



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PENGEMBANGAN APLIKASI WEB *E-COMMERCE* SEBAGAI
MEDIA TRANSAKSI JUAL-BELI *ONLINE*
(STUDI KASUS: CHILI.ID)**

TUGAS AKHIR

**ANNISA TAHIRA
0110216019**

**PROGRAM STUDI INFORMATIKA
JAKARTA
JANUARI 2020**



SEKOLAH TINGGI TEKNOLOGI TERPADU NURUL FIKRI

**PENGEMBANGAN APLIKASI WEB E-COMMERCE SEBAGAI
MEDIA TRANSAKSI JUAL-BELI ONLINE
(STUDI KASUS: CHILI.ID)**

TUGAS AKHIR

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana

**ANNISA TAHIRA
0110216019**

**PROGRAM STUDI INFORMATIKA
JAKARTA
FEBRUARI 2020**

HALAMAN PERNYATAAN ORISINALITAS

**Skripsi/Tugas Akhir ini adalah hasil karya penulis,
dan semua sumber baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.**

Nama :

NIM :

Tanda Tangan :

Tanggal :

HALAMAN PENGESAHAN

Skripsi/Tugas Akhir ini diajukan oleh :

Nama :

NIM :

Program Studi :

Judul Skripsi :

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana pada Program Studi, Sekolah Tinggi Teknologi Terpadu Nurul Fikri

DEWAN PENGUJI

Pembimbing

(Hilmy Abidzar Tawakal, ST., M.Kom.)

Penguji I

Penguji II

(Ahmad Rio Adriansyah, S.Si. M.Si.)

(Drs. Rusmanto, M.M)

Ditetapkan di :

Tanggal :

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Allah SWT, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan skripsi/Tugas Akhir ini. Penulisan skripsi/Tugas Akhir ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana komputer Program Studi Informatika pada Sekolah Tinggi Teknologi Terpadu Nurul Fikri. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi/tugas akhir ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Allah SWT.
2. Bapak Lukman Rosyidi, ST, M.M., M.T., selaku Ketua Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
3. Bapak Ahmad Rio Adriansyah, S.Si. M.Si selaku Ketua Program Studi Informatika Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
4. Bapak Zaki Imaduddin S.T, M.Kom. selaku Dosen Pembimbing Akademik yang telah membimbing penulis selama berkuliah di Sekolah Tinggi Teknologi Terpadu Nurul Fikri.
5. Bapak Hilmy Abidzar Tawakal, ST., M.Kom. selaku Dosen Pembimbing Tugas Akhir penulis dalam menyelesaikan penulisan ilmiah ini.
6. Para Dosen di lingkungan Sekolah Tinggi Teknologi Terpadu Nurul Fikri yang telah membimbing penulis dalam menuntut ilmu yang telah diberikan.
7. Kedua orang tuaku, adik beserta keluarga besarku yang senantiasa selalu mendo'akan, memberi dukungan, motivasi dan nasehat yang tak ternilai harganya.
8. Bapak Reza Aldiansyah, S.T., M.T.I, Selaku Project and Product Manager startup Chilli Indonesia yang telah meluangkan waktunya untuk memberikan data yang diperlukan bagi penulisan ilmiah ini.
9. Sahabat terbaikku, Anifatul Aufah, Kuati Septiani dan Syifa Tazkiy Fauziah serta teman-teman angkatan 2016 yang tidak dapat disebutkan satu persatu yang

banyak membantu, memberikan dukungan sehingga akhirnya penulis dapat menyelesaikan skripsi ini.

Dalam penulisan ilmiah ini tentu saja masih banyak terdapat kekurangan-kekurangan yang mungkin disebabkan oleh keterbatasan kemampuan dan pengetahuan yang penulis miliki. Walaupun demikian, penulis telah berusaha menyelesaikan penulisan ilmiah ini sebaik mungkin. Oleh karena itu apabila terdapat kekurangan di dalam penulisan ilmiah ini, dengan rendah hati penulis menerima kritik dan saran dari pembaca.

Akhir kata, penulis berharap Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Depok, 10 Maret 2020

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Sekolah Tinggi Teknologi Terpadu Nurul Fikri, saya yang bertanda tangan di bawah ini:

Nama :

NIM :

Program Studi :

Jenis karya : Skripsi / Tugas Akhir

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada STT-NF **Hak Bebas Royalti Noneksklusif** (*Non-exclusive Royalty - Free Right*) atas karya ilmiah saya yang berjudul :

.....
.....
.....
.....

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini STT-NF berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di :

Pada tanggal :

Yang menyatakan

(.....)

ABSTRAK

Nama : Annisa Tahira
NIM : 0110216019
Program Studi : Teknik Informatika
Judul : Pengembangan Aplikasi Web E-Commerce sebagai Media Transaksi Jual-Beli Online (Studi Kasus: Chili.id)

Kebutuhan cabai yang sangat tinggi dari tiga sisi yaitu: rumah tangga, industri dan juga horeka (hotel, restaurant, kafe) terus meningkat setiap tahunnya. Namun masalah utama yang setiap tahunnya terjadi adalah fluktuasi harga cabai yang cukup signifikan hingga mempengaruhi inflasi. Kondisi demikian disebabkan karena tidak seimbangnya antara persediaan dan permintaan cabai di pasar. Untuk itu Chili Indonesia (Chili.id) merancang sistem bisnis pertanian terintegrasi khusus komoditas cabai untuk memenuhi permintaan cabai tersebut. Dimulai dari sisi on farm, off farm, food processing hingga pemasaran dengan menggunakan teknologi informasi (IT). Penulisan penelitian ini bertujuan untuk mengimplementasikan sebuah E-Commerce untuk kebutuhan jual-beli online pada platform chili.id. Pengembangan sistem E-Commerce ini menggunakan metode Agile Development Scrum dengan sprint sebanyak 4 iterasi. implementasi aplikasi E-Commerce ini adalah 90% untuk fungsional keseluruhan sistem pada bagian sistem pengguna user dan 62% untuk dashboard admin. Sistem E-Commerce ini dapat digunakan sebagai media transaksi jual-beli online produk-produk yang disediakan oleh chili.id.

Kata kunci : Cabai, *E-Commerce*, *Agile Development Scrum*

ABSTRACT

Name : Annisa Tahira
NIM : 0110216019
Study Program : Informatics
Title : Development of E-Commerce Web Applications as an Online
Buying and Selling Media (Case Study: Chili.id)

The need for chili is very high from three sides, there are: household, industry and also horeka (hotels, restaurants, cafes) and its continue increase every year. But the main problem that happens every year is the fluctuations in chili prices that are significant enough to affect inflation. This condition is caused by the imbalance between the supply and demand for chili on the market. For this reason, Chili Indonesia (Chili.id) designed an integrated agricultural business system specifically for chili to meet the demand for chili. Starting from the side of on farm, off farm, food processing to marketing using information technology (IT). The writing of this research aims to implement an E-Commerce for online trading needs on the chili.id platform. The development of the E-Commerce system uses the Agile Development Scrum method with 4 iterations of sprints. E-Commerce application implementation is 90% for the overall functional system in the user system and 62% for the admin dashboard. This e-commerce system can be used as a medium for online transactions of products provided by chili.id.

Key words : Chili, E-Commerce, Agile Development Scrum

DAFTAR ISI

HALAMAN PERNYATAAN ORISINALITAS	i
HALAMAN PENGESAHAN	ii
KATA PENGANTAR	iii
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
BAB I PENDAHULUAN	1
1.1 Latar belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan dan Manfaat Penelitian	3
1.4 Batasan Masalah	3
1.5 Sistematika Penulisan	4
BAB II KAJIAN LITERATUR	6
2.1 Landasan Teori	6
2.1.1 Chili Indonesia	6
2.1.2 <i>E-Commerce</i>	7
2.1.3 React	9
2.1.4 Node.js	10
2.1.5 <i>Unified Modelling Language (UML)</i>	11
2.1.6 Model Pengembangan Scrum	12
2.1.7 <i>User Story</i>	16
2.2 Pengujian	16
2.2.1 <i>Black Box Testing</i>	16
2.2.2 Skala Likert	16
2.2.3 <i>User Acceptance Testing</i>	17
2.3 Penelitian Terkait	18
2.4 Aplikasi Terkait	21

2.5	Posisi Penelitian.....	22
BAB III METODOLOGI PENELITIAN.....		23
3.1	Tahapan Penelitian	23
3.2	Rancangan Penelitian	24
3.2.1	Jenis Penelitian.....	24
3.2.2	Studi Pendahuluan.....	24
3.2.3	Metode Penelitian.....	24
3.2.4	Metode Perancangan Sistem	24
3.2.5	Metode Pengembangan Software.....	25
3.2.6	Lingkungan Pengembangan	26
3.2.7	Alat dan Bahan.....	26
3.2.8	Metode Pengujian.....	26
3.2.9	Evaluasi	27
BAB VI ANALISIS DAN PERANCANGAN SISTEM.....		28
4.1	Analisis Sistem	28
4.1.1	Analisis Sistem Berjalan	28
4.1.1	Analisis Kebutuhan Sistem dan <i>User</i>	32
4.1.2	Analisis Perbandingan Sistem.....	39
4.1.3	<i>Use Case</i>	40
4.2	Perancangan Sistem.....	43
4.2.1	Domain Model	43
4.2.2	<i>Activity Diagram</i>	44
4.2.3	Rancangan Tampilan.....	58
4.2.4	Rancangan Pengujian	62
BAB V HASIL DAN PEMBAHASAN.....		74
5.1	Implementasi Perangkat Lunak	74
5.1.1	Perangkat yang digunakan	74
5.2	Implementasi SCRUM	74
5.2.1	Scrum Team	74
5.2.2	Persiapan Sprint	75
5.1	Rangkuman hasil pengujian blackbox.....	102

5.2	Perbandingan Hasil Analisis PIECES	108
5.3	HASIL UAT (<i>User Acceptance Test</i>)	111
BAB VI SIMPULAN DAN SARAN		117
6.1	Simpulan	117
6.2	Saran	118
DAFTAR PUSTAKA		119
LAMPIRAN		

DAFTAR GAMBAR

Gambar 1. Scrum Framework [13].....	15
Gambar 2. Tahapan Penelitian.....	23
Gambar 3. Alur Diagram Sistem Berjalan.....	29
Gambar 4. Rich Picture Sistem yang Diusulkan	33
Gambar 5. Diagram Alur Sistem E-Commerce yang Diusulkan.....	34
Gambar 6. Use Case Diagram	40
Gambar 7. Package Diagram Pembelian	41
Gambar 8. Package Diagram Pengelolaan	42
Gambar 9. <i>Domain Model</i>	43
Gambar 10. Activity Diagram Melihat Produk	44
Gambar 11. Activity Diagram Pencarian Produk.....	45
Gambar 12. Activity Diagram Menambah Produk ke Shopping Cart.....	46
Gambar 13. Activity Diagram Register.....	47
Gambar 14. Activity Diagram Membeli Produk	49
Gambar 15. Activity Diagram Memperbarui Halaman Profil User	51
Gambar 16. Activity Diagram Mendapatkan Informasi Order Produk Cabai.....	52
Gambar 17. Activity Diagram Memberi Komentar Pada Halaman Produk.....	53
Gambar 18. Activity Diagram Menambah Produk.....	54
Gambar 19. Activity Diagram Update Produk	56
Gambar 20. Activity Diagram Melihat Informasi Order Customer	57
Gambar 21. Wireframe Melihat Halaman Website	58
Gambar 22. Wireframe Halaman Produk	59
Gambar 23. Wireframe Shopping Cart Produk	60
Gambar 24. Wireframe Register Akun.....	60
Gambar 25. Wireframe Login	61
Gambar 26. Wireframe Pembelian Produk.....	61
Gambar 27. Wireframe Pembayaran Produk.....	62
Gambar 28. Frontend Admin untuk Create Product.....	77
Gambar 29. Produk Baru ditampilkan pada Frontend Store	77

Gambar 30. Hasil Pencarian Produk Sesuai Kata Kunci.....	78
Gambar 31. Hasil Pencarian Produk Ketika Kata Kunci Tidak Ditemukan	78
Gambar 32. Fungsi Delete dan Edit pada Dashboard Admin.....	82
Gambar 33. Register pada chili.id	83
Gambar 34. Aktivasi akun dikirimkan ke Mailtrap.io.....	83
Gambar 35. Login pada chili.id.....	84
Gambar 36. Redirect ke user profile setelah berhasil login.....	84
Gambar 37. Menambahkan Product ke Shopping Cart	85
Gambar 38. Data yang harus diisi untuk melakukan checkout produk	91
Gambar 39. Page Konfirmasi Data Order	92
Gambar 40. Notifikasi Order Diterima.....	92
Gambar 41. Edit Profile User	93
Gambar 42. Menu Untuk Menambahkan atau Melihat Order Customer	93
Gambar 43. Menu Untuk Edit dan Delete Order.....	94
Gambar 44. Visualisasi Order berdasarkan waktu	98

DAFTAR TABEL

Tabel 1. Penelitian Terkait	18
Tabel 2. Aplikasi Terkait.....	21
Tabel 3. Penelitian Terkait	22
Tabel 4. Analisis PIECES	30
Tabel 5. User Deskripsi.....	32
Tabel 6. Istilah Baku	35
Tabel 7. User Story	36
Tabel 8. Product Backlog.....	38
Tabel 9. Deskripsi Bobot Product Backlog.....	39
Tabel 10. Perbandingan Sistem.....	39
Tabel 11. UAT Customer.....	69
Tabel 12. UAT Admin	72
Tabel 13. Skala Likert.....	73
Tabel 14 Sprint Backlog untuk sprint 1	76
Tabel 15. Hasil Pengujian Sprint Backlog pada Sprint 1	79
Tabel 16. Sprint Review untuk sprint 1	80
Tabel 17. Sprint Backlog untuk sprint 2	81
Tabel 18. Hasil Pengujian Sprint Backlog pada Sprint 2.....	86
Tabel 19. Sprint Review untuk Sprint 2.....	88
Tabel 20. Sprint Backlog untuk Sprint 3.....	90
Tabel 21. Hasil Pengujian Sprint Backlog pada Sprint 3.....	94
Tabel 22. Sprint Review untuk Sprint 3.....	96
Tabel 23. Sprint Backlog untuk Sprint 4.....	97
Tabel 24. Hasil pengujian backlog pada Sprint 4	99
Tabel 25. Hasil Pengujian Keseluruhan Backlog.....	102
Tabel 26. Perbandingan Hasil Analisis PIECES.....	108
Tabel 27. UAT pada sistem pengguna user (customer)	111
Tabel 28. UAT pada sistem pengguna admin	114

BAB I

PENDAHULUAN

1.1 Latar belakang

Cabai merupakan salah satu komoditas strategis yang memiliki nilai ekonomi yang tinggi serta peluang pasar yang besar di Indonesia. Kebutuhan cabai yang sangat tinggi dari tiga sisi yaitu: rumah tangga, industri dan juga horeka (hotel, restaurant, kafe) terus meningkat setiap tahunnya. Direktur Sayuran dan Tanaman Obat Kementan menjelaskan tingkat kebutuhan cabai nasional mengalami tren kenaikan dari April hingga Juni 2019. Berdasarkan catatan Kementerian Pertahanan, tingkat kebutuhan cabai nasional pada April 2019 mencapai 73.999 ton, Mei 75.877 ton, dan Juni berkisar 77.755 ton [3].

Namun masalah utama yang setiap tahunnya terjadi adalah fluktuasi harga cabai yang cukup signifikan hingga mempengaruhi inflasi. Berdasarkan data dari Bank Indonesia (BI) tahun 2016, harga cabai yang tidak stabil di pasaran telah menyumbang inflasi sebesar 16,1 % dari komponen pembentuk inflasi. Kondisi demikian dikarenakan adanya ketidakseimbangan antara persediaan dan permintaan cabai di pasar [2].

Potensi ini menjadi peluang untuk Chili Indonesia atau chili.id untuk memenuhi kebutuhan cabai nasional sekaligus menjadi solusi nasional untuk komoditas cabai. Chili Indonesia merancang sistem bisnis pertanian terintegrasi dari hulu hingga hilir khusus komoditas cabai. Dimulai dari *sisi on farm, off farm, food processing* hingga pemasaran dengan menggunakan teknologi informasi (IT) [1].

Pada sisi IT dibutuhkan suatu solusi yang dapat menyediakan suatu sistem yang dapat menyediakan persediaan cabai untuk kebutuhan di pasar. Salah satu caranya adalah dengan membuat sebuah *e-commerce* yang dapat menyediakan produk cabai secara cepat, dapat diakses oleh siapa saja dan dimana saja, serta mudah dibeli secara *online*.

Penggunaan *e-commerce* sebagai media penyedia transaksi jual beli secara *online* tidak lepas dari peningkatan pengguna internet di Indonesia. Menurut hasil riset yang dikeluarkan oleh Asosiasi Pengguna Jasa Internet Indonesia (APJII), jumlah pengguna internet di Indonesia pada tahun 2017 lalu mencapai angka 143,26 juta pengguna dan diperkirakan akan terus mengalami pertumbuhan [5]. Selain itu, berdasarkan keterangan dari CupoNation yang diterima Kompas.com pada tahun 2018, jumlah *online shopper* di Indonesia diperkirakan mencapai 11,9 persen dari total populasi di Indonesia dan jumlah tersebut diperkirakan akan terus meningkat setiap tahunnya [4]. Hal ini tentu menjadi peluang agar pemasaran produk cabai mencapai akses pasar yang lebih luas, dan pasokan produk cabai akan selalu terjamin dan menjaga harga cabai ditingkat pemasok maupun konsumen tetap stabil karena memotong rantai distribusi penjualan cabai yang cukup panjang.

Namun, meskipun saat ini sudah banyak tersedia *e-commerce* yang menyediakan pembelian produk cabai secara *online*, kualitas dari produk cabai terutama produk cabai segar masih diragukan. Hal ini dikarenakan tidak adanya kontrol kualitas dari perusahaan *e-commerce* nya sendiri. Karena setiap penjual dari mana saja yang memiliki produk cabai dapat dengan mudah berjualan di *e-commerce* yang ada di Indonesia, dan tidak ada jaminan dari *e-commercenya* bahwa produk yang dijualnya tersebut berkualitas sesuai dengan yang dituliskann penjual pada deskripsinya. Selain itu, persediaan cabai pada penjual tersebut tidak bisa selalu dipastikan, sehingga kurang menjamin selalu tersedianya kebutuhan cabai sesuai permintaan pasar.

Berdasarkan uraian latar belakang diatas, maka diperlukan sebuah penelitian untuk mengembangkan sebuah aplikasi berbasis web *e-commerce* pada chilli.id yang akan berfokus pada penyediaan produk cabai berkualitas yang dihasilkan dari hasil pertanian terintegrasi Chilli Indonesia. Selain itu, diharapkan *e-commerce* ini juga kedepannya dapat memberikan kemudahan kepada konsumen (*user*) dalam melakukan proses transaksi jual-beli produk cabai secara *online*.

1.2 Rumusan Masalah

Berdasarkan latar belakang diatas, maka rumusan masalah dalam penelitian ini yaitu bagaimana cara mengimplementasikan sistem transaksi pada *e-commerce* untuk kebutuhan jual-beli *online* pada marketplace chili.id?

1.3 Tujuan dan Manfaat Penelitian

Tujuan dari penelitian ini adalah mengimplementasikan sistem *e-commerce* pada chili.id untuk keperluan transaksi jual-beli *online*.

Adapun manfaat yang diharapkan dari penelitian ini adalah terimplementasinya sistem *e-commerce* untuk kebutuhan jual-beli produk-produk chili.id.

1.4 Batasan Masalah

Batasan masalah dari aplikasi web yang dikembangkan adalah sebagai berikut:

- Modul yang dibangun berkaitan dengan transaksi jual beli *online* yang berada pada platform chili.id.
- *E-Commerce* dibangun dengan menggunakan konsep *business to customer* (B2C).
- Proses transaksi belum menggunakan sistem pembayaran melalui *payment gateway*

1.5 Sistematika Penulisan

Sistem penulisan dalam tugas akhir ini disusun dalam bentuk karya ilmiah dengan struktur penulisan sebagai berikut:

BAB 1 PENDAHULUAN

Bab ini berisi penjelasan tentang latar belakang masalah yang mendasari diadakan penelitian, perumusan masalah, pembatasan masalah penelitian, maksud dan tujuan penelitian, dan sistematika pembahasan tugas akhir ini.

BAB 2 LANDASAN TEORI

Bab ini berisi tentang penjelasan teori-teori yang mendukung dalam perancangan dan pembangunan sistem *e-commerce* pada chilli.id, serta teori-teori yang diperlukan dalam penulisan tugas akhir ini.

BAB 3 ANALISIS DAN PERANCANGAN SISTEM

Bab ini berisi analisis dan perancangan sistem pada *e-commerce* chilli.id. Penulis akan menguraikan *requirement*, analisis dan perancangan sistem yang digambarkan dalam dengan menggunakan UML, serta antarmuka dari aplikasi web yang akan dibangun.

BAB 4 TESTING DAN IMPLEMENTASI

Bab ini berisi tentang implementasi sistem yang disertai dengan pembahasan sistem dan hasil pengujian yang telah didapatkan.

BAB 5 PENUTUP

Bab ini berisi tentang kesimpulan terhadap sistem dan aplikasi yang telah dibuat serta saran terhadap penelitian yang telah dilakukan sebagai referensi untuk keperluan pengembangan sistem selanjutnya.

DAFTAR PUSTAKA

Daftar pustaka berisi daftar referensi yang digunakan untuk keperluan pembangunan sistem dan penulisan tugas akhir ini.

BAB II

KAJIAN LITERATUR

2.1 Landasan Teori

2.1.1 Chili Indonesia

Chili Indonesia atau chili.id merupakan startup yang khusus memberikan solusi atas segala permasalahan cabai di Indonesia. Chili Indonesia merancang sistem bisnis pertanian terintegrasi dari hulu hingga hilir khusus komoditas cabai. Dimulai dari *sisi on farm, off farm, food processing* hingga Teknologi Informasi (IT) [1].

Pada sisi hulu, chili.id menerapkan Standar Operasional Prosedure (SOP) dan *Good Agricultural Practices* (GAP), baik pada sistem pertanian konvensional dan teknologi *greenhouse*. Selain itu, untuk memenuhi kebutuhan pasar maka diterapkan kalender tanam serta pola tanam sehingga lonjakan kebutuhan cabai bisa terjaga [1].

Pada sisi hilir, chili.id memproduksi produk olahan dari cabai. Produk olahan untuk kebutuhan konsumsi hingga obat dan kosmetik. Produk yang akan difokuskan pada tahap awal adalah produksi Sambal beserta produk turunan cabai untuk konsumsi dengan merk Rumah Sambal yang diproduksi sesuai prinsip *Good Manufacturing Practice* (GMP) [1].

Sedangkan pada sisi IT, platform ini bisa digunakan bagi petani, umkm, pembeli, serta *investor*. Platform dapat digunakan sebagai *marketplace*, Kemudian, petani atau penggiat UMKM juga dapat mendapatkan modal usaha dari investor dengan akad salam serta mendapatkan materi-materi pengetahuan dan praktis yang berkaitan dengan cabai pada platform ini [1].

2.1.2 *E-Commerce*

E-Commerce atau kependekan dari perdagangan elektronik, merupakan transaksi bisnis yang terjadi dalam jaringan elektronik, seperti internet. Siapapun yang dapat mengakses komputer, memiliki sambungan ke internet, dan memiliki sarana untuk melakukan pembayaran terhadap barang-barang atau jasa yang mereka beli, dapat berpartisipasi dalam *e-commerce* [15].

Sedangkan Kalakota R, dan Whinston, A.B, [16] mendefinisikan *e-commerce* dari beberapa perspektif berikut:

- Perspektif komunikasi (*communications*), Menurut perspektif ini, *e-commerce* merupakan pengiriman informasi, produk/jasa, dan pembayaran melalui lini telepon, jaringan komputer atau sarana elektronik lainnya.
- Perspektif proses bisnis (*business*), Menurut perspektif ini, *e-commerce* merupakan aplikasi teknologi menuju otomatisasi transaksi dan aliran kerja perusahaan (*workflow*).
- Perspektif layanan (*service*), Menurut perspektif ini *e-commerce* merupakan satu alat yang memenuhi keinginan perusahaan, konsumen, dan manajemen dalam memangkas *service cost* ketika meningkatkan mutu barang dan ketepatan pelayanan.
- Perspektif *online* (*online*), Menurut perspektif ini *e-commerce* berkaitan dengan kapasitas jual beli produk dan informasi di internet dan jasa *online* lainnya.

Selain itu, menurut Karmawan [6] *e-commerce* adalah suatu jenis dari mekanisme bisnis secara elektronik yang memfokuskan diri pada transaksi bisnis berbasis individu dengan menggunakan internet (teknologi berbasis jaringan digital) sebagai medium pertukaran barang atau jasa baik antara dua buah institusi (*business to business*) dan konsumen langsung (*business to consumer*), melewati kendala ruang dan waktu yang selama ini merupakan hal-hal yang dominan.

Dengan aplikasi *e-commerce*, hubungan antar perusahaan dengan entitas eksternal lainnya (pemasok, distributor, rekanan, konsumen) dapat dilakukan secara lebih cepat, lebih intensif, dan lebih murah daripada aplikasi prinsip manajemen secara konvensional (*door-to-door, one-to-one relationship*).

a) Teknologi yang diperlukan untuk *E-Commerce*

Menurut Turban, dkk [12] pengembangan teknologi *e-commerce* memerlukan satu set keterpaduan aplikasi yang meliputi :

1. Katalog *Online*

Katalog *online* digunakan untuk memberikan informasi kepada pelanggan atau calon pelanggan potensial untuk mendapatkan informasi yang lengkap dari suatu produk.

2. Transaksi *Online*

Transaksi *online* adalah sebuah fasilitas yang disediakan oleh suatu situs *e-commerce* kepada pelanggan untuk dapat melakukan pembelian atau *Order* barang secara *online* lewat media internet, transaksi *online* ini dibagi menjadi dua bagian yaitu:

a) *Shopping Cart*

Shopping cart software merupakan sistem yang digunakan agar calon pembeli dapat membeli barang-barang yang ditawarkan melalui katalog *online*, mengawasi account kita setiap saat dan menggabungkan semua aspek *e-commerce* pada suatu situs.

b) *Online Order*

Online order menyediakan layanan pembayaran secara *online* dari transaksi yang sudah terjadi.

2.1.3 React

React merupakan sebuah pustaka/library *JavaScript* yang digunakan untuk membangun *user interface* atau UI. Berikut merupakan keunggulan react, diantaranya [7]:

- Deklaratif

React membuat proses pembuatan antarmuka pengguna interaktif menjadi lebih mudah. Buat tampilan sederhana untuk setiap *state* di aplikasi, dan React akan secara efisien memperbarui dan *me-render* hanya komponen yang diperlukan ketika data berubah.

Tampilan yang deklaratif membuat jalannya kode menjadi lebih terprediksi dan lebih mudah untuk *di-debug*.

- Berbasis Komponen

Komponen-komponen terenkapsulasi akan mengatur *State*-nya sendiri, kemudian pengembang dapat menggabungkan komponen-komponen tersebut untuk membentuk antarmuka pengguna yang lebih kompleks.

Selain itu, karena logika pada komponen React ditulis langsung menggunakan JavaScript (bukan menggunakan template), maka dapat dengan mudah mengoper data melalui aplikasi dan menempatkan *State* agar tetap berada di luar DOM.

- React juga dapat digunakan untuk *me-render* di server menggunakan Node.
- React memungkinkan untuk dapat berinteraksi dengan *library* dan *framework* lain.

2.1.4 Node.js

Node.js adalah lingkungan atau *platform* untuk mengeksekusi kode-kode yang ditulis dalam JavaScript, yang dikenal dengan sebutan JavaScript *runtime environment*. Dalam melaksanakan tugasnya, Node.js menggunakan V8, yaitu mesin JavaScript yang diproduksi oleh Google. V8 itu sendiri bertugas untuk mengubah kode JavaScript ke dalam bentuk *bytecode* [8].

Aplikasi Node.js dijalankan dalam satu proses tunggal, tanpa membuat *thread* yang baru untuk setiap *request*. Node.js menyediakan satu set perangkat I / O asinkron yang primitif dalam pustaka standarnya yang akan mencegah kode JavaScript dari pemblokiran yang sering terjadi pada bahasa pemrograman sisi server pada umumnya, pustaka di Node.js ditulis dengan menggunakan paradigma *non-blocking* yang menjadikan perilaku pemblokiran sebagai pengecualian [8].

Melalui Node.js, kita dapat menggunakan JavaScript sebagai bahasa pemrograman untuk membuat semua tipe aplikasi, baik yang berupa aplikasi *console* (*command-line interface* – *CLI*), aplikasi berbasis GUI, aplikasi web, aplikasi *mobile* (Android atau iOS), atau aplikasi untuk sistem kontrol dan akses perangkat keras. Node.js diyakini dapat memproses permintaan-permintaan yang dikirimkan oleh klien (*web browser*) dalam jumlah besar dengan performansi yang baik [8].

Node.js masih menggunakan bahasa JavaScript standar 2015, yang dikenal dengan ECMAScript 6, atau yang sering disingkat ES6. Dalam ECMAScript 6, JavaScript telah mengalami perubahan yang berupa penambahan fitur-fitur baru seperti dukungan terhadap parameter opsional di dalam fungsi, interpolasi variabel di dalam *string*, pembuatan *string* yang terdiri dari beberapa baris, pendefinisian kelas, pewarisan kelas dan lain-lain [8].

2.1.5 *Unified Modelling Language (UML)*

Menurut Sri Dharwiyanti [16] *Unified Modelling Language (UML)* adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem.

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan *syntax*/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML *syntax* mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. UML mendefinisikan diagram-diagram sebagai berikut:

a) *Usecase Diagram*

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Use case diagram dapat sangat membantu bila kita sedang menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang test case untuk semua feature yang ada pada sistem.

b) *Class Diagram*

Class diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). *Class* diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.

c) *Activity Diagram*

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity Diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity Diagram* merupakan *state diagram* khusus, di mana sebagian besar *State* adalah *action* dan sebagian besar transisi di-*trigger* oleh selesainya *State* sebelumnya (*internal processing*). Oleh karena itu *Activity Diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

d) *Sequence Diagram*

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, *display*, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri antar dimensi vertikal (waktu) dan dimensi *horizontal* (objek-objek yang terkait). *Sequence diagram* biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan output tertentu. Diawali dari apa yang men-*trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan.

2.1.6 Model Pengembangan Scrum

a) *Scrum*

Scrum merupakan metode pengembangan perangkat lunak dengan menggunakan prinsip-prinsip pendekatan Agile yang tanggap dalam menangani setiap perubahan dan menekankan pada proses iterasi yang menghasilkan produk berupa *incremental product* [9].

1) ***Komponen Scrum***

Scrum memiliki tiga peranan penting yaitu *Product Owner*, *Scrum master*, dan *Development Team*, yaitu [11]:

- a) ***Product Owner***, seseorang yang memiliki visi, otoritas, dan ketersediaan waktu. Pemilik produk bertanggung jawab untuk terus mengkomunikasikan visi dan prioritas untuk tim pengembangan.
- b) ***Scrum Master***, seseorang yang bertindak sebagai fasilitator untuk pemilik produk dan tim pengembangan yang terdiri dari *developer* dan *tester* (*Quality Assurance*). *Scrum master* tidak bertanggung jawab dengan pengelolaan tim. Fungsi dari *scrum master* untuk menghilangkan hambatan yang menghalangi tim dari mencapai tujuannya. Hal ini membantu tim tetap kreatif dan produktif sambil memastikan keberhasilan yang terlihat ke pemilik produk. Scrum Master juga bekerja untuk memberikan nasihat kepada pemilik produk tentang bagaimana memaksimalkan Return On Investment (ROI) untuk tim.
- c) ***Scrum Development Team***, menurut pendiri Scrum “*the team is utterly*”. Tim pengembangan bertanggung jawab untuk mengatur diri untuk menyelesaikan pekerjaannya. Sebuah tim pengembangan Scrum memiliki sekitar tujuh anggota yang penuh dedikasi (resminya tiga sampai sembilan orang), idealnya dalam satu ruangan tim dilindungi dari gangguan dari luar. Untuk proyek pengembangan perangkat lunak, tim umum meliputi insinyur perangkat lunak (*software engineers*), arsitek (*architects*), *programmer*, analis (*analyst*), ahli QA (*Quality Assurance*), penguji (*testers*), dan UI designers. Setiap *sprint*, tim bertanggung jawab untuk menentukan bagaimana ia akan menyelesaikan pekerjaan yang harus diselesaikan. Tim memiliki otonomi dan tanggung jawab untuk memenuhi tujuan dari *sprint*.

Scrum juga mengidentifikasi empat objek artefak yang dioperasikan oleh tim Scrum selama siklus pengembangan, yaitu:

a) Product Backlog

1. Menentukan peringkat dari fungsi yang diinginkan.
2. Bisa dilihat dan ditambahkan oleh semua stakeholder (termasuk tim).
3. Pemilik produk menentukan prioritas kerja setiap saat.
4. Item yang berada di paling atas lebih rinci dari yang bawah.
5. Dipertahankan selama Backlog Refinement Meeting.

b) Sprint Backlog

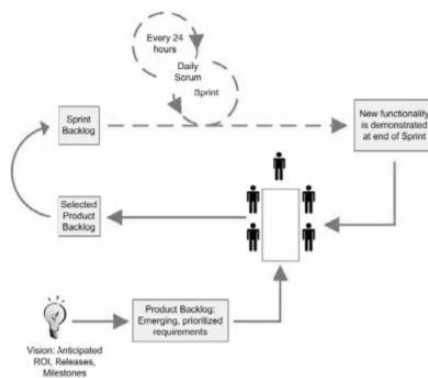
1. Terdiri dari item yang sudah dinegosiasikan antara tim dan pemilik produk.
2. Lingkup komitmen tidak berubah selama Sprint Execution.
3. Semua tugas awal sudah diidentifikasi selama Sprint Planning Meeting.
4. Tim akan menemukan tugas tambahan yang diperlukan untuk memenuhi komitmen yang ada.
5. Semua anggota tim bisa melihat tugas – tugas apa saja yang ada di dalam Sprint Execution.

c) Sprint Task

1. Menentukan bagaimana cara untuk memenuhi / mencapai Product Backlog Item (PBI).
2. Membutuhkan satu hari atau kurang dari satu hari kerja.
3. Sisa dari upaya (waktu) pengerjaan akan diestimasi ulang setiap harinya, biasanya dalam bentuk jam.
4. Selama Sprint Execution, anggota tim bisa menjadi relawan untuk jadi penanggung jawab atas sebuah tugas.
5. Sprint Task menjadi tanggung jawab seluruh anggota tim, jadi semua anggota tim harus bisa berkolaborasi.

Interaksi antara peran-peran menggunakan objek artefak di atas menggunakan beberapa jenis pertemuan pada Scrum [13], yaitu:

- 1) ***Sprint***, kerangka waktu iterasi dengan durasi maksimal satu bulan untuk mengembangkan produk.
- 2) ***Sprint Planning Meeting***, rapat perencanaan *Sprint* yang dilakukan di awal untuk memilih fitur-fitur apa saja yang akan dikerjakan.
- 3) ***Sprint Review***, pertemuan evaluasi pelaksanaan *Sprint* yang dilakukan di akhir *Sprint*. Pada pertemuan ini produk perangkat lunak akan didemonstrasikan kepada *Product Owner*.
- 4) ***Daily Scrum***, pertemuan harian bagi tim pengembang.
- 5) ***Sprint Retrospective***, pertemuan yang dilakukan setelah *Sprint Review* dan sebelum *Sprint Planning* berikutnya untuk kilas balik *Sprint* yang bertujuan mencari hal-hal yang dapat ditingkatkan pada *Sprint* berikutnya. Siklus iterasi Scrum ditampilkan pada gambar



Gambar 1. Scrum Framework [13]

Scrum memperkenalkan konsep *sprint* yang merepresentasikan sebuah iterasi dari siklus pengembangan berbasis waktu dengan durasi selama dua minggu sampai satu bulan. Inti dari Scrum terdiri dari satu set *sprint-sprint* yang menghasilkan perangkat lunak jadi pada setiap akhir *sprint*. *Sprint* adalah kerangka waktu iterasi dengan durasi maksimal satu bulan untuk menghasilkan produk dengan definisi “Done” atau “Selesai”, dapat digunakan, dan berpotensi untuk dirilis .

2.1.7 User Story

User Story merupakan salah satu hal utama yang ada pada artifacts scrum. *User story* adalah kebutuhan yang paling tinggi dan berisi informasi mengenai informasi-informasi yang dapat membantu *developer* dalam menjalankan dan mengestimasi waktu pada produk atau proyek yang sedang berlangsung [20].

2.2 Pengujian

2.2.1 Black Box Testing

Black Box Testing berfokus pada spesifikasi fungsional dari perangkat lunak. *Tester* dapat mendefinisikan kumpulan kondisi *input* dan melakukan pengujian pada spesifikasi fungsional program [17]. *Black Box Testing* cenderung untuk menemukan hal-hal berikut:

1. Fungsi yang tidak benar atau tidak ada.
2. Kesalahan antarmuka (*interface errors*).
3. Kesalahan pada struktur data dan akses basis data.
4. Kesalahan performansi (*performance errors*).
5. Kesalahan inisialisasi dan terminasi.

2.2.2 Skala Likert

Menurut Bilson Simamora [18], Skala Likert disebut juga *summated rating scale*. Skala ini banyak digunakan karena memberi peluang kepada responden untuk mengekspresikan perasaan mereka dalam bentuk persetujuan terhadap suatu pernyataan. Pertanyaan yang diberikan berjenjang, mulai dari tingkat terendah sampai tertinggi. Jumlah pilihan jawabannya bisa tiga, lima, tujuh, sembilan, yang jelas harus ganjil.

2.2.3 *User Acceptance Testing*

User Acceptance Testing (UAT) merupakan sekumpulan urutan langkah pengujian sebuah aplikasi di sisi pengguna, menggunakan format yang telah disepakati bersama, dengan tujuan untuk mengetahui sejauh mana pemahaman pengguna terhadap aplikasi yang disajikan, serta apakah aplikasi telah cukup mampu memenuhi kebutuhan pengguna dan menyelesaikan permasalahan yang terjadi, dengan hasil akhir sebuah dokumen pelengkap pengembangan aplikasi [19].

2.3 Penelitian Terkait

Dalam penyusunan proposal tugas akhir ini, dilakukan studi literatur terhadap penelitian terkait untuk membantu penelitian yang dilakukan. Berikut beberapa penelitian yang berkaitan dengan penelitian yang akan dilakukan, diantaranya:

Tabel 1. Penelitian Terkait

No	Nama dan Tahun	Judul	Topik	Tools	Metodologi	Deskripsi dan Hasil
1.	Rahmad Ilham Pratama Institut Pertanian Bogor 2018	Pengembangan Back End Bagian Provider pada <i>Marketplace</i> Travnesia.com dengan Rest Api	Pengembangan Back End Bagian <i>Provider</i> pada <i>Marketplace</i>	Node.Js Database : MongoDB	<i>Scrum</i>	Travnesia.com dikerjakan oleh delapan orang pengembang yaitu pengembangan UI/UX, back end, dan sisi <i>front end</i> website. Pengembangan menggunakan bahasa pemrograman Node.Js dengan framework Express.Js dan basis data MongoDB. Penelitian ini berfokus pada pengembangan <i>back end</i> bagian <i>provider</i> dengan REST API menggunakan metode <i>Scrum</i> . Sistem ini berhasil dikembangkan dengan iterasi sprint sebanyak 4 kali

2.	Utin Kasma, Windy Agasia STMIK Pontianak 2017	Perancangan Sistem E-Grocery Pada Minimarket XYZ Pontianak Menggunakan Scrum Methodology	Merancang Sistem E- Grocery	PHP, Database : MySQL	<i>Scrum</i>	Penelitian ini menggunakan metode Agile yaitu dengan pendekatan Scrum Methodology. Pengembangan e-grocery diperlukan tiga tim pengembang yaitu, programmer, designer dan tester. Bahasa pemrograman yang digunakan adalah PHP dan database MySQL. Penelitian ini menghasilkan rancangan website e-grocery yang dapat memberikan kemudahan kepada konsumen dalam berbelanja.
3.	Fachry Maulana Prabowo Universitas Lampung Bandar Lampung 2019	Pengembangan Aplikasi Marketplace Pakan Ternak dan Produk Peternak (ePakan.id) Berbasis Android dengan Metode Scrum	Pengembang an Aplikasi Marketplace Pakan Ternak dan Produk Peternak	Android, Laravel API, JSON	<i>Scrum</i>	Epakan.id dikembangkan menggunakan metode scrum yang ditujukan bagi pengembangan aplikasi dengan skala menengah dan dikerjakan secara tim. Pada jurnal ini peneliti mengerjakan bagian android untuk ePakan. Sistem ini berhasil dikembangkan dengan iterasi sprint sebanyak 5 kali.

4.	Ricky Handoyo, Leo Willyanto Santoso, Alexander Setiawan Universitas Kristen Petra 2019	Real-Time BPMN Website Menggunakan Teknologi MERN Stack	Pembuatan Real-Time BPMN Website	MongoDB, Express, React, Node (MERN Stack)	-	Pembuatan website real-time bpmn dirancang dengan memanfaatkan MERN stack <i>technology</i> sebagai <i>single page application</i> dan <i>library javascript</i> yang dimanfaatkan dalam pembuatan website terutama <i>library</i> bpmn untuk pembuatan <i>diagram</i> dan socket untuk komunikasi dua arah secara real-time antara user dengan sistem. Sehingga user dapat berbagi project dan membuat diagram secara bersama-sama
5.	Annisa Tahira Sekolah Tinggi Teknologi Terpadu Nurul Fikri, 2019.	Pengembangan Aplikasi Web <i>E-Commerce</i> Sebagai Media Transaksi Jual-Beli Online (Studi Kasus: chili.id)	Web <i>E-Commerce</i> pada chilli.id	React.Js dan Node.Js Database server : <i>MongoDB</i>	<i>Scrum</i>	-

2.4 Aplikasi Terkait

Dalam penyusunan proposal tugas akhir ini, dilakukan penelitian terhadap *e-commerce* yang sejenis dimana menjual produk cabai pada websitenya. Berikut beberapa *e-commerce* yang ditemukan:

Tabel 2. Aplikasi Terkait

No	Nama Website	Kelebihan Web	Kekurangan Web
1	https://www.agromaret.com/	<ul style="list-style-type: none">• Menjual berbagai macam sayuran• Terdapat banyak penjual sayuran yang berbeda-beda	<ul style="list-style-type: none">• Tidak adanya kualitas kontrol dari <i>E-Commercenya</i> terhadap produk yang dijual
2	https://www.bukalapak.com	<ul style="list-style-type: none">• Menjual berbagai macam barang• Tersedia barang-barang yang berbeda yang dapat dibeli pembeli	<ul style="list-style-type: none">• Tidak adanya kualitas kontrol terhadap produk yang dijual
3	http://www.wildfirechilli.com.au/	<ul style="list-style-type: none">• Website khusus untuk jual-beli produk cabai	<ul style="list-style-type: none">• <i>Base market</i> di luar negeri
4	https://pepperworldhotshop.com/	<ul style="list-style-type: none">• Website khusus untuk jual-beli produk cabai	<ul style="list-style-type: none">• <i>Base market</i> di luar negeri
5	chili.id	<ul style="list-style-type: none">• Website khusus untuk jual-beli produk cabai	<ul style="list-style-type: none">• Sedang tahap pengembangan

2.5 Posisi Penelitian

Dalam penyusunan proposal tugas akhir ini, dilakukan penelitian terhadap *e-commerce* yang sejenis dimana menjual produk cabai pada websitenya. Berikut beberapa *e-commerce* yang ditemukan:

Tabel 3. Penelitian Terkait

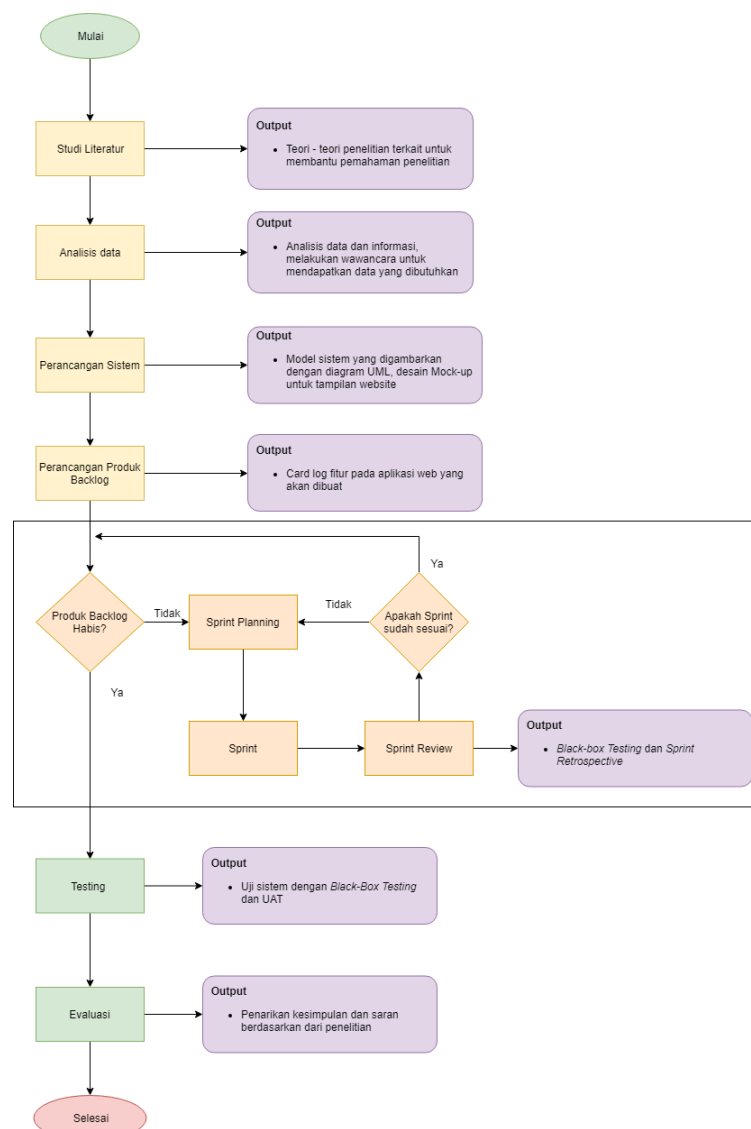
No	SCRUM	E-Commerce	Web	MongoDB	NodeJS	ReactJS
1	<u>Muh Mudrikul Falaq (2014)</u> Pengembangan Back End Bagian Provider pada Marketplace Travinesia.com dengan Rest API					
2	<u>Utin Kasma, Windy Agasia (2017)</u> Perancangan Sistem <i>E-Grocery</i> Pada Minimarket XYZ Pontianak Menggunakan Scrum Methodology					
3	<u>Fachry Maulana Prabowo (2019)</u> Pengembangan Aplikasi <i>Marketplace</i> Pakan Ternak dan Produk Peternak (ePakan.id) Berbasis Android Dengan Metode Scrum					
4			<u>Leo Willyanto Santoso, Alexander Setiawan (2019)</u> <i>Real-Time</i> BPMN Website Menggunakan Teknologi MERN Stack			
5	<u>Annisa Tahira (2019)</u> Pengembangan Aplikasi Web <i>E-Commerce</i> Sebagai Media Transaksi Jual-Beli Online (Studi Kasus: chili.id)					

BAB III

METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian yang digunakan untuk pengembangan aplikasi *e-commerce* ini dapat dijelaskan seperti alur pada gambar 2 yang terdiri dari studi literatur, analisis kebutuhan data, pengembangan sistem metode *Agile Development Scrum*, *testing* aplikasi serta evaluasi.



Gambar 2. Tahapan Penelitian

3.2 Rancangan Penelitian

3.2.1 Jenis Penelitian

Pengembangan sistem pada penelitian ini menggunakan model *agile development scrum*, sehingga akan berfokus pada pengembangan web dengan menggunakan tahapan pengembangan yang ada pada scrum. Data yang digunakan untuk pengembangan aplikasi web ini didapatkan dengan metodi studi pustaka serta wawancara.

3.2.2 Studi Pendahuluan

Pada tahapan ini dilakukan studi pendahuluan untuk mendapatkan informasi dan data acuan penelitian yang dapat dijadikan untuk merumuskan masalah, tujuan dan manfaat penelitian, serta penyusunan metode penelitian yang akan digunakan untuk penelitian ini.

3.2.3 Metode Penelitian

Metode penelitian yang digunakan adalah metode penelitian kualitatif. Data yang dibutuhkan untuk penelitian ini dilakukan dengan melakukan wawancara terkait informasi dan data yang dibutuhkan untuk keperluan penelitian ini dalam mengembangkan aplikasi web yang dibuat.

3.2.4 Metode Perancangan Sistem

Pada tahapan ini dilakukan perancangan sistem yang akan digambarkan dengan menggunakan diagram UML serta gambaran sistem berbentuk desain mock-up yang disesuaikan dengan kebutuhan sistem yang telah didapatkan dari tahapan sebelumnya.

3.2.5 Metode Pengembangan Software

Metode pengembangan sistem yang dilakukan dalam penelitian ini adalah metode *agile development scrum*. Sehingga dilakukan tahapan pengembangan software sesuai dengan tahapan yang ada pada scrum.

a) Perancangan Produk Backlog

Pada tahapan ini dilakukan perancangan fitur-fitur apa saja yang akan dikembangkan dalam proses pembuatan web. Fitur-fitur yang diperlukan akan di list dengan menggunakan *user story*.

b) Sprint Planning

Pada tahapan ini dilakukan perancangan jadwal iterasi/pengulangan dari sprint yang akan dilakukan. Selain itu akan ditentukan fitur-fitur utama yang dibutuhkan pada pengembangan web yang akan dibuat serta akan dilakukan pengurutan prioritas setiap fitur-fitur yang akan dibuat.

c) Sprint

Pada tahapan ini akan dilakukan pengembangan web sesuai dengan proses waktu yang telah dibuat pada *sprint planning*. Dalam tahapan ini mulai dilakukan kegiatan-kegiatan yang berhubungan dengan pengembangan web ini, mulai dari desain, proses coding dan *test* fitur yang telah dibuat.

d) Sprint Review

Pada tahapan ini dilakukan *review* dan evaluasi terhadap sprint yang telah dilakukan, seperti pengecekan setiap fitur yang telah dibuat untuk pengembangan web, *review* kinerja yang telah dilakukan, kendala-kendala yang terjadi, serta perencanaan untuk proses *sprint* selanjutnya.

3.2.6 Lingkungan Pengembangan

Penelitian yang dilakukan dalam pengembangan web ini dilaksanakan di Kampus B2 STT Terpadu Nurul Fikri yang berlokasi di Jalan Raya Lenteng Agung No.20, RT.4/RW.1, Srengseng Sawah, Kec. Jagakarsa, Kota Jakarta Selatan, Daerah Khusus Ibukota Jakarta 12640

3.2.7 Alat dan Bahan

Pengembangan web pada penelitian ini menggunakan hardware dan software dengan spesifikasi sebagai berikut:

- 1. Laptop dengan spesifikasi:**
 - a. Processor CORE i3 7th Gen
 - b. RAM 4GB
 - c. Kapasitas harddisk 500GB
 - d. Sistem Operasi Windows
 - e. System type 64 bit
- 2. Alat pengembangan yang akan digunakan**
 - a. Browser
 - b. Laragon
 - c. Microsoft Office
 - d. Visual Studio Code

3.2.8 Metode Pengujian

Pada tahapan ini akan dilakukan pengujian terhadap web yang telah dibuat dan menjawab perumusan masalah yang telah dibuat dengan melakukan pengujian sebagai berikut:

a) Pengujian Blackbox

Pengujian pada web dilakukan dengan menggunakan blackbox testing untuk pengujian fungsionalitas pada web, seperti detail tampilan aplikasi, fungsi-fungsi yang ada pada aplikasi dan kesesuaian alur fungsi dengan bisnis proses yang telah ditetapkan.

b) Pengujian UAT

Metode pengujian ini menggunakan metode UAT untuk menguji kepuasan *user* terhadap web yang telah dibuat.

3.2.9 Evaluasi

Pada tahapan ini dilakukan kegiatan evaluasi untuk mengetahui apakah penelitian yang telah dilakukan mampu menjawab permasalahan yang telah ditentukan pada rumusan masalah yang telah ditentukan.

BAB VI

ANALISIS DAN PERANCANGAN SISTEM

Pada bab ini diuraikan mengenai analisis dan perancangan sistem pada *E-Commerce* chili.id. Penulis akan menguraikan analisis *requirement*, perancangan sistem yang digambarkan dengan menggunakan UML yaitu *use case*, *Activity Diagram*, dan *domain model*. Selain itu akan digambarkan juga antarmuka dari aplikasi web yang akan dibangun.

4.1 Analisis Sistem

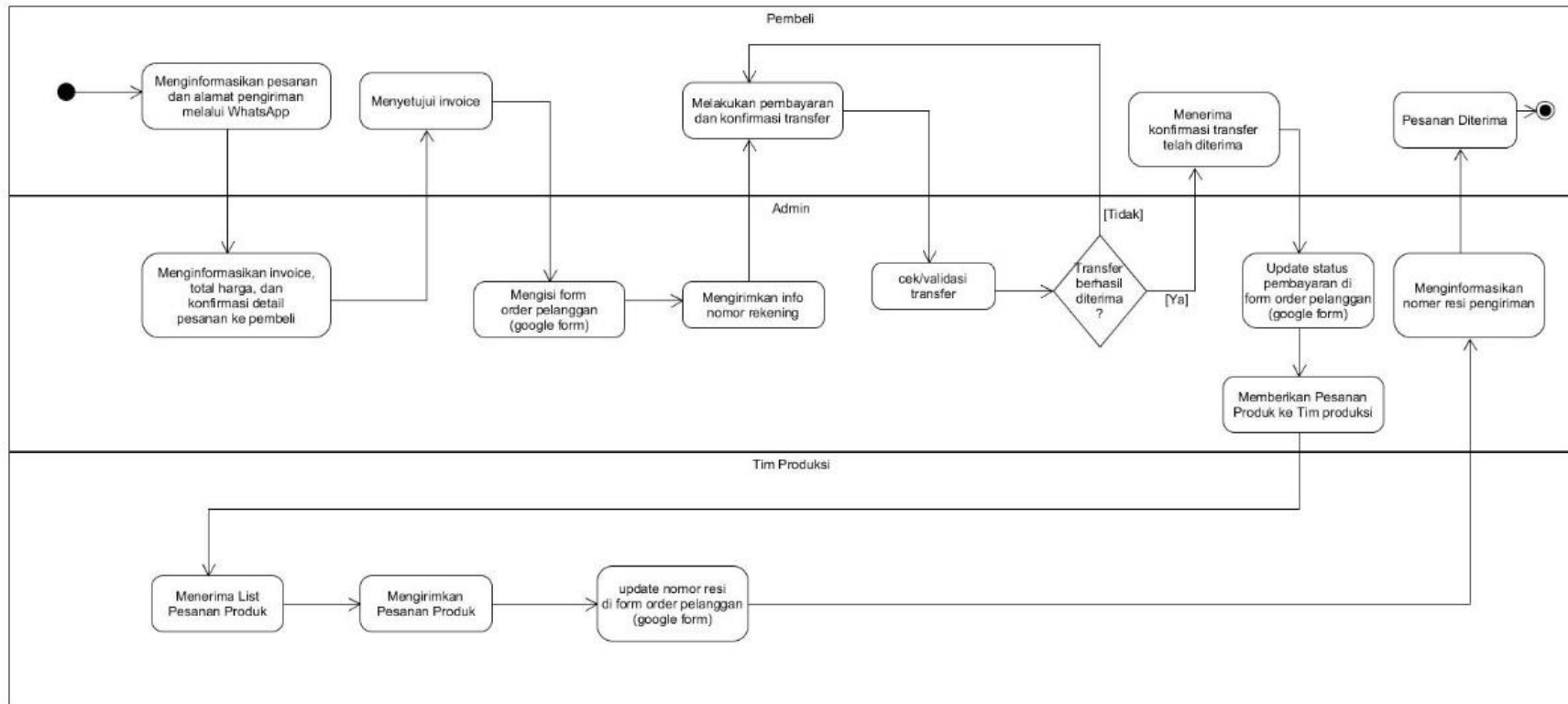
4.1.1 Analisis Sistem Berjalan

Penulis melakukan wawancara dengan Bapak Reza Aldiansyah, selaku Project and Product Manager startup Chilli Indonesia. Tujuan dari wawancara ini adalah untuk mendapatkan informasi yang dibutuhkan untuk mengumpulkan *user requirement* dan gambaran perancangan sistem *e-commerce* pada chilli.id.

Berdasarkan hasil wawancara yang telah dilakukan, maka didapatkan beberapa informasi terkait sistem yang berjalan saat ini. Informasi-informasi tersebut akan digunakan dalam perancangan sistem *e-commerce* chili.id pada penelitian ini.

Sistem yang telah berjalan sekarang ini dilakukan dengan cara manual atau kegiatan transaksi melalui whatsapp. Pembeli menginformasikan pesanan produk dan alamat pengiriman melalui whatsapp jika ingin memesan produk. Setelah produk dipesan, maka admin akan memberikan *invoice* dan total pembayaran sebagai tanda jadi pembelian produk. Kemudian admin akan mencatat pesanan di *form order* pelanggan yang tersedia pada google *form* dan mengecek transaksi yang dilakukan oleh pembeli.

Admin akan meminta bagian produksi untuk mengumpulkan produk-produk sesuai dengan pemesanan pembeli. Setelah produk-produk terpenuhi maka selanjutnya produk akan dikirimkan ke pembeli melalui kurir yang bekerjasama dengan chili.id. Selanjutnya admin akan memberikan nomer resi kepada pembeli untuk mengecek pemesanannya.



Gambar 3. Alur Diagram Sistem Berjalan

a) Kelemahan Sistem Berjalan

Berdasarkan hasil analisis yang didapatkan dari chili.id, penulis dapat menguraikan beberapa kelemahan pada sistem yang berjalan saat ini, diantaranya:

Tabel 4. Analisis PIECES

Analisis PIECES	Sistem yang sedang berjalan
Kinerja (<i>Performance</i>)	Kinerja dari sistem jual-beli yang ada di chili.id masih kurang, karena melakukan kegiatan transaksi melalui whatsapp. Cara tersebut kurang begitu efektif ketika proses jual-beli transaksi ingin dilakukan dengan cepat.
Informasi (<i>Information</i>)	Sistem yang ada saat ini walaupun sudah menggunakan media promosi seperti whatsapp, facebook, instagram dan media sosial tim yang terlibat, masih dirasa kurang dalam penyebaran informasinya. Hal ini dikarenakan belum adanya platform khusus yang menyediakan informasi detail produk-produk yang ada pada chili.id sehingga masih banyak masyarakat yang tidak mengetahui chili.id
Analisis Ekonomi (<i>Economic</i>)	Penerapan sistem baru yang diusulkan diharapkan dapat menekan biaya-biaya yang dikeluarkan untuk media promosi dan bisa menambah pemasukan biaya dari sistem <i>E-Commerce</i> yang dibuat dengan membuka jasa iklan pada website. Selain itu, jika ada perubahan informasi produk, maka informasi akan mudah diubah tanpa membutuhkan waktu dan biaya produksi. Jadi sistem baru yang diusulkan akan lebih ekonomis dan daripada sistem lama.

Keamanan atau kontrol (<i>Security or Control</i>)	Kurang maksimalnya kontrol terhadap data produk dan juga data pembelian, karena saat ini data masih disimpan pada <i>google form</i> .
Efisiensi (<i>Efficiency</i>)	Admin harus melakukan pengisian form <i>Order</i> setiap ada pembelian produk sehingga akan membutuhkan waktu yang lebih dalam proses pencatatan transaksi. Penyampaian informasi juga tidak bisa diakses setiap saat. Sehingga pemanfaatan sumber daya yang tersedia seperti sumber daya operasional dan teknologi masih kurang.
Layanan (<i>Service</i>)	Pelayanan yang diberikan kepada konsumen menjadi kurang maksimal, karena keterbatasan media promosi menyebabkan proses pelayanan hanya sebatas kepada orang-orang yang mengikuti media sosial yang terkait produk-produk chili.id atau tempat dimana produk di jual. Selain itu, apabila ada pemesanan produk dalam jumlah banyak dalam waktu yang bersamaan, maka pelayanan sering tidak maksimal.

4.1.1 Analisis Kebutuhan Sistem dan User

Kelemahan pada sistem yang berjalan mengakibatkan proses penyebaran informasi produk kurang maksimal karena saat ini belum tersedianya website khusus yang menyediakan semua informasi terkait produk-produk yang ada pada chili.id. Selain itu proses transaksi yang ada saat ini juga kurang efektif karena dilakukan secara manual melalui whatsapp. Berdasarkan permasalahan diatas maka diperlukan sebuah sistem yang dapat membantu mempermudah pemesanan produk yang dilakukan oleh pembeli sehingga dapat memaksimalkan kinerja chili.id dalam memasarkan produk kepada pembeli atau calon pembeli.

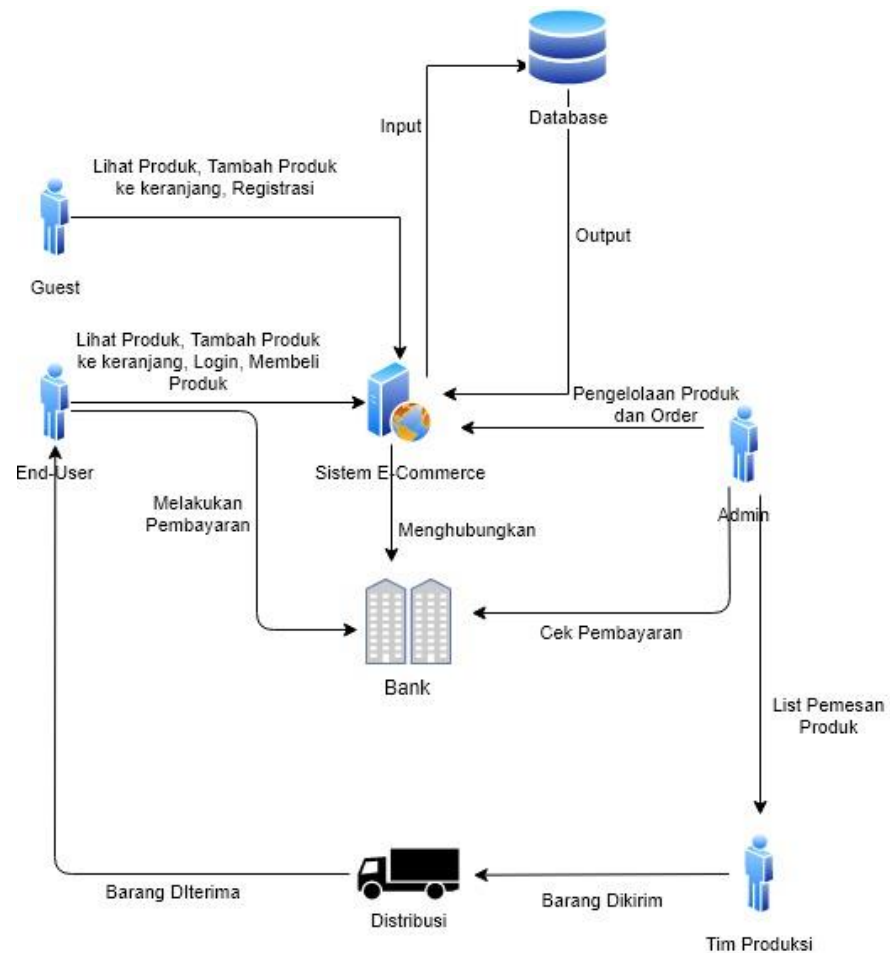
Kebutuhan *user* yang berkaitan dengan sistem dan semua kegiatan yang dapat dilakukan *user* pada sistem *e-commerce* ini akan dijelaskan melalui 3 *user* berikut, antara lain:

Tabel 5. *User Deskripsi*

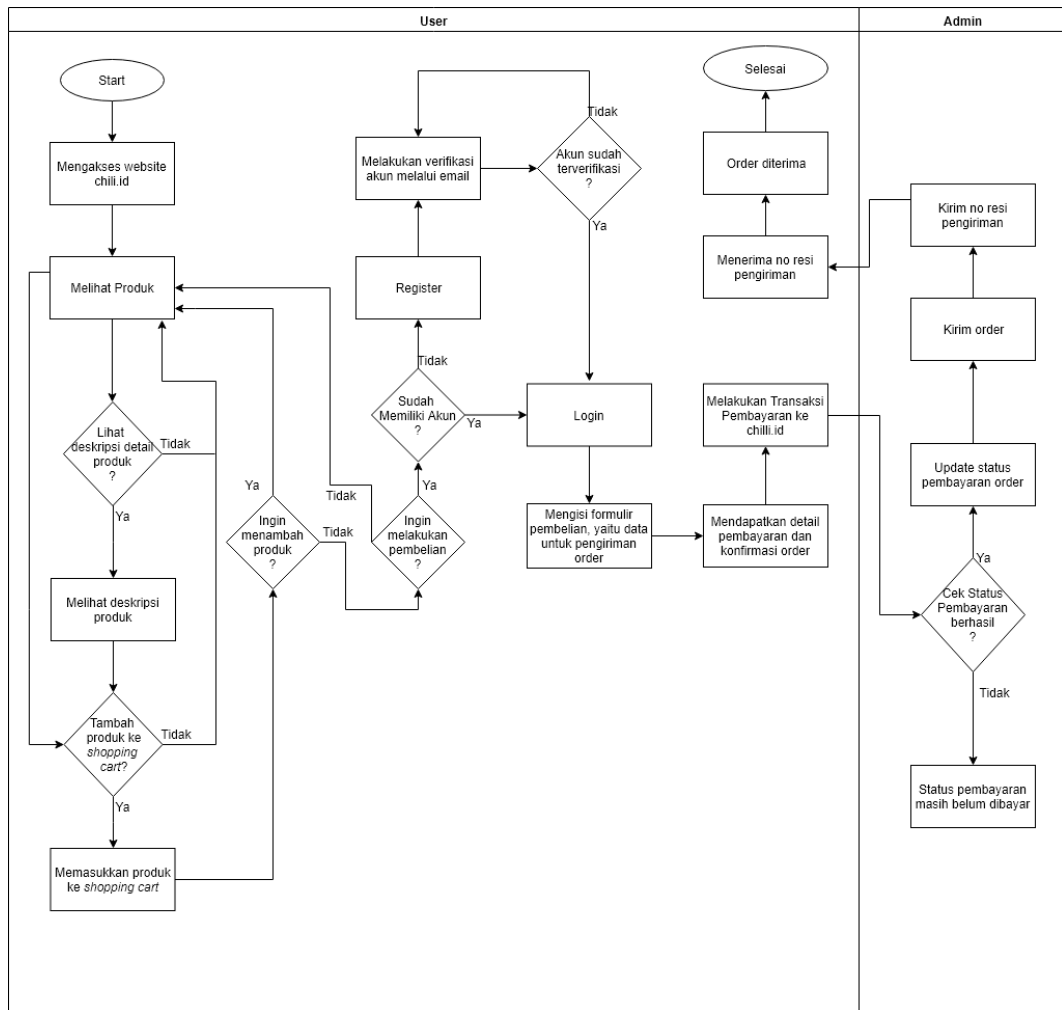
<i>User</i>	Deskripsi
<i>Guest</i>	<i>User</i> yang belum <i>login</i> ke sistem
<i>Registered-user</i>	<i>User</i> yang sudah <i>login</i> ke sistem
<i>User</i>	<i>Guest</i> atau <i>registered-user</i> pada aplikasi chili.id
Admin	<i>User</i> yang mengelola produk pada website

Pada kategori *user* aplikasi dijelaskan mengenai hak-hak akses yang dimiliki oleh setiap *user*. Kategori *user* dibedakan menjadi tiga yaitu *guest*, *registered-user* dan admin. *User Guest* hanya memiliki menu melihat produk, *register*, *login* dan *cart*. *Registered-user* yang telah *login* ke sistem akan memiliki *dashboard user* untuk keperluan pembelian dan pembayaran produk. Admin memiliki *dashboard* sendiri yang bertujuan untuk mengelola produk dan data *user* pada website chilli.id. Sistem *e-commerce* pada chili.id akan digambarkan pada

gambar 4 dan diagram alur dari sistem *e-commerce* chilli.id akan digambarkan pada gambar 5.



Gambar 4. Rich Picture Sistem yang Diusulkan



Gambar 5. Diagram Alur Sistem *E-Commerce* yang Diusulkan

a) Istilah Baku

Pada aplikasi web yang dirancang menggunakan beberapa istilah yang akan dipakai dalam web. Berikut istilah-istilah yang digunakan pada pengembangan aplikasi web.

Tabel 6. Istilah Baku

No.	Istilah	Arti
1.	Produk	Benda yang dijual pada aplikasi web
2.	<i>Login</i>	Masuk ke aplikasi web menggunakan akun yang sudah didaftarkan sebelumnya
3.	<i>Register</i>	Membuat atau mendaftar akun baru di aplikasi web
4.	<i>Shopping Cart</i>	Wadah yang memungkinkan pelanggan belanja online untuk mengumpulkan daftar item untuk pembelian
5.	<i>Shipping</i>	Kurir patner untuk mengirimkan pesanan ke alamat tujuan
6.	<i>Payment Order</i>	Metode pembayaran pesanan yang dilakukan <i>user</i>
7.	<i>Order</i>	Pemesanan produk yang dilakukan <i>user</i>
8.	<i>Customer</i>	<i>User</i> yang melakukan pembelian di aplikasi web

b) User Story

User story merupakan deskripsi mengenai kebutuhan sistem dalam bentuk bahasa natural yang dapat dengan mudah dipahami oleh *registered-user* yang tidak memiliki background TI. *User story* ini didapatkan dari hasil wawancara yang telah dilakukan dengan Bapak Reza Aldiansyah, selaku *Project and Product Manager startup* Chilli Indonesia.

Tabel 7. User Story

<i>User Story ID</i>	<i>As a <type of user></i>	<i>I want to <perform some task></i>	<i>so that I can <achieve some goal></i>
R.001	<i>Guest</i>	Melihat produk	Memastikan apa yang saya cari ada di platform ini
R.003	<i>Guest</i>	Melakukan pencarian produk	Memastikan apa yang saya cari ada di platform ini
R.004	<i>Guest</i>	Menambah produk ke <i>Shopping Cart</i>	Mendapatkan list produk yang ingin dibeli
R.005	<i>Guest</i>	Melakukan <i>Register</i>	Menjadi <i>user</i> terdaftar pada platform
R.006	<i>Guest</i>	<i>Login</i> sebagai <i>Registered-user</i>	Melakukan pembelian produk yang ada pada platform ini
R.007	<i>Registered-user</i>	Membeli produk	Membeli dan melakukan pembayaran produk
R.010	<i>Registered-user</i>	Memperbarui halaman profil <i>user</i>	Melakukan perubahan identitas diri pada halaman profile <i>user</i>
R.011	<i>Registered-user</i>	Mendapatkan informasi <i>Order</i> produk cabai	Mendapatkan informasi sedetailnya terkait produk yang dipilih

R.012	<i>Registered-user</i>	Melakukan konfirmasi pembayaran	Memberikan bukti bahwa telah melakukan pembayran terhadap produk yang dipesan
R.013	<i>Registered-user</i>	Memberikan komentar	mendapatkan kualitas pelayanan dan produk yang semakin baik
R.014	Admin	Menambah produk	Menambahkan produk baru ke <i>platfrom</i>
R.017	Admin	Melakukan update produk (<i>edit/delete</i>)	Melakukan perubahan data produk sesuai dengan kebutuhan
R.018	Admin	Melakukan konfirmasi penerimaan pembayaran dan update status pembayaran <i>Order</i>	Memberikan konfirmasi bahwa produk telah dibayar oleh <i>Customer</i> sehingga bisa mengirimkan pesanan ke alamat <i>user</i> dan memberikan nomer resi pengiriman ke <i>user</i> melalui email
R.019	Admin	Melihat informasi <i>Order Customer</i>	Melihat penjualan yang terjadi

c) Product Backlog

Pada tahapan ini dilakukan perancangan fitur-fitur apa saja yang akan dikembangkan dalam proses pembuatan web.

Tabel 8. *Product Backlog*

Sprint	User Story ID	Requirement	Bobot
1	R.001	Melihat produk	5
1	R.003	Melakukan pencarian produk	8
1	R.014	Menambah produk	8
2	R.004	Menambah produk ke <i>Shopping Cart</i>	8
2	R.005	Melakukan <i>Register</i>	3
2	R.006	<i>Login</i> sebagai <i>Registered-user</i>	3
2	R.017	Melakukan update produk (<i>edit/delete</i>)	13
3	R.010	Memperbarui halaman profil <i>user</i>	13
3	R.007	Membeli produk	13
3	R.011	Mendapatkan informasi <i>Order</i> produk cabai	8
3	R.013	Memberikan komentar	5
4	R.012	Melakukan konfirmasi pembayaran	21
4	R.018	Melakukan konfirmasi penerimaan pembayaran dan update status pembayaran <i>Order</i>	21
4	R.019	Melihat informasi <i>Order Customer</i>	21

Tabel 9. Deskripsi Bobot *Product Backlog*

Bobot	Deskripsi
3	Sangat mudah
5	Mudah
8	Sedang
13	Sulit
21	Sangat Sulit

Pada setiap Sprint pada tiap-tiap fitur yang dilakukan atau yang akan dikerjakan memiliki bobot penilaian masing-masing. Nilai bobot tersebut diambil berdasarkan tingkat kesulitan dan durasi pengerjaan dari masing-masing fitur yang ada.

4.1.2 Analisis Perbandingan Sistem

Penulis melakukan analisis perbandingan sistem untuk membandingkan kekurangan yang ada di sistem yang berjalan dengan kelebihan sistem yang akan diusulkan.

Tabel 10. Perbandingan Sistem

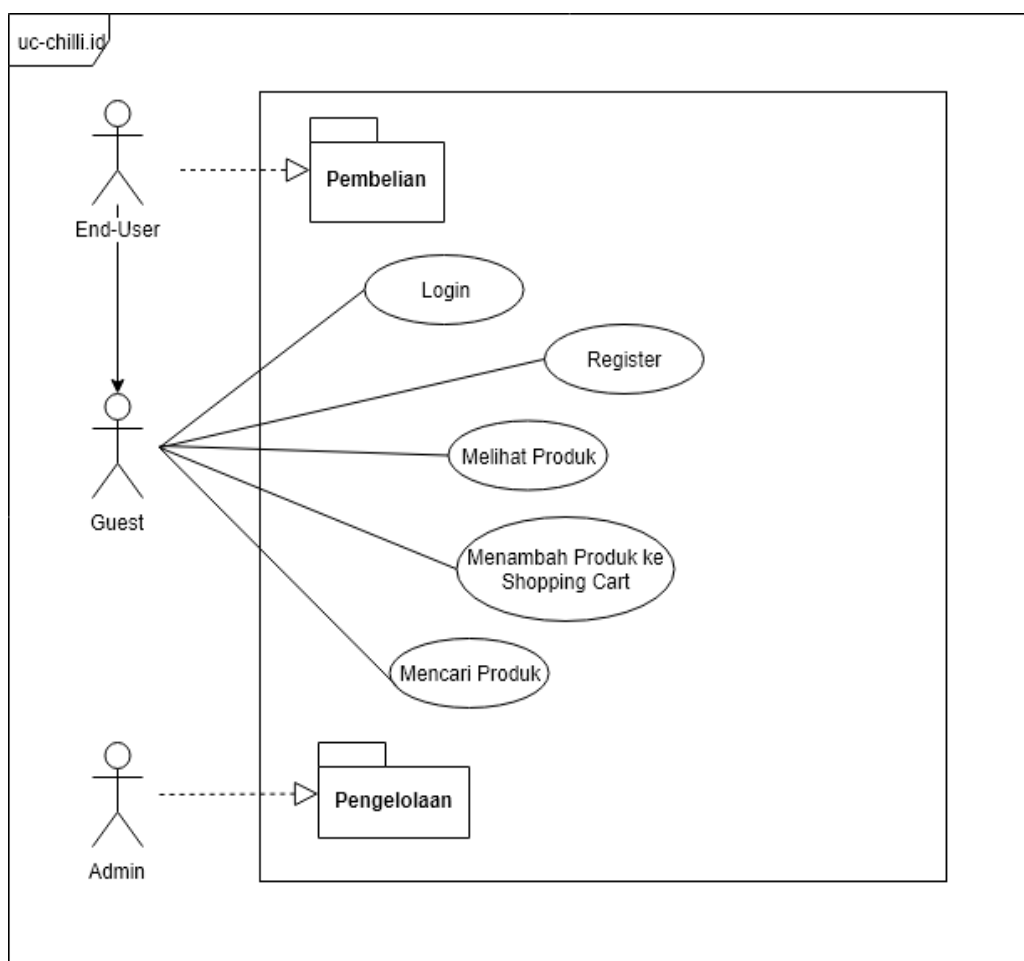
No.	Sistem Yang Berjalan	Sistem Yang Diusulkan
1.	Belum dapat melakukan pemesanan secara langsung pada website	Sistem dapat melakukan pemesanan produk secara langsung dengan dilengkapi informasi yang lengkap
2.	Belum dapat memberikan informasi produk langsung ke pembeli atau calon pembeli	Sistem mampu memberikan informasi secara langsung ke pembeli melalui website

4.1.3 Use Case

Pada bagian ini akan dijelaskan tentang *use case* dari *e-commerce chilli.id* yang didasarkan pada hasil analisis yang telah dilakukan.

4.1.3.1 Use Case Diagram

Gambar dibawah ini menggambarkan tentang *use case diagram* dari *e-commerce chilli.id*



Gambar 6. Use Case Diagram

Dari gambar di atas terdapat tiga kategori *user* yaitu *guest*, *registered-user* dan *admin*. *Guest* hanya bisa melihat produk, menambahkan produk ke *shopping cart*, mencari produk, *register* dan *login*. Sedangkan untuk menggunakan fitur lain maka *guest* harus melakukan *register* apabila belum mempunyai akun dan

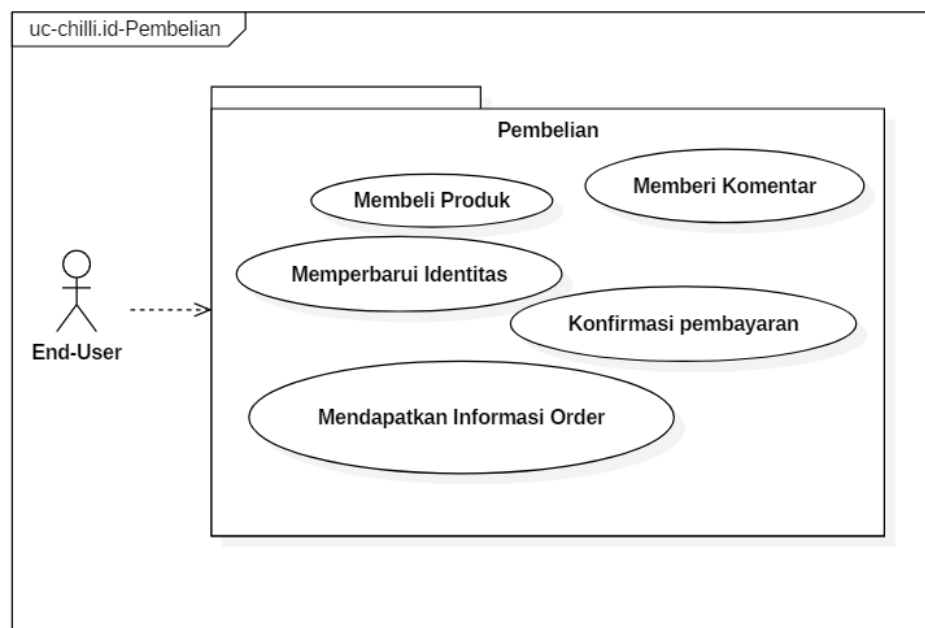
kemudian *login* terlebih dahulu agar bisa menggunakan fitur-fitur yang berhubungan dengan pembelian produk. Selain itu juga ada admin yang dapat melakukan pengelolaan terhadap data *user* dan produk-produk yang ada pada website.

4.1.3.2 Package Diagram

Gambar di bawah ini merupakan *package diagram* (diagram paket) yang digunakan untuk mengelompokkan elemen-elemen model dari *use case diagram* yang telah dibuat sebelumnya.

a. Package Diagram Pembelian

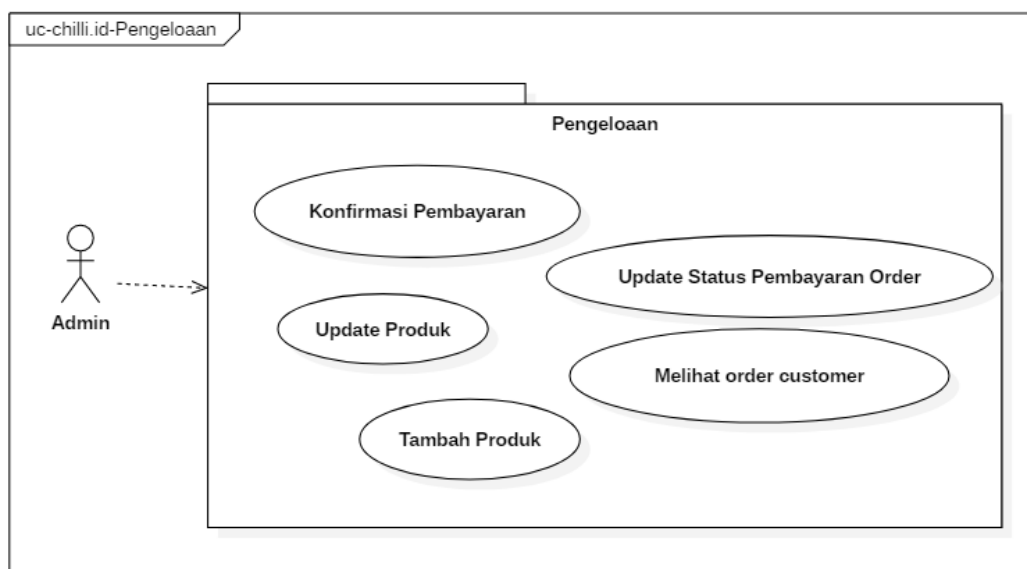
Pada *package diagram* pembelian menggambarkan aktivitas yang dapat dilakukan oleh *registered-user*, dimana *registered-user* dapat menggunakan fitur-fitur yang berhubungan dengan pembelian produk diantaranya membeli produk, melakukan konfirmasi pembayaran, mendapatkan informasi *order*, memperbarui identitas untuk keperluan pengiriman *order* dan memberikan komentar terhadap produk yang sudah dibeli.



Gambar 7. *Package Diagram Pembelian*

b. *Package Diagram* Pengelolaan

Pada *package diagram* pengelolaan menggambarkan aktivitas yang dapat dilakukan oleh admin, dimana admin dapat menggunakan fitur-fitur yang berhubungan dengan pengelolaan produk dan data *order user* diantaranya menambahkan produk, melakukan update produk yaitu edit atau delete produk, konfirmasi pembayaran, mengubah status pembayaran *order* dan melihat informasi *order* pembeli

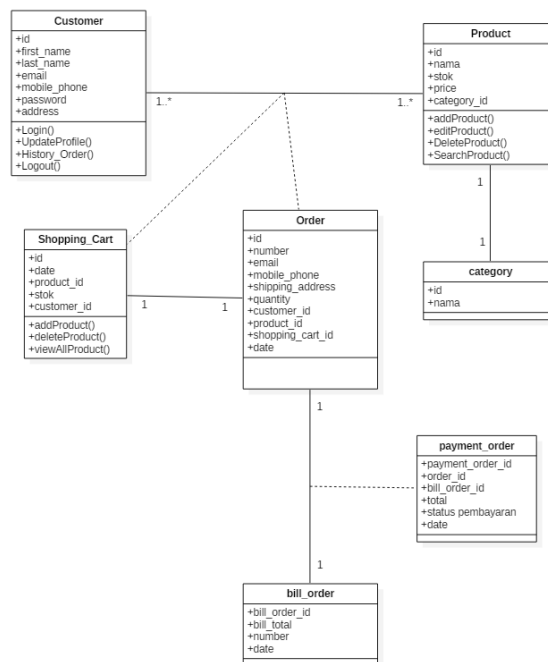


Gambar 8. *Package Diagram* Pengelolaan

4.2 Perancangan Sistem

4.2.1 Domain Model

Pada bagian ini akan digambarkan relasi atau cara kerja pembelian produk pada *e-commerce* chili.id. Database yang digunakan adalah MongoDB, sehingga relasi antar document menggunakan objectId untuk referensi antara satu dokumen ke dokumen lainnya. Proses yang terjadi melibatkan tujuh dokumen, dokumen-dokumen tersebut adalah dokumen *customer* yang berfungsi menampung data-data *user-user* yang terdaftar pada sistem, dokumen *product* yang berfungsi menampung data-data penambahan produk untuk halaman penjualan ataupun pembelian, dokumen kategori yang berfungsi untuk menampung data-data kategori produk cabai tabel, *shopping cart* yang berfungsi untuk me-record data produk yang ingin dibeli oleh *customer*. Kemudian ada tiga dokumen lainnya yang berhubungan dengan proses pembelian yakni tabel *order* yang berfungsi untuk menyimpan data-data transaksi pemesanan produk yang dilakukan *customer*, dokumen *bill order* untuk menyimpan data tagihan pembayaran *customer*, dan *order* yang berfungsi untuk menyimpan data-data pembayaran yang sudah dilakukan oleh *user*.



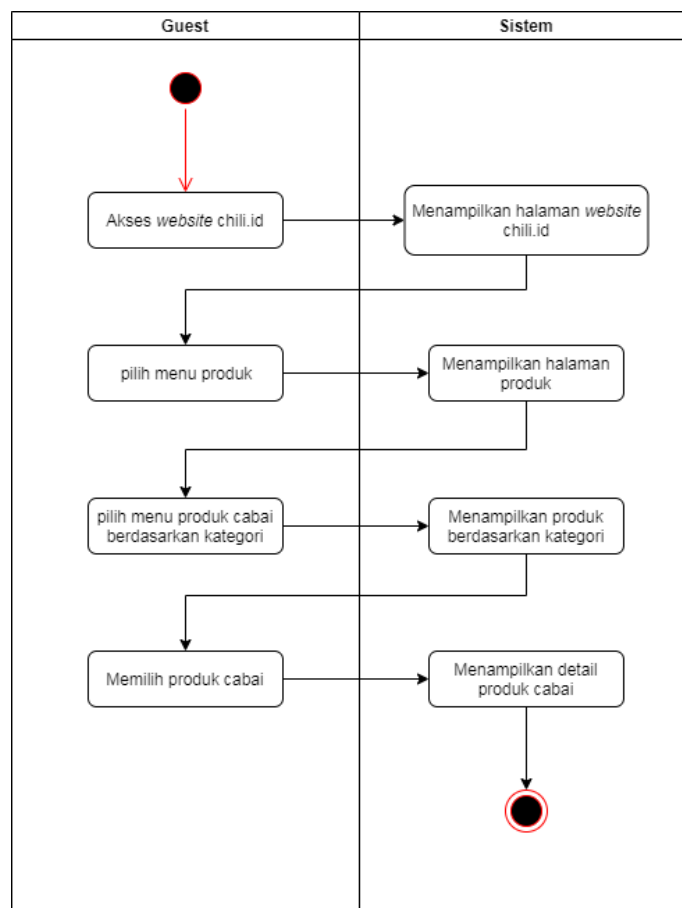
Gambar 9. Domain Model

4.2.2 Activity Diagram

Activity Diagram merupakan bagian dari UML yang menggambarkan aktivitas proses yang berjalan pada sistem.

a) *Activity Diagram* Melihat produk

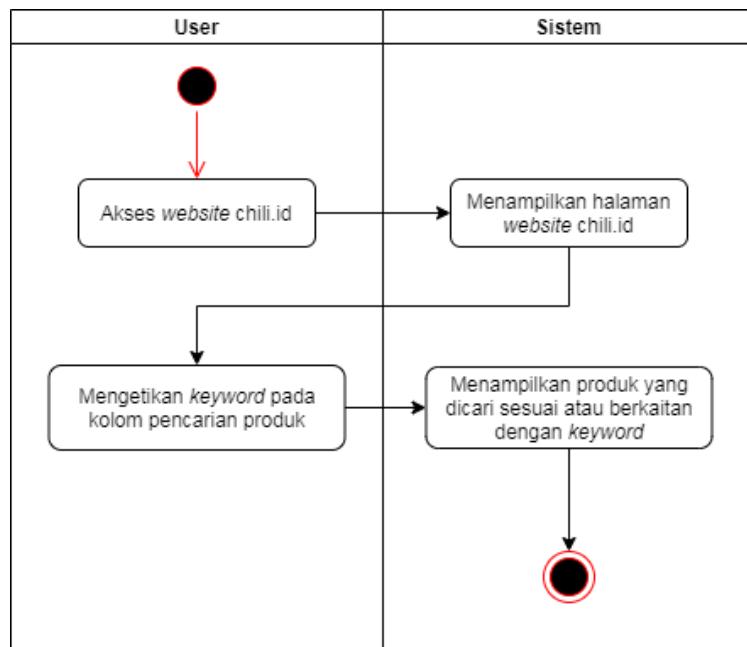
Activity diagram melihat produk menggambarkan proses melihat produk-produk yang terdapat pada *e-commerce* chili.id. *guest* atau *registered-user* mengakses website chili.id kemudian sistem akan menampilkan produk-produk tertentu pada halaman awal chili.id. *Guest* atau *registered-user* dapat memilih produk berdasarkan kategori produk, kemudian memilih produk yang diinginkan dan akan menampilkan detail dari produk yang dipilih.



Gambar 10. *Activity Diagram* Melihat Produk

b) Activity Diagram Pencarian Produk

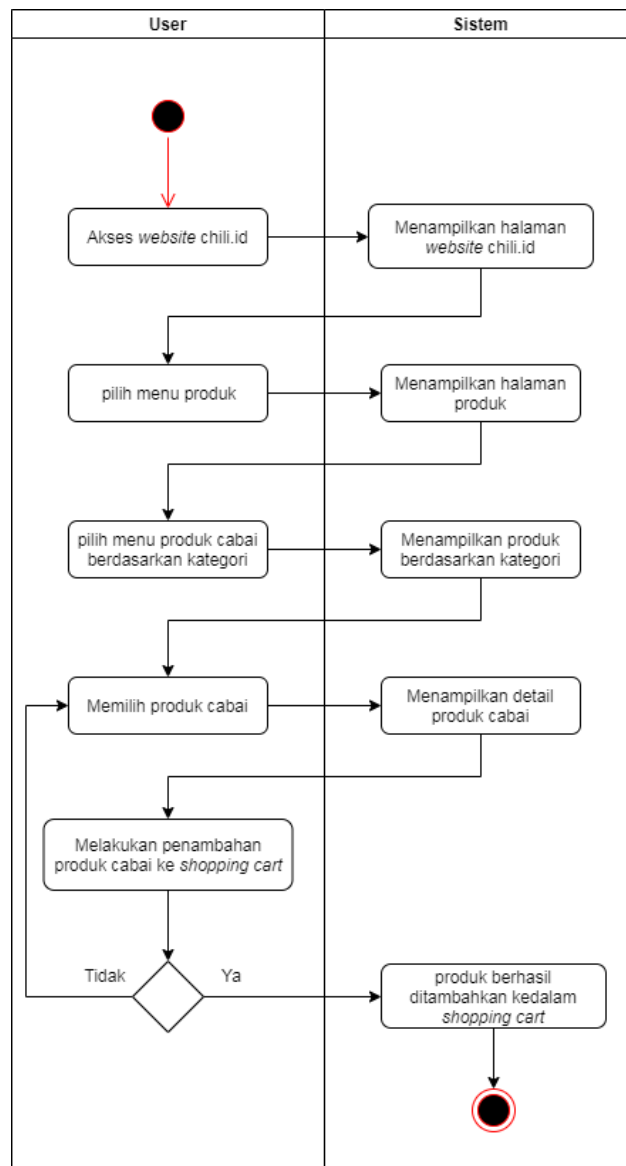
Activity Diagram pencarian produk menggambarkan proses bagaimana melakukan pencarian produk yang terdapat pada *e-commerce* chilli.id. *User* mengakses website chilli.id kemudian sistem akan menampilkan produk-produk tertentu pada halaman awal chilli.id. *User* dapat mengetikkan *keyword* atau kata kunci pencarian pada kolom pencarian produk yang terdapat pada website chilli.id. Kemudian sistem akan melakukan pencarian produk berdasarkan kata kunci yang dimasukkan.



Gambar 11. *Activity Diagram* Pencarian Produk

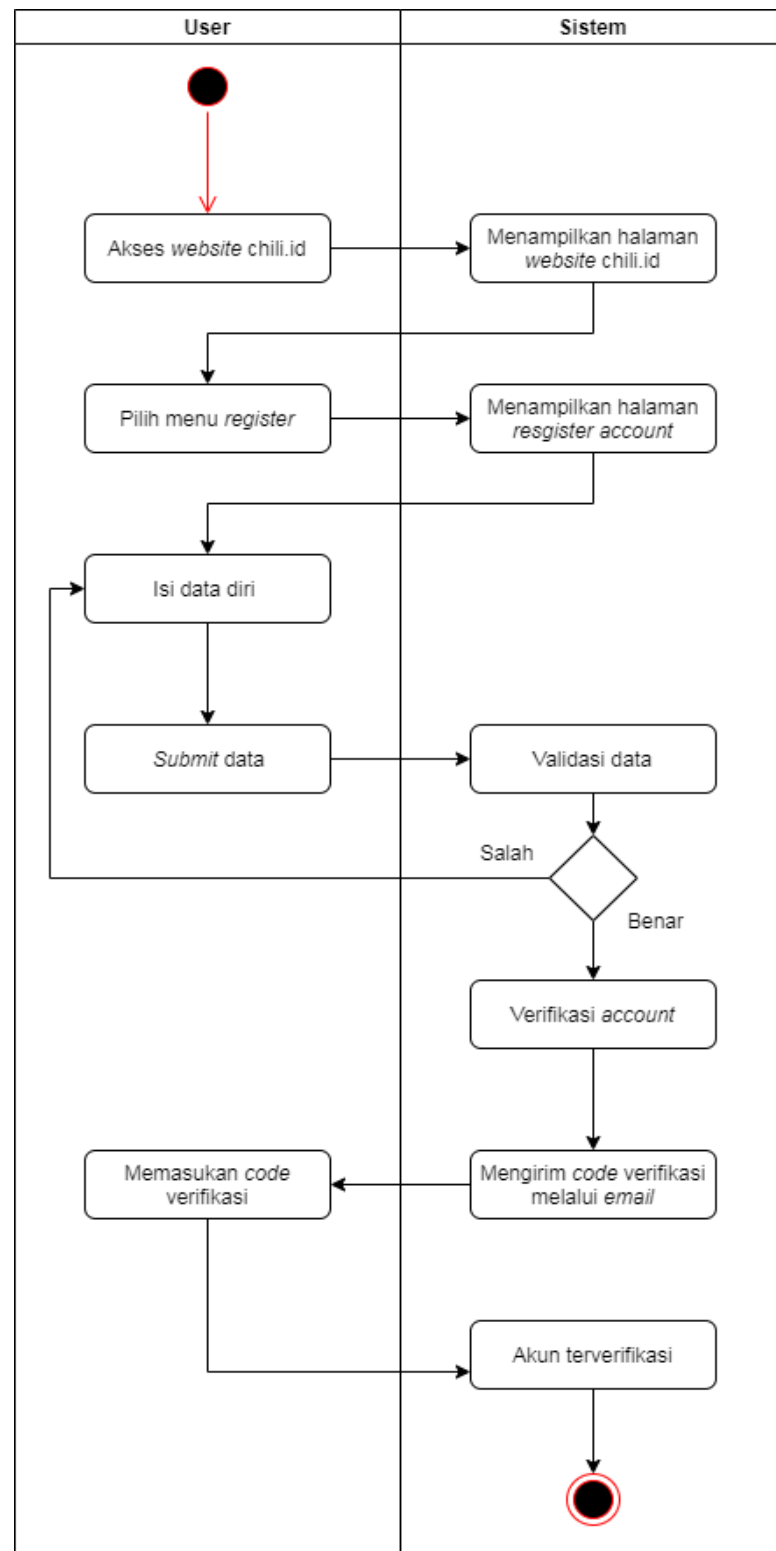
c) **Activity Diagram Menambah Produk ke Shopping Cart**

Activity diagram menambah produk ke *shopping cart* menggambarkan proses bagaimana *user* melakukan penambahan produk yang terdapat pada *e-commerce chilli.id*. *User* mengakses website *chilli.id* kemudian sistem akan menampilkan produk-produk tertentu pada halaman awal *chilli.id*. *user* memilih produk-produk apa saja yang ingin ditambahkan pada *shopping cart*. Kemudian secara otomatis *shopping cart* *user* akan terupdate sesuai dengan produk-produk yang dipilih dan ditambahkan oleh *user*.



Gambar 12. Activity Diagram Menambah Produk ke Shopping Cart

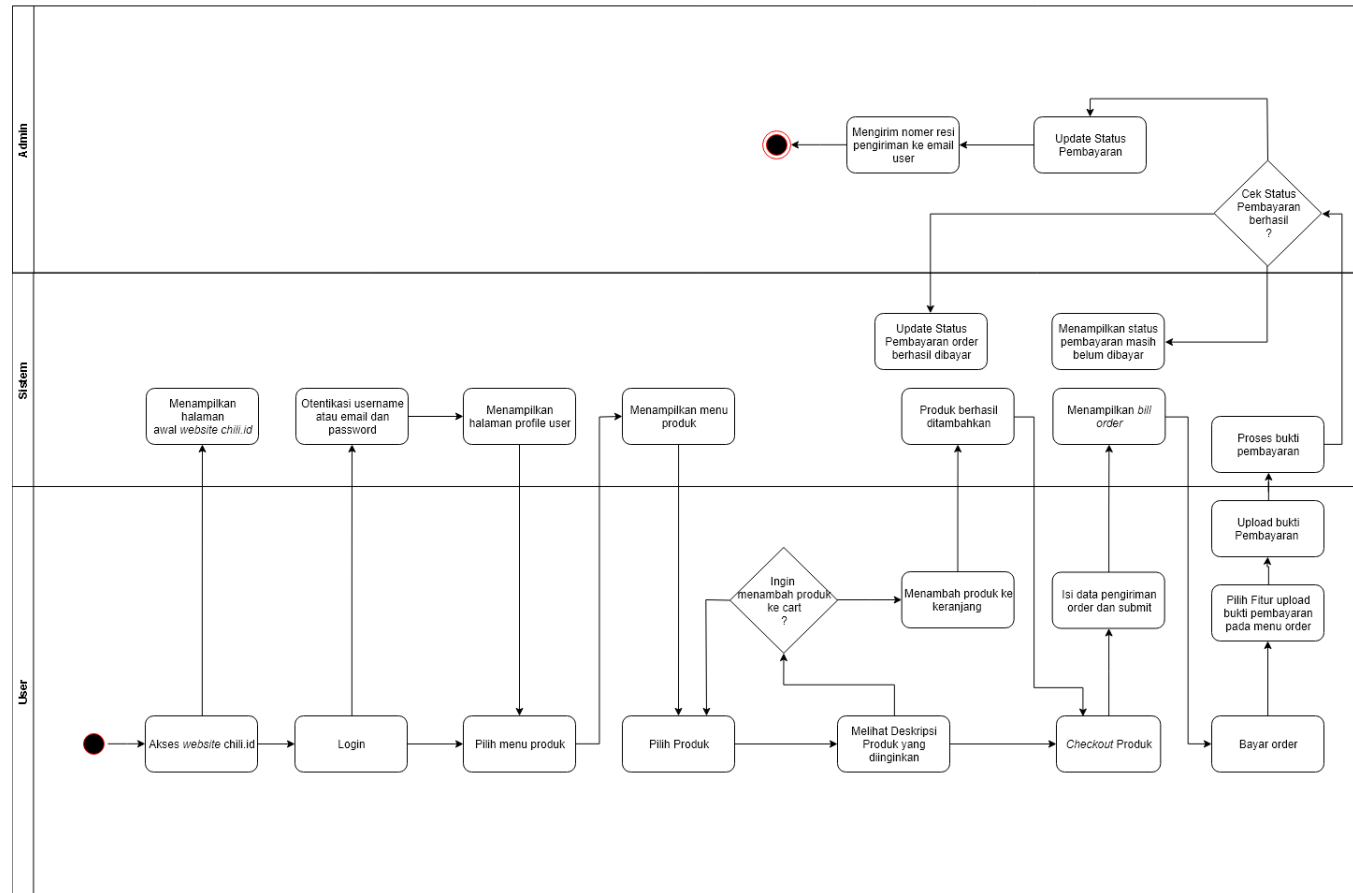
d) *Activity Diagram Register*



Gambar 13. *Activity Diagram Register*

Activity diagram register pada gambar 13 menggambarkan proses bagaimana *Guest* mendaftar sebagai *user* terdaftar atau yang disebut *registered-user* pada *e-commerce chili.id*. *User* mengakses website *chili.id* kemudian sistem akan menampilkan produk-produk tertentu pada halaman awal *chili.id*. *User* memilih menu *Register* terlebih dulu kemudian sistem akan menampilkan halaman *register account* dan menampilkan form data diri yang harus diisi oleh *user*. *User* mengisi data diri kemudian sistem akan memvalidasi data yang *user* masukkan apakah sudah terdaftar atau belum. Jika berhasil maka sistem akan mengirimkan kode verifikasi ke email *user*. *User* melakukan aktivasi akun dan akun *user* baru berhasil dibuat.

e) **Activity Diagram Pembelian produk**

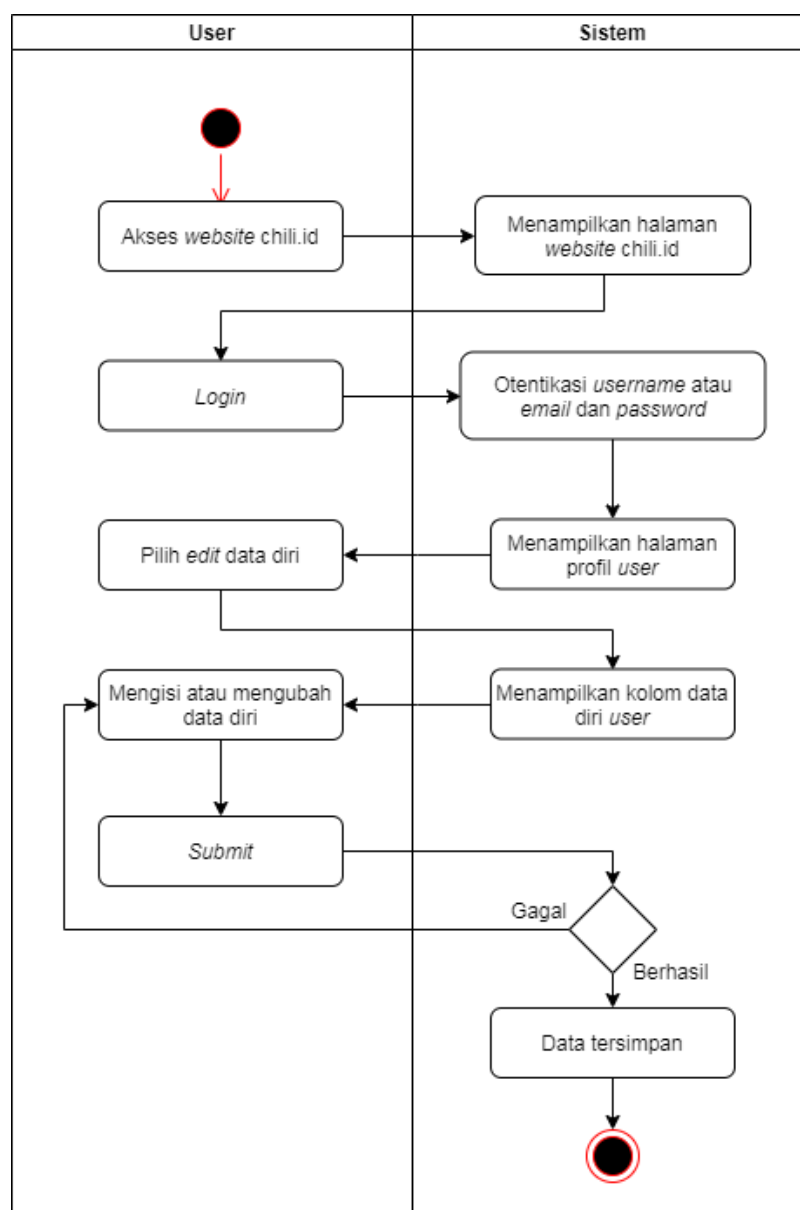


Gambar 14. Activity Diagram Membeli Produk

Activity Diagram pembelian produk pada gambar 14. menggambarkan proses bagaimana *registered-user* melakukan pembelian produk pada *e-commerce* chili.id. *User* diharuskan *Login* terlebih dahulu apabila ingin melakukan pembelian produk. Setelah *login*, *user* dapat melakukan checkout pada produk yang ada pada *shopping cart* ataupun langsung membeli produk tersebut. Kemudian setelah checkout sistem akan menampilkan halaman *order* dan *user* diharuskan mengisi data untuk keperluan pengiriman produk. Setelah isi data sistem akan menampilkan invoice produk dan total harga (termasuk ongkir) yang harus dibayar oleh *registered-user*. *Registered-user* melakukan pembayaran dan mengupload bukti pembayaran. Kemudian sistem akan memproses dan akan di verifikasi oleh admin. Apabila berhasil maka sistem akan menampilkan status pembayaran berhasil dan produk akan dikirimkan ke alamat *customer*. Kemudian admin memberikan nomer resi pengiriman kepada *customer* untuk dicek pengiriman orderannya.

f) Activity Diagram Memperbarui Halaman Profil User

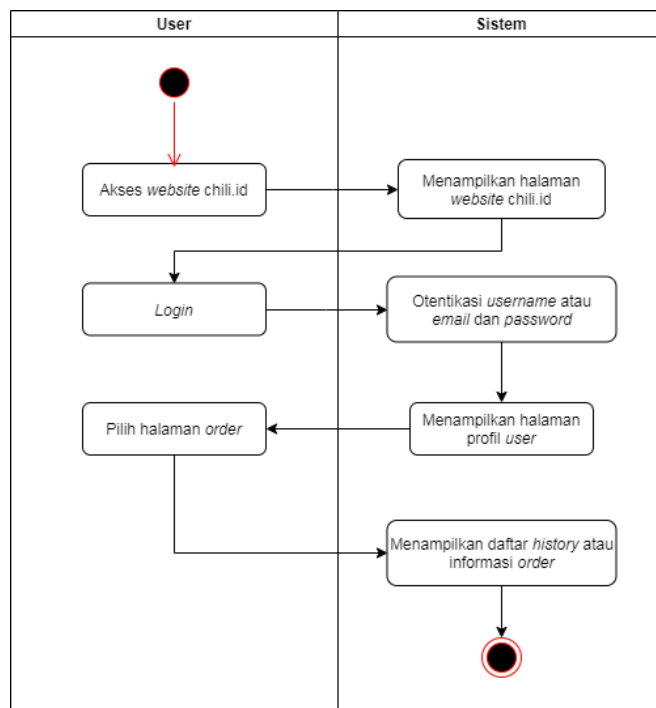
Activity diagram mengisi identitas diri menggambarkan proses *Registered-user* melakukan update pada data identitas diri *user*. *User* mengakses website chili.id kemudian *user* melakukan *login* dan diarahkan ke halaman profile *user*. *User* kemudian memilih fitur edit identitas diri dan kemudian mengisi kolom-kolom yang ingin dilakukan perubahan identitas diri. *User* lalu mengklik tombol submit untuk menyimpan data diri identitas yang baru pada akun di chili.id



Gambar 15. Activity Diagram Memperbarui Halaman Profil User

g) Activity Diagram Mendapatkan Informasi Order Produk Cabai

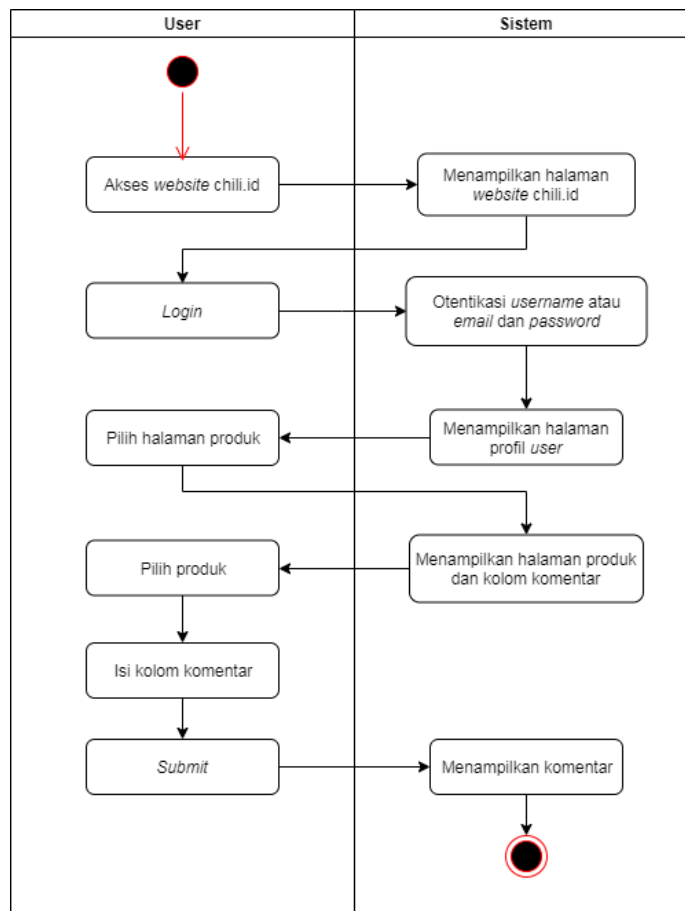
Activity diagram mendapatkan informasi *order* produk cabai menggambarkan proses *Registered-user* dalam melihat informasi *order* yang telah dilakukan pada website di chili.id. *User* terlebih dahulu harus melakukan *login*. Setelah berhasil *login*, *user* memilih halaman *order* dan sistem akan menampilkan daftar history atau informasi *oder* yang telah dilakukan oleh *user*.



Gambar 16. Activity Diagram Mendapatkan Informasi Order Produk Cabai

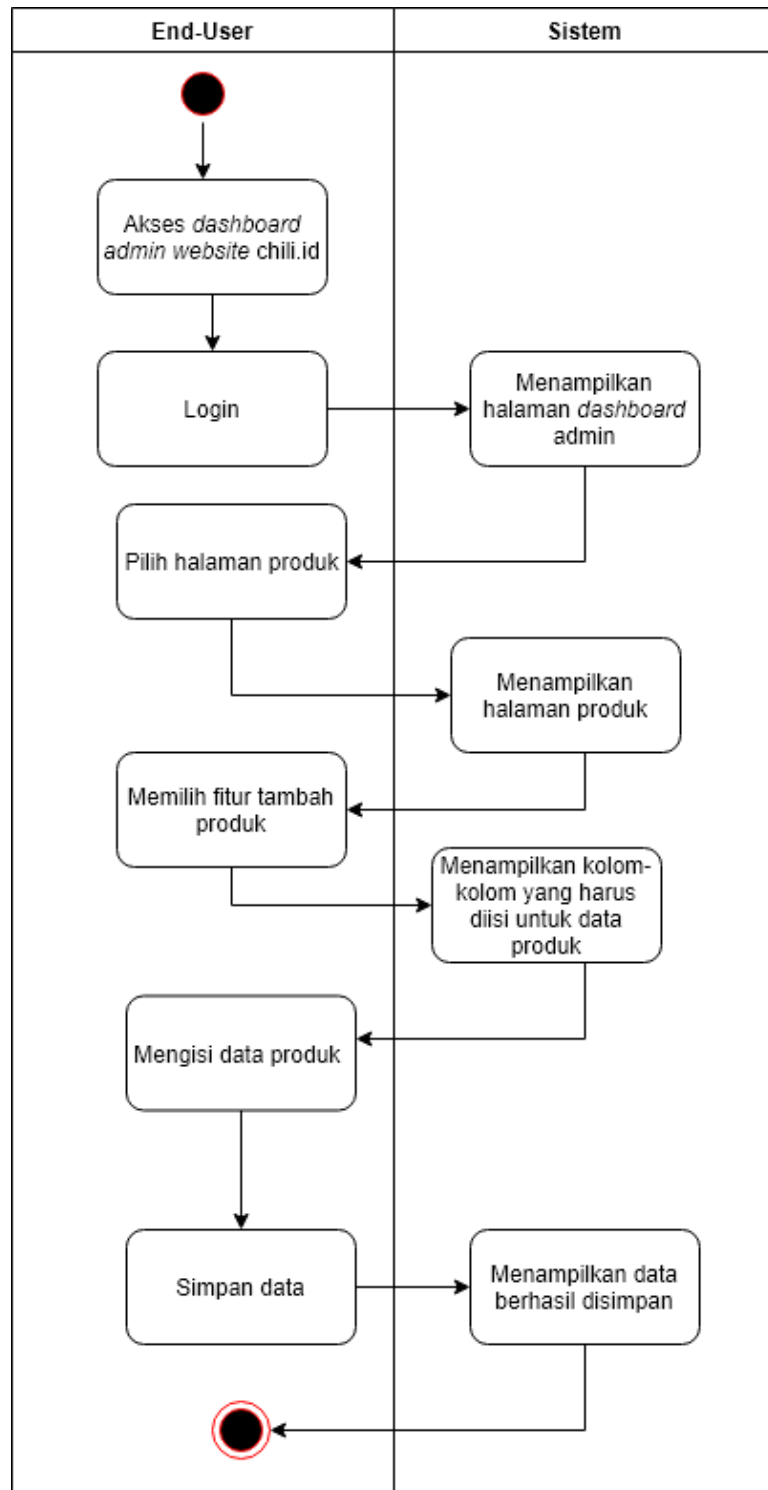
h) Activity Diagram Memberikan komentar

Activity diagram memberikan komentar menggambarkan proses *Registered-user* dalam memberikan komentar terhadap produk yang ada pada website di chili.id. *Registered-user* terlebih dahulu harus melakukan *login*. Setelah berhasil *login*, *user* memilih produk yang ingin diberikan komentar. Kemudian pada halaman produk yang dipilih sistem akan menampilkan kolom komentar yang bisa diisi oleh *registered-user*. Setelah *Registered-user* mengisi komentar dan melakukan submit maka sistem akan menampilkan komentar dari *user* pada halaman produk tersebut.



Gambar 17. Activity Diagram Memberi Komentar Pada Halaman Produk

i) *Activity Diagram Menambah produk*

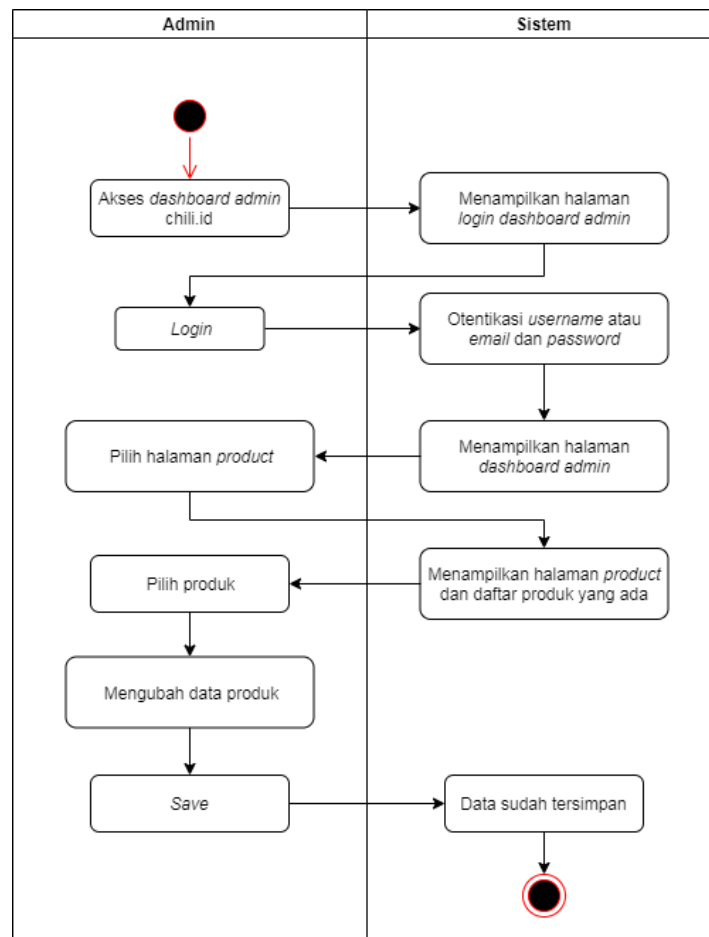


Gambar 18. *Activity Diagram Menambah Produk*

Activity Diagram menambah produk pada gambar 18 menggambarkan proses bagaimana admin menambahkan produk baru pada website di chili.id. Admin terlebih dahulu harus melakukan *login* pada *dashboard*. Setelah berhasil *login*, admin memilih menu produk. Kemudian pada halaman produk yang dipilih sistem akan menampilkan *list* produk yang ada pada website chili.id. Kemudian admin memilih fitur menambahkan produk dan sistem akan menampilkan kolom-kolom yang harus diisi untuk menambahkan data produk baru. Setelah admin mengisi data produk dan menyimpan datanya maka sistem akan menampilkan data baru tersebut berhasil ditambahkan.

j) **Activity Diagram Melakukan update produk**

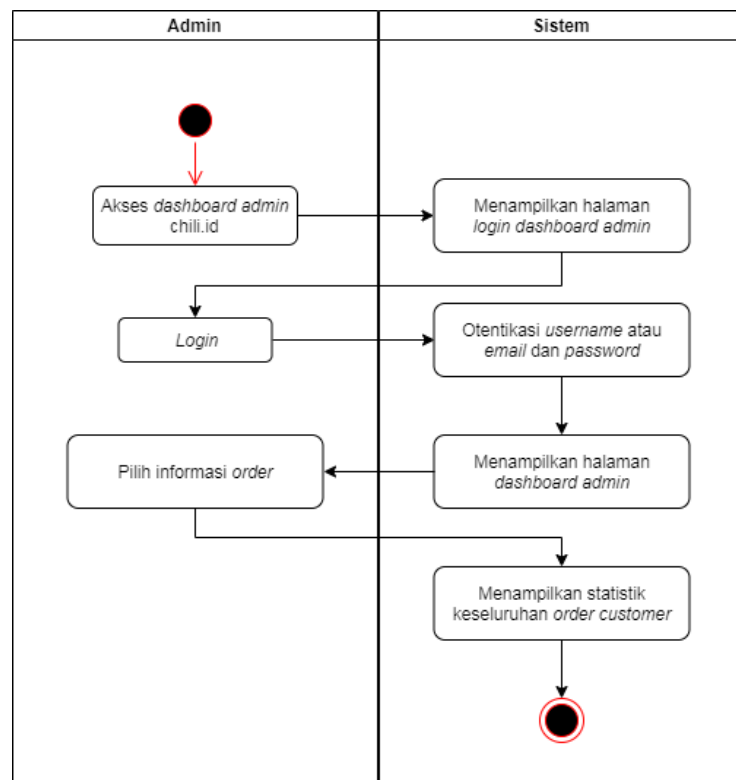
Activity diagram update produk menggambarkan proses bagaimana admin melakukan perubahan atau menghapus produk yang ada website di chili.id. Admin terlebih dahulu harus melakukan *login* pada *dashboard*. Setelah berhasil *login*, admin memilih menu produk. Kemudian pada halaman produk yang dipilih sistem akan menampilkan *list* produk yang ada pada *website* chili.id. Kemudian admin memilih fitur *update* dan sistem akan menampilkan kolom-kolom yang harus diisi untuk mengubah data produk. Setelah admin mengisi data produk dan menyimpan datanya maka sistem akan menampilkan data produk tersebut berhasil di-*update*.



Gambar 19. *Activity Diagram Update Produk*

k) Activity Diagram Melihat informasi Order Customer

Activity diagram update produk menggambarkan proses bagaimana admin melihat informasi *order* yang telah dilakukan *customer* pada *chili.id*. Admin terlebih dahulu harus melakukan *login* pada *dashboard*. Setelah berhasil *login*, pada halaman awal *dashboard* admin, sistem akan menampilkan statistik keseluruhan *order customer*.



Gambar 20. Activity Diagram Melihat Informasi Order Customer

4.2.3 Rancangan Tampilan

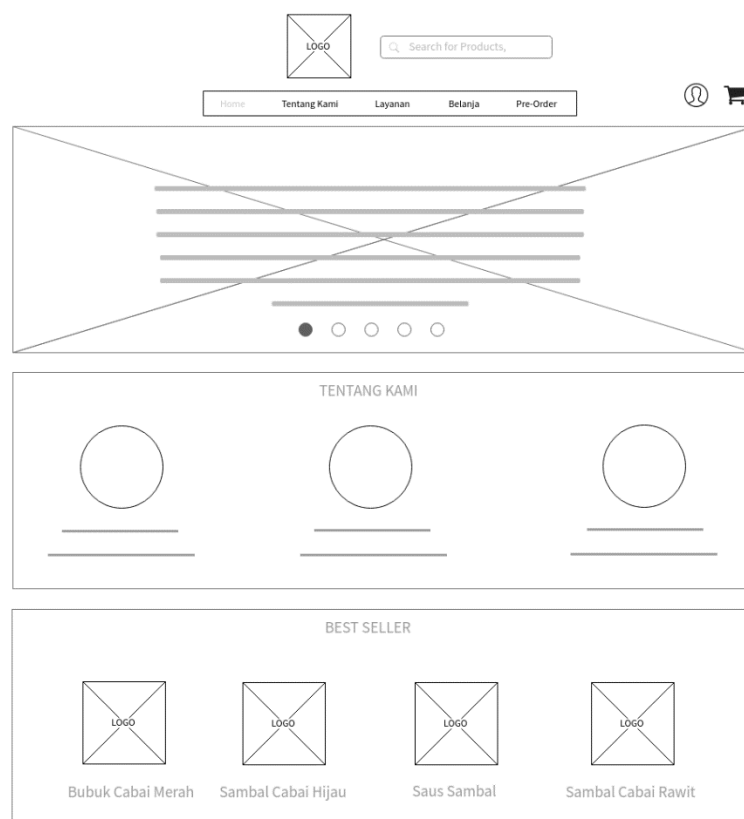
Setelah melakukan perancangan atau desain sistem, selanjutnya adalah membuat antarmuka sistem yang menggambarkan tampilan dari aplikasi yang akan dirancang. Berikut antarmuka sistem utama yang akan dibangun.

a) *Wireframe*

Wireframe adalah kerangka atau coretan kasar untuk penataan item-item pada halaman website sebelum proses desain sesungguhnya dimulai.

1) *Wireframe* Halaman Awal Website

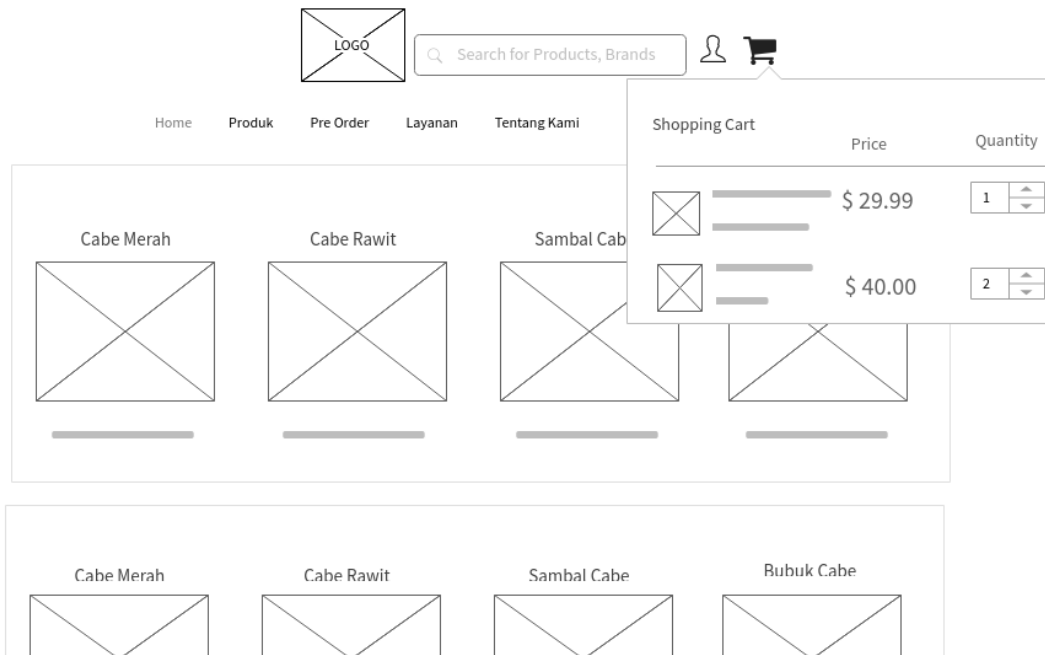
Wireframe halaman awal website menampilkan produk-produk terbaru dan informasi mengenai *e-commerce* chili.id



Gambar 21. *Wireframe* Melihat Halaman Website

2) Wireframe Halaman Produk

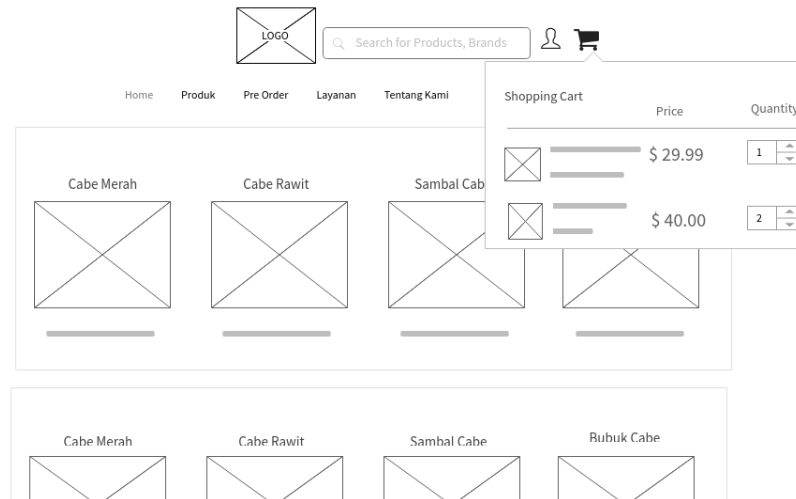
Wireframe halaman produk menampilkan produk-produk yang ada pada chili.id dimana pada menu ini terdapat sub-menu kategori produk.



Gambar 22 Wireframe Halaman Produk

3) **Wireframe Shopping Cart**

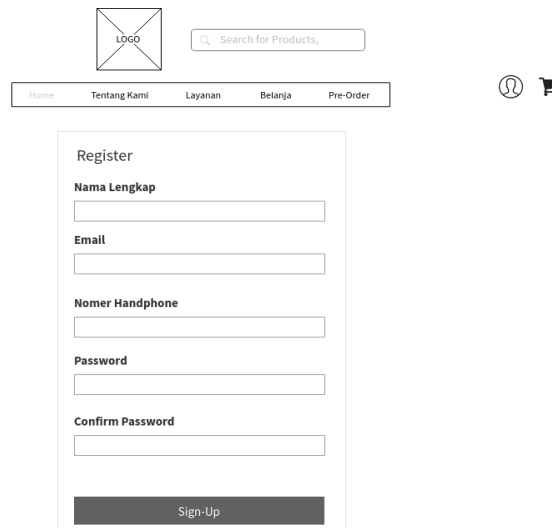
Wireframe shopping cart menampilkan produk-produk apa saja yang ditambahkan pada *shopping cart* dan terdapat informasi terkait harga dan kuantitas nya.



Gambar 23. Wireframe Shopping Cart Produk

4) **Wireframe Register Akun**

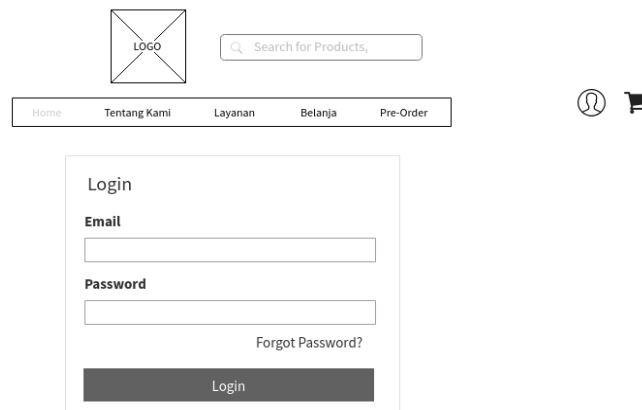
Wireframe register akun menampilkan halaman *register* akun yang berisi kolom-kolom data diri yang harus diisi jika ingin mempunyai akun pada website chili.id.



Gambar 24. Wireframe Register Akun

5) *Wireframe Login*

Wireframe login menampilkan halaman *login* yang berisi kolom-kolom data *email* dan *password* yang harus diisi jika ingin melakukan *login* pada website chili.id.

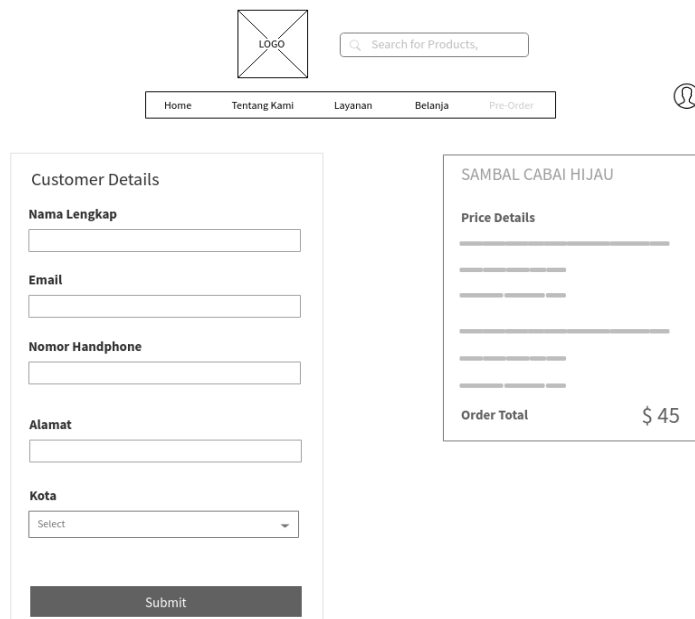


The wireframe shows a login page layout. At the top, there is a header area containing a placeholder for a logo (a square with an 'X' and the word 'LOGO'), a search bar with the text 'Search for Products,', and a navigation menu with links: 'Home', 'Tentang Kami', 'Layanan', 'Belanja', and 'Pre-Order'. To the right of the navigation menu are icons for a user profile and a shopping cart. The main content area is a 'Login' form. It includes labels for 'Email' and 'Password', each followed by a text input field. Below the password field is a link that says 'Forgot Password?'. At the bottom of the form is a dark 'Login' button.

Gambar 25. *Wireframe Login*

6) *Wireframe Identitas Pembelian Produk*

Wireframe identitas pembelian produk menampilkan halaman pembelian produk yang berisi kolom-kolom yang harus diisi untuk keperluan pengiriman *order* dan pada sebelah kanan halaman terdapat informasi total *order* yang dilakukan.



The wireframe shows a product purchase page layout. The header is identical to the login page, with the logo placeholder, search bar, navigation menu, and user/shopping cart icons. The main content area is divided into two columns. The left column is titled 'Customer Details' and contains form fields for 'Nama Lengkap', 'Email', 'Nomor Handphone', 'Alamat', and 'Kota' (a dropdown menu with 'Select' as the placeholder). At the bottom of this column is a dark 'Submit' button. The right column is titled 'SAMBAL CABAI HIJAU' and contains a 'Price Details' section with several horizontal lines representing item prices. At the bottom of this column, it shows 'Order Total' followed by '\$ 45'.

Gambar 26. *Wireframe Pembelian Produk*

7) Wireframe Pembayaran Order

Wireframe pembayaran produk menampilkan halaman pembayaran produk yang berisi pilihan untuk keperluan pembayaran *order* dan pada sebelah kanan halaman terdapat informasi total *order* yang dilakukan.

Gambar 27. Wireframe Pembayaran Produk

4.2.4 Rancangan Pengujian

a) Black Box Testing

Tabel 11. Black Box Testing

No	Skenario Pengujian	Butir Uji	Hasil Yang Diharapkan	Hasil Uji
1.	Menambah Produk	Menambah produk dengan tidak mengisi <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	-
		Menambah produk dengan	Produk baru berhasil disimpan	-

		mengisi semua <i>requirement field</i>	
2.	Mengakses Halaman Produk	Menguji fungsional halaman landing page	Menampilkan semua produk -
3.	Mencari Produk	Menguji pencarian produk tanpa memasukkan kata kunci yang dicari	Tidak memunculkan apa-apa -
		Menguji pencarian produk dengan memasukkan kata kunci yang dicari	Menampilkan produk yang dicari berdasarkan kata kunci yang terkait -
4.	Register Akun	Mendaftar akun dengan tidak mengisi salah satu <i>field</i>	Menampilkan pesan <i>error</i> untuk mengisi semua <i>field</i> -
		Mendaftar akun dengan memasukkan email yang sama	Menampilkan <i>error</i> bahwa email sudah terdaftar -
		Mendaftar akun dengan mengisi seluruh <i>field</i>	<i>User</i> berhasil terdaftar dan mengirim ke email <i>user</i> untuk aktivasi akun -
5.	Login Akun	<i>Login event</i> dengan tidak	Menampilkan <i>error</i> untuk mengisi semua <i>field</i> -

		mengisi salah satu <i>field</i>	
		<i>Login event</i> dengan <i>username</i> dan <i>password</i> yang salah	Menampilkan <i>error</i> - bahwa akun tidak terdaftar
		<i>Login event</i> dengan <i>username</i> dan <i>password</i> yang benar	Berhasil <i>Login</i> dan - diarahkan ke halaman <i>dashboard user</i>
6	Menambahkan Produk ke Shopping Cart	Menguji tombol <i>add to cart</i> dalam keadaan <i>user</i> belum <i>login</i>	Menampilkan produk - baru dalam keranjang bertambah
		Menguji tombol <i>add to cart</i> dalam keadaan <i>user</i> sudah <i>login</i>	Menampilkan produk - dalam keranjang bertambah
		Menguji tombol <i>cart</i> yang ada pada menu	Menampilkan list - produk yang telah ditambahkan
		Menguji tombol delete produk pada <i>shopping</i> <i>cart</i>	Tidak menampilkan - produk yang dihapus
7.	Mengakses Detail Produk	Menguji detail produk dalam menampilkan informasi detail produk	Menampilkan - informasi detail produk

8	Memperbarui Produk	Menguji fungsionalitas produk	Menampilkan Produk	-
		Mengedit data produk dengan tidak mengisi salah satu <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	-
		Mengedit data produk dengan mengisi semua <i>requirement field</i>	Identitas berhasil diedit dan memunculkan pesan berhasil diedit	-
		Menghapus produk yang ada	Produk berhasil dihapus dan tidak menampilkan produk yang dihapus	-
9	Membeli Produk	Menguji tombol beli pada produk dalam keadaan <i>user</i> belum <i>Login</i>	<i>User</i> akan diarahkan ke halaman <i>Login</i> sebelum membeli produk	-
		Menguji tombol beli pada produk dalam keadaan <i>user</i> sudah <i>login</i>	<i>User</i> berhasil <i>login</i> dan diarahkan ke halaman isi data untuk keperluan pembelian produk	-
		Mengisi identitas pada halaman <i>order</i> dengan salah satu <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	-

		Mengisi identitas pada halaman <i>order</i> dengan mengisi seluruh <i>requirement field</i>	<i>User</i> akan diarahkan ke halaman berhasil <i>order</i>	-
10	Memperbarui Identitas User	Menguji fungsionalitas <i>profile</i>	Menampilkan <i>user profile</i>	-
		Mengedit data identitas dengan tidak mengisi salah satu <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	-
		Mengedit data identitas dengan mengisi semua <i>requirement field</i>	Identitas berhasil diedit	-
11	Mengakses Informasi Order	Menguji informasi produk yang telah di <i>order</i>	Menampilkan informasi produk yang telah di <i>order</i>	-
12	Memberi Komentar	Mengisi komentar pada produk yang dipilih	Menampilkan komentar yang disimpan	-
13	Mengakses Order-Order Customer	Menguji data <i>order</i> visualisasi pada <i>dashboard</i> admin	Menampilkan visualisasi dari data <i>order-order customer</i>	-
14	Mengakses Halaman Landing Page	Menguji fungsional	Menampilkan tentang Chilli Indonesia dan produk terbaru	-

		halaman landing page		
15	Memberikan Konfirmasi Pembayaran	Menguji upload bukti transaksi pembayaran pada <i>dashboard user</i>	Menampilkan transaksi pembayaran berhasil diupload	-
16	Cek Transaksi Pembayaran Update status Order	Mengecek transaksi bukti pembayaran sudah diterima	Menampilkan transaksi pembayaran sudah diterima	-
		Memberikan nomer resi pengiriman pesanan ke email <i>user</i>	Menampilkan nomer resi berhasil diterima oleh <i>user</i>	-
		Mengganti status pembayaran menjadi berhasil	Menampilkan status pembayaran berhasil pada <i>dashboard user</i> dan admin	-

b) Fitur-Fitur

Tabel 12. Fitur chili.id

No	Fitur	Deskripsi
1	<i>Landing Page</i>	<i>User</i> mengakses halaman landing page untuk menampilkan tentang Chilli Indonesia dan produk terbaru
2	<i>Search Produk</i>	<i>User</i> mencari produk berdasarkan kata kunci
3	<i>Detail Produk</i>	<i>User</i> bisa melihat informasi detail suatu produk
4	<i>Tambah ke Shopping Cart</i>	<i>User</i> bisa menambahkan produk yang dipilih ke keranjang
5	<i>Register</i>	<i>User</i> bisa membuat akun
6	<i>Login</i>	<i>User</i> bisa <i>login</i> menggunakan akun yang telah dibuat
7	<i>Beli Produk</i>	<i>User</i> bisa membeli produk yang dipilih
8	<i>Edit Profile</i>	<i>User</i> bisa mengedit data <i>user profile</i>
9	<i>Order</i>	<i>User</i> bisa mendapatkan informasi dari <i>order-order</i> yang telah dilakukan
10	<i>Komentar</i>	<i>User</i> bisa memberikan komentar/ulasan pada produk
11	<i>Add Product</i>	Admin bisa menambahkan produk baru
12	<i>Edit Product</i>	Admin bisa mengedit produk yang sudah ada
13	<i>Order Visualization</i>	Admin bisa melihat visualisasi dari <i>order-order</i> produk <i>customer</i>
14	Konfirmasi Penerimaan Pembayaran	<i>User</i> bisa mengupload bukti transaksi pembayaran <i>order</i> Admin bisa mengecek bukti upload transaksi <i>order</i>

c) UAT

Pengujian UAT adalah suatu proses pengujian oleh pengguna untuk menghasilkan dokumen yang dijadikan bukti bahwa sistem yang dikembangkan dapat diterima atau tidaknya oleh pengguna, apabila hasil pengujian sudah bisa dianggap memenuhi kebutuhan dari pengguna maka aplikasi dapat diterapkan.

Pengujian dengan UAT dilakukan dengan mengajukan beberapa pertanyaan terhadap Bapak Reza Aldiansyah yang merupakan project owner dari pengembangan aplikasi web *E-Commerce* chili.id ini.

Tabel 13. UAT *Customer*

PENGUJIAN SISTEM PENGGUNA USER (CUSTOMER)					
NO	PERTANYAAN	SS	S	KS	TS
INTERFACE					
1	Tampilan pada landing page aplikasi web chili.id menarik				
2	Tampilan menu pada aplikasi web chili.id sesuai dengan yang diharapkan				
3	Tampilan menu produk pada aplikasi web chili.id sesuai dengan yang diharapkan				
4	Tampilan pada detail produk mempermudah dalam mengetahui informasi mengenai produk tersebut				
5	Tampilan halaman <i>Register</i> pada aplikasi web chili.id sesuai dengan yang diharapkan				
6	Tampilan halaman <i>Login</i> pada aplikasi web chili.id sesuai dengan yang diharapkan				
7	Tampilan <i>user</i> profile pada aplikasi web chili.id sesuai dengan yang diharapkan				

8	Tampilan <i>shopping cart</i> pada aplikasi web chili.id sesuai dengan yang diharapkan				
9	Tampilan <i>checkout order</i> pada aplikasi web chili.id sesuai dengan yang diharapkan				
10	Tampilan konfirmasi <i>order</i> pada aplikasi web chili.id sesuai dengan yang diharapkan				
11	Tampilan notifikasi <i>order</i> diterima pada aplikasi web chili.id sesuai dengan yang diharapkan				
12	Tampilan upload bukti transaksi pada aplikasi web chili.id sesuai dengan yang diharapkan				
13	Tampilan daftar <i>order</i> yang telah dibeli pada aplikasi web chili.id sesuai dengan yang diharapkan				
Total					
PROSES YANG TERJADI PADA SISTEM					
14	Proses pada menu <i>register</i> sesuai yang diinginkan				
15	Proses pada menu <i>login</i> sesuai yang diinginkan				
16	Proses pada menambahkan produk ke keranjang sesuai yang diinginkan				
17	Proses pada <i>checkout</i> atau pembelian produk sesuai yang diinginkan				
18	Proses pada upload bukti transaksi sesuai yang diinginkan				

19	Proses pada <i>user profile</i> sesuai yang diinginkan				
20	Proses pencarian produk sesuai yang diinginkan				
Total					
FUNGSIONAL SISTEM					
21	Aplikasi web <i>e-commerce</i> ini sesuai yang diharapkan				
22	Saat Aplikasi ini dijalankan tidak terdapat <i>error</i>				
23	Saat Aplikasi ini dijalankan tidak terdapat menu yang tidak berjalan				
24	Aplikasi web <i>e-commerce</i> ini cocok diterapkan pada chili.id				
25	Aplikasi ini bersifat <i>user friendly</i> (mudah digunakan)				
Total					

Tabel 14. UAT Admin

PENGUJIAN SISTEM PENGGUNA ADMIN					
NO	PERTANYAAN	SS	S	KS	TS
INTERFACE					
1	Tampilan pertama yang muncul pada <i>dashboard</i> admin ini menarik				
2	Tampilan menu pada <i>dashboard</i> admin ini menarik				
3	Tampilan menu untuk mengelola produk pada <i>dashboard</i> admin ini menarik				
4	Sebelum masuk ke sistem harus <i>login</i> terlebih dulu				
5	Tampilan pada menu produk pada <i>dashboard</i> admin ini menarik				
6	Tampilan pada menu <i>customer</i> pada <i>dashboard</i> admin ini menarik				
7	Tampilan pada menu <i>order</i> pada <i>dashboard</i> admin ini menarik				
8	Visualisasi <i>order customer</i> pada <i>dashboard</i> admin ini menarik				
Total					
PROSES YANG TERJADI PADA SISTEM					
9	Proses CRUD (<i>Create, Read, Update, Delete</i>) produk sesuai yang diinginkan				
10	Proses CRUD (<i>Create, Read, Update, Delete</i>) <i>Order</i> sesuai yang diinginkan				
11	Proses CRUD (<i>Create, Read, Update, Delete</i>) <i>customer</i> sesuai yang diinginkan				
12	Proses tracking <i>order</i> sesuai yang diharapkan				

13	Proses memperbarui tampilan store sesuai yang diinginkan				
Total					
FUNGSIONAL SISTEM					
14	Visualisasi <i>order customer</i> pada <i>dashboard</i> ini sesuai yang diharapkan				
15	<i>Dashboard</i> admin ini sesuai dengan yang diharapkan				
16	Saat <i>dashboard</i> admin ini dijalankan tidak terdapat menu yang tidak berjalan				
17	Saat aplikasi ini dijalankan tidak terdapat <i>error</i>				
18	<i>Dashboard</i> admin ini cocok diterapkan pada chili.id				
19	<i>Dashboard</i> admin ini bersifat <i>user friendly</i> (mudah digunakan)				
Total					

d) Skala Likert

Berikut skala likert yang digunakan sebagai acuan untuk mendapatkan persentase dari hasil UAT yang telah dibuat.

Tabel 15. Skala Likert

Variable	Keterangan	Nilai
Sangat Setuju	SS	4
Setuju	S	3
Kurang Setuju	KS	2
Tidak Setuju	TS	1

BAB V

HASIL DAN PEMBAHASAN

5.1 Implementasi Perangkat Lunak

Implementasi perangkat lunak yang digunakan menggunakan metode Scrum. Metode scrum merupakan salah satu dari metode agile dimana metode ini melihat teknik *requirement* sebagai proses yang berulang.

5.1.1 Perangkat yang digunakan

Untuk pengembangan aplikasi *E-Commerce* ini menggunakan ReactJS untuk tampilan web, Node.js dengan framework, MongoDB sebagai Database Management System, serta Postman sebagai software penguji REST API.

Peneliti menggunakan boilerplate Cezerin2 dalam mengembangkan aplikasi web *E-Commerce* chili.id. Boilerplate adalah template dasar untuk memulai sebuah proyek web baru. Boilerplate berisi potongan kode yang dapat dipakai berulang-ulang. Penggunaan boilerplate dilakukan karena terdapat beberapa fitur yang peneliti dapat digunakan untuk mengembangkan aplikasi web *e-commerce* chili.id. Sehingga akan membuat pengembangan aplikasi menjadi lebih efektif.

5.2 Implementasi SCRUM

5.2.1 Scrum Team

Pada pengembangan web *E-Commerce* chili.id terdapat tiga tokoh utama scrum, yaitu *scrum master*, *product owner*, dan tim pengembang. *Role Scrum* pada pengembangan sistem ini yaitu Bapak Reza Aldiansyah sebagai *Product Owner*, Dhimas Akbar Noor Baihaque sebagai Scrum Master. Adapun *Developer Team* pada bagian modul *pre-Order* marketplace oleh Muhammad Fadhil Hilmi dan peneliti pada bagian pengembangan modul marketplace.

5.2.2 Persiapan Sprint

Sesuai dengan kesepakatan antara *scrum master* dan anggota developer yang lain, yang disetujui oleh *product owner*. Peneliti menggunakan boilerplate Cezerin2 dalam mengembangkan aplikasi web *e-commerce* chili.id. Boilerplate adalah template dasar untuk memulai sebuah proyek web baru. Boilerplate berisi potongan kode yang dapat dipakai berulang-ulang. Penggunaan boilerplate dilakukan karena terdapat beberapa fitur yang peneliti dapat digunakan untuk mengembangkan aplikasi web *e-commerce* chili.id. Sehingga akan membuat pengembangan aplikasi menjadi lebih efektif. Boilerplate Cezerin2 ini memiliki frontend admin untuk manajemen data product, tampilan dan *customer*. Kemudian terdapat bagian *frontend store* yang dapat digunakan oleh *user* dalam melihat produk atau membeli produk yang terdapat pada aplikasi web. Selain itu terdapat juga backend yang akan mengatur logika aplikasi dan mengolah data-datanya pada aplikasi.

Pengembangan aplikasi web *e-commerce* ini menggunakan metode scrum dengan sprint sebanyak 4 iterasi. Sehingga pengembangan web aplikasi *e-commerce* akan dilaksanakan sesuai dengan *sprint backlog* yang telah dibuat.

1) Sprint 1

a. Sprint Planning Meeting

Product backlog yang telah dibuat secara detail didiskusikan dan dievaluasi oleh *product owner*. Diskusi dan evaluasi dilakukan secara bersama dengan developer team. Peneliti memilih *sprint backlog* yang akan dikerjakan pada sprint satu. Pada *sprint backlog* dipilih 3 dari 14 item *product backlog* yang ada pada *product backlog*. *Sprint backlog* yang telah ditentukan dapat dilihat pada Tabel 16.

Tabel 16 *Sprint Backlog* untuk sprint 1

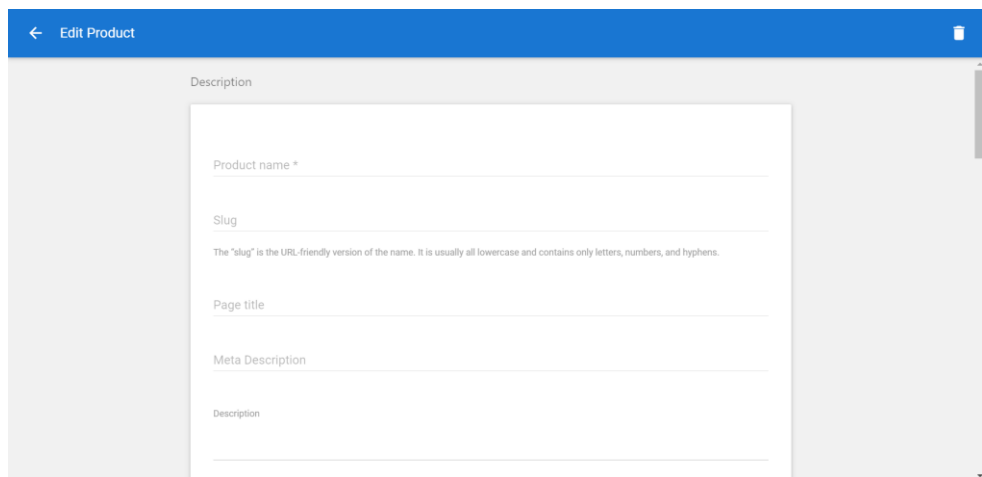
Sprint	User Story ID	Requirement	Prioritas
1	R.001	Melihat produk	Tinggi
1	R.003	Melakukan pencarian produk	Tinggi
1	R.014	Menambah produk	Tinggi

b. Daily Scrum

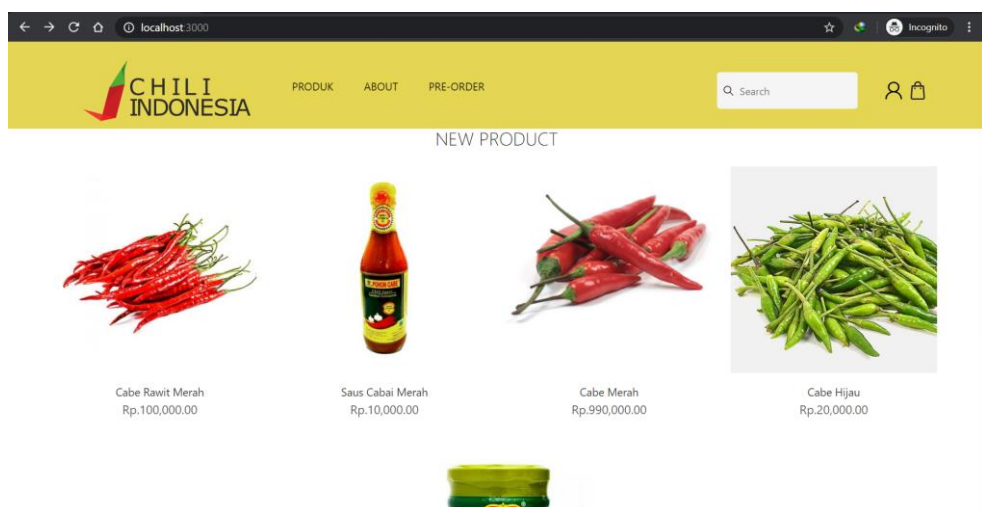
Berdasarkan analisis proses bisnis yang telah dilakukan sebelumnya maka diperlukan sistem pada aplikasi web *e-commerce* yang dapat menampilkan produk-produk pada *chili.id*. Untuk itu fungsi yang diperlukan, yaitu membuat model database pada MongoDB untuk menyimpan data produk, membuat fungsi *create* dan *read* untuk produk pada bagian *back-end* aplikasi, dan membuat REST API yang akan digunakan menghubungkan dengan bagian *frontend* aplikasi web.

REST API dapat diakses melalui URL yang digunakan untuk melakukan *request* yang diberikan pada web. URL yang berhubungan

dengan produk terdapat pada `localhost:3002/v1/product`. Untuk melakukan create atau read dengan maka kita menggunakan 2 metode yang terdapat pada REST API yaitu POST dan GET. Apabila API produk telah ada maka pada bagian frontend dapat menggunakan API tersebut untuk melakukan create dan read produk pada produk yang dilakukan pada bagian produk. Gambar 28 menunjukkan potongan bagian *frontend* admin untuk melakukan create produk dan gambar 29 menunjukkan produk yang berhasil dibuat dan dapat dilihat pada *frontend store*.

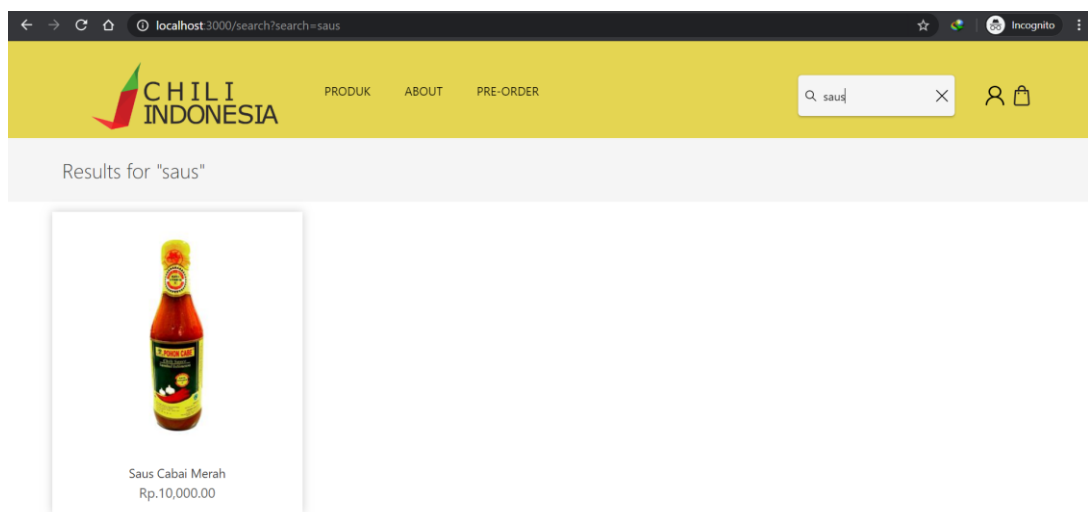
A screenshot of a web application's admin interface for editing a product. The page has a blue header bar with a back arrow and the text 'Edit Product'. Below the header, there is a form titled 'Description' with several input fields: 'Product name *', 'Slug' (with a note: 'The "slug" is the URL-friendly version of the name. It is usually all lowercase and contains only letters, numbers, and hyphens.'), 'Page title', 'Meta Description', and 'Description'. The form is set against a light gray background.

Gambar 28. *Frontend Admin untuk Create Product*

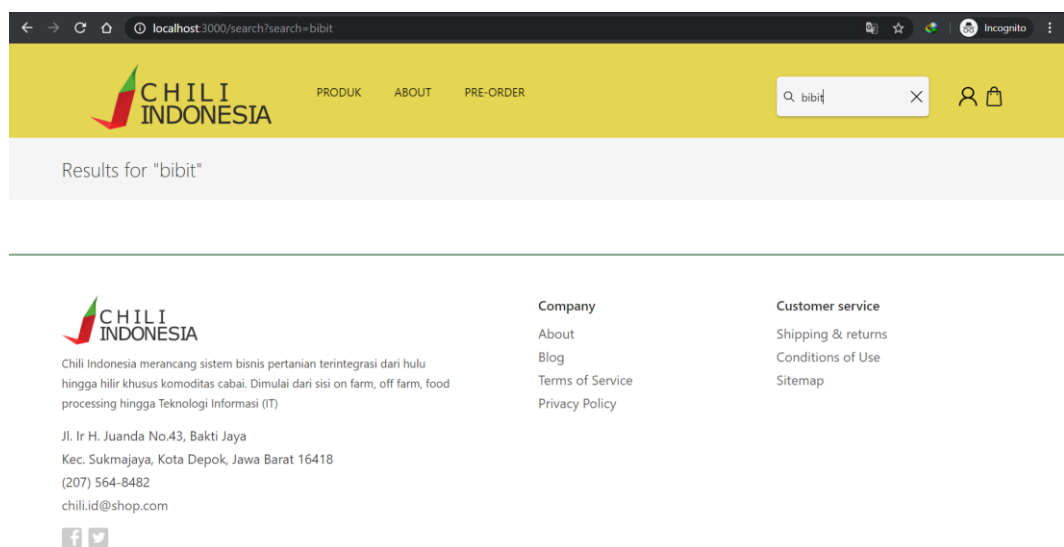


Gambar 29. Produk Baru ditampilkan pada *Frontend Store*

Untuk melakukan pencarian produk maka diperlukan sebuah fungsi pada backend yang akan dijalankan ketika *user* melakukan pencarian produk. Fungsi ini akan melakukan pencarian pada model produk sesuai dengan inputan yang dimasukkan. Gambar 30 menunjukkan bagian frontend produk dapat melakukan pencarian pada produk sesuai dengan nama yang diinput. Gambar 31 menunjukkan hasil apabila kata kunci yang dicari tidak ditemukan



Gambar 30. Hasil Pencarian Produk Sesuai Kata Kunci



Gambar 31. Hasil Pencarian Produk Ketika Kata Kunci Tidak Ditemukan

c. Sprint Review Meeting

Pada tahapan ini *developer team* kembali menjelaskan dan mempresentasikan hasil pada sprint kedua kepada *product owner*. *Review* dilakukan untuk menerima masukan berupa *feedback* atas fungsi *create*, *read* dan *search* produk. Hasil pengujian yang dipresentasikan kepada *product owner* dapat dilihat pada Table 17.

Tabel 17. Hasil Pengujian *Sprint Backlog* pada *Sprint 1*

Skenario Pengujian	Butir Uji	Hasil Yang Diharapkan	Hasil Uji
Menambah Produk	Menambah produk dengan tidak mengisi <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	Berhasil
	Menambah produk dengan mengisi semua <i>requirement field</i>	Produk baru berhasil disimpan	Berhasil
Mengakses Halaman Produk	Menguji fungsional halaman landing page	Menampilkan semua produk	Berhasil
Mencari Produk	Menguji pencarian produk dengan memasukkan kata kunci yang tidak ada pada aplikasi	Tidak memunculkan apa-apa	Berhasil
	Menguji pencarian produk dengan memasukkan kata kunci yang dicari	Menampilkan produk yang dicari berdasarkan kata kunci yang terkait	Berhasil

Product owner melihat hasil pengujian terhadap fungsi yang dilakukan. Pada tahapan ini scrum master memastikan bahwa tahapan ini berlangsung. Item sprint backlog yang telah dipresentasikan disetujui oleh product owner sehingga tidak terdapat perubahan dari sprint backlog yang dikerjakan pada sprint 1. Sprint review untuk *sprint 1* dapat dilihat pada Tabel 18.

Tabel 18. Sprint Review untuk sprint 1

No.	<i>Product Backlog</i>	<i>Product Owner</i>	Keterangan
1.	Melihat produk	Setuju	Tidak ada perubahan terhadap <i>product backlog</i>
2.	Melakukan pencarian produk	Setuju	Tidak ada perubahan terhadap <i>product backlog</i>
3.	Menambah produk	Setuju	Tidak ada perubahan terhadap <i>product backlog</i>

d. Sprint Retrospective

Pada tahap ini *developer team* dan peneliti melakukan diskusi dengan *product owner* untuk membahas kekurangan dan kelebihan yang sudah ada pada sprint 1. *Developer team* tidak mengalami kekurangan pada hasil fitur pada sprint 1. *Product owner* menyampaikan hasil yang sudah dikerjakan sesuai dengan *sprint backlog* yang ada pada sprint 1. Masalah yang terjadi pada sprint 1 ini yaitu harus memahami boilerplate cezerin2 yang menggunakan MERN (MongoDB, Express.js, ReactJS dan NodeJS) karena logic pada *boilerplate* ini sangat kompleks. Sehingga butuh waktu untuk memahaminya.

2) Sprint 2

a. *Sprint Planning Meeting*

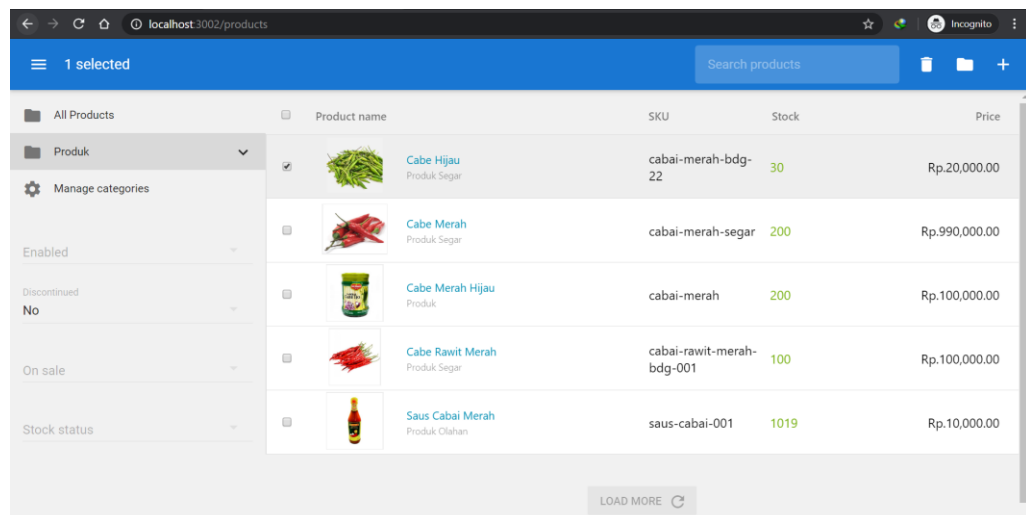
Sprint planning pada sprint 2 masih berhubungan dengan produk dan ditambah dengan fungsi *authentication* pada *user*. Dikerjakan 5 dari dari 14 item *product backlog* yang tersisa pada sprint 2. Fungsi yang dikerjakan sesuai dengan *requirement* yang dapat dilihat pada Tabel 19.

Tabel 19. Sprint Backlog untuk sprint 2

Sprint	<i>User Story ID</i>	<i>Requirement</i>	Prioritas
2	R.004	Menambah produk ke <i>Shopping Cart</i>	Sedang
2	R.005	Melakukan <i>Register</i>	Tinggi
2	R.006	<i>Login</i> sebagai <i>Registered-user</i>	Tinggi
2	R.017	Melakukan update produk (<i>edit/delete</i>)	Tinggi

b. Daily Scrum

Pada *sprint backlog update* produk yaitu *edit/delete* menggunakan method *put* untuk update dan method untuk *delete* pada *route product*. Fungsi edit atau delete produk ini dihubungkan dengan bagian frontend admin, sehingga hanya admin yang dapat menggunakan fungsi tersebut. Gambar 32 menunjukkan fungsi *delete* dan apabila ingin melakukan edit diharuskan untuk mengklik produknya lebih dulu kemudian mengedit produk tersebut.



The screenshot shows a web application interface for managing products. On the left, there is a sidebar with a menu containing 'All Products', 'Produk', and 'Manage categories'. The 'Produk' section is expanded, showing a list of products. Each product entry includes a checkbox, a product image, the product name, SKU, stock quantity, and price. The products listed are 'Cabe Hijau', 'Cabe Merah', 'Cabe Merah Hijau', 'Cabe Rawit Merah', and 'Saus Cabai Merah'. At the bottom of the product list, there is a 'LOAD MORE' button.

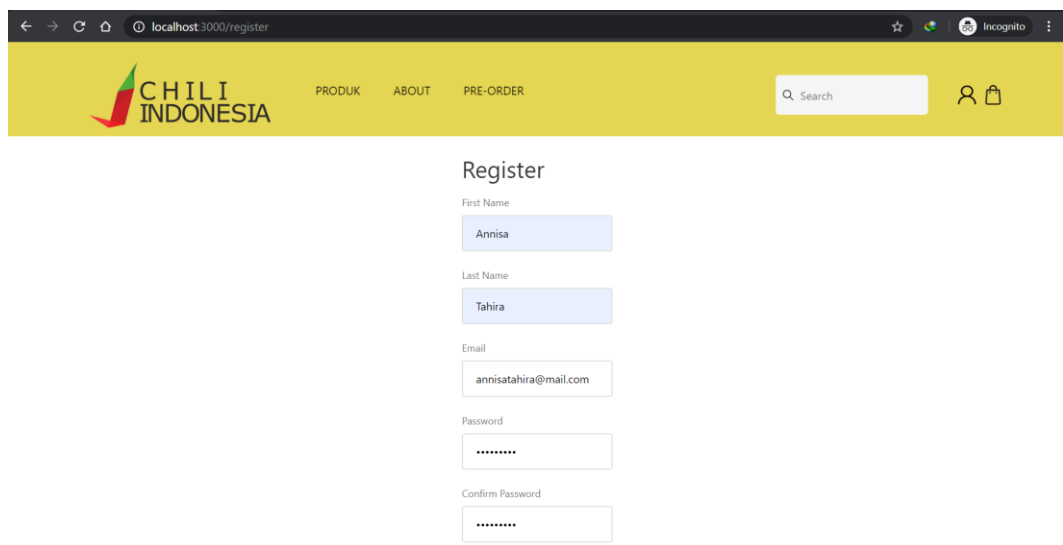
	Product name	SKU	Stock	Price
<input checked="" type="checkbox"/>	Cabe Hijau Produk Segar	cabai-merah-bdg-22	30	Rp.20,000.00
<input type="checkbox"/>	Cabe Merah Produk Segar	cabai-merah-segar	200	Rp.990,000.00
<input type="checkbox"/>	Cabe Merah Hijau Produk	cabai-merah	200	Rp.100,000.00
<input type="checkbox"/>	Cabe Rawit Merah Produk Segar	cabai-rawit-merah-bdg-001	100	Rp.100,000.00
<input type="checkbox"/>	Saus Cabai Merah Produk Olahan	saus-cabai-001	1019	Rp.10,000.00

Gambar 32. Fungsi Delete dan Edit pada *Dashboard Admin*

Kemudian untuk fungsi autentikasi *user* aplikasi web ini menggunakan *Jason Web Token (JWT)*. JWT adalah sebuah token berbentuk string panjang yang sangat random untuk melakukan sistem autentikasi dan pertukaran informasi. JWT pada sistem ini berisi informasi dari pengguna yang di-*encode* dalam bentuk base64. *User* diharuskan untuk melakukan *Register* lebih dulu apabila ingin *login* dan belum mempunyai akun sebelumnya. Bila masukan *user* benar, dilakukan *jwt.sign*, yaitu pembangkitan JWT token. Bila pengguna melakukan proses *logout*, token menjadi tidak valid

Untuk *testing register* akun ini menggunakan *mailtrap.io*. Jadi ketika *user* mengisi dan mengirimkan data pendaftarannya, maka

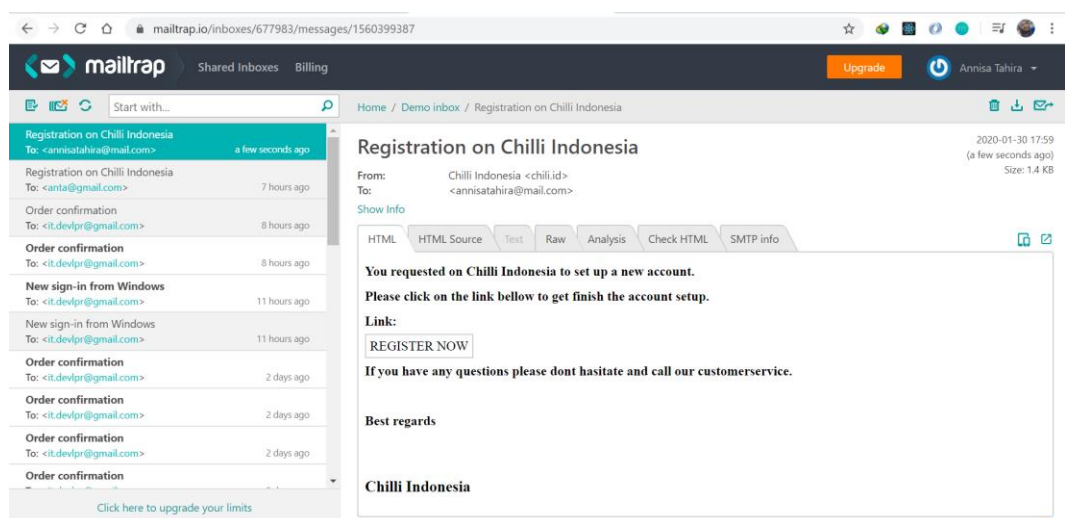
selanjutnya *user* akan mendapatkan token untuk melakukan aktivasi akun sehingga nantinya bisa *login* di website aplikasi. Form *Register* dan *login* terdapat bagian pada *frontend store* yang datanya akan di proses oleh bagian *backend* aplikasi kemudian ketika sukses *login* akan redirect pada *user profile* yang terdapat pada *frontend store*. Gambar 33 sampai gambar 36 menunjukkan bagaimana proses *Register*, aktivasi akun, hingga bisa melakukan *login* ke aplikasi web.



The screenshot shows a web browser at localhost:3000/register. The header features the 'CHILI INDONESIA' logo, navigation links for 'PRODUK', 'ABOUT', and 'PRE-ORDER', a search bar, and user icons. The main content area is titled 'Register' and contains the following form fields:

- First Name:** A text input field containing 'Annisa'.
- Last Name:** A text input field containing 'Tahira'.
- Email:** A text input field containing 'annisatahira@mail.com'.
- Password:** A password input field with masked characters '*****'.
- Confirm Password:** A password input field with masked characters '*****'.

Gambar 33. Register pada chili.id



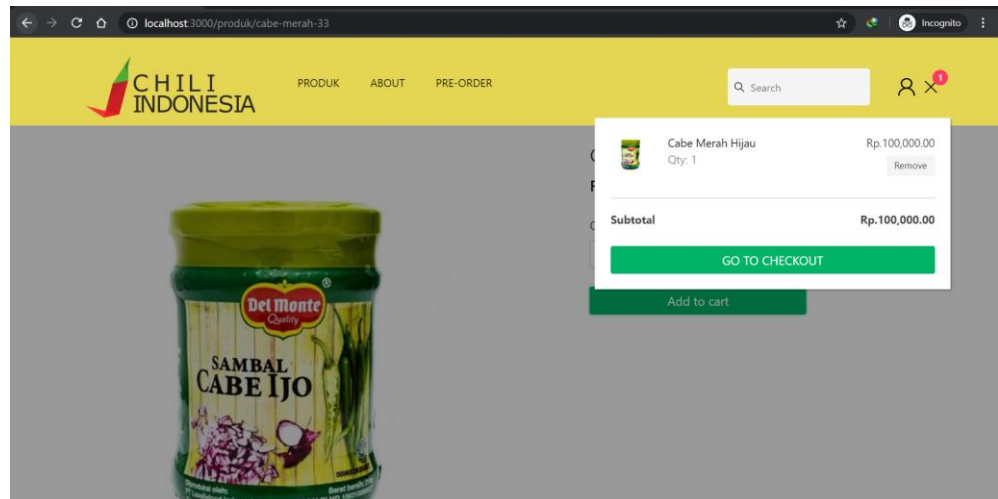
Gambar 34. Aktivasi akun dikirimkan ke Mailtrap.io

Gambar 35. Login pada chili.id

Gambar 36. Redirect ke user profile setelah berhasil login

Kemudian untuk melakukan penambahan produk ke *Shopping Cart*, *boilerplate* pada *cezerin2* menggunakan ajax untuk menambahkan item ke *Shopping Cart*. AJAX merupakan singkatan dari *Asynchronous JavaScript* dan XML. Dengan menggunakan ajax kita dapat mengirim dan mengambil data dari server secara *background*. Kemudian tampilan web akan tereload secara otomatis tanpa harus melakukan refresh lebih dulu. Ajax sangat sesuai digunakan dalam mengimplementasikan fungsi

menambahkan produk pada *shopping cart* ini. Gambar 37 menunjukkan penambahan item pada *cart* ketika ada produk baru ditambahkan.



Gambar 37. Menambahkan Product ke Shopping Cart

c. *Sprint Review Meeting*

Pada tahapan ini *developer team* kembali menjelaskan dan mempresentasikan hasil pada sprint kedua kepada *product owner*. *Review* dilakukan untuk menerima masukan berupa *feedback* atas fungsi *register*, *login*, *edit* atau *delete* produk, dan menambahkan produk pada *shopping cart*. Hasil pengujian chili.id dapat dilihat pada Tabel 20.

Tabel 20. Hasil Pengujian *Sprint Backlog* pada *Sprint 2*

Skenario Pengujian	Butir Uji	Hasil Yang Diharapkan	Hasil Uji
Register Akun	Mendaftar akun dengan tidak mengisi salah satu <i>field</i>	Menampilkan pesan <i>error</i> untuk mengisi semua <i>field</i>	Berhasil
	Mendaftar akun dengan memasukkan email yang sama	Menampilkan <i>error</i> bahwa email sudah terdaftar	Berhasil
	Mendaftar akun dengan mengisi seluruh <i>field</i>	<i>User</i> berhasil terdaftar dan mengirim ke email <i>user</i> untuk aktivasi akun	Berhasil
Login Akun	<i>Login event</i> dengan tidak mengisi salah satu <i>field</i>	Menampilkan <i>error</i> untuk mengisi semua <i>field</i>	Berhasil
	<i>Login event</i> dengan <i>username</i> dan <i>password</i> yang salah	Menampilkan <i>error</i> bahwa akun tidak terdaftar	Berhasil

	<i>Login event</i> dengan <i>username</i> dan <i>password</i> yang benar	Berhasil <i>Login</i> dan diarahkan ke halaman <i>dashboard user</i>	Berhasil
Menambahkan Produk ke Shopping Cart	Menguji tombol <i>add to cart</i> dalam keadaan <i>user</i> belum <i>login</i>	Menampilkan produk baru dalam keranjang bertambah	Berhasil
	Menguji tombol <i>add to cart</i> dalam keadaan <i>user</i> sudah <i>login</i>	Menampilkan produk dalam keranjang bertambah	Berhasil
	Menguji tombol <i>cart</i> yang ada pada menu	Menampilkan <i>list</i> produk yang telah ditambahkan	Berhasil
	Menguji tombol <i>delete</i> produk pada <i>Shopping Cart</i>	Tidak menampilkan produk yang dihapus	Berhasil
Mengakses Detail Produk	Menguji detail produk dalam menampilkan informasi detail produk	Menampilkan informasi detail produk	Berhasil
Memperbarui Produk	Menguji fungsionalitas produk	Menampilkan Produk	Berhasil
	Mengedit data produk dengan tidak mengisi salah satu <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	Berhasil

Mengedit data produk dengan mengisi semua <i>requirement field</i>	Produk berhasil diedit dan memunculkan pesan berhasil diedit	Berhasil
Menghapus produk yang ada	Produk berhasil dihapus dan tidak menampilkan produk yang dihapus	Berhasil

Product owner melihat hasil pengujian terhadap fungsi yang telah dipresentasikan. Pada tahapan ini scrum master memastikan bahwa tahapan ini berlangsung. *Item sprint backlog* yang telah dikerjakan disetujui oleh product owner sehingga tidak terdapat perubahan dari sprint backlog yang dikerjakan pada *sprint 2. Sprint review* untuk sprint 2 dapat dilihat pada Tabel 21.

Tabel 21. *Sprint Review* untuk *Sprint 2*

No.	Product Backlog	Product Owner	Keterangan
1.	Menambah produk ke <i>Shopping Cart</i>	Setuju	Tidak ada perubahan terhadap <i>product backlog</i>
2.	Melakukan <i>Register</i>	Setuju	Tidak ada perubahan terhadap <i>product backlog</i>
3.	<i>Login</i> sebagai <i>Registered-user</i>	Setuju	Tidak ada perubahan terhadap <i>product backlog</i>
4.	Melakukan update produk (<i>edit/delete</i>)	Setuju	Tidak ada perubahan terhadap <i>product backlog</i>

d. *Sprint Retrospective*

Pada tahap ini peneliti dan *developer team* melakukan diskusi dengan *product owner* untuk membahas kekurangan dan kelebihan yang sudah dikerjakan pada sprint 2. *Product owner* menyampaikan hasil yang sudah dikerjakan sesuai dengan *sprint backlog* yang ada pada sprint 2. *Product backlog* yang tersisa akan dikerjakan pada sprint 3. Masalah yang dihadapi pada sprint 2 yaitu masih kurangnya pengetahuan mengenai AJAX. Solusi yang dilakukan adalah mempelajari AJAX lebih banyak lagi baik itu dari web atau video pembelajaran dari youtube. Masalah yang terjadi pada sprint 2 ini yaitu harus memahami boilerplate cezerin2 yang menggunakan MERN (MongoDB, Express.js, ReactJS dan NodeJS) karena *logic* pada *boilerplate* ini sangat kompleks. Sehingga butuh waktu untuk memahaminya dan coba berkonsultasi dengan *scrum master*.

3) *Sprint 3*

a. Sprint Planning Meeting

Product backlog yang tersisa pada sprint 3 terdapat 7 item, dikerjakan empat item pada sprint 3. *Sprint planning* pada sprint 3 fokus kepada fungsi pembelian produk dan *order* sesuai dengan sprint backlog yang telah dibuat pada Tabel 17. Sprint backlog pada sprint ini terdapat tiga product backlog item yaitu fungsi *create*, *read*, *update* dan *delete* (CRUD) pada *order* yang dilakukan oleh *user*. Fungsi *review produk*, yang berguna untuk *user* dalam memberi komentar *review* pada produk yang sudah dibeli.

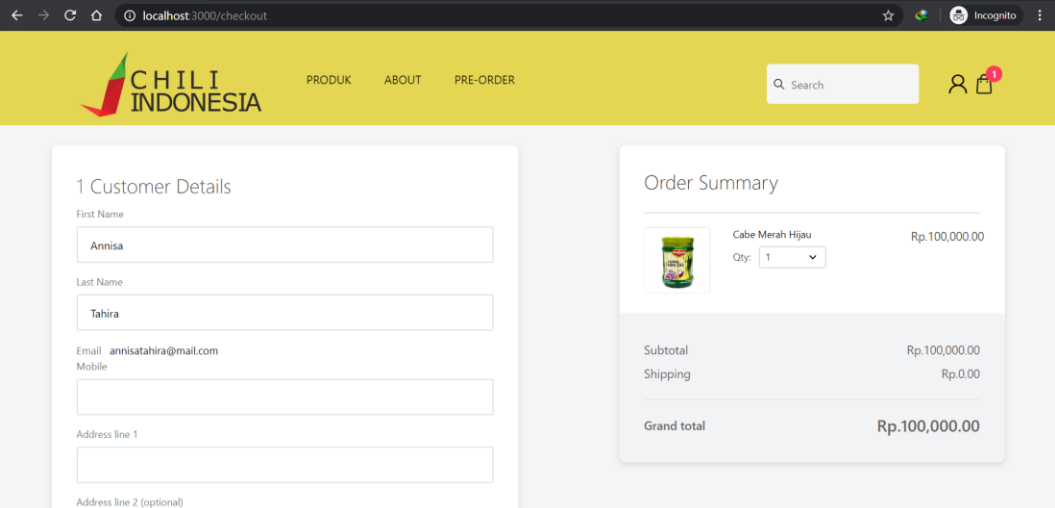
Tabel 22. *Sprint Backlog untuk Sprint 3*

Sprint	User Story ID	Requirement	Prioritas
3	R.010	Memperbarui halaman profil <i>user</i>	Tinggi
3	R.007	Membeli produk	Tinggi
3	R.011	Mendapatkan informasi <i>Order</i> produk cabai	Tinggi

b. Daily Scrum

Sprint ke 3 ini fokus kepada fungsi dimana *user* dapat melakukan pembelian produk pada aplikasi web. Pembelian produk pada aplikasi web hanya dapat dilakukan oleh *user* yang telah melakukan *login*. Fungsi ini dapat dilakukan oleh *user* yang tervalidasi menggunakan JWT validasi.

Untuk melakukan *order* ini akan mengakses api backend yaitu `http://localhost:3002/v1/orders`. Setelah memilih produk yang ingin dibeli *user* akan dialihkan ke halaman checkout product, dimana di halaman checkout ini *user* harus mengisi data-data untuk keperluan pembelian dan pengiriman *order*. Setelah diisi maka data *order* akan diproses kemudian disimpan di server. Apabila berhasil *user* akan dialihkan ke halaman checkout-success yang berisi informasi bahwa *order* telah diterima. Gambar 38 merupakan potongan *field* yang harus diisi *user* ketika melakukan ingin melakukan *checkout order*, gambar 40 menunjukkan page konfirmasi *order* dan gambar 39 menunjukkan notifikasi bahwa *order* selesai dibuat.




The screenshot shows a web browser at `localhost:3000/checkout` for the Chili Indonesia application. The page is divided into two main sections: '1 Customer Details' and 'Order Summary'.

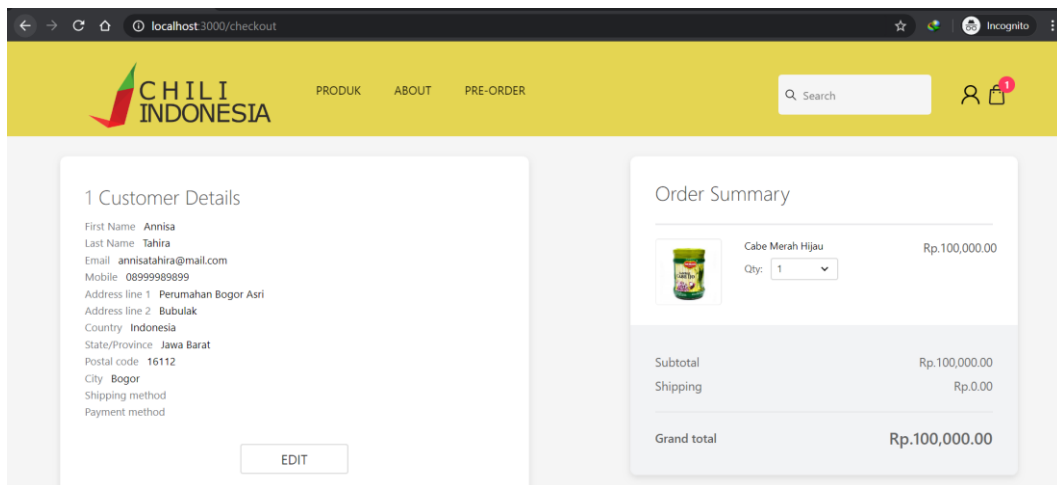
1 Customer Details

- First Name:
- Last Name:
- Email:
- Mobile:
- Address line 1:
- Address line 2 (optional):

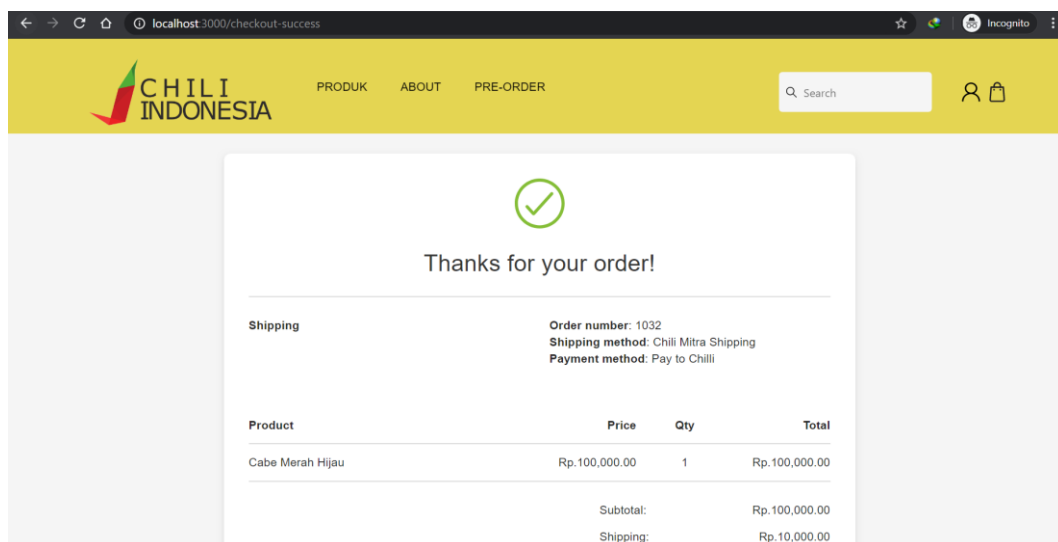
Order Summary

Order Summary	
	Cabe Merah Hijau Qty: <input type="text" value="1"/>
	Rp.100,000.00
<hr/>	
Subtotal	Rp.100,000.00
Shipping	Rp.0.00
<hr/>	
Grand total	Rp.100,000.00

Gambar 38. Data yang harus diisi untuk melakukan checkout produk



Gambar 39. Page Konfirmasi Data *Order*



Gambar 40. Notifikasi *Order* Diterima

Fungsi selanjutnya yang dikembangkan yaitu CRUD pada data *customer*. *User* dapat mengubah dan melihat data diri pada bagian *profile user*. Sedangkan untuk membuat, mengubah, melihat dan menghapus data hanya dapat dilakukan pada bagian *frontend* admin. Berikut gambar 41 yang menunjukkan form data diri *user* pada *profile user* yang bisa diisi oleh *user* dan frontend admin yang dapat melakukan fungsi CRUD pada data *order* yang dapat dilihat pada gambar 42 dan gambar 43.

localhost:3000/customer-account

CHILI INDONESIA

PRODUK ABOUT PRE-ORDER

Search

Welcome Annisa Tahira

Profile Orders Logout

Edit personal data

First Name
Annisa

Last Name
Tahira

Email
annisatahira@mail.com

Gambar 41. Edit Profile User

localhost:3002/orders

Orders

Search orders

Order	Shipping to	Total	Status
1000 October 30, 2019	Chili Mitra Shipping	Rp.10,950.00 PayPal	Sudah Upload Bukti Transaksi
1001 October 30, 2019	Chili Mitra Shipping	Rp.1,250.00 Pay to Chili	
1004 November 8, 2019	Annisa Tahira Chili Mitra Shipping	Rp.950.00 Pay to Chili	
1005 November 12, 2019	Annisa Tahira Chili Mitra Shipping	Rp.950.00 Pay to Chili	
1007 November 18, 2019	Annisa Tahira Chili Mitra Shipping	Rp.36,000.00 Pay to Chili	
1008 November 18, 2019	Annisa Tahira Chili Mitra Shipping	Rp.36,000.00 Pay to Chili	
1009 January 27, 2020	Chili Mitra Shipping	Rp.26,328.00 Pay to Chili	

All Statuses

Sudah Upload Bukti Transaksi

Manage statuses

Date placed from

Date placed to

Draft

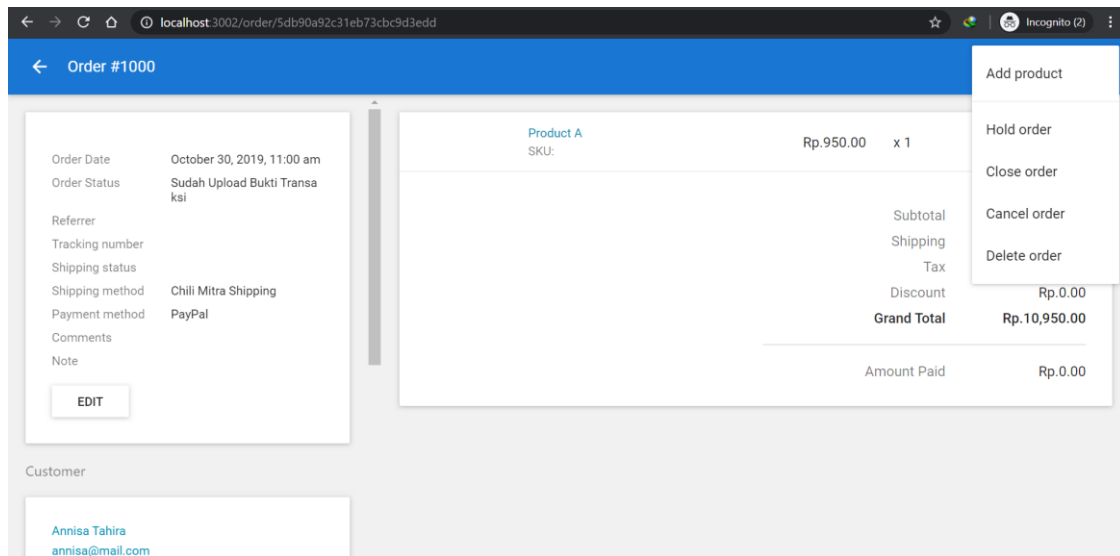
No

Hold

Paid

Delivered

Gambar 42. Menu Untuk Menambahkan atau Melihat *Order Customer*



Gambar 43. Menu Untuk *Edit* dan *Delete Order*

c. *Sprint Review Meeting*

Tabel 23. Hasil Pengujian *Sprint Backlog* pada *Sprint 3*

Skenario Pengujian	Butir Uji	Hasil Yang Diharapkan	Hasil Uji
Membeli Produk	Menguji tombol beli pada produk dalam keadaan <i>user</i> belum <i>login</i>	<i>User</i> akan diarahkan ke halaman <i>login</i> sebelum membeli produk	Berhasil
	Menguji tombol beli pada produk dalam keadaan <i>user</i> sudah <i>login</i>	<i>User</i> berhasil <i>login</i> dan diarahkan ke halaman isi data untuk keperluan pembelian produk	Berhasil
	Mengisi identitas pada halaman <i>order</i> dengan salah satu <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	Berhasil

	Mengisi identitas pada halaman <i>order</i> dengan mengisi seluruh <i>requirement field</i>	<i>User</i> akan diarahkan ke halaman berhasil <i>order</i>	Berhasil
Memperbarui Identitas <i>User</i>	Menguji fungsionalitas <i>profile</i>	Menampilkan <i>Profile User</i>	Berhasil
	Mengedit data identitas dengan tidak mengisi salah satu <i>requirement field</i>	Menampilkan <i>error</i> untuk mengisi <i>requirement field</i>	Berhasil
	Mengedit data identitas dengan mengisi semua <i>requirement field</i>	Identitas berhasil diedit	Berhasil
Mengakses Informasi <i>Order</i>	Menguji informasi produk yang telah di <i>Order</i>	Menampilkan informasi produk yang telah di <i>order</i>	Berhasil
Mengakses <i>Order-Order Customer</i>	Menguji data <i>order</i> visualisasi pada <i>dashboard</i> admin	Menampilkan visualisasi dari data <i>order-order customer</i>	Berhasil

Pada tahapan ini peneliti dan developer team kembali menjelaskan dan mempresentasikan hasil pada sprint 3 kepada *product owner*. *Review* dilakukan untuk menerima masukan berupa *feedback* atas pengembangan fungsi *CRUD order* dan *CRUD* pada data *Customer*. Hasil pengujian sprint backlog untuk sprint 3 dapat dilihat pada Tabel 23 diatas. Namun, terdapat 1 item sprint backlog yang belum berhasil dan dikerjakan pada sprint tiga. Item sprint backlog lainnya yang telah dikerjakan disetujui oleh *product owner* sehingga tidak terdapat perubahan dari *sprint backlog* yang dikerjakan pada sprint 3. Pada tahapan ini scrum master memastikan bahwa tahapan ini berlangsung. *Sprint review* untuk sprint 3 dapat dilihat pada Tabel 24

Tabel 24. *Sprint Review* untuk *Sprint 3*

No.	Product Backlog	Product Owner	Keterangan
1.	Memperbarui halaman profil <i>user</i>	Setuju	Tidak ada perubahan terhadap product backlog
2.	Membeli produk	Setuju	Tidak ada perubahan terhadap product backlog
3.	Mendapatkan informasi <i>order</i> produk cabai	Setuju	Tidak ada perubahan terhadap product backlog
4.	Memberikan Komentar	Tidak Setuju	Fungsi Komentar tidak dapat dijalankan karena tidak dapat mengirimkan komentar sehingga atas izin product owner product backlog komentar dihapus

d. Sprint Retrospective

Pada tahap ini *developer team* melakukan diskusi dengan *product owner* untuk membahas kekurangan dan kelebihan yang sudah dikerjakan pada sprint 3. Masalah yang terjadi pada *sprint 3* ini yaitu harus memahami boilerplate cezerin2 yang menggunakan karena *logic* pada *boilerplate* ini sangat kompleks, baik itu pada bagian *frontend* maupun *backend*. Sehingga butuh waktu untuk memahaminya. Solusi yang dilakukan menemui *Scrum Master* untuk membantu memberikan pemahaman terhadap *boilerplate cezerin2* ini.

4) Sprint 4

a. Sprint Planning Meeting

Tabel 25. *Sprint Backlog* untuk *Sprint 4*

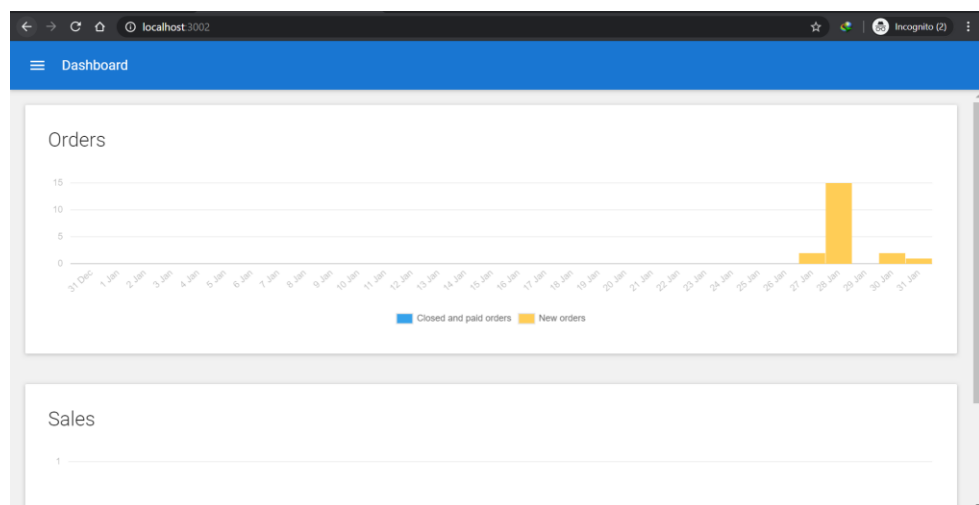
Sprint	User Story ID	Requirement	Prioritas
4	R.012	Melakukan konfirmasi pembayaran	Tinggi
4	R.018	Melakukan konfirmasi penerimaan pembayaran dan update status pembayaran <i>order</i>	Tinggi
4	R.019	Melihat informasi <i>oder customer</i>	Tinggi

Product backlog yang tersisa pada sprint 4 terdapat 4 item, 3 item baru yang dikerjakan pada sprint 4 dan 1 item yang belum selesai pada sprint 3. Sprint planning pada sprint 4 masih berfokus kepada pengodean pengembangan sistem sesuai dengan *sprint backlog* yang telah dibuat pada Tabel 25. Fungsi yang akan dikembangkan berhubungan dengan konfirmasi transaksi. *Product backlog* yang dikerjakan pada sprint 4 memiliki prioritas yang tinggi dan kesulitan yang tinggi. Fungsi konfirmasi *order* merupakan fungsi yang tidak terdapat pada boilerplate

cezerin2 sehingga pengembang mengharuskan menambahkan fungsi tersebut ke *boilerplate*.

b. *Daily Scrum*

Tahapan ini mengerjakan *sprint backlog* yang telah direncanakan pada sprint planning pada *sprint* 4. Fungsi melihat *order customer* yang divisualisasikan pada bagian frontend admin menggunakan barChart untuk memvisualisasikan data *order customer* berdasarkan waktu. *Order Bar* akan memanggil api *order* yang didalamnya terdapat fungsi-fungsi yang data *order customer* kemudian di mapping untuk mendapatkan visualisasi report. Visualisasi *order* dapat dilihat pada gambar 44 yang menunjukkan visualisasi *order* berdasarkan waktu pada bagian frontend admin.



Gambar 44. Visualisasi *Order* berdasarkan waktu

Fungsi yang dikembangkan selanjutnya yaitu membuat page pada frontend store, sehingga *user* dapat mengupload bukti transaksi terhadap *Order* yang telah dibeli. *Page* ini akan mengambil data dari *order customer* kemudian memberikan tempat untuk unduh *file*. Kemudian data transaksi ini akan dikirimkan ke *backend* untuk diproses, dan dilihat oleh admin pada bagian frontend admin bahwa bukti transaksi sudah

diupload. Kemudian admin akan mengubah status pembayaran menjadi sudah dibayar.

Penambahan fungsi baru pada *boilerplate* ini cukup sangat rumit karena menggunakan logic yang tinggi. Jadi ketika peneliti melakukan penambahan button untuk redirect ke halaman upload bukti transaksi, request tersebut tidak bisa digunakan. Sehingga fungsi yang diharapkan tidak berhasil dibuat.

c. *Sprint Review Meeting*

Pekerjaan yang telah dilakukan pada *sprint* 4 dipresentasikan kepada *product owner*. Kemudian dilakukan *review* untuk menerima masukan berupa feedback atas pengembangan aplikasi web sesuai *requirement* yang diharapkan. Pada tahapan ini scrum master memastikan bahwa tahapan ini berlangsung. Fungsi upload bukti transaksi pada *order* belum dapat dijalankan karena masih terdapat *error*. Hasil pengujian aplikasi chili.id dapat dilihat pada Tabel 26.

Tabel 26. Hasil pengujian *product backlog* pada *sprint* 4

Skenario Pengujian	Butir Uji	Hasil Yang Diharapkan	Hasil Uji
Mengakses <i>Order-Order Customer</i>	Menguji data <i>Order</i> visualisasi pada <i>dashboard</i> admin	Menampilkan visualisasi dari data <i>order-order customer</i>	Berhasil
Memberikan Konfirmasi Pembayaran	Menguji upload bukti transaksi pembayaran pada <i>dashboard user</i>	Menampilkan transaksi pembayaran berhasil diupload	Tidak Berhasil

Cek Transaksi	Mengecek	Menampilkan	Tidak
Pembayaran	transaksi bukti	transaksi pembayaran	Berhasil
Update status	pembayaran sudah	sudah diterima	
Order	diterima		
	Memberikan	Menampilkan nomer	Tidak
	nomer resi	resi berhasil diterima	Berhasil
	pengiriman	oleh <i>user</i>	
	pesanan ke email		
	<i>user</i>		
	Mengganti status	Menampilkan status	Tidak
	pembayaran	pembayaran berhasil	Berhasil
	menjadi berhasil	pada <i>dashboard user</i>	
		dan admin	

Hasil dari tahapan ini hanya 1 *sprint backlog* yang telah berhasil pengujian dan telah disetujui oleh *product owner*. *Item sprint backlog* lainnya yang telah dikerjakan tidak bisa diselesaikan tidak disetujui oleh *product owner*. Namun karena keterbatasan waktu dan sudah mencapai deadline akhir waktu untuk pengembangan aplikasi web, maka item backlog yang tersisa tidak dapat diselesaikan oleh peneliti. Dengan persetujuan *product owner* peneliti tidak bisa meneruskan pengembangan *product backlog* yang belum berhasil. *Sprint review* untuk sprint 4 dapat dilihat pada tabel 26 dengan sisa *item backlog* yang tidak dapat dipenuhi oleh peneliti.

d. Sprint Retrospective

Developer team diskusi dengan product owner untuk membahas kekurangan dan kelebihan yang sudah dikerjakan pada sprint 4. Dari keseluruhan *product backlog* yang telah dibuat tersisa 2 *product backlog* yang tidak bisa dikerjakan oleh peneliti. Permasalahan yang dihadapi yaitu karena dalam pengembangan aplikasi chili.id ini menggunakan boilerplate cezerin2 yang logiknya sangat rumit untuk dipahami. Pemahaman tentang MERN Stack dari peneliti juga masih kurang sehingga menyebabkan peneliti tidak bisa berhasil menambahkan fungsi yang belum ada pada *boilerplate* ini.

5.1 Rangkuman hasil pengujian blackbox

Berdasarkan pengujian yang dilakukan setiap sprintnya maka dapat didapatkan bahwa pengujian dari 14 *product backlog* yang dilakukan dengan *metode scrum* dengan *sprint* sebanyak 4 iterasi terdapat 12 *product backlog* yang berhasil diuji dan 2 *product backlog* tidak berhasil pengujian karena masih terdapat *error* pada sistem. Hasil keseluruhan pengujian product backlog dapat dilihat pada Tabel 27.

Tabel 27. Hasil Pengujian Backlog dengan Backlog

No	Skenario Pengujian	Butir Uji	Hasil Pengujian	Keterangan
1	Mengakses Halaman Landing Page	Menguji fungsional halaman <i>landing page</i>	Berhasil	Ketika mengakses halaman awal chili.id akan menampilkan slider banner dan produk terbaru chili.id
2	Mengakses Halaman Produk	Menguji fungsional halaman produk	Berhasil	Ketika mengakses halaman produk akan ditampilkan semua produk yang ada pada chili.id

3 Mencari Produk	Menguji pencarian produk dengan memasukkan kata kunci yang tidak ada pada chili.id	Berhasil	Hasil pencarian produk menampilkan pesan produk tidak ditemukan
	Menguji pencarian produk dengan memasukkan kata kunci yang dicari	Berhasil	Hasil pencarian produk sesuai dengan kata kunci yang diinputkan
4 Mengakses Detail Produk	Menguji detail produk dalam menampilkan informasi detail produk	Berhasil	Ketika mengklik produk maka akan ditampilkan informasi detail dari produk
5 Menambahkan Produk ke Shopping Cart	Menguji tombol <i>add to cart</i> dalam keadaan <i>user</i> belum <i>login</i>	Berhasil	Produk berhasil ditambahkan ke dalam <i>shopping cart</i>
	Menguji tombol <i>add to cart</i> dalam keadaan <i>user</i> sudah <i>login</i>	Berhasil	Produk berhasil ditambahkan ke dalam <i>shopping cart</i>
	Menguji tombol <i>cart</i> yang ada pada menu	Berhasil	Menampilkan produk-produk yang sudah ditambahkan ke <i>shopping cart</i> beserta jumlahnya
	Menguji tombol <i>delete</i> produk pada <i>shopping cart</i>	Berhasil	Produk yang dipilih untuk dihapus pada <i>shopping cart</i> berhasil dihilangkan

6 <i>Register Akun</i>	Mendaftar akun dengan tidak mengisi salah satu <i>field</i>	Berhasil	<i>Field</i> yang harus diisi akan memberikan peringatan yang mengharuskan user mengisi <i>field</i> tersebut
	Mendaftar akun dengan memasukkan email yang sama	Berhasil	Pada <i>field</i> email akan memberikan pesan bahwa email sudah terdaftar
	Mendaftar akun dengan mengisi seluruh <i>field</i>	Berhasil	Setelah melakukan pendaftaran akun maka akan dialihkan ke halaman register berhasil dan meminta <i>user</i> untuk mengecek email untuk mengaktifkan akun yang sudah didaftarkan tersebut.
7 <i>Login Akun</i>	<i>Login event</i> dengan tidak mengisi salah satu <i>field</i>	Berhasil	<i>Field</i> yang harus diisi akan memberikan peringatan yang mengharuskan user mengisi <i>field</i> tersebut
	<i>Login event</i> dengan <i>email</i> dan <i>password</i> yang salah	Berhasil	Menampilkan pesan bahwa email atau <i>password</i> salah sehingga tidak dialihkan ke halaman <i>profile user</i>
	<i>Login event</i> dengan <i>email</i> dan <i>password</i> yang benar	Berhasil	Setelah menekan tombol login maka akan dialihkan ke halaman <i>profile user</i>

8 Membeli Produk	Menguji tombol beli pada produk dalam keadaan <i>user</i> belum <i>login</i>	Berhasil	<i>User</i> akan dialihkan ke halaman <i>login</i>
	Menguji tombol beli pada produk dalam keadaan <i>user</i> sudah <i>login</i>	Berhasil	<i>User</i> akan dialihkan ke halaman <i>checkout</i> produk
	Mengisi identitas pada halaman <i>order</i> dengan tidak mengisi salah satu <i>requirement field</i>	Berhasil	<i>Field</i> yang harus diisi akan memberikan peringatan yang mengharuskan <i>user</i> mengisi <i>field</i> tersebut dan tidak bisa mengklik tombol <i>next</i> untuk melakukan konfirmasi <i>order</i> produk
	Mengisi identitas pada halaman <i>order</i> dengan mengisi seluruh <i>requirement field</i>	Berhasil	Bisa mengklik tombol <i>place order</i> untuk melakukan pembelian produk dan akan dialihkan ke halaman konfirmasi order
9 Memperbarui Identitas User	Menguji fungsionalitas profile	Berhasil	Menampilkan informasi identitas <i>user</i>
	Mengedit data identitas dengan tidak mengisi salah satu <i>requirement field</i>	Berhasil	<i>Field</i> yang harus diisi akan memberikan peringatan yang mengharuskan <i>user</i> mengisi <i>field</i> tersebut
	Mengedit data identitas dengan mengisi semua <i>requirement field</i>	Berhasil	Data <i>user</i> berhasil diperbarui dan akan dialihkan ke halaman <i>profile user</i> dengan menampilkan data yang telah diperbarui

10	Mengakses Informasi Order	Menguji informasi produk yang telah <i>diorder</i>	Berhasil	Mendapatkan list informasi produk yang telah <i>diorder</i>
11	Memberi Komentar	Mengisi komentar pada produk yang dipilih	Tidak Berhasil	Produk Backlog dihapus atas izin produk owner
12	Menambah Produk	Menambah produk dengan tidak mengisi <i>requirement field</i>	Berhasil	<i>Field</i> yang harus diisi akan memberikan peringatan yang mengharuskan user mengisi <i>field</i> tersebut
		Menambah produk dengan mengisi semua <i>requirement field</i>	Berhasil	Produk baru berhasil ditambahkan dan muncul pada halaman store
13	Memperbarui Produk	Menguji fungsionalitas produk	Berhasil	Menampilkan <i>field</i> data produk yang ingin diperbarui
		Mengedit data produk dengan tidak mengisi salah satu <i>requirement field</i>	Berhasil	<i>Field</i> yang harus diisi akan memberikan peringatan yang mengharuskan user mengisi <i>field</i> tersebut
		Mengedit data produk dengan mengisi semua <i>requirement field</i>	Berhasil	Data produk berhasil diperbarui dan akan dialihkan ke halaman list produk dengan menampilkan data yang telah diperbarui

		Menghapus produk yang ada	Berhasil	Produk berhasil dihapus dan tidak muncul kembali pada halaman store
14	Mengakses Order-Order Customer	Menguji data <i>order</i> visualisasi pada <i>dashboard</i> admin	Berhasil	Menampilkan visualisasi order pada <i>dashboard</i> admin berdasarkan waktu
15	Memberikan Konfirmasi Pembayaran	Menguji upload bukti transaksi pembayaran pada <i>dashboard user</i>	Tidak Berhasil	<i>Button</i> upload bukti pembayaran tidak redirect ke halaman upload bukti pembayaran
16	Cek Transaksi Pembayaran Update status Order	Mengecek transaksi bukti pembayaran sudah diterima	Tidak Berhasil	Fungsi upload bukti pembayaran tidak berhasil sehingga tidak bisa dilakukan pengecekan upload bukti transaksi
		Memberikan nomer resi pengiriman pesanan ke email <i>user</i>	Tidak Berhasil	Fungsi upload bukti pembayaran dan cek transaksi pembayaran tidak berhasil sehingga tidak bisa dilakukan pengiriman nomer resi pengiriman pesanan ke email user.
		Mengganti status pembayaran menjadi berhasil	Berhasil	Pada <i>dashboard</i> admin berhasil mengubah status pembayaran menjadi berhasil

5.2 Perbandingan Hasil Analisis PIECES

Berdasarkan aplikasi web *e-commerce* chili.id yang telah dibuat, peneliti menyimpulkan beberapa analisis terkait perbandingan antara sistem yang sudah berjalan dengan sistem yang telah dikembangkan ini. Hasil analisis dapat dilihat pada tabel

Tabel 28. Perbandingan Hasil Analisis PIECES

Analisis PIECES	Sistem yang sedang berjalan	Sistem yang telah dikembangkan
Kinerja (<i>Performance</i>)	Kinerja dari sistem jual-beli yang ada di chili.id masih kurang, karena melakukan kegiatan transaksi melalui whatsapp. Cara tersebut kurang begitu efektif ketika proses jual-beli transaksi ingin dilakukan dengan cepat.	Proses transaksi jual-beli dilakukan melalui aplikasi web chili.id dan secara otomatis tersimpan pada database. User dapat dengan mudah dan cepat melakukan order produk tanpa harus berintraksi dengan admin lebih dulu.
Informasi (<i>Information</i>)	Sistem yang ada saat ini walaupun sudah menggunakan media promosi seperti whatsapp, facebook, instagram dan media sosial tim yang terlibat, masih dirasa kurang dalam penyebaran informasinya. Hal ini dikarenakan belum adanya platform khusus yang menyediakan informasi detail produk-produk yang ada pada chili.id sehingga masih banyak	Informasi keseluruhan produk dan detail produk tersedia pada aplikasi web chili.id.

	masyarakat yang tidak mengetahui chili.id	
Analisis Ekonomi (<i>Economic</i>)	Penerapan sistem baru yang diusulkan diharapkan dapat menekan biaya-biaya yang dikeluarkan untuk media promosi dan bisa menambah pemasukan biaya dari sistem <i>E-Commerce</i> yang dibuat dengan membuka jasa iklan pada website. Selain itu, jika ada perubahan informasi produk, maka informasi akan mudah diubah tanpa membutuhkan waktu dan biaya promosi lagi. Jadi sistem baru yang diusulkan akan lebih ekonomis dan daripada sistem lama.	Perubahan informasi produk bisa langsung dilakukan pada aplikasi web sehingga tidak membutuhkan waktu dan biaya promosi lagi.
Keamanan atau kontrol (<i>Security or Control</i>)	Kurang maksimalnya kontrol terhadap data produk dan juga data pembelian, karena saat ini data masih disimpan pada <i>google form</i> .	Admin dengan mudah melakukan control terhadap data order atau customer. Perubahan data dapat langsung dilakukan pada aplikasi web dan akan langsung tersimpan secara otomatis pada database.
Efisiensi (<i>Efficiency</i>)	Admin harus melakukan pengisian form <i>Order</i> setiap ada pembelian produk sehingga akan membutuhkan waktu yang lebih dalam proses pencatatan transaksi. Penyampaian informasi juga	Proses pembelian order akan langsung tercatat pada database tanpa harus manual admin mengisi form order sehingga menjadi lebih efektif.

	tidak bisa diakses setiap saat. Sehingga pemanfaatan sumber daya yang tersedia seperti sumber daya operasional dan teknologi masih kurang.	
Layanan (<i>Service</i>)	Pelayanan yang diberikan kepada konsumen menjadi kurang maksimal, karena keterbatasan media promosi menyebabkan proses pelayanan hanya sebatas kepada orang-orang yang mengikuti media sosial yang terkait produk-produk chili.id atau tempat dimana produk di jual. Selain itu, apabila ada pemesanan produk dalam jumlah banyak dalam waktu yang bersamaan, maka pelayanan sering tidak maksimal.	Semua layanan yang diberikan oleh chili.id secara keseluruhan dapat ditemukan pada website.

5.3 HASIL UAT (*User Acceptance Test*)

UAT adalah proses pengujian aplikasi yang dilakukan oleh *user*. Pada pengembangan aplikasi web *e-commerce* ini, pengujian UAT dilakukan oleh *product owner*. Pengujian ini untuk mengetahui keberhasilan pembuatan aplikasi sesuai dengan *requirement*.

Tabel 29. UAT pada sistem pengguna *user (customer)*

PENGUJIAN SISTEM PENGGUNA <i>USER (CUSTOMER)</i>					
NO	PERTANYAAN	SS	S	KS	TS
INTERFACE					
1	Tampilan pada landing page aplikasi web chili.id menarik		v		
2	Tampilan menu pada aplikasi web chili.id sesuai dengan yang diharapkan		v		
3	Tampilan menu produk pada aplikasi web chili.id sesuai dengan yang diharapkan		v		
4	Tampilan pada detail produk mempermudah dalam mengetahui informasi mengenai produk tersebut		v		
5	Tampilan halaman <i>register</i> pada aplikasi web chili.id sesuai dengan yang diharapkan		v		
6	Tampilan halaman <i>login</i> pada aplikasi web chili.id sesuai dengan yang diharapkan		v		
7	Tampilan <i>user profile</i> pada aplikasi web chili.id sesuai dengan yang diharapkan			v	
8	Tampilan <i>shopping cart</i> pada aplikasi web chili.id sesuai dengan yang diharapkan		v		

9	Tampilan checkout <i>Order</i> pada aplikasi web chili.id sesuai dengan yang diharapkan		v		
10	Tampilan konfirmasi <i>Order</i> pada aplikasi web chili.id sesuai dengan yang diharapkan		v		
11	Tampilan notifikasi <i>Order</i> diterima pada aplikasi web chili.id sesuai dengan yang diharapkan	v			
12	Tampilan upload bukti transaksi pada aplikasi web chili.id sesuai dengan yang diharapkan				v
13	Tampilan daftar <i>Order</i> yang telah dibeli pada aplikasi web chili.id sesuai dengan yang diharapkan		v		
Total		1	10	1	1
PROSES YANG TERJADI PADA SISTEM					
14	Proses pada menu <i>Register</i> sesuai yang diinginkan	v			
15	Proses pada menu <i>Login</i> sesuai yang diinginkan	v			
16	Proses pada menambahkan produk ke keranjang sesuai yang diinginkan	v			
17	Proses pada checkout atau pembelian produk sesuai yang diinginkan	v			
18	Proses pada upload bukti transaksi sesuai yang diinginkan				v
19	Proses pada <i>user</i> profile sesuai yang diinginkan		v		

20	Proses pencarian produk sesuai yang diinginkan		v		
Total		4	2		1
FUNGSIONAL SISTEM					
21	Aplikasi web <i>E-Commerce</i> ini sesuai yang diharapkan		v		
22	Saat Aplikasi ini dijalankan tidak terdapat <i>error</i>			v	
23	Saat Aplikasi ini dijalankan tidak terdapat menu yang tidak berjalan			v	
24	Aplikasi web <i>E-Commerce</i> ini cocok diterapkan pada chili.id		v		
25	Aplikasi ini bersifat <i>user friendly</i> (mudah digunakan)		v		
Total			4	2	

Untuk mendapatkan hasil interpretasi maka rumus yang digunakan adalah :

$\text{Jumlah Total} / (\text{Nilai Skala Likert Tertinggi} * \text{Jumlah Kuesioner}) * 100$

Selain itu peneliti juga melakukan UAT pada bagian frontend admin untuk keperluan manajemen data produk dan customer. Berikut hasil UAT pada tabel 28 yang telah dilakukan.

Tabel 30. UAT pada sistem pengguna admin

PENGUJIAN SISTEM PENGGUNA ADMIN					
NO	PERTANYAAN	SS	S	KS	TS
INTERFACE					
1	Tampilan pertama yang muncul pada <i>dashboard</i> admin ini menarik				v
2	Tampilan menu pada <i>dashboard</i> admin ini menarik		v		
3	Tampilan menu untuk mengelola produk pada <i>dashboard</i> admin ini menarik		v		
4	Sebelum masuk ke sistem tidak harus login terlebih dulu				v
5	Tampilan pada menu produk pada <i>dashboard</i> admin ini menarik		v		
6	Tampilan pada menu customer pada <i>dashboard</i> admin ini menarik		v		
7	Tampilan pada menu order pada <i>dashboard</i> admin ini menarik		v		
8	Visualisasi order customer pada <i>dashboard</i> admin ini menarik			v	
Total			5	1	1
PROSES YANG TERJADI PADA SISTEM					
9	Proses CRUD (Create, Read, Update, Delete) produk sesuai yang diinginkan			v	
10	Proses CRUD (Create, Read, Update, Delete) order sesuai yang diinginkan		v		
11	Proses CRUD (Create, Read, Update, Delete) customer sesuai yang diinginkan		v		
12	Proses tracking order sesuai yang diharapkan				v
13	Proses memperbarui tampilan store sesuai yang diinginkan		v		
Total			3	1	1
FUNGSIONAL SISTEM					
14	Visualisasi order customer pada <i>dashboard</i> ini sesuai yang diharapkan			v	

15	<i>Dashboard</i> admin ini sesuai dengan yang diharapkan			v	
16	Saat <i>dashboard</i> admin ini dijalankan tidak terdapat menu yang tidak berjalan		v		
17	Saat aplikasi ini dijalankan tidak terdapat <i>error</i>			v	
18	<i>Dashboard</i> admin ini cocok diterapkan pada chili.id		v		
19	<i>Dashboard</i> admin ini bersifat user friendly (mudah digunakan)		v		
Total			3	3	

Tabel 31. Nilai Persentase

Rentang Nilai	Keterangan
81 – 100	Sangat Baik
61 – 80	Baik
41 – 60	Cukup
21 – 40	Kurang
0 – 20	Sangat Kurang

Tabel 31 merupakan nilai persentase dari hasil akhir penilaian kuisisioner UAT dari responden. Berdasarkan nilai ini akan diketahui hasil pengujian akhir aplikasi web apakah aplikasi ini menghasilkan hasil yang baik atau tidak.

Dari hasil pengujian dengan menggunakan metode UAT pada bagian sistem pengguna user pada tabel 29 yang dilakukan oleh product owner maka dapat disimpulkan bahwa bahwa interface yang telah dibuat 71%, proses yang terjadi pada sistem 82% dan fungsional keseluruhan sistem 90%. Sehingga dapat disimpulkan bahwa hasil pengujian dengan UAT sistem pengguna (*user*) ini untuk interface sudah baik, proses dan fungsional keseluruhan aplikasi sudah sangat baik.

Selain itu, berdasarkan hasil pengujian dengan menggunakan metode UAT sistem admin pada tabel 30 maka didapatkan hasil bahwa interface pada admin yang telah dibuat 56%, proses yang terjadi pada sistem admin 75% dan fungsional keseluruhan sistem pada *dashboard* admin 62%. Sehingga dapat disimpulkan

bahwa hasil pengujian dengan UAT ini untuk *interface* sudah cukup, proses pada web berjalan dengan baik, dan fungsional keseluruhan aplikasi sudah baik.

Sehingga berdasarkan kedua hasil UAT ini secara fungsional sistem ini dapat diterima oleh *product owner* dan aplikasi dapat digunakan sebagai media transaksi jual-beli *online* pada chili.id.

BAB VI

SIMPULAN DAN SARAN

6.1 Simpulan

Berdasarkan hasil analisis, perancangan hingga pengujian yang telah dilakukan maka didapatkan kesimpulan sebagai berikut:

- Sistem *E-Commerce* dapat diimplementasikan ke dalam aplikasi web chili.id dengan memanfaatkan boilerplate cezerin2 yang telah disesuaikan dengan *requirement* dari chili.id.
- Hasil pengujian terhadap 14 *product backlog* yang diuji dengan menggunakan metode pengujian Black Box terdapat 12 *product backlog* yang berhasil diuji dan sistem menghasilkan respon yang sesuai dengan yang diharapkan, sedangkan 2 *product backlog* tidak berhasil diuji karena masih terdapat *error* pada sistem.
- Hasil pengujian dengan menggunakan UAT maka dapat disimpulkan bahwa aplikasi web *E-Commerce* dapat diterima oleh *product owner* dengan nilai persentase yang sangat baik yaitu 90% untuk fungsional keseluruhan sistem pada bagian sistem pengguna *user*, sedangkan untuk fungsional keseluruhan sistem pada *dashboard* admin mendapatkan nilai persentase yang baik yaitu 62%. Sehingga dapat disimpulkan bahwa web *E-Commerce* yang telah dibuat dapat digunakan sebagai media transaksi jual-beli produk-produk chili.id secara *online*.

6.2 Saran

Peneliti mengembangkan aplikasi web *E-Commerce* dengan memanfaatkan boilerplate cezerin2, dimana code dan logic yang ada pada boilerplate sangat rumit untuk dimodifikasi sesuai kebutuhan, sehingga beberapa produk *backlog* tidak dapat diselesaikan. Hal ini dapat dilihat dari kesimpulan hasil UAT dimana masih terdapat beberapa hasil implementasi yang kurang sesuai dengan *requirement* dari product owner.

Oleh Karena itu, peneliti menyarankan pada penelitian selanjutnya sebaiknya tidak menggunakan boilerplate apabila pemahaman terkait teknologi yang terdapat pada *boilerplate* masih kurang.

DAFTAR PUSTAKA

- [1] Chili Indonesia, CHILI.ID, Jakarta, 2019, p. 3.
- [2] G. Gumelar, “Bank Indonesia Sebut 'Pedasnya' Cabai Jadi Biang Inflasi,” *cnnindonesia*, 03 12 2016. [Online]. Available: <https://www.cnnindonesia.com/ekonomi/20161203144931-78-177112/bank-indonesia-sebut-pedasnya-cabai-jadi-biang-inflasi>. [Diakses 21 07 2019].
- [3] G. Amanda, “Harga Cabai Petani Anjlok, Ekonom: Distributor Untung Besar,” *republika*, 19 04 2019. [Online]. Available: <https://republika.co.id/berita/ekonomi/pertanian/pq7mug383/harga-cabai-petani-anjlok-ekonom-distributor-untung-besar>. [Diakses 21 07 2019].
- [4] Kompas.com, “Jumlah Pembeli "Online" Indonesia Capai 11,9 Persen dari Populasi,” *Kompas*, 07 09 2018. [Online]. Available: <https://ekonomi.kompas.com/read/2018/09/07/164100326/jumlah-pembeli-online-indonesia-capai-119-persen-dari-populasi>. [Diakses 21 07 2019].
- [5] R. Triwijanarko, “143 Juta Internet *User* di Indonesia, Masih Bisa Ditingkatkan?,” *Marketeers*, 21 02 2018. [Online]. Available: <https://marketeers.com/143-juta-internet-user-di-indonesia56566-2/>. [Diakses 21 07 2019].
- [6] Karmawan, “Analisis Dan Perancangan *E-Commerce* Pd. Garuda Jaya,” dalam *Seminar Nasional Aplikasi Teknologi Informasi 2010 (SNATI 2010)*., 2010.
- [7] “React,” [Online]. Available: <https://id.reactjs.org/>. [Diakses 2019 07 22].
- [8] Raharjo, B. (2019). *Pemograman Web dengan Node.js dan JavaScript*. Bandung: INFORMATIKA.
- [9] K. Schwaber, *Agile project management*, Redmond: WA: Microsoft Press, 2004.

- [10] A. Nuraminah, “Analisis Tingkat Kematangan Manajemen Proyek Pengembangan Perangkat Lunak Menggunakan Scrum Maturity Model: Studi Kasus PT. XYZ,” *STIMIK ESQ / I-4*, vol. 2, pp. 3-4, 2016.
- [11] N. Ruseno, “IMPLEMENTASI SCRUM PADA PENGEMBANGAN APLIKASI SISTEM RESERVASI ONLINE MENGGUNAKAN PHP,” *JURNAL GERBANG*, vol. 9, pp. 9-10, 2019.
- [12] T. Efrain, L. Jae, K. David dan V. Dennis, *Electronic Commerce A Managerial Perspective* 2004, Prentice Hall, 2004.
- [13] K. Schwaber dan M. Beedle, *Agile Software Development with Scrum*, Series in Agile Software Development, Prentice Hall, 2002.
- [14] Shelly, Cashman dan Varmaat, *Discovering Computers: Menjelajah Dunia Komputer Fundamental Edisi 3*, Jakarta: Salemba Infotek, 2007.
- [15] R. Kalakota dan A. B. Whinston, *Electronic Commerce: A Manager’s Guide*. New Jersey: Addison-Wesley Professional., 1997.
- [16] S. Dharwiyanti dan R. S. Wahono, *Pengantar Unified Modeling Language (UML)*, IlmuKomputer.Com, 2003.
- [17] M. S. Mustaqbal, R. F. Firdaus dan H. Rahmadi, “PENGUJIAN APLIKASI MENGGUNAKAN BLACK BOX TESTING BOUNDARY VALUE ANALYSIS (Studi Kasus : Aplikasi Prediksi Kelulusan SNMPTN),” *Jurnal Ilmiah Teknologi Informasi Terapan*, vol. 1, p. 34, 2015.
- [18] B. Simamora, “Skala Pengukuran,” 08 01 2018. [Online]. Available: www.bilsonsiamora.com/skala-pengukuran/. [Diakses 19 07 2019].
- [19] M. Rouli, I. S. Paulus dan F. Ridi, “Usulan Metode Evaluasi *User Acceptance Testing* (UAT) dalam Pengembangan Perangkat Lunak,” dalam *Prosiding Seminar Nasional Pendidikan Teknik Informatika (SENAPATI 2015)*, Singaraja – Bali, 2015.
- [20] agilemodeling.com, “*User Stories: An Agile Introduction*,” www.agilemodeling.com, [Online]. Available:

<http://www.agilemodeling.com/artifacts/userStory.htm>. [Diakses 17 07 2019].

- [21] R. Kalakota dan A. Whinston, "Electronic Commerce: A Manager's Guide. New Jersey: Addison-Wesley Professional.," 1997.