

Implementasi Teknologi *Mern Stack* dalam Merancang Aplikasi Penjualan *Voucher Game Online* dengan Metode *Extreme Programming (XP)*

Alfad Indrawan Z^{a,1,*}, Poetri Lestari lokapitasari B^{a,2}, Lutfi Budi Ilmawan^{a,3}

^a Program studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muslim Indonesia, JL. Urip Sumoharjo KM.05 Makassar dan 90231, Indonesia

¹ alfadindrawanz@gmail.com; ² poetrilestarilokapitasarib@umi.ac.id; ³ lutfibudiilmawan@umi.ac.id
*corresponding author

INFORMASI ARTIKEL	ABSTRAK
Diterima : 23 – 02 – 2023 Direvisi : Diterbitkan : <i>Kata Kunci:</i> <i>Mern Stack</i> <i>Aplikasi Penjualan Voucher Game</i> <i>Extreme Programming</i> <i>Website</i>	Perkembangan <i>Game</i> yang sebelumnya hanya bisa dimainkan secara <i>offline</i> , dengan adanya <i>Game</i> di internet, bisa dimainkan secara <i>online</i> tanpa kendala waktu. Banyak dari mereka menghabiskan waktunya untuk bermain <i>Game</i> dan membeli suatu item yang bertujuan untuk membuat para pemainnya menerima item yang berguna dalam <i>Game</i> tersebut dengan istilah <i>cash in-Game</i> . Nato Store salah satu pelaku bisnis penjualan <i>voucher Game</i> yang dimana masalah utama yang terdapat yaitu pada saat membeli <i>voucher</i> begitu banyak dalam sehari sekiranya 20 orderan lebih sehingga terdapat antrian ketika membeli <i>voucher</i> , sehingga untuk mengatasi permasalahan itu, diperlukan sistem berupa aplikasi penjualan <i>Voucher Game</i> yang lebih efisien dalam memonitoring penjualan ketika ada orderan masuk. Adapun metode yang digunakan yaitu <i>Extreme Programming</i> yang ditujukan untuk dihadapkan dengan <i>requirement</i> yang tidak jelas maupun terjadi perubahan yang sangat cepat. Berdasarkan <i>kuesioner</i> dari 20 responden yang merupakan masyarakat yang hobi bermain <i>Game</i> menggunakan pengujian <i>black box</i> , maka diperoleh pengujian antarmuka 90%, pengujian kinerja 90%, pengujian database 89,6%, pengujian fungsi yang hilang atau rusak 30% dan pengujian inisialisasi/terminasi 90%. Adapun index keseluruhan pada pengujian <i>black box</i> sebesar 82,4%. Hasil yang didapatkan yaitu menghasilkan aplikasi penjualan <i>voucher Game</i> yang dapat digunakan oleh pengguna yang ingin mengisi <i>voucher Game</i> tanpa harus lagi menghubungi admin dari Nato Store selaku penyedia <i>Voucher</i> .

This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license



I. Pendahuluan

Perkembangan *Game* yang sebelumnya hanya bisa dimainkan secara *offline*, dengan adanya *Game* di internet, bisa dimainkan secara *online* tanpa kendala waktu [1]. Banyak dari mereka menghabiskan waktunya untuk bermain *Game* dan membeli suatu item yang bertujuan untuk membuat para pemainnya menerima barang yang berguna pada *Game* tersebut dengan istilah *cash in-Game*. Seiring perkembangan *Game* online sekarang ini banyak agen penjualan yang mangambil kesempatan untuk menjual *Voucher Game*. Mereka biasanya menjual *Voucher* ini melalui forum, group FB, akun sosmed jual beli dan media sosial lainnya [2].

Nato Store yang merupakan salahsatu pelaku bisnis penjualan *Voucher Game* secara digital yang dimana penjualannya masih mengandalkan promosi di sosial media dan forum jual-beli. untuk mengatasi permasalahan itu, diperlukan sistem berupa aplikasi penjualan *Voucher Game* yang lebih efisien dalam memonitoring penjualan ketika ada orderan masuk, juga dapat meningkatkan mutu pelayanan suatu informasi. MERN merupakan salah satu kombinasi teknologi yang digunakan untuk membangun aplikasi web yang terdiri dari MongoDB, Express, React, dan Node.js. Kelebihan utama dari MERN adalah digunakannya satu bahasa, yaitu JavaScript baik pada sisi client maupun server bahkan untuk mengelola basis data [3]. JavaScript sendiri menggunakan konsep asynchronous dan single-threaded yang dimana kode akan dieksekusi tanpa menunggu eksekusi kode lain selesai sehingga seakan-akan dieksekusi secara bersamaan. Hal ini akan membuat kinerja proses lebih ringan dan cepat dalam suatu aplikasi [4]. Diperlukan metode untuk merencanakan pengembangan sistem yang sesuai dengan kasus yang dihadapi. Metode pengembangan sistem merupakan kerangka yang menjadi dasar dalam perancangan dan pengembangan perangkat lunak dengan tujuan menghasilkan aplikasi yang sesuai dengan kebutuhan.

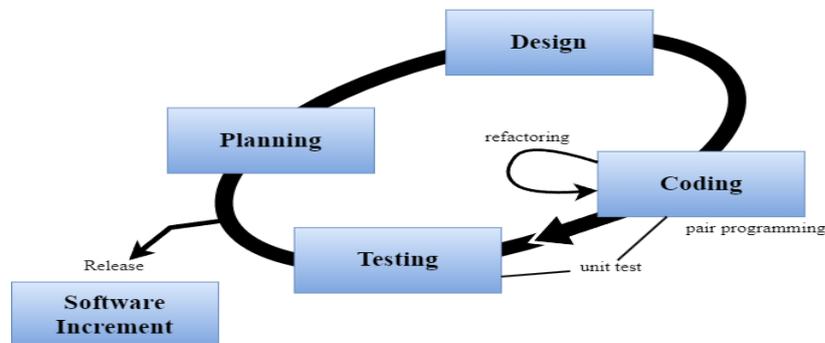
Salah satu metode pengembangan perangkat lunak yang merupakan cabang metode agile development yang digunakan untuk menyesuaikan kebutuhan pengembangan adalah extreme programming (XP). XP merupakan pengembangan rekayasa perangkat lunak yang dapat digunakan untuk pengembangan sistem dengan requirement yang tidak jelas maupun terjadi perubahan terhadap requirement yang sangat cepat [5]. Berdasarkan latar belakang yang telah diuraikan diatas, maka penulis mengimplementasikan teknologi *MERN Stack* dalam Merancang Aplikasi Penjualan *Voucher Game Online*.

II. Metode

A. Extreme Programming

Extreme Programming (XP) merupakan sebuah proses rekayasa perangkat lunak yang cenderung menggunakan pendekatan berorientasi objek dan serta metode ini juga sesuai jika dihadapkan dengan *requirement* yang tidak jelas maupun terjadi perubahan–perubahan *requirement* yang sangat cepat. asal mula *Extreme Programming* (XP) digunakan karena pada saat itu kebutuhan pelanggan sering berubah dengan cepat sehingga mengakibatkan siklus hidup metode pengembangan perangkat lunak. Perangkat lunak tradisional lebih pendek dan tidak selaras dengan metode tradisional. karena seringkali membutuhkan desain yang luas dan ini menyebabkan terjadinya perubahan desain dan tentu saja membutuhkan biaya yang lebih tinggi. Metode *Extreme Programming* (XP) melakukan penyederhanaan berbagai tahapan pengembangan sistem informasi menjadi lebih efisien, adaptif dan fleksibel [6]. Tujuan dari *Extreme Programming* (XP) adalah untuk mengurangi biaya yang terkait dengan proses pengembangan perangkat lunak[7].

Adapun tahapan Metode Extreme Programming sebagai berikut



Gambar 1. Tahapan *Extreme Programming*

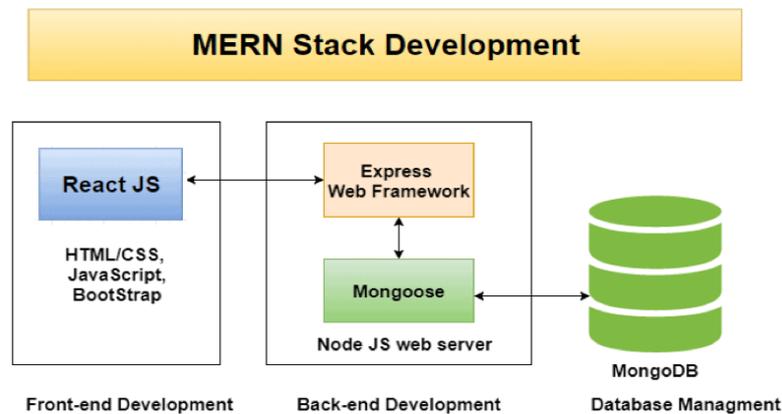
Terdapat empat tahapan yang harus dikerjakan pada metode extreme programmin (xp) yaitu [6]:

- Planning* (Perencanaan). Fase ini merupakan langkah awal dalam pengembangan sistem dimana berbagai kegiatan perencanaan yaitu mengidentifikasi masalah, dan menganalisa kebutuhan pengguna terhadap sistem yang akan dibangun.
- Design* (Perancangan). Fase selanjutnya adalah perancangan dimana kegiatan pemodelan berdasarkan hasil analisa kebutuhan yang didapatkan, XP mengikuti prinsip Keep It Simple (KIS). XP juga mendukung refactoring dimana software system diubah sedemikian rupa dengan cara mengubah stuktur kode dan menyederhanakannya namun hasil dari kode tidak berubah. Pada fase ini juga digunakan *Class Responsibilities Colaboration Card* (*CRC-Card*) untuk merancang sebuah sistem dan untuk mengetahui hubungan antar objek yang ada pada sistem. Pemodelan sistem yang digunakan yaitu Unified Modelling Language (UML) yang terdiri dari beberapa diagram antara lain Use-Case Diagram, dan Activity Diagram.
- Coding* (Pengkodean). Fase ini merupakan kegiatan pemodelan yang dilakukan dalam bentuk *user interface* dengan menggunakan bahasa pemrograman. Proses coding pada XP diawali dengan membangun serangkaian unit test.
- Testing* (Pengujian). Setelah fase coding selesai dilakukan tahap pengujian sistem untuk mengetahui kesalahan apa saja yang terjadi saat aplikasi dijalankan dan untuk mengetahui apakah sistem yang dibangun sudah memenuhi kebutuhan pengguna.
- Software Increment (Peningkatan Perangkat Lunak) Tahapan ini merupakan tahap pengembangan sistem yang sudah dibuat secara bertahap yang dilakukan setelah sistem diterapkan dengan menambahkan layanan atau konten yang mengakibatkan bertambahnya kemampuan fungsionalitas dari sistem.

B. MERN STACK

MERN adalah kamus yang digunakan untuk menggambarkan seperangkat teknologi berbasis JavaScript yang digunakan dalam proses pengembangan aplikasi web [8]. *MERN stack* merupakan salah satu teknologi yang menggunakan konsep SPA yang terdiri dari *MongoDB*, *Express.js*, *ReactJS*, dan *Node.js* [9]. Istilah *MERN Stack* disusun dari komponen-komponen sumber terbuka yang menyediakan kerangka kerja *end-to-end*

untuk pengembangan aplikasi berbasis web yang komprehensif dan memudahkan peramban untuk terhubung dengan database [10].



Gambar 2. Struktur *MERN Stack*

C. Node Js

Node.js adalah sebuah platform yang dirancang untuk *webservice*. Aplikasi ini ditulis dalam bahasa *javascript* dan berbasis pada event (*event driven*). Tidak seperti kebanyakan bahasa *javascript* yang dijalankan di *client*, *Node.js* dieksekusi sebagai aplikasi *server*. *Node.js* memiliki efisiensi memory yang lebih baik apabila bekerja dalam keadaan muatan yang banyak. Tetapi keunggulan *Node.js* ini terdapat pada *asynchronousnya* atau bisa juga disebut dengan *non-blocking I/O*. *Node.js* menggunakan pendekatan *event-driven* berbasis *infinite event loop* dalam satu *thread* untuk mengurangi jumlah *memory* [10].

D. MongoDB

MongoDB adalah database berbasis dokumen *open source*, yang juga dapat disebut sebagai database *noSQL*. *MongoDB* menggunakan format dokumen *JavaScript Object Notation (JSON)* dengan skema untuk manajemen data. Dengan bertambahnya jumlah dan variasi data, maka akan semakin mudah untuk dikelola menggunakan database *non-relasional*. *MongoDB* mengalami peningkatan yang signifikan dalam penggunaan di antara pengembang karena memenuhi kebutuhan ini dengan penyimpanan data berbasis dokumen. Setiap database dapat memiliki koleksi yang berbeda dan di setiap koleksi data disimpan sebagai daftar objek *JSON* [10].

E. Express Js

Express Js adalah kerangka kerja yang dibangun di atas fungsionalitas *server web Node.js* untuk menyederhanakan API dan menambahkan fitur baru yang berguna. Memfasilitasi pengelolaan fungsionalitas aplikasi dengan *middleware* dan *routing*. Ini menambah kenyamanan objek *HTTP Node.js*, membuatnya lebih mudah untuk membuat tampilan *HTML* dinamis, dan mendefinisikan standar skalabilitas yang mudah diterapkan [10].

F. React Js

React.js adalah library *JavaScript front-end* yang efisien dan fleksibel untuk pengembangan antarmuka pengguna. *React.js* dikelola oleh *Facebook* sebagai perpustakaan *JavaScript* sumber terbuka dapat digunakan untuk membuat tampilan yang berjalan pada *HTML*. *React.js* tidak seperti kerangka kerja komprehensif yang menangani semua aspek model seperti Model View Controller (MVC), tetapi ini hanya library *JavaScript*, jadi hanya menangani penulisan dan eksekusi tampilan web [10].

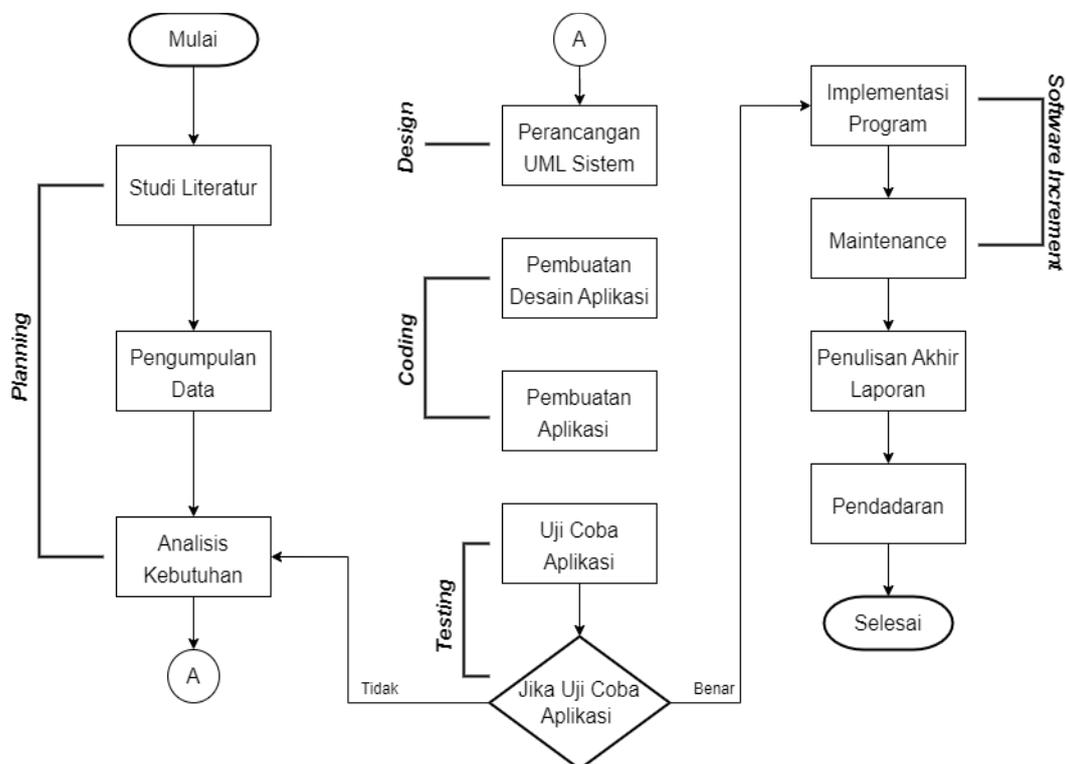
G. Black Box Testing

Pengujian dilakukan dengan menggunakan pengujian *black box* bertujuan melihat apakah sistem yang dibuat memenuhi tujuan awal dan layak untuk digunakan. Pengujian sistem menggunakan metode *black box*, tujuannya adalah untuk mengetahui bagian mana dari sistem aplikasi yang menampilkan pesan kesalahan dengan benar jika terjadi kesalahan pada saat input. Pengujian yang dilakukan hanya dengan mengamati hasil yang dijalankan melalui data pengujian dan memverifikasi fungsionalitas perangkat lunak [12].

H. Tahapan Penelitian

Tahapan penelitian ini berisikan tahapan penelitian yang dilakukan sebagai berikut:

1. *Planning* (Persiapan)
 - a. Studi Literatur, Penulis mengumpulkan acuan referensi dari berbagai sumber berupa artikel yang berkaitan dengan aplikasi yang ingin dibuat.
 - b. Pengumpulan data, Penulis melakukan pengumpulan data yang dilakukan melalui observasi, dan wawancara untuk mengumpulkan data *Voucher* yang dibutuhkan dari pemilik usaha *Nato Store*.
 - c. Analisis Kebutuhan, Penulis melakukan analisis untuk menentukan kebutuhan pengguna yang dibutuhkan dalam perancangan aplikasi.
2. *Design* (Perancangan)
 - a. Perancangan UML Sistem, Merancang UML sistem aplikasi penjualan *Voucher* berupa Use Case dan Activity Diagram sistem yang dibuat.
 - b. Perancangan Arsitektur Sistem, Pada penelitian ini, sistem yang akan dibuat menggunakan arsitektur dari Teknologi *MERN STACK*.
 - c. Perancangan *CRC-Card*, Perancangan kartu *Class Responsibility Collaborator (CRC-Card)* yang memiliki fungsi untuk membangun kelas-kelas beserta fungsionalitas yang akan digunakan.
3. *Coding* (Pengkodean)
 - a. Pembuatan Desain Aplikasi, Pembuatan desain aplikasi *Voucher* yang terdiri dari *login* dan *register* untuk pengguna, menu utama, nominal pengisian, metode pembayaran, dan halaman *admin*.
 - b. Pembuatan Aplikasi, Membuat aplikasi dimana fitur-fitur didalamnya sesuai dengan desain yang dirancang sebelumnya menggunakan dengan *tools Visual Studio Code* serta menggunakan *MERN STACK* sebagai kombinasi teknologi yang digunakan untuk membangun aplikasi *web* yang terdiri dari *MongoDB, Express, React, dan Node.js*.
4. *Testing* (Pengujian)
 - a. Uji Coba Aplikasi, Pengujian dilakukan dengan *Alpha Testing* dengan metode *Black Box* yang dimana dilakukan langsung oleh pengembang pada aplikasi menjelang produk hampir dalam keadaan dapat digunakan. untuk mengetahui apakah fungsi, masukan, dan keluaran telah sesuai dengan fungsinya masing-masing sebelum digunakan oleh pengguna.
5. *Software Increment* (Peningkatan Perangkat Lunak)
 - a. Implementasi Program, Pada aplikasi yang telah dibangun akan diimplementasikan dan mendeploy ke publik sehingga dapat diakses oleh pengguna.
 - b. *Maintenance*, Melakukan pemeliharaan jika aplikasi yang dibuat mengalami masalah
6. Penulisan akhir laporan, Penulis menyusun hasil laporan yang telah rampung ke dalam bentuk deskripsi hasil perancangan dan uji coba program.
7. Pendadaran, Setelah semua tahapan selesai maka ke tahapan akhir dimana melakukan presentasi hasil dari penelitian yang telah dilakukan.



Gambar 3. Tahapan Penelitian

III. Hasil dan Pembahasan

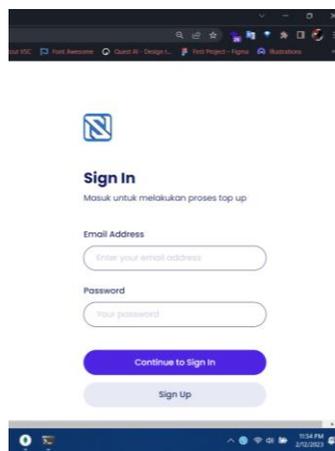
A. Hasil

Berdasarkan penelitian yang dilakukan menghasilkan keluaran berupa aplikasi penjualan *Voucher Game* berbasis *web* yang dapat digunakan oleh pengguna yang ingin mengisi *Voucher Game* tanpa harus lagi menghubungi admin dari Nato Store selaku penyedia *Voucher* dengan menggunakan metode pengembangan sistem *Extreme Programming*.

Berdasarkan kuesioner dari 20 responden yang merupakan masyarakat yang hobi bermain *Game* menggunakan pengujian *black box*, maka diperoleh pengujian antarmuka 90%, pengujian kinerja 90%, pengujian database 89,6%, pengujian fungsi yang hilang atau rusak 30% dan pengujian inialisasi/terminasi 90%. Adapun index keseluruhan pada pengujian *black box* sebesar 82,4%, hasil kuesioner terlampir.

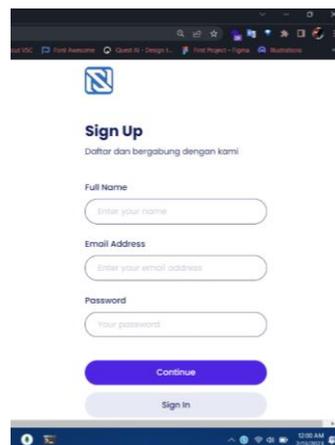
Adapun hasil penelitian yang diperoleh setelah sistem diimplementasikan dapat dilihat dari beberapa interface sebagai berikut:

1. Tampilan Pengguna



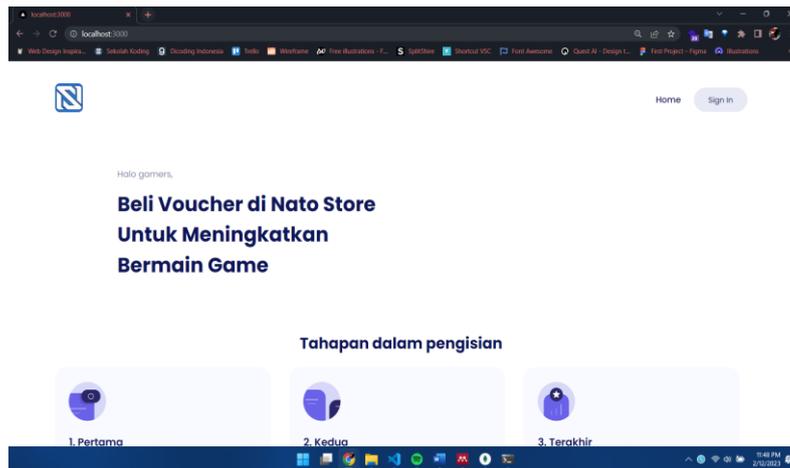
Gambar 4. Tampilan Halaman Login

Pada gambar 4 merupakan tampilan halaman login pada pengguna untuk dapat melakukan proses pembelian.



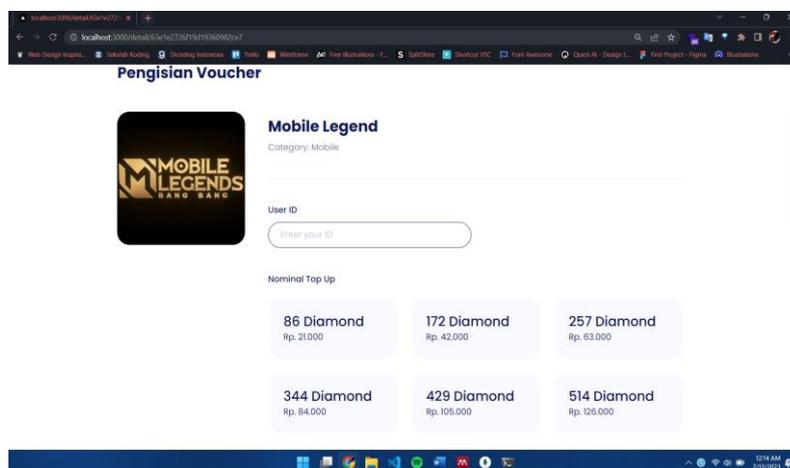
Gambar 5. Tampilan Halaman Registrasi

Pada gambar 5 merupakan tampilan halaman registrasi pada pengguna untuk dapat melakukan login.

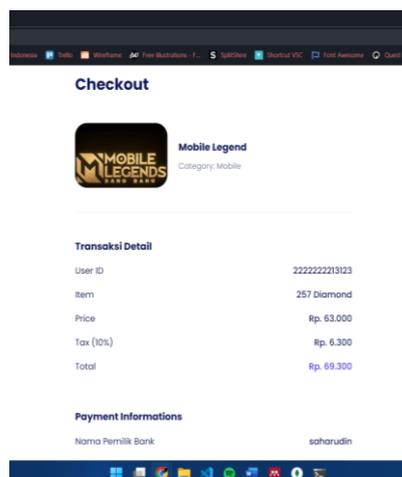


Gambar 6. Tampilan Halaman Utama

Pada gambar 6 merupakan tampilan halaman utama untuk melihat pilihan *Game* yang tersedia.

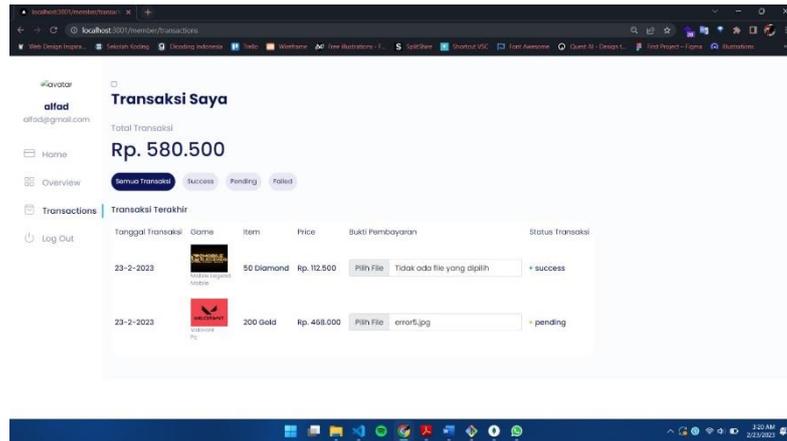
Gambar 7. Tampilan Pengisian *Voucher*

Pada gambar 7 merupakan tampilan pengisian *Voucher* sesuai yang diinginkan.



Gambar 8. Tampilan Halaman Checkout

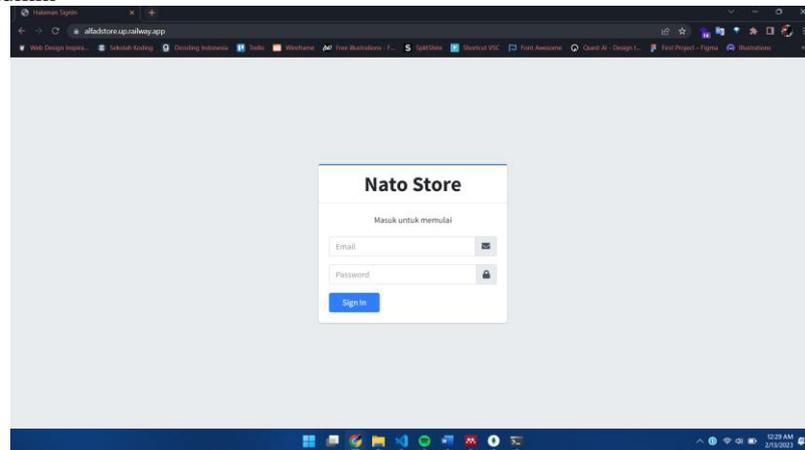
Pada gambar 8 merupakan tampilan checkout ketika selesai mengisi *Voucher* sesuai yang diinginkan.



Gambar 9. Tampilan Halaman *Dashboard* Pengguna

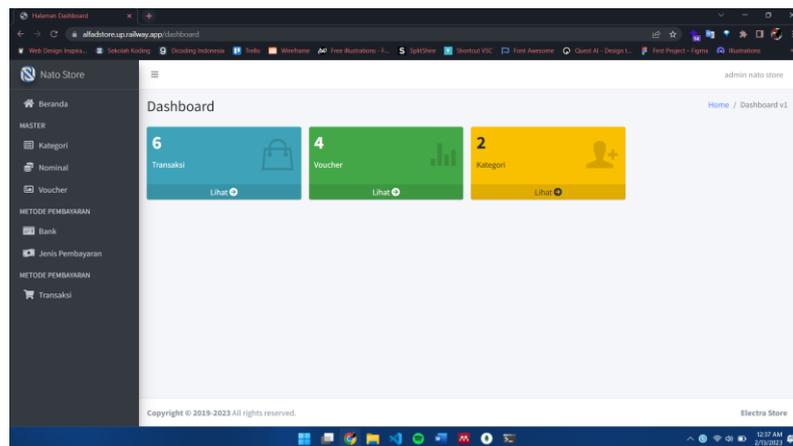
Pada gambar 9 merupakan tampilan dashboard pengguna yang menampilkan riwayat pembelian.

2. Tampilan Admin



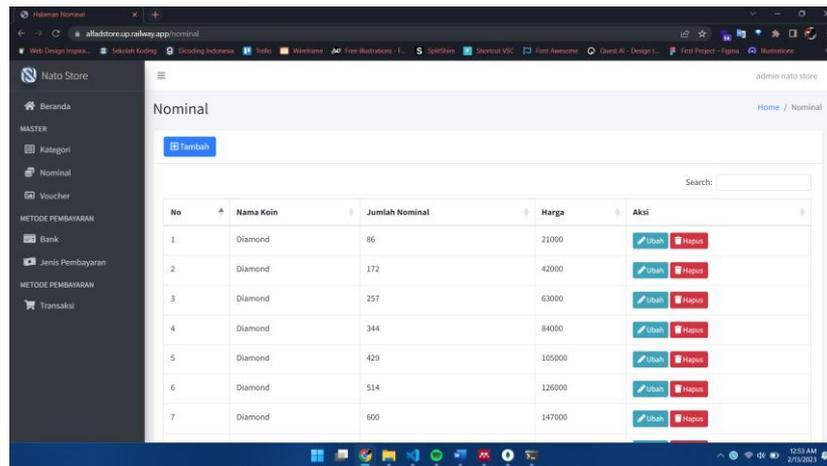
Gambar 10. Tampilan Halaman Login Admin

Pada gambar 10 merupakan tampilan halaman registrasi admin untuk dapat melakukan login.



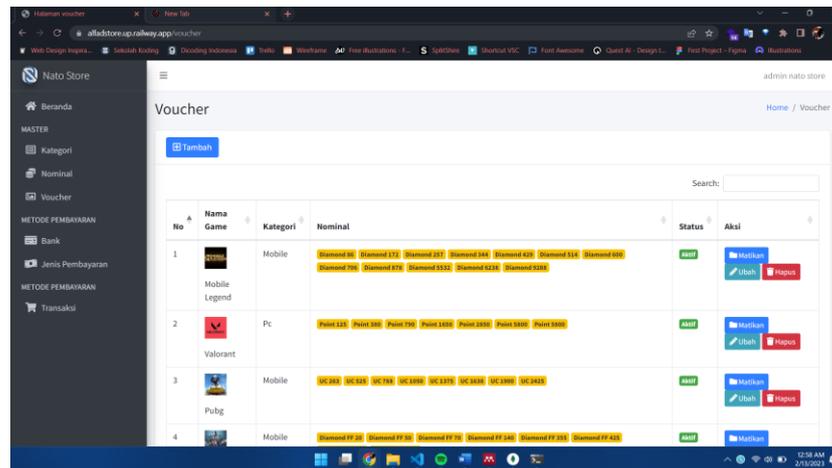
Gambar 11. Tampilan Halaman *Dashboard* Admin

Pada gambar 11 merupakan tampilan halaman *dashboard* admin untuk melihat jumlah transaksi, *Voucher*, dan kategori.



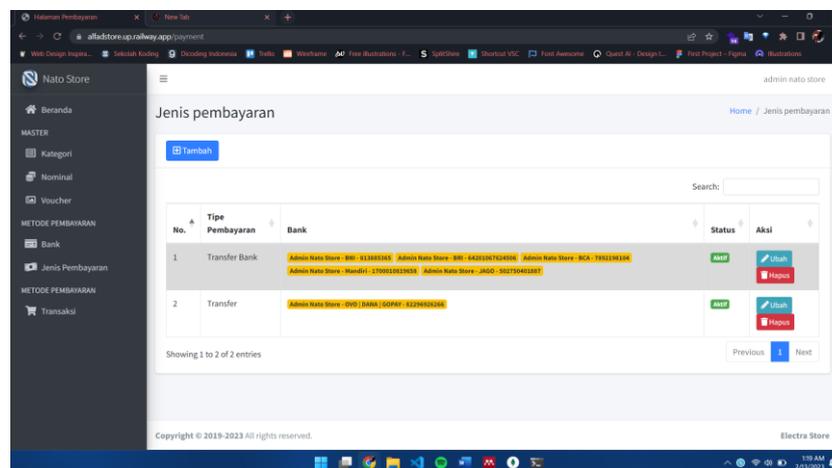
Gambar 12. Tampilan Halaman Nominal

Pada gambar 12 merupakan tampilan halaman nominal, Digunakan admin untuk menambah, mengubah, dan menentukan harga sebuah *Voucher Game*.



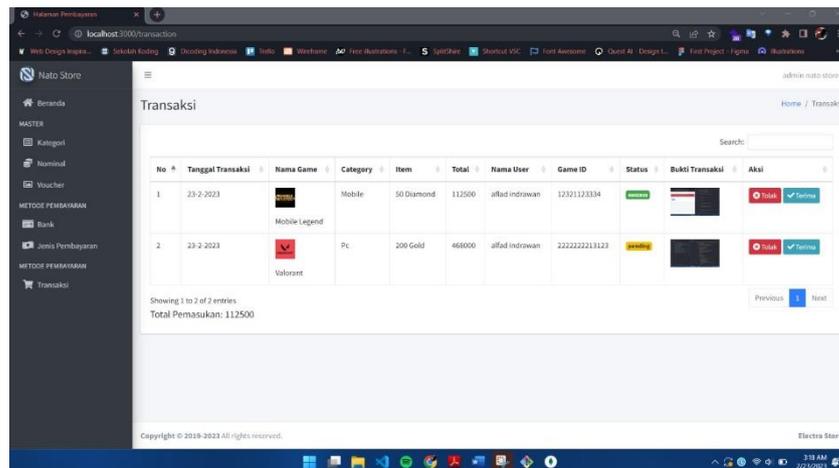
Gambar 13. Tampilan Halaman *Voucher Game*

Pada gambar 13 merupakan tampilan halaman *Voucher Game*, agar admin dapat menambah, menghapus dan mengubah isi *Voucher Game* berdasarkan nama *Gamenya* dengan lebih spesifik.



Gambar 14. Tampilan Halaman Jenis Pembayaran

Pada gambar 14 merupakan tampilan halaman jenis pembayaran, agar admin dapat menambah, menghapus dan mengubah jenis pembayaran yang dapat dilakukan saat melakukan transaksi



Gambar 15. Tampilan Halaman Transaksi

Pada gambar 15 merupakan tampilan halaman transaksi, agar admin dapat melihat dan menerima transaksi dari pengguna.

IV. Kesimpulan dan saran

Adapun hasil perancangan dan berdasarkan hasil persentase dari pengujian yang telah dilakukan berdasarkan hasil kuesioner sebanyak 20 responden maka dapat dinyatakan bahwa Perancangan Aplikasi Penjualan *Voucher Game* dengan Penerapan Metode *Extreme Programming* pada masyarakat yang hobi bermain *Game* yang dapat digunakan oleh pengguna yang ingin mengisi *Voucher Game* tanpa harus lagi menghubungi admin dari *Nato Store* selaku penyedia *Voucher*. Oleh karena itu maka dapat ditarik kesimpulan bahwa berdasarkan penelitian yang dilakukan lebih kurang tiga bulan menghasilkan aplikasi penjualan *voucher Game* dengan penerapan metode *Extreme Programming* dan telah di implementasikan pada masyarakat yang hobi bermain *Game* dengan tingkat keberhasilan user sebesar 82,4%. Pada aplikasi penjualan *voucher Game* dilakukan uji coba (testing) berdasarkan aspek kriteria pengujian *black box*, diperoleh pengujian antarmuka 90%, pengujian kinerja 90%, pengujian database 89,6%, pengujian fungsi yang hilang atau rusak 30% dan pengujian inisialisasi/terminasi 90%. Pada aplikasi penjualan *voucher Game* dapat menampilkan fitur pengisian *voucher*, *checkout*, kategori, nominal, jenis pembayaran, bank, unggah bukti pembayaran, serta transaksi.

Berdasarkan kesimpulan diatas, maka penulis memberikan saran-saran yang dapat dijadikan masukan, yaitu dalam perancangan aplikasi bisa dikembangkan dengan metode *SDLC* contohnya yaitu metode *Scrum*. Diharapkan juga pada penelitian berikutnya dapat dikembangkan aplikasi dengan fitur *payment gateway* dengan berbasis *Android* dan *IOS*.

Daftar Pustaka

- [1] C. Fajri, "Tantangan Industri Kreatif-Game Online di Indonesia," *J. ASPIKOM*, vol. 1, no. 5, p. 443, 2012, doi: 10.24329/aspikom.v1i5.47.
- [2] Vit, "Risiko Berbahaya Jika Kamu Beli Diamond Mobile Legend Ilegal, Rugi Bandar!," <https://ggwp.id/>, 2020. <https://ggwp.id/media/esports/mobile-legends/risiko-diamond-mobile-legend-ilegal> (accessed Apr. 10, 2022).
- [3] R. Triandy and N. Santoso, "Pengembangan Aplikasi Web Reservasi Paket Wisata menggunakan MERN Stack (Studi Kasus: Zona Tamasya Tour Organizer)," vol. 4, no. 6, pp. 1616–1624, 2020, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [4] R. Romadhon, "Kerja di Startup: Node Js. atau PHP Developer?," <https://www.softwareseni.co.id/>, 2021. <https://www.softwareseni.co.id/blog/kerja-di-startup-node-js-atau-php-developer> (accessed Apr. 10, 2022).
- [5] R. I. Borman, A. T. Priandika, and A. R. Edison, "Implementasi Metode Pengembangan Sistem

- Extreme Programming (XP) pada Aplikasi Investasi Peternakan,” *J. Sist. dan Teknol. Inf.*, vol. 8, no. 3, p. 272, 2020, doi: 10.26418/justin.v8i3.40273.
- [6] I. Carolina and A. Supriyatna, “Penerapan Metode Extreme Programming dalam Perancangan Aplikasi Perhitungan Kuota SKS Mengajar Dosen,” *J. IKRA-ITH Inform.*, vol. 3, no. 1, pp. 106–113, 2019.
- [7] D. Andika, “Apa itu Extreme Programming??,” *www.it-jurnal.com*, 2018. <https://www.it-jurnal.com/apa-itu-extreme-programming/> (accessed May 27, 2022).
- [8] M. Mehra *et al.*, “Assistant Professor , Computer Science and Engineering , Arya Institute of Engineering & B . Tech Student , Computer Science and Engineering , Arya College of Engineering &,” vol. 25, no. 6, pp. 11756–11761, 2021.
- [9] S. Afif and I. K. D. Nuryana, “Rancang Bangun Sistem Informasi Staycation Berbasis Web Dengan Implementasi Teknologi Mern Stack,” *Angew. Chemie Int. Ed. 6(11)*, 951–952., pp. 1–12, 2021.
- [10] M. P. Widodo, “Pengembangan Aplikasi Pelaporan Progress-Plan- Pengembangan Aplikasi Pelaporan Progress-Plan-,” 2020.
- [11] R. Handoyo, L. W. Santoso, and A. Setiawan, “Real-Time BPMN Website Menggunakan Teknologi MERN Stack,” *J. Infra*, vol. 7, no. 2, pp. 75–80, 2019, [Online]. Available: <http://publication.petra.ac.id/index.php/teknik-informatika/article/view/8758>
- [12] A. Krismadi, A. F. Lestari, A. Pitriyah, I. W. P. A. Mardangga, M. Astuti, and A. Saifudin, “Pengujian Black Box berbasis Equivalence Partitions pada Aplikasi Seleksi Promosi Kenaikan Jabatan,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 2, no. 4, p. 155, 2019, doi: 10.32493/jtsi.v2i4.3771.