

RANCANG BANGUN *BACK-END E-COMMERCE* U-MARKET BERBASIS *WEBSITE* MENGGUNAKAN *FRAMEWORK* LARAVEL DENGAN METODE *SCRUM*

Annisa Tri Hapsari^{1*}, Alya Ananda Putri², Yessi Mulyani³, Nurrahma⁴, Wahyu Eko Sulistiono⁵

^{1,2,3,4,5} Program Studi Teknik Informatika, Jurusan Teknik Elektro, Universitas Lampung; Jl. Prof. Sumantri Bojonegoro No.1, Bandar Lampung.

Keywords:

E-commerce;
Back-end;
Laravel;
Scrum;
U-Market.

Correspondent Email:

annisatrih1212@gmail.com

Abstrak. Penelitian ini berfokus pada pengembangan sisi *back-end* yang mencakup pengelolaan data pengguna, produk, dan transaksi melalui REST API. Metode *Scrum* diterapkan untuk mendukung pengembangan sistem secara iteratif melalui 4 tahap *sprint*, sehingga setiap fitur dapat dikembangkan, diuji, dan dievaluasi secara bertahap. Pengujian sistem dilakukan menggunakan metode *black box testing* untuk memastikan seluruh fungsi berjalan sesuai kebutuhan, serta *stress testing* menggunakan Apache JMeter untuk mengukur performa sistem dalam menangani beban pengguna. Hasil pengujian menunjukkan bahwa seluruh *endpoint* API yang dikembangkan berjalan dengan baik dan valid sesuai skenario pengujian. Selain itu, berdasarkan hasil *stress testing* dengan variasi beban hingga 300 sampel pengguna, sistem mampu memberikan respon yang stabil tanpa mengalami kegagalan signifikan, sehingga menunjukkan bahwa performa *back-end* cukup andal dalam menangani permintaan secara bersamaan. Implementasi fitur utama seperti autentikasi, manajemen produk, keranjang, transaksi, serta integrasi *payment gateway* Midtrans juga berjalan sesuai fungsinya. Dengan demikian, sistem U-Market mampu menjadi solusi yang lebih terstruktur, aman, dan efisien dibandingkan metode sebelumnya, serta penerapan metode *Scrum* terbukti meningkatkan efektivitas proses pengembangan.



Copyright © 2026 The Author(s), Published by Jurnal Informatika dan Teknik Elektro Terapan. This is an open-access article distributed under the terms of the Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Abstract. This research focuses on back-end development, including the management of user, product, and transaction data through a structured REST API. The Scrum method is applied to support iterative and incremental system development through 4 stages of sprints, allowing each feature to be developed, tested, and evaluated progressively. System testing is conducted using black box testing to ensure all functionalities operate as intended, and stress testing using Apache JMeter to evaluate system performance under concurrent user loads. The testing results indicate that all developed API endpoints function correctly and meet the defined testing scenarios. Furthermore, stress testing with varying loads of up to 300 samples demonstrates that the system maintains stable performance without significant failures, indicating that the back-end is capable of handling concurrent requests effectively. The implementation of core features, including user authentication, product management, shopping cart, transaction processing, and integration with the Midtrans payment gateway, operates as expected. Therefore, the U-Market system provides a more structured, secure, and efficient solution compared to the previous method, while the application of the Scrum method contributes to improving the effectiveness of the development process and the quality of the resulting system.

1. PENDAHULUAN

Perkembangan teknologi informasi telah mendorong pertumbuhan *e-commerce* sebagai sistem perdagangan daring yang mampu meningkatkan efisiensi transaksi dan memperluas jangkauan pasar [1]. Di lingkungan Universitas Lampung, aktivitas jual beli sebelumnya banyak dilakukan melalui grup WhatsApp yang tidak terstruktur, sehingga informasi penjualan mudah tertimbun, tidak terdapat pencatatan data yang jelas, serta minim keamanan transaksi [2]. Padahal, Universitas Lampung memiliki potensi pasar yang besar dengan jumlah mahasiswa mencapai 44.839 orang yang mayoritas aktif menggunakan internet dan perangkat digital.

Berdasarkan permasalahan tersebut, dikembangkan *website e-commerce* U-Market yang dapat digunakan oleh masyarakat umum, mahasiswa, maupun pelaku usaha lokal. Sistem dibangun menggunakan *framework* Laravel karena memiliki arsitektur MVC yang terstruktur dan mendukung pengembangan REST API secara efisien [3]. Proses pengembangan dilakukan menggunakan metode *Scrum* melalui *Sprint* yang terstruktur sehingga setiap fitur dapat dikembangkan, diuji, dan dievaluasi secara bertahap [4]. Kebaruan dari penelitian ini terletak pada penerapan metode *Scrum* dalam pengembangan *back-end e-commerce* U-Market berbasis Laravel yang dirancang tidak hanya untuk lingkungan kampus, tetapi juga dapat dimanfaatkan secara luas oleh masyarakat umum, termasuk pelaku usaha lokal dan pengguna umum. Penerapan *Scrum* pada pengembangan *back end e-commerce* komunitas lokal masih relatif jarang dilakukan, khususnya dengan fokus pada pengelolaan sistem secara iteratif dan inkremental melalui pembagian pekerjaan ke dalam beberapa *Sprint*.

Penelitian ini menitikberatkan pada perancangan sisi *back end* untuk mengelola data produk, transaksi, dan pengguna secara aman, efisien, serta mudah dikembangkan, di mana setiap fitur dikembangkan secara bertahap, diuji, dan dievaluasi pada akhir *Sprint*. Dengan pendekatan tersebut, sistem dapat beradaptasi terhadap perubahan kebutuhan pengguna secara lebih fleksibel dan berkelanjutan. Oleh karena itu, penelitian ini diharapkan mampu memberikan kontribusi nyata dalam mendukung transformasi digital sektor

perdagangan lokal sekaligus memperkaya literatur terkait penerapan metode *Scrum* dan *framework* Laravel dalam pengembangan sistem *e-commerce* modern.

2. TINJAUAN PUSTAKA

2.1 E-Commerce

E-commerce adalah rangkaian aktivitas bisnis yang memanfaatkan jaringan elektronik untuk melakukan transaksi jual-beli barang dan/atau jasa beserta dukungan proses pendukungnya, seperti pencatatan produk, katalog digital, manajemen pesanan, pembayaran elektronik, logistik, dan layanan purna-jual [5].

2.2 Back-end Development

Back-end development merupakan proses pengembangan sisi *server* yang bertanggung jawab terhadap pengolahan data, logika bisnis, autentikasi pengguna, dan komunikasi dengan basis data. Dalam sistem *e-commerce*, *back-end* berperan penting dalam menjaga keamanan, stabilitas, dan performa sistem [7].

2.3 Website

Website atau situs merupakan salah satu komponen paling fundamental dalam era digital saat ini, berfungsi sebagai media penyampaian informasi, sarana komunikasi, dan platform layanan interaktif bagi masyarakat [8].

2.4 Unified Modelling Language

Unified modelling language (UML) adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi dan mengkomunikasikan rancangan mereka dengan yang lain [10].

2.5 MySQL

MySQL merupakan salah satu DBMS yang paling umum digunakan dalam pengembangan aplikasi web modern karena bersifat *open-source*, mendukung bahasa SQL (*Structured Query Language*), dan memiliki

performa tinggi pada transaksi berorientasi data besar [14].

2.6 Framework Laravel

Laravel merupakan *framework* PHP berbasis MVC yang menyediakan struktur pengembangan aplikasi web secara modular dan terorganisasi. Laravel mendukung pengembangan REST API melalui fitur *routing*, *middleware*, Eloquent ORM, serta sistem autentikasi yang aman [15].

2.7 Scrum

Scrum merupakan metode pengembangan perangkat lunak berbasis *Agile* yang dilakukan secara iteratif dan inkremental melalui pembagian pekerjaan ke dalam *Sprint*. *Scrum* terdiri dari beberapa tahapan utama seperti *Sprint Planning*, *Daily Scrum*, *Sprint Review*, dan *Sprint Retrospective* [18].

2.8 Black Box Testing

Black-box testing merupakan salah satu metode pengujian perangkat lunak yang berfokus pada fungsi sistem tanpa memperhatikan struktur internal atau logika program yang digunakan untuk menghasilkan keluaran tersebut. Pengujian jenis ini dilakukan dengan tujuan memastikan bahwa perangkat lunak dapat berfungsi sesuai dengan kebutuhan pengguna (*user requirements*) yang telah ditentukan pada tahap analisis [19].

2.9 Stress Test

Stress Testing adalah cara untuk melihat bagaimana situasi rawan kegagalan mempengaruhi tingkat stres sistem. Pengujian akan dilakukan dengan variabel jumlah pengguna dalam satu waktu dan akan menguji respon sistem ketika diakses oleh pengguna dalam jumlah besar dan dalam jangka waktu tertentu [21].

2.10 Payment Gateway

Payment gateway merupakan layanan perantara dalam sistem transaksi elektronik yang berfungsi menghubungkan pelanggan, merchant, dan lembaga perbankan dalam proses pembayaran daring secara aman. Dalam praktik *e-commerce*, *payment gateway* berperan sebagai pihak ketiga tepercaya (*trusted third party*) yang menerima, mengenkripsi, serta

meneruskan data pembayaran pelanggan untuk proses otorisasi tanpa harus mengekspos informasi sensitif kepada merchant secara langsung [22].

3. METODE PENELITIAN

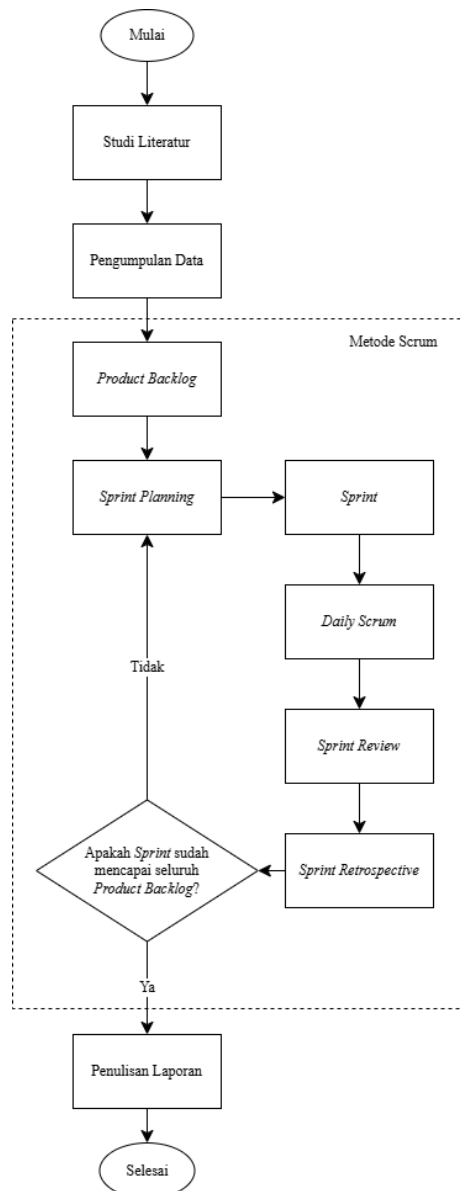
3.1 Alat dan Bahan

Adapun alat dan bahan yang digunakan pada penelitian ini adalah sebagai berikut:

1. Laptop Dell Latitude E7450
2. *Software* Visual Studio Code dengan ekstensi Laravel versi 12
3. Web server Apache
4. DBMS MySQL versi 8.0
5. *Testing tools* Postman dan Apache JMeter

3.2 Diagram Alir Penelitian

Penelitian ini dilakukan melalui beberapa tahapan yang disusun secara sistematis agar tujuan penelitian dapat tercapai dengan baik. Proses penelitian dimulai dari tahap studi literatur dan pengumpulan data, kemudian dilanjutkan dengan penerapan metode *Scrum* yang terdiri dari *Product Backlog*, *Sprint Planning*, *Sprint*, *Daily Scrum*, *Sprint Review*, dan *Sprint Retrospective*. Tahapan *Scrum* dilakukan secara iteratif hingga seluruh *Product Backlog* berhasil diselesaikan. Setelah seluruh proses pengembangan selesai, penelitian dilanjutkan pada tahap penulisan laporan. Adapun diagram alir penelitian dapat dilihat pada Gambar 1.



Gambar 1. Diagram Alir Penelitian

3.3 Analisis Kebutuhan

Analisis kebutuhan dilakukan untuk mengidentifikasi kebutuhan sistem yang diperlukan dalam pengembangan *back-end e-commerce* U-Market. Kebutuhan tersebut dibagi menjadi kebutuhan fungsional dan kebutuhan non fungsional agar sistem yang dikembangkan dapat berjalan sesuai tujuan penelitian.

3.3.1 Kebutuhan Fungsional

Berikut merupakan kebutuhan fungsional dari sistem U-Market.

Tabel 1. Kode Fungsional

Kode	Deskripsi
KF-01	Sistem harus mampu melakukan proses <i>login</i> dan <i>logout</i> pengguna dengan mekanisme autentikasi yang aman, serta membedakan hak akses antara admin dan pengguna berdasarkan peran (<i>role</i>) yang tersimpan di <i>database</i> .
KF-02	Sistem harus menyediakan layanan <i>back-end</i> yang memungkinkan pengguna (penjual) untuk menambahkan, memperbarui, menampilkan, dan menghapus data produk yang dijual, dengan data tersimpan secara konsisten pada <i>database</i> .
KF-03	Sistem harus menyediakan fitur pencarian produk melalui <i>endpoint back-end</i> berdasarkan nama produk atau kategori, sehingga memudahkan pengguna dalam menemukan produk yang diinginkan.
KF-04	Sistem harus mampu mencatat pesanan pengguna, memproses dan memperbarui status pembayaran, serta menyimpan riwayat transaksi pengguna secara terstruktur pada <i>database</i> .
KF-05	Sistem harus menyediakan fitur <i>back-end</i> yang memungkinkan admin melihat, memperbarui, serta menonaktifkan akun pengguna apabila diperlukan untuk keperluan pengelolaan sistem.
KF-06	Sistem harus menyediakan layanan <i>back-end</i> yang memungkinkan pengguna untuk melihat dan memperbarui data profil dengan menerapkan validasi <i>input</i> serta menyimpan perubahan data secara aman ke dalam <i>database</i> .
KF-07	Sistem <i>back-end</i> harus menyediakan <i>endpoint</i> REST API yang dapat diakses oleh <i>front-end</i> untuk menampilkan data produk, transaksi, dan data pengguna secara terstruktur dan aman.
KF-08	Sistem harus melakukan validasi input pada setiap proses (<i>login</i> , pengelolaan produk, transaksi) untuk memastikan data yang tersimpan tetap konsisten, valid, dan terhindar dari kesalahan <i>input</i> .
KF-09	Sistem harus menyediakan fitur <i>back-end</i> yang memungkinkan admin melihat laporan penjualan dan data transaksi sebagai bahan monitoring dan evaluasi aktivitas <i>e-commerce</i> .

3.3.2 Kebutuhan Non-Fungsional

Berikut merupakan kebutuhan non fungsional dari sistem U-Market.

Tabel 2. Kebutuhan Non Fungsional

Kode	Deskripsi
KNF-01	Sistem harus menerapkan mekanisme keamanan berupa enkripsi kata sandi pengguna dan token API, untuk melindungi data dari akses tidak sah.
KNF-02	Sistem harus mampu memberikan respon API yang cepat dan stabil, meskipun menangani banyak permintaan secara bersamaan, guna menjamin kenyamanan pengguna.
KNF-03	Sistem harus dapat beroperasi secara stabil dan andal tanpa sering mengalami gangguan, khususnya pada saat proses transaksi dan pengolahan data berlangsung.
KNF-04	Sistem harus dirancang dengan arsitektur yang mudah dikembangkan dan dipelihara, sehingga penambahan fitur di masa depan dapat dilakukan tanpa perubahan besar pada struktur utama sistem.
KNF-05	Kode program harus mengikuti pola arsitektur MVC pada <i>framework</i> Laravel agar struktur kode lebih terorganisasi dan memudahkan proses pengembangan serta pemeliharaan sistem.
KNF-06	Sistem harus dapat berjalan pada berbagai perangkat dan terintegrasi dengan antarmuka pengguna berbasis web melalui layanan REST API.

3.4 Product Backlog

Pada tahap *Product Backlog*, dilakukan penyusunan daftar seluruh fitur, fungsionalitas, dan kebutuhan produk yang akan dikembangkan. *Product backlog* dalam sistem U-Market ditampilkan dalam Tabel 3 sampai Tabel 5.

Tabel 3. *Product Backlog* Pengguna

No.	Item	Prioritas	Points
1	Inisialisasi proyek <i>backend</i> Laravel, konfigurasi struktur folder dan <i>routing</i> REST API	Sedang	2
2	Konfigurasi koneksi <i>database</i> MySQL dan <i>environment backend</i>	Rendah	0.5

3	Pemodelan <i>database website</i> U-Market	Rendah	1
4	Implementasi API autentikasi (<i>register, login, logout</i>)	Tinggi	5
5	<i>Hash Password</i>	Rendah	1
6	Implementasi <i>Role Based Access Control</i> (RBAC) pengguna	Sedang	2
7	Implementasi API manajemen profil pengguna (lihat & edit profil)	Sedang	3
8	Implementasi API daftar produk	Tinggi	3
9	Implementasi API detail/deskripsi produk	Tinggi	3
10	Implementasi API pencarian produk (nama/kategori)	Sedang	2
11	Implementasi API kelola keranjang pengguna	Tinggi	3
12	Implementasi API <i>checkout</i> dengan integrasi <i>Payment gateway</i> Midtrans	Tinggi	5
13	Implementasi API <i>callback</i> Midtrans untuk validasi dan <i>update</i> status transaksi	Tinggi	5
14	Implementasi API akses halaman penjual	Sedang	3
15	Implementasi API riwayat transaksi pengguna	Sedang	3
16	Implementasi Integrasi Kontak Penjual (WhatsApp)	Sedang	2
17	Implementasi API verifikasi email pengguna	Tinggi	5
18	Implementasi API manajemen produk penjual	Tinggi	5
19	Implementasi API manajemen rekening bank	Tinggi	5
20	Implementasi API manajemen saldo dan penarikan dana	Tinggi	5

Tabel 4. *Product Backlog Admin*

No.	Item	Prioritas	Points
1	Implementasi API login dan autentikasi admin	Tinggi	3
2	Implementasi API beranda Admin	Tinggi	5
3	Implementasi API manajemen pengguna	Tinggi	5
4	Implementasi API manajemen produk	Tinggi	5
5	Implementasi API melihat seluruh transaksi	Tinggi	3
6	Implementasi pembatasan akses endpoint admin	Tinggi	3

Tabel 5. *Product Backlog Pengujian*

No.	Item	Prioritas	Points
1	Dokumentasi REST API	Rendah	0.5
2	Deployment backend ke server produksi	Sedang	2
3	Pengujian REST API Black box Testing	Sedang	3
4	Pengujian performa berupa stress test	Tinggi	5

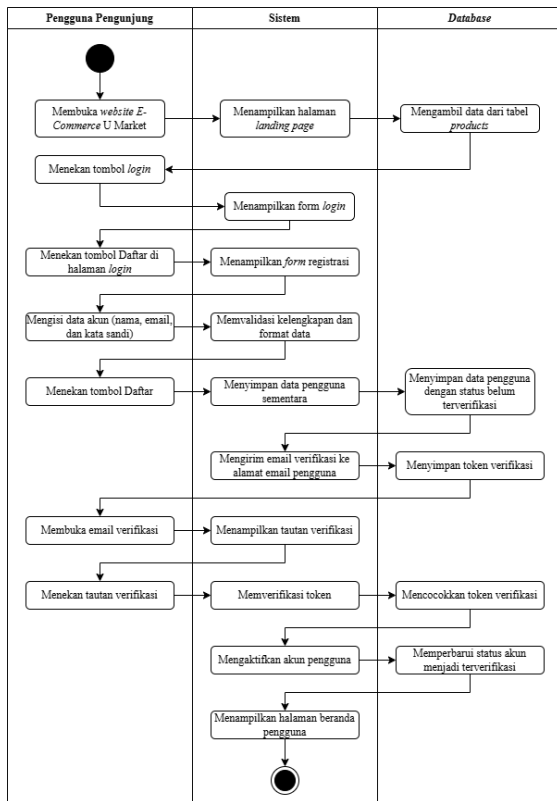
3.4.1 Use Case Diagram

Perancangan *use case diagram* membantu dalam proses pengembangan sistem karena memberikan gambaran awal mengenai modul-modul yang harus diimplementasikan. *Use case diagram* U-Market ditampilkan pada gambar 2.

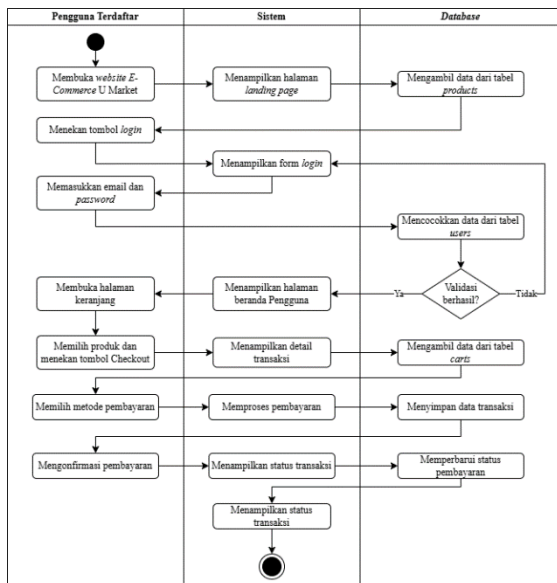
Gambar 2. *Use Case Diagram*

3.4.2 Activity Diagram

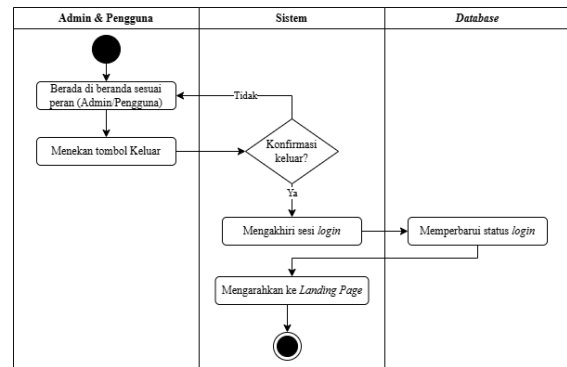
Activity diagram digunakan untuk memodelkan alur aktivitas yang terjadi dalam sistem, menggambarkan bagaimana proses bisnis berjalan dari awal hingga akhir. *Activity diagram* sistem U-Market ditampilkan pada Gambar 3 sampai Gambar 5.



Gambar 3. Activity Diagram Registrasi



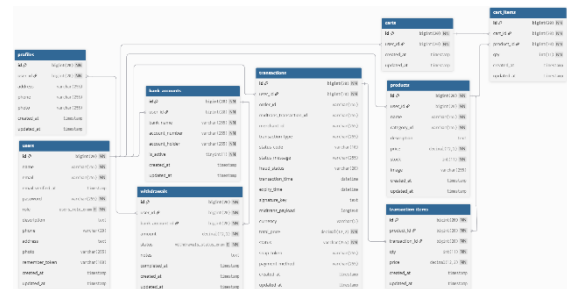
Gambar 4. Activity Diagram Melakukan Transaksi



Gambar 5. Activity Diagram Logout

3.4.3 Class Diagram

Class Diagram menggambarkan struktur sistem secara statis, termasuk kelas, atribut, operasi (*method*), serta hubungan antar kelas dalam sistem. *Class diagram* sistem U-Market ditampilkan pada gambar 6.



Gambar 6. Class Diagram

3.5 Sprint Planning

Sprint Planning dilaksanakan pada awal setiap *Sprint* sebagai tahap perencanaan pengembangan sistem. Pada tahap ini, tim pengembang bersama pihak terkait melakukan pembahasan untuk menentukan ruang lingkup pekerjaan yang akan dikerjakan selama *Sprint* berlangsung.

3.6 Daily Scrum

Daily Scrum merupakan kegiatan koordinasi yang dilaksanakan secara rutin oleh tim pengembang selama *Sprint* berlangsung. Kegiatan ini difasilitasi oleh *Scrum Master* dan diikuti oleh seluruh anggota tim pengembang dengan durasi singkat, umumnya sekitar 15 menit.

3.7 Sprint Review

Sprint Review merupakan tahapan evaluasi yang dilaksanakan pada akhir setiap *Sprint* untuk menilai hasil pengembangan sistem yang

telah dicapai selama periode *Sprint* berlangsung.

3.8 *Sprint Retrospective*

Sprint Retrospective merupakan tahapan refleksi yang dilakukan oleh tim pengembang setelah *Sprint* berakhir.

4. HASIL DAN PEMBAHASAN

Pada penelitian ini dilakukan pengembangan sistem *back-end e-commerce* U-Market menggunakan framework Laravel dengan arsitektur berbasis REST API. Pengembangan sistem difokuskan pada penyediaan layanan utama yang mendukung proses jual beli secara digital, meliputi autentikasi pengguna, manajemen profil, manajemen produk, keranjang belanja, transaksi, riwayat transaksi, integrasi *payment gateway* Midtrans, serta fitur tarik saldo penjual.

4.1 Hasil Implementasi Sistem

Tahap implementasi dimulai dengan pembangunan sistem autentikasi menggunakan token API yang mencakup fitur *register*, *login*, *logout*, dan verifikasi email pengguna. Selain itu, diterapkan *middleware* dan mekanisme *Role Based Access Control* (RBAC) untuk mengatur hak akses antara admin dan pengguna sehingga keamanan serta pengelolaan akses sistem dapat berjalan dengan lebih baik.

Selanjutnya, pada proses transaksi dilakukan integrasi dengan *payment gateway* Midtrans melalui *endpoint checkout* dan *callback notification*. Integrasi ini memungkinkan sistem melakukan validasi pembayaran serta pembaruan status transaksi secara otomatis dan *real-time*. Berdasarkan hasil implementasi dan pengujian yang dilakukan, seluruh layanan utama pada sistem *back-end* U-Market berhasil berjalan sesuai dengan kebutuhan yang telah ditentukan.

4.2 *Black Box Testing*

Tabel berikut akan menampilkan hasil pengujian *black box* yang dilakukan pada sistem U-Market.

Tabel 6. *Black Box Testing*

Deskripsi Pengujian	Data Pengujian	Status
Pengujian melakukan <i>register</i>	POST <i>/api/register</i> Request body diisi benar	Valid
Pengujian melakukan <i>login</i>	POST <i>/api/login</i> Email & password benar	Valid
Pengujian melakukan <i>logout</i>	POST <i>/api/logout</i> Token valid	Valid
Pengujian melakukan <i>checkout</i>	POST <i>/api/checkout</i> Request body diisi benar	Valid
Pengujian melihat profil	GET <i>/api/profile</i> Menggunakan <i>Bearer Token</i>	Valid
Pengujian menambahkan produk ke keranjang	POST <i>/api/cart/add</i> <i>product_id</i> & <i>quantity</i> valid	Valid
Pengujian melihat data penjualan admin	GET <i>/api/admin/sales-data</i> Login sebagai admin	Valid
Pengujian melihat daftar pengguna	GET <i>/api/admin/users</i> Login sebagai admin	Valid
Pengujian melihat seluruh transaksi admin	GET <i>/api/admin/transactions</i> Login sebagai admin	Valid
Pengujian callback Midtrans dengan status settlement	POST <i>/api/midtrans/notification</i> <i>transaction_status</i> = "settlement" <i>status_code</i> = "200" <i>signature_key</i> valid	Valid
Pengujian menambahkan rekening bank	POST <i>/api/user-banks</i> Data diisi lengkap dan benar	Valid
Pengujian melihat saldo	GET <i>/api/seller-balance</i> User login	Valid

4.3 *Stress Test*

Stress Test dilakukan untuk mengukur performa dan ketahanan sistem REST API U-Market ketika menerima sejumlah permintaan dalam waktu yang bersamaan. Pengujian ini bertujuan untuk mengetahui kemampuan sistem dalam menangani beban tinggi serta memastikan stabilitas layanan *back-end*. *Stress*

Test dilakukan menggunakan *tools* Apache JMeter. Berikut merupakan hasil dari pengujian *stress test*.

Tabel 7. Hasil Pengujian *Stress Test*

User	Samp le	Average	Error %	Throug hput
15	60	2222	0.00%	4.3/sec
30	120	4956	0.00%	3.6/sec
60	240	9038	0.00%	3.8/sec
150	600	16503	14.83%	5.3/sec

Berdasarkan hasil pengujian *Stress Test* yang telah dilakukan, terlihat bahwa peningkatan jumlah pengguna virtual berpengaruh terhadap waktu respons sistem. Dengan demikian, sistem dapat dikatakan berjalan optimal hingga 60 pengguna virtual, sedangkan pada 150 pengguna virtual mulai terjadi penurunan performa yang perlu diantisipasi melalui optimasi kode, *caching*, atau peningkatan spesifikasi *server* pada lingkungan produksi.

5. KESIMPULAN

Berdasarkan hasil penelitian, sistem *back-end e-commerce* U-Market berhasil dikembangkan menggunakan *framework* Laravel dan dinyatakan layak digunakan dengan kinerja stabil hingga 60 pengguna virtual (error 0%) serta masih dapat berjalan pada 150 pengguna virtual (error 14,83%). Penerapan Laravel dengan arsitektur MVC, Eloquent ORM, dan RBAC terbukti efektif dalam membangun sistem yang terstruktur dan aman. Selain itu, metode *Scrum* berhasil diterapkan dalam empat *sprint* sehingga pengembangan sistem berjalan secara iteratif dan terukur. Secara keseluruhan, sistem ini mampu meningkatkan efisiensi proses jual beli dibandingkan metode sebelumnya yang masih manual melalui grup WhatsApp karena seluruh transaksi dan data tersimpan otomatis dalam basis data.

Sebagai saran untuk pengembangan selanjutnya, sistem dapat dikembangkan menjadi aplikasi *mobile* berbasis Android/iOS yang terintegrasi dengan REST API untuk meningkatkan kenyamanan dan jangkauan pengguna. Selain itu, fitur pelacakan pesanan secara *real-time* dapat ditambahkan untuk meningkatkan transparansi proses pengiriman. Pengembangan juga disarankan pada sistem

pencarian produk dengan filter yang lebih lengkap seperti harga, rating, dan lokasi agar pengguna lebih mudah menemukan produk yang diinginkan.

DAFTAR PUSTAKA

- [1] M. R. Fahlevi dan S. Mu'minin, "Web-Based Online Learning Platform with RAD and laravel Methods," *Ultimatics : Jurnal Teknik Informatika*, vol. 17, no. 1, hlm. 107, Jul 2025.
- [2] D. Permana dan K. Santosa, "Perancangan Sistem Informasi Administrasi Transaksi E-Commerce Menggunakan Metode RAD dan MVC Framework Laravel Pada KOMA Collection," *OKTAL : Jurnal Ilmu Komputer dan Science*, vol. 3, no. 5, hlm. 1303–1313, Mei 2024.
- [3] A. Sunardi dan Suharjito, "MVC architecture: A comparative study between laravel framework and slim framework in freelancer project monitoring system web based," *Procedia Computer Science*, Elsevier B.V., 2019, hlm. 134–141. doi: 10.1016/j.procs.2019.08.150.
- [4] Y. Candra Kurniawan dan T. Raharjo, "Scrum Effectiveness Measurement in Human Resources Technology Product at Telecommunication Company," *Journal of Informatics and Communications Technology*, vol. 5, no. 1, hlm. 149–158, 2023, doi: 10.52661.
- [5] S. A. Bhat, K. Kansana, dan J. M. Khan, "A Review Paper On E-Commerce," *Asian Journal of Technology & Management Research*, vol. 6, no. 1, hlm. 2249–0892, Feb 2021.
- [6] B. Chessa Sarmanela, A. N. Samiya, dan M. E. Yuliana, "Mengenal Backend Developer: Skill, dan Tanggung Jawab Menjadi Backend Developer," *Prosiding Seminar Nasional Teknologi Informasi dan Bisnis (SENATIB)*, 2023.
- [7] B. N. Ramadhan dan S. Sugiyanto, "Analisis Usability Website Sistem Informasi Sd Negeri 1 Wangon Menggunakan Metode System Usability Scale," *Jurnal Informatika Teknologi dan Sains*, vol. 6, no. 3, hlm. 421–431, Agu 2024.
- [8] S. W. Ramdany, S. Aulia Kaidar, B. Aguchino, C. Amelia, A. Putri, dan R. Anggie, "Penerapan UML Class diagram dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web," *Journal of Industrial and Engineering System*, vol. 5, no. 1, hlm. 30–41, Jun 2024.

- [9] A. C. Dea, G. Ginting, dan N. P. I. Muhammad, "Analisis Kinerja *Database* Dalam Sistem *E-Commerce*," *Masip: Jurnal Manajemen Administrasi Bisnis dan Publik Terapan*, vol. 1, no. 3, hlm. 42–45, Jul 2023, doi: 10.59061/masip.v1i3.303.
- [10] F. Luthfi, "Penggunaan *Framework* Laravel Dalam Rancang Bangun Modul *Back-End* Artikel *Website* Bisnisbisnis.ID," *JISKa*, 2017.
- [11] E. Pangestu Wonohardjo, R. Febriyanto Sunaryo, Y. Sudiyono, dan N. Surantha, "A *Systematic Review Of SCRUM In Software Development*," *International Journal On Informatics Visualization*, vol. 3, 2019.
- [12] Ian. Sommerville, *Software engineering*. Pearson, 2016.
- [13] Luh N, Ginasari A, Wibawa K, Kadek N, Wirdiani A, "Pengujian *Stress Testing* API Sistem Pelayanan dengan Apache JMeter," *Jurnal Ilmiah Teknologi dan Komputer (JITTER)*, Vol. 2, no. 3, 2021.
- [14] K. Zay Oo, "Design and Implementation of *Electronic Payment gateway for Secure Online Payment System the Creative Commons Attribution License (CC BY 4.0)*," *International Journal of Trend in Scientific Research and Development (IJTSRD)* *International Journal of Trend in Scientific Research and Development*, vol. 3, no. 5, hlm. 1329–1334, 2019, doi: 10.31142/ijtsrd26635.
- [15] M. F. Allard dan A. Voutama, "Rancang Bangun Sistem Informasi Reservasi Hotel "Hotel Hebat" Berbasis *Website*," *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, vol.12, no.2, eISSN: 2830-7062.