

**PEMBANGUNAN SISTEM PENGELOLAAN BARANG
BERBASIS *WEB*
(STUDI KASUS : PT NIAGARA KOSMETIK)**

Tugas Akhir

**Diajukan untuk Memenuhi Salah Satu Persyaratan Mencapai Derajat
Sarjana Informatika**



Dibuat Oleh:

RIUS SABARNO

160708832

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ATMA JAYA YOGYAKARTA
2020**

HALAMAN PENGESAHAN

Tugas Akhir Berjudul

PEMBANGUNAN SISTEM PENGELOLAAN BARANG BERBASIS WEB

yang disusun oleh

RIUS SABARNO

160708832

dinyatakan telah memenuhi syarat pada tanggal 20 Juli 2020

		Keterangan
Dosen Pembimbing 1	: Joseph Eric Samodra, S.Kom, MIT.	Telah menyetujui
Dosen Pembimbing 2	: Eddy Julianto, ST., MT.	Telah menyetujui
Tim Penguji		
Penguji 1	: Joseph Eric Samodra, S.Kom, MIT.	Telah menyetujui
Penguji 2	: Thomas Adi Purnomo Sidhi, ST., MT.	Telah menyetujui
Penguji 3	: Dr. Pranowo, S.T., M.T.	Telah menyetujui

Yogyakarta, 20 Juli 2020

Universitas Atma Jaya Yogyakarta

Fakultas Teknologi Industri

Dekan

ttd

Dr. A. Teguh Siswanto, M.Sc

PERNYATAAN ORISINALITAS & PUBLIKASI ILMIAH

Saya yang bertanda tangan di bawah ini:

Nama Lengkap : Rius Sabarno
NPM : 160708832
Program Studi : Teknik Informatika
Fakultas : Teknologi Industri
Judul Penelitian : PEMBANGUNAN SISTEM PENGELOLAAN
BARANG BERBASIS *WEB* (STUDI KASUS : PT NIAGARA KOSMETIK)

Menyatakan dengan ini:

1. Tugas Akhir ini adalah benar tidak merupakan salinan sebagian atau keseluruhan dari karya penelitian lain.
2. Memberikan kepada Universitas Atma Jaya Yogyakarta atas penelitian ini, berupa Hak untuk menyimpan, mengelola, mendistribusikan, dan menampilkan hasil penelitian selama tetap mencantumkan nama penulis.
3. Bersedia menanggung secara pribadi segala bentuk tuntutan hukum atas pelanggaran Hak Cipta dalam pembuatan Tugas Akhir ini.

Demikianlah pernyataan ini dibuat dan dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 21 Juli 2020

Yang menyatakan,

Rius Sabarno

160708832

PERNYATAAN PERSETUJUAN DARI INSTANSI ASAL PENELITIAN

Saya yang bertanda tangan di bawah ini:

Nama Lengkap Pembimbing : Saptono Hadi Sutikno
Jabatan : Direktur PT Niagara Kosmetik
Departemen : Direksi

Menyatakan dengan ini:

Nama Lengkap : Rius Sabarno
NPM : 160708832
Program Studi : Teknik Informatika
Fakultas : Teknologi Industri
Judul Penelitian : **PEMBANGUNAN SISTEM PENGELOLAAN
BARANG BERBASIS *WEB* (STUDI KASUS : PT NIAGARA KOSMETIK)**

1. Penelitian telah selesai dilaksanakan pada perusahaan.
2. Perusahaan telah melakukan sidang internal berupa kelayakan penelitian ini dan akan mencantumkan lembar penilaian secara tertutup kepada pihak universitas sebagai bagian dari nilai akhir mahasiswa.
3. Memberikan kepada Instansi Penelitian dan Universitas Atma Jaya Yogyakarta atas penelitian ini, berupa hak untuk menyimpan, mengelola, mendistribusikan, dan menampilkan hasil penelitian selama tetap mencantumkan nama penulis.

Demikianlah pernyataan ini dibuat dan dapat dipergunakan sebagaimana mestinya.

Surakarta, 21 Juli 2020

Yang menyatakan,

Saptono Hadi Sutikno

Direktur

HALAMAN PERSEMBAHAN

Semua akan indah pada waktu-Nya



KATA PENGANTAR

Puji dan syukur penulis haturkan kepada Tuhan Yang Maha Esa karena berkat rahmat dan karunia-Nya penulis dapat menyelesaikan pembuatan tugas akhir “Rancang Bangun Aplikasi Pariwisata Berbasis Lokasi” ini dengan baik.

Penulisan tugas akhir ini bertujuan untuk memenuhi salah satu syarat untuk mencapai derajat sarjana Teknik Informatika dari Program Studi Teknik Informatika, Fakultas Teknologi Industri di Universitas Atma Jaya Yogyakarta.

Penulis menyadari bahwa dalam pembuatan tugas akhir ini penulis telah mendapatkan bantuan, bimbingan, dan dorongan dari banyak pihak. Untuk itu, pada kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Tuhan Yesus Kristus yang selalu membimbing dalam iman-Nya, memberikan berkat-Nya, dan menyertai penulis selalu.
2. Bapak Dr. A. Teguh Siswantoro, selaku Dekan Fakultas Teknologi Industri, Universitas Atma Jaya Yogyakarta.
3. Bapak Joseph Eric S, S.Kom., MIT., selaku dosen pembimbing I yang telah membimbing dan memberikan masukan serta motivasi kepada penulis untuk menyelesaikan tugas akhir ini.
4. Bapak Eddy Julianto, S.T., M.T., selaku dosen pembimbing II yang telah membimbing dan memberikan masukan serta motivasi kepada penulis untuk menyelesaikan tugas akhir ini.

Demikian laporan tugas akhir ini dibuat, dan penulis mengucapkan terima kasih kepada semua pihak. Semoga laporan ini dapat bermanfaat bagi pembaca.

Yogyakarta, 06 Juli 2020

Rius Sabarno

160708832

DAFTAR ISI

LEMBAR PENGESAHAN	Error! Bookmark not defined.
PERNYATAAN ORISINALITAS & PUBLIKASI ILMIAH.....	iii
PERNYATAAN PERSETUJUAN DARI INSTANSI ASAL PENELITIAN	iv
HALAMAN PERSEMBAHAN	v
KATA PENGANTAR	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR	viii
DAFTAR TABEL.....	xiv
INTISARI.....	xv
BAB I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan Masalah.....	22
1.4. Tujuan Penelitian	2
1.5. Metode Penelitian.....	2
1.6. Sistematika Penulisan	4
BAB II. TINJAUAN PUSTAKA.....	5
BAB III. LANDASAN TEORI.....	11
BAB IV. ANALISIS DAN PERANCANGAN SISTEM	17
4.1. Analisis Sistem.....	17
4.2. Lingkup Masalah.....	18
4.3. Perspektif Produk	18
4.4. Fungsi Produk	19
4.5. Kebutuhan Antarmuka	80
4.6. Perancangan	81
4.6.1 Perancangan Arsitektur.....	81
4.6.1 Perancangan Antarmuka.....	84
BAB V. IMPLEMENTASI DAN PENGUJIAN SISTEM	146
5.1. Implementasi Sistem Implementasi Antarmuka	146
5.2. Pengujian Fungsionalitas Perangkat Lunak	185
5.3. Hasil Pengujian Terhadap Pengguna	234
BAB VI. PENUTUP	146
6.1. Kesimpulan	236
6.1. Saran.....	236
DAFTAR PUSTAKA	237
LAMPIRAN.....	239

DAFTAR GAMBAR

Gambar 4. 1. Use Case Diagram aplikasi SIGARA.....	19
Gambar 4. 2. <i>Overview Sistem</i> SIGARA	82
Gambar 4. 3. <i>Package Diagram</i> SIGARA.....	83
Gambar 4. 4. <i>Deployment Diagram</i> SIGARA	84
Gambar 4. 5. Antarmuka Halaman <i>Login</i>	84
Gambar 4. 6. Antarmuka Halaman <i>Owner</i>	85
Gambar 4. 7. Antarmuka Halaman Pegawai.....	86
Gambar 4. 8. Antarmuka Halaman Kelola Data Pengguna	86
Gambar 4. 9. Antarmuka <i>Form</i> Tambah Pengguna	87
Gambar 4. 10. Antarmuka Halaman Aktivitas.....	87
Gambar 4. 11. Antarmuka Halaman Kelola Data Suplier.....	88
Gambar 4. 12. Antarmuka <i>Form</i> Tambah Data Suplier.....	89
Gambar 4. 13. Antarmuka Halaman Cetak Daftar Suplier	89
Gambar 4. 14. Antarmuka Halaman Kelola Data Stok.....	90
Gambar 4. 15. Antarmuka <i>Form</i> Tambah Data Stok.....	90
Gambar 4. 16. Antarmuka Halaman Kelola Data Pembelian	91
Gambar 4. 17. Antarmuka <i>Form</i> Tambah Data Pembelian	92
Gambar 4. 18. Antarmuka Halaman Cetak Laporan Pembelian.....	92
Gambar 4. 19. Antarmuka Halaman Kelola Data Tabel Oplos	93
Gambar 4. 20. Antarmuka <i>Form</i> Tambah Data Tabel Oplos.....	93
Gambar 4. 21. Antarmuka Halaman Kelola Data <i>Entry</i> Oplosan.....	94
Gambar 4. 22. Antarmuka <i>Form</i> Tambah Data Entry Oplosan	95
Gambar 4. 23. Antarmuka <i>Form</i> Posting Oplosan.....	95
Gambar 4. 24. Antarmuka Halaman Mutasi Stok.....	96
Gambar 4. 25. Antarmuka Halaman Kelola Data <i>Entry</i> Produksi.....	96
Gambar 4. 26. Antarmuka <i>Form</i> Tambah Data Entry Produksi	97
Gambar 4. 27. Antarmuka <i>Form</i> Posting Etiket Belakang	98
Gambar 4. 28. Antarmuka Halaman Kelola Data Produksi Harian	98
Gambar 4. 29. Antarmuka <i>Form</i> Cetak Laporan Produksi	99

Gambar 4. 30. Antarmuka Halaman Laporan Produksi.....	99
Gambar 4. 31. Antarmuka <i>Form</i> Cetak Laporan Pemakaian Bahan	100
Gambar 4. 32. Antarmuka Halaman Laporan Pemakaian Bahan	100
Gambar 4. 33. Antarmuka <i>Form</i> Posting Produksi.....	101
Gambar 4. 34. Antarmuka Halaman Cek Stok Per Item	101
Gambar 4. 35. Antarmuka Halaman Kelola Data LRelasi.....	102
Gambar 4. 36. Antarmuka <i>Form</i> Tambah Data LRelasi	103
Gambar 4. 37. Antarmuka Halaman Kelola Data Jenis Pelanggan	103
Gambar 4. 38. Antarmuka <i>Form</i> Tambah Data Jenis Pelanggan.....	104
Gambar 4. 39. Antarmuka Halaman Kelola Data Spesifikasi Tabel.....	104
Gambar 4. 40. Antarmuka <i>Form</i> Tambah Data Spesifikasi Tabel.....	105
Gambar 4. 41. Antarmuka Halaman Kelola Data BG.....	106
Gambar 4. 42. Antarmuka <i>Form</i> Tambah Data BG.....	106
Gambar 4. 43. Antarmuka Halaman Kelola Data Entry Order	107
Gambar 4. 44. Antarmuka <i>Form</i> Tambah Data Entry Order	108
Gambar 4. 45. Antarmuka <i>Form</i> Entry Nota Kanvas	108
Gambar 4. 46. Antarmuka <i>Form</i> Entry Pengiriman Barang	109
Gambar 4. 47. Antarmuka <i>Form</i> Cetak Nota Satuan	110
Gambar 4. 48. Antarmuka <i>Form</i> Cetak Nota Rayon	110
Gambar 4. 49. Antarmuka <i>Form</i> Cetak Nota Semua	111
Gambar 4. 50. Antarmuka <i>Form</i> Entry Tanggal Nota	111
Gambar 4. 51. Antarmuka <i>Form</i> Cetak Nota Ulang	112
Gambar 4. 52. Antarmuka <i>Form</i> Entry Estimasi Kanvas	112
Gambar 4. 53. Antarmuka <i>Form</i> Entry Barang Dibawa Kanvas	113
Gambar 4. 54. Antarmuka <i>Form</i> Entry Barang Kembali Kanvas.....	113
Gambar 4. 55. Antarmuka <i>Form</i> Cetak Daftar Barang Dibawa Kanvas	114
Gambar 4. 56. Antarmuka Halaman Daftar Barang Dibawa Kanvas	114
Gambar 4. 57. Antarmuka <i>Form</i> Cetak Laporan Kanvas	115
Gambar 4. 58. Antarmuka Halaman Laporan Kanvas	115
Gambar 4. 59. Antarmuka <i>Form</i> Posting Kanvas	116
Gambar 4. 60. Antarmuka <i>Form</i> Entry Surat Jalan	116

Gambar 4. 61. Antarmuka <i>Form</i> Cetak Surat Jalan	117
Gambar 4. 62. Antarmuka Halaman Surat Jalan.....	117
Gambar 4. 63. Antarmuka <i>Form Entry</i> Pengiriman Surat Jalan	118
Gambar 4. 64. Antarmuka <i>Form</i> Cetak Kwitansi	118
Gambar 4. 65. Antarmuka Halaman Kwitansi	119
Gambar 4. 66. Antarmuka Halaman Kelola Data Pembayaran Pengiriman	119
Gambar 4. 67. Antarmuka <i>Form</i> Tambah Data Pembayaran Pengiriman	120
Gambar 4. 68. Antarmuka Halaman Kelola Data Berkas Pengiriman.....	120
Gambar 4. 69. Antarmuka <i>Form</i> Tambah Data Berkas Pengiriman.....	121
Gambar 4. 70. Antarmuka <i>Form</i> Cetak Daftar Barang Dikirim	122
Gambar 4. 71. Antarmuka Halaman Daftar Barang Dikirim	122
Gambar 4. 72. Antarmuka Halaman Kelola Data Batasan Faktur PPN.....	123
Gambar 4. 73. Antarmuka <i>Form</i> Tambah Data Batasan Faktur PPN.....	123
Gambar 4. 74. Antarmuka <i>Form</i> Cetak Faktur Satuan	124
Gambar 4. 75. Antarmuka <i>Form</i> Cetak Faktur Rayon.....	124
Gambar 4. 76. Antarmuka <i>Form</i> Cetak Faktur Semua	125
Gambar 4. 77. Antarmuka <i>Form</i> Posting Faktur PPN	125
Gambar 4. 78. Antarmuka <i>Form</i> Entry Kanvas Faktur.....	126
Gambar 4. 79. Antarmuka <i>Form</i> Cetak Laporan Kanvas Faktur	125
Gambar 4. 80. Antarmuka Halaman Laporan Kanvas Faktur.....	127
Gambar 4. 81. Antarmuka Halaman Kelola Data Hutang	127
Gambar 4. 82. Antarmuka <i>Form</i> Tambah Data Pembayaran Hutang.....	128
Gambar 4. 83. Antarmuka <i>Form</i> Detail Data Pembayaran Hutang	128
Gambar 4. 84. Antarmuka <i>Form</i> Posting Hutang Akhir Bulan	129
Gambar 4. 85. Antarmuka Halaman Kelola Data Piutang Dagang.....	129
Gambar 4. 86. Antarmuka <i>Form</i> Detail Data Pembayaran Piutang.....	130
Gambar 4. 87. Antarmuka Halaman Kelola Data Piutang Kliring	131
Gambar 4. 88. Antarmuka Halaman Kelola Data Berkas Penagihan Piutang	131
Gambar 4. 89. Antarmuka <i>Form</i> Tambah Data Berkas Penagihan.....	132
Gambar 4. 90. Antarmuka <i>Form</i> Cetak Daftar Tagih	132
Gambar 4. 91. Antarmuka Halaman Daftar Penagihan.....	133

Gambar 4. 92. Antarmuka Halaman Kelola Data Kembali Penagihan.....	133
Gambar 4. 93. Antarmuka <i>Form</i> Tambah Data Kembali Penagihan.....	134
Gambar 4. 94. Antarmuka <i>Form</i> Cetak Laporan Kelilingan	135
Gambar 4. 95. Antarmuka Halaman Laporan Kelilingan	135
Gambar 4. 96. Antarmuka <i>Form</i> Tambah Data Pembayaran Penagihan.....	136
Gambar 4. 97. Antarmuka Halaman Cetak Label Karton.....	136
Gambar 4. 98. Antarmuka Halaman Label Karton	137
Gambar 4. 99. Antarmuka Halaman Cetak Label Burenan MW Kecil	137
Gambar 4. 100. Antarmuka Halaman Label Burenan MW Kecil.....	138
Gambar 4. 101. Antarmuka Halaman Cetak Label Burenan MW EDC Besar	138
Gambar 4. 102. Antarmuka Halaman Label Burenan MW EDC Besar	139
Gambar 4. 103. Antarmuka Halaman Cetak Label Kemiri KM	139
Gambar 4. 104. Antarmuka Halaman Label Kemiri KM	140
Gambar 4. 105. Antarmuka Halaman Cetak Label Kemiri KQ.....	140
Gambar 4. 106. Antarmuka Halaman Label Kemiri KQ	141
Gambar 4. 107. Antarmuka Halaman Cetak Label MO Besar	141
Gambar 4. 108. Antarmuka Halaman Label MO Besar.....	142
Gambar 4. 109. Antarmuka Halaman Cetak Label MO Kecil.....	142
Gambar 4. 110. Antarmuka Halaman Label MO Kecil	143
Gambar 4. 111. Antarmuka Halaman Cetak Label MR Besar.....	143
Gambar 4. 112. Antarmuka Halaman Label MR Besar	144
Gambar 4. 113. Antarmuka Halaman Cetak Label MR Kecil	144
Gambar 4. 114. Antarmuka Halaman Label MR Kecil	145
Gambar 5. 1. Antarmuka Halaman Kelola Data Transaksi Pembelian.....	147
Gambar 5. 2. Antarmuka <i>Form</i> Tambah Data Transaksi Pembelian.....	147
Gambar 5. 3. Antarmuka Tampilan Total Pembelian PPN.....	148
Gambar 5. 4. Antarmuka Tampilan Total Pembelian NON PPN	148
Gambar 5. 5. Antarmuka <i>Form</i> Detail Transaksi Pembelian.....	149
Gambar 5. 6. Antarmuka <i>Form</i> Ubah Data Transaksi Pembelian	150
Gambar 5. 7. Tampilan Pesan Proses Hapus Berhasil	150
Gambar 5. 8. Fungsi Tampil Semua Data Pembelian Bagian <i>Frontend</i>	151

Gambar 5. 9. Fungsi Tampil Semua Data Pembelian Bagian <i>Backend</i>	151
Gambar 5. 10. Fungsi Simpan Data Transaksi Pembelian Bagian <i>Frontend</i>	152
Gambar 5. 11. Fungsi Simpan Data Transaksi Pembelian Bagian <i>Backend</i>	153
Gambar 5. 12. Fungsi Simpan Data Detail Pembelian Bagian <i>Backend</i>	154
Gambar 5. 13. Fungsi Simpan Data Hutang Bagian <i>Backend</i>	154
Gambar 5. 14. Fungsi Hapus Data Transaksi Pembelian Bagian <i>Frontend</i>	155
Gambar 5. 15. Fungsi Hapus Data Transaksi Pembelian Bagian <i>Backend</i>	155
Gambar 5. 16. Antarmuka Halaman Kelola Data <i>Entry</i> Oplosan.....	157
Gambar 5. 17. Antarmuka <i>Form</i> Tambah Data <i>Entry</i> Oplosan	157
Gambar 5. 18. Antarmuka <i>Form</i> Posting Oplosan.....	157
Gambar 5. 19. Fungsi Simpan Data <i>Entry</i> Oplosan Bagian <i>Frontend</i>	158
Gambar 5. 20. Fungsi Simpan Data <i>Entry</i> Oplosan Bagian <i>Backend</i>	159
Gambar 5. 21. Fungsi Posting Oplosan Bagian <i>Frontend</i>	160
Gambar 5. 22. Fungsi Posting Oplosan Bagian <i>Backend</i>	162
Gambar 5. 23. Antarmuka Halaman Kelola Data <i>Entry</i> Produksi	163
Gambar 5. 24. Antarmuka <i>Form</i> Tambah Data <i>Entry</i> Produksi	164
Gambar 5. 25. Antarmuka <i>Form</i> Posting Etiket Belakang	164
Gambar 5. 26. Fungsi Simpan Data <i>Entry</i> Produksi <i>Frontend</i> Bagian 1	165
Gambar 5. 27. Fungsi Simpan Data <i>Entry</i> Produksi <i>Frontend</i> Bagian 2.....	165
Gambar 5. 28. Fungsi Simpan Data <i>Entry</i> Produksi Bagian <i>Backend</i>	166
Gambar 5. 29. Fungsi Posting Etiket Belakang Bagian <i>Frontend</i>	167
Gambar 5. 30. Fungsi Posting Etiket Belakang Bagian <i>Backend</i>	167
Gambar 5. 31. Antarmuka Halaman Kelola Data <i>Entry</i> Order	169
Gambar 5. 32. Antarmuka <i>Form</i> Pilih Kode Daerah	169
Gambar 5. 33. Antarmuka <i>Form</i> Tambah Data <i>Entry</i> Order.....	169
Gambar 5. 34. Fungsi Memilih Kode Daerah Bagian <i>Frontend</i>	170
Gambar 5. 35. Fungsi Memilih Kode Daerah Bagian <i>Backend</i>	170
Gambar 5. 36. Fungsi Simpan Data <i>Entry</i> Order Bagian <i>Frontend</i>	171
Gambar 5. 37. Fungsi Simpan Data <i>Entry</i> Order <i>Backend</i> Bagian 1	172
Gambar 5. 38. Fungsi Simpan Data <i>Entry</i> Order <i>Backend</i> Bagian 2	172
Gambar 5. 39. Antarmuka Halaman Kelola Data Pembayaran Pengiriman	173

Gambar 5. 40. Antarmuka <i>Form</i> Tambah Data Pembayaran Pengiriman	174
Gambar 5. 41. Antarmuka <i>Form</i> Detail Barang Pada Nota	174
Gambar 5. 42. Fungsi Simpan Data Pembayaran Pengiriman Bagian <i>Frontend</i> .	175
Gambar 5. 43. Fungsi Simpan Data Pembayaran Pengiriman Bagian <i>Backend</i> ..	176
Gambar 5. 44. Antarmuka Halaman Kelola Data Hutang	177
Gambar 5. 45. Antarmuka <i>Form</i> Tambah Data Pembayaran Hutang.....	178
Gambar 5. 46. Antarmuka <i>Form</i> Detail Data Pembayaran Hutang	179
Gambar 5. 47. Antarmuka <i>Form</i> Posting Hutang Akhir Bulan	179
Gambar 5. 48. Fungsi Simpan Data Pembayaran Hutang Bagian <i>Frontend</i>	180
Gambar 5. 49. Fungsi Simpan Data Pembayaran Hutang Bagian <i>Backend</i>	180
Gambar 5. 50. Fungsi Posting Hutang Akhir Bulan Bagian <i>Frontend</i>	181
Gambar 5. 51. Fungsi Posting Hutang Akhir Bulan Bagian <i>Backend</i>	181
Gambar 5. 52. Antarmuka Halaman Kelola Data Piutang	182
Gambar 5. 53. Antarmuka <i>Form</i> Detail Data Pembayaran Piutang.....	183
Gambar 5. 54. Antarmuka <i>Form</i> Posting Piutang Akhir Bulan.....	183
Gambar 5. 55. Fungsi Posting Piutang Akhir Bulan Bagian <i>Frontend</i>	183
Gambar 5. 56. Fungsi Posting Piutang Akhir Bulan Bagian <i>Backend</i>	184

DAFTAR TABEL

Tabel 2. 1. Perbandingan Penelitian.....	8
Tabel 5. 1. Pengujian Fungsionalitas Perangkat Lunak	Error! Bookmark not defined.
Tabel 5. 2. Hasil Pengujian Terhadap Pengguna Pakar	Error! Bookmark not defined.



INTISARI

PEMBANGUNAN SISTEM PENGELOLAAN BARANG BERBASIS *WEB* (STUDI KASUS: PT NIAGARA KOSMETIK)

Intisari

Rius Sabarno

160708832

PT Niagara kosmetik merupakan perusahaan yang berfokus pada pembuatan produk-produk kosmetik, dengan penyebaran produk hampir ke seluruh wilayah di pulau Jawa. Oleh karena itu perusahaan memerlukan sebuah sistem untuk mengelola data barang mulai dari pembelian, produksi, penjualan, hutang dan piutang. Pada saat ini untuk melakukan proses pengelolaan data barang tersebut, PT Niagara Kosmetik menggunakan sistem berbasis desktop Visual Basic dan Microsoft Access 2003. Pihak perusahaan merasa sistem yang sekarang berjalan masih kurang maksimal karena berbasis desktop dan hanya bisa dijalankan pada sistem operasi tertentu.

Penelitian ini memberikan solusi berupa sistem berbasis *website* menggunakan bahasa pemrograman PHP. *Website* dibangun menggunakan *framework* Laravel pada bagian *backend* dan *framework* Vue JS pada bagian *frontend* serta MySQL untuk keperluan basis data. Pada penelitian ini, sistem memberikan akses hanya pada pihak yang sudah terdaftar pada sistem. Hasil dari penerapan sistem pengelolaan barang berbasis *website* yaitu perusahaan dapat melakukan proses pengelolaan barang mulai dari pembelian, produksi, penjualan, hutang dan piutang.

Kata Kunci: PT Niagara Kosmetik, Laravel, MySQL, Vue JS

Dosen Pembimbing I : Joseph Eric Samodra, S.Kom., MIT.

Dosen Pembimbing II : Eddy Julianto, S.T., M.T.

Jadwal Sidang Tugas Akhir : 20 Juli 2020

BAB I. PENDAHULUAN

1.1. Latar Belakang

PT Niagara Kosmetik merupakan sebuah perusahaan kosmetik yang berada di kota Surakarta. Perusahaan ini sudah berdiri lebih dari 60 tahun dan menghasilkan produk kosmetik yang berkualitas[1]. Produk kosmetik yang dihasilkan oleh perusahaan antara lain minyak orang aring yang masih menjadi produk unggulan perusahaan sampai saat ini. Produk kosmetik yang dihasilkan di PT Niagara Kosmetik saat ini distribusikan hampir di seluruh daerah di pulau jawa dan beberapa kabupaten di Yogyakarta.

Dengan luasnya pemasaran yang ada, perusahaan memanfaatkan sistem untuk melakukan pengelolaan barang seperti pembelian, produksi, penjualan, hutang dan piutang. Sesuai dengan surat yang dikirimkan pihak perusahaan kepada Universitas Atma Jaya Yogyakarta, saat ini perusahaan masih menggunakan sistem yang lama yaitu Visual Basic dan Microsoft Access 2003. Sistem ini memiliki fitur terbatas untuk dapat melakukan pengelolaan barang. Dari permasalahan tersebut, perusahaan membutuhkan aplikasi yang dapat melakukan pengelolaan barang sehingga mempermudah perusahaan dalam mengelola barang secara lebih baik[2].

Sistem akan dibangun di *platform web*, karena sistem yang berbasis *web* memiliki banyak kelebihan. Kelebihan ini yaitu pengguna tidak perlu menginstalasi program pada komputer mereka seperti aplikasi desktop pada umumnya melainkan pengguna hanya perlu membuka *browser* untuk mengakses aplikasi. Selain itu sistem berbasis *web* memiliki fleksibilitas waktu dalam pengaksesannya. Oleh karena itu, pengguna dapat mengakses sistem ini dimana saja dan kapan saja bahkan dapat mengaksesnya dengan menggunakan *smartphone* mereka sendiri[3].

Sesuai dengan permintaan perusahaan, pembangunan aplikasi ini menggunakan bahasa pemrograman PHP dengan memanfaatkan *framework* Laravel. Ada beberapa keuntungan menggunakan *framework* Laravel untuk pengembangan *web* diantaranya menggunakan *framework* Laravel dapat membuat

proses pengembangan sangat cepat, dari ide hingga rilis peluncuran. [4].

Hasil penelitian ini yaitu sebuah aplikasi *web* untuk melakukan pengelolaan barang pada PT Niagara Kosmetik antara lain melakukan pengelolaan pembelian, produksi, penjualan, hutang dan piutang.

1.2. Rumusan Masalah

Berdasarkan latar belakang, maka didapatkan rumusan masalah yaitu bagaimana membangun sistem berbasis *web* yang bisa membantu PT Niagara Kosmetik melakukan proses pengelolaan barang antara lain pembelian, produksi, penjualan, hutang dan piutang.

1.3. Batasan Masalah

Dalam proses pembangunan sistem ini, ditetapkan batasan masalah sebagai berikut:

1. Menggunakan bahasa pemrograman PHP dengan memanfaatkan *framework* Laravel versi 6 untuk bagian *backend*
2. Menggunakan bahasa pemrograman javascript dengan memanfaatkan *framework* Vue Js versi 4.0.5

1.4. Tujuan Penelitian

Tujuan dari dilakukannya penelitian ini yaitu mampu membangun sistem berbasis *web* yang dapat membantu PT Niagara Kosmetik untuk mengelola yang ada diantaranya pembelian, produksi, penjualan, hutang dan piutang dengan baik.

1.5. Metode Penelitian

1. Metode Studi Pustaka

Metode digunakan untuk mengumpulkan data yang dijadikan acuan dalam proses penelitian antara lain buku, jurnal, dan media daring.

2. Wawancara

Wawancara dilakukan agar dapat mengumpulkan seluruh informasi mengenai kebutuhan sistem. Pada penelitian ini, wawancara dilakukan dengan Saptono Hadi Suktino selaku direktur utama PT Niagara Kosmetik.

3. Analisis

Analisis kebutuhan perangkat lunak dilakukan setelah mengumpulkan keseluruhan kebutuhan berdasarkan hasil wawancara. Analisis kebutuhan perangkat lunak berupa fungsi produk, kebutuhan antarmuka dan kebutuhan data.

4. Perancangan Perangkat Lunak

Perancangan perangkat lunak yaitu berupa perancangan arsitektur dan perancangan antarmuka. Perancangan dilakukan berdasarkan hasil analisis yang dilakukan sebelumnya.

5. Pengkodean

Pengkodean adalah tahap dimana dimulainya pembangunan sistem dengan bahasa pemrograman PHP dan *framework* Laravel serta Vue JS. Hasil dari tahap ini berupa sistem pengelolaan barang PT Niagara Kosmetik berbasis *web*.

6. Pengujian dan Evaluasi

Pengujian dilakukan ketika sistem sudah selesai dibangun dan dilakukan pengujian terhadap fungsi-fungsi pada sistem. tahap pengujian dilakukan untuk menemukan bug dan pengecekan terhadap kelengkapan sistem.

7. Penulisan Laporan

Penulisan laporan merupakan langkah terakhir dari penelitian. Pada tahap akhir ini akan dibuat laporan tentang hasil akhir dari keseluruhan sistem. Penulisan ini bertujuan untuk dapat digunakan sebagai bahan referensi untuk penelitian yang serupa.

1.6. Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang masalah, rumusan masalah, Batasan masalah, tujuan penelitian, metode yang digunakan selama penelitian, serta sistematika penulisan yang digunakan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi penelitian-penelitian terdahulu yang menyangkut dengan penelitian yang dilakukan. Terdapat juga tabel perbandingan antara penelitian yang dilakukan dengan penelitian-penelitian terdahulu.

BAB III LANDASAN TEORI

Bab ini membahas mengenai uraian dasar teori yang akan digunakan dalam melakukan pengembangan dan pembuatan program meliputi referensi teknik pengembangan *web* dan referensi mengenai pengelolaan barang.

BAB IV ANALISIS DAN PERANCANGAN

Bab ini berisi penjelasan mengenai tahap-tahap perancangan perangkat lunak yang akan dibuat, serta desain sistem yang akan digunakan dalam proses pengembangan.

BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM

Bab ini berisi tentang implementasi antarmuka dari sistem yang dikembangkan, pengujian fungsionalitas perangkat lunak, serta hasil yang dilakukan terhadap pengguna.

BAB VI PENUTUP

Bab ini berisi kesimpulan dari pembahasan tugas akhir secara keseluruhan dan saran untuk pengembangan lebih lanjut.

BAB II. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Pada tahun 2017 Wandi melakukan pengembangan aplikasi pengolahan data barang berbasis *web* pada PD. Alfa Multi Usaha. Aplikasi ini dikembangkan dengan tujuan untuk mempermudah pihak PD. Alfa Multi Usaha dalam melakukan pengelolaan terhadap data barang yang ada. Pengelolaan data barang yang dapat dilakukan oleh sistem ini antara lain mengolah data barang masuk, barang keluar, data karyawan, data *distributor*, data pengguna, laporan barang masuk dan laporan barang keluar. Sistem ini juga memiliki sistem keamanan dimana pengguna dengan *role* karyawan hanya dapat melihat stok barang dan melakukan proses transaksi barang keluar. Pengguna dengan *role* pimpinan juga dibatasi hanya dapat melihat data karyawan, stok barang dan dapat mencetak data seperti barang masuk dan barang keluar sedangkan pengguna dengan *role* administrator di beri hak akses untuk menambah, menghapus, mengubah, dan mencetak laporan data barang. Sistem ini dikembangkan menggunakan bahasa pemrograman PHP dan menggunakan basis data MySQL[5].

Nur Aristania dan Indah Uly Wardati pada tahun 2017 juga mengembangkan sistem pengelolaan barang dan data proyek gypsum pada Utomo Gypsum. Sistem ini dibuat dengan tujuan untuk membuat dan menghasilkan perangkat lunak yang dapat membantu proses pengelolaan persediaan barang dan data proyek gypsum serta mempercepat waktu *input* dan *output* data. Pengelolaan data yang dapat dilakukan oleh sistem antara lain pengelolaan data barang dan data proyek. Kemudian sistem juga mempunyai kemampuan untuk melakukan pengelolaan terhadap data transaksi penjualan. Sistem juga dapat mencetak laporan yang di butuhkan oleh pengguna antara lain laporan perhari, perbulan, dan perperiode penjualan barang serta laporan data proyek. Sistem ini dikembangkan menggunakan Bahasa pemrograman PHP dan menggunakan basis data MySQL 4.1[6].

Andrini Hanariana, Riza Agustiansyah, dan Rachmadita Andreswari pada tahun 2015 juga membangun sebuah sistem berbasis *web* dengan menggunakan metode *waterfall* untuk melakukan pengelolaan data PO BOX PT Pos Indonesia Bandung. Pengelolaan data yang dimaksud antara lain pengelolaan data penyewa, data operator, data nomor PO BOX, data surat masuk, data surat keluar, data pengumuman dan data pesan. Sistem juga memiliki kemampuan untuk mengelola laporan PO BOX seperti laporan jumlah penyewa dan laporan jumlah surat yang tersimpan dan terambil. Sistem juga memiliki kemampuan untuk melihat grafik jumlah penyewa PO BOX per tahun berdasarkan jenis penyewa dan grafik jumlah surat tersimpan dan terambil. Sistem informasi pengolahan data PO BOX ini dirancang agar memperoleh proses yang lebih cepat, akurat dan efisien dari segi waktu dan biaya. Sistem ini dibangun menggunakan bahasa pemrograman PHP dan basis data MySQL [7].

Muhdar Abdurahman pada tahun 2019 juga membangun sebuah sistem berbasis *web* untuk melakukan pengolahan data pembelian dan penjualan pada toko Koloncucu Ternate. Toko Koloncucu merupakan salah satu toko di kota Ternate yang bergerak dalam bidang penjualan peralatan rumah tangga. Proses pengolahan data pembelian dan penjualan pada toko ini masih dilakukan secara manual. Sistem ini memiliki kemampuan untuk melakukan pengolahan data pembelian antara lain data supplier, pengiriman, tanggal jatuh tempo pembayaran dan data barang. Sistem juga memiliki kemampuan untuk melakukan pengolahan data penjualan antara lain data transaksi penjualan dan stok barang. Sistem ini diharapkan dapat menunjang kinerja dan pelayanan pada toko. Sistem ini dibangun menggunakan bahasa pemrograman PHP dan basis data MySQL[8].

Nurul Huda dan Rahayu Amalia pada tahun 2020 juga membangun sebuah sistem informasi inventaris barang pada PT PLN Persero Palembang. pengolahan data inventaris pada PT PLN Persero Palembang masih sering mengalami banyak kendala seperti sering terjadi kesulitan dalam penyajian laporan data barang inventaris. Hal ini mengakibatkan laporan inventaris menjadi tidak efisien. Disamping itu, pengolahan data yang ada masih dilakukan secara manual sehingga pembuatan laporannya sering terlambat dan juga sering terjadi duplikasi. Sistem ini

memiliki kemampuan untuk melakukan pengolahan data bagian. Sistem juga memiliki kemampuan untuk melakukan pengolahan laporan inventaris antara lain laporan inventaris bagian administrasi, bagian teknik, bagian pelayanan pelanggan, bagian transaksi energi dan bagian penagihan. Sistem ini dibangun menggunakan bahasa pemrograman PHP dan basis data MySQL[9].



Tabel 2.1. Perbandingan Penelitian

No	Pembanding	Wandi[5]	Nur Aristania, Indah Uly Wardati[6]	Andrini Hanariana, Riza Agustiansyah, Rachmadita Andreswari[7]	Muhdar Abdurahman[8]	Nurul Huda,Rahayu Amalia[9]	*Penulis
1	Judul	Sistem Informasi Pengelolaan Data Barang Berbasis <i>Web</i> Pada PD. Alfa Multi Usaha Pontianak	Pembuatan Sistem Informasi Pengelolaan Persediaan Barang Dan Data Proyek Gypsum Pada Utomo Gypsum	Membangun Sistem Informasi Pengolahan Data Po Box Berbasis <i>Web</i> Menggunakan Metode Waterfall Studi Kasus PT. Pos Indonesia Bandung	Sistem Informasi Pengolahan Data Pembelian Dan Penjualan Pada Toko Koloncucu Ternate	Impementasi Sistem Informasi Inventaris Barang Pada PT. PLN (Persero) Palembang	Pembangunan Sistem Pengelolaan Barang Berbasis <i>Web</i> (Studi Kasus: PT Niagara Kosmetik)
2	Permasalahan	Pencatatan dan	Pengelolaan	Sistem penyewaan	Pendataan pembelian	Pengolahan	Proses

		pengolahan data barang masih dilakukan secara manual	persediaan barang dan data proyek gypsum masih dilakukan secara manual	PO BOX di kantor pos masih dilakukan manual	dan penjualan barang masih dilakukan secara manual	data inventaris masih dilakukan secara manual	pengelolaan pembelian, produksi, penjualan, hutang dan piutang masih menggunakan sistem yang sederhana
3	<i>Platform</i>	<i>Website</i>	<i>Website</i>	<i>Website</i>	<i>Website</i>	<i>Website</i>	<i>Website</i>
4	Bahasa pemrograman	PHP	PHP	PHP	PHP	PHP	PHP
5	Basis data	MySQL	MySQL	MySQL	MySQL	MySQL	MySQL

BAB III. LANDASAN TEORI

3.1. World Wide Web

World wide web adalah sistem yang saling terkait mengenai dokumen yang diakses melalui Internet. Dengan *browser web*, seseorang dapat melihat halaman *web* yang mungkin berisi teks, gambar, video, dan multimedia lainnya dan navigasi di antara mereka melalui *hyperlink*. Pada 12 Maret 1989, Tim Berners - Lee, seorang ilmuwan komputer Inggris menulis proposal yang pada akhirnya menjadi *world wide web*. Pada tahun 1989 Berners-Lee akhirnya menyadari konsep itu bisa diterapkan di seluruh dunia. Berners - Lee dan ilmuwan komputer Belgia Robert Cailliau mengusulkan pada tahun 1990 untuk menggunakan *hypertext* “untuk menghubungkan dan mengakses informasi dari berbagai jenis sebagai *web* node di mana pengguna dapat menelusuri sesuka hati ”[12]. Dengan cara ini layanan *web* pertama dirancang dan diuji dan akhirnya dikenal sebagai *world wide web*.

3.2. PHP

PHP merupakan bahasa *scripting open source* yang banyak digunakan dan sangat cocok untuk pengembangan *web* dan dapat disematkan ke dalam *sintaks* HTML. Perbedaan PHP dengan bahasa pemrograman untuk *web* lainnya seperti JavaScript adalah bahwa JavaScript merupakan bahasa *client-side*, Sedangkan PHP adalah *server-side* dengan kata lain kode PHP dieksekusi di *server* dan menghasilkan HTML yang kemudian dikirim ke klien. Klien akan menerima hasil menjalankan skrip itu, tetapi tidak akan tahu apa kode yang mendasarinya. Kita bahkan dapat mengkonfigurasi *server web* untuk memproses semua file HTML dengan PHP, dan kemudian benar-benar tidak ada cara bagi pengguna untuk mengetahui apa yang kita miliki[13].

3.3. Framework

Framework pertama memperkenalkan *template* untuk menyederhanakan

pembuatan HTML. *Template* adalah dokumen HTML yang telah dianotasi dengan kode di bahasa *framework*. Untuk menghasilkan halaman *web*, *template* diperluas dengan mengganti anotasi dengan HTML dihitung oleh kode. Kode dalam anotasi dapat terdiri dari substitusi *variabel* sederhana, lebih banyak pernyataan kompleks yang menghitung konten, atau bahkan struktur perulangan yang mereplikasi bagian dari *template* (seperti baris tabel). *Template* telah menjadi inti dari hampir semua *framework web*[14].

Beberapa tahun terakhir telah terlihat kedatangan generasi kedua *framework web*. *Framework* ini menyediakan Fasilitas ORM (Object Relational Mapping) yang menyederhanakan penggunaan basis data relasional dalam aplikasi *web*. Salah satu contoh terbaik ORM disediakan oleh *framework* Rails. Rails mengaitkan setiap tabel basis data dengan kelas dalam bahasa Ruby yang mendasarinya, dengan satu objek kelas untuk setiap baris dalam a tabel dan satu bidang objek untuk setiap kolom a baris. Rails mengelola pergerakan informasi antara basis data dan objek, demikian pengembang *web* dapat berpikir dalam hal kelas Ruby daripada SQL. Rails juga mengelola hubungan antara tabel basis data sehingga mereka tampil sebagai referensi yang nyaman antara koleksi objek Ruby daripada asing kunci. Fasilitas ORM serupa telah muncul di lainnya *framework* generasi kedua seperti Laravel, dan Perpustakaan ORM telah ditambahkan ke beberapa *framework* generasi pertama[14].

Framework generasi kedua juga memperkenalkan pembagian kerja bergaya untuk pembuatan halaman berbasis *web* pada paradigma *model-view-controller* (MVC). Kelas *model* mengelola data back-end menggunakan mekanisme ORM kerangka kerja. Tampilan adalah *template* yang menghasilkan halaman *web* akhir. Kontroler menyediakan lem yang mengikat potongan. Misalnya, di Rails masing-masing URL yang masuk dikirim ke *controller* yang memanggil *model* untuk mengambil data untuk halaman *web*, kemudian meminta tampilan untuk menghasilkan HTML halaman[14].

3.4. Laravel Framework

Laravel diluncurkan sejak tahun 2011 dan mengalami pertumbuhan yang

cukup eksponensial. Di tahun 2015, Laravel adalah *framework* yang paling banyak mendapatkan bintang di Github. Laravel adalah salah satu *framework* untuk pengembangan aplikasi *web* dengan sintaks yang ekspresif dan elegan dan memberikan solusi untuk pengembangan dengan memfasilitasi tugas umum di sebagian besar proyek *web* besar. Laravel memiliki fungsi-fungsi yang sangat bermanfaat seperti *Bundle*, *Eloquent ORM*, *Application Logic*, *Reverse Routing*, *Class Auto Loading*, dan lain-lain. Laravel adalah *framework* pengembangan *web* MVC yang ditulis dalam PHP. *Framework* ini dirancang untuk meningkatkan kualitas perangkat lunak dengan mengurangi baik biaya pengembangan awal dan biaya pemeliharaan berkelanjutan, dan untuk meningkatkan pengalaman bekerja dengan aplikasi dengan memberikan sintaks ekspresif yang jelas dan serangkaian inti fungsi yang akan menghemat waktu pengembangan aplikasi[15].

3.5. Object Relational Mapping (ORM)

ORM adalah mekanisme yang memungkinkan untuk mengatasi, mengakses dan memanipulasi objek tanpa harus mempertimbangkan bagaimana objek itu berhubungan dengan data mereka sumber. ORM memungkinkan program mempertahankan pandangan yang konsisten dari objek dari waktu ke waktu. Bahkan ketika sumber yang mengirimkannya, wastafel yang menerimanya dan aplikasi yang mengaksesnya perubahan[16].

ORM mengelola detail pemetaan antara seperangkat objek dan relasional yang mendasarinya basis data, repositori XML atau sumber data lainnya dan sink, sekaligus menyembunyikan sering mengubah detail antarmuka terkait dari pengembang dan kode yang mereka buat. ORM menyembunyikan dan merangkum perubahan dalam sumber data itu sendiri. Sehingga ketika sumber data atau API mereka berubah, hanya ORM yang perlu diubah untuk mengikuti — bukan aplikasi yang menggunakan ORM untuk melindungi diri dari upaya semacam ini. Kapasitas ini memungkinkan pengembang mengambil keuntungan kelas baru saat tersedia dan juga memudahkan untuk memperluas berbasis ORM aplikasi. Dalam banyak kasus, perubahan ORM dapat menggabungkan teknologi dan kemampuan baru tanpa memerlukan perubahan kode untuk aplikasi terkait[16].

3.6. DBMS (Database Management System)

Pada dasarnya DBMS adalah kumpulan program yang memungkinkan *user* untuk memasukkan, memodifikasi, mengambil data agar membuatnya mudah untuk merancang, memelihara, dan mengakses basis data. DBMS membebaskan pengguna dari pengetahuan bagaimana data disimpan secara fisik dan algoritma yang rumit digunakan untuk melakukan operasi pada basis data. Itu hanya berkonsentrasi tentang bagaimana operasi harus dilakukan untuk mengambil data dari basis data. DBMS juga bertanggung jawab atas akses, keamanan, penyimpanan, dan sejumlah fungsi lainnya untuk sistem basis data. Hal ini memungkinkannya untuk mengelola data set besar dan data terstruktur yang membuatnya basis data dan menyediakan akses ke data untuk beberapa pengguna secara bersamaan dengan tetap menjaga integritas data[17].

3.7. MySQL

MySQL tumbuh dari kebutuhan di awal 1990-an untuk basis data yang cepat dan fleksibel dan digunakan untuk aplikasi berbasis *web*. Karena tidak ada basis data yang cocok, maka Michael mengembangkan suatu basis data yang baru. Dia menamainya dengan MySQL yang diambil dari nama putrinya, My dan SQL, yang merupakan singkatan dari *Structured Query Language*. SQL adalah bahasa pemrograman yang digunakan untuk berinteraksi dengan MySQL dan banyak basis data lainnya. Ketika orang berpikir tentang basis data *open source* yang mudah digunakan dan andal yang pertama mereka pikirkan adalah MySQL[18].

Basis data populer ini telah ada selama bertahun-tahun, stabil dan dapat diandalkan, dan memang demikian digunakan oleh banyak perusahaan. Baik dari perusahaan rintisan atau kecil sampai dengan perusahaan besar dalam segala bidang industri menggunakan basis data ini. MySQL dikembangkan selama bertahun-tahun oleh perusahaan MySQL Ab. MySQL Ab diakuisisi oleh Sun Microsystems pada tahun 2008. Setelah itu, Sun Microsystems diakuisisi oleh Oracle Corporation pada tahun 2009[18].

3.8. Arsitektur MVC (Model View Controller)

Pola arsitektur MVC membagi interaksi aplikasi menjadi tiga komponen. *Model* berisi fungsi dan data inti. *View* menunjukkan informasi kepada pengguna sedangkan *controller* menangani input pengguna. Tampilan dan pengontrol keduanya terdiri dari antarmuka pengguna[19]. MVC berpatokan dengan pendekatan paling umum untuk membagi kode kita menjadi beberapa fungsi[20]. Pendekatan ini mudah dikenali dan paling banyak diterima. Keuntungan utama dari pendekatan ini adalah kode yang dapat digunakan kembali.

Arsitektur MVC dibagi menjadi 3 lapisan antara lain :

1. Model

Model memiliki tugas untuk mengatur, menyiapkan, memanipulasi dan mengorganisasikan data (dari basis data) sesuai dengan instruksi dari *controller*. Fungsi-fungsi yang terkandung dalam *model* akan dipanggil oleh *controller*[21]. *Model* merangkum lebih dari sekedar data dan fungsi yang beroperasi di dalamnya. *Model* ini terdiri dari kelas yang mendefinisikan domain yang diminati. Benda dimiliki oleh *domain* ini sering merangkum data yang disimpan dalam basis data, tetapi juga menyertakan kode yang digunakan untuk memanipulasi data ini dan menegakkan aturan bisnis. Dapat diartikan bahwa *model* adalah bagian yang berinteraksi dengan basis data untuk menangani data, logika, dan aturan[22]. Ini tidak hanya menangkap keadaan proses atau sistem tetapi cara kerja sistem. Misalnya, programmer dapat mendefinisikan *model* yang menjembatani *backend* komputasi dengan *frontend*.

2. View

View adalah komponen yang bertanggung jawab untuk presentasi kepada pengguna[21]. *View* mengontrol bagaimana data ditampilkan dan bagaimana pengguna berinteraksi. *View* juga menyediakan cara mengumpulkan data dari pengguna. Teknologi yang digunakan pada bagian *view* adalah HTML, CSS, dan JavaScript. Sebagai aturan umum, *view* tidak boleh mengandung elemen logika aplikasi. Ini berarti blok logika harus dijaga agar tetap minimum[23].

View biasanya memiliki hubungan 1-1 dengan permukaan layar dan tahu cara membuatnya. *View* menghubungkan pengguna ke *model* dan menampilkan isinya ke permukaan layar. Selain itu, ketika *model* berubah *view* secara otomatis memuat ulang bagian yang terpengaruh untuk menunjukkan perubahan. Mungkin ada banyak *view* pada *model* yang sama dan setiap *view* dapat membuat konten *model* ke permukaan *view* yang berbeda[20].

3. Controller

Controller merupakan komponen yang berfungsi untuk memanggil fungsi yang ada pada *model* dan mengirimkan hasilnya menuju *view*. *Controller* juga mengambil masukan dari pengguna yang kemudian akan diproses oleh *model*[21]. Jadi, *controller* bertanggung jawab untuk memetakan aksi pengguna terhadap respon aplikasi. Sebagai contoh ketika pengguna menekan tombol atau memilih item menu, *controller* yang memiliki tanggung jawab untuk menentukan apa yang harus di respon oleh aplikasi. *Controller* mengirim perintah ke *model* untuk memperbarui data dan juga mengirim perintah ke *view* untuk mengubah data yang diterima atau ditampilkan. Ketika permintaan tiba di *server*, kerangka kerja MVC mengirimkannya kepada bagian *controller* berdasarkan *URL*[20]

BAB VI. PENUTUP

6.1. Kesimpulan

Berdasarkan hasil dari perancangan, pembahasan dan pengujian SIGARA maka dapat disimpulkan beberapa hal antara lain:

1. SIGARA berhasil membantu perusahaan untuk mengelola proses transaksi pembelian, produksi, penjualan, hutang dan piutang.
2. Fitur untuk pengelolaan data master dapat memudahkan persiapan untuk melakukan proses transaksi.
3. Secara keseluruhan dari kuesioner yang diberikan kepada pihak yang akan menggunakan SIGARA, berpendapat bahwa sistem berjalan dengan baik sesuai dengan yang diharapkan.

6.2. Saran

Berikut adalah saran dari hasil pengembangan dan pengujian SIGARA:

1. Melakukan pengembangan SIGARA pada *platform mobile*.
2. Peninjauan kembali terhadap atribut-atribut data master karena terdapat beberapa inputan yang sama.

DAFTAR PUSTAKA

- [1] “Profil Perusahaan – PT Niagara Kosmetik.” [Online]. Available: <https://www.niagarakosmetik.co.id/about-us/>. [Accessed: 06-Oct-2019].
- [2] “Niagara,” no. September, p. 2019, 2019.
- [3] N. Yuliani, “Konsep Pemrograman *Web*,” *World Wide Web Internet Web Inf. Syst.*, pp. 1–25.
- [4] Q. H. Nguyen, “Building a *web* application with Laravel 5,” *Oulu Univ. Appl. Sci.*, pp. 1–35, 2015.
- [5] Wandu, “Sistem Informasi Pengelolaan Data Barang Berbasis *Web* Pada PD. Alfa Multi Usaha Pontianak,” AMIK BSI Pontianak, 2017.
- [6] I. U. W. Nur Aristania, “Pembuatan Sistem Informasi Pengelolaan Persediaan Barang Dan Data Proyek Gypsum Pada Utomo Gypsum,” *J. Speed – Sentra Penelit. Eng. dan Edukasi*, vol. 9, no. 1, pp. 62–68, 2017.
- [7] A. Hanariana, R. Agustiansyah, and R. Andreswari, “Membangun Sistem Informasi Pengolahan Data Po Box Berbasis *Web* Menggunakan Metode Waterfall Studi Kasus PT. Pos Indonesia Bandung,” *eProceedings Eng.*, vol. 2, no. 1, pp. 1004–1011, 2015.
- [8] Muhdar Abdurahman, “IJIS Indonesian Journal on Information System ISSN 2548-6438,” *IJIS-Indonesia J. Inf. Syst.*, vol. 4, no. April, pp. 69–76, 2019.
- [9] R. A. Nurul Huda, “Implementasi Sistem Informasi Inventaris Barang pada PT.PLN (Persero) Palembang,” *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 9, no. 1, p. 13, 2020.
- [10] N. Lediwara and M. Rivaldi, “Perancangan Sistem Informasi Inventaris Barang Laboratorium Komputer SMPN 11 Kota Bengkulu,” *J. Teknol. Sist. Inf. dan Apl.*, vol. 2, no. 4, p. 117, 2019.
- [11] N. Novi Oktaviani, I Made Widiarta, “Sistem Informasi Inventaris Barang Berbasis *Web* Pada Smp Negeri 1 Buer,” *J. JINTEKS*, vol. 1, no. 2, pp. 160–168, 2019.
- [12] N. Choudhury, “World Wide *Web* and Its Journey from *Web* 1.0 to *Web* 4.0,”

- Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 6, pp. 8096–8100, 2014.
- [13] “PHP: What is PHP? - Manual.” [Online]. Available: <https://www.php.net/manual/en/intro-what-is.php>. [Accessed: 13-Jan-2020].
- [14] J. K. Ousterhout, “Fiz: A Component Framework for *Web* Applications,” *Stanford CSD Tech. Rep.*, pp. 1–14, 2009.
- [15] V. V. Parkar, P. P. Shinde, S. C. Gadade, and P. M. Shinde, “Utilization of Laravel Framework for Development of *Web* Based Recruitment Tool,” *Natl. Conf. "Changing Technol. Rural Dev.*, pp. 36–41, 2016.
- [16] C. Ireland, D. Bowers, M. Newton, and K. Waugh, “Understanding object-relational mapping: A framework based approach,” *Int. J. Adv. Softw.*, vol. 2, no. 2, pp. 202–216, 2009.
- [17] N. Sharma, “Overview of the Basis data Management System,” *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 4, pp. 362–369, 2017.
- [18] D. Bartholomew, “MariaDB vs . MySQL Similarities Between MariaDB and MySQL,” pp. 1–6, 2012.
- [19] F. Buscman, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal, *PATTERN-ORIENTED SOFTWARE ARCHITECTURE A SYSTEM OF PATTERN*, vol. 1. .
- [20] R. K. Hapsari, A. W. Azinar, and Sugiyanto, “Architecture Application Model View Controller (MVC) in Designing Information System of MSME Financial Report Architecture Application Model View Controller (Mvc) in Designing Information System of Msme Financial Report,” *J. Softw. Eng. Simul.*, vol. 3, no. 7, pp. 36–41, 2017.
- [21] N. L. P. P. Utpatadevi, “Implementation of MVC (Model-View-Controller) Architectural to Academic Management Information System with Android Platform Base,” *Int. J. Comput. Appl.*, vol. 57, no. 8, pp. 1–6, 2012.
- [22] A. Verma, “Mvc Architecture : a Comparative Study Between Ruby on Rails and Laravel,” *Indian J. Comput. Sci. Eng.*, vol. 5, no. 5, pp. 196–198, 2014.
- [23] D. P. Pop and A. Altar, “Designing an MVC model for rapid *web* application development,” *Procedia Eng.*, vol. 69, pp. 1172–1179, 2014.

LAMPIRAN

Lampiran I : Bukti Kuesioner Pengujian Terhadap Pengguna Pakar

Nama
3 tanggapan

Hendro Susanto

Saptono Hadi Sutikno

Didik Yulianto

Jabatan Pada PT Niagara Kosmetik
3 tanggapan

Staff Administrasi

Direktur

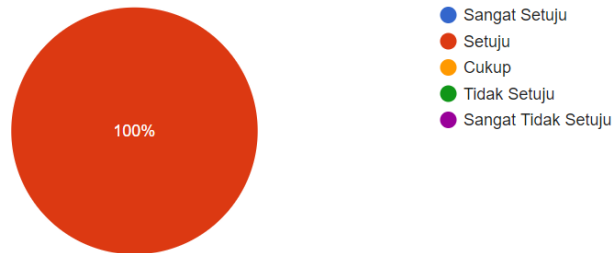
Penanggung jawab produksi

SIGARA mempunyai tampilan yang mudah digunakan dan dipahami.
3 tanggapan

Kategori	Persentase
Sangat Setuju	33,3%
Setuju	66,7%
Cukup	0%
Tidak Setuju	0%
Sangat Tidak Setuju	0%

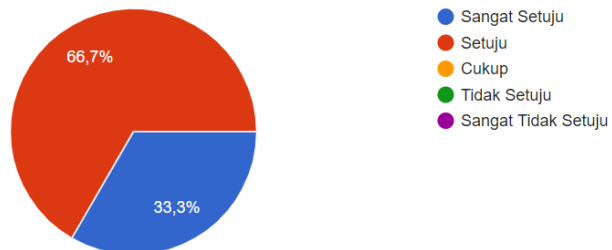
Data yang diolah dan diperlukan dalam sistem sudah sesuai

3 tanggapan



SIGARA dapat membantu proses pengelolaan pembelian, produksi, penjualan, hutang dan piutang

3 tanggapan



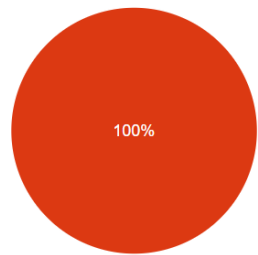
Tampilan laporan-laporan dan dokumen yang dicetak oleh sistem mudah dipahami dan data yang ditampilkan sudah benar

3 tanggapan



SIGARA tidak ada error atau bug yang mengganggu pada saat dijalankan

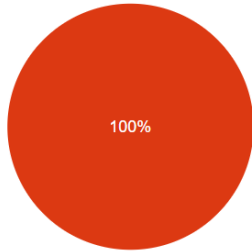
3 tanggapan



- Sangat Setuju
- Setuju
- Cukup
- Tidak Setuju
- Sangat Tidak Setuju

Saya sudah puas menggunakan SIGARA secara keseluruhan

3 tanggapan



- Sangat Setuju
- Setuju
- Cukup
- Tidak Setuju
- Sangat Tidak Setuju