

PENERAPAN METODE SCRUM PADA RANCANG BANGUN SISTEM INFORMASI PENJUALAN TOKO SEPATU RABBANI SHOES

Rahayu Noveandini, Maria Sri Wulandari dan Abdul Hakim

STMIK Jakarta STI&K

Jl. BRI No. 17, Radio Dalam, Kebayoran Baru, Jakarta Selatan 12140
ayu.noveandini@gmail.com, mswuland@yahoo.com, hkiem09@gmail.com

ABSTRAK

Metodologi agile development merupakan salah satu metodologi dalam membangun sebuah sistem informasi. Metode agile development adalah pendekatan pengembangan perangkat lunak yang berfokus pada fleksibilitas, kolaborasi, dan adaptasi yang cepat. Metodologi ini menekankan pada pengiriman iteratif dan inkremental dari perangkat lunak, dengan melibatkan keterlibatan aktif antara tim pengembang dan pemangku kepentingan. Agile development mempunyai metode antara lain: Scrum, Kanban, Extreme Programming (XP) dan Lean Software Development. Metode Scrum adalah framework atau metode dalam Agile development yang paling populer. Metode ini menggunakan pendekatan iteratif dan inkremental untuk mengelola proyek pengembangan perangkat lunak. Dari hasil observasi pada Rabbani Shoes terdapat beberapa permasalahan yaitu proses pencatatan penjualan, proses transaksi dan laporan penjualan, saat ini masih dikerjakan secara manual. Keadaan ini menyebabkan proses penjualan menjadi tidak optimal dan efisien. Alternatif yang ditawarkan adalah membangun sistem informasi penjualan pada toko Rabbani Shoes. Hasil penelitian ini berupa sistem informasi penjualan yang membantu kegiatan penjualan yang dilakukan oleh toko Rabbani Shoes.

Kata Kunci: *agile development, scrum, penjualan, Rabbani Shoes*

PENDAHULUAN

Dalam membangun sistem informasi diperlukan sebuah metode pengembangan sistem. Metodologi pengembangan perangkat lunak yang biasa dipakai salah satunya adalah *agile development*. Metode *agile development* adalah pendekatan atau filosofi dalam pengembangan perangkat lunak yang menekankan pada kolaborasi tim adaptabilitas terhadap perubahan, dan pengiriman nilai secara cepat dan terus menerus. Metode *agile* berfokus pada kerja tim yang terorganisir, transparan, dan responsif terhadap kebutuhan pengguna. Berbeda dengan metode pengembangan perangkat lunak tradisional yang lebih kaku dan berorientasi pada rencana yang sudah ditentukan sejak awal, metode Agile memungkinkan tim untuk mengatasi ketidakpastian dan perubahan yang mungkin terjadi dalam proyek. [1]

Metode *agile* mencakup sejumlah prinsip dan praktik. Manifesto *agile* menyatakan nilai – nilai dan prinsip – prinsip inti *agile*, antara lain: (1) Individu dan

interaksi lebih mengutamakan perhatian pada kolaborasi dan komunikasi yang efektif antara individu dalam tim pengembangan, dibanding mengandalkan proses dan alat, (2) Perangkat lunak lebih mengutamakan pengembangan perangkat lunak dibanding dokumentasi yang terperinci, (3) Kerjasama dengan pelanggan lebih diutamakan karena mendorong interaksi yang terus menerus dan kolaborasi yang erat dengan pelanggan, mengakui bahwa perubahan kebutuhan dapat terjadi seiring waktu dibanding dengan negosiasi kontrak. Metode *agile* mampu mengadvokasi praktik – praktik seperti pengembangan melalui iterasi dan *increment*, kolaborasi intensif, pemecahan pekerjaan menjadi tugas yang lebih kecil dan terukur, pengujian berkelanjutan, dan refleksi serta peningkatan terus-menerus melalui proses inspeksi dan adaptasi. Beberapa *framework* yang menerapkan metode *agile* secara praktis, antara lain Scrum, Kanban, Extreme Programming (XP), Lean Software Development, Crystal, dan Feature-Driven Development (FDD). Dengan menggunakan metode *agile*, tim

pengembangan dapat menghasilkan perangkat lunak dengan cepat, adaptif, dan berorientasi pada nilai bisnis yang memberikan kepuasan kepada pengguna dan pemangku kepentingan. [1].

Metode *scrum* merupakan *framework* dalam *agile development* yang populer serta digunakan untuk mengelola proyek pengembangan perangkat lunak. Metodologi ini bertujuan untuk meningkatkan fleksibilitas, transparansi, dan kolaborasi dalam tim pengembangan. Metode ini dikembangkan oleh Jeff Sutherland tahun 1993.[2]. Dalam *scrum*, pengembangan perangkat lunak atau proyek dikelola dalam iterasi pendek yang disebut *sprint*. Setiap *sprint* biasanya berlangsung selama dua hingga empat minggu, di mana tim bekerja untuk menciptakan produk yang dapat disampaikan secara potensial. Selama *sprint*, tim memiliki tujuan jelas yang ditetapkan dalam bentuk daftar tugas atau cerita pengguna.[2].

Metode ini fokus pada kerja tim kolaboratif dan pengaturan fleksibel terhadap perubahan kebutuhan. Berdasarkan penelitian – penelitian sebelumnya, metode *scrum* merupakan metode dengan pendekatan iteratif dan incremental dimana pekerjaan dibagi dalam *sprint* atau iterasi pendek yang menghasilkan produk yang digunakan dengan waktu yang singkat [3]. Pengembangan sistem menggunakan *scrum* memerlukan waktu yang singkat [4]. Penggunaan metode *scrum* menghasilkan output yang sesuai dengan kebutuhan [5]. Kelebihan metode *agile* adalah: (1) Memungkinkan tim pengembangan untuk merespons perubahan kebutuhan dengan cepat. Pendekatan ini mengakui bahwa kebutuhan pengguna dapat berubah seiring waktu, dan metode *agile* memungkinkan pengembangan perangkat lunak yang fleksibel dan mudah diubah. (2) Menggunakan iterasi singkat, yang dikenal sebagai *sprint*, untuk menghasilkan potongan perangkat lunak yang berfungsi. Ini memungkinkan pengiriman perangkat lunak secara bertahap dan terukur, sehingga pelanggan dan pemangku kepentingan dapat melihat dan memberikan umpan balik lebih awal dalam proses pengembangan. (3) Mendorong kolaborasi yang erat antara

anggota tim pengembangan, pelanggan, dan pemangku kepentingan lainnya. Komunikasi yang terbuka dan terus-menerus memastikan pemahaman yang baik tentang kebutuhan dan memungkinkan pemecahan masalah yang lebih efektif. (4) Membantu dalam mengidentifikasi masalah secara dini dan memastikan kualitas perangkat lunak yang lebih tinggi. (5) Dengan pengujian berkelanjutan, metode *agile* membantu dalam mengurangi risiko dalam pengembangan perangkat lunak. Masalah dan kesalahan dapat diidentifikasi dan diperbaiki lebih awal, sehingga mengurangi dampak yang lebih besar pada proyek. [6].

Hasil observasi dari Rabbani Shoes terdapat beberapa permasalahan yaitu, sering terjadinya kesalahan dalam pencatatan pemesanan barang, proses transaksi dapat mengalami keterlambatan karena bergantung pada interaksi pelanggan dan penjual, jika penjual tidak cepat menganggapi pertanyaan pelanggan, maka niat pelanggan untuk belanja mengalami penundaan bahkan batal. Proses penjualan produk hanya bisa dilakukan dengan kontak informasi berupa nomor telpon atau kontak sosial media yang dicantumkan oleh Rabbani Shoes.

METODE PENELITIAN

Metode *scrum* berjalan baik untuk proyek yang sering berubah dan memiliki waktu yang singkat. Metode *scrum* adalah kerangka kerja pengembangan perangkat lunak yang berfokus pada kerjasama tim, transparansi, dan adaptasi dalam proyek yang kompleks. Metode ini digunakan dalam pengembangan perangkat lunak, tetapi dapat diterapkan dalam proyek – proyek lain yang melibatkan tim yang bekerja secara kolaboratif. [7].

Berikut ini cara kerja metode *Scrum* [7]:

1. **Product Backlog:** Tahapan proses diawali dengan membuat *product backlog*, yaitu daftar prioritas yang berisi semua fitur, kebutuhan, dan perubahan yang diinginkan untuk produk yang sedang dikembangkan serta bertanggung jawab untuk mengelola *product backlog* dan menentukan prioritasnya. Pada proses ini tim membuat spesifikasi sistem dari hasil observasi yang dilakukan.

2. **Sprint Planning:** Melakukan pertemuan *sprint planning* untuk memilih dan merencanakan pekerjaan yang akan dilakukan selama *sprint* berikutnya. Pada proses ini melakukan pembuatan fitur – fitur yang diperlukan dari hasil analisis kebutuhan sistem dan akan diimplementasikan selama *sprint*.
3. **Sprint:** *Sprint* adalah periode waktu terbatas di mana tim bekerja untuk menghasilkan potongan fungsional dari produk yang dapat disampaikan. Pada proses ini melakukan pengembangan perangkat lunak sesuai dengan *product backlog* yang telah disepakati. Tim berkolaborasi secara teratur dalam pertemuan harian singkat yang disebut *daily scrum*, untuk berbagi perkembangan, mengidentifikasi hambatan, dan menyinkronkan aktivitas. Tahap dalam *sprint* adalah:
 - a. **Scrum Board:** Untuk melacak kemajuan dan mengelola pekerjaan, tim menggunakan *scrum board*. *Scrum Board* biasanya terdiri dari tiga kolom: "To Do" (belum dilakukan), "In Progress" (sedang dilakukan), dan "Done" (selesai). Tugas-tugas diorganisir dalam bentuk kartu atau catatan, dan dipindahkan antar kolom saat progres dilakukan.
 - b. **Sprint Review:** Setelah *sprint* selesai, tim melakukan pertemuan *sprint review* dengan pengguna. Pada tahap ini dilakukan demonstrasi produk yang telah selesai dan menerima umpan balik dari pengguna. Hasilnya dapat mengarah pada perubahan pada *product backlog* atau prioritas berikutnya.
 - c. **Sprint Retrospective:** Setelah *sprint review*, tim melakukan pertemuan *Sprint Retrospective* untuk mengevaluasi kinerja mereka selama *sprint*. Pada tahap ini dilakukan identifikasi hal-hal yang berjalan baik, hambatan yang dihadapi, dan peluang perbaikan. Hasil dari pertemuan ini digunakan untuk meningkatkan proses kerja tim di *sprint* berikutnya.

Pada tabel 1 merupakan modul yang dibangun. Modul yang dibangun terdiri dari empat modul, yaitu: menu login digunakan sebagai komponen untuk mengakses sistem dengan memasukkan identitas informasi yang valid, manajemen produk adalah modul yang berisi tentang produk yang akan dipasarkan, manajemen user berguna untuk mengatur data pengguna, dan manajemen transaksi dipakai untuk mengatur kegiatan transaksi. Daftar komponen yang akan dibangun disebut dengan istilah *backlog* [8].

Tabel 1. Daftar Fitur

| No | Fitur |
|----|---------------------|
| 1 | Login user |
| 2 | Manajemen Produk |
| 3 | Manajemen User |
| 4 | Manajemen Transaksi |

Setelah menentukan komponen pada *product backlog* langkah berikutnya adalah pertemuan untuk memilih waktu pengerjaan komponen tersebut. Tabel 2 memperlihatkan *sprint fitur backlog* yang akan dikerjakan.

Tabel 2. Sprint Fitur Backlog

| No | Fitur Backlog | Tugas | Perkiraan Waktu |
|----|---------------------|----------------------------|-----------------|
| 1 | Login | Insert Delete Update | 1 minggu |
| 2 | Manajemen Produk | Insert Delete Update | 2 minggu |
| 3 | Manajemen User | Insert Delete Update | 1 minggu |
| 4 | Manajemen Transaksi | Insert Delete Update | 3 minggu |

4. **Pengulangan:** Pada proses ini seluruh fitur yang ada dalam *product backlog* telah selesai dilakukan. Jika hasil sudah sesuai maka sistem akan dipublikasikan, tetapi jika masih ada ketidaksesuaian maka tim kembali ke langkah *Sprint Planning* untuk memulai *sprint* berikutnya. Proses ini berlanjut dengan iterasi yang berulang-ulang sampai

produk akhir tercapai atau tujuan proyek tercapai.

Prinsip *scrum* melibatkan fleksibilitas, adaptasi, dan transparansi, yang memungkinkan tim untuk merespons perubahan kebutuhan dan hambatan selama pengembangan produk. Pengembangan sistem menggunakan metode *scrum* disajikan pada gambar 1.



Gambar 1. Framework Metode Scrum

HASIL DAN PEMBAHASAN

Pada penelitian ini dibangun sistem informasi penjualan sepatu pada toko sepatu Rabbani Shoes. Sistem ini dapat digunakan untuk memudahkan melakukan pencarian informasi mengenai layanan penjualan sepatu baik sepatu pria, wanita maupun anak – anak. Pada tahap ini menguraikan hasil dari sistem yang telah dibangun.

Sprint Planning

Merupakan tahap awal pada Scrum. Pada tahap ini dilakukan perencanaan kegiatan perancangan aplikasi sesuai dengan kebutuhan. Tabel 3 menyajikan data sprint planning.

| No | Nama Backlog | Waktu (Hari) | Keterangan |
|----|-----------------|--------------|--|
| 1 | Perancangan UML | 3 | Merancang UML sesuai dengan kebutuhan aplikasi |
| 2 | Login | 6 | Merancang halaman login |

| | | | |
|------------------|-------------------|---|---------------------------------------|
| 3 | Produk | | |
| | Mengelola Data | 3 | Merancang halaman produk |
| | Mengelola Produk | 3 | |
| | Mengelola Pesanan | 3 | |
| | Mengelola Laporan | 2 | |
| Pemesanan Produk | 3 | | |
| 4 | User | 7 | Merancang insert, delete, update user |
| 5 | Keranjang | 7 | Merancang halaman keranjang belanja |
| 6 | Checkout | 7 | Merancang halaman checkout |

Daily Scrum

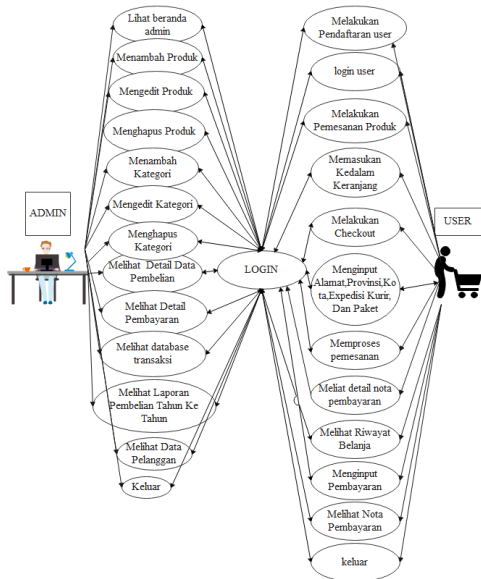
Pada tahap ini dilakukan evaluasi karena setiap anggota tim melakukan update data untuk melihat kemajuan kegiatan anggota tim.

Sprint Review

Pada tahap ini dilakukan uji coba setiap tampilan yang dibuat. Apakah fitur – fitur yang disertakan berfungsi dengan baik.

Perancangan Use Case Diagram

Pada *use case diagram* seperti tersaji pada gambar 2, aktor yang belum daftar dan *login* hanya dapat mengakses *website* dengan melihat kategori produk. Saat melakukan pemesanan dan ingin menyelesaikan pemesanan *user* diharuskan mendaftarkan dan login terlebih dahulu jika sebelumnya belum pernah melakukan pemesanan pada *website* tersebut, agar dapat melakukan kegiatan lainnya seperti pada gambar aktor sebagai *user*. Aktor sebagai admin hanya memiliki 1 akun untuk mengolah data semua *website*, aktor *admin* harus login dahulu supaya dapat mengakses semuanya.

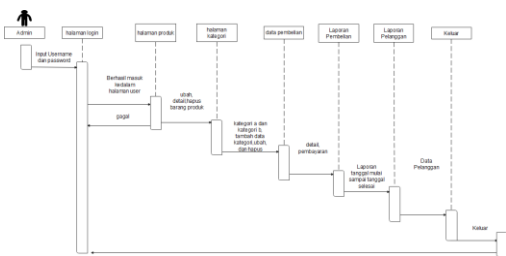


Gambar 2. Diagram Use Case Admin

Sequence Diagram

Sequence Diagram dipakai guna memperlihatkan bagaimana objek-objek berinteraksi dalam suatu proses atau alur kerja. Diagram ini membantu dalam memahami urutan peristiwa yang terjadi dalam sistem dan memberikan gambaran visual yang jelas tentang komunikasi antara objek – objek tersebut.

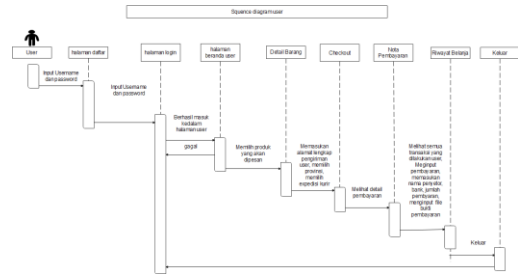
Sequence diagram pada Rancang Bangun Sistem Informasi Pemesanan Sepatu Toko Rabbani Shoes Berbasis Web ini terbagi dua yaitu diagram sequence admin dan diagram sequence user. Gambar 3 memperlihatkan sequence diagram admin dan gambar 4 memperlihatkan sequence diagram user.



Gambar 3. Diagram Sequence Admin

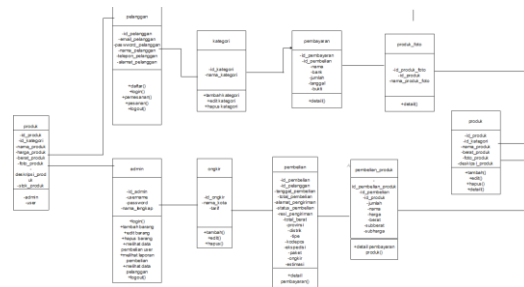
Sequence diagram user menggambarkan interaksi antara pengguna (user) dan sistem dalam suatu skenario atau alur kerja. Diagram ini fokus pada komunikasi dan aktivitas yang terjadi antara

pengguna dan sistem yang sedang berjalan dan tersaji pada gambar 4.



Gambar 4. Diagram Sequence User

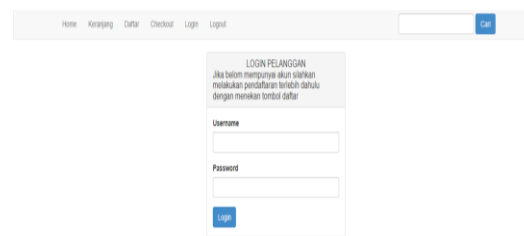
Class Diagram terdapat pada Gambar 5 yang menjelaskan relasi antar class yaitu, class user, class admin, produk, admin, pelanggan, kategori, ongkir, pembayaran, pembelian, produk_foto, pembelian_produk, produk_foto, produk. Masing-masing tabel berelasi sesuai kebutuhannya.



Gambar 5. Class Diagram

Halaman Login

Halaman login untuk mengotentikasi pengguna dan memberikan akses ke fitur – fitur yang terbatas hanya untuk pengguna yang memiliki kredensial yang valid. Tampilan login tersaji pada gambar 6.

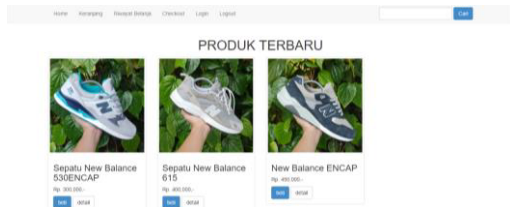


Gambar 6. Halaman Login User

Halaman Beranda

Halaman beranda adalah halaman awal atau tampilan pertama yang muncul

ketika pengguna mengakses suatu situs web, aplikasi, atau sistem. Halaman ini bertindak sebagai titik masuk utama yang memberikan ringkasan atau gambaran umum tentang konten, fitur, atau informasi yang tersedia di dalam situs atau aplikasi tersebut. Tampilan halaman beranda tersaji pada gambar 7.



Gambar 7. Halaman Beranda

Halaman Keranjang

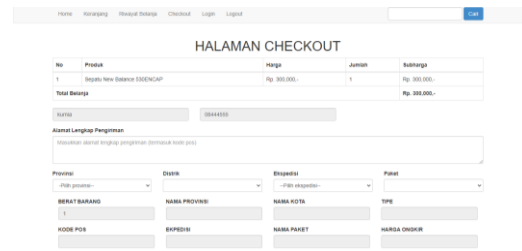
Halaman keranjang adalah halaman dalam situs web atau aplikasi *e-commerce* yang menampilkan daftar produk yang telah dipilih oleh pengguna untuk dibeli. Halaman ini memberikan ringkasan visual mengenai barang yang ada di keranjang belanja pengguna sebelum melakukan proses checkout atau pembayaran. Halaman keranjang tersaji pada gambar 8.



Gambar 8. Halaman Keranjang User

Halaman Checkout

Halaman *checkout* adalah halaman dalam situs web atau aplikasi *e-commerce* yang digunakan untuk melengkapi proses pembelian atau transaksi. Halaman ini menampilkan formulir dan informasi yang diperlukan untuk menyelesaikan pembayaran, pengiriman, dan konfirmasi pesanan. Gambar halaman *checkout* tersaji pada gambar 9.



Gambar 9. Halaman Checkout

Halaman Nota Pembelian

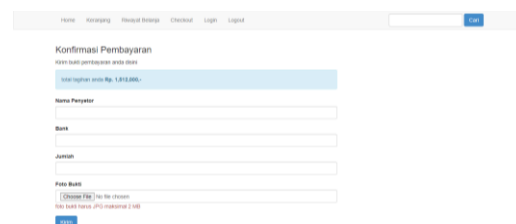
Halaman nota pembelian merupakan halaman untuk melihat detail pembelian barang. Halaman nota pembelian tersaji pada gambar 10.



Gambar 10. Halaman Nota Pembelian

Halaman Konfirmasi Pembayaran

Halaman konfirmasi pembayaran adalah halaman *user* melakukan konfirmasi pembayaran dengan memasukkan nama penyetor, nama bank, jumlah pembayaran, dan bukti pembayaran dengan memasukkan foto bukti transfer. Halaman konfirmasi pembayaran tersaji pada gambar 11.






Gambar 11. Halaman Input Pembayaran User

Uji Coba Aplikasi


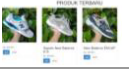
Uji coba aplikasi dilakukan berdasarkan product backlog. Untuk pengujian halaman beranda tersaji pada tabel 4.



Tabel 4. Uji Coba Halaman Beranda

| Kasus dan Hasil Uji (Data Benar) | | | |
|---|--|--------------------------------|--|
| Data yang dimasukkan | Yang Diharapkan | Uji Coba | Kesimpulan |
| Mengkonfirmasi akun user | Sistem akan mengaktifkan akun user | Menampilkan beranda user | [x] Diterima [] ditolak  |
| Mengklik tombol beli di form barang dan mengarah ke keranjang belanja | Sistem akan mengarahkan ke keranjang belanja | Menampilkan keranjang belanja | [x] Diterima [] ditolak  |
| Mengklik tombol detail pada form barang | Sistem akan melihat detail barang | Menampilkan data detail barang | [x] Diterima [] ditolak  |

Pengujian Halaman Keranjang
Halaman keranjang tersaji pada tabel 5.


Tabel 5. Halaman Keranjang

| Kasus dan Hasil Uji (Data Benar) | | | |
|----------------------------------|---|---------------------------------------|--|
| Data yang dimasukkan | Yang Diharapkan | Uji Coba | Kesimpulan |
| Form halaman Keranjang | Mengklik tombol beli dan diarahkan ke keranjang belanja | Menampilkan keranjang belanja | [x] Diterima [] ditolak  |
| Lanjutkan Belanja | Mengarah ke halaman beranda dan membeli barang lagi | Menampilkan barang di halaman beranda | [x] Diterima [] ditolak  |

| | | | |
|----------|---------------------------------|--|--|
| Checkout | Mengarahkan ke halaman checkout | Menampilkan halaman checkout | [x] Diterima [] ditolak  |
| Hapus | Membatalkan pembelian | Berhasil menghapus barang di keranjang, dan menampilkan pesan produk berhasil dihapus di keranjang | [x] Diterima [] ditolak  |

Pengujian Halaman Checkout
Hasil uji halaman checkout tersaji pada tabel 6.

Tabel 6. Uji Coba Halaman Checkout

| Kasus dan Hasil Uji (Data Benar) | | | |
|----------------------------------|--|------------------------------|--|
| Data yang dimasukkan | Yang Diharapkan | Uji Coba | Kesimpulan |
| Halaman Checkout | Mengarahkan ke halaman checkout setelah memasukkan barang ke keranjang | Menampilkan halaman checkout | [x] Diterima [] ditolak  |

PENUTUP

Kesimpulan

Implementasi *scrum* pada Rancang Bangun Sistem Informasi Pembelian pada Toko Rabbani Shoes berhasil dibangun berdasarkan *product backlog* yang ada dan semua fungsi pada sistem ini dapat berjalan baik untuk melakukan pembelian sepatu. Metode *scrum* memberikan pendekatan yang terstruktur dan terorganisir untuk

mengelola proyek dengan tim yang *cross-functional*.

Saran

Pengembangan aplikasi ini memerlukan inovasi yang lebih lugas seperti dapat dikembangkan pada *platform* lainnya seperti berbasis *mobile* dan pemanfaatan keamanan data untuk memastikan data – data transaksi tersebut aman dari gangguan peretas data.

DAFTAR PUSTAKA

- [1] Sallaby, A. F., & Kanedi, I. Perancangan Sistem Informasi Jadwal Dokter Menggunakan Framework Codeigniter. *Jurnal Media Infotama*, Volume 16 Nomor 1, Februari 2020.
- [2] Schwaber, K., & Sutherland, J. Panduan Definitif untuk Scrum: Aturan Main. <https://www.scrumguides.org/docs/scrumguide/v2017/2017-ScrumGuide-Indonesian.pdf>, diakses tanggal 20 Mei 2023.
- [3] Dafitri, H., Elsera, M. Rancang Bangun Sistem Informasi Akademik Berbasis Web (Studi Kasus: SMA Sawasta Harapan 1, Medan), *Jurnal Sistem Informasi*, Volume 1 Nomor 2 Oktober 2017.
- [4] Ningrum, S.K, Cahyono, A.K. Implementasi Scrum Pada Manajemen Proyek Perangkat Lunak Pemesanan Undangan (Studi Kasus: paperlust), *Journal UII*, Volume 1, Nomor 1, Januari 2020.
- [5] Mutawali, L., Fathoni, B.K., & Asyhari H. Implementasi Scrum Pada Pengembangan Sistem Informasi Jasa Desain Grafis, *Jurnal Manajemen Informatika dan Sistem Informasi (MISI)*, Volume 3 Nomor 2 Juni 2020.
- [6] Hilmyansyah, M., Malabay, Simorangkir, H., & Yulhendri. Implementasi Metode Scrum Pada Pembangunan Sistem Informasi Monitoring Progress Proyek Berbasis Web (Studi Kasus: PT. Quatra Engineering Mandiri), *Jurnal Ikhraith-Informatika* Volume 6 Nomor 3, November 2022.
- [7] Ependi, U. Implementasi Model Scrum Pada Sistem Informasi Seleksi Masuk Mahasiswa Politeknik Pariwisata Palembang, *Jurnal Informatika : Jurnal Pengembangan IT (JPIT)* Volume 3 Nomor 1, Januari 2018.
- [8] Dafitri, H., & Elsera, M. Rancang Bangun Sistem Informasi Akademik Berbasis Web (studi kasus : SMA Swasta Harapan I Medan), *Query : Jurnal Sistem Informasi* Volume 1 Nomor 2, Oktober 2017
- [9] Ellis, G. *Agile Project Management : Scrum , eXtreme Programming , and Scrumban* (Issue 1). <https://doi.org/10.1016/B978-0-12-802322-8.00008-5>, 2016.
- [10] R. C. Martin dan M. Martin, *Agile Principles, Patterns, and Practice in C#*, Upper Saddle River: Pearson Education, 2006
- [11] Scrum, *Fase Scrum Metodologi, Scrum Agile*, [Online]. Available: <https://www.scrum-agile.com/scrumphases.html>, 2018.