/10.24198/adbispreneur.v7i1.39452>
- Ulfie, M., Marthasari, G. I., & Nuryasin, I. (2024, Jan.). Implementasi metode personal extreme programming dalam pengembangan sistem manajemen transaksi perusahaan (studi kasus : Cv. todjoe sinar group). *Jurnal Reppositor*, 2(3). Retrieved from <https://ejournal.umm.ac.id/index.php/repositor/article/view/30489> doi: 10.22219/repositor.v2i3.30489

LAMPIRAN

LAMPIRAN A

SIMBOL USE CASE DIAGRAM

Tabel A.1: Simbol *Use Case Diagram*

SIMBOL	NAMA	DESKRIPSI
	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi dari luar sistem.
	<i>Use Case</i>	Fungsi/layanan yang disediakan sistem untuk aktor.
	<i>Association</i>	Hubungan komunikasi antara <i>actor</i> dan <i>use case</i> .
	<i>Include</i>	Relasi ketika suatu <i>use case</i> selalu memanggil <i>use case</i> lain sebagai bagian prosesnya.
	<i>Extend</i>	Relasi perluasan yang bersifat opsional pada kondisi tertentu.
	<i>Generalization</i>	Hubungan generalisasi/spesialisasi antar aktor atau antar <i>use case</i> .

LAMPIRAN B

SIMBOL ENTITY-RELATIONSHIP DIAGRAM

Tabel B.1: Simbol *Entity-Relationship Diagram*

SIMBOL	NAMA	DESKRIPSI
	<i>Entity</i>	Objek yang dapat diidentifikasi dalam lingkungan sistem.
	<i>Relation</i>	Menunjukkan hubungan di antara entitas yang berbeda.
	<i>Attribute</i>	Karakteristik/atribut yang mendeskripsikan entitas.
	<i>Line</i>	Penghubung antara entitas dengan atribut/relasi.

LAMPIRAN C

SIMBOL ACTIVITY DIAGRAM

Tabel C.1: Simbol *Activity Diagram*

SIMBOL	NAMA	DESKRIPSI
	<i>Start Node</i>	Status awal aktivitas sistem.
	<i>Activity</i>	Aktivitas yang dilakukan sistem.
	<i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	<i>Join Fork</i>	Menunjukkan kegiatan yang dilakukan secara paralel.
	<i>End Node</i>	Status akhir yang dilakukan sistem.