

LAPORAN PROYEK AKHIR

**PENERAPAN METODE SCRUM PADA SISTEM
BACK-END APLIKASI MOBILE PENCARIAN
RUTE TERPENDEK DENGAN ALGORITMA
DIJKSTRA
(STUDI KASUS: PENGANTARAN PAKET DI PT.
SICEPAT PAYUNG SEKAKI)**

Wilson Wijaya
NIM 2055301146

Pembimbing
Erzi Hidayat, S.T., M.Kom

PROGRAM STUDI TEKNIK INFORMATIKA
POLITEKNIK CALTEX RIAU
2024

HALAMAN PENGESAHAN

**PENERAPAN METODE SCRUM PADA SISTEM BACK-END
APLIKASI MOBILE PENCARIAN RUTE TERPENDEK
DENGAN ALGORITMA DIJKSTRA
(STUDI KASUS: PENGANTARAN PAKET DI PT. SICEPAT
PAYUNG SEKAKI)**

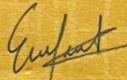
Wilson Wijaya
NIM. 2055301146

Proyek Akhir ini diajukan sebagai salah satu persyaratan untuk
memperoleh gelar Sarjana Terapan Teknik (S.Tr.Kom.)
di Politeknik Caltex Riau

Pekanbaru, 20 Agustus 2024

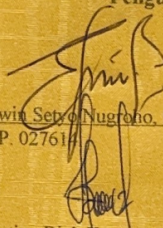
Disetujui oleh:

Pembimbing,



1. Erzi Hidayat, S.T., M.Kom.
NIP. 209002

Penguji,




1. Erwin Setyo Nugroho, S.T., M.Eng.
NIP. 027614

2. Kartina Diah Kesuma Wardhani, S.T., M.T.
NIP. 078310

Mengetahui,

Ketua Program Studi Teknik Informatika



Silvana Rasio Henim, S.ST., M.T.
NIP. 068407

ii

Politeknik Caltex Riau

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam proyek akhir yang berjudul :
**“Penerapan Metode Scrum Pada Sistem Back-End Aplikasi Mobile
Pencarian Rute Terpendek Dengan Algoritma Dijkstra Studi Kasus
: Pengantaran Paket Di PT.SICEPAT PAYUNG SEKAKI”**

Adalah benar hasil karya saya, dan tidak mengandung karya ilmiah atau tulisan yang pernah diajukan di suatu Perguruan Tinggi.

Setiap kata yang dituliskan tidak mengandung plagiat, pernah ditulis maupun diterbitkan orang lain kecuali yang secara tertulis diacu dalam laporan proyek akhir ini dan disebutkan pada daftar pustaka, Saya siap menanggung seluruh akibat terbukti melakukan plagiat.

Pekanbaru, 20 Agustus 2024



Wilson Wijaya

ABSTRAK

Dengan perkembangan teknologi saat ini, proses pembelian barang secara *online* semakin berkembang pesat. Dengan banyaknya pemesanan barang ini tentunya menambah kuantitas paket dalam proses pengiriman. PT. SICEPAT PAYUNG SEKAKI merupakan salah satu perusahaan yang bergerak dalam bidang jasa pengiriman barang, dengan banyaknya paket yang dikirim saat ini tentunya membutuhkan waktu serta tenaga yang banyak, dengan demikian maka dibutuhkan sistem pengantaran barang guna mengoptimalkan kinerja kurir dalam proses pengantaran barang. Dengan alasan demikian maka direncanakan akan dikembangkan sebuah *website* yang akan menggunakan arsitektur *REST API* dan algoritma djikstra untuk mencari rute terpendek pengantaran paket. *Website* ini dibangun dengan menggunakan metode *Scrum*. Metode ini digunakan karena lebih fleksibel dalam menyikapi setiap perubahan yang diinginkan oleh *user* guna mengoptimalkan *kualitas* produk sehingga menghasilkan keluaran dengan antarmuka yang sesuai dengan *user requirement*. Dalam proses pengembangan *website* ini hasil dari sistem yang dibangun akan diuji menggunakan *Usability Test*. Pengujian sistem dilakukan dengan menggunakan *usability testing*, berdasarkan pengujian yang dilakukan didapat hasil bahwa semua fitur pada sistem telah diimplementasikan sesuai dengan tujuan dan manfaat dari sisi *admin* dan kurir, hal ini didapat berdasarkan hasil pengujian yang dilakukan dimana rata-rata 96,67% pada *admin* dan 94,8% pada kurir sangat setuju dengan dibangunnya sistem ini.

Kata kunci : PT. SICEPAT PAYUNG SEKAKI, *REST API*, Djikstra, *Scrum*, *Usability Test*

ABSTRACT

With the advancement of technology today, the process of buying goods online has grown rapidly. This increase in orders naturally leads to more packages that need to be delivered. PT. SICEPAT PAYUNG SEKAKI is a company that provides delivery services, and with the large number of packages being sent, it requires a lot of time and effort. Therefore, a system is needed to optimize couriers' work in the delivery process. To address this, a website is planned to be developed using REST API architecture and Dijkstra's algorithm to find the shortest delivery routes. The website will be built using the Scrum method, which is chosen because it's more flexible in handling any changes requested by users, ensuring the final product meets their needs. The system will be tested through a Usability Test. Based on the testing, it was found that all the features work as intended, benefiting both admins and couriers. The results show that 96.67% of admins and 94.8% of couriers strongly agree with the system.

Keywords: *PT. SICEPAT PAYUNG SEKAKI, REST API, Djikstra, Scrum, Usability Test*

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas segala rahmat dan karunia-Nya penulis dapat menyelesaikan proyek akhir yang berjudul “PENERAPAN METODE SCRUM PADA SISTEM BACK-END APLIKASI MOBILE PENCARIAN RUTE TERPENDEK DENGAN ALGORITMA DIJKSTRA (STUDI KASUS: PENGANTARAN PAKET DI PT. SICEPAT PAYUNG SEKAKI)”. Proyek akhir ini disusun sebagai salah satu syarat untuk menyelesaikan jenjang pendidikan Diploma IV pada Program Studi Teknik Informatika Politeknik Caltex Riau.

Pada kesempatan ini, penulis ingin mengucapkan terima kasih kepada pihak yang telah banyak memberikan bantuan dan dukungan yang tiada terhingga baik secara langsung maupun tidak langsung. Ucapan terima kasih tersebut penulis tujukan kepada:

1. Tuhan Yang Maha Esa yang telah melimpahkan rahmat, hidayah, dan karunia-Nya, sehingga penulis bisa menyelesaikan tugas akhir ini tepat waktu.
2. Kedua orang tua dan keluarga penulis yang selalu memberikan doa, nasihat serta kasih sayang yang tak terhingga sehingga penulis bisa menyelesaikan proyek akhir ini tepat waktu.
3. Bapak Dr. Dadang Syarif Sihabudin Sahid, S.Si., M.Sc. selaku direktur Politeknik Caltex Riau yang telah memberikan dukungan moral dalam menyelesaikan proyek akhir ini.
4. Ibu Silvana Rasio Henim, S.S.T., M.T. selaku Ketua Program Studi Teknik Informatika yang telah memberikan izin untuk menyelesaikan proyek akhir.
5. Bapak Muhammad Arif Fadhly Ridha.S.Kom.,M.T. selaku koordinator proyek akhir sekaligus dosen wali yang telah membantu, mengarahkan, dan mengingatkan untuk selalu mengerjakan proyek akhir.
6. Bapak Erzi Hidayat,S.T.,M.Kom. selaku dosen pembimbing yang telah memberikan ilmu dan bimbingan dengan penuh kesabaran kepada penulis dalam menyelesaikan proyek akhir ini.
7. Sekali lagi kepada Bapak Erwin Setyo Nugroho, S.T., M.Eng. dan Ibu Kartina Diah Kesuma Wardhani, S.T., M.T. selaku dosen penguji yang telah memberikan masukan dan saran dalam menyelesaikan proyek akhir.

8. Seluruh dosen di Politeknik Caltex Riau khususnya Program Studi Teknik Informatika yang telah memberikan bekal ilmu kepada penulis dalam menyelesaikan proyek akhir.
9. Bapak Fadli Irham selaku koordinator PT. Sicepat Payung Sekaki yang sudah bersedia meluangkan waktunya selama proses penelitian ini.
10. Kepada teman terdekat penulis Fadillah Farhan, Muhammad Apriandi Akbar, Muhammad Rizqi, Saniyyah Azhari, dan Rasyid Tarmizi yang tidak pernah henti memberikan semangat kepada penulis untuk dapat menyelesaikan penelitian ini.
11. Teman-teman TI B G20 senasib seperjuangan yang saling memberikan dukungan sehingga penulis dapat menyelesaikan proyek akhir tepat waktu serta teman-teman dan kerabat lainnya yang tidak mungkin disebutkan satu-persatu.

Penulis sangat menyadari sepenuhnya bahwa laporan proyek akhir ini masih jauh dari sempurna, oleh karena itu segala jenis kritik, saran, dan masukan yang membangun sangat penulis harapkan agar dapat memberikan wawasan bagi pembaca dan paling utama penulis sendiri.

Pekanbaru, 09 Agustus 2024



Wilson Wijaya

DAFTAR ISI

| | |
|--|------|
| HALAMAN PENGESAHAN..... | I |
| PERNYATAAN..... | II |
| ABSTRAK | II |
| <i>ABSTRACT</i> | III |
| KATA PENGANTAR..... | IV |
| DAFTAR ISI..... | VI |
| DAFTAR GAMBAR | X |
| DAFTAR TABLE | XIII |
| BAB I PENDAHULUAN | 1 |
| 1.1 Latar Belakang | 1 |
| 1.2 Perumusan Masalah | 2 |
| 1.3 Batasan Masalah..... | 3 |
| 1.4 Tujuan dan Manfaat | 3 |
| 1.4.1 Tujuan | 3 |
| 1.4.2 Manfaat | 3 |
| 1.5 Metodologi Penelitian | 4 |
| 1.6 Sistematika Penulisan..... | 5 |
| BAB II TINJAUAN PUSTAKA..... | 6 |
| 2.1 Penelitian Terdahulu | 6 |
| 2.2 Landasan Teori..... | 10 |
| 2.2.1 Sistem Informasi Geografis | 10 |
| 2.2.2 Algoritma | 10 |
| 2.2.3 Komponen Graf | 11 |
| 2.2.4 Algoritma Djikstra | 12 |

| | | |
|------------------------------------|---|----|
| 2.2.5 | <i>Google API</i> | 13 |
| 2.2.6 | <i>Global Positioning System (GPS)</i> | 14 |
| 2.2.7 | <i>REST API</i> | 14 |
| 2.2.8 | <i>Scrum</i> | 15 |
| 2.2.9 | <i>Usability Testing</i> | 17 |
| 2.3 | Sumber Data..... | 17 |
| BAB III PERANCANGAN | | 20 |
| 3.1 | Proses Bisnis | 20 |
| 3.2 | <i>Flowchart Dijkstra</i> | 20 |
| 3.3 | Perhitungan Algoritma Dijkstra dengan Studi Kasus | 21 |
| 3.4 | Identifikasi Kebutuhan Pengguna | 23 |
| 3.5 | Perancangan Metodologi <i>Scrum</i> | 24 |
| 3.5.1 | <i>Initiate</i> | 24 |
| 3.5.1.1 | <i>Create Project Vision</i> | 24 |
| 3.5.1.2 | <i>Identify Scrum Master</i> | 24 |
| 3.5.1.3 | <i>Form Scrum</i> | 24 |
| 3.6 | Perancangan Sistem | 34 |
| 3.6.1 | Identifikasi Aktor | 34 |
| 3.6.2 | Perancangan <i>Use Case Diagram</i> | 35 |
| 3.6.3 | Skenario <i>Use Case Diagram</i> | 35 |
| 3.7 | <i>Entity Relationship Diagram</i> | 47 |
| 3.8 | Tabel <i>Database</i> | 48 |
| 3.9 | Pengujian <i>Usability Testing</i> | 49 |
| BAB IV PENGUJIAN DAN ANALISIS..... | | 62 |
| 4.1 | Implementasi Kode Program..... | 62 |
| 4.2 | Implementasi Metode <i>Scrum</i> | 67 |

| | | |
|----------|---|----|
| 4.2.1 | <i>Sprint 1</i> | 68 |
| 4.2.1.1 | <i>Sprint Planning</i> | 68 |
| 4.2.1.2 | <i>Sprint</i> | 69 |
| 4.2.1.3 | <i>Sprint Review dan Sprint Retrospective</i> | 70 |
| 4.2.2 | <i>Sprint 2</i> | 70 |
| 4.2.2.1 | <i>Sprint Planning</i> | 70 |
| 4.2.2.2 | <i>Sprint</i> | 71 |
| 4.2.2.3 | <i>Sprint Review dan Sprint Retrospective</i> | 72 |
| 4.3 | Hasil Implementasi Perancangan | 73 |
| 4.3.1 | Hasil Perancangan Admin (<i>Website</i>)..... | 73 |
| 4.3.1.1 | Antarmuka Halaman Login..... | 73 |
| 4.3.1.2 | Antarmuka Halaman Home Admin | 74 |
| 4.3.1.3 | Halaman Data Admin..... | 74 |
| 4.3.1.4 | Halaman Tambah Data Admin..... | 75 |
| 4.3.1.5 | Halaman Edit Data Admin | 75 |
| 4.3.1.6 | Halaman Data Kurir | 76 |
| 4.3.1.7 | Halaman Tambah Data Kurir | 77 |
| 4.3.1.8 | Halaman Edit Data Kurir | 78 |
| 4.3.1.9 | Halaman Data Pengantaran | 79 |
| 4.3.1.10 | Halaman Tambah Data Pengantaran..... | 79 |
| 4.3.1.11 | Halaman Edit Data Pengantaran | 80 |
| 4.3.1.12 | Halaman Data Bukti Pengantaran | 82 |
| 4.3.1.13 | Halaman Edit Data Bukti Pengantaran | 82 |
| 4.3.2 | Hasil Perancangan Kurir (<i>Android</i>)..... | 83 |
| 4.3.2.1 | Antarmuka Halaman Login..... | 83 |
| 4.3.2.2 | Halaman Akun Kurir..... | 83 |

| | | |
|---|---------------------------------------|-------|
| 4.3.2.3 | Halaman List Rute Pengantaran..... | 84 |
| 4.3.2.4 | Halaman Melihat Rute Pengantaran..... | 84 |
| 4.3.2.5 | Halaman Upload Bukti Pengantaran..... | 85 |
| 4.3.2.6 | Halaman Riwayat Pengantaran | 85 |
| 4.4 | Pengujian..... | 86 |
| 4.4.1 | <i>Pengujian Usability</i> | 86 |
| 4.5 | Analisis..... | 90 |
| BAB V KESIMPULAN DAN SARAN..... | | 91 |
| 5.1 | Kesimpulan | 91 |
| 5.2 | Saran..... | 91 |
| DAFTAR PUSTAKA | | 92 |
| LAMPIRAN A INTERVIEW | | A - 1 |
| LAMPIRAN B DOKUMENTASI WAWANCARA..... | | B - 1 |
| LAMPIRAN C IMPLEMENTASI METODE SCRUM..... | | C - 1 |
| LAMPIRAN D DAILY MEETING SCRUM..... | | D - 1 |
| LAMPIRAN E BUKTI PENGUJIAN | | E - 1 |

DAFTAR GAMBAR

| | |
|--|----|
| Gambar 2. 1 Graf Berarah dan Berbobot | 11 |
| Gambar 2. 2 Graf Tidak Berarah dan Berbobot | 11 |
| Gambar 2. 3 Graf Berarah dan Tidak Berbobot | 12 |
| Gambar 2. 4 Graf Tidak Berarah dan Tidak Berbobot | 12 |
| Gambar 2. 5 Diagram Prioritas | 16 |
| Gambar 3. 1 Flowchart Algoritma Dijkstra | 20 |
| Gambar 3. 2 Contoh Perhitungan Dijkstra | 21 |
| Gambar 3. 3 Graf Jalur yang ditempuh | 22 |
| Gambar 3. 4 Diagram Prioritas <i>Product Backlog</i> | 30 |
| Gambar 3. 5 <i>Use Case Diagram</i> | 35 |
| Gambar 3. 6 Skenario <i>Use Case Login Admin</i> | 36 |
| Gambar 3. 7 Skenario <i>Use Case Mengelola Data Admin</i> | 36 |
| Gambar 3. 8 Skenario <i>Use Case Menambahkan Data Admin</i> | 37 |
| Gambar 3. 9 Skenario <i>Use Case Mengelola Data Admin</i> | 38 |
| Gambar 3. 10 Skenario <i>Use Case Menghapus Data Admin</i> | 38 |
| Gambar 3. 11 Skenario <i>Use Case Mengelola Data Kurir</i> | 39 |
| Gambar 3. 13 Skenario <i>Use Case Menambahkan Data Kurir</i> | 39 |
| Gambar 3. 14 Skenario <i>Use Case Mengubah Data Kurir</i> | 40 |
| Gambar 3. 15 Skenario <i>Use Case Menghapus Data Kurir</i> | 41 |
| Gambar 3. 16 Skenario <i>Use Case Mengelola Rute Pengantaran</i> | 41 |
| Gambar 3. 17 Skenario <i>Use Case Menambah Rute Pengantaran</i> | 42 |
| Gambar 3. 18 Skenario <i>Use Case Mengubah Rute Pengantaran</i> | 43 |
| Gambar 3. 19 Skenario <i>Use Case Menghapus Rute Pengantaran</i> | 44 |

| | |
|---|----|
| Gambar 3. 20 Skenario <i>Use Case</i> Melakukan <i>Export</i> Laporan | |
| Pengantaran | 44 |
| Gambar 3. 21 Skenario <i>Use Case</i> Melakukan <i>Login</i> | 45 |
| Gambar 3. 22 Skenario <i>Use Case</i> Melihat Rute Pengantaran..... | 45 |
| Gambar 3. 23 Skenario <i>Use Case</i> Melakukan Konfirmasi Setelah Pengantaran Paket | 46 |
| Gambar 3. 24 Skenario <i>Use Case</i> Melihat Riwayat Pengantaran | 47 |
| Gambar 3. 25 <i>Entity Relationship Diagram (ERD)</i> | 47 |
| Gambar 4. 1 Metode <code>optimizeRoute</code> | 62 |
| Gambar 4. 2 Lanjutan Metode <code>optimizeRoute</code> | 62 |
| Gambar 4. 3 Metode <code>getPendingDetailsByKurir</code> | 63 |
| Gambar 4. 4 Metode <code>calculateDistances</code> | 63 |
| Gambar 4. 5 Metode <code>haversine</code> | 64 |
| Gambar 4. 6 Metode <code>sortLocationsByDistance</code> | 64 |
| Gambar 4. 7 Metode <code>dijkstra</code> | 65 |
| Gambar 4. 8 Metode <code>minDistance</code> | 65 |
| Gambar 4. 9 Metode <code>reconstructPath</code> | 66 |
| Gambar 4. 10 Metode <code>calculateTotalDistance</code> | 66 |
| Gambar 4. 11 Mengembalikan hasil ke dalam respons JSON | 67 |
| Gambar 4. 12 Scrum Board pada Aplikasi Jira | 68 |
| Gambar 4. 13 Halaman Login Admin..... | 73 |
| Gambar 4. 14 Halaman Home Admin..... | 74 |
| Gambar 4. 15 Halaman Menu Admin | 74 |
| Gambar 4. 16 Halaman Tambah Admin | 75 |
| Gambar 4. 17 Halaman Edit Admin..... | 76 |
| Gambar 4. 18 Halaman Menu Kurir..... | 77 |

| | |
|---|----|
| Gambar 4. 19 Halaman Tambah Kurir..... | 77 |
| Gambar 4. 20 Halaman Edit Kurir | 78 |
| Gambar 4. 21 Halaman Data Pengantaran | 79 |
| Gambar 4. 22 Halaman Tambah Data Pengantaran | 79 |
| Gambar 4. 23 Halaman Edit Data Pengantaran..... | 81 |
| Gambar 4. 24 Halaman Data Bukti Pengantaran | 82 |
| Gambar 4. 25 Halaman Edit Data Bukti Pengantaran..... | 82 |
| Gambar 4. 26 Login Kurir | 83 |
| Gambar 4. 27 Akun Kurir | 84 |
| Gambar 4. 28 Rute Pengantaran..... | 84 |
| Gambar 4. 29 Penentuan Jarak Rute | 85 |
| Gambar 4. 30 Unggah Bukti Pengantaran..... | 85 |
| Gambar 4. 31 Riwayat Pengantaran..... | 86 |

DAFTAR TABLE

| | |
|---|----|
| Tabel 2. 1 Penelitian Terdahulu | 8 |
| Tabel 2. 2 Daftar Pertanyaan <i>Interview</i> | 18 |
| Tabel 3. 1 Perhitungan Jalur Terpendek..... | 22 |
| Tabel 3. 2 <i>Form Scrum</i> | 24 |
| Tabel 3. 3 <i>User Story</i> | 25 |
| Tabel 3. 4 Penilaian Prioritas | 27 |
| Tabel 3. 5 <i>Product Backlog</i> | 27 |
| Tabel 3. 6 <i>Sprint Planning Meeting</i> | 31 |
| Tabel 3. 7 Admin..... | 48 |
| Tabel 3. 8 Kurir | 48 |
| Tabel 3. 9 Pengantaran | 48 |
| Tabel 3. 10 Detail Pengantaran | 49 |
| Tabel 3. 11 Bukti..... | 49 |
| Tabel 4. 1 Perancangan <i>Sprint 1</i> | 68 |
| Tabel 4. 2 Durasi Pengerjaan <i>Sprint 1</i> | 69 |
| Tabel 4. 3 Perencanaan <i>Sprint 2</i> | 70 |
| Tabel 4. 4 Durasi Pengerjaan <i>Sprint 2</i> | 71 |

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan zaman, tidak dapat dipungkiri bahwa masyarakat sangat membutuhkan jasa pengiriman barang. Karena persoalan demikian dapat disimpulkan bahwa semakin hari kuantitas paket yang ada pada setiap jasa pengiriman barang tentunya semakin meningkat. Penelitian ini dilakukan berdasarkan permasalahan yang terjadi di lapangan pada proses pengantaran paket yang dilakukan oleh kurir. Dengan peningkatan kuantitas pengantaran paket maka muncullah permasalahan bagaimana paket-paket tersebut dapat diantar dengan cepat dan tepat sesuai dengan estimasi waktu yang ditargetkan.

Berdasarkan permasalahan demikian, PT. SICEPAT PAYUNG SEKAKI belum memiliki solusi bagaimana agar setiap paket yang diantar dapat sampai lebih cepat dari biasanya sehingga paket yang diantar dalam satu hari dapat lebih banyak dan pekerjaan menjadi efektif. Maka berdasarkan hasil wawancara kepada koordinator dari kantor PT. SICEPAT PAYUNG SEKAKI mengungkapkan beberapa kendala dalam proses pengantaran paket. Kendala yang dihadapi kurir yaitu belum mengetahui jalur pengantaran paling efektif untuk pengantaran paket konsumennya sehingga menyebabkan terjadinya beberapa masalah. Untuk itu diperlukan sebuah sistem yang dapat membantu kurir dalam menentukan jalur mana yang perlu dilalui.

Dari permasalahan diatas maka pada penelitian ini dibuat sebuah sistem pencarian jalur terpendek yang diharapkan dapat membantu untuk menentukan alternatif jalur pengantaran paket. Dari penjelasan tersebut perlu adanya suatu cara dalam menyelesaikan masalah. Salah satunya adalah dengan menggunakan algoritma djikstra. Algoritma ini dipilih karena dapat menyelesaikan pencarian jalur terpendek dari satu simpul ke semua simpul yang ada pada suatu graf berarah. algoritma djikstra merupakan salah satu algoritma untuk menentukan rute terpendek dari suatu lokasi ke lokasi lainnya. Perhitungan algoritma djikstra diawali dengan menentukan node-node yang akan dibentuk menjadi sebuah graf. Node merupakan alamat-alamat dari penerima paket. Algoritma djikstra menentukan bobot terkecil dari node awal menuju node akhir dan pada

setiap perhitungan dari node awal menuju node selanjutnya akan diperbarui jika menemukan jarak terpendek.

Aplikasi ini dibangun berbasis website untuk admin dan berbasis android untuk kurir, sehingga kurir dapat dengan mudah melakukan akses terhadap sistem dimana saja dan kapan saja melalui *smartphone*. Serta kurir dapat melihat jalur dan rute menuju tujuan pengantaran secara *real time* ketika dalam perjalanan pengantaran paket.

Guna untuk mengintegrasikan informasi antara aplikasi dari *website* dan *mobile* maka digunakanlah arsitektur *REST API* yang memungkinkan sistem informasi berkomunikasi dan bertukar data melalui jaringan. *REST API* merupakan pendekatan arsitektur untuk komunikasi antar sistem perangkat lunak yang berbasis pada protokol HTTP. Penggunaan dari *REST API* diharapkan dapat mempermudah dalam bertukar data antar *platform*.

Dikarenakan *website* ini nantinya akan digunakan oleh *Admin* PT. PAYUNG SEKAKI, sistem ini akan dibangun dengan metode *Scrum*. Metode *Scrum* ini akan melibatkan calon pengguna pada tahap awal pengembangan, dengan tujuan agar pengguna dapat memberikan saran dan masukan pada antarmuka sistem sehingga nantinya sistem akan sesuai dengan harapan dari pengguna. Pada intinya tahapan *Scrum* merupakan tahap yang serupa dengan kerangka metode *Agile* dengan tujuan untuk menyelesaikan tugas yang kompleks dan selalu berubah. *Scrum* ini digunakan sebagai media untuk melakukan kerja sama antar tim dalam menyelesaikan masalah yang adaptif dan kompleks, sehingga dapat menghasilkan keluaran yang inovatif dan kreatif.

1.2 Perumusan Masalah

Berdasarkan latar belakang yang telah diuraikan terdapat beberapa rumusan masalah sebagai berikut:

- 1) Bagaimana penerapan metode Scrum memengaruhi proses pengembangan sistem backend pada aplikasi mobile pencarian rute terpendek?
- 2) Bagaimana menerapkan Algoritma Dijkstra dalam menentukan rute terpendek yang memudahkan kurir dalam melakukan pengantaran barang?
- 3) Seberapa efektif penggunaan Algoritma Dijkstra dalam mencari rute terpendek terhadap pengantaran paket?

1.3 Batasan Masalah

Adapun batasan masalah dalam pembuatan proyek akhir ini adalah:

- 1) *Website* yang dirancang berbasis web menggunakan bahasa pemrograman PHP
- 2) *Website* yang dirancang menggunakan *framework* Codeigniter
- 3) *Website* ini hanya dapat digunakan oleh *Admin* PT. SICEPAT PAYUNG SEKAKI.
- 4) *Website* ini dirancang untuk pengantaran paket dengan roda dua dan roda empat
- 5) *Website* dapat digunakan pada area yang memiliki akses internet
- 6) *Website* belum terintegrasi dengan aplikasi lainnya
- 7) *Website* dibangun untuk PT. SICEPAT PAYUNG SEKAKI Kota Pekanbaru.
- 8) *Website* ini menggunakan arsitektur *REST API*.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan dari penelitian ini adalah:

- 1) Menghasilkan sebuah sistem navigasi yang dapat menunjukkan rute terpendek
- 2) Mengimplementasikan algoritma djikstra pada sistem yang dikembangkan
- 3) Mengimplementasikan metode *Scrum* dalam proses pengembangan system

1.4.2 Manfaat

Manfaat yang diharapkan dari penelitian ini adalah:

- 1) Mempermudah kurir PT. SICEPAT PAYUNG SEKAKI Kota Pekanbaru untuk menentukan rute terpendek dalam proses pengantaran paket
- 2) Meningkatkan efisiensi waktu pada proses pengantaran paket
- 3) Mempermudah kurir dalam proses pengantaran paket

1.5 Metodologi Penelitian

Metode penelitian yang dipakai dalam pembuatan proyek akhir ini adalah:

- 1) Studi Literatur
Studi literatur dilakukan dengan cara mengumpulkan serta membaca buku-buku jurnal, *paper*, makalah maupun situs internet.
- 2) Melakukan wawancara
Dilakukannya secara langsung dengan kurir PT. SICEPAT PAYUNG SEKAKI serta pengumpulan data dengan melakukan dialog secara langsung dan tanya jawab untuk memperoleh informasi mengenai kondisi dan permasalahan dalam pengolahan data informasi dan permasalahan yang ada akan menjadi pedoman dalam pengerjaan proyek akhir.
- 3) Melakukan pengumpulan data
Pada tahapan ini data-data yang terkait dengan pengerjaan proyek akhir ini akan dikumpulkan dan di analisa dari hasil wawancara yang telah dilakukan.
- 4) *Product backlog*
Product backlog adalah daftar yang diprioritaskan dari apa saja yang diperlukan, diberi peringkat berdasarkan nilai bagi pelanggan dengan item bernilai tertinggi di bagian atas daftar. *Product backlog* berkembang sepanjang masa proyek dan item harus ditambahkan, dihapus atau diprioritaskan ulang.
- 5) *Scrum*
Scrum menyusun pengembangan produk dalam siklus kerja yang disebut *Sprint*, iterasi pekerjaan yang biasanya 1-4 minggu. *Sprint* memiliki durasi tetap dan berakhir pada tanggal tertentu apakah pekerjaan telah selesai atau belum.
- 6) *Daily scrum meeting*
Daily scrum meeting adalah pertemuan yang dilakukan setiap harinya untuk membahas apa saja yang telah dikerjakan pada hari itu.
- 7) *Review Sprint* dan *Sprint* Retrospektif
Review sprint adalah pemeriksaan hasil *sprint* yang telah dibuat. Setelah *review sprint*, dilakukan retrospektif untuk mengevaluasi bagaimana cara bekerja lebih baik lagi secara tim maupun individu.

8) Pengujian

Pengujian berfungsi untuk mengukur tingkat kualitas dari sebuah *website* dan mengevaluasi apakah sudah sesuai dengan hasil yang diharapkan. Pengujian yang akan dilakukan pada penelitian ini yaitu *usability test*

1.6 Sistematika Penulisan

Sistematika penulisan proposal proyek akhir ini secara keseluruhan terdiri dari empat bab, masing-masing terdiri dari beberapa sub bab. Adapun pokok pembahasan dari masing-masing bab tersebut secara garis besar sebagai berikut :

BAB I PENDAHULUAN

Bab ini menguraikan tentang latar belakang masalah, perumusan masalah dan lingkup masalah, tujuan dan manfaat proyek akhir, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini menguraikan beberapa hasil penelitian terdahulu dan landasan teori yang diperlukan untuk merancang sistem.

BAB III PERANCANGAN

Bab ini menjelaskan tentang perancangan sistem terdiri dari perancangan sistem yang akan dibangun.

BAB IV PENGUJIAN DAN ANALISIS

Bab ini berisi penjelasan mengenai pengujian sistem dan analisa mengenai hasil yang didapat.

BAB V KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dan saran setelah melakukan pengujian dan analisa sistem

BAB II TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian mengenai pencarian rute terpendek ini telah banyak dilakukan sebelumnya. Oleh karena itu diperlukan studi literatur agar tidak ditemukannya kesamaan penelitian dan terulangnya penelitian. Penelitian yang dilakukan oleh Sunardi dkk., (2019) dari Universitas Ahmad Dahlan yang berjudul “Implementasi Algoritma Dijkstra dan Algoritma Semut Untuk Analisis Rute Transjogja Berbasis Android”. Penelitian ini bertujuan untuk membandingkan kinerja dari algoritma djikstra dan algoritma semut dalam penentuan jalur terpendek dan tercepat. Hasil dari penelitian yang telah dilakukan adalah rata-rata waktu proses dan penggunaan memori dari algoritma djikstra lebih kecil dibandingkan algoritma *ant colony optimization* sehingga sangat cocok dijalankan pada aplikasi yang menggunakan fitur navigasi penentuan jalur.

Penelitian yang dilakukan oleh Suryani dan Murniyasih, (2022) dari Politeknik Saint Paul Sorong yang berjudul “Pencarian Rute Terpendek Pada Aplikasi Ojek Sampah Dengan Menggunakan Algoritma Dijkstra” dengan menggunakan pendekatan *waterfall*. Penelitian ini bertujuan untuk mencari rute terpendek dengan algoritma djikstra dan menampilkan estimasi waktu tempuh untuk mempermudah sopir mendapatkan informasi rute terpendek penjemputan sampah. Hasil dari penelitian yang telah dilakukan adalah aplikasi yang dibangun dapat menentukan rute terpendek untuk proses pengangkutan sampah pada rumah warga berdasarkan inputan alamat awal dari supir menuju tujuan penjemputan.

Penelitian yang dilakukan oleh Kurniawan dkk., (2018) dari Universitas Malikussaleh Lhokseumawe yang berjudul “Sistem Informasi Geografis Pencarian Rute Terpendek Menuju Rumah Sakit Dengan Menggunakan Metode Algoritma Dijkstra” dengan menggunakan pendekatan *prototyping*. Penelitian ini bertujuan untuk memudahkan pengguna dalam mencari rumah sakit terdekat di sekitar Aceh. Hasil dari penelitian yang telah dilakukan adalah aplikasi dapat menampilkan lokasi rumah sakit terdekat beserta waktu tempuh tercepat dan rute terpendeknya, aplikasi ini menggunakan 2 algoritma yaitu

algoritma *hill climbing* dan algoritma djikstra yang membedakan dari kedua algoritma ini adalah algoritma *hill climbing* akan berhenti jika mencapai nilai optimum lokal dan hanya berjalan dalam satu operasi/langkah terhadap *node* sedangkan algoritma djikstra melakukan kalkulasi terhadap semua bobot terkecil / terdekat dari *node* yang ada.

Penelitian lainnya adalah penelitian yang dilakukan oleh Sibarani dan Paridudin, (2022) dari Universitas Satya Negara Indonesia yang berjudul “Rancang Bangun Aplikasi Pencarian Masjid Terdekat di Wilayah Jakarta Barat Menggunakan Algoritma Dijkstra Berbasis Android”. Penelitian ini bertujuan untuk memudahkan pengguna dalam mencari rute terpendek dari lokasi pengguna berada ke masjid yang ingin dituju. Hasil dari penelitian yang telah dilakukan adalah penggunaan dari algoritma djikstra lebih efisien dan lebih menghemat waktu untuk digunakan masyarakat umum yang sedang berpergian dan ingin mencari masjid terdekat didaerah tersebut yang ingin melakukan sholat.

Penelitian lainnya adalah penelitian yang dilakukan oleh (Yosua dkk., 2021) dari Universitas Mercu Buana yang berjudul “Implementasi Algoritma Dijkstra dalam Pencarian Klinik Hewan Terdekat” dengan menggunakan pendekatan *Scrum*. Penelitian ini bertujuan untuk membantu para pemilik hewan yang kesulitan dalam mencari klinik dan toko yang menjual berbagai macam kebutuhan hewan terdekat dari lokasi pengguna berada. Hasil dari penelitian yang dilakukan adalah algoritma djikstra dapat membantu pengguna untuk menemukan klinik hewan terdekat dari lokasi pengguna berada pada saat itu.

Berdasarkan beberapa penelitian terdahulu yang telah dilakukan, membuktikan bahwa algoritma djikstra mampu menentukan rute terpendek, selain itu algoritma ini telah banyak digunakan untuk pemecahan masalah pencarian rute terpendek. Penelitian ini bertujuan untuk membuat *website & aplikasi mobile* dengan mengimplementasikan algoritma djikstra dalam menentukan rute terpendek yang memudahkan kurir dalam melakukan pengantaran paket pada sistem ini akan menampilkan rute terdekat berdasarkan alamat-alamat yang akan di tuju. Diharapkan penelitian ini dapat memenuhi kebutuhan pengguna dengan baik dan melengkapi kekurangan-kekurangan yang dimiliki oleh penelitian sebelumnya.

Tabel 2. 1 Penelitian Terdahulu

| Penelitian | Judul | Bahasa Pemrograman | Metode / Algoritma | Hasil Penelitian |
|--------------------------------|--|---------------------------|---|--|
| Sunardi dkk., (2019) | Implementasi Algoritma Djikstra dan Algoritma Semut Untuk Analisis Rute Transjogja Berbasis Android | Java | Djikstra & <i>Ant Colony Optimization</i> / - | Rata-rata waktu proses dan penggunaan memori dari algoritma Djikstra lebih kecil dibandingkan <i>Ant Colony Optimization</i> . |
| Suryani dan Murniyasih, (2022) | Pencarian Rute Terpendek Pada Aplikasi Ojek Sampah Dengan Menggunakan Algoritma Djikstra | Dart | Djikstra / <i>Waterfall</i> | Aplikasi dapat menentukan rute terpendek untuk proses pengangkutan sampah pada rumah warga |
| Kurniawan dkk., (2018) | Sistem Informasi Geografis Pencarian Rute Terpendek Menuju Rumah Sakit Dengan Menggunakan Algoritma Djikstra | PHP | Djikstra / <i>Prototyping</i> | Aplikasi dapat menampilkan lokasi rumah sakit terdekat beserta waktu tempuh tercepat dan rute terpendeknya |
| Sibarani dan | Rancang Bangun Aplikasi | Menunjukkan, Dart | Djikstra / - | Aplikasi dapat menampilkan letak masjid |

| Penelitian | Judul | Bahasa Pemrograman | Metode / Algoritma | Hasil Penelitian |
|---------------------|--|---------------------------|---------------------------|--|
| Paridudin, (2022) | Pencarian Masjid Terdekat Di Wilayah Jakarta Barat Menggunakan Algoritma Dijkstra Berbasis Android | | | terdekat dari lokasi pengguna berada |
| Yosua dkk., (2021) | Implementasi Algoritma Dijkstra Dalam Pencarian Klinik Hewan Terdekat | Dart | Dijkstra / <i>Scrum</i> | Aplikasi dapat menentukan lokasi klinik hewan terdekat dan menampilkan rute terpendek sehingga waktu tempuh yang dibutuhkan sangat cepat |
| Penelitian Sekarang | Penerapan Metode Scrum Pada Sistem Back-End Aplikasi Mobile Pencarian Rute Terpendek Dengan Algoritma Dijkstra | PHP | Dijkstra / <i>Scrum</i> | Metode yang digunakan dapat di aplikasikan dengan baik pada aplikasi dan membantu kurir dalam mengurutkan alamat pengantaran barang berdasarkan jarak terdekat dari posisi kurir |

| Penelitian | Judul | Bahasa Pemrograman | Metode / Algoritma | Hasil Penelitian |
|------------|-------|--------------------|--------------------|--|
| | | | | dan juga menampilkan rute terpendek yang akan dilalui oleh kurir |

2.2 Landasan Teori

2.2.1 Sistem Informasi Geografis

Istilah “informasi geografis” mempunyai makna mengenai informasi baik tempat-tempat yang terletak di permukaan bumi, pengetahuan mengenai letak titik di mana suatu objek di permukaan bumi berada, dan informasi mengenai atribut atau keterangan-keterangan yang terdapat di permukaan bumi serta posisinya diberikan atau diketahui (Albes Fajri, dkk, 2018). Sistem Informasi Geografis atau Geographic Information System (GIS) juga merupakan suatu sistem informasi yang berbasis pada komputer, yang dirancang serta dioperasikan dengan menggunakan data yang mempunyai informasi spasial (bereferensi ke ruangan). Sistem ini mengambil, menganalisis, mengecek, mengintegrasikan, mengelola atau memproses, dan menampilkan data secara spasial serta merepresentasikan pada bentuk atau kondisi di bumi (Kurniawan et al., 2018).

2.2.2 Algoritma

Ditelaah dari asal usul kata, kata algorism berarti proses mengjitung dengan angka Arab. Kata tersebut berasal dari nama penulis buku AArab yang terkenal bernama Abu Ja’far Muhammad Ibnu Musa al-Khuwarizimi di mana orang barat membaca al-Khuwarizimi menjadi algorism. Beliau menulis buku yang berjudul Kitab al jabar walmuqubala, yang artinya “Buku Pemugaran dan Pengurangan” (The Book of Restoration and Reduction). Dari judul buku itu juga merupakan asal kata “aljabar” (algebra). Perubaha kata algorism menjadi algorithm terjadi karena kata algorism sering dikelirukan dengan arithmetic, sehingga

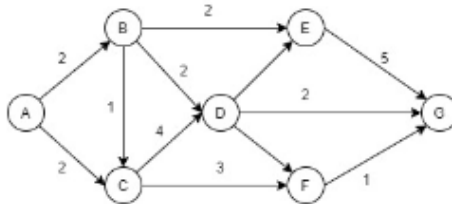
akhiran -sm berubah menjadi -thm. Lambat laun kata algorithm berangsur-angsur dipakai sebagai metode perhitungan (komputasi) secara umum, sehingga kehilangan makna aslinya. Dalam bahasa Indonesia, kata algorithm diserap menjadi algoritma (Karlina, 2022).

2.2.3 Komponen Graf

Komponen graf adalah jumlah maksimum graf bagian dalam sebuah graf G (Sibarani & Paridudin, 2022).

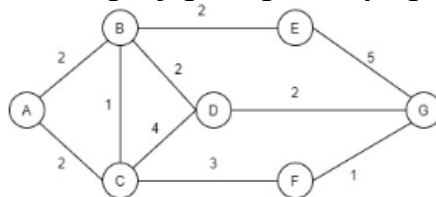
Menurut arah dan bobotnya, graf dibagi menjadi 4, yaitu:

- Pada gambar 2.1 merupakan graf berarah dan berbobot dimana tiap busur mempunyai anak panah dan bobot. Graf berarah dan berbobot yang terdiri dari tujuh titik yaitu titik A, B, C, D, E, F, G. Titik menunjukkan arah ke titik B dan titik C, titik B menunjukkan arah ke titik D dan titik C, dan seterusnya. Bobot antar titik A dan titik B pun telah diketahui.



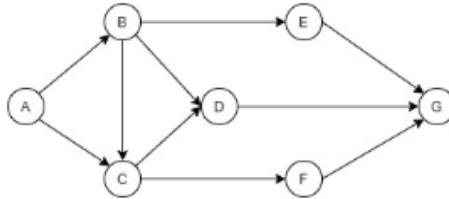
Gambar 2. 1 Graf Berarah dan Berbobot

- Pada gambar 2.2 merupakan merupakan graf tidak berarah dan berbobot dimana tiap busur tidak mempunyai anak panah tetapi mempunyai bobot. Pada gambar di bawah menunjukkan graf tidak berarah dan berbobot. Graf terdiri dari tujuh titik yaitu titik A, B, C, D, E, F, G. Titik A tidak menunjukkan arah ke titik B atau C, namun bobot antara titik A dan titik B telah diketahui. Begitu juga dengan titik yang lain.



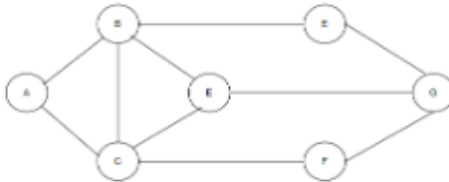
Gambar 2. 2 Graf Tidak Berarah dan Berbobot

- c. Pada gambar 2.3 merupakan graf berarah dan tidak berbobot dimana tiap busur mempunyai anak panah yang tidak berbobot.



Gambar 2. 3 Graf Berarah dan Tidak Berbobot

- d. Pada gambar 2.4 merupakan graf tidak berarah dan tidak berbobot dimana tiap busur tidak mempunyai anak panah dan tidak berbobot.



Gambar 2. 4 Graf Tidak Berarah dan Tidak Berbobot

2.2.4 Algoritma Dijkstra

Metode yang tepat dalam penentuan jalur transportasi terpendek adalah dengan menggunakan algoritma Dijkstra. Algoritma Dijkstra ditemukan oleh Edsger W. Dijkstra dan dipublikasi pada tahun 1959 pada sebuah jurnal *Numerische Mathematik* yang berjudul “A Note on Two Problems in Connexion with Graphs “. Algoritma ini sering digambarkan sebagai algoritma *greedy*. Algoritma ini digunakan dalam memecahkan permasalahan untuk jarak terpendek (*shortest path problem*) untuk *graph* berarah di mana setiap arahnya memiliki nilainya masing-masing sesuai dengan panjangnya jarak yang ditempuh. Metode ini membentuk sebuah jaringan yang terdiri dari rangkaian noda (*node*) dan arah (*arrow*). Algoritma Dijkstra bekerja dengan cara menghitung semua *vertex* atau titik yang tersedia. Algoritma Dijkstra dapat menemukan jalur terpendek pada *graph* yang memiliki *vertex* dan jarak antar *vertex* yang memiliki bobot positif (Nurritzky et al., 2020).

Cara kerja algoritma Dijkstra memakai strategi greedy, di mana pada setiap langkah dipilih sisi dengan bobot terkecil yang menghubungkan sebuah simpul lain yang belum terpilih. Algoritma Dijkstra membutuhkan parameter berupa tempat asal dan tempat tujuan (Junanda et al., 2016). Algoritma Dijkstra akan menentukan titik node yang menjadi awalan (*origin*) dan memberikan nilai atau bobot ke node yang terdekat hingga *node-node* selanjutnya satu persatu (Nurrizky et al., 2020) Logika pada Algoritma Dijkstra sebagai berikut:

1. Dari titik awal ke setiap titik lainnya diberikan nilai bobot (jarak), dengan cara set nilai 0 pada node awal dan nilai tak hingga terhadap node lain yang belum memiliki bobot nilai.
2. Set node awal sebagai node keberangkatan kemudian ukur jarak dari node awal ke setiap node yang ada.
3. Dari node keberangkatan, pertimbangkan node lain yang belum dihitung jaraknya dari titik keberangkatan. Misalnya, dengan nilai keberangkatan O ke A memiliki nilai bobot jarak 3 dan dari A ke node D berjarak 2, maka jarak ke D melewati A menjadi $3+2=5$.
4. Hitung setiap jarak terhadap node lain. Nilai jarak yang telah tersimpan merupakan nilai jarak terpendek terakhir sehingga tidak perlu mengulang dan ke belakang.
5. Set node belum terjamah dengan jarak terkecil (dari node keberangkatan) sebagai node keberangkatan selanjutnya dan lanjutkan kembali ke step 3.

2.2.5 Google API

Google API adalah kumpulan data suatu peta yang dapat membantu dalam menambahkan data peta di google. Dan dapat membantu dalam membuat aplikasi web atau seluler (android) yang menarik dengan platform pemetaan dari Google, termasuk dalam basis data citra satelit, *street view*, profil ketinggian, petunjuk arah, analisis yang basis data peta besar (Parapat et al., 2017).

Dengan cakupan global yang paling akurat di dunia, dan komunitas pemetaan yang aktif memperbarui setiap kali pengguna dapat manfaat dari layanan yang ditingkatkan secara terus menerus.

2.2.6 *Global Positioning System (GPS)*

GPS adalah sistem navigasi yang menggunakan satelit yang didesain agar dapat menyediakan posisi secara instan, kecepatan dan informasi waktu di hampir semua tempat di muka bumi, setiap saat dan dalam kondisi cuaca apapun. Alat untuk menerima sinyal satelit yang dapat digunakan oleh pengguna secara umum dinamakan GPS Tracker atau GPS Tracking, dengan menggunakan alat ini maka kemungkinan user dapat melacak posisi kendaraan, armada ataupun mobil dalam keadaan Real-Time (Roza, 2020).

2.2.7 *REST API*

Teknologi *Representational State Transfer* (REST) diciptakan oleh Roy Fielding dari *University of California*. *REST* sering digunakan untuk pengembangan layanan berbasis *web* atau *mobile* karena ringan dan sederhana. Prinsip utama desain dari *REST API* adalah kinerja, portabilitas dan kesederhanaan.

REST API mendukung beragam sistem untuk berinteraksi serta menerima/mengirim data dengan mudah. Setiap penggunaan *REST API*, didukung oleh URL dan HTTP. Data dalam *database* pada suatu aplikasi dipetakan dengan *endpoint* API pada *REST API* (Naufal et al., 2022).

REST API menggunakan HTTP untuk menentukan *request* dan *response* agar informasi dapat lebih sederhana dibaca dan dianalisa pada sisi *client*, keluaran yang dihasilkan oleh *API server* berupa JSON. Beberapa *method* dan tipe *request* yang tersedia untuk melakukan komunikasi data, antara lain :

1. GET, adalah *method* yang digunakan untuk mendapatkan data dari *server*.
2. POST, adalah *method* yang digunakan untuk melakukan *create* data dan data yang telah dibuat akan disimpan ke dalam *database*.
3. DELETE, adalah *method* yang digunakan untuk menghapus data dalam *database*.

2.2.8 Scrum

Menurut Schwaber (2016) yang dimuat pada Suhaimi dkk. (2020), *Scrum* adalah kerangka kerja di mana sebuah tim menyelesaikan masalah adaptif yang rumit secara produktif kreatif dan menghasilkan produk dengan nilai tinggi (Fadli, 2023).

Pada tim *scrum* terdapat tiga peran yang bersifat terorganisir, *scrum* didesain untuk mengoptimalkan *flexibility*, *creativity*, dan *productivity*, berikut adalah peran dari tim *scrum*:

- 1) *Product Owner*, adalah orang yang bertanggung jawab untuk memaksimalkan hasil dari produk berdasarkan pekerjaan dari *Development Team*.
- 2) *Scrum Master*, adalah orang yang bertanggung jawab untuk membimbing tim agar dapat berjalan dengan baik.
- 3) *Development Team*, adalah profesional yang mengerjakan produk sampai berstatus selesai pada setiap akhir *sprint*.

Scrum memiliki beberapa tahapan atau cara kerja dalam proses pengembangan *website*, diantaranya:

- 1) *Product Backlog*
Product Backlog adalah daftar yang diprioritaskan dari apa saja yang diperlukan, diberi peringkat berdasarkan nilai bagi pelanggan dengan item bernilai tertinggi di bagian atas daftar. *Product backlog* berkembang sepanjang masa proyek dan item terus ditambahkan, dihapus, atau diprioritaskan ulang.
- 2) *Sprint*
Scrum menyusun pengembangan produk dalam siklus kerja yang disebut *Sprint*, iterasi pekerjaan yang biasanya 1-4 minggu. *Sprint* memiliki durasi tetap dan berakhir pada tanggal tertentu apakah pekerjaan telah selesai atau belum mereka tidak pernah diperluas.
- 3) *Sprint Planning*
Di awal setiap *Sprint*, pertemuan perencanaan *Sprint* dilakukan. Pemilik produk dan tim *Scrum* meninjau *product backlog*, membahas tujuan dan konteks untuk item, dan tim *Scrum* memilih item dari *product backlog* untuk dikerjakan dan menyelesaikan pada akhir *Sprint*, mulai di bagian atas *product backlog*. Setiap item yang dipilih dari *product*

backlog dirancang dan kemudian dipecah menjadi serangkaian tugas individu.

- 4) *Daily Scrum Meeting*
Meeting ini adalah pertemuan singkat (15 menit) yang dilakukan setiap hari kerja pada waktu yang ditentukan. Pada pertemuan ini, informasi yang diperlukan untuk memeriksa kemajuan disajikan. Informasi ini dapat mengakibatkan perencanaan ulang dan diskusi lebih lanjut segera setelah rapat harian *Scrum*
- 5) *Review Sprint* dan Retrospektif
Setelah *Sprint* berakhir, ada *review Sprint*, di mana tim *Scrum* dan pemangku kepentingan memeriksa apa yang dilakukan selama *Sprint*, mendiskusikannya dan mencari tahu apa yang harus dilakukan selanjutnya. Setelah *review Sprint*, tim berkumpul untuk retrospeksi *Sprint* yang merupakan kesempatan bagi tim untuk membahas apa yang bekerja dan apa yang tidak bekerja, lalu menyetujui segala bentuk perubahan.

Dalam tahapan penentuan nilai *product backlog* digunakanlah diagram prioritas sebagai penentuan *product backlog* mana yang akan menjadi prioritas. Dalam penilaian *product backlog* akan dilakukan dengan cara mempertimbangkan nilai (*value*) dan usaha (*effort*).



Gambar 2. 5 Diagram Prioritas

Menurut Ibanez (2021), terdapat 4 uraian penilaian dari diagram prioritas antara lain:

- 1) Kuadran *Do Now*
Kuadran ini tidak membutuhkan *effort* yang banyak dan *value* yang dimiliki paling penting. *Product backlog* adalah kuadran dengan fitur paling penting yang masuk dalam kategori yang akan di kerjakan pada *sprint* awal.
- 2) Kuadran *Do Soon*
Dengan mengerjakan kuadran ini *effort* dibutuhkan besar, *value* juga tidak penting namun tidak ada kebutuhan mendesak. Pada *product backlog* masuk ke dalam kuadran ini biasanya berada di *sprint* kedua dan kasus tertentu *product backlog* yang masuk dalam kategori akan pindah kuadran *Do Now*.
- 3) Kuadran *Do Later*
Kuadran ini memerlukan *effort* yang kecil dan *value* nya yang tidak terlalu signifikan. *Product backlog* yang masuk pada tahap ini akan diproses setelah *product backlog* yang berada di kuadran *do now* dan *do soon* telah selesai.
- 4) Kuadran *Do Last*
Kuadran ini memiliki *effort* yang besar dan *value* nya kecil. *Product backlog* yang masuk pada kuadran ini akan menjadi *product backlog* yang pada umumnya akan masuk pada *sprint* terakhir. Tim tidak perlu fokus pada *product backlog* yang ada pada kuadran ini.

2.2.9 Usability Testing

Usability testing merupakan cara bagi developer *software* dan *hardware* untuk melihat dan menilai kesesuaian interaksi yang terjadi antara pengguna dengan produk sebelum digunakan secara publik, proses *usability testing* ini melibatkan beberapa orang sebagai pengguna yang mewakili target pengguna (Wahab, 2021).

2.3 Sumber Data

Adapun sumber data yang akan digunakan dalam penelitian ini adalah berdasarkan proses wawancara yang dilakukan dengan pegawai dari PT. SICEPAT PAYUNG SEKAKI khususnya kurir dan *Admin*

gudang yang bertugas dalam proses pembagian barang yang akan diantar oleh kurir berdasarkan daerah tertentu. Berikut ini merupakan beberapa data dari hasil wawancara yang dilakukan oleh *Development Team*.

Tabel 2. 2 Daftar Pertanyaan *Interview*

| Kode | Pertanyaan |
|------|---|
| P001 | Apakah terjadi perubahan signifikan pada transaksi pengantaran paket saat covid dan setelah covid? |
| P002 | Berapa jumlah paket yang terdapat di Region Pekanbaru? |
| P003 | Berapa jumlah paket yang dikirim di daerah region arengka dalam sehari? |
| P004 | Dalam sehari berapa lama waktu kerja yang dilakukan oleh kurir? |
| P005 | Kapan jumlah paket dari SICEPAT PAYUNG SEKAKI mengalami peningkatan? |
| P006 | Apakah ada tanggal atau momen tertentu jumlah paket sangat banyak sehingga menyulitkan proses pengantaran? |
| P007 | Bagaimana sistem kerja dari proses pengantaran paket pada SICEPAT PAYUNG SEKAKI ? (dari barang disortir di gudang sampai barang tiba ditangan konsumen) |
| P008 | Apakah admin kantor berperan dalam pembagian distribusi paket yang akan dikirim oleh kurir? Jika ada tolong dijelaskan. |
| P009 | Apakah PT. SICEPAT PAYUNG SEKAKI sudah memiliki aplikasi yang dapat menentukan list rute untuk pengantaran paket sehingga proses pengantaran lebih mudah? |
| P010 | Bagaimana sistem pembagian pengantaran paket setiap kurir, apakah pembagian ditentukan berdasarkan jarak, wilayah, dan jenis paket atau kurir dapat memilih paket yang ingin diantar? |
| P011 | Apakah kurir dapat menentukan jumlah paket yang ingin diantar di hari itu? |

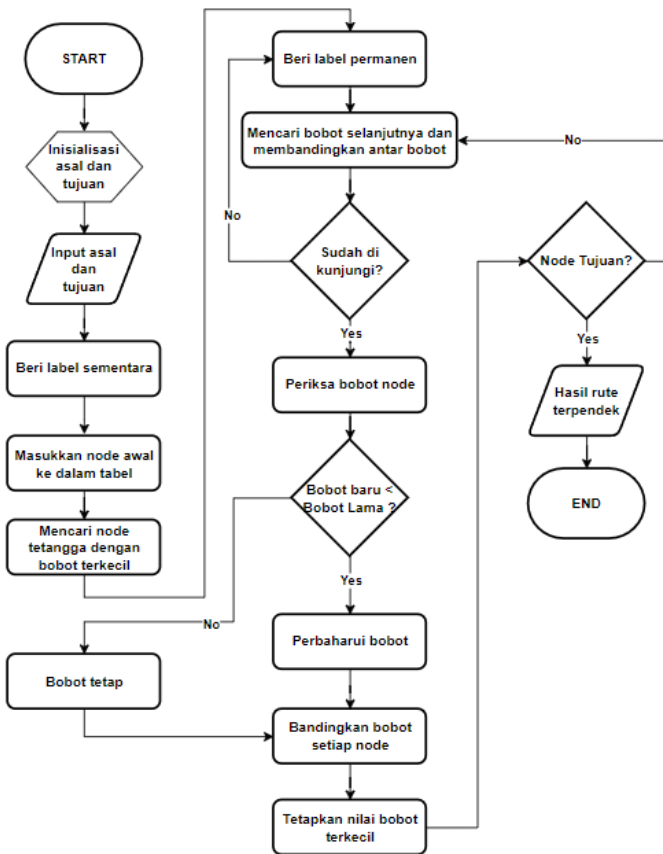
| | |
|------|--|
| P012 | Apa saja hal yang menjadi penghambat saat proses pengiriman paket? |
| P013 | Jika terjadi hal seperti ban bocor, cuaca tidak mendukung, kesalahan dalam pengambilan paket, apakah kurir akan melakukan proses konfirmasi dengan konsumen? |
| P014 | Jika aplikasi ini dibuat apakah harapan kurir untuk aplikasi ini? Apakah bisa digunakan untuk mencari lokasi atau yang lainnya. |

BAB III PERANCANGAN

3.1 Proses Bisnis

Pada *website* pencarian rute terpendek pada proses pengantaran barang di PT. SICEPAT PAYUNG SEKAKI diharapkan dapat memudahkan kurir lebih mudah dalam menjalankan tugasnya, sistem ini dibangun dengan 2 aktor yaitu, *Admin* dan kurir.

3.2 Flowchart Dijkstra

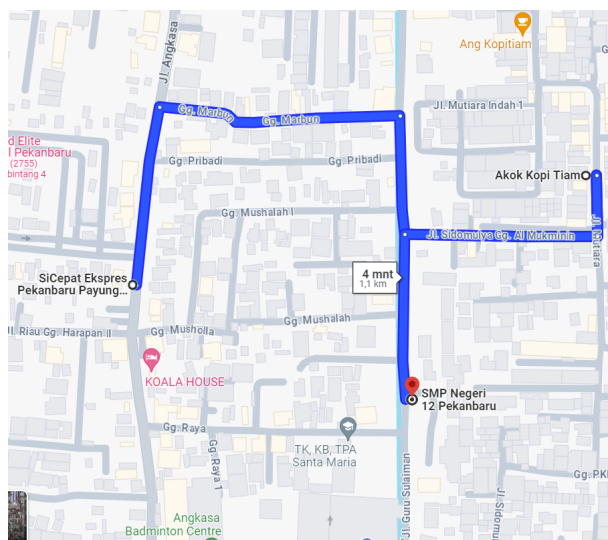


Gambar 3. 1 Flowchart Algoritma Dijkstra

Pada Gambar 3.1, merupakan flowchart algoritma djikstra. Flowchart ini dimulai dengan menentukan lokasi awal lalu djikstra akan melakukan pencarian dari satu titik ke titik lainnya dengan memberikan bobot pada titik atau node pertama ke node terdekat satu per satu. Lalu, djikstra akan membandingkan jarak dari node awal ke node yang belum dilewati, jika jarak tersebut lebih kecil dari jarak sebelumnya hapus data lama dan simpan ulang dengan data jarak yang baru. Node yang sudah dilewati tidak akan dilewati kembali. Langkah tersebut berulang untuk setiap node yang belum dilewati dan jarak yang disimpan adalah jarak terkecil yang merupakan hasil rute terpendek yang direkomendasikan.

3.3 Perhitungan Algoritma Dijkstra dengan Studi Kasus

Perhitungan matematika digunakan untuk memperlihatkan proses bagaimana algoritma djikstra berjalan. Tentukan lintasan terpendek untuk pengantaran paket dari SiCepat Express Pekanbaru Payung Sekaki ke CV. GEMA RIAU



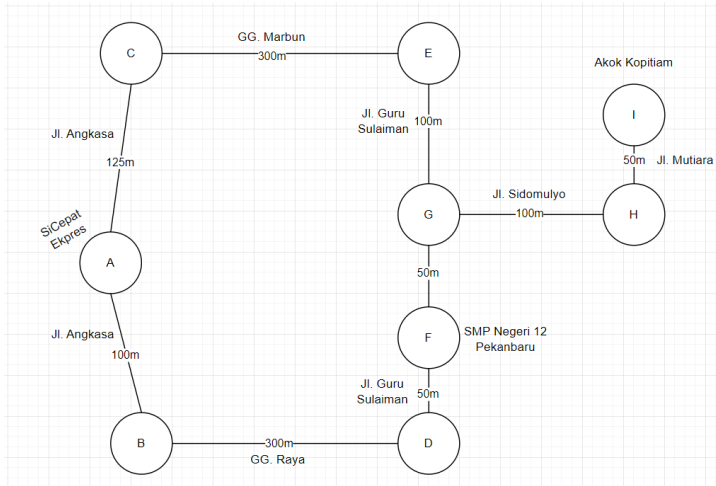
Gambar 3. 2 Contoh Perhitungan Dijkstra

Perhitungan dilakukan dengan tabel dan dilakukan evaluasi dimana titik awalnya adalah SiCepat Ekspres Pekabaru Payung Sekaki, dan terdapat 2 titik tujuan yang akan dikirim yaitu SMP Negeri 12 Pekanbaru dan Akok Kopi Tiam. Untuk dapat menentukan rute terpendek

pengantaran paket dilakukan evaluasi dimana terdapat 2 kemungkinan yaitu :

1. Rute Start -> SMP Negeri 12 Pekanbaru -> Akok Kopitiam
2. Rute Start -> Akok Kopitiam -> SMP Negeri 12 Pekanbaru

Untuk menentukannya dilakukan perhitungan jarak dan perbandingan. Graf Jalur yang ditempuh akan ditampilkan pada Gambar 3.3 di bawah ini.



Gambar 3. 3 Graf Jalur yang ditempuh

Dari gambar di atas, lakukan perhitungan dengan Tabel 3.1 di bawah ini dengan membandingkan jarak pada setiap titik yang dilalui.

| Rute | Titik yang Dilewati | Total Jarak (meter) |
|-------|---|---------------------|
| A-F-I | A → B(100m) → D(400m) → F(450m) → G(500m) → H(600m) → I(650m) | 650 |
| A-I-F | A → C(125m) → E(425m) → G(525m) → H(625m) → I(675m) → H(725m) → G(825m) → F(875m) | 875 |

Tabel 3. 1 Perhitungan Jalur Terpendek

Dari perhitungan yang dilakukan antara rute A-F-I dan A-I-F didapat kesimpulan bahwa rute terpendek yaitu rute A-F-I dimana rute yang dilewati yaitu :

A→B→D→F→G→H→I dengan total jarak 650m

Sehingga pengantaran paket dilakukan mulai dari mengantar paket pada titik F yaitu SMP Negeri 12 Pekanbaru kemudian dilanjutkan dengan paket kedua titik I yaitu Akok Kopitiam

3.4 Identifikasi Kebutuhan Pengguna

Untuk mengidentifikasi kebutuhan pengguna telah dilakukan wawancara dengan pihak PT. SICEPAT PAYUNG SEKAKI untuk mendapatkan data kebutuhan, berikut adalah tahapan pengumpulan kebutuhan pengguna :

1) Identifikasi subjek wawancara

Orang yang akan diwawancarai adalah Bapak Fadli Irham sebagai koordinator PT. SICEPAT PAYUNG SEKAKI. Dengan adanya wawancara ini bertujuan untuk memenuhi kebutuhan yang nantinya akan digunakan untuk pengembangan *website* pencarian rute terpendek dalam proses pengantaran barang pada PT. SICEPAT PAYUNG SEKAKI.

2) Melakukan Wawancara

Pada tahap wawancara bersama Bapak Fadli Irham pada tanggal 22 Februari 2024 sudah dilakukan, proses wawancara ini dilakukan dengan mengajukan pertanyaan yang terdapat pada formulir pertanyaan, berikut ini merupakan foto dokumentasi wawancara yang sudah dilakukan bersama Bapak Fadli Irham.

3) Hasil Wawancara

Setelah proses wawancara dilakukan bersama Bapak Fadli Irham didapatkan jawaban-jawaban dari beberapa pertanyaan yang sudah diberikan. Hasil dari wawancara dengan Bapak Fadli Irham ditampilkan pada lampiran di bawah.

3.5 Perancangan Metodologi *Scrum*

Tahapan selanjutnya yang dilakukan adalah mengimplementasikan metode *Scrum* pada proses pengembangan *website* ini. Metodologi *Scrum* ini dipilih karena lebih terstruktur dan memiliki transparansi yang lebih tinggi. Adapun beberapa tahapan yang dilakukan dalam proses implementasi dari metode *Scrum* adalah sebagai berikut ini :

3.5.1 *Initiate*

Initiate merupakan tahap awal yang bertujuan untuk membuat visi dari proyek secara menyeluruh dan *backlog* proyek atau daftar tugas yang harus dikerjakan oleh tim *Scrum*. Adapun pembagian dari proses *initiate* ini adalah sebagai berikut :

3.5.1.1 Create Project Vision

Visi utama dari proyek yang akan dikembangkan ini adalah menciptakan sebuah *website* untuk mencari rute terbaik dalam proses pengantaran barang yang akan digunakan oleh karyawan PT. SICEPAT PAYUNG SEKAKI. *Website* ini diharapkan dapat mempermudah dan mempercepat kurir dalam pengantaran barang.

3.5.1.2 Identify Scrum Master

Dalam proses pengembangan ini akan terdapat seorang *Scrum Master*, yaitu Bapak Erzi Hidayat, S.T., M.Kom. Adapun peran dari *Scrum Master* yaitu memastikan kelancaran selama proses pengembangan.

3.5.1.3 Form Scrum

Tabel 3. 2 *Form Scrum*

| Role | Nama | Tugas |
|----------------------|----------------------------|--|
| <i>Product Owner</i> | Fadli Irham | Orang yang bertanggung jawab untuk membimbing tim agar dapat berjalan dengan baik. |
| <i>Scrum Master</i> | Erzi Hidayat, S.T., M.Kom. | Orang yang bertanggung jawab untuk |

| | | |
|---------------------------|---------------|--|
| | | memaksimalkan hasil dari produk berdasarkan pekerjaan dari Team Development. |
| <i>Back End Developer</i> | Wilson Wijaya | Fokus pada pembuatan rancangan struktur basis data serta implementasi dari fitur-fitur yang terdapat pada <i>website</i> |

Selama proses pengembangan *website* terdapat tahapan-tahapan kerja dalam proses *implementasi* metodologi *scrum* sebagai berikut tahapan nya :

1) Pengumpulan Kebutuhan

Pada tahap pengumpulan kebutuhan dengan tujuan mendapatkan cara *product backlog* yang dikerjakan, agar kebutuhan *product backlog* terpenuhi maka ada tahapan-tahapan berikut :

a) *User Story*

Tabel 3. 3 *User Story*

| <i>ID</i> | <i>User Story</i> |
|-----------|--|
| US01 | Sebagai <i>Admin Kantor</i> Saya ingin melihat <i>dashboard</i> sehingga saya dapat melihat data apa saja yang sudah masuk ke dalam sistem. |
| US02 | Sebagai <i>Admin Kantor</i> Saya ingin mengelola data <i>Admin</i> seperti menambahkan, mengubah dan menghapus data <i>Admin</i> , sehingga <i>Admin</i> lain juga dapat berkontribusi dalam mengelola data lainnya |
| US03 | Sebagai <i>Admin Kantor</i> Saya ingin mengelola data kurir seperti menambahkan, mengubah dan menghapus data |

| <i>ID</i> | <i>User Story</i> |
|-----------|--|
| | kurir. Sehingga saya dapat mengatur rute perjalanan berdasarkan daftar kurir yang sudah saya miliki |
| US04 | <p>Sebagai <i>Admin</i> Kantor</p> <p>Saya ingin mengelola data rute pengantaran paket seperti menambahkan, mengubah, mengekspor dan menghapus data rute pengantaran. Dengan adanya data ini kemudian sistem akan memberikan rekomendasi rute terpendek yang dapat dilalui oleh kurir sehingga memudahkan kurir dalam proses pengantaran paket</p> |
| US05 | <p>Sebagai Kurir</p> <p>Saya ingin melihat rute pengantaran dari paket-paket yang akan saya antar sehingga memudahkan saya untuk mengantar paket dengan cepat.</p> |
| US06 | <p>Sebagai Kurir</p> <p>Saya ingin melakukan konfirmasi pengantaran paket jika sudah berhasil mengirim paket ataupun terjadi penundaan pengiriman paket pada kondisi tertentu</p> |
| US07 | <p>Sebagai Kurir</p> <p>Saya ingin melihat Riwayat rute pengantaran paket yang sudah saya antar, dengan demikian saya dapat memastikan kembali mengenai paket-paket yang sudah saya kirim.</p> |

b) Nilai prioritas *product backlog*

Dalam menetapkan nilai prioritas pada *product backlog*, tujuannya adalah untuk menentukan prioritas pengerjaan yang harus dilakukan terlebih dahulu dalam daftar *product backlog*. Penilaian prioritas dilakukan berdasarkan dua faktor utama, yaitu nilai (*value*) yang diberikan oleh setiap item *product backlog* dan usaha (*effort*) yang diperlukan untuk menyelesaikannya.

Berikut adalah hasil penilaian prioritas pada *product backlog*.

Tabel 3. 4 Penilaian Prioritas

| No. | <i>Value</i> | <i>Effort</i> | Nilai |
|-----|--------------|---------------|-------|
| 1. | <i>High</i> | <i>Low</i> | 4 |
| 2. | <i>High</i> | <i>High</i> | 3 |
| 3. | <i>Low</i> | <i>Low</i> | 2 |
| 4. | <i>Low</i> | <i>High</i> | 1 |

Tabel 3. 5 *Product Backlog*

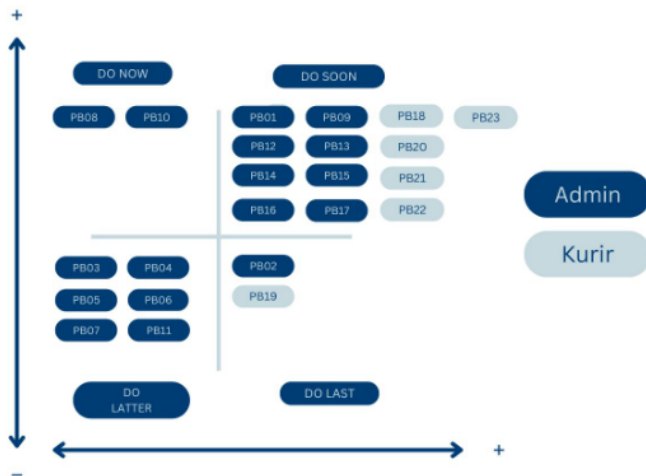
| <i>User</i> | <i>ID</i> | <i>Product Backlog</i> | <i>Durasi (Hari)</i> | <i>Value</i> | <i>Effort</i> | <i>Nilai Prioritas</i> |
|---------------------|-----------|--|----------------------|--------------|---------------|------------------------|
| <i>Admin, Kurir</i> | PB01 | Membuat halaman <i>login</i> | 4 | <i>High</i> | <i>High</i> | 3 |
| <i>Admin</i> | PB02 | Membuat halaman Dashboard | 2 | <i>Low</i> | <i>High</i> | 1 |
| | PB03 | Membuat halaman Kelola data <i>Admin</i> | 2 | <i>Low</i> | <i>Low</i> | 2 |
| | PB04 | Membuat halaman tambah data <i>Admin</i> | 2 | <i>Low</i> | <i>Low</i> | 2 |
| | PB05 | Membuat halaman ubah data <i>Admin</i> | 3 | <i>Low</i> | <i>Low</i> | 2 |
| | PB06 | Membuat halaman hapus data <i>Admin</i> | 1 | <i>Low</i> | <i>Low</i> | 2 |

| <i>User</i> | <i>ID</i> | <i>Product Backlog</i> | <i>Durasi (Hari)</i> | <i>Value</i> | <i>Effort</i> | <i>Nilai Prioritas</i> |
|-------------|-----------|---|----------------------|--------------|---------------|------------------------|
| | PB07 | Membuat halaman Kelola data kurir | 2 | <i>Low</i> | <i>Low</i> | 2 |
| | PB08 | Membuat halaman tambah data kurir | 2 | <i>High</i> | <i>Low</i> | 4 |
| | PB09 | Membuat halaman ubah data kurir | 3 | <i>High</i> | <i>High</i> | 3 |
| | PB10 | Membuat halaman hapus data kurir | 2 | <i>High</i> | <i>Low</i> | 4 |
| | PB11 | Membuat halaman Kelola rute pengantaran | 2 | <i>Low</i> | <i>Low</i> | 2 |
| | PB12 | Membuat halaman tambah rute pengantaran | 5 | <i>High</i> | <i>High</i> | 3 |
| | PB13 | Membuat halaman ubah rute pengantaran | 2 | <i>High</i> | <i>High</i> | 3 |
| | PB14 | Membuat halaman hapus | 1 | <i>High</i> | <i>High</i> | 3 |

| <i>User</i> | <i>ID</i> | <i>Product Backlog</i> | <i>Durasi (Hari)</i> | <i>Value</i> | <i>Effort</i> | <i>Nilai Prioritas</i> |
|-------------|-----------|--|----------------------|--------------|---------------|------------------------|
| | | rute pengantaran | | | | |
| | PB15 | Membuat halaman kelola bukti pengantaran | 2 | <i>High</i> | <i>High</i> | 3 |
| | PB16 | Membuat halaman ubah dan hapus bukti pengantaran | 2 | <i>High</i> | <i>High</i> | 3 |
| | PB17 | Membuat API untuk menghubungkan ke aplikasi Mobile | 11 | <i>High</i> | <i>High</i> | 3 |
| | PB18 | Mendeploy sistem ke hosting | 2 | <i>High</i> | <i>High</i> | 3 |
| Kurir | PB19 | Membuat halaman profil kurir | 4 | <i>Low</i> | <i>High</i> | 1 |
| | PB20 | Membuat halaman list rute pengantaran | 4 | <i>High</i> | <i>High</i> | 3 |

| <i>User</i> | <i>ID</i> | <i>Product Backlog</i> | <i>Durasi (Hari)</i> | <i>Value</i> | <i>Effort</i> | <i>Nilai Prioritas</i> |
|-------------|-----------|--|----------------------|--------------|---------------|------------------------|
| | PB21 | Membuat halaman lihat rute pengantaran | 4 | High | High | 3 |
| | PB22 | Membuat halaman upload bukti pengantaran | 4 | High | High | 3 |
| | PB23 | Membuat halaman riwayat pengantaran | 4 | High | High | 3 |

Diagram prioritas pada gambar 3.4 berikut adalah diagram yang menentukan prioritas *product backlog*.



Gambar 3. 4 Diagram Prioritas *Product Backlog*

2) *Sprint Planning Meeting*

Pada *sprint planning meeting* tim *developer* memilih *product backlog* yang akan dikerjakan untuk menentukan tujuan dari *product backlog* tersebut. Setelah diskusi dengan tim *scrum* berikut rancangan *sprint planning* dari sistem.

Tabel 3. 6 *Sprint Planning Meeting*

| <i>Sprint</i> | <i>Product Backlog</i> | Durasi Pengerjaan (Hari) | PIC |
|--|--|---------------------------------|-----------------------------------|
| <i>Sprint 1</i> (6 Mei – 28 Mei) 23 hari | Membuat halaman <i>Login (All User)</i> | 4 | Saniyyah Azhari, Wilson Wijaya |
| | Membuat halaman <i>dashboard</i> | 2 | |
| | Membuat halaman kelola data <i>Admin</i> | 2 | |
| | Membuat halaman tambah data <i>Admin</i> | 2 | |
| | Membuat halaman ubah data <i>Admin</i> | 3 | |
| | Membuat halaman hapus data <i>Admin</i> | 1 | |
| | Membuat halaman Kelola data kurir | 2 | |
| | Membuat halaman tambah data kurir | 2 | |

| | | | |
|--|--|----|-----------------------------------|
| | Membuat halaman ubah data kurir | 3 | |
| | Membuat halaman hapus data kurir | 2 | |
| <i>Sprint 2</i> (4 Juni – 20 Juli) 47 hari | Membuat halaman Kelola rute pengantaran | 2 | Saniyyah Azhari, Wilson Wijaya |
| | Membuat halaman tambah rute pengantaran | 5 | |
| | Membuat halaman ubah rute pengantaran | 2 | |
| | Membuat halaman hapus rute pengantaran | 1 | |
| | Membuat halaman kelola bukti pengantaran | 2 | |
| | Membuat halaman ubah dan hapus bukti pengantaran | 2 | |
| | Membuat API untuk menghubungkan ke aplikasi Mobile | 11 | Wilson Wijaya |

| | | | |
|--|--|---|-----------------|
| | Mendeploy sistem ke hosting | 2 | |
| | Membuat halaman profil kurir | 4 | Saniyyah Azhari |
| | Membuat halaman list rute pengantaran | 4 | |
| | Membuat halaman lihat rute pengantaran | 4 | |
| | Membuat halaman upload bukti pengantaran | 4 | |
| | Membuat halaman riwayat pengantaran | 4 | |
| | | | |

3) *Sprint Backlog*

Setelah *sprint planning meeting* terlaksana dengan lancar, maka tim *developer* akan mengerjakan satu persatu *product backlog* yang sudah dipilih, durasi dalam pengerjaan *product backlog* ini berkisar antara 1-5 minggu. Selama pengerjaan *sprint backlog* akan dilakukan *daily scrum meeting* yang di mana pertemuan ini akan dilakukan selama 15 menit guna membahas apa saja yang telah dikerjakan sebelumnya, apa yang sedang dikerjakan pada saat ini, dan apa saja kendala yang dihadapi selama proses pengerjaan.

Dengan adanya *daily scrum meeting* ini akan memudahkan untuk mendapatkan informasi apakah ada kendala mengenai *progress* pengerjaan proyek dari setiap tim.

4) *Sprint Review* dan *Sprint Retrospective*

Sprint review merupakan tahapan di mana setiap *product backlog* yang sudah dikerjakan oleh tim akan di *review*, yang di mana jika ada *product backlog* yang pengerjaannya belum selesai maka proses pengerjaannya akan dilanjutkan. Apabila *product backlog* sebelumnya sudah selesai maka akan dilanjutkan proses pengerjaan *product backlog* selanjutnya. Tahapan *sprint review* ini melibatkan *project owner* serta tim *developer* dan tidak menutup kemungkinan *scrum master* akan terlibat dalam proses *sprint review*.

Sprint retrospective melibatkan keseluruhan tim *scrum* dengan tujuan untuk mengevaluasi kinerja dari seluruh tim apakah sudah mengimplementasikan *scrum* dalam proses kerjanya, dengan demikian *sprint retrospective* ini tidak membahas mengenai sistem yang akan dikembangkan.

3.6 Perancangan Sistem

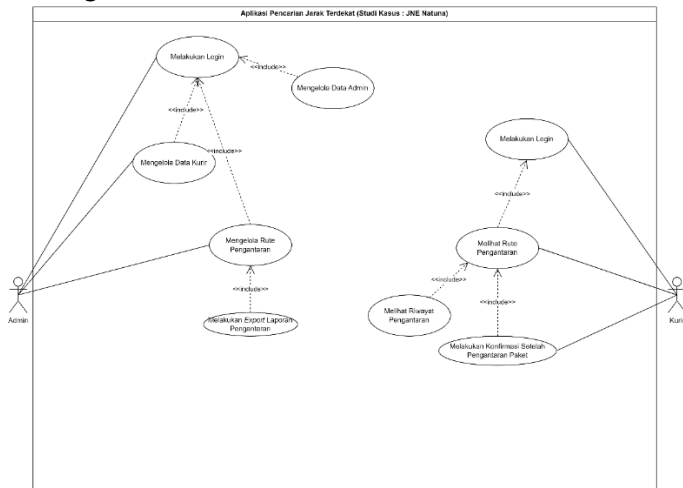
3.6.1 Identifikasi Aktor

Identifikasi aktor ini bertujuan untuk menentukan siapa saja aktor yang akan digunakan untuk membuat *use case diagram* untuk menjelaskan fungsi dari sistem yang di mana pada sistem ini akan terdapat 2 aktor yaitu kurir dan *Admin* dari PT. SICEPAT PAYUNG SEKAKI.

Aktor dengan *role Admin* ini mempunyai wewenang untuk mengelola data *Admin*, data kurir serta data paket yang akan diantarkan, sedangkan aktor dengan *role kurir* dapat melihat rute dan melihat riwayat rute pengantaran.

3.6.2 Perancangan Use Case Diagram

Berdasarkan kebutuhan sistem, maka dirancang *use case diagram* dengan aktor sebagai berikut :



Gambar 3. 5 Use Case Diagram

Use case pada gambar 3.3 merupakan *use case* dari keseluruhan sistem yang akan dikembangkan oleh tim *scrum*.

3.6.3 Skenario Use Case Diagram

Skenario *use case diagram* ini digunakan untuk mendeskripsikan aktor-aktor yang akan melakukan prosedur dalam sistem

1) Melakukan Login

Use case : Login Admin

Aktor : Admin

Kondisi Awal : Admin sudah memiliki akun

Kondisi Akhir : Admin berhasil login dan masuk ke dalam sistem

| | | |
|----------------------------|---|---|
| Usecase | Melakukan <i>Login</i> | |
| Aktor | <i>Admin</i> | |
| Kondisi Awal | <i>Admin</i> sudah memiliki akun | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. <i>Admin</i> mengisi formulir <i>login</i> dengan <i>username</i> dan <i>password</i> yang telah di daftarkan | |
| | | 2. Mengecek kesesuaian akun |
| | | 3. Menampilkan halaman <i>dashboard</i> |
| Skenario Alternatif | 1. Jika <i>username</i> tidak terdaftar, maka sistem akan menampilkan pesan error " <i>Username</i> tidak terdaftar". 2. Jika <i>username</i> atau <i>password</i> tidak diisi, maka sistem akan menampilkan pesan error " <i>Isi form</i> dengan benar". 3. Jika <i>password</i> yang dimasukkan salah, maka sistem akan menampilkan pesan error " <i>Password</i> yang dimasukkan salah". | |
| Kondisi Akhir | <i>Admin</i> berhasil <i>login</i> dan masuk ke dalam sistem | |

Gambar 3. 6 Skenario *Use Case Login Admin*

2) Mengelola Data *Admin*

Use case : Mengelola Data *Admin*

Aktor : *Admin*

Kondisi Awal : *Admin* telah *login* ke dalam sistem

Kondisi Akhir : *Admin* memilih aksi tambah *Admin* atau list *Admin*

| | | |
|----------------------------|---|--|
| Usecase | Mengelola Data <i>Admin</i> | |
| Aktor | <i>Admin</i> | |
| Kondisi Awal | <i>Admin</i> telah <i>login</i> ke dalam sistem | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. <i>Admin</i> memilih menu Data <i>Admin</i> | |
| | | 2. Menampilkan halaman Data <i>Admin</i> |
| | 3. <i>Admin</i> memilih sub-menu Tambah <i>Admin</i> atau List <i>Admin</i> | |
| | | 4. Menampilkan aksi kelola data <i>admin</i> sesuai dengan sub-menu yang dipilih |
| Skenario Alternatif | - | |
| Kondisi Akhir | <i>Admin</i> memilih aksi Tambah <i>Admin</i> atau List <i>Admin</i> | |

Gambar 3. 7 Skenario *Use Case Mengelola Data Admin*

3) Menambahkan Data *Admin*

Use case : Menambahkan Data *Admin*

Aktor : *Admin*

Kondisi Awal : *Admin* memilih *sub-menu* Tambah *Admin* pada halaman Data *Admin*

Kondisi Akhir : *Admin* dapat menambahkan data *Admin*

| | | |
|----------------------------|--|--|
| Usecase | Menambahkan Data Admin | |
| Aktor | <i>Admin</i> | |
| Kondisi Awal | <i>Admin</i> memilih sub-menu Tambah Admin pada halaman Data Admin | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan formulir Tambah Admin |
| | 2. <i>Admin</i> mengisi formulir tambah data admin berupa Nama Lengkap, Nomor HP, <i>Username</i> , dan <i>Password</i> . | |
| | 3. <i>Admin</i> menekan tombol Tambah | |
| | | 4. Memvalidasi dan menyimpan informasi yang dimasukkan |
| | | 5. Menampilkan halaman Data Admin |
| Skenario Alternatif | 1. Jika menekan tombol Batal, maka akan kembali ke formulir Tambah Admin 2. Jika Nama Lengkap tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> "Nama Lengkap tidak sesuai dengan aturan". 3. Jika Nomor HP tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> "Nomor HP tidak sesuai dengan aturan". 4. Jika <i>Password</i> tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> " <i>Password</i> tidak sesuai dengan aturan". 5. Jika diantara Nama Lengkap, Nomor HP, <i>Username</i> , atau <i>Password</i> tidak diisi, maka sistem akan menampilkan pesan <i>error</i> "Lengkapi <i>Form</i> dengan benar". | |
| Kondisi Akhir | <i>Admin</i> dapat menambahkan data admin | |

Gambar 3. 8 Skenario *Use Case* Menambahkan Data *Admin*

4) Mengubah Data *Admin*

Use case : Mengubah Data *Admin*

Aktor : *Admin*

Kondisi Awal : *Admin* memilih *sub-menu* List *Admin* pada halaman Data *Admin*

Kondisi Akhir : *Admin* dap at mengubah data *Admin*

| | | |
|----------------------------|---|--|
| Usecase | Mengubah Data Admin | |
| Aktor | Admin | |
| Kondisi Awal | Admin memilih sub-menu List Admin pada halaman Data Admin | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan table data List Admin |
| | 2. Admin memilih tombol Edit pada data yang ingin diubah | |
| | | 3. Menampilkan formulir Ubah Data Admin |
| | 4. Admin mengubah data yang ingin diubah seperti Nama Lengkap, Nomor HP, Username, dan Password | |
| | 5. Admin menekan tombol Ubah | |
| | | 6. Memvalidasi dan menyimpan informasi yang dimasukkan |
| | 7. Menampilkan halaman Data Admin | |
| Skenario Alternatif | 1. Jika menekan tombol Batal, maka akan kembali ke table data List Admin 1. Jika Nama Lengkap tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "Nama Lengkap tidak sesuai dengan aturan". 2. Jika Nomor HP tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "Nomor HP tidak sesuai dengan aturan". 3. Jika Password tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "Password tidak sesuai dengan aturan". 4. Jika diantara Nama Lengkap, Nomor HP, Username, atau Password tidak diisi, maka sistem akan menampilkan pesan error "Lengkapi Form dengan benar". | |
| Kondisi Akhir | Admin dapat mengubah data admin | |

Gambar 3. 9 Skenario Use Case Mengelola Data Admin

5) Menghapus Data Admin

Use case : Menghapus Data Admin

Aktor : Admin

Kondisi Awal : Admin memilih sub-menu List Admin pada halaman Data Admin

Kondisi Akhir : Admin dapat menghapus data Admin

| | | |
|----------------------------|--|--|
| Usecase | Menghapus Data Admin | |
| Aktor | Admin | |
| Kondisi Awal | Admin memilih sub-menu List Admin pada halaman Data Admin | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan table data List Admin |
| | 2. Admin memilih tombol Hapus pada data yang ingin dihapus | |
| | | 3. Menampilkan pesan konfirmasi Hapus Data |
| | 4. Memilih tombol Hapus | |
| | | 5. Melakukan update pada database |
| | 6. Menampilkan table data List Admin | |
| Skenario Alternatif | 1. Jika memilih tombol Batal, maka data admin tidak akan dihapus | |
| Kondisi Akhir | Admin dapat menghapus data admin | |

Gambar 3. 10 Skenario Use Case Menghapus Data Admin

6) Mengelola Data Kurir

Use case : Mengelola Data Kurir

Aktor : *Admin*

Kondisi Awal : *Admin* telah *login* ke dalam sistem

Kondisi Akhir : *Admin* memilih aksi tambah kurir atau list kurir

| | | |
|---------------------|--|---|
| Usecase | Mengelola Data Kurir | |
| Aktor | Admin | |
| Kondisi Awal | Admin telah login ke dalam sistem | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. Admin memilih menu Data Kurir | 2. Menampilkan halaman Data Kurir |
| | 3. Admin memilih sub-menu Tambah Kurir atau List Kurir | 4. Menampilkan aksi kelola data kurir sesuai dengan sub-menu yang dipilih |
| | | |
| Skenario Alternatif | - | |
| Kondisi Akhir | Admin memilih aksi Tambah Kurir atau List Kurir | |

Gambar 3. 11 Skenario *Use Case* Mengelola Data Kurir

7) Menambahkan Data Kurir

Use case : Menambahkan Data Kurir

Aktor : *Admin*

Kondisi Awal : *Admin* memilih *sub-menu* Tambah Kurir pada halaman Data Kurir

Kondisi Akhir : *Admin* dapat menambahkan data kurir

| | | |
|---------------------|--|--|
| Usecase | Menambahkan Data Kurir | |
| Aktor | Admin | |
| Kondisi Awal | Admin memilih sub menu Tambah Kurir pada halaman Data Kurir | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan formulir Tambah Kurir |
| | 2. Admin mengisi formulir tambah data kurir berupa Nama Lengkap, Nomor HP, Username, Password, Region, Nomor Polisi. | |
| | 3. Admin menekan tombol Tambah | 4. Memvalidasi dan menyimpan informasi yang dimasukkan |
| | | 5. Menampilkan halaman Data Kurir |
| Skenario Alternatif | 1. Jika menekan tombol Batal, maka akan kembali ke formulir Tambah kurir 2. Jika Nama Lengkap tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "Nama Lengkap tidak sesuai dengan aturan" 3. Jika No. HP tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "No. HP tidak sesuai dengan aturan" 4. Jika Password tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "Password tidak sesuai dengan aturan" 5. Jika diantara Nama Lengkap, Nomor HP, Username, Password, Region, atau Nomor Polisi tidak diisi, maka sistem akan menampilkan pesan error "Lengkapi Form dengan benar". | |
| Kondisi Akhir | Admin dapat menambahkan data kurir | |

Gambar 3. 12 Skenario *Use Case* Menambahkan Data Kurir

8) Mengubah Data Kurir

Use case : Mengubah Data Kurir
Aktor : *Admin*
Kondisi Awal : *Admin* memilih *sub-menu* List Kurir pada halaman Data Kurir
Kondisi Akhir : *Admin* dapat mengubah data kurir

| | | |
|----------------------------|---|--|
| Usecase | Mengubah Data Kurir | |
| Aktor | <i>Admin</i> | |
| Kondisi Awal | <i>Admin</i> memilih sub-menu List Kurir pada halaman Data Kurir | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan <i>table</i> data List Kurir |
| | 2. <i>Admin</i> memilih tombol Edit pada data yang ingin diubah | |
| | | 3. Menampilkan formulir Ubah Data Kurir |
| | 4. <i>Admin</i> mengubah data yang ingin diubah seperti Nama Lengkap, Nomor HP, <i>Username</i> , <i>Password</i> , <i>Region</i> , dan Nomor Polisi | |
| | 5. <i>Admin</i> menekan tombol Ubah | |
| | | 6. Memvalidasi dan menyimpan informasi yang dimasukkan |
| | 7. Menampilkan halaman Data Kurir | |
| Skenario Alternatif | 1. Jika menekan tombol Batal, maka akan kembali ke <i>table</i> data List Kurir 2. Jika Nama Lengkap tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> "Nama Lengkap tidak sesuai dengan aturan" 3. Jika Nomor HP tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> "Nomor HP tidak sesuai dengan aturan". 4. Jika <i>Password</i> tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> " <i>Password</i> tidak sesuai dengan aturan". 5. Jika diantara Nama Lengkap, Nomor HP, <i>Username</i> , <i>Password</i> , <i>Region</i> , atau Nomor Polisi tidak diisi, maka sistem akan menampilkan pesan <i>error</i> "Lengkapi <i>Form</i> dengan benar". | |
| Kondisi Akhir | <i>Admin</i> dapat mengubah data kurir | |

Gambar 3. 13 Skenario *Use Case* Mengubah Data Kurir

9) Menghapus Data Kurir

Use case : Menghapus Data Kurir
Aktor : *Admin*
Kondisi Awal : *Admin* memilih *sub-menu* List Kurir pada halaman Data Kurir
Kondisi Akhir : *Admin* menghapus data kurir

| | | |
|----------------------------|--|--|
| Usecase | Menghapus Data Kurir | |
| Aktor | Admin | |
| Kondisi Awal | Admin memilih sub-menu List Kurir pada halaman Data Kurir | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan table data List Kurir |
| | 2. Admin memilih tombol Hapus pada data yang Ingin dihapus | |
| | | 3. Menampilkan pesan konfirmasi Hapus Data |
| | 4. Memilih tombol Hapus | |
| | | 5. Melakukan update pada database |
| | | 6. Menampilkan table data List Kurir |
| Skenario Alternatif | 1. Jika memilih tombol Batal, maka data kurir tidak akan dihapus | |
| Kondisi Akhir | Admin dapat menghapus data kurir | |

Gambar 3. 14 Skenario *Use Case* Menghapus Data Kurir

10) Mengelola Rute Pengantaran

Use case : Mengelola Rute Pengantaran

Aktor : Admin

Kondisi Awal : Admin telah *login* ke dalam sistem

Kondisi Akhir : Admin memilih aksi Tambah atau List Rute

| | | |
|----------------------------|--|---|
| Usecase | Mengelola Rute Pengantaran | |
| Aktor | Admin | |
| Kondisi Awal | Admin telah login ke dalam sistem | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. Admin memilih menu Rute Pengantaran | |
| | | 2. Menampilkan halaman Rute Pengantaran |
| | 3. Admin memilih sub-menu Tambah Rute atau List Rute | |
| | | 4. Menampilkan aksi kelola rute pengantaran sesuai dengan sub-menu yang dipilih |
| Skenario Alternatif | - | |
| Kondisi Akhir | Admin memilih aksi Tambah Rute atau List Rute | |

Gambar 3. 15 Skenario *Use Case* Mengelola Rute Pengantaran

11) Menambah Rute Pengantaran

- Use case* : Menambah Rute Pengantaran
 Aktor : *Admin*
 Kondisi Awal : *Admin* memilih *sub-menu* Tambah Rute pada halaman Rute Pengantaran
 Kondisi Akhir : *Admin* dapat menambahkan rute pengantaran paket

| | | |
|----------------------------|--|--|
| Usecase | Menambah Rute Pengantaran | |
| Aktor | <i>Admin</i> | |
| Kondisi Awal | <i>Admin</i> memilih sub-menu Tambah Rute pada halaman Rute Pengantaran | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan formulir Tambah Rute |
| | 2. <i>Admin</i> mengisi formulir tambah data kurir berupa Tanggal, <i>Region</i> , Nama Kurir, Jumlah Paket, dan Informasi Rute berupa Nomor Resi, Nama Penerima, Nomor HP, Alamat Penerima, dan Titik Rute. | |
| | 3. <i>Admin</i> menekan tombol Tambah | |
| | | 4. Memvalidasi dan menyimpan informasi yang dimasukkan |
| | | 5. Menampilkan halaman Rute Pengantaran Paket |
| Skenario Alternatif | 1. Jika menekan tombol Batal, maka akan kembali ke formulir Tambah Rute 2. Jika Nama Penerima tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> "Nama Penerima tidak sesuai dengan aturan". 3. Jika Nomor HP tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan <i>error</i> "No. HP tidak sesuai dengan aturan". 4. Jika diantara Tanggal, <i>Region</i> , Nama Kurir, Jumlah Paket, Nomor Resi, Nama Penerima, Nomor HP, Alamat Penerima, atau Titik Rute tidak diisi, maka sistem akan menampilkan pesan <i>error</i> "Lengkapi <i>Form</i> dengan benar". | |
| Kondisi Akhir | <i>Admin</i> dapat menambahkan rute pengantaran paket | |

Gambar 3. 16 Skenario *Use Case* Menambah Rute Pengantaran

12) Mengubah Rute Pengantaran

- Use case* : Mengubah Rute Pengantaran
 Aktor : *Admin*
 Kondisi Awal : *Admin* memilih *sub-menu* List Rute pada halaman Rute Pengantaran
 Kondisi Akhir : *Admin* dapat mengubah rute pengantaran

| | | |
|----------------------------|--|--|
| Usecase | Mengubah Rute Pengantaran | |
| Aktor | Admin | |
| Kondisi Awal | Admin memilih sub-menu List Rute pada halaman Rute Pengantaran | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan <i>table</i> data List Rute |
| | 2. Admin memilih tombol Edit pada data yang ingin diubah | |
| | | 3. Menampilkan formulir Ubah Data Rute |
| | 4. Admin mengisi data yang ingin diubah berupa Tanggal, Region, Nama Kurir, Jumlah Paket, dan Informasi Rute berupa Nomor Resi, Nama Penerima, Nomor HP, Alamat Penerima, dan Titik Rute. | |
| | 4. Admin menekan tombol Ubah | |
| | | 5. Memvalidasi dan menyimpan informasi yang dimasukkan |
| | 6. Menampilkan <i>table</i> data List Rute | |
| Skenario Alternatif | 1. Jika menekan tombol Batal, maka akan kembali ke <i>table</i> List Rute 2. Jika Nama Penerima tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "Nama Penerima tidak sesuai dengan aturan". 3. Jika Nomor HP tidak sesuai dengan aturan yang ditetapkan maka menampilkan pesan error "No. HP tidak sesuai dengan aturan". 4. Jika diantara Tanggal, Region, Nama Kurir, Jumlah Paket, Nomor Resi, Nama Penerima, Nomor HP, Alamat Penerima, atau Titik Rute tidak diisi, maka sistem akan menampilkan pesan error "Lengkapi Form dengan benar". | |
| Kondisi Akhir | Admin dapat mengubah rute pengantaran | |

Gambar 3. 17 Skenario *Use Case* Mengubah Rute Pengantaran

13) Menghapus Rute Pengantaran

Use case : Menghapus Rute Pengantaran
Aktor : Admin
Kondisi Awal : Admin memilih *sub-menu* List Rute pada halaman Rute Pengantaran
Kondisi Akhir : Admin dapat menghapus rute pengantaran

| | | |
|----------------------------|--|---|
| Usecase | Menghapus Rute Pengantaran | |
| Aktor | Admin | |
| Kondisi Awal | Admin memilih sub-menu List Rute pada halaman Rute Pengantaran | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan <i>table</i> data List Rute |
| | 2. Admin memilih tombol Edit pada data yang ingin diubah | |
| | | 3. Menampilkan pesan konfirmasi Hapus Data |
| | 4. Admin memilih tombol Hapus | |
| | | 5. Melakukan <i>update</i> pada <i>database</i> |
| | | 6. Menampilkan <i>table</i> data List Rute |
| Skenario Alternatif | 1. Jika memilih tombol Batal, maka rute pengantaran tidak akan dihapus | |
| Kondisi Akhir | Admin dapat menghapus rute pengantaran | |

Gambar 3. 18 Skenario *Use Case* Menghapus Rute Pengantaran

14) Melakukan *Export* Laporan Pengantaran

Use case : Admin memilih *sub-menu* List Rute pada halaman Rute Pengantaran

Aktor : Admin

Kondisi Awal : Admin memilih *sub-menu* List Rute pada halaman Rute Pengantaran

Kondisi Akhir : Admin berhasil melakukan *export* laporan pengantaran

| | | |
|----------------------------|---|--|
| Usecase | Melakukan <i>Export</i> Laporan Pengantaran | |
| Aktor | Admin | |
| Kondisi Awal | Admin memilih sub-menu List Rute pada halaman Rute Pengantaran | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | | 1. Menampilkan <i>table</i> data List Rute |
| | 2. Admin memilih tombol <i>Export</i> pada data yang ingin diexport | |
| | | 3. Menampilkan pilihan format export |
| | 4. Memilih format export | |
| | | 5. Melakukan export |
| Skenario Alternatif | - | |
| Kondisi Akhir | Admin berhasil melakukan <i>export</i> laporan pengantaran | |

Gambar 3. 19 Skenario *Use Case* Melakukan *Export* Laporan Pengantaran

15) Melakukan *Login* Kurir

Use case : Melakukan *login*
 Aktor : Kurir
 Kondisi Awal : Kurir sudah memiliki akun
 Kondisi Akhir : Kurir berhasil *login* dan masuk ke dalam sistem

| | | |
|----------------------------|---|--|
| Usecase | Melakukan <i>Login</i> | |
| Aktor | Kurir | |
| Kondisi Awal | Kurir sudah memiliki akun | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. Kurir mengisi formulir <i>login</i> dengan <i>username</i> dan <i>password</i> yang telah di daftarkan | 2. Mengecek kesesuaian akun |
| | | 3. Menampilkan halaman <i>homepage</i> |
| Skenario Alternatif | 1. Jika <i>username</i> tidak terdaftar, maka sistem akan menampilkan pesan <i>error</i> " <i>Username</i> tidak terdaftar". 2. Jika <i>username</i> dan <i>password</i> tidak diisi, maka sistem akan menampilkan pesan <i>error</i> " <i>Isi form</i> dengan benar". 3. Jika <i>password</i> yang dimasukkan salah, maka sistem akan menampilkan pesan <i>error</i> " <i>Password</i> yang dimasukkan salah". | |
| Kondisi Akhir | Kurir berhasil <i>login</i> dan masuk ke dalam sistem | |

Gambar 3. 20 Skenario *Use Case* Melakukan *Login*

16) Melihat Rute Pengantaran

Use case : Melihat Rute Pengantaran
 Aktor : Kurir
 Kondisi Awal : Kurir berada pada halaman *Homepage*
 Kondisi Akhir : Kurir dapat melihat Rute Pengantaran Terpendek

| | | |
|----------------------------|--|---|
| Usecase | Melihat Rute Pengantaran | |
| Aktor | Kurir | |
| Kondisi Awal | Kurir berada pada halaman <i>Homepage</i> | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. Kurir menekan tombol Rute Pengantaran | 2. Menampilkan halaman Rute Pengantaran |
| | 3. Kurir menekan tombol Lihat Rute | 4. Menampilkan Rute Pengantaran Terpendek |
| Skenario Alternatif | - | |
| Kondisi Akhir | Kurir dapat melihat Rute Pengantaran Terpendek | |

Gambar 3. 21 Skenario *Use Case* Melihat Rute Pengantaran

- 17) Melakukan Konfirmasi Setelah Pengantaran Paket
Use case : Melakukan Konfirmasi Setelah Pengantaran Paket
 Aktor : Kurir
 Kondisi Awal : Kurir berada pada halaman Rute Pengantaran
 Kondisi Akhir : Kurir dapat melakukan konfirmasi pengantaran paket

| | | |
|----------------------------|---|--|
| Usecase | Melakukan Konfirmasi Setelah Pengantaran Paket | |
| Aktor | Kurir | |
| Kondisi Awal | Kurir berada pada halaman Rute Pengantaran | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. Kurir menekan tombol Status Pengantaran | |
| | | 2. Menampilkan halaman Status Pengantaran |
| | 3. Kurir mengisi data berupa Bukti Pengantaran dan Keterangan | |
| | 4. Kurir menekan tombol Selesaikan Pengantaran | |
| | | 5. Memvalidasi dan menyimpan informasi yang dimasukkan |
| | | 6. Menampilkan halaman Rute Pengantaran |
| Skenario Alternatif | 1. Jika penerima tidak dapat menerima barang, maka Kurir menekan tombol Tunda Pengantaran | |
| Kondisi Akhir | Kurir dapat melakukan konfirmasi pengantaran paket | |

Gambar 3. 22 Skenario *Use Case* Melakukan Konfirmasi Setelah Pengantaran Paket

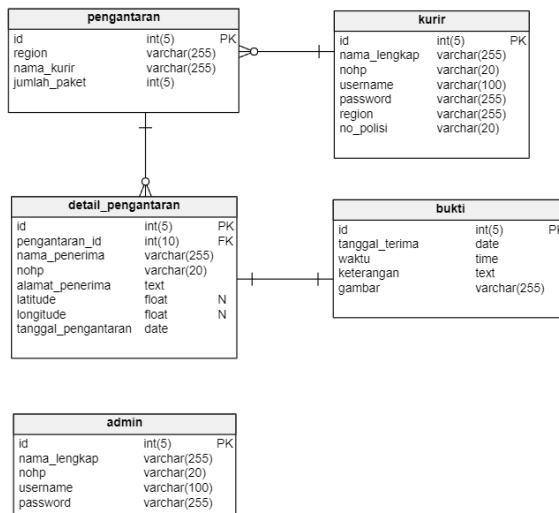
- 18) Melihat Riwayat Pengantaran
Use case : Melihat Riwayat Pengantaran
 Aktor : Kurir
 Kondisi Awal : Kurir berada pada halaman Rute Pengantaran
 Kondisi Akhir : Kurir dapat melihat Riwayat Pengantaran

| | | |
|----------------------------|--|---|
| Usecase | Melihat Riwayat Pengantaran | |
| Aktor | Kurir | |
| Kondisi Awal | Kurir berada pada halaman Rute Pengantaran | |
| Skenario Utama | Aksi Aktor | Reaksi Sistem |
| | 1. Kurir menekan tombol Riwayat Pengantaran | 2. Menampilkan halaman Riwayat Pengantaran |
| | 3. Kurir memilih Tanggal Pengantaran | |
| | 4. Kurir menekan tombol Cari | |
| | | 5. Menampilkan Riwayat Pengantaran tanggal di pilih |
| Skenario Alternatif | 1. Jika kurir tidak menetapkan tanggal pengantaran, maka akan ditampilkan Riwayat Pengantaran hari ini | |
| Kondisi Akhir | Kurir dapat melihat Riwayat Pengantaran | |

Gambar 3. 23 Skenario *Use Case* Melihat Riwayat Pengantaran

3.7 Entity Relationship Diagram

Entity Relationship Diagram (ERD) dirancang sesuai dengan kebutuhan *website* yang akan dikembangkan. Berikut tampilan *ERD* dalam *website*:



Gambar 3. 24 *Entity Relationship Diagram (ERD)*

3.8 Tabel Database

Tabel adalah objek penting untuk menyimpan semua data dan informasi dengan tujuan untuk memberikan kemudahan dalam membaca data karena tersusun secara sistematis sehingga memudahkan pemahaman. Berikut tabel *database* dalam perancangan *website* :

1. Tabel Admin

Tabel 3. 7 Admin

| <i>ID Field</i> | <i>Type</i> | <i>Length</i> | <i>Keterangan</i> |
|-----------------|-------------|---------------|--------------------|
| id | Int | 5 | <i>Primary Key</i> |
| nama_lengkap | Varchar | 255 | |
| nohp | Varchar | 20 | |
| username | Varchar | 100 | |
| password | Varchar | 255 | |

2. Tabel Kurir

Tabel 3. 8 Kurir

| <i>ID Field</i> | <i>Type</i> | <i>Length</i> | <i>Keterangan</i> |
|-----------------|-------------|---------------|--------------------|
| id | Int | 5 | <i>Primary Key</i> |
| nama_lengkap | Varchar | 255 | |
| nohp | Varchar | 20 | |
| username | Varchar | 100 | |
| password | Varchar | 255 | |
| region | Varchar | 255 | |
| no polisi | Varchar | 20 | |

3. Tabel Pengantaran

Tabel 3. 9 Pengantaran

| <i>ID Field</i> | <i>Type</i> | <i>Length</i> | <i>Keterangan</i> |
|-----------------|-------------|---------------|--------------------|
| id | Int | 5 | <i>Primary Key</i> |
| region | Varchar | 255 | |
| nama kurir | Varchar | 255 | |
| jumlah paket | Int | 5 | |

4. Tabel Detail Pengantaran

Tabel 3. 10 Detail Pengantaran

| <i>ID Field</i> | <i>Type</i> | <i>Length</i> | <i>Keterangan</i> |
|---------------------|-------------|---------------|--------------------|
| id | Varchar | 5 | <i>Primary Key</i> |
| pengantaran_id | Int | 10 | <i>Foreign Key</i> |
| nama penerima | Varchar | 255 | |
| nohp | Varchar | 20 | |
| alamat penerima | Varchar | 255 | |
| longitude | Float | | <i>Nullable</i> |
| latitude | Float | | <i>Nullable</i> |
| tanggal pengantaran | Timestamp | | |

5. Tabel Bukti

Tabel 3. 11 Bukti

| <i>ID Field</i> | <i>Type</i> | <i>Length</i> | <i>Keterangan</i> |
|-----------------|-------------|---------------|--------------------|
| id | Varchar | 5 | <i>Primary Key</i> |
| tanggal terima | Date | | |
| waktu | Time | | |
| keterangan | Text | 20 | |
| gambar | Varchar | 255 | |

3.9 Pengujian *Usability Testing*

Pengujian *usability* dilakukan untuk menentukan sejauh mana kepuasan pengguna terhadap sistem yang dibangun. Pada penelitian ini digunakan metode kuisisioner multidimensional, yaitu kuisisioner yang digunakan untuk mengukur berbagai aspek atau dimensi dari kegunaan suatu sistem. Aspek yang diambil yaitu efisiensi (*efficiency*), kemudahan belajar (*learnability*), kemudahan mengingat (*memorability*), kepuasan (*satisfaction*). Pada kuisisioner ini dibagi menjadi 2 aktor yaitu terhadap *admin* sebanyak 15 butir pertanyaan, dan aktor kurir sebanyak 14 butir pertanyaan dimana *admin* akan menguji sistem di *website* dan kurir akan menguji sistem di *mobile*. Untuk mengukur perhitungan dari pengujian digunakan skala likert. Skala likert merupakan jenis skala penilaian yang

digunakan untuk mengukur sikap, pendapat, atau persepsi seseorang terhadap suatu pernyataan.

Tabel 3. 12 Skala Likert

| Hasil Pengujian | Keterangan | Bobot |
|-----------------|---------------------|-------|
| SS | Sangat Setuju | 5 |
| S | Setuju | 4 |
| N | Netral | 3 |
| TS | Tidak Setuju | 2 |
| STS | Sangat Tidak Setuju | 1 |

Tabel 3. 13 Tabel Rancangan Pengujian *Usability* Kurir

| No | Butir Kriteria | Skala | | | | |
|---------------------|---|-----------|----------|----------|-----------|------------|
| | | SS (5) | S (4) | N (3) | TS (2) | STS (1) |
| EFFICIENCY | | | | | | |
| 1 | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | | | | | |
| 2 | Sistem ini memudahkan saya dalam menemukan alamat pengantaran | | | | | |
| 3 | Sistem ini bekerja sesuai dengan yang saya harapkan | | | | | |
| 4 | Sistem ini menampilkan informasi urutan alamat yang tepat | | | | | |
| LEARNABILITY | | | | | | |
| 5 | Sistem ini mudah digunakan | | | | | |
| 6 | Saya tidak kesulitan menggunakan sistem ini | | | | | |

| | | | | | | |
|---------------------|--|--|--|--|--|--|
| 7 | Saya dapat menggunakan sistem ini tanpa instruksi | | | | | |
| 8 | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | | | | | |
| MEMORABILITY | | | | | | |
| 9 | Sistem ini mudah dipahami | | | | | |
| 10 | Saya belajar dengan cepat dalam menggunakan sistem ini | | | | | |
| 11 | Saya mudah mengingat bagaimana cara menggunakan sistem ini | | | | | |
| 12 | Sistem ini mudah untuk dipelajari cara penggunaannya | | | | | |
| SATISFACTION | | | | | | |
| 13 | Saya puas dengan sistem ini | | | | | |
| 14 | Sistem ini menyenangkan dan nyaman untuk digunakan | | | | | |
| 15 | Saya akan tetap menggunakan sistem ini untuk kedepannya | | | | | |

Selanjutnya dilakukan perhitungan skala likert :

- a. Melakukan pencarian total bobot (TB) setiap kriteria dimana :

$$TB = (\text{total jumlah responden}) \times (\text{nilai tingkat kepuasan})$$

| No | Jumlah Nilai | | | | | Total Nilai |
|----|--------------|----------|----------|-----------|------------|-------------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | |
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |

| | | | | | | |
|----|--|--|--|--|--|--|
| 5 | | | | | | |
| 6 | | | | | | |
| 7 | | | | | | |
| 8 | | | | | | |
| 9 | | | | | | |
| 10 | | | | | | |
| 11 | | | | | | |
| 12 | | | | | | |
| 13 | | | | | | |
| 14 | | | | | | |
| 15 | | | | | | |

b. Menentukan jumlah skor indeks X dan Y

| Indeks | Keterangan | Jumlah skor tertinggi | Jumlah skor terendah |
|--------|---|-----------------------|----------------------|
| X | (skor likert terendah) x (total responden) | | |
| Y | (skor likert tertinggi) x (total responden) | | |

c. Menentukan nilai interval dari 0% - 100%

Interval = 100 / jumlah likert

Interval = 100/5 = 20

d. Menentukan interval skor interpretasi berdasarkan nilai interval pada poin 3.

| Interval | Skor interpretasi |
|-----------------|--------------------------|
| 0% - 19,99% | Sangat Tidak Baik |
| 20% - 39,99% | Tidak Baik |
| 40% - 59,99% | Normal |
| 60% - 79,99% | Baik |
| 80% - 100% | Sangat Baik |

e. Mencari indeks% untuk setiap pernyataan dengan rumus :

$$\text{Indeks\%} = \text{TB} / \text{Y} \times 100$$

| No | Jumlah Nilai | | | | | Total Bobot | Skor Max | Indeks (%) | Kategori |
|----|--------------|-------|-------|--------|---------|-------------|----------|------------|----------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | | | | |
| 1 | | | | | | | | | |
| 2 | | | | | | | | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| 6 | | | | | | | | | |
| 7 | | | | | | | | | |
| 8 | | | | | | | | | |
| 9 | | | | | | | | | |
| 10 | | | | | | | | | |
| 11 | | | | | | | | | |
| 12 | | | | | | | | | |
| 13 | | | | | | | | | |
| 14 | | | | | | | | | |
| 15 | | | | | | | | | |

| No. | Butir Kriteria | Persentase | Keterangan |
|---------------------|---|------------|------------|
| EFFICIENCY | | | |
| 1. | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | | |
| 2. | Sistem ini memudahkan saya dalam menemukan alamat pengantaran | | |
| 3. | Sistem ini bekerja sesuai dengan yang saya harapkan | | |
| 4. | Sistem ini menampilkan informasi urutan alamat yang tepat | | |
| LEARNABILITY | | | |
| 5. | Sistem ini mudah digunakan | | |
| 6. | Saya tidak kesulitan menggunakan sistem ini | | |
| 7. | Saya dapat menggunakan sistem ini tanpa instruksi | | |
| 8. | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | | |

| MEMORABILITY | | | |
|---------------------|--|--|--|
| 9. | Sistem ini mudah dipahami | | |
| 10. | Saya belajar dengan cepat dalam menggunakan sistem ini | | |
| 11. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | | |
| 12. | Sistem ini mudah untuk dipelajari cara penggunaannya | | |
| SATISFACTION | | | |
| 13. | Saya puas dengan sistem ini | | |
| 14. | Sistem ini menyenangkan dan nyaman untuk digunakan | | |
| 15. | Saya akan tetap menggunakan sistem ini untuk kedepannya | | |
| | Rata-rata Indeks | | |

Berdasarkan rekapitulasi hasil persentase diatas, dapat dihitung persentase keberhasilan dari masing-masing butir kriteria dengan rata-rata sebagai berikut :

1. Butir kriteria *Efficiency*
=Total Persentase / 4
2. Butir kriteria *Leanability*
=Total Persentase / 4

3. Butir kriteria *Memorability*
=Total Persentase / 4
4. Butir kriteria *Satisfaction*
=Total Persentase / 4

Tabel 3. 14 Tabel Rancangan *Usability Admin*

| No. | Butir Kriteria | SS | S | N | TS | STS |
|---------------------|--|----|---|---|----|-----|
| EFFICIENCY | | | | | | |
| 1. | Sistem ini memudahkan saya dalam menginputkan alamat | | | | | |
| 2. | Sistem ini bekerja sesuai yang saya harapkan | | | | | |
| 3. | Sistem ini menampilkan informasi urutan alamat yang tepat | | | | | |
| LEARNABILITY | | | | | | |
| 4. | Sistem ini mudah digunakan | | | | | |
| 5. | Saya tidak kesulitan menggunakan sistem ini | | | | | |
| 6. | Saya dapat menggunakan sistem ini tanpa instruksi | | | | | |
| 7. | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | | | | | |
| MEMORABILITY | | | | | | |
| 8. | Sistem ini mudah dipahami | | | | | |
| 9. | Saya belajar dengan cepat dalam menggunakan sistem ini | | | | | |

| | | | | | | |
|---------------------|--|--|--|--|--|--|
| 10. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | | | | | |
| 11. | Sistem ini mudah untuk dipelajari cara penggunaannya | | | | | |
| SATISFACTION | | | | | | |
| 12. | Saya puas dengan sistem ini | | | | | |
| 13. | Sistem ini menyenangkan dan nyaman untuk digunakan | | | | | |
| 14. | Saya akan tetap menggunakan sistem ini untuk kedepannya | | | | | |

Selanjutnya dilakukan perhitungan skala likert :

- a. Melakukan pencarian total bobot (TB) setiap kriteria dimana :

$$TB = (\text{total jumlah responden}) \times (\text{nilai tingkat kepuasan})$$

| No | Jumlah Nilai | | | | | Total Nilai |
|----|--------------|----------|----------|-----------|------------|-------------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | |
| 1 | | | | | | |
| 2 | | | | | | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |
| 7 | | | | | | |
| 8 | | | | | | |
| 9 | | | | | | |
| 10 | | | | | | |

| | | | | | | |
|----|--|--|--|--|--|--|
| 11 | | | | | | |
| 12 | | | | | | |
| 13 | | | | | | |
| 14 | | | | | | |

b. Menentukan jumlah skor indeks X dan Y

| Indeks | Keterangan | Jumlah skor tertinggi | Jumlah skor terendah |
|--------|---|-----------------------|----------------------|
| X | (skor likert terendah) x (total responden) | | |
| Y | (skor likert tertinggi) x (total responden) | | |

c. Menentukan nilai interval dari 0% - 100%

Interval = 100 / jumlah likert

Interval = $100/5 = 20$

d. Menentukan interval skor interpretasi berdasarkan nilai interval pada poin 3.

| Interval | Skor interpretasi |
|--------------|-------------------|
| 0% - 19,99% | Sangat Tidak Baik |
| 20% - 39,99% | Tidak Baik |
| 40% - 59,99% | Normal |
| 60% - 79,99% | Baik |
| 80% - 100% | Sangat Baik |

e. Mencari indeks% untuk setiap pernyataan dengan rumus :

$$\text{Indeks\%} = \text{TB} / \text{Y} \times 100$$

| No | Jumlah Nilai | | | | | Total Bobot | Skor Max | Indeks (%) | Kategori |
|----|--------------|-------|-------|--------|---------|-------------|----------|------------|----------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | | | | |
| 1 | | | | | | | | | |
| 2 | | | | | | | | | |
| 3 | | | | | | | | | |
| 4 | | | | | | | | | |
| 5 | | | | | | | | | |
| 6 | | | | | | | | | |
| 7 | | | | | | | | | |
| 8 | | | | | | | | | |
| 9 | | | | | | | | | |
| 10 | | | | | | | | | |
| 11 | | | | | | | | | |
| 12 | | | | | | | | | |
| 13 | | | | | | | | | |
| 14 | | | | | | | | | |

| No. | Butir Kriteria | Persentase | Keterangan |
|---------------------|--|------------|------------|
| EFFICIENCY | | | |
| 1. | Sistem ini memudahkan saya dalam menginputkan alamat | | |
| 2. | Sistem ini bekerja sesuai yang saya harapkan | | |
| 3. | Sistem ini menampilkan informasi urutan alamat yang tepat | | |
| LEARNABILITY | | | |
| 4. | Sistem ini mudah digunakan | | |
| 5. | Saya tidak kesulitan menggunakan sistem ini | | |
| 6. | Saya dapat menggunakan sistem ini tanpa instruksi | | |
| 7. | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | | |
| MEMORABILITY | | | |
| 8. | Sistem ini mudah dipahami | | |
| 9. | Saya belajar dengan cepat dalam | | |

| | | | |
|---------------------|--|--|--|
| | menggunakan sistem ini | | |
| 10. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | | |
| 11. | Sistem ini mudah untuk dipelajari cara penggunaannya | | |
| SATISFACTION | | | |
| 12. | Saya puas dengan sistem ini | | |
| 13. | Sistem ini menyenangkan dan nyaman untuk digunakan | | |
| 14. | Saya akan tetap menggunakan sistem ini untuk kedepannya | | |
| | Rata-rata Indeks | | |

Berdasarkan rekapitulasi hasil persentase diatas, dapat dihitung persentase keberhasilan dari masing-masing butir kriteria dengan rata-rata sebagai berikut :

1. Butir kriteria *Efficiency*
=Total Persentase / 3
2. Butir kriteria *Leanability*
=Total Persentase / 4
3. Butir kriteria *Memorability*
=Total Persentase / 4
4. Butir kriteria *Satisfaction*
=Total Persentase / 3

BAB IV

PENGUJIAN DAN ANALISIS

4.1 Implementasi Kode Program

Sistem dibangun dengan menggunakan formula haversine untuk menghitung jarak antara 2 titik kemudian untuk penentuan rute terpendek menggunakan algoritma djikstra. Berikut kode program penerapan dari formula haversine dan algoritma djikstra.

```
13 public function optimizeRoute($kurirId)
14 {
15     $startlat = $this->request->get('start_lat');
16     $startlon = $this->request->get('start_lon');
17
18     if (!$startlat || !$startlon) {
19         return $this->fail('start latitude and longitude are required', 400);
20     }
21
22     $pengantaranModel = new PengantaranModel();
23     $detailPengantaranModel = new DetailPengantaranModel();
24
25     $pengantaran = $detailPengantaranModel->getPendingDetailsByKurir($kurirId);
26
27     if (empty($pengantaran)) {
28         return $this->fail('No pending pengantaran found for kurir ID: ' . $kurirId);
29     }
30
31     // Tambahkan posisi awal kurir ke daftar koordinat
32     $coordinates = [
33         [
34             'id' => 'start',
35             'lat' => $startlat,
36             'lon' => $startlon,
37             'nama_penerima' => 'Kurir Start Position',
38             'alamat_penerima' => 'Current Location'
39         ]
40     ];
41
42     // Tambahkan koordinat dari detail pengantaran dengan status pending
43     foreach ($pengantaran as $item) {
44         $coordinates[] = [
45             'id' => $item['id'],
46             'lat' => $item['latitude'],
47             'lon' => $item['longitude'],
48             'nama_penerima' => $item['nama_penerima'],
49             'alamat_penerima' => $item['alamat_penerima']
50         ];
51     }
52 }
```

Gambar 4. 1 Metode optimizeRoute

```
51 }
52
53 // Hitung jarak menggunakan Formula Haversine
54 $distances = $this->calculateDistances($coordinates);
55
56 // Urutkan lokasi berdasarkan jarak dari posisi awal
57 $sortedLocations = $this->sortByLocationByDistance($distances[0]);
58
59 // Terapkan algoritma Dijkstra untuk menentukan rute optimal
60 $optimalRoute = $this->diJikstra($distances);
61
62 // Simpan hasil rute optimal
63 $optimalResult = [];
64 foreach ($optimalRoute as $index) {
65     $optimalResult[] = $coordinates[$index];
66 }
67
68 // Simpan hasil lokasi terdekat
69 $nearestResult = [];
70 foreach ($sortedLocations as $index) {
71     $nearestResult[] = $coordinates[$index];
72 }
73
74 // Hitung total jarak untuk rute optimal
75 $totalDistance = $this->calculateTotalDistance($optimalRoute, $distances);
76
77 return $this->respond([
78     'nearest_locations' => $nearestResult,
79     'optimal_route' => $optimalResult,
80     'total_distance' => $totalDistance
81 ]);
82 }
```

Gambar 4. 2 Lanjutan Metode optimizeRoute

Metode optimizeRoute mengambil parameter dari kurirId dan melakukan pengoptimalan rute. Dimana dilakukan pengambilan titik latitude dan longitude awal dari user. Kemudian diambil data pengantaran

yang memiliki status pending pada kurir yang dipilih melalui metode `getPendingDetailsByKurir`.

```
21 // Metode untuk mendapatkan detail pengantaran dengan status pending
22 public function getPendingDetailsByKurir($kurirId)
23 {
24     return $this->select('detail_pengantaran.*')
25         ->join('pengantaran', 'pengantaran.id = detail_pengantaran.pengantaran_id')
26         ->where('pengantaran.kurir_id', $kurirId)
27         ->where('detail_pengantaran.status', 'pending')
28         ->findAll();
29 }
30 }
```

Gambar 4. 3 Metode `getPendingDetailsByKurir`

Kemudian dilakukan penyusunan koordinat dimana berisi koordinat posisi awal kurir serta semua lokasi pengantaran yang berstatus pending. Jika sudah dilakukan perhitungan jarak semua titik koordinat dengan metode `calculateDistances`.

```
89 private function calculateDistances($coordinates)
90 {
91     $distances = [];
92     $n = count($coordinates);
93
94     for ($i = 0; $i < $n; $i++) {
95         for ($j = 0; $j < $n; $j++) {
96             if ($i != $j) {
97                 $distances[$i][$j] = $this->haversine(
98                     $coordinates[$i]['lat'],
99                     $coordinates[$i]['lon'],
100                    $coordinates[$j]['lat'],
101                    $coordinates[$j]['lon']
102                );
103             } else {
104                 $distances[$i][$j] = 0;
105             }
106         }
107     }
108
109     return $distances;
110 }
```

Gambar 4. 4 Metode `calculateDistances`

```

112     private function haversine($lat1, $lon1, $lat2, $lon2)
113     {
114         $earthRadius = 6371; // in kilometers
115
116         $dLat = deg2rad($lat2 - $lat1);
117         $dLon = deg2rad($lon2 - $lon1);
118
119         $a = sin($dLat / 2) * sin($dLat / 2) +
120             cos(deg2rad($lat1)) * cos(deg2rad($lat2)) *
121             sin($dLon / 2) * sin($dLon / 2);
122
123         $c = 2 * atan2(sqrt($a), sqrt(1 - $a));
124
125         return $earthRadius * $c;
126     }

```

Gambar 4. 5 Metode haversine

Didalam metode dari calculateDistances setelah dibuat array 2 dimensi yang bertujuan untuk menyimpan jarak antara semua titik koordinat, kemudian dilakukan perhitungan haversine. Pada metode haversine dengan memproses array yang sebelumnya dikirim dari calculateDistances maka didapat hasil jaraknya dengan mengalikan hasil dengan jari jari bumi untuk mendapatkan jarak dalam satuan kilometer.

Kemudian dilanjutkan dengan mengurutkan lokasi berdasarkan jarak dari posisi awal kurir dengan menjalankan metode sortLocationByDistance.

```

84     private function sortLocationsByDistance($distancesFromStart)
85     {
86         asort($distancesFromStart);
87         return array_keys($distancesFromStart);
88     }

```

Gambar 4. 6 Metode sortLocationsByDistance

Pada metode sortLocationsByDistance digunakan fungsi asort (mengurutkan elemen tanpa mengubah kunci) untuk mengurutkan array paling awal berdasarkan nilai jarak kemudian mengembalikan array dari kunci yang telah diurutkan.

Jika sudah selanjutnya, dilakukan penentuan rute terpendek djikstra dengan memproses distances yang ada.

```

128 private function dijkstra($graph)
129 {
130     $n = count($graph);
131     $visited = array_fill(0, $n, false);
132     $distance = array_fill(0, $n, PHP_INT_MAX);
133     $parent = array_fill(0, $n, -1);
134
135     $distance[0] = 0;
136
137     for ($i = 0; $i < $n - 1; $i++) {
138         $u = $this->minDistance($distance, $visited);
139         $visited[$u] = true;
140
141         for ($v = 0; $v < $n; $v++) {
142             if (
143                 !$visited[$v] && $graph[$u][$v] && $distance[$u] != PHP_INT_MAX
144                 && $distance[$u] + $graph[$u][$v] < $distance[$v]
145             ) {
146                 $distance[$v] = $distance[$u] + $graph[$u][$v];
147                 $parent[$v] = $u;
148             }
149         }
150     }
151     return $this->reconstructPath($parent);
152 }

```

Gambar 4. 7 Metode dijkstra

Pada metode dijkstra, dilakukan pembuatan array yang bertujuan untuk menandai node yang sudah dikunjungi, jarak minimum, serta parent dari setiap node. Kemudian dilakukan set dimana jarak node awal adalah posisi kurir yaitu 0. Dilakukan perulangan untuk menemukan node dengan jarak minimum yang belum dikunjungi. Pada perulangan ini dijalankan metode minDistance yang bertujuan untuk menemukan node dengan jarak minimum yang belum dikunjungi.

```

154 private function minDistance($distance, $visited)
155 {
156     $min = PHP_INT_MAX;
157     $minIndex = -1;
158
159     foreach ($distance as $i => $dist) {
160         if (!$visited[$i] && $dist <= $min) {
161             $min = $dist;
162             $minIndex = $i;
163         }
164     }
165
166     return $minIndex;
167 }

```

Gambar 4. 8 Metode minDistance

Pada minDistance dilakukan inisialisasi nilai minimum dan indeks minimum. Kemudian dilakukan perulangan untuk mengecek semua node

dan memperbarui nilai minimum jika ditemukan jarak yang lebih kecil lagi. Jika sudah maka akan dikembalikan ke minIndex.

Dilakukan update jarak minimum untuk semua node tetangga dari node yang dipilih, kemudian dilakukan rekonstruksi rute dengan menjalankan metode reconstructPath.

```
169     private function reconstructPath($parent)
170     {
171         $path = [];
172         $v = count($parent) - 1;
173
174         while ($v != -1) {
175             array_unshift($path, $v);
176             $v = $parent[$v];
177         }
178
179         return $path;
180     }
```

Gambar 4. 9 Metode reconstructPath

Pada metode ini, dilakukan pembuatan array path kosong. Kemudian dilakukan loop untuk menambahkan node ke path yang mengarah ke array parentnya. Jika sudah maka mengembalikan array path yang berisi urutan node dari awal ke lokasi tujuan.

Setelah sudah mendapatkan rute terpendeknya melalui metode djisktra, dilakukan perhitungan total jarak rute dari rute optimal yang didapat.

```
182     private function calculateTotalDistance($route, $distances)
183     {
184         $totalDistance = 0;
185         for ($i = 0; $i < count($route) - 1; $i++) {
186             $totalDistance += $distances[$route[$i]][$route[$i + 1]];
187         }
188         return $totalDistance;
189     }
190 }
```

Gambar 4. 10 Metode calculateTotalDistance

Jika sudah maka nilai akan dikembalikan melalui respon JSON yang berisi lokasi terdekat, rute optimal, dan total jarak rute optimal.

```

62 // Susun hasil rute optimal
63 $optimalResult = [];
64 foreach ($optimalRoute as $index) {
65     $optimalResult[] = $coordinates[$index];
66 }
67
68 // Susun hasil lokasi terdekat
69 $nearestResult = [];
70 foreach ($sortedLocations as $index) {
71     $nearestResult[] = $coordinates[$index];
72 }
73
74 // Hitung total jarak untuk rute optimal
75 $totalDistance = $this->calculateTotalDistance($optimalRoute, $distances);
76
77 return $this->respond([
78     'nearest_locations' => $nearestResult,
79     'optimal_route' => $optimalResult,
80     'total_distance' => $totalDistance
81 ]);
82

```

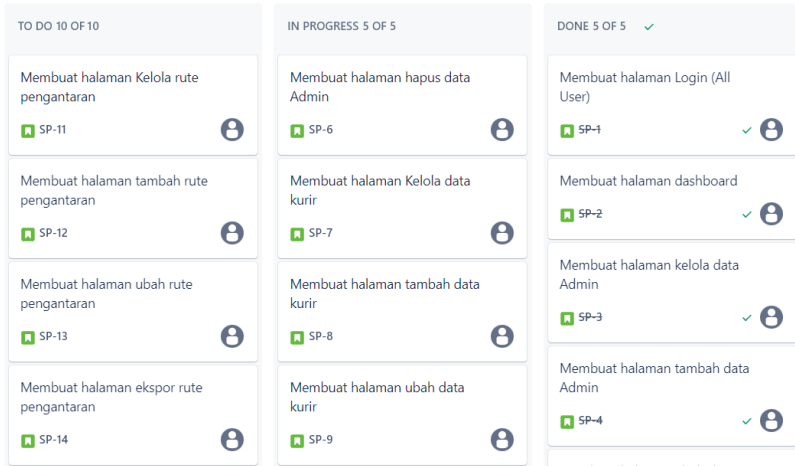
Gambar 4. 11 Mengembalikan hasil ke dalam respons JSON

4.2 Implementasi Metode Scrum

Pada penelitian ini, implementasi Scrum dilaksanakan dari tanggal 6 Mei 2024 hingga 17 Juli 2024. *Daily Scrum Meeting*. *Sprint planning* dilakukan pada awal sebelum *sprint* dimulai yang berguna untuk memilih *product backlog* mana yang nantinya akan di kerjakan.

Daily Scrum Meeting berisi pembahasan mengenai progress pengerjaan dan membahas kendala yang dihadapi saat mengerjakannya pada hari itu. Fase *sprint review* yaitu fase tim akan membahas seluruh *product backlog* yang telah diselesaikan selama masa *sprint* yang dilakukan pada akhir *sprint*. Setelah pembahasan *sprint review* dilakukan, maka akan dilanjutkan dengan pembahasan *sprint planning* untuk *sprint* selanjutnya.

Agar mempermudah kerja sama tim, dibuat *Scrum Board* menggunakan aplikasi Jira. Pada *Scrum Board* terbagi menjadi 3 status pengerjaan yaitu *To Do*, *In Progress* dan *Done*. Status *To Do* berisikan *product backlog* yang belum dikerjakan, *In Progress* berisikan *product backlog* yang sedang dikerjakan dan *Done* berisikan *product backlog* yang sudah selesai dikerjakan.



Gambar 4. 12 Scrum Board pada Aplikasi Jira

4.2.1 *Sprint* 1

4.2.1.1 *Sprint* Planning

Pada *Sprint* 1, tim berfokus pada pengerjaan *product backlog* yang telkah disepakati sebelumnya.

Tabel 4. 1 Perancangan *Sprint* 1

| <i>Sprint</i> | <i>Product Backlog</i> |
|--|--|
| <i>Sprint</i> 1 (6 Mei – 28 Mei) 23 hari | Membuat halaman <i>Login (All User)</i> |
| | Membuat halaman <i>dashboard</i> |
| | Membuat halaman <i>kelola data Admin</i> |
| | Membuat halaman <i>tambah data Admin</i> |
| | Membuat halaman <i>ubah data Admin</i> |
| | Membuat halaman <i>hapus data Admin</i> |
| | Membuat halaman <i>Kelola data kurir</i> |

| <i>Sprint</i> | <i>Product Backlog</i> |
|---------------|-----------------------------------|
| | Membuat halaman tambah data kurir |
| | Membuat halaman ubah data kurir |
| | Membuat halaman hapus data kurir |

4.2.1.2 Sprint

Sprint 1 berlangsung dari tanggal 6 Mei 2024 hingga 2024 dengan total durasi 15 hari. Adapun proses pengembangan pada *sprint 1* ini dapat dilihat pada Tabel 4.2 berikut:

Tabel 4. 2 Durasi Pengerjaan *Sprint 1*

| ID | Item Backlog | Pengerjaan |
|-----------|--|------------------------|
| PB01 | Membuat halaman Login (Admin) | 06 Mei - 08 Mei 2024 |
| PB01 | Membuat halaman Login (Kurir) | 04 Juni - 07 Juni 2024 |
| PB02 | Membuat halaman Dashboard | 09 Mei - 14 Mei 2024 |
| PB03 | Membuat halaman Kelola data <i>Admin</i> | 15 Mei 2024 |
| PB04 | Membuat halaman tambah data <i>Admin</i> | 15 Mei 2024 |
| PB05 | Membuat halaman ubah data <i>Admin</i> | 16 Mei 2024 |
| PB06 | Membuat halaman hapus data <i>Admin</i> | 16 Mei 2024 |
| PB07 | Membuat halaman Kelola data kurir | 17 Mei - 18 Mei 2024 |
| PB08 | Membuat halaman tambah data kurir | 18 Mei 2024 |
| PB09 | Membuat halaman ubah data kurir | 20 Mei 2024 |
| PB10 | Membuat halaman hapus data kurir | 20 Mei 2024 |

4.2.1.3 Sprint Review dan Sprint Retrospective

Pada *sprint review* yang dilakukan pada tanggal 21 Mei 2024, terdapat pertemuan antara tim developer dengan *product owner*. Hasil dari pertemuan ini adalah semua fitur yang dikerjakan sesuai dengan yang diharapkan. *Product owner* mengapresiasi pekerjaan ini dengan baik. Setelah itu dilanjutkan untuk melakukan *sprint retrospective* untuk membahas semua fitur yang berhasil diimplementasikan dalam *sprint* 1. Selama pengerjaan *sprint* 1 ini dapat berjalan dengan baik berkat kerjasama tim dan komunikasi.

4.2.2 *Sprint* 2

4.2.2.1 Sprint Planning

Pada *sprint* 2 ini memfokuskan untuk melanjutkan pengerjaan dari hasil *sprint* 1, sesuai dengan *product backlog* yang telah disepakati.

Tabel 4. 3 Perencanaan *Sprint* 2

| <i>Sprint</i> | <i>Product Backlog</i> |
|---------------|--|
| | Membuat halaman Kelola rute pengantaran |
| | Membuat halaman tambah rute pengantaran |
| | Membuat halaman ubah rute pengantaran |
| | Membuat halaman hapus rute pengantaran |
| | Membuat halaman kelola bukti pengantaran |

| | |
|---|---|
| <i>Sprint 2</i> (4 Juni – 20 Juli) 47 hari | Membuat halaman ubah dan hapus bukti pengantaran |
| | Membuat API untuk menghubungkan ke aplikasi <i>Mobile</i> |
| | Mendeploy sistem ke hosting |
| | Membuat halaman profil kurir |
| | Membuat halaman list rute pengantaran |
| | Membuat halaman lihat rute pengantaran |
| | Membuat halaman upload bukti pengantaran |
| | Membuat halaman riwayat pengantaran |

4.2.2.2 Sprint

Sprint 2 berlangsung dari tanggal 22 Mei 2024 hingga 8 Juli 2024 dengan total durasi 44 hari. Adapun proses pengembangan pada *sprint 2* ini dapat dilihat pada Tabel 4.4 berikut:

Tabel 4. 4 Durasi Pengerjaan *Sprint 2*

| ID | Item Backlog | Pengerjaan |
|------|---|-----------------------|
| PB11 | Membuat halaman Kelola rute pengantaran | 22 Mei - 23 Mei 2024 |
| PB12 | Membuat halaman tambah rute pengantaran | 23 Mei - 05 Juni 2024 |

| | | |
|------|---|------------------------|
| PB13 | Membuat halaman ubah rute pengantaran | 06 Juni - 07 Juni 2024 |
| PB14 | Membuat halaman hapus rute pengantaran | 07 Juni 2024 |
| PB15 | Membuat halaman kelola bukti pengantaran | 13 Juni 2024 |
| PB16 | Membuat halaman ubah dan hapus bukti pengantaran | 14 Juni 2024 |
| PB17 | Membuat API untuk menghubungkan ke aplikasi <i>Mobile</i> | 10 Juni - 10 Juli 2024 |
| PB18 | Mendeploy sistem ke hosting | 11 Juli - 17 Juli 2024 |
| PB19 | Membuat halaman profil kurir | 10 Juni - 13 Juni 2024 |
| PB20 | Membuat halaman list rute pengantaran | 14 Juni - 19 Juni 2024 |
| PB21 | Membuat halaman lihat rute pengantaran | 20 Juni - 25 Juni 2024 |
| PB22 | Membuat halaman upload bukti pengantaran | 26 Juni - 1 Juli 2024 |
| PB23 | Membuat halaman riwayat pengantaran | 2 Juli - 8 Juli 2024 |

4.2.2.3 *Sprint Review dan Sprint Retrospective*

Pada *sprint* kedua ini pengerjaan *product backlog* selesai dalam waktu 44 hari dan seluruh *backlog* diterima dengan baik tanpa ada revisi dari *product owner*.

4.3 Hasil Implementasi Perancangan

Aplikasi ini menggunakan *website* untuk *admin* dan *android* untuk kurir. Berikut merupakan hasil implementasi perancangan sistem.

4.3.1 Hasil Perancangan Admin (Website)

4.3.1.1 Antarmuka Halaman Login

Pada antarmuka halaman login terdapat form dimana admin akan mengisi username dan password dimana jika berhasil login maka akan mengarah ke halaman Dashboard.



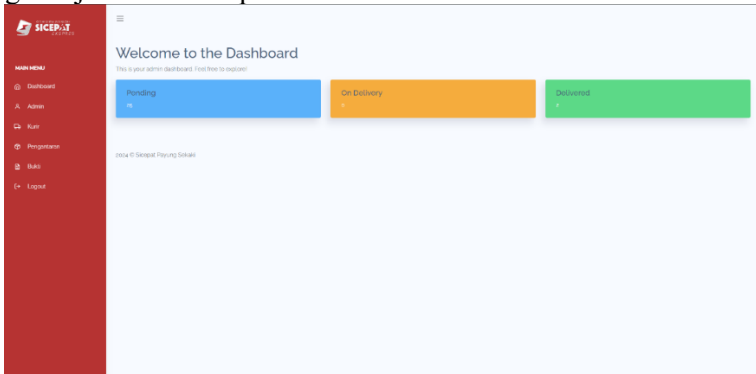
Gambar 4. 13 Halaman Login Admin

Terdapat beberapa elemen yang ada pada halaman login yaitu :

1. Form Input Username
Identitas dari admin yang akan login ke dalam aplikasi
2. Form Input Password
Kata sandi dari admin yang akan login ke dalam aplikasi
3. Button Submit
Setelah klik button submit maka username dan password akan dicocokkan, jika terdapat identitas dari yang telah diinputkan maka admin akan diarahkan ke halaman Dashboard

4.3.1.2 Antarmuka Halaman Home Admin

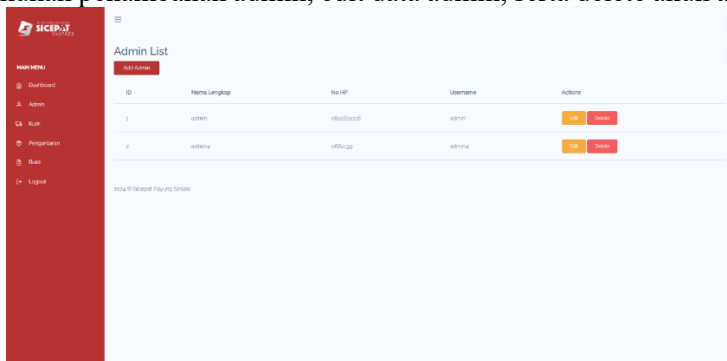
Pada antarmuka halaman home admin terdapat informasi mengenai jumlah status paket.



Gambar 4. 14 Halaman Home Admin

4.3.1.3 Halaman Data Admin

Pada antarmuka halaman menu admin terdapat informasi mengenai daftar admin dan pada halaman ini admin juga dapat melakukan penambahan admin, edit data admin, serta delete akun admin.



Gambar 4. 15 Halaman Menu Admin

4.3.1.4 Halaman Tambah Data Admin

Pada tambah admin, admin dapat melakukan tambah data admin baru, dimana data yang akan diisi yaitu nama lengkap, no hp, username, serta password.

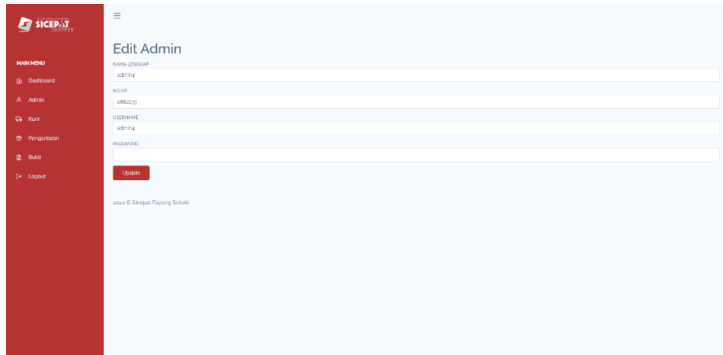
Gambar 4. 16 Halaman Tambah Admin

Terdapat beberapa elemen yang ada pada halaman tambah admin yaitu :

1. Form Input Nama Lengkap
Nama Lengkap dari admin
2. Form Input No. HP
No. HP dari admin
3. Form Input Username
Identitas dari admin yang akan login ke dalam aplikasi
4. Form Input Password
Kata sandi dari admin yang akan login ke dalam aplikasi
5. Button Save
Setelah klik button save maka informasi yang diinputkan pada form input akan tersimpan pada database

4.3.1.5 Halaman Edit Data Admin

Pada edit admin, admin dapat melakukan perubahan data pada akun admin yang telah di tambahkan sebelumnya, data yang dapat diubah berupa nama lengkap, no hp, username, serta password.



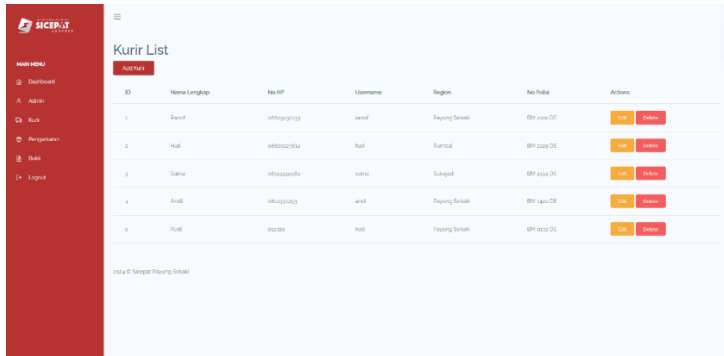
Gambar 4. 17 Halaman Edit Admin

Terdapat beberapa elemen yang ada pada halaman edit admin yaitu :

1. Form Input Nama Lengkap
Nama Lengkap dari admin
2. Form Input No. HP
No. HP dari admin
3. Form Input Username
Identitas dari admin yang akan login ke dalam aplikasi
4. Form Input Password
Kata sandi dari admin yang akan login ke dalam aplikasi
5. Button Update
Setelah klik button update maka informasi yang diubah pada form input akan berubah pada database

4.3.1.6 Halaman Data Kurir

Pada antarmuka halaman menu kurir terdapat informasi mengenai daftar kurir dan pada halaman ini admin juga dapat melakukan penambahan kurir, edit data kurir, serta delete akun kurir.



Gambar 4. 18 Halaman Menu Kurir

4.3.1.7 Halaman Tambah Data Kurir

Pada tambah kurir, admin dapat melakukan tambah data kurir baru, dimana data yang akan diisi yaitu nama lengkap, no hp, username, password, region, dan no polisi.



Gambar 4. 19 Halaman Tambah Kurir

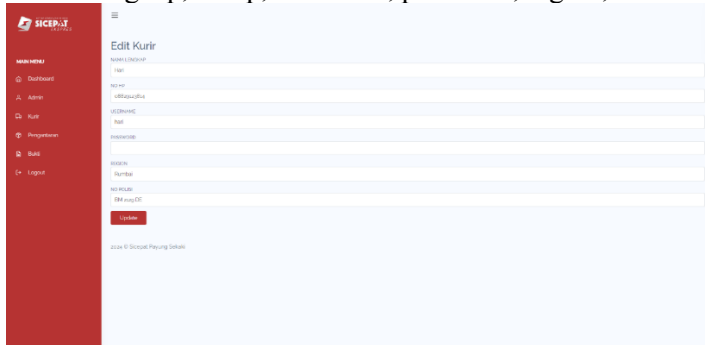
Terdapat beberapa elemen yang ada pada halaman tambah kurir yaitu :

1. Form Input Nama Lengkap
Nama Lengkap dari kurir
2. Form Input No. HP
No. HP dari kurir
3. Form Input Username
Identitas dari kurir yang akan login ke dalam aplikasi
4. Form Input Password
Kata sandi dari kurir yang akan login ke dalam aplikasi

5. Form Select Region
Region PIC dari kurir
6. Form Input No. Polisi
No. Polisi dari kurir
7. Button Save
Setelah klik button save maka informasi yang diinputkan pada form input akan tersimpan pada database

4.3.1.8 Halaman Edit Data Kurir

Pada edit kurir, admin dapat melakukan perubahan data pada akun kurir yang telah di tambahkan sebelumnya, data yang dapat diubah berupa nama lengkap, no hp, username, password, region, serta no polisi.



Gambar 4. 20 Halaman Edit Kurir

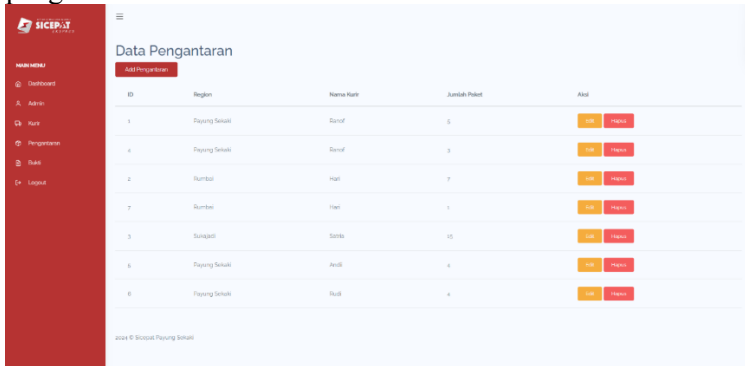
Terdapat beberapa elemen yang ada pada halaman edit kurir yaitu :

1. Form Input Nama Lengkap
Nama Lengkap dari kurir
2. Form Input No. HP
No. HP dari kurir
3. Form Input Username
Identitas dari kurir yang akan login ke dalam aplikasi
4. Form Input Password
Kata sandi dari kurir yang akan login ke dalam aplikasi
5. Form Select Region
Region PIC dari kurir
6. Form Input No. Polisi
No. Polisi dari kurir
7. Button Update

Setelah klik button update maka informasi yang diubah pada form input akan berubah pada database

4.3.1.9 Halaman Data Pengantaran

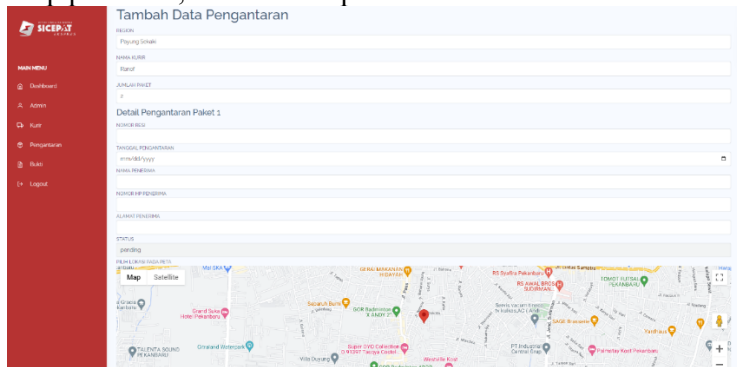
Pada antarmuka halaman data pengantaran terdapat informasi mengenai daftar pengantaran dan pada halaman ini admin juga dapat melakukan penambahan pengantaran, edit data pengantaran, serta delete data pengantaran.



Gambar 4. 21 Halaman Data Pengantaran

4.3.1.10 Halaman Tambah Data Pengantaran

Pada tambah data pengantaran, admin dapat melakukan tambah data pengantaran baru, dimana data yang akan diisi yaitu region, nama kurir, jumlah paket, nomor resi, tanggal pengantaran, nama penerima, nomor hp penerima, serta alamat penerima.



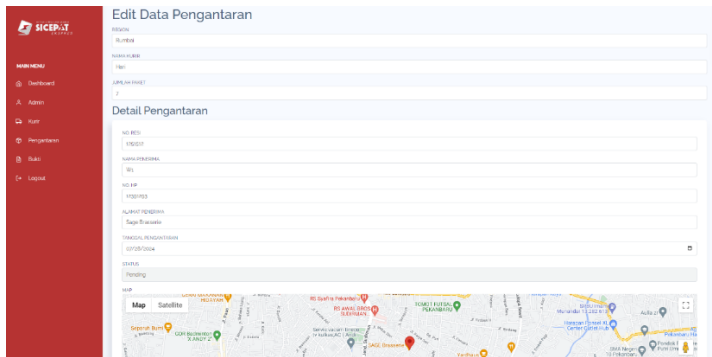
Gambar 4. 22 Halaman Tambah Data Pengantaran

Terdapat beberapa elemen yang ada pada halaman tambah data pengantaran yaitu :

1. Form Select Region
Region PIC dari kurir
2. Form Input Nama Kurir
Nama kurir yang ditugaskan untuk pengiriman paket
3. Form Input Jumlah Paket
Jumlah paket yang akan diantar oleh kurir, jika 2, maka akan muncul 2 form detail pengantaran paket.
4. Form Input No. Resi
No. Resi dari paket yang akan diantar
5. Form Input Tanggal Pengantaran
Tanggal pengantaran paket yang akan dilakukan kurir
6. Form Input Nama Penerima
Nama penerima paket
7. Form Input No. HP Penerima
No. HP penerima paket yang dapat dihubungi
8. Form Input Alamat Penerima
Alamat pengantaran penerima paket
9. Form Input Status
Status pengantaran paket, field ini tidak dapat diubah dan otomatis bernilai pending
10. Button Simpan
Setelah klik button simpan maka informasi yang diinputkan pada form input akan tersimpan pada database

4.3.1.11 Halaman Edit Data Pengantaran

Pada edit data pengantaran, admin dapat melakukan edit data pengantaran, dimana data yang akan dapat diedit yaitu region, nama kurir, jumlah paket, nomor resi, tanggal pengantaran, nama penerima, nomor hp penerima, serta alamat penerima.



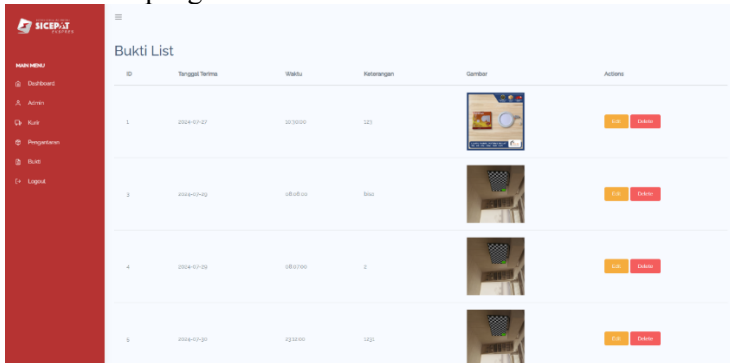
Gambar 4. 23 Halaman Edit Data Pengantaran





Terdapat beberapa elemen yang ada pada halaman edit data pengantaran yaitu :

1. Form Select Region
Region PIC dari kurir
2. Form Input Nama Kurir
Nama kurir yang ditugaskan untuk pengiriman paket
3. Form Input Jumlah Paket
Jumlah paket yang akan diantar oleh kurir, jika 2, maka akan muncul 2 form detail pengantaran paket.
4. Form Input No. Resi
No. Resi dari paket yang akan diantar
5. Form Input Tanggal Pengantaran
Tanggal pengantaran paket yang akan dilakukan kurir
6. Form Input Nama Penerima
Nama penerima paket
7. Form Input No. HP Penerima
No. HP penerima paket yang dapat dihubungi
8. Form Input Alamat Penerima
Alamat pengantaran penerima paket
9. Form Input Status
Status pengantaran paket, field ini tidak dapat diubah dan otomatis bernilai pending
10. Button Update Pengantaran
Setelah klik button update pengantaran maka informasi yang diubah pada form input akan berubah pada database

4.3.1.12 Halaman Data Bukti Pengantaran

Pada antarmuka halaman bukti pengantaran, terdapat informasi mengenai bukti pengantaran yang telah dilakukan oleh kurir pada halaman ini, admin dapat melakukan edit data bukti pengantaran serta delete data bukti pengantaran.

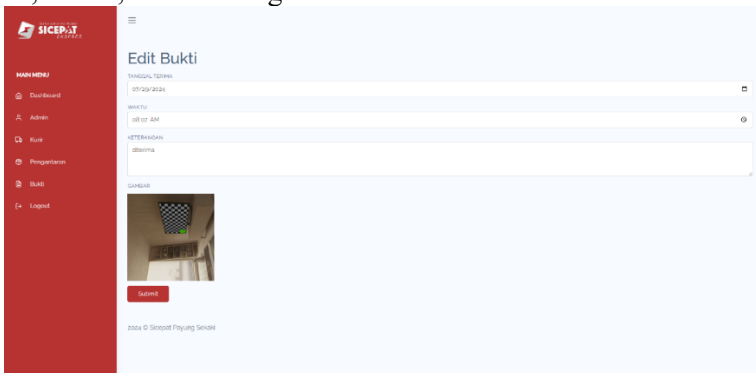


| ID | Tanggal Terima | Waktu | Keterangan | Gambar | Aksi |
|----|----------------|----------|------------|---|---|
| 1 | 2024-01-27 | 02:00:00 | 123 |  | Edit Delete |
| 2 | 2024-01-27 | 08:00:00 | 345 |  | Edit Delete |
| 4 | 2024-01-28 | 08:00:00 | 2 |  | Edit Delete |
| 5 | 2024-01-28 | 03:00:00 | 123 |  | Edit Delete |

Gambar 4. 24 Halaman Data Bukti Pengantaran

4.3.1.13 Halaman Edit Data Bukti Pengantaran

Pada edit data pengantaran, admin dapat melakukan edit data bukti pengantaran, dimana data yang akan dapat diedit yaitu tanggal terima, waktu, dan keterangan.




Edit Bukti

TANGGAL TERIMA: 2024-01-28

WAKTU: 08:00 AM

KETERANGAN: 123456789

GAMBAR: 

[Simpan](#)

2024 © Sicep@Plyang Sende

Gambar 4. 25 Halaman Edit Data Bukti Pengantaran

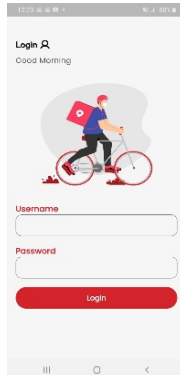
Terdapat beberapa elemen yang ada pada halaman edit data bukti pengantaran yaitu :

1. Form Input Tanggal Terima
Tanggal paket diterima oleh penerima
2. Form Input Waktu
Waktu paket diterima oleh penerima
3. Form Input Keterangan
Keterangan dari kurir mengenai paket yang diantar ke penerima
4. Gambar
Bukti pengantaran berhasil diterima oleh penerima
5. Button Submit
Setelah klik button submit maka informasi yang diubah pada form input akan berubah pada database

4.3.2 Hasil Perancangan Kurir (Android)

4.3.2.1 Antarmuka Halaman Login

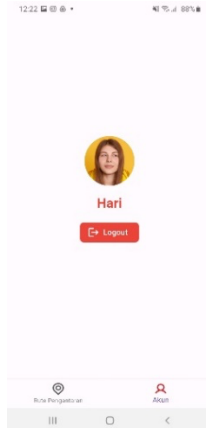
Halaman pertama adalah halaman Login Kurir. Kurir dapat login menggunakan *username* dan *password* yang telah diberikan sebelumnya.



Gambar 4. 26 Login Kurir

4.3.2.2 Halaman Akun Kurir

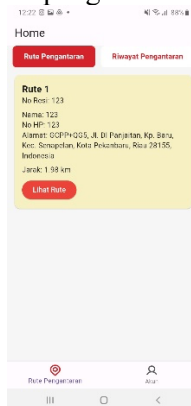
Setelah berhasil login, kurir dapat melihat akun melalui menu akun yang berada pada bagian bawah kanan.



Gambar 4. 27 Akun Kurir

4.3.2.3 Halaman List Rute Pengantaran

Untuk mempermudah kurir dalam melihat urutan paket yang akan diantar, terdapat List Rute Pengantaran, yang mana list yang ditampilkan merupakan urutan pengantaran paket.

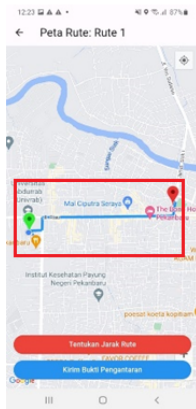


Gambar 4. 28 Rute Pengantaran

4.3.2.4 Halaman Melihat Rute Pengantaran

Pada halaman melihat rute pengantaran akan ditampilkan jalur terpendek yang akan dilalui oleh kurir menuju *customer*. Jika kurir

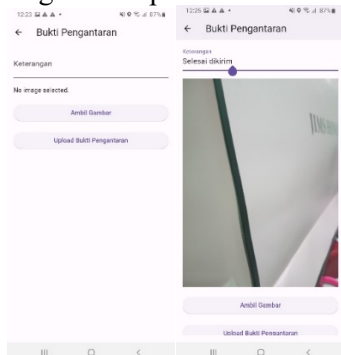
menekan tombol Tentukan Jarak Rute maka akan menampilkan jalur yang akan dilalui oleh kurir yang berwarna biru.



Gambar 4. 29 Penentuan Jarak Rute

4.3.2.5 Halaman Upload Bukti Pengantaran

Pada halaman upload bukti pengantaran, ketika kurir sudah sampai pada titik lokasi *customer* kurir dapat menekan tombol Kirim Bukti Pengantaran dan akan diarahkan ke halaman Upload Bukti Pengantaran, dimana kurir akan diminta untuk menyisipkan keterangan dan foto dokumentasi sebagai bukti paket telah diterima oleh *customer*.



Gambar 4. 30 Unggah Bukti Pengantaran

4.3.2.6 Halaman Riwayat Pengantaran

Pada halaman riwayat pengantaran, ketika kurir sudah melakukan upload bukti pengantaran dan berhasil, maka akan muncul di

riwayat pengantaran, halaman riwayat pengantaran berfungsi untuk menampilkan history pengantaran yang telah selesai diantar.



Gambar 4. 31 Riwayat Pengantaran

4.4 Pengujian

Pada bagian pengujian sistem diuji dengan menggunakan *Usability Testing*. Pengujian ini dilakukan guna untuk melihat dan memastikan apakah sistem yang telah dibangun telah memenuhi segala hal yang diharapkan sebelumnya. Pengujian juga dilakukan dengan tujuan melihat apakah sistem yang dibangun berjalan dengan benar.

4.4.1 Pengujian *Usability*

Usability Testing merupakan salah satu cara untuk mengetahui apakah pengguna dapat mudah dalam menggunakan aplikasi, seberapa efisien dan efektif sebuah aplikasi dapat membantu pengguna mencapai tujuannya dan apakah pengguna puas dengan aplikasi yang digunakan.

Hasil pengujian tersebut dilampirkan berupa kuisisioner yang diisi oleh pengguna aplikasi. Untuk mendapatkan tanggapan pengguna dilakukan dengan survey kepada 4 responden, yaitu 1 responden mewakili aktor *admin* dan 3 responden mewakili aktor kurir.

Tabel 4. 5 Hasil Kuisioner Kurir

| No. | Butir Kriteria | Persentase | Keterangan |
|---------------------|---|------------|-------------|
| EFFICIENCY | | | |
| 1. | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | 100% | Sangat Baik |
| 2. | Sistem ini memudahkan saya dalam menemukan alamat pengantaran | 86,7% | Sangat Baik |
| 3. | Sistem ini bekerja sesuai dengan yang saya harapkan | 93,3% | Sangat Baik |
| 4. | Sistem ini menampilkan informasi urutan alamat yang tepat | 100% | Sangat Baik |
| LEARNABILITY | | | |
| 5. | Sistem ini mudah digunakan | 100% | Sangat Baik |
| 6. | Saya tidak kesulitan menggunakan sistem ini | 93,3% | Sangat Baik |
| 7. | Saya dapat menggunakan sistem ini tanpa instruksi | 93,3% | Sangat Baik |
| 8. | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | 93,3% | Sangat Baik |
| MEMORABILITY | | | |
| 9. | Sistem ini mudah dipahami | 100% | Sangat Baik |
| 10. | Saya belajar dengan cepat dalam menggunakan sistem ini | 100% | Sangat Baik |
| 11. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | 100% | Sangat Baik |

| | | | |
|---------------------|---|-------|-------------|
| 12. | Sistem ini mudah untuk dipelajari cara penggunaannya | 93,3% | Sangat Baik |
| SATISFACTION | | | |
| 13. | Saya puas dengan sistem ini | 93,3% | Sangat Baik |
| 14. | Sistem ini menyenangkan dan nyaman untuk digunakan | 86,7% | Sangat Baik |
| 15. | Saya akan tetap menggunakan sistem ini untuk kedepannya | 93,3% | Sangat Baik |
| | Rata-rata Indeks | 94,8% | Sangat Baik |

Tabel 4. 6 Hasil Kuisioner Admin

| No. | Butir Kriteria | Persentase | Keterangan |
|---------------------|--|------------|-------------|
| EFFICIENCY | | | |
| 1. | Sistem ini memudahkan saya dalam menginputkan alamat | 100% | Sangat Baik |
| 2. | Sistem ini bekerja sesuai yang saya harapkan | 100% | Sangat Baik |
| 3. | Sistem ini menampilkan informasi urutan alamat yang tepat | 100% | Sangat Baik |
| LEARNABILITY | | | |
| 4. | Sistem ini mudah digunakan | 100% | Sangat Baik |
| 5. | Saya tidak kesulitan menggunakan sistem ini | 100% | Sangat Baik |
| 6. | Saya dapat menggunakan sistem ini tanpa instruksi | 100% | Sangat Baik |
| 7. | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | 100% | Sangat Baik |
| MEMORABILITY | | | |
| 8. | Sistem ini mudah dipahami | 100% | Sangat Baik |
| 9. | Saya belajar dengan cepat dalam menggunakan sistem ini | 100% | Sangat Baik |
| 10. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | 100% | Sangat Baik |
| 11. | Sistem ini mudah untuk dipelajari cara penggunaannya | 100% | Sangat Baik |
| SATISFACTION | | | |
| 12. | Saya puas dengan sistem ini | 80% | Sangat Baik |

| | | | |
|-----|---|--------|-------------|
| 13. | Sistem ini menyenangkan dan nyaman untuk digunakan | 100% | Sangat Baik |
| 14. | Saya akan tetap menggunakan sistem ini untuk kedepannya | 80% | Sangat Baik |
| | Rata-rata Indeks | 96,67% | Sangat Baik |

4.5 Analisis

Implementasi metode *Scrum* dalam Penerapan Metode *Scrum* pada Sistem yang dibangun dapat diselesaikan dalam waktu 2,5 bulan yaitu dari 6 Mei 2024 hingga 17 Juli 2024. Penggunaan metode *Scrum* sangat membantu dalam menghemat waktu pengerjaan dan memberikan kemudahan bagi peneliti untuk berkoordinasi dengan pihak PT. SiCepat Payung Sekaki dari proses awal hingga proses akhir yaitu pengujian sistem.

Fitur yang ada pada sistem yang dibangun dapat berjalan dengan baik sesuai dengan fungsi yang diharapkan oleh pengguna. Hal ini dapat dibuktikan dari hasil pengujian yang dilakukan yaitu *Usability Testing*. Pengujian ini digunakan untuk mengamati bagaimana pengguna berinteraksi terhadap sistem yang dibangun dimulai dari aspek efisiensi (*efficiency*), aspek kemudahan belajar (*learnability*), aspek kemudahan mengingat (*memorability*), aspek kepuasan (*satisfaction*).

Berdasarkan hasil pengujian *Usability* terhadap 3 kurir dan 1 *admin* yang nantinya akan menggunakan sistem ini, didapatkan hasil bahwa sistem yang dibangun berjalan dengan sangat baik, baik dari sisi kurir maupun *admin*. Hal ini dapat dilihat berdasarkan aspek pengujian yang telah dilakukan dimana pada sisi kurir : *Efficiency* (95%), *Learnability* (94,9%), *Memorability* (98,3%), *Satisfaction* (91,1%). Kemudian pada sisi *admin* : *Efficiency* (100%), *Learnability* (100%), *Memorability* (100%), *Satisfaction* (86,7%). Dari sisi kurir maupun *admin* terhadap aspek pengujian yang telah dilakukan memperoleh predikat Sangat Baik. Hasil ini menunjukkan bahwa sistem ini efisien digunakan, mudah digunakan, dapat diingat oleh user untuk penggunaannya, serta memuaskan pengguna.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Adapun kesimpulan yang dapat diambil dalam penelitian ini adalah sebagai berikut :

1. Pada pengembangan aplikasi mobile rute terpendek Si Cepat Payung Sekaki menerapkan metodologi *scrum*, penggunaan metodologi *scrum* juga membantu proses pengembangan sistem. Hal ini dapat ditunjukkan dengan selesainya *Scrum* dalam waktu 2,5 bulan (6 Mei 2024 hingga 17 Juli 2024) serta penetapan dari durasi pengerjaan dan *product backlog* sangat membantu dalam menyelesaikan sistem yang dibangun dengan tepat waktu.
2. Fitur untuk mengurutkan lokasi sesuai dengan jarak terpendek menggunakan algoritma djikstra telah berhasil diterapkan pada aplikasi, hal ini didapat berdasarkan pengujian *usability* dimana pada pengujian *usability* pada aspek *efficiency* terdapat butir pernyataan yang membahas tentang fungsionalitas dari djikstra, pada aspek *efficiency* didapat rata-rata 95% responden setuju bahwa fitur rute terpedendek dapat berjalan dengan baik pada sistem.
3. Berdasarkan hasil pengujian *usability* kepada 4 responden didapatkan hasil bahwa sistem telah dibangun dengan baik, dimana hasil dari pengujian terhadap *admin* didapat rata rata 96,67%, dan kurir 94,8%.

5.2 Saran

Adapun saran yang diberikan untuk proyek akhir ini dan dapat dikembangkan pada penelitian selanjutnya adalah sebagai berikut :

1. Penginputan Nama Penerima dan Tanda Tangan Digital pada Upload Bukti Pengantaran agar dapat mengkonfirmasi penerima paket dan menghindari terjadinya kehilangan barang
2. Menambahkan Fitur Scan QR untuk mendapatkan data pengantaran
3. Menambahkan Time Stamp dan Lokasi saat ini pada Foto Bukti Pengantaran

DAFTAR PUSTAKA

- Al Wahab, M. W (2020). Evaluasi Usability Sistem Informasi Akademik Menggunakan Metode Usability Testing dengan ISO 25023. *Jurnal Teknik Informatika dan Sistem Informasi*.
- Fadli, A. (2023). Pengembangan Front-End Profil Kelurahan Umban Sari. *Politeknik Caltex Riau*.
- Kurniawan, R., Fachrurrazi, S., & Ula, M. (2020). Sistem Informasi Geografis Pencarian Rute Terpendek Menuju Rumah Sakit dengan Menggunakan Metode Algoritma Dijkstra. *Sisfo: Jurnal Ilmiah Sistem Informasi*, 4(2), 1-19.
- Naufal, M. K., Affrianto, F., & Cahyono, A. B. (2022). Implementasi REST API Untuk Fitur Rencana Strategis Program Pada SIMPEDA. *AUTOMATA*, 3(2), 1-7.
- Nurriszky, M. F., Dwiagnes, S., & Anggraeni, N. F. (2020). Pencarian Rute Terpendek pada Distribusi Raw Material Metode Dijkstra di PT. SHP. *Jurnal Teknik: Media Pengembangan Ilmu Dan Aplikasi Teknik*, 19(1), 34-39.
- Parapat, M. N., Kusbianto, D., & Rahmat, C. (2017). Rancang Bangun Aplikasi Pencarian Rute Terpendek Jasa Kiriman Barang Berbasis Mobile dengan Metode Algoritma Dijkstra. *Jurnal Informatika Polinema*, 3(3), 15-19.
- Sibarani, R., & Paridudin, A. D. (2022). Rancang Bangun Aplikasi Pencarian Masjid Terdekat Di Wilayah Jakarta Barat

- Menggunakan Algoritma Dijkstra Berbasis Android. *Jurnal Limits*, 19(02), 41-50.
- Sunardi, Yudhana, A., & Kadim, A. A. (2019). Implementasi Algoritma Dijkstra dan Algoritma Semut Untuk Analisis Rute Transjogja Berbasis Android. *It Journal Research and Development*, 4(1), 1-9.
- Suryani, L., & Murniyasih, E. (2022). Pencarian Rute Terpendek Pada Aplikasi Ojek Sampah Dengan Menggunakan Algoritma Dijkstra. *Jurnal Tekinkom (Teknik Informasi dan Komputer)*, 5(2), 385-392.
- Yosua, S., Sigalingging, C., Jipesya, J., & Jumaryadi, Y. (2021). Implementasi Algoritma Dijkstra dalam Pencarian Klinik Hewan Terdekat. *Jurnal Ilmiah FIFO*, 13(1), 85-92.

LAMPIRAN A INTERVIEW

Berikut ini merupakan hasil wawancara bersama Bapak Fadli Irham selaku Koordinator SiCepat Pekanbaru Cabang Payung Sekaki:

| Kode | Pertanyaan | Jawaban |
|------|--|---|
| P001 | Apakah terjadi perubahan signifikan pada transaksi pengantaran paket saat covid dan setelah covid? | Ya. Terjadi perubahan yang signifikan pada saat pandemic covid dan setelah covid. Hal demikian dapat diperhatikan berdasarkan lingkungan pengantaran, sebagai contoh di daerah Rumbai yang kebanyakan adalah mahasiswa maka pada saat tersebut jumlah paket menurun signifikan. Tetapi lain halnya dengan seperti di lingkungan perumahan dan perkantoran, jumlah paket relatif tetap. Setelah pandemic covid, jumlah paket yang semula turun pun meningkat terutama seperti daerah yang merupakan mayoritas penduduknya mahasiswa, peningkatannyapun terasa signifikan setelah covid mereda. |
| P002 | Berapa jumlah paket yang terdapat di Region Pekanbaru? | 8000-10000 pada hari biasa dan pada <i>peak season</i> dapat terjadi penambahan sekitar 5000-7000. |
| P003 | Berapa jumlah paket yang dikirim di daerah region Payung Sekaki dalam sehari? | Pada hari biasa normalnya region Payung Sekaki dapat mengirim 1000-1200 paket dengan jumlah kurir tetap sebanyak 15 orang. |
| P004 | Dalam sehari berapa lama waktu kerja yang dilakukan oleh kurir? | Dalam waktu sehari kurir bekerja selama 9 jam dengan waktu mulai 08.00 WIB – 17.00 WIB. |

| Kode | Pertanyaan | Jawaban |
|------|---|--|
| P005 | Kapan jumlah paket dari PT Sicepat Payung Sekaki mengalami peningkatan? | Peningkatan jumlah paket ini terasa signifikan pada saat tanggal kembar maupun hari besar. |
| P006 | Apakah ada tanggal tanggal atau momen tertentu jumlah paket sangat banyak sehingga menyulitkan proses pengantaran? | Hal ini biasa disebut oleh pihak Si Cepat sebagai <i>peak season</i> , dimana jumlah paket mengalami peningkatan yang biasanya dalam waktu sehari hanya mengirimkan 1000-1200 paket, maka pada <i>peak season</i> ini pihak Si Cepat dapat mengirimkan sebanyak 1800-2000 paket. Dengan keadaan demikian, pihak Si Cepat memiliki <i>freelancer</i> yang ditugaskan pada <i>peak season</i> ini. Sehingga target pengantaran dapat sesuai dengan estimasi yang sudah ditetapkan. |
| P007 | Bagaimana sistem kerja dari proses pengantaran paket pada PT Sicepat Payung Sekaki? (dari barang disortir di gudang sampai barang tiba ditangan konsumen) | Adapun proses distribusi paket SiCepat ini dimulai dari penyortiran gudang utama yang ada di SiCepat Pekanbaru Transit (Sortation) yang ada di Jl. Soekarno – Hatta No.13. Dari pusat sortir ini, kemudian paket-paket yang akan dikirim didistribusikan ke masing-masing kantor region yang ada di Pekanbaru. |
| P008 | Apakah admin kantor berperan dalam pembagian distribusi paket yang akan dikirim oleh kurir? Jika ada tolong dijelaskan. | Ya. Adapun peran admin kantor ini dapat dilihat saat pembagian paket berdasarkan region-region yang akan dituju oleh kurir. Misalnya pembagian paket pada wilayah Payung Sekaki, Sukajadi, Senapelan dan Rumbai akan dibagi sesuai dengan kelompok wilayah masing-masing, sehingga mempermudah kurir dalam proses pengantaran paket. |

| Kode | Pertanyaan | Jawaban |
|------|---|---|
| P009 | Apakah PT Sicepat Payung Sekaki sudah memiliki aplikasi yang dapat menentukan list rute untuk pengantaran paket sehingga proses pengantaran lebih mudah? | Sampai saat ini SiCepat Region Payung Sekaki belum memiliki aplikasi yang dapat menentukan daftar rute dalam proses pengantaran paket. |
| P010 | Bagaimana sistem pembagian pengantaran paket setiap kurir, apakah pembagian ditentukan berdasarkan jarak, wilayah, dan jenis paket atau kurir dapat memilih paket yang ingin diantar? | Pembagian paket yang akan dikirim dibagi berdasarkan wilayah kerja yang dibagi berdasarkan kelurahan , misalnya kurir A mengirimkan di Sukajadi , yang dimana wilayah tersebut luas maka kurir akan dibagi kembali wilayah tersebut menjadi beberapa kurir. Jadi tidak menutup kemungkinan wilayah Sukajadi akan dipegang oleh beberapa kurir. Adapun untuk pilihan jenis paket tidak bisa dipilih oleh kurir sendiri, karena pembagiannya dilakukan oleh Admin kantor. |
| P011 | Apakah kurir dapat menentukan jumlah paket yang ingin diantar di hari itu? | Untuk jumlahnya tidak dapat ditentukan oleh kurir. Tetapi misalnya kurir tersebut memiliki tanggung jawab pengiriman 200 paket, maka seminimalnya paket yang dapat dikirim adalah 150 paket. |
| P012 | Apa saja hal yang menjadi penghambat saat proses pengiriman paket? | Hal utama yang menjadi penghambat dalam proses pengiriman paket ini adalah cuaca. Karena menimbang rusaknya paket akibat hujan dan faktor lainnya, maka pihak SiCepat memilih untuk melakukan penundaaan pengiriman paket. |
| P013 | Jika terjadi hal seperti ban bocor,cuaca tidak mendukung, | Jika hal demikian terjadi, biasanya kurir akan melaporkan kepada pihak kantor melalui <i>WhatsApp Group</i> yang sudah |

| Kode | Pertanyaan | Jawaban |
|------|--|---|
| | kesalahan dalam pengambilan paket, apakah kurir akan melakukan proses konfirmasi dengan konsumen? | dimiliki, sehingga dapat meminta bantuan agar mempermudah proses pengiriman. |
| P014 | Pada saat proses pengiriman barang, jika terjadi penundaan proses kirim apakah ada batas maksimal untuk mengulang pengiriman? | Ya. Proses pengiriman akan diulang sebanyak 2 kali, jika setelah 2 kali percobaan pengiriman gagal maka paket akan di retur ke alamat pengirim. |
| P015 | Jika paket tidak sampai ke tangan konsumen sesuai dengan estimasi yang di tetapkan apakah akan ada dampak atau sanksi yang diperoleh oleh pihak ekspedisi? | Ya, sanksi yang akan diterima oleh pihak SiCepat adalah terkena pinalti atau biasa disebut dengan <i>Claim</i> yang dimana <i>claim</i> ini merupakan pembayaran barang yang tidak terkirim oleh pihak koordinator region setempat. |
| P016 | Jika aplikasi ini dibuat apakah harapan kurir untuk aplikasi ini? Apakah bisa digunakan untuk mencari lokasi atau yang lainnya. | Harapannya jika aplikasi ini sudah selesai dibangun tentunya akan mempermudah kurir dalam mencari lokasi pengantaran sehingga semakin mudah dalam proses distribusi paket untuk sampai ke tangan konsumen. |

LAMPIRAN B DOKUMENTASI WAWANCARA



LAMPIRAN C

IMPLEMENTASI METODE SCRUM

Sprint 1

☐ **SP Sprint 1** 3 Aug – 27 Aug (10 issues) 0 0 0 Complete sprint ...

Menyelesaikan semua halaman yang telah direncanakan

| | | | | |
|-------|-----------------------------------|------|---|---|
| 5P-1 | Membuat halaman Login (All User) | DONE | - | + |
| 5P-2 | Membuat halaman dashboard | DONE | - | + |
| 5P-3 | Membuat halaman kelola data Admin | DONE | - | + |
| 5P-4 | Membuat halaman tambah data Admin | DONE | - | + |
| 5P-5 | Membuat halaman ubah data Admin | DONE | - | + |
| 5P-6 | Membuat halaman hapus data Admin | DONE | - | + |
| 5P-7 | Membuat halaman Kelola data kurir | DONE | - | + |
| 5P-8 | Membuat halaman tambah data kurir | DONE | - | + |
| 5P-9 | Membuat halaman ubah data kurir | DONE | - | + |
| 5P-10 | Membuat halaman hapus data kurir | DONE | - | + |

Sprint 2

☐ **SP Sprint 2** 4 Jun – 20 Jul (10 issues) 0 0 0 Complete sprint ...

Menyelesaikan seluruh halaman yang direncanakan pada sprint kedua

| | | | | |
|-------|--|------|---|---|
| 5P-20 | Membuat halaman Riwayat pengantaran | DONE | - | + |
| 5P-19 | Membuat halaman status pengantaran | DONE | - | + |
| 5P-18 | Membuat halaman lihat rute pengantaran | DONE | - | + |
| 5P-17 | Membuat halaman profil kurir | DONE | - | + |
| 5P-16 | Membuat API untuk menghubungkan ke aplikasi Mobile | DONE | - | + |
| 5P-14 | Membuat halaman ekspor rute pengantaran | DONE | - | + |
| 5P-13 | Membuat halaman ubah rute pengantaran | DONE | - | + |
| 5P-12 | Membuat halaman tambah rute pengantaran | DONE | - | + |
| 5P-11 | Membuat halaman Kelola rute pengantaran | DONE | - | + |
| 5P-15 | Membuat halaman hapus rute pengantaran | DONE | - | + |

LAMPIRAN D DAILY MEETING SCRUM

Sprint 1

| Tanggal | Wilson Wijaya | Saniyyah Azhari |
|----------------|---|---|
| 6/5/24 | <ul style="list-style-type: none"> • Mencari referensi untuk tampilan login Admin <ul style="list-style-type: none"> • Tidak ada kendala | <ul style="list-style-type: none"> • Mencari referensi untuk membuat sistem aplikasi • Tidak terdapat kendala |
| 7/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend login user untuk Admin • (Membuat Model, Migration, Seed) • Unit Testing Model • Tidak ada kendala | <ul style="list-style-type: none"> • Mencari referensi untuk membuat sistem aplikasi • Tidak terdapat kendala |
| 8/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend login user untuk Admin • (Controller function auth, logout) • Unit Testing Controller • Error login, masalah enkripsi password | <ul style="list-style-type: none"> • Mencari referensi untuk membuat sistem aplikasi • Tidak terdapat kendala |
| 9/5/24 | <ul style="list-style-type: none"> • Mencari referensi untuk tampilan dashboard • Tidak ada kendala | <ul style="list-style-type: none"> • Mencari referensi untuk membuat sistem aplikasi • Tidak terdapat kendala |
| 10/5/24 | Libur | Libur |
| 11/5/24 | Libur | Libur |
| 12/5/24 | Libur | Libur |

| 13/5/24 | Libur | Libur |
|---------|---|---|
| 14/5/24 | <ul style="list-style-type: none"> • Mengerjakan tampilan dashboard • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Login pada Figma untuk antarmuka mobile apps • Tidak terdapat kendala |
| 15/4/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola data admin • Mengerjakan tampilan kelola data admin • (Controller function index, create, store) • Unit Testing Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Profile Kurir pada Figma untuk antarmuka mobile apps • Tidak terdapat kendala |
| 16/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola data admin • (Controller function edit, update, delete) • Unit Testing Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman List Rute Pengantaran pada Figma untuk antarmuka mobile apps • Tidak terdapat kendala |
| 17/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola data kurir • Mengerjakan tampilan kelola data kurir • (Membuat Model, Migration, Seed) • Unit Testing Model • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman List Rute Pengantaran pada Figma untuk antarmuka mobile apps • Tidak terdapat kendala |
| 18/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola data kurir | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Lihat |

| | | |
|---------|---|--|
| | <ul style="list-style-type: none"> • (Controller function index, create, store) • Unit Testing Controller • Tidak ada kendala | Rute Pengantaran pada Figma untuk antarmuka mobile apps <ul style="list-style-type: none"> • Tidak terdapat kendala |
| 19/5/25 | Libur | Libur |
| 20/5/24 | Libur | Libur |
| 21/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola data kurir • (Controller function edit, update, delete) • Unit Testing Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Lihat Rute Pengantaran pada Figma untuk antarmuka mobile apps • Tidak terdapat kendala |

Sprint 2

| Tanggal | Wilson Wijaya | Saniyyah Azhari |
|----------------|---|--|
| 22/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran • Mengerjakan tampilan kelola data rute pengantaran • (Membuat Model, Migration, Seed) • Unit Testing Model • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Upload Bukti Pengantaran pada Figma untuk antarmuka mobile apps <ul style="list-style-type: none"> • Tidak terdapat kendala |
| 23/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Upload Bukti Pengantaran |

| | | |
|---------|--|---|
| | <ul style="list-style-type: none"> • (Controller function index, create, store) • Unit Testing Controller • Masih error belum bisa input pengantaran | <p>pada Figma untuk antarmuka mobile apps</p> <ul style="list-style-type: none"> • Tidak terdapat kendala |
| 24/5/24 | Libur | Libur |
| 25/5/24 | Libur | Libur |
| 26/5/24 | Libur | Libur |
| 27/5/24 | Libur | Libur |
| 28/5/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran • (Controller function index, create, store) • Mencari cara untuk get api google maps • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Riwayat Pengantaran pada Figma untuk antarmuka mobile apps • Tidak terdapat kendala |
| 29/5/24 | <ul style="list-style-type: none"> • Mencari cara untuk mengaplikasikan algoritma djikstra pada sistem • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat tampilan ui/ux halaman Riwayat Pengantaran pada Figma untuk antarmuka mobile apps • Tidak terdapat kendala |
| 30/5/24 | <ul style="list-style-type: none"> • Mencari cara untuk mengaplikasikan algoritma djikstra pada sistem • Tidak ada kendala | <ul style="list-style-type: none"> • Mempelajari cara membuat tampilan pada dart • Tidak terdapat kendala |

| | | |
|---------|--|---|
| 31/5/24 | <ul style="list-style-type: none"> • Mencari cara untuk mengaplikasikan algoritma djikstra pada sistem • Tidak ada kendala | <ul style="list-style-type: none"> • Mempelajari cara membuat tampilan pada dart • Tidak terdapat kendala |
| 1/6/24 | Libur | Libur |
| 2/6/24 | Libur | Libur |
| 3/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran • (Controller function index, create, store) • Unit Testing Controller • Belum bisa get api google maps | <ul style="list-style-type: none"> • Mempelajari cara membuat tampilan pada dart • Tidak terdapat kendala |
| 4/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran • (Controller function index, create, store) • Unit Testing Controller • Belum bisa get api google maps | <ul style="list-style-type: none"> • Membuat halaman login untuk kurir • Tidak terdapat kendala |
| 5/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran • (Controller function index, create, store) • Unit Testing Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman login untuk kurir • Tidak terdapat kendala |
| 6/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran | <ul style="list-style-type: none"> • Membuat halaman login untuk kurir • Tidak terdapat kendala |

| | | |
|---------|---|---|
| | <ul style="list-style-type: none"> • (Controller function edit, update) • Unit Testing Controller • Tidak ada kendala | |
| 7/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola rute pengantaran • (Controller function getKurirsByRegion, delete) • Unit Testing Controller • Error delete data rute pengantaran, karena tidak deklarasi pengantaran_id pada function delete | <ul style="list-style-type: none"> • Membuat halaman login untuk kurir • Tidak terdapat kendala |
| 8/6/24 | Libur | Libur |
| 9/6/24 | Libur | Libur |
| 10/6/24 | <ul style="list-style-type: none"> • Mencari cara untuk mengaplikasikan algoritma djikstra & haversine pada sistem • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman profile kurir • Tidak terdapat kendala |
| 11/6/24 | <ul style="list-style-type: none"> • Mencari cara untuk mengaplikasikan algoritma djikstra & haversine pada sistem • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman profile kurir • Tidak terdapat kendala |
| 12/6/24 | <ul style="list-style-type: none"> • Mencari cara untuk mengaplikasikan algoritma djikstra & haversine pada sistem • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman profile kurir • Tidak terdapat kendala |

| | | |
|---------|--|---|
| 13/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola bukti pengantaran • Membuat tampilan kelola bukti pengantaran (Membuat Model, Migration, Seed) • (Controller function index, create, store) • Unit Testing Model dan Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman profile kurir • Tidak terdapat kendala |
| 14/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola bukti pengantaran (Controller function edit, update, delete) • Unit Testing Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman List Rute Pengantaran • Tidak terdapat kendala |
| 15/6/24 | Libur | Libur |
| 16/6/24 | Libur | Libur |
| 17/6/24 | Libur | Libur |
| 18/6/24 | Libur | Libur |
| 19/6/24 | <ul style="list-style-type: none"> • Mengerjakan backend kelola bukti pengantaran (Membuat Model, Migration, Seed) • (Controller function index, create, store, edit, update, delete) • Unit Testing Model dan Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman List Rute Pengantaran • Tidak terdapat kendala |

| | | |
|---------|--|--|
| 20/6/24 | <ul style="list-style-type: none"> • Membuat API Admin • Function index, show, create • Unit Testing API • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman Lihat Rute Pengantaran • Tidak terdapat kendala |
| 21/6/24 | <ul style="list-style-type: none"> • Membuat API Admin • Function update, delete • Unit Testing API • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman Lihat Rute Pengantaran • Tidak terdapat kendala |
| 22/6/24 | Libur | Libur |
| 23/6/24 | Libur | Libur |
| 24/6/24 | <ul style="list-style-type: none"> • Membuat API Kurir • Function login, index, show, create • Unit Testing API • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman Lihat Rute Pengantaran • Tidak terdapat kendala |
| 25/6/24 | <ul style="list-style-type: none"> • Membuat API Kurir • Function update, delete • Unit Testing API • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman Lihat Rute Pengantaran • Tidak terdapat kendala |
| 26/6/24 | <ul style="list-style-type: none"> • Membuat API Pengantaran • Function getPengantaranByKurir, updateStatus • Unit Testing API • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman upload bukti pengantaran • Tidak terdapat kendala |
| 27/6/24 | <ul style="list-style-type: none"> • Membuat API Rute • Function optimizeRoute • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman upload bukti pengantaran |

| | | |
|---------|---|--|
| | | <ul style="list-style-type: none"> • Tidak terdapat kendala |
| 28/6/24 | <ul style="list-style-type: none"> • Membuat API Rute • Function <code>getPendingDetailsByKuri</code> <code>r</code>, <code>calculateDistances</code> • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman upload bukti pengantaran • Tidak terdapat kendala |
| 29/6/24 | Libur | Libur |
| 30/6/24 | Libur | Libur |
| 1/7/24 | <ul style="list-style-type: none"> • Membuat API Rute • Function <code>haversine</code>, <code>sortLocationsByDistance</code> • Error logic dimana jalur yang ditampilkan pada aplikasi mengarahkan ke lokasi dengan jarak terjauh | <ul style="list-style-type: none"> • Membuat halaman upload bukti pengantaran • Tidak terdapat kendala |
| 2/7/24 | <ul style="list-style-type: none"> • Membuat API Rute • Function <code>dijkstra</code>, <code>minDistance</code> • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman riwayat pengantaran • Tidak terdapat kendala |
| 3/7/24 | <ul style="list-style-type: none"> • Membuat API Rute • Function <code>reconstructPath</code>, <code>calculateTotalDistance</code> • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman riwayat pengantaran • Tidak terdapat kendala |
| 4/7/24 | <ul style="list-style-type: none"> • Membuat API Rute • Function <code>reconstructPath</code>, <code>calculateTotalDistance</code> • Unit Testing API • Tidak ada kendala | <ul style="list-style-type: none"> • Membuat halaman riwayat pengantaran • Tidak terdapat kendala |

| | | |
|---------|--|--|
| 5/7/24 | <ul style="list-style-type: none"> • Membuat API Bukti • Function create • Unit Testing API • Error tidak bisa upload gambar | <ul style="list-style-type: none"> • Membuat halaman riwayat pengantaran • Tidak terdapat kendala |
| 6/7/24 | Libur | Libur |
| 7/7/24 | Libur | Libur |
| 8/7/24 | <ul style="list-style-type: none"> • Membuat API Bukti • Function create • Unit Testing API • Error tidak bisa upload gambar | <ul style="list-style-type: none"> • Membuat halaman riwayat pengantaran • Tidak terdapat kendala |
| 9/7/24 | <ul style="list-style-type: none"> • Membuat API Bukti • Function create • Unit Testing API • Upload gambar berhasil | <ul style="list-style-type: none"> • Menghubungkan api kurir untuk halaman login • Bug user logout ketika berpindah tab antara rute pengantaran dan akun |
| 10/7/24 | <ul style="list-style-type: none"> • Membuat controller Home • Function index • Unit Testing Controller • Tidak ada kendala | <ul style="list-style-type: none"> • Implementasi api rute dijkstra untuk halaman lihat rute pengantaran • Tidak terdapat kendala |
| 11/7/24 | <ul style="list-style-type: none"> • Mencari cara untuk hosting sistem ke heroku • Tidak ada kendala | <ul style="list-style-type: none"> • Implementasi api get pengantaran by kurir untuk halaman list rute pengantaran • Tidak terdapat kendala |

| | | |
|---------|---|---|
| 12/7/24 | <ul style="list-style-type: none"> • Deploy sistem ke heroku • Mencari cara untuk mengkoneksikan database ke dalam hosting • Tidak ada kendala | <ul style="list-style-type: none"> • Implementasi update status pengantaran untuk kurir ketika sudah menyelesaikan paket • Tidak terdapat kendala |
| 13/7/24 | Libur | Libur |
| 14/7/24 | Libur | Libur |
| 15/7/24 | <ul style="list-style-type: none"> • Melakukan koneksi database ke dalam hosting heroku dengan JawsDB • Tidak ada kendala | <ul style="list-style-type: none"> • Implementasi api bukti pengantaran di halaman bukti • Bukti gambar tidak dapat diupload |
| 16/7/24 | <ul style="list-style-type: none"> • Mencari cara untuk store gambar pada hosting • Tidak ada kendala | <ul style="list-style-type: none"> • Perbaiki bukti gambar tidak dapat diupload • Bukti gambar berhasil diupload |
| 17/7/24 | <ul style="list-style-type: none"> • Menambahkan add ons Cloudinary untuk upload gambar pada hosting • Tidak ada kendala | <ul style="list-style-type: none"> • Perbaiki bug button upload bukti tidak tampil pada layar device • Menambahkan scroll bar pada layar upload bukti pengantaran |
| 18/7/24 | <ul style="list-style-type: none"> • Testing Sistem • Tidak terdapat kendala | <ul style="list-style-type: none"> • Testing Sistem • Tidak terdapat kendala |

LAMPIRAN E BUKTI PENGUJIAN

Pengujian Usability Testing

Berikut merupakan pernyataan dan hasil penilaian pada form pengujian *usability testing* yang dilakukan pada aktor *admin* dan kurir. Skor likert ditunjukkan pada tabel dibawah ini.

| Hasil Pengujian | Keterangan | Bobot |
|-----------------|---------------------|-------|
| SS | Sangat Setuju | 5 |
| S | Setuju | 4 |
| N | Netral | 3 |
| TS | Tidak Setuju | 2 |
| STS | Sangat Tidak Setuju | 1 |

Tabel dibawah ini merupakan rekapitulasi nilai hasil pengisian kuisioner oleh respon yang mewakili kurir.

| No | Butir Kriteria | Skala | | | | |
|-------------------|---|-----------|----------|----------|-----------|------------|
| | | SS (5) | S (4) | N (3) | TS (2) | STS (1) |
| EFFICIENCY | | | | | | |
| 1 | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | 3 | 0 | 0 | 0 | 0 |
| 2 | Sistem ini memudahkan saya dalam menemukan alamat pengantaran | 1 | 2 | 0 | 0 | 0 |
| 3 | Sistem ini bekerja sesuai dengan yang saya harapkan | 2 | 1 | 0 | 0 | 0 |
| 4 | Sistem ini menampilkan informasi urutan alamat yang tepat | 3 | 0 | 0 | 0 | 0 |

| LEARNABILITY | | | | | | |
|---------------------|--|---|---|---|---|---|
| 5 | Sistem ini mudah digunakan | 3 | 0 | 0 | 0 | 0 |
| 6 | Saya tidak kesulitan menggunakan sistem ini | 2 | 1 | 0 | 0 | 0 |
| 7 | Saya dapat menggunakan sistem ini tanpa instruksi | 2 | 1 | 0 | 0 | 0 |
| 8 | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | 2 | 1 | 0 | 0 | 0 |
| MEMORABILITY | | | | | | |
| 9 | Sistem ini mudah dipahami | 3 | 0 | 0 | 0 | 0 |
| 10 | Saya belajar dengan cepat dalam menggunakan sistem ini | 3 | 0 | 0 | 0 | 0 |
| 11 | Saya mudah mengingat bagaimana cara menggunakan sistem ini | 3 | 0 | 0 | 0 | 0 |
| 12 | Sistem ini mudah untuk dipelajari cara penggunaannya | 2 | 1 | 0 | 0 | 0 |
| SATISFACTION | | | | | | |
| 13 | Saya puas dengan sistem ini | 2 | 1 | 0 | 0 | 0 |
| 14 | Sistem ini menyenangkan dan nyaman untuk digunakan | 1 | 2 | 0 | 0 | 0 |
| 15 | Saya akan tetap menggunakan sistem ini untuk kedepannya | 2 | 1 | 0 | 0 | 0 |

Selanjutnya dilakukan perhitungan skala likert :

- f. Melakukan pencarian total bobot (TB) setiap kriteria dimana :

$$TB = (\text{total jumlah responden}) \times (\text{nilai tingkat kepuasan})$$

| No | Jumlah Nilai | | | | | Total Nilai |
|----|--------------|----------|----------|-----------|------------|-------------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | |
| 1 | 3 | 0 | 0 | 0 | 0 | 15 |
| 2 | 1 | 2 | 0 | 0 | 0 | 13 |
| 3 | 2 | 1 | 0 | 0 | 0 | 14 |
| 4 | 3 | 0 | 0 | 0 | 0 | 15 |
| 5 | 3 | 0 | 0 | 0 | 0 | 15 |
| 6 | 2 | 1 | 0 | 0 | 0 | 14 |
| 7 | 2 | 1 | 0 | 0 | 0 | 14 |
| 8 | 2 | 1 | 0 | 0 | 0 | 14 |
| 9 | 3 | 0 | 0 | 0 | 0 | 15 |
| 10 | 3 | 0 | 0 | 0 | 0 | 15 |
| 11 | 3 | 0 | 0 | 0 | 0 | 15 |
| 12 | 2 | 1 | 0 | 0 | 0 | 14 |
| 13 | 2 | 1 | 0 | 0 | 0 | 14 |
| 14 | 1 | 2 | 0 | 0 | 0 | 13 |
| 15 | 2 | 1 | 0 | 0 | 0 | 14 |

- g. Menentukan jumlah skor indeks X dan Y

| Indeks | Keterangan | Jumlah skor tertinggi | Jumlah skor terendah |
|--------|---|-----------------------|----------------------|
| X | (skor likert terendah) x (total responden) | 3 | |
| Y | (skor likert tertinggi) x (total responden) | | 15 |

- h. Menentukan nilai interval dari 0% - 100%

Interval = 100 / jumlah likert

Interval = 100/5 = 20

- i. Menentukan interval skor interpretasi berdasarkan nilai interval pada poin 3.

| Interval | Skor interpretasi |
|--------------|-------------------|
| 0% - 19,99% | Sangat Tidak Baik |
| 20% - 39,99% | Tidak Baik |
| 40% - 59,99% | Normal |
| 60% - 79,99% | Baik |
| 80% - 100% | Sangat Baik |

- j. Mencari indeks% untuk setiap pernyataan dengan rumus :

Indeks% = $TB / Y \times 100$

| No | Jumlah Nilai | | | | | Total Bobot | Skor Max | Indeks (%) | Kategori |
|----|--------------|-------|-------|--------|---------|-------------|----------|------------|-------------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | | | | |
| 1 | 3 | 0 | 0 | 0 | 0 | 15 | 15 | 100% | Sangat Baik |
| 2 | 1 | 2 | 0 | 0 | 0 | 13 | 15 | 86,7% | Sangat Baik |
| 3 | 2 | 1 | 0 | 0 | 0 | 14 | 15 | 93,3% | Sangat Baik |
| 4 | 3 | 0 | 0 | 0 | 0 | 15 | 15 | 100% | Sangat Baik |

| | | | | | | | | | |
|----|---|---|---|---|---|----|----|-------|-------------|
| 5 | 3 | 0 | 0 | 0 | 0 | 15 | 15 | 100% | Sangat Baik |
| 6 | 2 | 1 | 0 | 0 | 0 | 14 | 15 | 93,3% | Sangat Baik |
| 7 | 2 | 1 | 0 | 0 | 0 | 14 | 15 | 93,3% | Sangat Baik |
| 8 | 2 | 1 | 0 | 0 | 0 | 14 | 15 | 93,3% | Sangat Baik |
| 9 | 3 | 0 | 0 | 0 | 0 | 15 | 15 | 100% | Sangat Baik |
| 10 | 3 | 0 | 0 | 0 | 0 | 15 | 15 | 100% | Sangat Baik |
| 11 | 3 | 0 | 0 | 0 | 0 | 15 | 15 | 100% | Sangat Baik |
| 12 | 2 | 1 | 0 | 0 | 0 | 14 | 15 | 93,3% | Sangat Baik |
| 13 | 2 | 1 | 0 | 0 | 0 | 14 | 15 | 93,3% | Sangat Baik |
| 14 | 1 | 2 | 0 | 0 | 0 | 13 | 15 | 86,7% | Sangat Baik |
| 15 | 2 | 1 | 0 | 0 | 0 | 14 | 15 | 93,3% | Sangat Baik |

| No. | Butir Kriteria | Persentase | Keterangan |
|---------------------|---|------------|-------------|
| EFFICIENCY | | | |
| 1. | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | 100% | Sangat Baik |
| 2. | Sistem ini memudahkan saya dalam menemukan alamat pengantaran | 86,7% | Sangat Baik |
| 3. | Sistem ini bekerja sesuai dengan yang saya harapkan | 93,3% | Sangat Baik |
| 4. | Sistem ini menampilkan informasi urutan alamat yang tepat | 100% | Sangat Baik |
| LEARNABILITY | | | |
| 5. | Sistem ini mudah digunakan | 100% | Sangat Baik |
| 6. | Saya tidak kesulitan menggunakan sistem ini | 93,3% | Sangat Baik |
| 7. | Saya dapat menggunakan sistem ini tanpa instruksi | 93,3% | Sangat Baik |
| 8. | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | 93,3% | Sangat Baik |
| MEMORABILITY | | | |
| 9. | Sistem ini mudah dipahami | 100% | Sangat Baik |
| 10. | Saya belajar dengan cepat dalam menggunakan sistem ini | 100% | Sangat Baik |
| 11. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | 100% | Sangat Baik |

| | | | |
|---------------------|---|-------|-------------|
| 12. | Sistem ini mudah untuk dipelajari cara penggunaannya | 93,3% | Sangat Baik |
| SATISFACTION | | | |
| 13. | Saya puas dengan sistem ini | 93,3% | Sangat Baik |
| 14. | Sistem ini menyenangkan dan nyaman untuk digunakan | 86,7% | Sangat Baik |
| 15. | Saya akan tetap menggunakan sistem ini untuk kedepannya | 93,3% | Sangat Baik |
| | Rata-rata Indeks | 94,8% | Sangat Baik |

Berdasarkan rekapitulasi hasil persentase diatas, dapat dihitung persentase keberhasilan dari masing-masing butir kriteria dengan rata-rata sebagai berikut :

5. Butir kriteria *Efficiency*

$$=(100+86,7+93,3+100) / 4$$

$$=95\%$$
6. Butir kriteria *Leanability*

$$=(100+93,3+93,3+93,3) / 4$$

$$=94,9\%$$
7. Butir kriteria *Memorability*

$$=(100+100+100+93,3) / 4$$

$$=98,3\%$$
8. Butir kriteria *Satisfaction*

$$=(93,3+86,7+93,3) / 3$$

$$=91,1\%$$

Tabel dibawah ini merupakan rekapitulasi nilai hasil pengisian kuisisioner oleh respon yang mewakili admin.

| No | Butir Kriteria | Skala | | | | |
|---------------------|--|-----------|----------|----------|-----------|------------|
| | | SS (5) | S (4) | N (3) | TS (2) | STS (1) |
| EFFICIENCY | | | | | | |
| 1 | Sistem ini memudahkan saya dalam menginputkan alamat | 1 | 0 | 0 | 0 | 0 |
| 2 | Sistem ini bekerja sesuai yang saya harapkan | 1 | 0 | 0 | 0 | 0 |
| 3 | Sistem ini menampilkan informasi urutan alamat yang tepat | 1 | 0 | 0 | 0 | 0 |
| LEARNABILITY | | | | | | |
| 4 | Sistem ini mudah digunakan | 1 | 0 | 0 | 0 | 0 |
| 5 | Saya tidak kesulitan menggunakan sistem ini | 1 | 0 | 0 | 0 | 0 |
| 6 | Saya dapat menggunakan sistem ini tanpa instruksi | 1 | 0 | 0 | 0 | 0 |
| 7 | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | 1 | 0 | 0 | 0 | 0 |
| MEMORABILITY | | | | | | |
| 8 | Sistem ini mudah dipahami | 1 | 0 | 0 | 0 | 0 |
| 9 | Saya belajar dengan cepat dalam menggunakan sistem ini | 1 | 0 | 0 | 0 | 0 |

| | | | | | | |
|---------------------|--|---|---|---|---|---|
| 10 | Saya mudah mengingat bagaimana cara menggunakan sistem ini | 1 | 0 | 0 | 0 | 0 |
| 11 | Sistem ini mudah untuk dipelajari cara penggunaannya | 1 | 0 | 0 | 0 | 0 |
| SATISFACTION | | | | | | |
| 12 | Saya puas dengan sistem ini | 0 | 1 | 0 | 0 | 0 |
| 13 | Sistem ini menyenangkan dan nyaman untuk digunakan | 1 | 0 | 0 | 0 | 0 |
| 14 | Saya akan tetap menggunakan sistem ini untuk kedepannya | 0 | 1 | 0 | 0 | 0 |

Selanjutnya dilakukan perhitungan skala likert :

- a. Melakukan pencarian total bobot (TB) setiap kriteria dimana :

$$TB = (\text{total jumlah responden}) \times (\text{nilai tingkat kepuasan})$$

| No | Jumlah Nilai | | | | | Total Nilai |
|----|--------------|----------|----------|-----------|------------|-------------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | |
| 1 | 1 | 0 | 0 | 0 | 0 | 5 |
| 2 | 1 | 0 | 0 | 0 | 0 | 5 |
| 3 | 1 | 0 | 0 | 0 | 0 | 5 |
| 4 | 1 | 0 | 0 | 0 | 0 | 5 |
| 5 | 1 | 0 | 0 | 0 | 0 | 5 |
| 6 | 1 | 0 | 0 | 0 | 0 | 5 |

| | | | | | | |
|----|---|---|---|---|---|---|
| 7 | 1 | 0 | 0 | 0 | 0 | 5 |
| 8 | 1 | 0 | 0 | 0 | 0 | 5 |
| 9 | 1 | 0 | 0 | 0 | 0 | 5 |
| 10 | 1 | 0 | 0 | 0 | 0 | 5 |
| 11 | 1 | 0 | 0 | 0 | 0 | 5 |
| 12 | 0 | 1 | 0 | 0 | 0 | 4 |
| 13 | 1 | 0 | 0 | 0 | 0 | 5 |
| 14 | 0 | 1 | 0 | 0 | 0 | 4 |

b. Menentukan jumlah skor indeks X dan Y

| Indeks | Keterangan | Jumlah skor tertinggi | Jumlah skor terendah |
|--------|---|-----------------------|----------------------|
| X | (skor likert terendah) x (total responden) | 1 | |
| Y | (skor likert tertinggi) x (total responden) | | 5 |

c. Menentukan nilai interval dari 0% - 100%

Interval = 100 / jumlah likert

Interval = 100/5 = 20

d. Menentukan interval skor interpretasi berdasarkan nilai interval pada poin 3.

| Interval | Skor interpretasi |
|--------------|-------------------|
| 0% - 19,99% | Sangat Tidak Baik |
| 20% - 39,99% | Tidak Baik |
| 40% - 59,99% | Normal |

| | |
|--------------|-------------|
| 60% - 79,99% | Baik |
| 80% - 100% | Sangat Baik |

e. Mencari indeks% untuk setiap pernyataan dengan rumus :

$$\text{Indeks\%} = \text{TB} / \text{Y} \times 100$$

| No | Jumlah Nilai | | | | | Total Bobot | Skor Max | Indeks (%) | Kategori |
|----|--------------|-------|-------|--------|---------|-------------|----------|------------|-------------|
| | SS (5) | S (4) | N (3) | TS (2) | STS (1) | | | | |
| 1 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 2 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 3 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 4 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 5 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 6 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 7 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 8 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 9 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 10 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 11 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 12 | 0 | 1 | 0 | 0 | 0 | 4 | 5 | 80% | Sangat Baik |
| 13 | 1 | 0 | 0 | 0 | 0 | 5 | 5 | 100% | Sangat Baik |
| 14 | 0 | 1 | 0 | 0 | 0 | 4 | 5 | 80% | Sangat Baik |

| No. | Butir Kriteria | Persentase | Keterangan |
|---------------------|--|------------|-------------|
| EFFICIENCY | | | |
| 1. | Sistem ini memudahkan saya dalam menginputkan alamat | 100% | Sangat Baik |
| 2. | Sistem ini bekerja sesuai yang saya harapkan | 100% | Sangat Baik |
| 3. | Sistem ini menampilkan informasi urutan alamat yang tepat | 100% | Sangat Baik |
| LEARNABILITY | | | |
| 4. | Sistem ini mudah digunakan | 100% | Sangat Baik |
| 5. | Saya tidak kesulitan menggunakan sistem ini | 100% | Sangat Baik |
| 6. | Saya dapat menggunakan sistem ini tanpa instruksi | 100% | Sangat Baik |
| 7. | Saya tidak melihat adanya tidak konsistenan selama saya menggunakannya | 100% | Sangat Baik |
| MEMORABILITY | | | |
| 8. | Sistem ini mudah dipahami | 100% | Sangat Baik |
| 9. | Saya belajar dengan cepat dalam | 100% | Sangat Baik |

| | | | |
|---------------------|--|--------|-------------|
| | menggunakan sistem ini | | |
| 10. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | 100% | Sangat Baik |
| 11. | Sistem ini mudah untuk dipelajari cara penggunaannya | 100% | Sangat Baik |
| SATISFACTION | | | |
| 12. | Saya puas dengan sistem ini | 80% | Sangat Baik |
| 13. | Sistem ini menyenangkan dan nyaman untuk digunakan | 100% | Sangat Baik |
| 14. | Saya akan tetap menggunakan sistem ini untuk kedepannya | 80% | Sangat Baik |
| | Rata-rata Indeks | 96,67% | Sangat Baik |

Berdasarkan rekapitulasi hasil persentase diatas, dapat dihitung persentase keberhasilan dari masing-masing butir kriteria dengan rata-rata sebagai berikut :

5. Butir kriteria *Efficiency*

$$=(100+100+100) / 3$$

$$=100\%$$
6. Butir kriteria *Leanability*

$$=(100+100+100+100) / 4$$

$$=100\%$$
7. Butir kriteria *Memorability*

$$=(100+100+100+100) / 4$$

$$=100\%$$
8. Butir kriteria *Satisfaction*

$$=(80+100+80) / 3$$
$$=86,67\%$$

Berita Acara Pengujian

Pada hari Rabu tanggal 7 Agustus 2024, dengan ini menerangkan bahwa yang bertanda tangan dibawah ini:

Nama : Wilson Wijaya
NIM : 2055301146
Prodi : Teknik Informatika
Jurusan : Teknologi Informasi

Telah melakukan pengujian dengan:

Nama Narasumber : Fadli Irham
Jabatan : Koordinator PT. SICEPAT PAYUNG SEKAKI

Dalam rangka Menyusun Tugas Akhir dengan judul "Penerapan Metode SCRUM Pada Sistem Back-End Aplikasi Mobile Pencarian Rute Terpendek Dengan Algoritma Dijkstra PT. SICEPAT PAYUNG SEKAKI"

Narasumber



Fadli Irham



Pekalongan, 7 Agustus 2024


Pewawancara



Wilson Wijaya

LEMBAR PENGESAHAN


| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |
|--|--|--|--|
| Disiapkan Oleh: | | Diperiksa Oleh: | |
| Nama : Wilson Wijaya | | Nama : LUTHFI PRIF | |
| Posisi : Pembuat Sistem | | Posisi : <i>for</i> | |
| Tanda Tangan :  | | Tanda Tangan :  | |
| Disetujui Oleh: | | | |
| Nama : Erzi Hidayat.S.T.M.Kom. | | | |
| Pekerjaan : Dosen Pembimbing | | | |
| Instansi : Politeknik Caltex Riau | | | |
| Tanda Tangan : | | | |
| Catatan Akhir: | | | |
| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |

| Disiapkan Oleh | Diperiksa Oleh | Disetujui Oleh | Tanggal |
|----------------|---|----------------|------------|
| Wilson Wijaya |  | | 07-08-2021 |

Form Pengujian Admin



Keterangan :
 SS : Sangat Setuju
 S : Setuju
 N : Netral
 TS : Tidak Setuju
 STS : Sangat Tidak Setuju


| No. | Butir Kriteria | SS | S | N | TS | STS |
|---------------------|---|----|---|---|----|-----|
| EFFICIENCY | | | | | | |
| 1. | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | ✓ | | | | |
| 2. | Sistem ini memudahkan saya dalam menentukan alamat pengantaran | | ✓ | | | |
| 3. | Sistem ini bekerja sesuai dengan yang saya harapkan | ✓ | | | | |
| 4. | Sistem ini menampilkan informasi urutan alamat yang tepat | ✓ | | | | |
| LEARNABILITY | | | | | | |
| 5. | Sistem ini mudah digunakan | ✓ | | | | |
| 6. | Saya tidak kesulitan menggunakan sistem ini | ✓ | | | | |
| 7. | Saya dapat menggunakan sistem ini tanpa instruksi | | ✓ | | | |
| 8. | Saya tidak melihat adanya tidak konsistensi selama saya menggunakannya | | ✓ | | | |
| MEMORABILITY | | | | | | |
| 9. | Sistem ini mudah dipahami | ✓ | | | | |
| 10. | Saya belajar dengan cepat dalam menggunakan sistem ini | ✓ | | | | |
| 11. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | ✓ | | | | |
| 12. | Sistem ini mudah untuk dipelajari cara penggunaannya | ✓ | | | | |
| SATISFACTION | | | | | | |
| 13. | Saya puas dengan sistem ini | ✓ | | | | |
| 14. | Sistem ini menyenangkan dan nyaman untuk digunakan | ✓ | | | | |
| 15. | Saya akan tetap menggunakan sistem ini untuk kedepannya | ✓ | | | | |

| Disiapkan Oleh | Diperiksa Oleh | Disetujui Oleh | Tanggal |
|----------------|---|----------------|------------|
| Wilson Wijaya |  | | 07-08-2021 |

| | | |
|-------------------------------|--|--------------|
| Politeknik Caltex Riau | PENERAPAN METODE SCRUM PADA APLIKASI MOBILE PENGANTARAN PAKET PL. SICEPAT PAYUNG SEKAKI | Hal 2 dari 6 |
|-------------------------------|--|--------------|

LEMBAR PENGESAHAN


| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |
|--|--|--|--|
| Disiapkan Oleh: | | Diperiksa Oleh: | |
| Nama : Wilson Wijaya | | Nama : <i>Erri Hidayat</i> | |
| Posisi : Pembuat Sistem | | Posisi : <i>Kore</i> | |
| Tanda Tangan :  | | Tanda Tangan :  | |
| Disetujui Oleh: | | | |
| Nama : Erri Hidayat, S.T., M.Kem. | | | |
| Pekerjaan : Dosen Pembimbing | | | |
| Instansi : Politeknik Caltex Riau | | | |
| Tanda Tangan : | | | |
| Catatan Akhir: | | | |
| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |

| Disiapkan Oleh | Diperiksa Oleh | Disetujui Oleh | Tanggal |
|----------------|---|----------------|------------|
| Wilson Wijaya |  | | 07-08-2019 |

| | | |
|-------------------------------|--|--------------|
| Politeknik Caltex Riau | PENERAPAN METODE SCRUM PADA APLIKASI MOBILE PENGANTARAN PAKET PL. SICEPAT PAYUNG SEKAKI | Hal 5 dari 6 |
|-------------------------------|--|--------------|



Form Pengujian Admin


Keterangan :
 SS : Sangat Setuju
 S : Setuju
 N : Netral
 TS : Tidak Setuju
 STS : Sangat Tidak Setuju

| No. | Butir Kriteria | SS | S | N | TS | STS |
|---------------------|---|---|---|----------------|----|------------|
| EFFICIENCY | | | | | | |
| 1. | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | ✓ | | | | |
| 2. | Sistem ini memudahkan saya dalam menemukan alamat pengantaran | ✓ | | | | |
| 3. | Sistem ini bekerja sesuai dengan yang saya harapkan | ✓ | | | | |
| 4. | Sistem ini menampilkan informasi urutan alamat yang tepat | ✓ | | | | |
| LEARNABILITY | | | | | | |
| 5. | Sistem ini mudah digunakan | ✓ | | | | |
| 6. | Saya tidak kesulitan menggunakan sistem ini | ✓ | ✓ | | | |
| 7. | Saya dapat menggunakan sistem ini tanpa instruksi | ✓ | | | | |
| 8. | Saya tidak melihat adanya tidak konsistensi selama saya menggunakannya | ✓ | | | | |
| MEMORABILITY | | | | | | |
| 9. | Sistem ini mudah dipahami | ✓ | | | | |
| 10. | Saya belajar dengan cepat dalam menggunakan sistem ini | ✓ | ✓ | | | |
| 11. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | ✓ | | | | |
| 12. | Sistem ini mudah untuk dipelajari cara penggunaannya | ✓ | | | | |
| SATISFACTION | | | | | | |
| 13. | Saya puas dengan sistem ini | ✓ | ✓ | | | |
| 14. | Sistem ini menyenangkan dan nyaman untuk digunakan | ✓ | | | | |
| 15. | Saya akan tetap menggunakan sistem ini untuk kedepannya | ✓ | | | | |
| Disiapkan Oleh | | Diperiksa Oleh | | Disetujui Oleh | | Tanggal |
| Wilson Wijaya | |  | | | | 07-08-2019 |

| | | |
|-------------------------------|--|--------------|
| PoliTeknik Caltex Riau | PENERAPAN METODE SCRUM PADA APLIKASI MOBILE PENGANTARAN PAKET PL. SICEPAT PAVUNG SERAKI | Hal 2 dari 6 |
|-------------------------------|--|--------------|

LEMBAR PENGESAHAN


| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |
|--|-------------------------|--|----------------------|
| Disiapkan Oleh: | | Diperiksa Oleh: | |
| Nama : Wilson Wijaya | Posisi : Pembuat Sistem | Nama : BAND F HUBA SATHIA | Posisi : user |
| Tanda Tangan :  | | Tanda Tangan :  | |
| Disenjui Oleh: | | | |
| Nama : Erzi Hidayat,S.T.M.Kom. | | | |
| Pekerjaan : Dosen Pembimbing | | | |
| Instansi : PoliTeknik Caltex Riau | | | |
| Tanda Tangan : | | | |
| Catatan Akhir: | | | |
| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |

| Disiapkan Oleh | Diperiksa Oleh | Disetujui Oleh | Tanggal |
|----------------|---|----------------|------------|
| Wilson Wijaya |  | | 07-08-2019 |



| | | |
|-------------------------------|--|--------------|
| PoliTeknik Caltex Riau | PENERAPAN METODE SCRUM PADA APLIKASI MOBILE PENGANTARAN PAKET PL. SICEPAT PAVUNG SERAKI | Hal 5 dari 6 |
|-------------------------------|--|--------------|


Form Pengujian Admin

Keterangan :
 SS : Sangat Setuju
 S : Setuju
 N : Netral
 TS : Tidak Setuju
 STS : Sangat Tidak Setuju


| No. | Basis Kriteria | SS | S | N | TS | STS |
|---------------------|---|----------------|------------|---|----|-----|
| EFFICIENCY | | | | | | |
| 1. | Sistem ini memudahkan saya dalam menentukan alamat yang harus di tuju terlebih dahulu | ✓ | | | | |
| 2. | Sistem ini memudahkan saya dalam menemukan alamat pengantaran | ✓ | | | | |
| 3. | Sistem ini bekerja sesuai dengan yang saya harapkan | ✓ | | | | |
| 4. | Sistem ini menampilkan informasi urutan alamat yang tepat | ✓ | | | | |
| LEARNABILITY | | | | | | |
| 5. | Sistem ini mudah digunakan | ✓ | | | | |
| 6. | Saya tidak kesulitan menggunakan sistem ini | ✓ | | | | |
| 7. | Saya dapat menggunakan sistem ini tanpa instruksi | ✓ | | | | |
| 8. | Saya tidak melihat adanya titik konsistensi selama saya menggunakannya | ✓ | | | | |
| MEMORABILITY | | | | | | |
| 9. | Sistem ini mudah dipelajari | ✓ | | | | |
| 10. | Saya belajar dengan cepat dalam menggunakan sistem ini | ✓ | | | | |
| 11. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | ✓ | | | | |
| 12. | Sistem ini mudah untuk dipelajari cara penggunaannya | ✓ | | | | |
| SATISFACTION | | | | | | |
| 13. | Saya puas dengan sistem ini | ✓ | | | | |
| 14. | Sistem ini menyenangkan dan nyaman untuk digunakan | ✓ | | | | |
| 15. | Saya akan tetap menggunakan sistem ini untuk kedepannya | ✓ | | | | |
| Disiapkan Oleh | Diperiksa Oleh | Disetujui Oleh | Tanggal | | | |
| Wilson Wijaya |  | | 07-08-2019 | | | |

LEMBAR PENGESAHAN

| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |
|--|-------------------------|--|------------------|
| Disiapkan Oleh: | | Diperiksa Oleh: | |
| Nama : Wilson Wijaya | Posisi : Pembuat Sistem | Nama : FADLY (R-H-R-M) | Posisi : SPV HUB |
| Tanda Tangan :  | | Tanda Tangan :  | |
| Disetujui Oleh: | | | |
| Nama : Erzi Hidayat, S.T., M.Kom. | | | |
| Pekerjaan : Dosen Pembimbing | | | |
| Instansi : Politeknik Caltex Riau | | | |
| Tanda Tangan : | | | |
| Catatan Akhir: | | | |
| TIM PEMBUAT SISTEM | | TIM PENGGUNA | |

| Disiapkan Oleh | Diperiksa Oleh | Disetujui Oleh | Tanggal |
|----------------|---|----------------|------------|
| Wilson Wijaya |  | | 07-08-2024 |

| No. | Butir Kriteria | SS | S | N | TS | STS |
|---------------------|--|-------------------------------------|-------------------------------------|---|----|-----|
| EFFICIENCY | | | | | | |
| 1. | Sistem ini memudahkan saya dalam menginputkan alamat | <input checked="" type="checkbox"/> | | | | |
| 2. | Sistem ini bekerja sesuai yang saya harapkan | <input checked="" type="checkbox"/> | | | | |
| 3. | Sistem ini menampilkan informasi urutan alamat yang tepat | <input checked="" type="checkbox"/> | | | | |
| LEARNABILITY | | | | | | |
| 4. | Sistem ini mudah digunakan | <input checked="" type="checkbox"/> | | | | |
| 5. | Saya tidak kesulitan menggunakan sistem ini | <input checked="" type="checkbox"/> | | | | |
| 6. | Saya dapat menggunakan sistem ini tanpa instruksi | <input checked="" type="checkbox"/> | | | | |
| 7. | Saya tidak melihat adanya tidak kesussteman selama saya menggunakannya | <input checked="" type="checkbox"/> | | | | |
| MEMORABILITY | | | | | | |
| 8. | Sistem ini mudah dipahami | <input checked="" type="checkbox"/> | | | | |
| 9. | Saya belajar dengan cepat dalam menggunakan sistem ini | <input checked="" type="checkbox"/> | | | | |
| 10. | Saya mudah mengingat bagaimana cara menggunakan sistem ini | <input checked="" type="checkbox"/> | | | | |
| 11. | Sistem ini mudah untuk dipelajari cara penggunaannya | <input checked="" type="checkbox"/> | | | | |
| SATISFACTION | | | | | | |
| 12. | Saya puas dengan sistem ini | | <input checked="" type="checkbox"/> | | | |
| 13. | Sistem ini menyenangkan dan nyaman untuk digunakan | | <input checked="" type="checkbox"/> | | | |
| 14. | Saya akan tetap menggunakan sistem ini untuk kedepannya | | <input checked="" type="checkbox"/> | | | |

| Disiapkan Oleh | Diperiksa Oleh | Disetujui Oleh | Tanggal |
|----------------|---|----------------|------------|
| Wilson Wijaya |  | | 07-08-2024 |

