

SKRIPSI

PEMBANGUNAN SISTEM INFORMASI PEMESANAN
RESTORAN MENGGUNAKAN *FRAMEWORK* NEXT.JS



Christopher William

NPM: 6181901048

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS
UNIVERSITAS KATOLIK PARAHYANGAN
2023

UNDERGRADUATE THESIS

**BUILDING A RESTAURANT ORDERING INFORMATION
SYSTEM USING NEXT.JS FRAMEWORK**



Christopher William

NPM: 6181901048

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2023**

LEMBAR PENGESAHAN

PEMBANGUNAN SISTEM INFORMASI PEMESANAN RESTORAN MENGGUNAKAN *FRAMEWORK* NEXT.JS

Christopher William

NPM: 6181901048

Bandung, 7 Juli 2023

Menyetujui,

Pembimbing

Raymond Chandra Putra, M.T.

Ketua Tim Penguji

Anggota Tim Penguji

Chandra Wijaya, M.T.

Gede Karya, M.T.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

PEMBANGUNAN SISTEM INFORMASI PEMESANAN RESTORAN MENGUNAKAN *FRAMEWORK* NEXT.JS

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 7 Juli 2023



Christopher William
NPM: 6181901048

ABSTRAK

Restoran adalah bisnis yang populer pada industri jasa pangan. Alur kegiatan pada sebuah restoran meliputi pemesanan makanan, pembuatan makanan di dapur, dan pembayaran pesanan. Restoran seringkali memiliki sejumlah tantangan dan masalah dalam kegiatan operasionalnya. Masalah-masalah seperti pencatatan pesanan yang salah dan antrian pesanan yang terlewat dapat mengurangi tingkat kepuasan pelanggan. Seiring berkembangnya teknologi, sistem informasi seringkali digunakan untuk membantu kegiatan operasional sebuah restoran agar dapat berlangsung secara lebih terstruktur.

Pada skripsi ini telah dilakukan analisis sistem pemesanan restoran sejenis dan wawancara dengan pemilik restoran untuk mengetahui kebutuhan sistem pemesanan restoran. Berdasarkan hasil analisis tersebut, dirancanglah sebuah sistem pemesanan restoran yang memiliki fitur utama menyediakan menu secara digital, menerima pesanan pelanggan, mengelola antrian pesanan, dan mencatat transaksi. Pembangunan sistem pemesanan restoran memanfaatkan Next.js, yaitu sebuah *framework* JavaScript yang dibangun di atas React. Pemanfaatan Next.js membawa fitur-fitur modern yang dapat meningkatkan kinerja sistem pemesanan restoran. Selain itu, sistem pemesanan restoran yang dibangun pun mendukung komunikasi *real-time* dengan memanfaatkan protokol WebSocket untuk meningkatkan kecepatan komunikasi antara pelanggan dan karyawan restoran.

Sistem restoran yang dibangun telah melalui tahap pengujian. Pada saat proses pengujian, seluruh fitur dan interaksi sistem dengan pengguna diuji untuk memastikan bahwa sistem pemesanan restoran dapat berfungsi dengan baik. *User Acceptance Test* (UAT) pun telah dilakukan untuk mengukur tingkat kepuasan pengguna terhadap sistem pemesanan restoran yang dibangun. Hasil UAT menunjukkan respons positif, yang menegaskan bahwa sistem pemesanan restoran yang dibangun dapat memenuhi harapan pengguna, baik sebagai pelanggan maupun karyawan restoran. Sistem restoran yang dibangun menggunakan *framework* Next.js dapat berfungsi dengan baik, dan dapat membantu kegiatan operasional restoran serta meningkatkan kepuasan pelanggan.

Kata-kata kunci: sistem pemesanan restoran, menu digital, pencatatan pesanan, *framework* Next.js, WebSocket

ABSTRACT

Restaurants are a popular business in the food service industry. The flow of activities in a restaurant includes ordering food, preparing food in the kitchen, and payment for orders. Restaurants often face a number of challenges and issues in their operational activities. Problems such as incorrect order recordings and overlooked order queues can reduce customer satisfaction levels. As technology advances, information systems are often used to assist the operational activities of a restaurant to make it more structured.

In this thesis, an analysis of similar restaurant ordering systems has been carried out and interviews with restaurant owners have been conducted to understand the requirements of a restaurant ordering system. Based on these analyses, a restaurant ordering system was designed that has key features including providing a digital menu, receiving customer orders, managing the order queue, and recording transactions. The development of the restaurant ordering system leverages Next.js, a JavaScript framework built on top of React. The use of Next.js brings modern features that can enhance the performance of the restaurant ordering system. Additionally, the restaurant ordering system supports real-time communication by utilizing the WebSocket protocol to speed up communication between customers and restaurant employees.

The restaurant ordering system has gone through testing stages. During the testing process, all system features and interactions with users were tested to ensure that the restaurant ordering system can function well. User Acceptance Testing (UAT) has also been conducted to measure user satisfaction with the restaurant ordering system. The results of the UAT show positive responses, affirming that the restaurant ordering system can meet user expectations, both as customers and restaurant employees. The restaurant ordering system built using the Next.js framework works well and can assist restaurant operational activities as well as increase customer satisfaction.

Keywords: restaurant ordering system, digital menu, order recording, Next.js framework, WebSocket

Penulis mempersembahkan skripsi ini kepada keluarga tercinta.

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa karena dengan rahmat dan karunia-Nya, penulis dapat menyelesaikan skripsi ini sebagai syarat kelulusan dan memperoleh gelar sarjana. Pengerjaan skripsi ini tentu tidak lepas dari tantangan dan kendala yang dialami oleh penulis namun, dukungan dari keluarga serta teman-teman selalu menyertai hingga skripsi ini selesai. Oleh sebab itu, penulis ingin secara khusus berterima kasih kepada:

1. Tuhan Yang Maha Esa.
2. Keluarga tercinta yang selalu memberikan doa dan dukungan sampai skripsi ini selesai.
3. Bapak Raymond Chandra Putra sebagai dosen pembimbing yang selalu memberikan bimbingan dan saran dalam membuat skripsi ini.
4. Bapak Chandra Wijaya dan Bapak Gede Karya sebagai dosen penguji yang telah memberikan kritik dan saran untuk perbaikan skripsi ini.
5. Pemilik restoran Nasi Kulit Malam Minggu yang telah bersedia menjadi narasumber wawancara dalam pembuatan skripsi ini.
6. Axel dan Russel sebagai anggota grup musik Scarlet Script yang selalu menemani dan menghibur penulis.
7. Grup teman dekat selama berkuliah di Teknik Informatika Unpar, yaitu Nicholas Khrisna, Ivan Limosi, Michael Manurung, Clement Rivandi, Bernadus Lukas, Jason, dan Julyus Andreas.
8. Orang-orang berjasa lainnya yang tidak disebutkan disini.

Skripsi merupakan karya yang dibuat untuk menemukan suatu manfaat dari apa yang telah diteliti sehingga dikemudian hari dapat membantu atau memudahkan pekerjaan orang lain. Penulis memohon maaf jika masih terdapat banyak kesalahan dan mengharapkan kritik dan saran yang membangun. Semoga penelitian ini dapat menjadi inspirasi bagi peneliti berikutnya maupun pengetahuan bagi yang membaca.

Bandung, Juli 2023

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xxi
DAFTAR TABEL	xxiii
DAFTAR KODE PROGRAM	xxix
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	2
1.6 Sistematika Pembahasan	3
2 LANDASAN TEORI	5
2.1 Sistem Pemesanan Restoran	5
2.2 Pemrograman Berbasis Web	6
2.2.1 HTML	6
2.2.2 CSS	6
2.2.3 JavaScript	7
2.2.4 Node.js	8
2.2.5 <i>Web Service</i>	8
2.2.6 <i>Web Framework</i>	9
2.3 React	9
2.3.1 <i>Component</i>	9
2.3.2 <i>Conditional Rendering</i>	10
2.3.3 <i>Rendering Lists</i>	11
2.3.4 <i>React Hook</i>	12
2.3.5 <i>State dan hook useState</i>	12
2.3.6 <i>Effect dan hook useEffect</i>	13
2.4 Next.js	13
2.4.1 <i>Pages</i>	13
2.4.2 <i>Routing</i>	14
2.4.3 <i>Pre-rendering</i>	14
2.4.4 <i>API Route</i>	16
2.5 NextAuth.js	17
2.6 PostgreSQL	18
2.7 WebSocket	19
2.8 Socket.IO	19

2.9	react-chartjs-2	20
2.10	Cloudinary	22
3	ANALISIS	23
3.1	Analisis Masalah	23
3.2	Analisis Wawancara Pemilik Restoran	25
3.3	Analisis Sistem Sejenis	26
3.4	Analisis Pemilihan <i>Tech Stack</i>	27
3.4.1	Next.js	28
3.4.2	NextAuth.js	28
3.4.3	PostgreSQL	28
3.4.4	Socket.IO	29
3.4.5	Cloudinary	29
3.5	Analisis Arsitektur	30
3.6	Analisis Kebutuhan Perangkat Lunak	31
3.6.1	Spesifikasi Perangkat Lunak	31
3.6.2	Analisis Pengguna	32
3.6.3	<i>Use Case Diagram</i>	33
3.6.4	<i>Use Case Scenario</i>	34
3.6.5	<i>Swimlane Diagram</i>	41
3.6.6	<i>Data Flow Diagram</i>	42
3.6.7	<i>Entity Relationship Diagram</i>	44
3.6.8	Analisis WebSocket	46
4	PERANCANGAN	49
4.1	Pembagian Modul Pembangunan	49
4.2	Perancangan Basis Data	51
4.3	<i>Data Flow Diagram Level 2</i>	55
4.4	Perancangan Antarmuka	59
4.4.1	<i>Screenflow</i>	59
4.4.2	Perancangan Antarmuka Halaman	60
4.4.3	Perancangan Antarmuka <i>Component</i>	67
4.5	Perancangan <i>Endpoint</i>	70
4.6	Perancangan <i>Pages</i> dan <i>Component</i>	75
4.6.1	Perancangan <i>Pages</i>	75
4.6.2	Perancangan <i>Component</i>	82
5	IMPLEMENTASI DAN PENGUJIAN	89
5.1	Implementasi	89
5.1.1	Lingkungan Implementasi	89
5.1.2	Implementasi Basis Data	89
5.1.3	Implementasi Antarmuka	91
5.2	Pengujian	100
5.2.1	Pengujian Fungsional	100
5.2.2	Pengujian Eksperimental	116
5.2.3	Analisis Hasil Pengujian	118
6	KESIMPULAN DAN SARAN	121
6.1	Kesimpulan	121
6.2	Saran	121
	DAFTAR REFERENSI	123

A KODE PROGRAM	125
B HASIL WAWANCARA	197

DAFTAR GAMBAR

2.1	Contoh aplikasi sistem pemesanan restoran	5
2.2	Contoh hasil penambahan CSS	7
2.3	React <i>component</i>	9
2.4	Mekanisme koneksi WebSocket	19
2.5	Hasil <i>bar chart</i>	21
2.6	Hasil <i>line chart</i>	21
3.1	Alur kegiatan restoran tradisional	23
3.2	Alur kegiatan restoran modern	25
3.3	Restoran narasumber	26
3.4	Diagram arsitektur <i>client-server</i>	30
3.5	<i>Use case diagram</i>	33
3.6	<i>Swimlane diagram</i> sistem pemesanan restoran	42
3.7	<i>Data Flow Diagram</i> level 0	43
3.8	<i>Data Flow Diagram</i> level 1	43
3.9	<i>Entity Relationship Diagram</i>	45
4.1	Modul pembangunan	49
4.2	<i>Data Flow Diagram</i> level 2 membuat pesanan	56
4.3	<i>Data Flow Diagram</i> level 2 mengelola pesanan	57
4.4	<i>Screenflow</i>	59
4.5	Rancangan halaman saat belum <i>sign In</i>	60
4.6	Rancangan halaman <i>sign In</i>	61
4.7	Rancangan halaman pemesanan menu	61
4.8	Rancangan halaman <i>dashboard</i> karyawan	62
4.9	Rancangan halaman <i>dashboard manager</i>	62
4.10	Rancangan halaman <i>table</i>	63
4.11	Rancangan halaman <i>kitchen</i>	63
4.12	Rancangan halaman <i>order queue</i>	64
4.13	Rancangan halaman <i>add menu</i>	64
4.14	Rancangan halaman <i>edit menu</i>	65
4.15	Rancangan halaman <i>transaction history</i>	65
4.16	Rancangan halaman <i>edit employee</i>	66
4.17	Rancangan halaman <i>add employee</i>	66
4.18	Rancangan <i>component</i> MenuCard	67
4.19	Rancangan <i>component</i> FormKaryawan	67
4.20	Rancangan <i>component</i> FormMenu	68
4.21	Rancangan <i>component</i> OrderCard	68
4.22	Rancangan <i>component</i> OrderCard	69
5.1	Implementasi halaman saat belum <i>sign in</i>	91
5.2	Implementasi halaman <i>sign in</i>	92
5.3	Implementasi halaman pemesanan menu (pemilihan menu)	92

5.4	Implementasi halaman pemesanan menu (pesanan dalam proses)	93
5.5	Implementasi halaman pemesanan menu (pesanan menunggu pembayaran)	93
5.6	Implementasi halaman <i>dashboard manager</i>	94
5.7	Implementasi halaman <i>dashboard</i> karyawan	94
5.8	Implementasi halaman <i>table</i>	95
5.9	Implementasi halaman <i>kitchen</i>	95
5.10	Implementasi halaman <i>order queue</i> pesanan baru	96
5.11	Implementasi halaman <i>order queue</i> pesanan dalam proses	96
5.12	Implementasi halaman <i>order queue</i> menunggu pembayaran	97
5.13	Implementasi halaman <i>add menu</i>	97
5.14	Implementasi halaman <i>edit menu</i>	98
5.15	Implementasi halaman <i>transaction history</i> untuk tabel transaksi	98
5.16	Implementasi halaman <i>transaction history</i> untuk tabel pesanan yang dibatalkan	99
5.17	Implementasi halaman <i>add employee</i>	99
5.18	Implementasi halaman <i>edit employee</i>	100
5.19	Hasil cetak riwayat transaksi	111
5.20	Hasil cetak daftar pesanan	114
5.21	Hasil cetak bon pembayaran	115
5.22	Proses UAT	117

DAFTAR TABEL

2.1	Parameter cloudinary	22
3.1	Hasil observasi sistem pemesanan restoran manual.	27
3.2	Skenario untuk melihat daftar menu.	34
3.3	Skenario untuk membuat pesanan.	34
3.4	Skenario untuk membatalkan pesanan.	35
3.5	Skenario untuk menambah pesanan baru.	35
3.6	Skenario untuk meminta bantuan.	36
3.7	Skenario untuk menambah data admin.	36
3.8	Skenario untuk mengubah data admin.	36
3.9	Skenario untuk melihat dan mencetak rekap riwayat transaksi.	37
3.10	Skenario untuk melihat daftar antrian pesanan.	37
3.11	Skenario untuk mengelola antrian pesanan baru.	38
3.12	Skenario untuk mengelola antrian pesanan yang sedang diproses.	38
3.13	Skenario untuk mengelola antrian pesanan yang sedang menunggu pembayaran.	38
3.14	Skenario untuk mengelola pesanan tambahan.	38
3.15	Skenario untuk mengelola permohonan pembatalan pesanan.	39
3.16	Skenario untuk mencetak pesanan dan bon pembayaran.	39
3.17	Skenario untuk menambah data menu baru.	40
3.18	Skenario untuk mengubah data menu.	40
3.19	Skenario untuk menambah data meja.	40
3.20	Skenario untuk mengubah data meja.	41
3.21	Skenario untuk melihat data operasional umum.	41
3.22	Skenario untuk melihat data pendapatan.	41
3.23	<i>Data dictionary</i> DFD Level 1.	44
3.24	Tabel <i>event</i> WebSocket	46
4.1	Modul pembangunan sistem pemesanan restoran	50
4.2	Rancangan tabel Admin.	52
4.3	Rancangan tabel Meja.	52
4.4	Rancangan tabel Menu.	52
4.5	Rancangan tabel Kategori.	53
4.6	Rancangan tabel TerdiriMenu.	53
4.7	Rancangan tabel Pesanan.	53
4.8	Rancangan tabel KelolaPesanan.	54
4.9	Rancangan tabel TerdiriPesanan.	54
4.10	Rancangan tabel PesananTambahan.	55
4.11	Rancangan tabel Transaksi.	55
4.12	Rancangan tabel TerdiriTransaksi.	55
4.13	<i>Data dictionary</i> DFD level 2 membuat pesanan.	56
4.14	<i>Data dictionary</i> DFD level 2 mengelola pesanan.	58
4.15	Referensi gambar perancangan antarmuka pada <i>screenflow</i>	60
4.16	<i>Parameter</i> API POST /signin.	70

4.17	<i>Parameter</i> API POST /karyawan.	71
4.18	<i>Parameter</i> API PUT /karyawan.	71
4.19	<i>Parameter</i> API POST /menu.	71
4.20	<i>Parameter</i> API PUT /menu.	72
4.21	<i>Parameter</i> API PUT /menu/activation/[idMenu].	72
4.22	<i>Parameter</i> API POST /order.	72
4.23	<i>Parameter</i> API PUT /order/[idPesanan].	73
4.24	<i>Parameter</i> API PUT /order/[idPesanan]/[isiPesanan].	73
4.25	<i>Parameter</i> API POST /order/additional.	73
4.26	<i>Parameter</i> API POST /cancellationrequest.	74
4.27	<i>Parameter</i> API PUT /cancellationrequest.	74
4.28	<i>Parameter</i> API POST /meja.	75
4.29	<i>Parameter</i> API POST /transaksi.	75
4.30	<i>State</i> halaman /auth/signin.js.	75
4.31	<i>State</i> halaman /index.js.	76
4.32	<i>Effect</i> halaman /index.js.	76
4.33	<i>Function</i> halaman /index.js.	77
4.34	<i>Props</i> halaman /admin/index.js.	77
4.35	<i>State</i> halaman /admin/index.js.	78
4.36	<i>Function</i> halaman /admin/index.js.	78
4.37	<i>Props</i> halaman /admin/kitchen.js.	79
4.38	<i>Function</i> halaman /admin/kitchen.js.	79
4.39	<i>Props</i> halaman /admin/orderqueue.js.	79
4.40	<i>State</i> halaman /admin/orderqueue.js.	79
4.41	<i>Effect</i> halaman /admin/orderqueue.js.	80
4.42	<i>Function</i> halaman /admin/orderqueue.js.	80
4.43	<i>State</i> halaman /admin/table.js.	80
4.44	<i>Function</i> halaman /admin/table.js.	81
4.45	<i>Props</i> halaman /admin/transactionhistory.js.	81
4.46	<i>State</i> halaman /admin/transactionhistory.js.	81
4.47	<i>Function</i> halaman /admin/transactionhistory.js.	82
4.48	<i>State component</i> Header.	82
4.49	<i>Effect component</i> Header.	82
4.50	<i>Function component</i> Header.	83
4.51	<i>State component</i> FormKaryawan.	83
4.52	<i>Props component</i> FormMenu.	83
4.53	<i>State component</i> FormMenu.	84
4.54	<i>Function component</i> FormMenu.	84
4.55	<i>Props component</i> MenuCard.	85
4.56	<i>Props component</i> OrderCard.	85
4.57	<i>Function component</i> OrderCard.	86
4.58	<i>Props component</i> OrderItem.	86
4.59	<i>Props component</i> PendingOrderCard.	86
4.60	<i>State component</i> PendingOrderCard.	87
4.61	<i>Effect component</i> PendingOrderCard.	87
4.62	<i>Function component</i> PendingOrderCard.	87
5.1	Pengujian fungsional halaman saat belum <i>sign in</i>	100
5.2	Pengujian fungsional halaman <i>sign in</i>	101
5.3	Pengujian fungsional halaman pemesanan menu (<i>layout</i>).	101
5.4	Pengujian fungsional halaman pemesanan menu (pemilihan menu dan pembuatan pesanan).	102

5.5	Pengujian fungsional halaman pemesanan menu (menunggu konfirmasi).	102
5.6	Pengujian fungsional halaman pemesanan menu (sedang diproses).	103
5.7	Pengujian fungsional halaman pemesanan (menunggu pembayaran).	104
5.8	Pengujian fungsional halaman <i>add employee</i> .	106
5.9	Pengujian fungsional halaman <i>edit employee</i> .	106
5.10	Pengujian fungsional halaman <i>add menu</i> .	107
5.11	Pengujian fungsional halaman <i>edit menu</i> .	108
5.12	Pengujian fungsional halaman <i>dashboard</i> admin.	108
5.13	Pengujian fungsional halaman <i>kitchen</i> .	109
5.14	Pengujian fungsional halaman <i>table</i> .	110
5.15	Pengujian fungsional halaman <i>transaction history</i> .	110
5.16	Pengujian fungsional halaman <i>order queue</i> (<i>layout</i>).	112
5.17	Pengujian fungsional halaman <i>order queue</i> (pesanan menunggu konfirmasi).	112
5.18	Pengujian fungsional halaman <i>order queue</i> (pesanan sedang diproses).	113
5.19	Pengujian fungsional halaman <i>order queue</i> (menunggu pembayaran).	114
5.20	Pengujian fungsional <i>header</i> admin.	115
5.21	Pengujian fungsional <i>navigation bar</i> admin.	116

DAFTAR KODE PROGRAM

2.1	Contoh elemen HTML.	6
2.2	Contoh penggunaan CSS.	6
2.3	Contoh penggunaan JavaScript.	7
2.4	Penggunaan JSX	10
2.5	Contoh <i>component</i> .	10
2.6	Contoh <i>conditional rendering</i> menggunakan sintaks <i>if</i> .	10
2.7	Contoh <i>conditional rendering</i> menggunakan <i>ternary operator</i> .	11
2.8	Contoh <i>conditional rendering</i> menggunakan <i>logical AND</i> .	11
2.9	Contoh pembuatan daftar menu menggunakan <i>map()</i> .	11
2.10	Contoh pembuatan daftar menu menggunakan <i>map()</i> yang sudah disaring.	11
2.11	Contoh kode pemanggilan dan konvensi penamaan <i>hook</i> <i>useState</i> .	12
2.12	Contoh kode pemanfaatan <i>hook</i> <i>useState</i> .	12
2.13	Contoh <i>useEffect()</i> .	13
2.14	Page “about.js” yang dibuat menggunakan <i>Next.js</i>	14
2.15	Contoh penggunaan <i>component</i> <i>Link</i> dan <i>useRouter</i>	14
2.16	Contoh kode <i>getServerSideProps</i>	15
2.17	Contoh kode <i>getStaticProps</i>	15
2.18	Contoh kode <i>getStaticPaths</i>	16
2.19	Contoh kode <i>On-Demand Revalidation</i>	16
2.20	Contoh kode <i>API routes</i>	16
2.21	Inisialisasi <i>NextAuth.js</i>	17
2.22	Contoh penggunaan <i>hook</i> <i>useSession()</i>	18
2.23	Inisialisasi dan konfigurasi <i>pool</i>	18
2.24	Contoh mengirimkan <i>query</i> ke basis data dan menerima hasil <i>query</i> tersebut.	18
2.25	Inisialisasi <i>Socket.IO</i> pada bagian <i>server</i>	19
2.26	Inisialisasi <i>Socket.IO</i> pada bagian <i>client</i>	20
2.27	Pembuatan <i>component</i> <i>Bar</i> menggunakan <i>react-chartjs-2</i>	20
2.28	Pembuatan <i>component</i> <i>Line</i> menggunakan <i>react-chartjs-2</i>	21
5.1	Implementasi pembuatan tabel <i>Admin</i> .	89
5.2	Implementasi pembuatan tabel <i>Meja</i> .	90
5.3	Implementasi pembuatan tabel <i>Menu</i> .	90
5.4	Implementasi pembuatan tabel <i>Kategori</i> .	90
5.5	Implementasi pembuatan tabel <i>TerdiriMenu</i> .	90
5.6	Implementasi pembuatan tabel <i>Pesanan</i> .	90
5.7	Implementasi pembuatan tabel <i>TerdiriPesanan</i> .	90
5.8	Implementasi pembuatan tabel <i>PesananTambahan</i> .	91
5.9	Implementasi pembuatan tabel <i>KelolaPesanan</i> .	91
5.10	Implementasi pembuatan tabel <i>Transaksi</i> .	91
5.11	Implementasi pembuatan tabel <i>TerdiriTransaksi</i> .	91
A.1	<i>/component/barchart.js</i>	125

A.2	/component/formkaryawan.js	125
A.3	/component/formmenu.js	126
A.4	/component/header.js	130
A.5	/component/layout.js	131
A.6	/component/linechart.js	132
A.7	/component/menucard.js	132
A.8	/component/navbar.js	133
A.9	/component/ordercard.js	134
A.10	/component/orderitem.js	136
A.11	/component/pendingordercard.js	137
A.12	/component/menucard.js	141
A.13	/module/orderformatter.js	142
A.14	/module/pg.js	142
A.15	/module/timecalculator.js	143
A.16	/pages/_app.js	143
A.17	/pages/index.js	143
A.18	/pages/admin/addemployee.js	147
A.19	/pages/admin/addmenu.js	148
A.20	/pages/admin/editemployee.js	148
A.21	/pages/admin/editmenu.js	149
A.22	/pages/admin/index.js	150
A.23	/pages/admin/kitchen.js	154
A.24	/pages/admin/orderqueue.js	155
A.25	/pages/admin/table.js	158
A.26	/pages/admin/transactionhistory.js	159
A.27	/pages/auth/signin.js	161
A.28	/pages/api/meja.js	162
A.29	/pages/api/revalidate.js	163
A.30	/pages/api/signin.js	163
A.31	/pages/api/socket.js	164
A.32	/pages/api/transaksi.js	165
A.33	/pages/api/auth/[...nextauth].js	165
A.34	/pages/api/cancellationrequest/index.js	166
A.35	/pages/api/help/[idMeja].js	167
A.36	/pages/api/karyawan/index.js	168
A.37	/pages/api/menu/index.js	169
A.38	/pages/api/menu/activation/[idMenu].js	170
A.39	/pages/api/order/index.js	171
A.40	/pages/api/order/[idPesanan]/[isiPesanan].js	171
A.41	/pages/api/order/[idPesanan]/index.js	173
A.42	/pages/api/order/additional/index.js	173
A.43	/pages/api/order/additional/[idPesanan]/[isiPesanan].js	175
A.44	/pages/api/order/additional/[idPesanan]/index.js	175
A.45	/pages/api/order/current/[idMeja].js	176
A.46	/styles/AddEmployee.module.css	176
A.47	/styles/AddMenu.module.css	177
A.48	/styles/Admin.module.css	177
A.49	/styles/EditEmployee.module.css	178
A.50	/styles/EditMenu.module.css	179
A.51	/styles/FormKaryawan.module.css	179
A.52	/styles/FormMenu.module.css	180

A.53 /styles/globals.css	181
A.54 /styles/Header.module.css	182
A.55 /styles/Home.module.css	182
A.56 /styles/Kitchen.module.css	184
A.57 /styles/Layout.module.css	185
A.58 /styles/MenuCard.module.css	185
A.59 /styles/NavBar.module.css	186
A.60 /styles/OrderCard.module.css	187
A.61 /styles/OrderItem.module.css	188
A.62 /styles/OrderQueue.module.css	188
A.63 /styles/PendingOrderCard.module.css	189
A.64 /styles/SignIn.module.css	192
A.65 /styles/Table.module.css	194
A.66 /styles/TransactionHistory.module.css	194

BAB 1

PENDAHULUAN

Pada bab ini, dijelaskan gambaran umum dari permasalahan yang dibahas. Bab ini terdiri dari latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi, dan sistematika pembahasan.

1.1 Latar Belakang

Restoran adalah bisnis atau usaha jasa pangan yang cukup populer. Hal ini terlihat dari banyaknya restoran yang tersebar di berbagai wilayah. Restoran memiliki alur kegiatan seperti pemesanan makanan, pembuatan makanan di dapur, hingga pembayaran. Sebuah restoran yang sedang ramai dikunjungi pelanggan berpotensi menimbulkan masalah pelayanan seperti pencatatan pesanan yang keliru atau antrian pesanan yang terlambat. Hal ini menyebabkan gangguan dalam alur kegiatan sebuah restoran dan dapat mengurangi kepuasan pengunjung.

Seiring berkembangnya teknologi semakin banyak perusahaan atau bisnis yang menggunakan sistem informasi untuk membantu kegiatan bisnis mereka. Dalam kasus bisnis restoran, sistem informasi dimanfaatkan untuk mengelola pesanan dan menangani transaksi. Dengan memanfaatkan sistem informasi, alur kegiatan dalam sebuah restoran, mulai dari pemesanan makanan, pembayaran, dan antrian pembuatan makanan di dapur, menjadi lebih terstruktur sehingga potensi terjadinya *human error* pun dapat diminimalisir.

Sistem informasi berbasis web adalah suatu aplikasi web yang diakses melalui jaringan internet untuk mengumpulkan, mengelola, dan mendistribusikan informasi. Pada zaman yang serba cepat dan kompetitif, pembangunan aplikasi web seringkali memanfaatkan *framework* untuk mempercepat dan mempermudah pengembangannya. *Framework* menyediakan *tools* dan layanan-layanan yang dapat membantu pengembang perangkat lunak dalam membangun perangkat lunak dengan fitur-fitur yang bermanfaat.

Salah satu *framework* yang cukup populer dalam membangun aplikasi web adalah Next.js. Kepopuleran Next.js dapat dilihat dari banyaknya situs web besar seperti TikTok, Netflix, dan Twitch yang menggunakannya. Sudah banyak juga situs-situs web dari bidang *finance*, *news*, *e-commerce*, *travel*, *creative*, *entertainment*, dan *marketing* yang memanfaatkan Next.js. Hal ini disebabkan oleh berbagai macam fitur yang disediakan oleh Next.js yang berguna dalam mengembangkan sebuah situs web seperti *pre-rendering*, sistem *routing*, dan *built-in* CSS.

Next.js adalah *framework* JavaScript yang dibangun di atas React. React adalah *library* JavaScript yang dikembangkan oleh Meta untuk membuat antarmuka pengguna. React sangat fleksibel, di mana para pengembang dapat membuat fitur baru menggunakan React pada aplikasi web mereka tanpa perlu mengubah kode yang sudah ada. Dikarenakan sifatnya yang deklaratif dan berbasis komponen, situs web yang dibangun menggunakan React menjadi lebih mudah dikembangkan dan lebih *reusable*.

Pemrograman deklaratif dalam React berarti mendeskripsikan hasil akhir antarmuka tanpa perlu menjelaskan langkah demi langkah bagaimana cara menyusun antarmuka tersebut. Setelah antarmuka dideskripsikan, React kemudian secara otomatis menyusun antarmuka yang sudah dideskripsikan dan memperbaharuihnya sesuai. Antarmuka aplikasi React dibangun dengan basis komponen. Hal ini berarti antarmuka aplikasi React disusun oleh komponen-komponen yang lebih

kecil dan terisolasi satu sama lain. Membagi-bagi antarmuka yang kompleks menjadi beberapa bagian dapat membuat kode menjadi lebih mudah dimengerti dan dipelihara.

Next.js adalah *framework* React, maka semua fitur dan keuntungan yang terdapat pada React juga disediakan oleh Next.js. Lebih dari itu, Next.js juga menyediakan fitur-fitur tambahan yang berguna. Salah satu fitur utama dari Next.js *server-side rendering*, yang memungkinkan komponen-komponen React pada halaman web akan di-*render* pada *server* sebelum dikirim kepada *client*. Hal ini mempersingkat waktu yang dibutuhkan oleh *client* untuk menampilkan isi halaman web.

Dalam sistem pemesanan restoran, komunikasi antara pelanggan dan karyawan restoran perlu terjadi secara *real-time* untuk mempercepat alur kegiatan di restoran. Contohnya saat pelanggan melakukan pemesanan menu, maka menu yang dipesan tersebut harus dapat langsung diterima oleh karyawan untuk diproses. Selain itu, saat karyawan mengelola pesanan (seperti mengonfirmasi, mengantarkan, atau menolak pesanan), data pesanan yang terbaru pun perlu langsung didapatkan oleh pelanggan. Komunikasi *real-time* seperti ini dapat dicapai dengan menggunakan WebSocket.

WebSocket adalah protokol komunikasi yang memungkinkan komunikasi 2 arah antara *client* dan *server*. WebSocket menyediakan koneksi yang persisten, sehingga komunikasi antara *client* dan *server* dapat terjadi secara *real-time*, tanpa diperlukannya *request* yang dibuat oleh *client* terlebih dahulu. Pemanfaatan WebSocket dapat meningkatkan daya tanggap aplikasi sistem pemesanan restoran.

Pada skripsi ini dikembangkan sebuah sistem informasi restoran berbasis web menggunakan Next.js. Sistem informasi yang dikembangkan menggunakan basis data PostgreSQL dan autentikasi NextAuth.js, sehingga dilakukan juga studi literatur dan implementasi menggunakan kedua teknologi tersebut. Agar komunikasi antara pelanggan dengan karyawan restoran terjadi secara *real-time*, maka dilakukan juga studi literatur dan implementasi *WebSocket* yang didukung oleh Socket.IO.

1.2 Rumusan Masalah

Rumusan masalah yang dibahas dalam skripsi ini adalah:

1. Bagaimana merancang sistem informasi restoran menggunakan *framework* Next.js?
2. Bagaimana mengimplementasikan sistem informasi restoran menggunakan *framework* Next.js?

1.3 Tujuan

Tujuan yang ingin dicapai dari skripsi ini adalah:

1. Merancang sistem informasi restoran menggunakan *framework* Next.js.
2. Mengimplementasikan sistem informasi restoran menggunakan *framework* Next.js.

1.4 Batasan Masalah

Beberapa batasan yang dibuat terkait skripsi ini adalah:

1. Sistem informasi restoran yang dibuat tidak terintegrasi dengan sistem pemesanan daring pihak ketiga yang umum digunakan di Indonesia seperti GoFood atau GrabFood.
2. Sistem informasi restoran yang dibuat hanya menangani pemesanan dan mengeluarkan bon pembayaran, dan tidak menangani keluar masuknya uang.

1.5 Metodologi

Langkah-langkah yang dilakukan untuk membangun perangkat lunak sistem pemesanan restoran ini adalah:

1. Mencari narasumber dan melakukan wawancara terhadap sistem pemesanan restoran.
2. Melakukan observasi terhadap sistem sejenis.

3. Merumuskan spesifikasi perangkat lunak dari hasil wawancara dan observasi.
4. Melakukan studi literatur mengenai React, Next.js, NextAuth.js, PostgreSQL, WebSocket, dan Socket.IO.
5. Menganalisis sistem informasi yang dibuat.
6. Merancang sistem informasi yang dibuat.
7. Mengimplementasikan perangkat lunak menggunakan Next.js.
8. Melakukan pengujian terhadap perangkat lunak yang sudah dibuat.
9. Melaporkan hasil penelitian dalam bentuk dokumen skripsi.

1.6 Sistematika Pembahasan

Sistematika pembahasan pada skripsi ini adalah sebagai berikut:

1. **Bab 1: Pendahuluan**

Bab ini berisi latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi penelitian, dan sistematika pembahasan mengenai pembangunan sistem pemesanan restoran menggunakan *framework* Next.js.

2. **Bab 2: Landasan Teori**

Bab ini berisi landasan teori mengenai sistem pemesanan restoran, pemrograman berbasis web, serta teknologi-teknologi yang digunakan dalam pengembangan sistem pemesanan restoran, yaitu React, Next.js, NextAuth.js, PostgreSQL, WebSocket, Socket.IO, react-chartjs-2, dan Cloudinary.

3. **Bab 3: Analisis**

Bab ini berisi analisis masalah, analisis wawancara dengan pemilik restoran, analisis sistem sejenis, analisis pemilihan *tech stack*, analisis arsitektur, analisis WebSocket, dan analisis kebutuhan perangkat lunak sistem pemesanan restoran.

4. **Bab 4: Perancangan**

Bab ini berisi pembagian modul pengembangan, perancangan basis data, *Data Flow Diagram* level 2, perancangan antarmuka, perancangan *endpoint*, dan perancangan *pages* dan *component* untuk perangkat lunak sistem pemesanan restoran.

5. **Bab 5: Implementasi dan Pengujian**

Bab ini berisi lingkungan implementasi, implementasi basis data, implementasi antarmuka, pengujian fungsional, pengujian eksperimental, dan analisis hasil pengujian pada pembangunan perangkat lunak sistem pemesanan restoran.

6. **Bab 6: Kesimpulan dan Saran**

Bab ini berisi kesimpulan dari keseluruhan sistem pemesanan restoran yang dibangun, serta saran dari penulis untuk pengembangan yang akan datang.

