

PROCEEDING NATIONAL CONFERENCE OF RESEARCH AND COMMUNITY SERVICE SISI INDONESIA 2025

<https://conference.sinesia.id/ncrcs-sinesia>

Modul *Billing* Berbasis Web untuk Pengelolaan *Invoice* di PT XYZ dengan *Laravel*

Web-Based Billing Module for Invoice Management at PT XYZ Using Laravel

Abiyyu Umar Thoyyib¹, Ahmad Ridha², Wendi Maksalmina³

abiyyuumar@apps.ipb.ac.id¹, ridha@apps.ipb.ac.id², wendi.maxalmina@telkomsat.co.id³

¹Program Studi Teknologi Rekayasa Perangkat Lunak, Sekolah Vokasi,
IPB University, Bogor, Indonesia

²Program Studi Ilmu Komputer, Sekolah Sains Data, Matematika, dan Informatika,
IPB University, Bogor, Indonesia

³Subunit Teknologi Informasi, PT Telkom Satelit Indonesia, Bogor, Indonesia

Info Article

| Submitted: 20 May 2025 | Revised: 4 June 2025 | Accepted: 9 June 2025

How to cite: Abiyyu Umar Thoyyib, etc., "Modul *Billing* Berbasis Web untuk Pengelolaan *Invoice* di PT XYZ dengan *Laravel*", *Proceeding National Conference Of Research And Community Service Sisi Indonesia*, 2025, P. 147-161.

ABSTRACT

An IT Managed Service company, PT XYZ, still uses a desktop-based billing module that limits its access, because making changes to the application requires reinstallation. This study develops a web-based billing module using the Laravel framework to digitalize the creation and management of invoices, as well as to provide data summaries in Excel format. The development methodology adopted is Extreme Programming (XP) due to its flexibility in responding to changing requirements and its emphasis on intensive collaboration between developers and users. Requirements were gathered through discussions and workflow analysis with the billing team, followed by the design of use cases, activity diagrams, and entity relationship diagram. The system has been tested using User Acceptance Testing (UAT), and the results showed that all features – automated invoice generation, centralized data storage, and Excel report export – functioned as expected. This implementation successfully reduces the risk of manual errors, accelerates administrative processes, and strengthens the company's digital culture.

Keyword: Billing, Extreme Programming, Laravel, User Acceptance Testing

ABSTRAK

Sebuah perusahaan IT Managed Service, PT XYZ, masih menggunakan modul *billing* berbasis *desktop* sehingga mengakibatkan aksesnya terbatas, karena jika ingin melakukan perubahan pada aplikasi membutuhkan instalasi ulang. Penelitian ini mengembangkan sebuah modul *billing* berbasis web dengan *framework* Laravel untuk digitalisasi pembuatan dan pengelolaan *invoice*, serta menyajikan rekap data dalam format Excel. Metodologi pengembangan yang digunakan adalah *Extreme Programming* (XP) karena fleksibilitasnya dalam merespon perubahan kebutuhan dan mendorong kolaborasi intensif antara tim pengembang dan pengguna. Pengumpulan kebutuhan dilakukan melalui diskusi dan analisis alur kerja tim *billing*, dilanjutkan dengan perancangan *use case*, *flowchart diagram*, dan *entity relationship diagram*. Sistem telah diuji menggunakan *User Acceptance Testing* (UAT) dan menunjukkan bahwa semua fitur (pembuatan *invoice* otomatis, penyimpanan data terpusat, dan ekspor laporan Excel) berjalan sesuai ekspektasi. Implementasi ini dapat mengurangi risiko kesalahan manual, mempercepat proses administrasi, dan memperkuat budaya digital perusahaan.

Kata Kunci: Billing, Extreme Programming, Laravel, User Acceptance Testing

Pendahuluan

PT XYZ adalah perusahaan *Information Technology Managed Service* yang memberikan layanan ke pelanggan korporasi. Modul *billing* yang digunakan masih



This work is licensed under [CC BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/)

berbasis sistem *desktop* sehingga hanya dapat diakses dari perangkat tertentu sementara aplikasi lainnya di perusahaan sudah beralih ke platform web dan *mobile*. Pembuatan *invoice* pada sistem yang sekarang juga belum otomatis dihasilkan berdasarkan layanan yang dilanggan.

Ketergantungan pada metode yang belum otomatis berdampak signifikan pada penurunan produktivitas dan kepuasan pelanggan¹. Transformasi digital melalui penerapan teknologi informasi mampu meningkatkan efisiensi pengelolaan data dan memperkuat budaya inovasi². Digitalisasi proses *billing* berbasis web menggunakan *Laravel* dan pendekatan *Extreme Programming* secara signifikan dapat mengurangi kesalahan manual serta meningkatkan efisiensi organisasi³. Hasil penelitian sebelumnya menunjukkan bahwa aplikasi berbasis web secara efektif mengelola data *invoice*, mengatasi kompleksitas penagihan, dan memfasilitasi pembayaran tepat waktu, yang secara langsung menyelesaikan masalah penundaan dan ketidakakuratan dalam proses penagihan tradisional yang dihadapi oleh vendor⁴.

Penelitian ini mengembangkan modul *billing* berbasis *Laravel* untuk mengotomatisasi pembuatan *invoice* serta menyajikan rekap data dalam format Excel. Pemilihan *Laravel* didasari kemampuannya menyediakan arsitektur modern, keamanan tinggi, dan kemudahan integrasi fitur⁵. Metodologi *Extreme Programming* diterapkan karena fleksibilitasnya dalam merespon perubahan kebutuhan pengguna serta mendorong kolaborasi intensif antara tim pengembang dan tim *billing*. Pengumpulan kebutuhan sistem dilakukan melalui diskusi mendalam dan analisis alur kerja tim, dilanjutkan dengan perancangan *use case diagram*, *activity diagram*, dan *Entity Relationship Diagram* (ERD) sebagai landasan pengembangan.

Pengujian sistem menggunakan *User Acceptance Testing* (UAT) membuktikan bahwa seluruh fitur inti—seperti pembuatan *invoice* otomatis, penyimpanan data

¹ Paramanantham, Shampave, dan Sidath Liyanage. 2022. "Assessing the Impact of Human Error Assessment on Organization Performance in the Software Industry." *International Journal of Information Systems and Social Change* 14 (1): 1–32. <https://doi.org/10.4018/ijissc.314563>.

² Cöster, Mathias, Mats Danielson, Love Ekenberg, Cecilia Gullberg, Gard Titlestad, Alf Westelius, dan Gunnar Wettergren. 2023. *Digital Transformation*. Cambridge, UK: Open Book Publishers. <https://doi.org/10.11647/OBP.0350>.

³ Abdullah, Moch Zawaruddin, Mamluatul Hani'ah, Yoppy Yunhasnawa, dan Rokhimatul Wakhidah. 2024. "Design and Implementation KP-SPAMS Transaction Information System utilizing *Laravel* Framework and *Extreme Programming* Methodology." *Journal of Informatics Information System Software Engineering and Applications (INISTA)* 7 (1): 51–67. <https://doi.org/10.20895/inista.v7i1.1645>.

⁴ Wiratama, Jansen, Santo Fernandi Wijaya, Hari Santoso, dan Jonathan. 2023. "Improving Invoice Management: A Web-based Application for the Hospitality IT Vendor Industry using Prototyping Model." *Jurnal Informatika Ekonomi Bisnis*, Desember, 1501–6. <https://doi.org/10.37034/infeb.v5i4.752>.

⁵ Sari, Purma. 2018. "Development of Information Systems Using *Extreme Programming* Method." *Inform : Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi* 3 (2): 95–99. <https://doi.org/10.25139/inform.v3i2.968>.

terpusat, dan ekspor laporan Excel—berjalan sesuai ekspektasi pengguna. Implementasi sistem ini tidak hanya mengurangi risiko kesalahan manual dan mempercepat proses administrasi, tetapi juga menjadi fondasi bagi pengembangan budaya digital di perusahaan *internet service provider* serta integrasi sistem yang lebih kompleks di masa depan.

Metode Penelitian

Metode pengembangan perangkat lunak yang digunakan adalah *Extreme Programming* (XP), salah satu pendekatan dalam *Agile Development* yang dirancang untuk menghadapi perubahan kebutuhan secara cepat dan terstruktur. XP dipilih karena kemampuannya dalam mendukung siklus iteratif, kolaborasi intensif dengan pengguna, serta pengujian berkelanjutan sepanjang proses pengembangan⁶.

XP pertama kali diperkenalkan oleh Kent Beck pada akhir 1990-an sebagai metode yang mengedepankan adaptabilitas dan komunikasi efektif dalam tim pengembang⁷. Metodologi ini terdiri atas lima tahap utama: *planning*, *design*, *coding*, *testing*, dan *release*, yang dijalankan secara berulang untuk menghasilkan sistem yang stabil, fleksibel, dan sesuai dengan kebutuhan pengguna⁸.

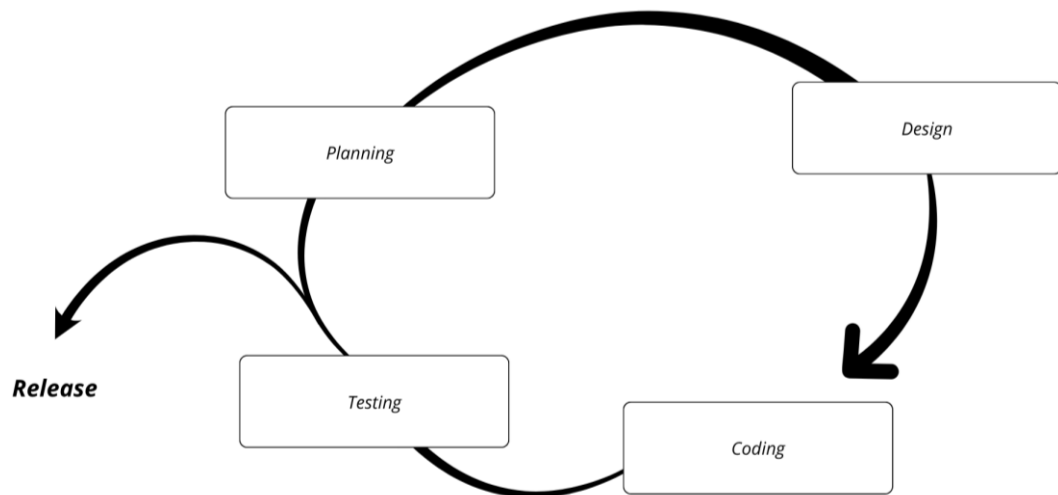
Dalam konteks penelitian ini, XP memungkinkan tim pengembang untuk menyesuaikan fitur secara bertahap berdasarkan umpan balik langsung dari tim *billing* PT XYZ. Pendekatan iteratif ini mempercepat validasi sistem dan meminimalisir risiko pengembangan. Gambar 1 menyajikan alur proses pengembangan sistem berdasarkan metodologi *Extreme Programming*.

Analisis dan pengembangan sistem dilakukan melalui empat tahap utama sesuai dengan metodologi XP. Tahap *planning* diawali dengan pengumpulan *user stories* untuk menentukan prioritas pengembangan fitur. Diskusi dan wawancara dilakukan dengan tim *billing* PT XYZ untuk mengidentifikasi kebutuhan fungsional dan non-fungsional sistem. Dokumen internal perusahaan digunakan sebagai referensi alur kerja *billing*, spesifikasi teknis infrastruktur TI. Selanjutnya, tahap *design* difokuskan pada visualisasi arsitektur sistem melalui pembuatan *activity diagram*, *use case diagram*, dan ERD.

⁶ Prasetyo, Deni, Annisaa Utami, dan Tri Ginanjar Laksana. 2024. "Website Based Academic Information System Design Using Extreme Programming Method." *Journal of Informatics Information System Software Engineering and Applications (INISTA)* 6 (2): 134–43. <https://doi.org/10.20895/inista.v6i2.1214>.

⁷ Kent Beck, Chyntia Andres. 2004. "Praise for Extreme Programming Explained, Second Edition." https://www.researchgate.net/publication/277072162_Extreme_Programming_Explained_Embrace_Change_2nd_Edition.

⁸ Susila, Mochamad Nandi, Sriyadi Sriyadi, Maruloh Maruloh, Ardian Dwi Praba, Raihan Pramusetto, dan Agustina Afriani. 2021. "Manajemen Donasi Digital Dengan Extreme Programming." *Jurnal INSAN: Journal of Information System Management Innovation* 1 (2): 112–19. <https://doi.org/10.31294/jinsan.v1i2.783>.



Gambar 1 Tahapan *Extreme Programming*

Tahap *coding* menggunakan *framework* Laravel, dengan menerapkan pola *Model-View-Controller* (MVC) untuk menjaga struktur kode yang modular dan mudah dikembangkan. Terakhir, tahap *testing* menggunakan *User Acceptance Testing* (UAT) guna memverifikasi kesesuaian sistem dengan kebutuhan pengguna.

Proses UAT mencakup perencanaan, penyusunan skenario uji, pelaksanaan pengujian, serta evaluasi hasil oleh pengguna akhir. UAT memastikan bahwa sistem yang dikembangkan benar-benar memenuhi ekspektasi dan mendukung proses bisnis sebagaimana diharapkan⁹.

Hasil dan pembahasan

1.1 Perencanaan

Pada tahap perencanaan, informasi dikumpulkan melalui diskusi langsung dengan tim *billing* terkait proses bisnis yang digunakan sekarang dan dampak yang terjadi. Dari diskusi tersebut, ditemukan bahwa modul *billing* sebelumnya yang masih berbasis desktop memiliki beberapa kelemahan, seperti ketergantungan pada perangkat tertentu, kesulitan dalam pembaruan sistem, serta terbatasnya akses jika digunakan di luar kantor. Hal ini menyulitkan tim dalam melakukan kolaborasi maupun pelacakan data secara *real-time*. Dari temuan tersebut, beberapa kebutuhan untuk proyek ini teridentifikasi yang mencakup kebutuhan perangkat keras, kebutuhan perangkat lunak, kebutuhan fungsional, dan kebutuhan non-fungsional yang disajikan pada Tabel 1, 2, 3, dan 4.

⁹ Kusum, Pritika Talwar, Amit Puri, dan Guneet Kumar. 2024. "Overview of software testing." *Global Journal of Engineering and Technology Advances* 19 (1): 104–12. <https://doi.org/10.30574/gjeta.2024.19.1.0060>.

Tabel 1 Kebutuhan perangkat keras

No	Komponen	Spesifikasi
1	Processor	Intel Core i5-1135G7
2	RAM	16 GB
3	Penyimpanan	512 GB SSD
4	Kartu Grafis	Geforce GTX 1650

Tabel 2 Kebutuhan perangkat lunak

No	Software	Versi	Kegunaan
1	Visual Studio Code	1.95	Lingkungan pengembangan
2	DBeaver	24.0.5	Antarmuka manajemen basis data
3	Gitlab	15.8.1-ee	Manajemen versi
4	Google Chrome	131.0.6778.109	<i>Web browser</i>

Tabel 3 Kebutuhan fungsional

No	Nama Fungsi	Keterangan
1	Kontrak	• Pengguna dapat melihat daftar kontrak • Pengguna dapat memasukkan nomor VA • Pengguna dapat memasukkan nomor rekening • Pengguna dapat melihat data kontrak
2	<i>Invoice</i>	• Pengguna dapat melihat daftar <i>invoice</i> • Pengguna dapat mengekspor daftar <i>invoice</i> ke Excel • Pengguna dapat melihat data <i>invoice</i> • Pengguna dapat <i>generate invoice</i> • Pengguna dapat <i>generate proforma invoice</i> • Pengguna dapat mengunggah <i>invoice</i> • Pengguna dapat mengunggah faktur pajak • Pengguna dapat ubah nama layanan • Pengguna dapat ubah data <i>invoice</i>

Tabel 4 Kebutuhan non-fungsional

No	Parameter	Deskripsi
1	Kegunaan	Antarmuka ramah pengguna dan dapat dipahami
2	Kinerja	Kinerja sistem yang cepat
3	Keamanan	Sistem memiliki autentikasi untuk akses pengguna

1.2 Desain

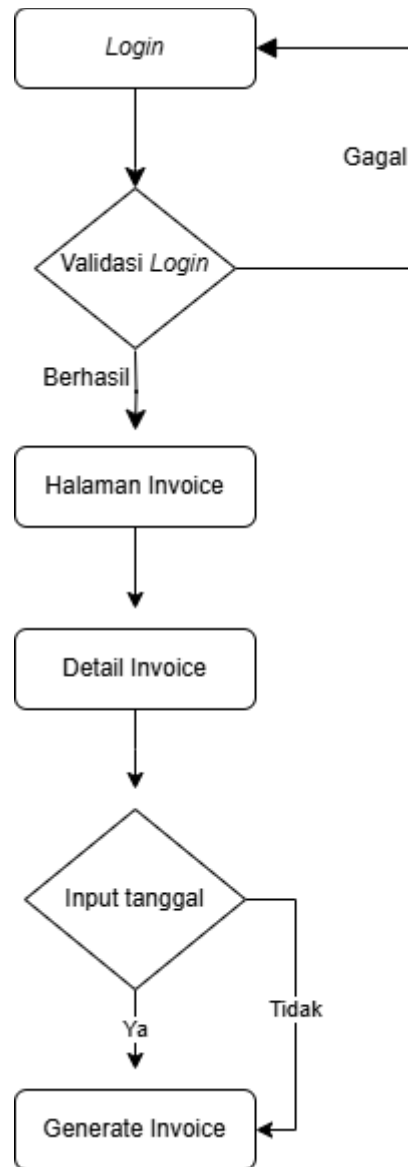
Desain sistem difokuskan pada visualisasi alur kerja dan perancangan struktur data untuk memastikan pengembangan serta pemeliharaan sistem yang efisien. Diagram *use case* menggambarkan interaksi antara pengguna dan sistem, mencakup fitur utama seperti autentikasi (*login*), manajemen data kontrak, dan pembuatan *invoice*. Diagram ini membantu mengidentifikasi kebutuhan fungsional dari perspektif pengguna.

Dalam perancangan basis data, *ERD* berperan penting untuk memetakan hubungan antar-entitas seperti *User*, *Invoice*, dan kontrak. *ERD* ini tidak hanya menjadi acuan dalam pembuatan struktur tabel basis data, tetapi juga memudahkan dalam dokumentasi dan komunikasi antarpengembang. Dengan visualisasi relasi yang jelas, proses integrasi antar-modul sistem dapat dilakukan secara konsisten dan minim kesalahan¹⁰.

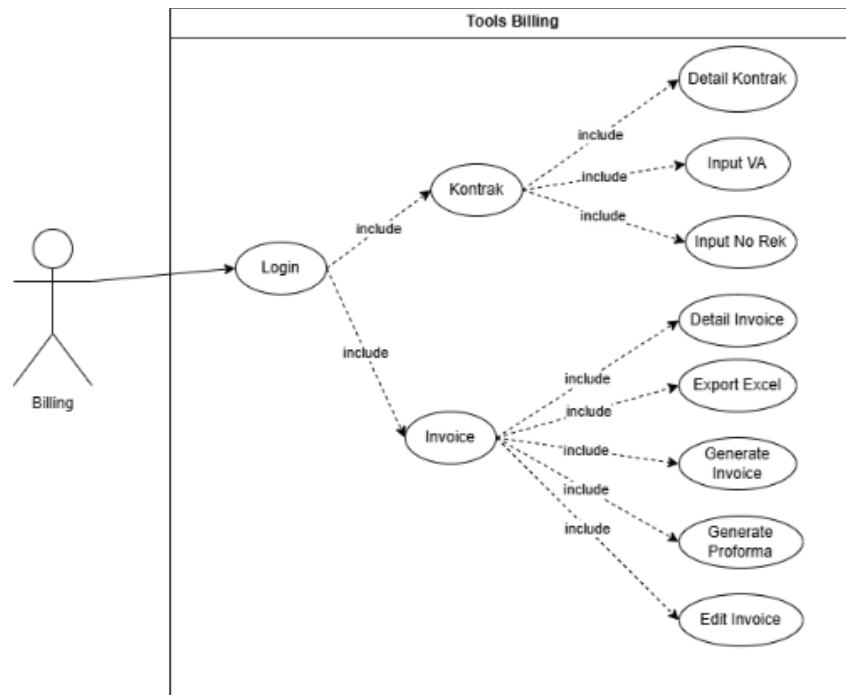
Gambar 2 menampilkan *flowchart* dari sistem *billing* yang dikembangkan. Diagram ini menggambarkan alur logika dari proses bisnis yang ditransformasi ke dalam sistem, mulai dari *login*, validasi, proses *generate invoice*, hingga ekspor data. *Flowchart* ini memberikan visualisasi menyeluruh terhadap bagaimana sistem merespon tindakan pengguna dan mengelola data secara berurutan.

Use case diagram yang disajikan pada Gambar 3 menggambarkan interaksi antara pengguna sistem (aktor) dengan berbagai fungsi utama yang tersedia. Pengguna dapat melakukan *login*, mengelola data kontrak, mengakses informasi *invoice*, dan mengekspor data. Dengan demikian, kebutuhan fungsional sistem dapat dipetakan secara jelas dari perspektif pengguna.

¹⁰ Trisnadoli, Anggy. 2021. "Implementasi Extreme Programming (XP) Agile Software Development pada Pengembangan Sistem Informasi KELUARGAKU" 6 (2): 305–11. <https://doi.org/10.32493/informatika.v6i2.10088>.



Gambar 2 *Flowchart modul billing*



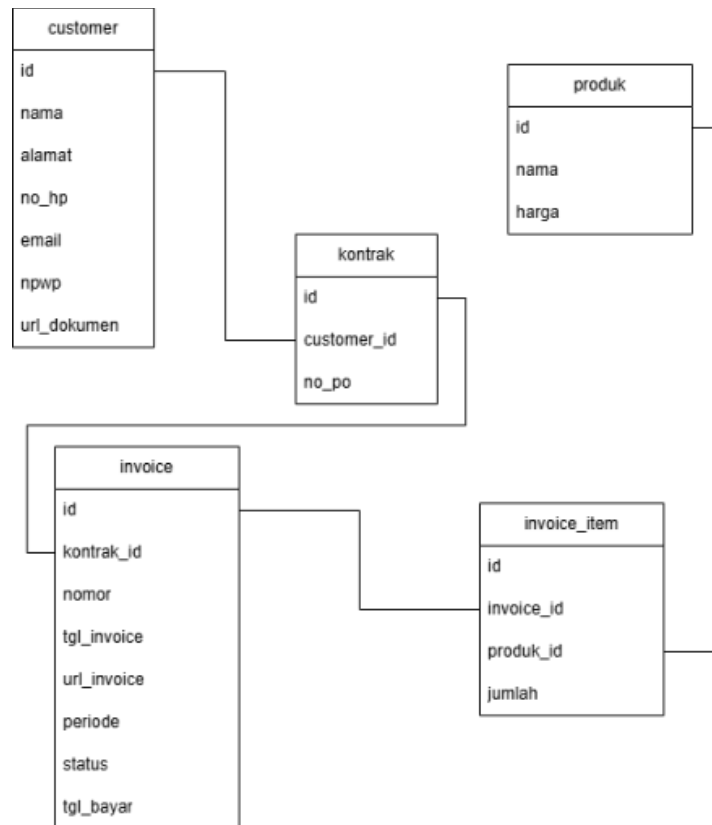
Gambar 3 Use case diagram

Gambar 4 menunjukkan hubungan antar-entitas utama dalam sistem *billing*, yaitu *User*, kontrak, dan *Invoice*. Entitas *User* memiliki relasi *one to many* terhadap kontrak, yang berarti satu pengguna dapat memiliki beberapa kontrak. Selanjutnya, artinya setiap kontrak juga dapat memiliki banyak *Invoice*, mencerminkan struktur tagihan berulang atau multi-layanan yang umum pada sistem *billing* PT XYZ.

Entitas *Invoice* memiliki atribut tambahan seperti tanggal, status, serta referensi dokumen seperti faktur pajak dan nomor VA, yang mendukung kebutuhan administratif perusahaan. Struktur ini dirancang untuk menjaga integritas data dan memudahkan proses *query* pada tahap ekspor laporan ke Excel. ERD menjadi dasar pengembang dalam menyusun skema *database* dan memastikan relasi antar-tabel sesuai dengan kebutuhan fungsional sistem.

1.3 Pengkodean

Sistem diimplementasikan menggunakan *framework* Laravel yang mengintegrasikan *backend* dan *frontend* dalam satu kesatuan arsitektur yang efisien. Pada sisi *backend*, struktur data didefinisikan melalui model seperti *User*, *Invoice*, dan kontrak yang merepresentasikan entitas utama dalam sistem. Setiap model berelasi dengan *controller* yang menangani logika bisnis, seperti proses autentikasi, pengelolaan kontrak, serta pembuatan dan pengolahan *invoice*. *Controller* ini juga bertugas menangani permintaan dari antarmuka pengguna dan merespon dengan data yang sesuai.

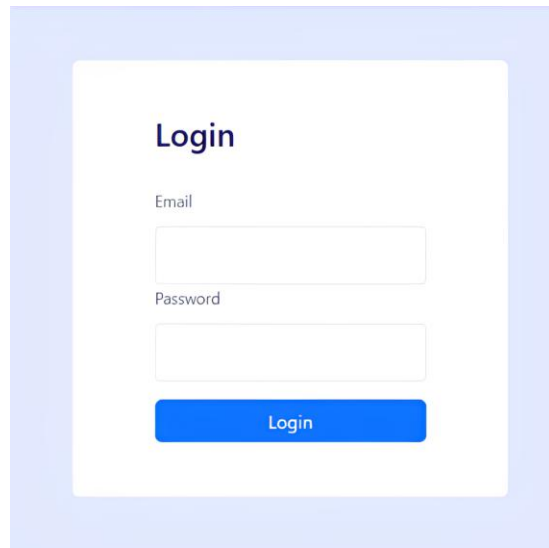


Gambar 4 ERD

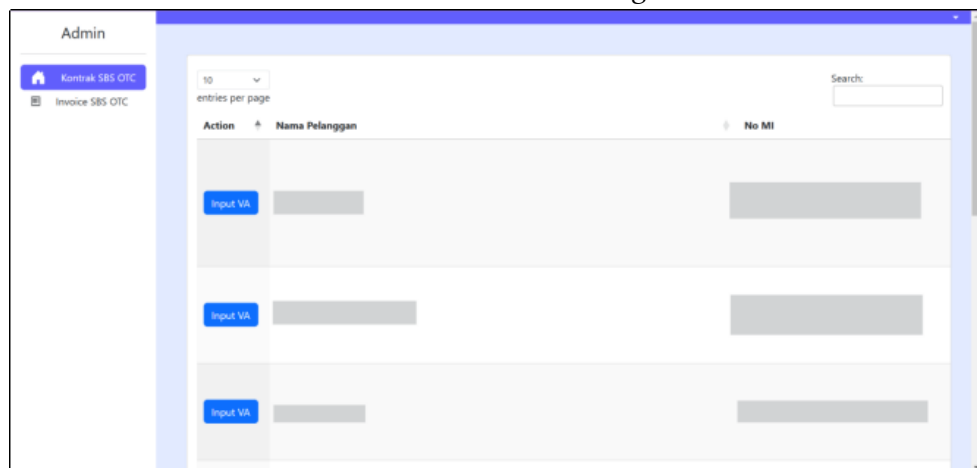
Pada sisi *frontend*, antarmuka dirancang agar ramah pengguna dan mudah dinavigasi. Halaman *login* dilengkapi dengan validasi *form* yang sederhana namun efektif untuk memastikan keamanan dan kenyamanan akses. Setelah *login*, pengguna diarahkan ke *dashboard* yang menyajikan navigasi menuju modul kontrak dan *invoice*. Di halaman *invoice*, pengguna dapat melihat daftar *invoice* yang telah dibuat, mengedit *invoice*, serta mengeksport data ke format Excel. Selain itu, sistem menyediakan fitur *generate invoice* dan *proforma invoice*, lengkap dengan isian tanggal serta pratinjau dalam bentuk PDF sebelum disimpan atau dikirim. Keseluruhan antarmuka dirancang untuk mendukung efisiensi kerja tim *billing* dan meminimalkan potensi kesalahan.

Gambar 5 menampilkan antarmuka halaman *login*. Halaman ini dirancang sederhana dan intuitif dengan fitur validasi dasar. Pengguna diminta untuk memasukkan NIK dan kata sandi sebelum mendapatkan akses ke *dashboard* utama. Desain ini mendukung keamanan dan kenyamanan dalam proses autentikasi pengguna.

Gambar 6 menampilkan halaman kontrak yang memungkinkan pengguna untuk melihat, menambahkan, dan memperbarui informasi terkait kontrak. Tim *billing* mengisi data seperti nomor *Virtual Account* (VA) dan nomor rekening, yang nantinya terhubung langsung dengan proses pembuatan *invoice*.



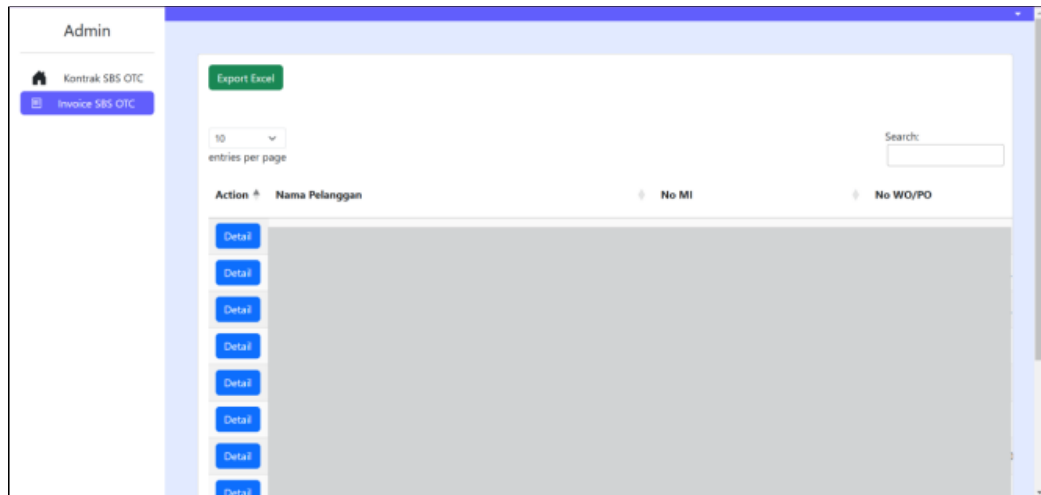
Gambar 5 Halaman *login*



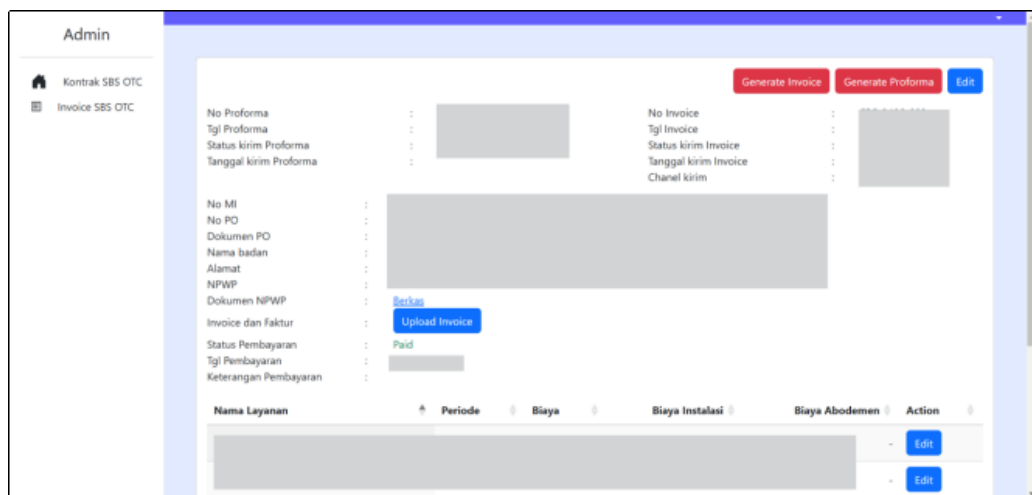
Gambar 6 Halaman kontrak

Halaman *invoice* pada Gambar 7 menampilkan daftar *invoice* yang telah dibuat, serta menyediakan fitur seperti ekspor data ke format Excel dan *generate invoice* baru. Gambar ini mencerminkan fungsionalitas utama dari sistem, yaitu digitalisasi proses *billing*.

Gambar 8 menyajikan halaman detail *invoice* yang menampilkan informasi tagihan secara lengkap dan memungkinkan pengguna untuk melihat atau mengedit isi *invoice*. Halaman tersebut juga menyediakan fitur untuk mengunggah dokumen seperti faktur pajak dan memodifikasi informasi layanan.



Gambar 7 Halaman *invoice*



Gambar 8 Halaman detail *invoice*

1.4 Pengujian

Pengujian dilakukan melalui User Acceptance Testing (UAT) oleh tim billing, dengan hasil yang ditampilkan pada Tabel 5. Delapan skenario uji mencakup fitur utama seperti *login*, *input* nomor VA, pembuatan *invoice*, dan ekspor ke Excel. Berdasarkan hasil pengujian metode UAT, seluruh fitur berjalan sesuai ekspektasi. UAT juga menunjukkan bahwa sistem dapat mengurangi kesalahan manual dan mempercepat proses administrasi¹¹. Misalnya, pengguna dapat masuk ke sistem, melihat daftar kontrak, mengisi nomor VA, serta membuat dan mengedit *invoice* tanpa kendala.

Tabel 5 Hasil pengujian

¹¹ Karima, Nida, dan Defri Kurniawan. 2025. "Design and Construction of a Website-Based Tourist Bus Rental System Using the Extreme Programming Method." *INOVTEK Polbeng - Seri Informatika* 10 (1): 61–72. <https://doi.org/10.35314/2ck7ea61>.

No	Kasus pengujian	Hasil yang diharapkan	Hasil
1	User login ke Modul Billing	User berhasil login	Diterima
2	User memilih menu kontrak	Menampilkan daftar kontrak	Diterima
3	User input Nomor VA	Berhasil memasukkan Nomor VA	Diterima
4	User memilih menu invoice	Menampilkan daftar invoice	Diterima
5	User memilih menu export Excel	Berhasil export Excel	Diterima
6	User memilih menu generate invoice	Berhasil membuat invoice	Diterima
7	User memilih menu generate proforma invoice	Berhasil membuat proforma invoice	Diterima
8	User memilih menu edit invoice	Data invoice berhasil diedit	Diterima

1.5 Kekurangan Modul

Modul yang dikembangkan telah memenuhi kebutuhan perusahaan saat ini. Namun, modul ini belum mendukung pembuatan *recurring invoice* sehingga *invoice* harus diinisiasi oleh bagian *billing* walaupun dokumennya kemudian dihasilkan secara otomatis. Selain itu, modul ini juga belum memiliki fitur pengecekan *billing* untuk pelanggan dan fitur notifikasi. Fitur pengecekan dapat membantu untuk mencegah perselisihan terkait besaran tagihan. Fitur notifikasi akan berguna untuk perusahaan dan pelanggan untuk meminimalkan keterlambatan pembayaran karena faktor lupa.

Penutup

Sebagai bagian dari upaya digitalisasi proses bisnis, modul *billing* berbasis web untuk PT XYZ telah berhasil dikembangkan menggunakan *framework* Laravel. Sistem ini menggantikan aplikasi *desktop* yang sebelumnya rawan kesalahan dan memakan waktu, sehingga mampu meningkatkan kecepatan, ketepatan, dan konsistensi data dalam kegiatan penagihan. Fitur utama yang diimplementasikan

antara lain pembuatan dokumen *invoice* otomatis dan rekap data dalam format Excel, yang tidak hanya mempercepat kinerja tim *billing* tetapi juga mempermudah penyusunan laporan keuangan untuk manajemen. Seluruh proses *billing* kini dapat dilakukan secara lebih efisien dan terdokumentasi dengan baik. Penerapan modul ini membuktikan bahwa Laravel dapat menjadi solusi efektif dalam pengembangan sistem informasi untuk mendukung kebutuhan bisnis lokal, serta membantu PT XYZ meningkatkan produktivitas operasional dan pengambilan keputusan berbasis data dalam menghadapi tantangan era digital.

Saran

Untuk meningkatkan kinerja dan fungsionalitas sistem *billing* berbasis web yang telah dikembangkan, sistem dapat dikembangkan lebih lanjut dengan fitur pengiriman email otomatis kepada pelanggan setiap kali *invoice* berhasil di-generate. Hal ini bertujuan untuk mempercepat proses penyampaian informasi tagihan dan mengurangi risiko keterlambatan pembayaran. Selain itu, pengembangan sistem berikutnya dapat mengarah ke fitur pengingat otomatis yang akan mengirimkan notifikasi kepada pelanggan sebelum dan sesudah jatuh tempo pembayaran. Dengan adanya fitur ini, perusahaan dapat lebih proaktif dalam proses penagihan, sehingga arus kas tetap terjaga dan risiko piutang macet dapat diminimalkan.

Ucapan Terima Kasih

Penulis mengucapkan terima kasih yang sebesar-besarnya kepada perusahaan IT *Managed Service* yang telah membantu, khususnya tim terkait yang telah memberikan dukungan penuh selama proses penelitian dan pengembangan modul *billing* berbasis web ini. Partisipasi aktif dalam diskusi kebutuhan sistem, uji coba fitur, serta masukan-masukan yang konstruktif sangat membantu dalam menyempurnakan hasil akhir dari sistem yang dikembangkan.

Daftar Pustaka

- Abdullah, Moch Zawaruddin, Mamluatul Hani'ah, Yoppy Yunhasnawa, dan Rokhimatul Wakhidah. 2024. "Design and Implementation KP-SPAMS Transaction Information System utilizing Laravel Framework and Extreme Programming Methodology." *Journal of Informatics Information System Software Engineering and Applications (INISTA)* 7 (1): 51-67. <https://doi.org/10.20895/inista.v7i1.1645>.
- Carolina, Irmawati, and Adi Supriyatna. 2019. "Penerapan Metode Extreme Programming Dalam Perancangan Aplikasi Perhitungan Kuota SKS Mengajar

- Dosen." IKRA-ITH INFORMATIKA 3 (1): 106–13. <https://journals.upi-yai.ac.id/index.php/ikraith-informatika/issue/view/2>.
- Cöster, Mathias, Mats Danielson, Love Ekenberg, Cecilia Gullberg, Gard Titlestad, Alf Westelius, dan Gunnar Wettergren. 2023. "Digital Transformation. Cambridge." UK: Open Book Publishers. <https://doi.org/10.11647/OBP.0350>.
- Juniarta, I Wayan, I Nyoman Yudi Anggara Wijaya, dan A.A Istri Ita Paramitha. 2024. "Rancang Bangun Sistem Informasi Restitusi dan Izin Siswa Menggunakan Metode Extreme Programming SMAN 1 Bebandem." *Jurnal Sosial Teknologi* 4 (6): 273–84. <https://doi.org/10.59188/jurnalsostech.v4i6.1238>.
- Karima, Nida, dan Defri Kurniawan. 2025. "Design and Construction of a Website-Based Tourist Bus Rental System Using the Extreme Programming Method." *INOVTEK Polbeng - Seri Informatika* 10 (1): 61–72. <https://doi.org/10.35314/2ck7ea61>.
- Kent Beck, Chyntia Andres. 2004. Praise for Extreme Programming Explained, Second Edition. 2nd ed. Addison-Wesley.
- Kusum, Pritika Talwar, Amit Puri, dan Guneet Kumar. 2024. "Overview of software testing." *Global Journal of Engineering and Technology Advances* 19 (1): 104–12. <https://doi.org/10.30574/gjeta.2024.19.1.0060>.
- Lestari, Widya, Lidya Wati, dan Eva Yumami. 2025. "Penerapan Metode Extreme Programming Pada Sistem Penerimaan Bantuan Langsung Tunai." *JEKIN - Jurnal Teknik Informatika* 5 (1): 70–90. <https://doi.org/10.58794/jekin.v5i1.926>.
- Nuurussa'adah, Fathimah, Reza Maulana, dan Bambang Harie Wiyono. 2024. "Penerapan Metode Extreme Programming menggunakan Framework Laravel dalam Pengembangan Sistem Informasi Khazaregsys." *DBESTI: Journal of Digital Business and Technology Innovation* 1 (2): 78–85. <https://doi.org/10.54914/dbesti.v1i2.1367>.
- Paramanantham, Shampave, dan Sidath Liyanage. 2022. "Assessing the Impact of Human Error Assessment on Organization Performance in the Software Industry." *International Journal of Information Systems and Social Change* 14 (1): 1–32. <https://doi.org/10.4018/ijjssc.314563>.
- Pramudya, Bintang, Dinda Chesar Putri Ramadhani, Hilda Nuzulul Mujaddidah, dan Risqy Siwi Pradini. 2025. "Implementation of Extreme Programming (XP) in the Development of Dental Clinic Information Systems." *Journal of Enhanced Studies in Informatics and Computer Applications* 2 (1): 20–28. <https://doi.org/10.47794/jesica.v2i1.22>.
- Prasetyo, Deni, Annisaa Utami, dan Tri Ginanjar Laksana. 2024. "Website Based Academic Information System Design Using Extreme Programming Method."

Journal of Informatics Information System Software Engineering and Applications (INISTA) 6 (2): 134–43. <https://doi.org/10.20895/inista.v6i2.1214>.

Sari, Purma. 2018. "Development of Information Systems Using Extreme Programming Method." *Inform: Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi* 3 (2): 95–99. <https://doi.org/10.25139/inform.v3i2.968>.

Susila, Mochamad Nandi, Sriyadi Sriyadi, Maruloh Maruloh, Ardian Dwi Praba, Raihan Pramuseto, dan Agustina Afriani. 2021. "Manajemen Donasi Digital Dengan Extreme Programming." *Jurnal INSAN: Journal of Information System Management Innovation* 1 (2): 112–19. <https://doi.org/10.31294/jinsan.v1i2.783>.

Trisnadoli, Anggy. 2021. "Implementasi Extreme Programming (XP) Agile Software Development pada Pengembangan Sistem Informasi KELUARGAKU" 6 (2): 305–11. <https://doi.org/10.32493/informatika.v6i2.10088>.

Wiratama, Jansen, Santo Fernandi Wijaya, Hari Santoso, dan Jonathan. 2023. "Improving Invoice Management: A Web-based Application for the Hospitality IT Vendor Industry using Prototyping Model." *Jurnal Informatika Ekonomi Bisnis*, Desember, 1501–6. <https://doi.org/10.37034/infec.v5i4.752>.

Biografi Singkat Penulis



Abiyyu Umar Thoyyib lahir di Depok pada 16 Juni 2003. Saat ini, ia merupakan mahasiswa semester 8 pada Program Studi Sarjana Terapan Teknologi Rekayasa Perangkat Lunak di IPB University. Sejak kelas 1 SMA, dia telah menekuni bidang teknologi informasi (TI) dan telah mengembangkan beberapa aplikasi berbasis web, salah satunya adalah sistem manajemen siswa di pesantren Quadrant Boarding School. Dia juga memiliki pengalaman magang di PT Telkom Satelit Indonesia.



Ahmad Ridha dilahirkan di Jakarta pada 7 Mei 1980. Gelar sarjana ilmu komputernya diraih di Institut Pertanian Bogor pada 2002. Selanjutnya gelar S2 ilmu komputernya diselesaikan di King Fahd University of Petroleum and Minerals, Arab Saudi pada 2008. Bidang minatnya mencakup temu kembali informasi, teknologi pendidikan, dan sistem rekomendasi.



Wendi Maksalmina. Bogor, 23 Juli 1991. Lulus D3 Teknik Komputer di Institut Pertanian Bogor (IPB University) pada tahun 2012. Selanjutnya di tahun 2019 mendapat gelar S1 jurusan Sistem Informasi di BINUS Online. Saat ini dia bekerja sebagai Engineer IT Enterprise yang bertanggung jawab untuk mengembangkan dan mengelola aplikasi inhouse enterprise perusahaan.