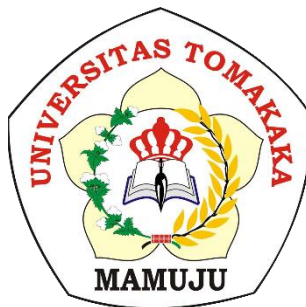


**APLIKASI KASIR UNTUK RESTORAN STUDI KASUS TRANSAKSI
PENJUALAN**

PROPOSAL



OLEH :

SRI WAHYUNI

NIM / NIRM : 190250502065 / 191025052060801065

**PROGRAM STRATA SATU (S1)
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS TOMAKAKA MAMUJU**

2022

DAFTAR TABEL

	Halaman
Tabel 1 Kebutuhan Data	21
Tabel 2 Informasi yang Dihasilkan	22
Tabel 3 Tabel Pelanggan	28
Tabel 4 Tabel Kasir	28
Tabel 5 Tabel Kepala Restoran	29
Tabel 6 Tabel Harga	30
Tabel 7 Tabel Menu	30
Tabel 8 Tabel Faktur	31
Tabel 9 Tabel Penjualan	31
Tabel 10 Tabel Pegawai	32
Tabel 11 Jadwal Penelitian	33

DAFTAR GAMBAR

	Halaman
Gambar 1 Simbol Flowchart 1.....	10
Gambar 2 Simbol Flowchart 2.....	11
Gambar 3 Relasi Tabel	27
Gambar 4 Membuat Proyek Baru	35
Gambar 5 Memilih Proyek.....	36
Gambar 6 Membuat Nama Proyek	36
Gambar 7 Membuat Package	37
Gambar 8 Memberi Nama Package.....	37
Gambar 9 Membuat Class	41
Gambar 10 Memberi Nama Class.....	41
Gambar 11 Membuat Class Kedua.....	42
Gambar 12 Membuat Class Ketiga	42
Gambar 13 Output Ketika Dijalankan.....	46
Gambar 14 Menambahkan Data Kedalam Tabel.....	47
Gambar 15 Memilih Proses Transaksi	47
Gambar 16 Melakukan Pembayaran	48
Gambar 17 Output Selesai Pembayaran	48
Gambar 18 Memilih Data yang Tidak Ada Dalam Sistem	49
Gambar 19 Menginputkan Jumlah yang Kurang.....	49
Gambar 20 Melakukan Pembayaran Ulang	50

DAFTAR SIMBOL

	Halaman
Simbol 1 Metode Waterfall	20
Simbol 2 Perancangan Kode	23
Simbol 3 Flowchart	24
Simbol 4 DFD Level 0	25
Simbol 5 Data Flow Diagrams.....	26

DAFTAR ISI

	Halaman
Daftar Tabel	ii
Daftar Gambar	iii
Daftar Simbol.....	iv
Daftar Isi.....	v
Kata Pengantar.....	viii

BAB I PENDAHULUAN

1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan dan Manfaat	3
1.4.1 Tujuan	3
1.4.2 Manfaat.....	3

BAB II KAJIAN PUSTAKA

2.1 Landasan Teori	4
2.1.1 Pengertian Aplikasi	4
2.1.2 Pengertian Kasir dan Program Kasir	4
2.1.3 Pengertian Restoran.....	5
2.1.4 Pengertian Transaksi.....	6
2.1.5 Pengertian Penjualan	6

2.1.6 Transaksi Penjualan	6
2.1.7 Konsep Dasar Sistem Informasi Kasir	7
2.2 Alat Perancangan.....	7
2.2.1 Data Flow Diagram	7
2.2.2 Flowchart	9
2.3 Sistem Basis Data.....	11
2.3.1 Definisi Basis Data.....	11
2.3.2 Normalisasi	12
2.3.3 ERD (Entity Relationship Diagram).....	14
2.3.4 Struktur Tabel	15
2.4 Bahasa Pemrograman Java.....	16
2.4.1 Java Development Kit (JDK).....	16
2.5 Aplikasi Netbeans	17
2.6 Waterfall	17
2.7 Teknik Pengumpulan Data.....	18
2.7.1 Studi Pustaka.....	18

BAB III ANALISIS DAN PERANCANGAN SISTEM

3.1 Waktu dan Tempat Penelitian	19
3.1.1 Waktu Penelitian.....	19
3.1.2 Tempat Penelitian.....	19
3.2 Metode Waterfall	19
3.3 ANALISIS SISTEM.....	20
3.3.1 Analisis Sistem Program Kasir.....	20

3.3.2 Struktur Organisasi	20
3.3.3 Aliran Informasi.....	21
3.4 PERANCANGAN SISTEM	22
3.4.1 Performansi Sistem	22
3.4.2 Batasan Sistem.....	23
3.4.3 Perancangan Kebutuhan Pengelolaan Data.....	23
3.4.4 Perancangan Program Aplikasi	24
3.4.5 Relasi Tabel.....	27
3.4.6 Struktur Tabel	28
3.5 Instrumen Penelitian	32
3.5.1 Perangkat Keras	32
3.5.2 Perangkat Lunak.....	32
3.6 Jadwal Penelitian	33

BAB IV HASIL DAN ANALISA

4.1 Hasil Penelitian	34
4.1.1 Pembuatan Aplikasi.....	34
4.1.2 Pengoperasian Aplikasi	50
4.1.3 Pengujian Aplikasi	51
4.2 Pembahasan	51

BAB V KESIMPULAN DAN SARAN

5.1 Kesimpulan	53
----------------------	----

5.2 Saran	53
-----------------	----

DAFTAR PUSTAKA

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan rahmat-Nyalah sehingga penulis dapat menyelesaikan proposal penelitian yang berjudul “Aplikasi Kasir Untuk Restoran (Khusus Bagian Transaksi Penjualan)” tepat pada waktunya.

Adapun tujuan dari penulisan proposal penelitian ini adalah untuk mempelajari cara pembuatan proposal pada Universitas Tomakaka Mamuju.

Pada kesempatan ini, penulis hendak menyampaikan terima kasih kepada semua pihak yang telah memberikan dukungan moril maupun materiil sehingga proposal penelitian ini dapat selesai. Ucapan terima kasih penulis tujukan kepada seluruh pihak yang telah membantu dalam penyelesaian proposal ini hingga selesai.

Meskipun telah berusaha menyelesaikan proposal penelitian ini sebaik mungkin, penulis menyadari bahwa proposal penelitian ini masih ada kekurangan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari para pembaca guna menyempurnakan segala kekurangan dalam penyusunan proposal penelitian ini. Akhir kata, penulis berharap semoga proposal penelitian ini berguna bagi para pembaca dan pihak-pihak lain yang berkepentingan.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan perkembangan teknologi dan informasi yang begitu cepat, semua aspek dan bidang pekerjaan seluruhnya menggunakan komputer. Teknologi yang semakin berkembang tentunya sangat berpengaruh di dunia bisnis. Bisnis dan pengusaha adalah satu kesatuan yang utuh yang tidak dapat dipisahkan.

Tantangan baru bagi penyedia jasa yang khususnya bergelud di bidang teknologi atau IT dengan mengikuti tren global perkembangan zaman dan teknologi-teknologi baru yang terus dirilis. Jurusan Teknik Informatika merupakan bidang yang mampu menyediakan jasa berupa pembuatan aplikasi dan analisis sistem serta dapat menyelesaikan masalah yang ada dalam sistem.

Pada pusat-pusat perbelanjaan dan restoran semua sudah menggunakan kasir digital yang memudahkan transaksi antara pembeli dan penjual. Kebanyakan pengusaha saat ini menggunakan aplikasi kasir untuk memudahkan transaksi pada bisnis yang dijalankannya. Aplikasi kasir biasa disebut dengan istilah *Point Of Sale* (POS) yaitu suatu program kasir yang fungsinya untuk membantu proses transaksi

pada sebuah bisnis. Fungsi lain dari *Point Of Sale* adalah mencatat setiap transaksi serta mengelola stok.

Studi kasus pada sebuah restoran yang membutuhkan aplikasi kasir yang mencakup detail menu restoran, detail harga, serta detail transaksi penjualan. Oleh karenanya, penulis mengangkat judul yaitu Aplikasi Kasir Untuk Restoran yang bertujuan memudahkan suatu bisnis restoran dalam mengelola transaksi penjualan.

1.2 Rumusan Masalah

Berdasarkan penjelasan latar belakang diatas, maka permasalahan yang dapat diidentifikasi adalah pengelolaan data dibagian penjualan belum dapat disajikan dengan cepat karena sistem yang berjalan masih dilakukan secara manual. Semua itu memerlukan pengolahan data yang baik untuk mendapatkan informasi yang relavan, tepat dan akurat agar tidak terjadi keterlambatan dalam penyampaian atau penyajiannya.

Oleh karenanya penulis mengangkat masalah yang akan dikaji dalam proposal ini yaitu membuat program aplikasi kasir sederhana yang dapat memudahkan sebuah restoran dalam proses transaksi penjualan yang dilakukan secara digital secara tepat dan akurat.

1.3 Batasan Masalah

Pembuatan program ini dibatasi hanya pada bagian penjualan saja, serta sistem ini hanya digunakan terbatas pada restoran dan hanya digunakan pada bagian penjualan atau kasir saja. Sistem ini juga dibatasi hanya pada proses pemasukan dan pengeluaran data serta transaksi penjualan yang berlaku.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan dari pembuatan program aplikasi kasir sesuai yang dijelaskan pada rumusan masalah dan latar belakang yaitu :

1. Untuk memberikan informasi yang diperlukan tentang transaksi penjualan kepada setiap bagian yang berhubungan dengan penjualan.

1.4.2 Manfaat

Adapun manfaat dari pembuatan aplikasi kasir ini yaitu :

1. Membuat suatu program aplikasi kasir sederhana yang membantu kasir dalam melakukan tugasnya pada proses transaksi penjualan.
2. Memberikan layanan informasi dibidang transaksi penjualan dan memberikan laporan faktur.

BAB II

KAJIAN PUSTAKA

2.1 Landasan Teori

2.1.1 Pengertian Aplikasi

Secara istilah pengertian aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju.

Menurut kamus komputer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputasi yang diinginkan atau diharapkan maupun pemrosesan data yang di harapkan.

Pengertian aplikasi menurut Kamus Besar Bahasa Indonesia yaitu “Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu” (Juansyah).

2.1.2 Pengertian Kasir dan Program Kasir

Kasir adalah seorang yang memegang uang atau seorang yang bertugas menerima dan membayarkan uang (KBBI).

Menurut (Anwar Muthohari) Program kasir atau sistem kasir adalah serangkaian prosedur yang dirancang untuk menganalisa dan menghasilkan suatu hasil rincian dalam bertransaksi seperti mengolah data barang, mengolah data penjualan, bukti transaksi konsumen, mengolah data suplier dan mengontrol keuangan melalui laporan-laporan data tersebut yang berfungsi untuk mempermudah pemilik usaha, konsumen maupun pegawai kasir (Aplikasi and Meng).

2.1.3 Pengertian Restoran

Definisi restoran berbeda dengan populer *catering* (seperti: kafe, *steak house*, *coffee shops*). Perbedaan restoran dengan populer *catering* adalah dalam hal tujuan dan historisnya, menurut Lillicrap dan Cousins (1994, p. 4).

Tujuan restoran adalah menyajikan makanan dan minuman umumnya pada harga tinggi dengan tingkat pelayanan yang tinggi, Dari sisi historisnya, restoran awalnya berasal dari restoran-restoran hotel dengan pelayanan yang bersifat formal, sampai kemudian mulailah para *chef* mendirikan bisnis sendiri (Rachmawati).

2.1.4 Pengertian Transaksi

Menurut Skousen (2009:71) yang dikutip dalam bukunya yang berjudul Pengantar Akuntansi Keuangan yang menyatakan bahwa pengertian transaksi adalah pertukaran barang dan jasa (baik individu, perusahaan-perusahaan dan organisasi lain) kejadian lain yang memiliki pengaruh ekonomi atas bisnis.

2.1.5 Pengertian Penjualan

Menurut (Marwan, 1991) penjualan adalah suatu usaha yang terpadu untuk mengembangkan rencana- rencana strategis yang diarahkan pada usaha pemuasan kebutuhan dan keinginan pembeli, guna mendapatkan penjualan yang menghasilkan laba.

Penjualan merupakan sumber hidup suatu perusahaan, karena dari penjualan dapat diperoleh laba serta suatu usaha memikat konsumen yang diusahakan untuk mengetahui daya tarik mereka sehingga dapat mengetahui hasil produk yang dihasilkan. Menurut Winardi (1982).

2.1.6 Transaksi Penjualan

Penjualan adalah transaksi antara penjual dan pembeli yang kemudian diikuti dengan proses menyerahkan produk yang berupa barang atau pun jasa. Dalam suatu bisnis, proses penjualan adalah kunci suksesnya suatu bisnis.

Pengertiannya secara umum adalah kegiatan jual beli dijalankan oleh dua belah pihak atau lebih dengan alat pembayaran yang sah.

2.1.7 Konsep Dasar Sistem Informasi Kasir

Konsep Dasar Sistem Informasi Kasir adalah sistem yang memberikan kemudahan mengolah informasi mengenai pembayaran dan penerimaan uang dalam sebuah transaksi penjualan dan pembelian sehingga tidak lagi diperlukan pembukuan atau kearsipan secara manual. Sehingga lebih efisien waktu dan biaya serta meminimalisir terjadinya kesalahan perhitungan saat proses pembayaran dilakukan. (Lumbantoruan and Hidayat)

2.2 Alat Perancangan

2.2.1 Data Flow Diagram

Data Flow Diagram merupakan suatu model logika data atau proses, yang dibuat untuk menggambarkan darimana asal data dan tujuan dari data tersebut, dimana data disimpan, proses apa yang menghasilkan data tersebut, dan interaksi antara data yang tersimpan serta proses yang dikenakan pada data tersebut (Krisnanto, 2008). DFD merupakan representasi grafis dari aliran

suatu data yang melalui sebuah proses (Balamuralithara dan Chong, 2011).

DFD menggunakan empat simbol dasar yang menunjukkan entitas, proses, alur data, dan data *store* yang digunakan untuk melacak dan menggambarkan pergerakan data/informasi (Balamuralithara dan Chong, 2011).

DFD sering digunakan untuk merancang suatu sistem, baik yang telah ada ataupun sistem baru yang dikembangkan secara logika, tanpa mempertimbangkan lingkungan fisik, dimana data tersebut mengalir atau disimpan. Dalam kegiatan analisis dan desain sistem tersebut, terdapat aturan dalam visualisasinya, yang disebut dengan *balanced fragment*. (Martasubrata and Priyadi).

2.2.1.1 Jenis-Jenis DFD

Jenis-jenis DFD dibagi menjadi tiga tingkatan, dimana masing-masing level tersebut menggambarkan detail dari level sebelumnya, berikut penjelasan tiga jenis DFD tersebut :

1. Level 0 (Diagram Konteks) Level ini merupakan sebuah proses yang berada di level pusat.
2. Level 1 (Diagram 0) Level ini merupakan sebuah proses yang terdapat di level 0 yang dipecahkan

menjadi beberapa proses lainnya. Sebaiknya maksimum 7 proses untuk sebuah diagram konteks.

3. Level 2 (Diagram Rinci) Pada level ini merupakan diagram yang merincikan diagram level 1. Tanda * pada proses menandakan bahwa proses tersebut tidak dapat dirincikan lagi. Penomoran yang dilakukan berdasarkan urutan proses.

2.2.2 Flowchart

Flowchart adalah alat pemetaan sederhana yang menunjukkan urutan tindakan dalam proses dalam bentuk yang mudah dibaca dan dikomunikasikan. Menurut Tague (2005).

Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urut-urutan prosedur dari suatu program. Flowchart menolong analis dalam untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian (Ilham Akhsanu Ridlo).

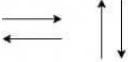


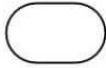

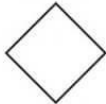
Tujuan digunakannya *flowchart* antara lain:

1. Untuk mengembangkan pemahaman tentang bagaimana proses dilakukan.
2. Untuk mempelajari perbaikan proses.







3. Untuk berkomunikasi dengan orang lain bagaimana proses dilakukan.
4. Untuk keperluan komunikasi yang lebih baik di antara orang-orang yang terlibat dalam proses yang sama.
5. Untuk mendokumentasikan proses.
6. Untuk merencanakan sebuah kegiatan.

2.2.2.1 Simbol *Flowchart*

Pada dasarnya simbol-simbol dalam *flowchart* memiliki arti yang berbeda-beda. Berikut adalah simbol-simbol yang sering digunakan dalam proses pembuatan *flowchart*.

	Flow Simbol yang digunakan untuk menggabungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga dengan Connecting Line.
	On-Page Reference Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang sama.
	Off-Page Reference Simbol untuk keluar - masuk atau penyambungan proses dalam lembar kerja yang berbeda.
	Terminator Simbol yang menyatakan awal atau akhir suatu program.
	Process Simbol yang menyatakan suatu proses yang dilakukan komputer.
	Decision Simbol yang menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu ya dan tidak.

Gambar 1
Simbol Flowchart 1

	Input/output Simbol yang menyatakan proses input atau output tanpa tergantung peralatan.
	Manual Operation Simbol yang menyatakan suatu proses yang tidak dilakukan oleh komputer.
	Document Simbol yang menyatakan bahwa input berasal dari dokumen dalam bentuk fisik, atau output yang perlu dicetak.
	Predefine Proses Simbol untuk pelaksanaan suatu bagian (sub-program) atau prosedur.
	Display Simbol yang menyatakan peralatan output yang digunakan.
	Preparation Simbol yang menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberikan nilai awal.

Gambar 2
Simbol Flowchart 2

2.3 Sistem Basis Data

2.3.1 Definisi Basis Data

Menurut Elmasri & Navathe (2013) basis data adalah Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

Basis data juga dapat diartikan sebagai kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan, Date C.J. (2014).

Basis data mutlak dibutuhkan pada suatu sistem informasi, dimana data akan disimpan dan diolah untuk menjadi informasi-informasi yang penting bagi suatu perusahaan atau organisasi. Jenis basis data yang masih sangat banyak digunakan adalah basis data relasional. Dimana data disimpan didalam baris pada sebuah tabel yang memiliki relasi dengan tabel lain.

Pada basis data relasional terdapat banyak pengaturan yang dikenal dengan *primary key*, *foreign key*, *record*, *field*/ kolom/ *attribute*, *domain*/ *data type* yang menjaga agar data dapat diproses dengan cepat dan mudah, penghematan ruang penyimpanan, serta data yang dihasilkan akurat, Fathansyah (2012).

2.3.2 Normalisasi

Normalisasi merupakan sebuah teknik dalam *logical* desain sebuah basis data relasional yang mengelompokkan atribut dari suatu relasi sehingga membentuk struktur relasi yang baik (tanpa redudansi), Fathansyah (2012).

Pada ilmu basis data, normalisasi digunakan untuk menghindari terjadinya berbagai anomali data dan tidak konsistensinya data. Ini merupakan fungsi secara umum. Dalam beberapa kasus normalisasi ini sangat penting untuk menunjang kinerja basidata dan memastikan bahwa data dalam basidata

tersebut aman dan tidak terjadi kesalahan jika mendapat perintah SQL terutama DML yaitu *update*, *insert*, dan *delete*. (Puspitasari et al.).

Tujuan dari normalisasi adalah untuk :

1. Untuk menghilangkan kerangkapan data sehingga meminimumkan pemakaian *storage* yang dipakai oleh *base relations* (file)
2. Untuk mengurangi kompleksitas
3. Untuk mempermudah pemodifikasian data Contoh

Gambaran proses normalisasi adalah :

1. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat,
2. Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

Untuk melakukan proses tersebut dibutuhkan beberapa tahapan. Tahapan dalam normalisasi dimulai dari tahap paling ringan (1NF) hingga paling ketat (5NF). Biasanya hanya sampai pada tingkat 3NF atau BCNF karena sudah cukup memadai untuk menghasilkan tabel-tabel yang berkualitas baik.

2.3.3 ERD (*Entity Relationship Diagram*)

“*Entity Relationship Diagram* (ERD) adalah suatu rancangan atau bentuk hubungan suatu kegiatan didalam sistem yang berkaitan langsung dan mempunyai fungsi didalam proses tersebut” [Yasin, 2012].

ERD adalah sebuah model data yang menggunakan beberapa notasi untuk menggambarkan data dalam hal entitas dan relasi yang digambarkan oleh data tersebut. (Rahmawati et al.)

Basis data dirancang dengan ERD (*Entity Relationship Diagram*) yang memberikan gambaran untuk data model yang berdasarkan konsep atau data model semantik. (Wellem)

2.3.4 Struktur Tabel

Desain struktur tabel database yang fleksibel akan meningkatkan kualitas Sistem Informasi, sebaliknya desain struktur tabel database yang tidak fleksibel berpotensi menimbulkan masalah dan kegagalan implementasi Sistem Informasi. Struktur tabel database seharusnya dirancang sedemikian rupa sehingga fleksibel terhadap perubahan kebutuhan informasi (dan data) untuk memenuhi kebutuhan para penggunaanya. (Ninla Elmawati Falabiba et al.)

Saran umum untuk memperoleh desain struktur tabel database yang fleksibel, adalah :

1. Meningkatkan fleksibilitas database dimana kebutuhan-kebutuhan baru dapat diakomodasi dengan tabel-tabel database yang ada,
2. Mengurangi biaya pembuatan dan pemeliharaan aplikasi-aplikasi,
3. Meningkatkan penggunaan kembali dan penggunaan bersama antar aplikasi,
4. Mengurangi *software* yang dibutuhkan untuk membuat aplikasi, dan
5. Mengurangi duplikasi dan inkonsistensi informasi

2.4 Bahasa Pemrograman Java

Java dikembangkan oleh *Sun Microsystems* pada Agustus 1991. Java disebut juga merupakan hasil perpaduan sifat dari sejumlah bahasa pemrograman, yaitu C dan C++. Pemrograman Java bersifat tidak bergantung pada *platform*, yang artinya, java dapat dijalankan pada sembarang komputer dan bahkan pada sembarang sistem operasi. Sebagaimana halnya C++, salah satu bahasa yang mengilhami Java, Java juga merupakan bahasa pemrograman berorientasi objek. Sebagai bahasa pemrograman berorientasi objek, Java menggunakan kelas untuk membentuk suatu objek.

Karakteristik Java antara lain adalah berorientasi objek (*object-oriented*), terdistribusi (*distributed*), sederhana (*simple*), aman (*secure*), *interpreted*, *robust*, *multithreaded*, dan dinamis.

2.4.1 Java Development Kit (JDK)

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang digunakan untuk mengembangkan perangkat lunak berbasis Java, sedangkan JRE adalah sebuah implementasi dari *Java Virtual Machine* yang benar-benar digunakan untuk menjalankan program java. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber *compiler java*, *bundling*, *debuggers*, *development libraries* dan lain sebagainya [5].

2.5 Aplikasi Netbeans

Menurut Wahana Komputer (2012:15) NetBeans adalah sebuah IDE (*Integrate Development Envirotment*) *open source* yang sering disosialisasikan dengan Java. NetBeans merupakan salah satu proyek *open source* yang disponsori oleh *Sun Microsystem*. Proyek ini berdiri pada tahun 2000 dan telah menghasilkan 2 produk, yaitu IDE dan NetBeans Platform.

2.6 Waterfall

Menurut Sholikhah, Sairan, dan Syamsiah (2017:47), menjelaskan bahwa, “*Waterfall* merupakan model klasik yang memiliki sifat berurut dalam merancang *software*”.

Metode *waterfall* adalah hal yang menggambarkan pendekatan secara sistematis dan juga berurutan (*step by step*) pada sebuah pengembangan perangkat lunak. Tahapan dengan spesifikasi kebutuhan pengguna lalu berlanjut melalui tahapan-tahapan perencanaan yaitu *planning*, permodelan, konstruksi, sebuah *system* dan penyerahan sistem kepada pengguna, dukungan pada perangkat lunak lengkap yang dihasilkan [Novitasari 2018]. (Kurniawan et al.)

2.7 Teknik Pengumpulan Data

Pengumpulan data merupakan salah satu tahapan sangat penting dalam penelitian. Teknik pengumpulan data yang benar akan menghasilkan data yang memiliki kredibilitas tinggi, dan sebaliknya. Oleh karena itu, tahap ini tidak boleh salah dan harus dilakukan dengan cermat sesuai prosedur dan ciri-ciri penelitian kualitatif (sebagaimana telah dibahas pada materi sebelumnya).

2.7.1 Studi Pustaka

Studi pustaka juga merupakan salah satu teknik pengumpulan data yang juga banyak digunakan oleh para peneliti.

Teknik pengumpulan data studi pustaka dilakukan dengan cara mengumpulkan data yang relevan atau sesuai yang dibutuhkan untuk penelitian dari buku, artikel ilmiah, berita, maupun sumber kredibel lainnya yang reliabel dan juga sesuai dengan topik penelitian yang dilakukan.

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Waktu dan Tempat Penelitian

3.1.1 Waktu Penelitian

Waktu penelitian dilaksanakan dalam kurung waktu selama 7 (tujuh) minggu pada tahun 2021.

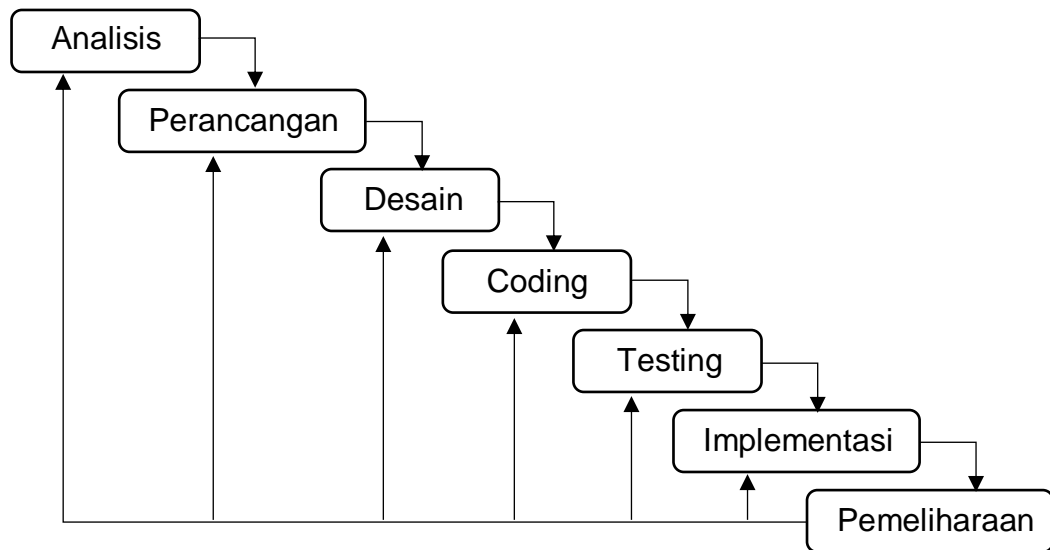
3.1.2 Tempat Penelitian

Penelitian ini rencananya dilaksanakan di Kabupaten Mamuju Provinsi Sulawesi Barat.

3.2 Metode *Waterfall*

Salah satu cara untuk mendapatkan hasil yang optimal dengan mengelola data yang cepat dan akurat maka digunakanlah sistem terkomputerisasi untuk pencapaian hasil tersebut dengan metode *Waterfall* dalam perancangan sistem informasi, dimulai dari menganalisa kebutuhan *software*, merancang kebutuhan *software*, desain, *coding*, *testing*, implementasi, dan pemeliharaan.

Simbol 1

*Metode Waterfall*

3.3 ANALISIS SISTEM

3.3.1 Analisis Sistem Program Kasir

Analisis sistem yang dibuat merupakan tahap selanjutnya yang akan dilakukan setelah tahap analisis permasalahan dan identifikasi kebutuhan. Pada tahap ini akan dijelaskan mengenai perancangan sistem yang akan dibangun.

3.3.2 Struktur Organisasi

Struktur organisasi dapat disebut sebagai suatu pola hubungan yang baik antara komponen-komponen dan bagian dalam organisasi. Struktur merupakan suatu kerangka yang

mewujudkan suatu pola yang tetap dari hubungan antara jabatan dan peran dalam suatu lingkungan kerjasama.

Setiap bisnis baik itu perusahaan, toko maupun restoran memiliki yang namanya organisasi. Pada dasarnya struktur organisasi inilah yang menggambarkan hubungan antara bagian-bagian yang terkait sebagai suatu kerangka yang utuh yang dapat digambarkan dalam grafik.

Adapun tujuan dari pengorganisasian dalam sebuah bisnis khususnya yang dimaksud adalah restoran, untuk menghindari penyalahgunaan wewenang dan tanggung jawab dari bagian-bagian organisasi yang ada didalamnya. Agar manajemen dapat berjalan dengan baik, maka salah satu fungsi dasarnya yaitu, organisasi harus disusun dengan baik pula.

3.3.3 Aliran Informasi

3.3.3.1 Kebutuhan Data

Tabel 1
Kebutuhan Data

No	Nama Data	Fungsi	Sumber
1.	Data pembeli/pelanggan		Pembeli /pelanggan
2.	Data makanan yang dibeli	Sebagai dasar pembuatan faktur	Pembeli /pelanggan

3.	Data menu dan harga	Mengetahui nama menu dan harga yang akan disajikan	Administrasi
----	---------------------	--	--------------

3.3.3.2 Informasi yang dihasilkan

Tabel 2
Informasi yang Dihasilkan

No	Nama Data	Fungsi	Sumber
1.	Faktur	Tanda bukti penjualan	Pelanggan, kasir, administrasi
2.	Menu dan harga	Landasan pemberian harga pada menu yang dipesan pelanggan	Administrasi

3.4 PERANCANGAN SISTEM

3.4.1 Performansi Sistem

Setelah menganalisa dan mengevaluasi sistem, maka dibuatlah perancangan sistem aplikasi kasis untuk restoran dalam transaksi penjualan. Perancangan sistem ini diharapkan dapat membantu dan mengatasi permasalahan yang dihadapi sesuai dengan rumusan masalah.

Adapun sasaran atau pengaruh yang ingin atau akan dicapai adalah :

1. Meningkatkan efisiensi dan efektifitas pekerjaan pada bagian penjualan, dalam hal ini adalah kasir.
2. Dapat menghemat tenaga dan mempercepat pekerjaan dan juga menghindari kesalahan dari pekerjaan
3. Mampu menyimpan, memperbaiki, menghapus, menampilkan, dan mencetak data penjualan secara cepat dan tepat
4. Menunjang kegiatan dalam pembuatan laporan

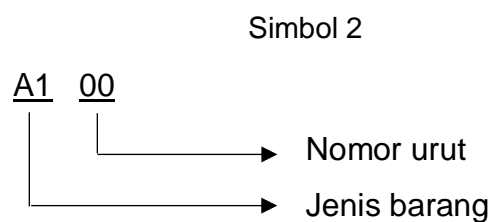
3.4.2 Batasan Sistem

Dalam sistem ini hanya dibahas tentang bagian penjualan yang berhubungan langsung dengan pelanggan dan hubungannya dengan transaksi penjualan di sebuah restoran.

3.4.3 Perancangan Kebutuhan Pengelolaan Data

3.4.3.1 Perancangan Kode

Kode yang digunakan dalam perancangan ini menggunakan 4 karakter, karakter pertama menggunakan huruf, karakter kedua, ketiga dan keempat menggunakan angka. Contoh format kodenya yaitu :



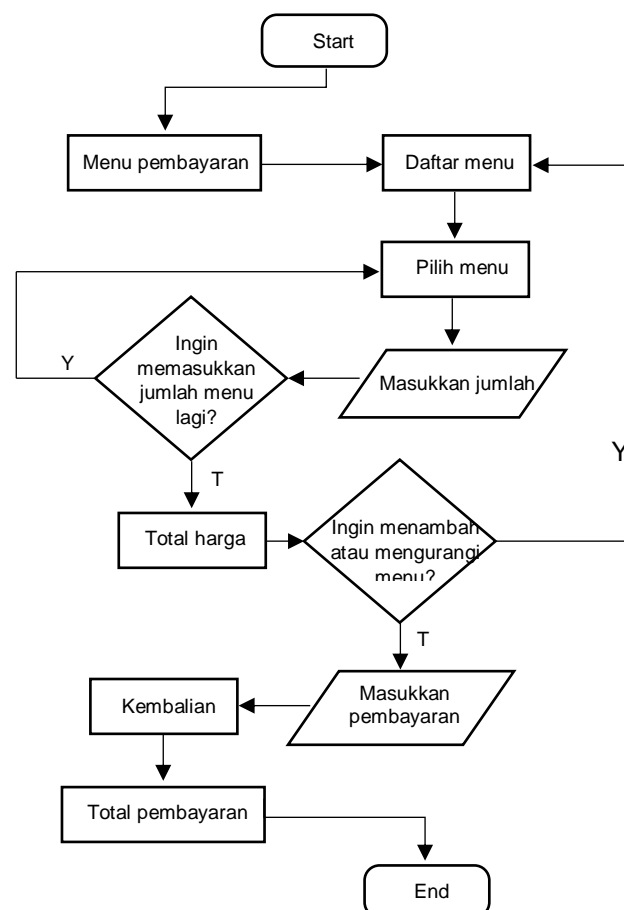
Perancangan Kode

3.4.4 Perancangan Program Aplikasi

3.4.4.1 Alur Sistem (*Flowchart*) Transaksi Penjualan

Alur sistem atau *flowchart* merupakan serangkaian bagian-bagian yang menggambarkan aliran program. Pada *flowchart* sistem ini digambar urutan prosedur dan program aplikasi kasir untuk restoran.

Simbol 3

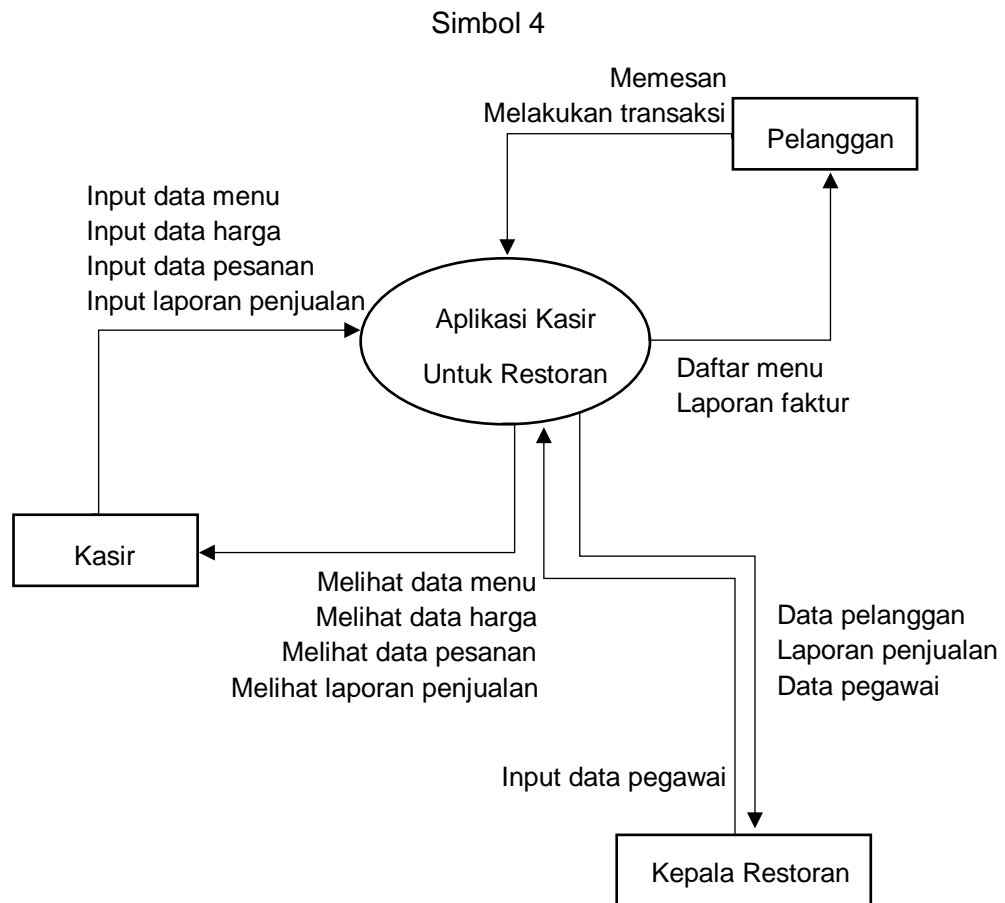


Flowchart

3.4.4.2 Data Flow Diagram Perancangan Aplikasi

Data flow diagram atau diagram alir adalah suatu diagram yang menggambarkan aliran data dari sebuah proses atau sistem. DFD juga menyediakan informasi mengenai luaran dan masukan dari setiap entitas dan proses itu sendiri. Seperti yang sudah dijabarkan pada analisis sistem bagian data flow diagram, berikut gambar DFD mengenai perancangan sistem aplikasi kasir pada sebuah restoran.

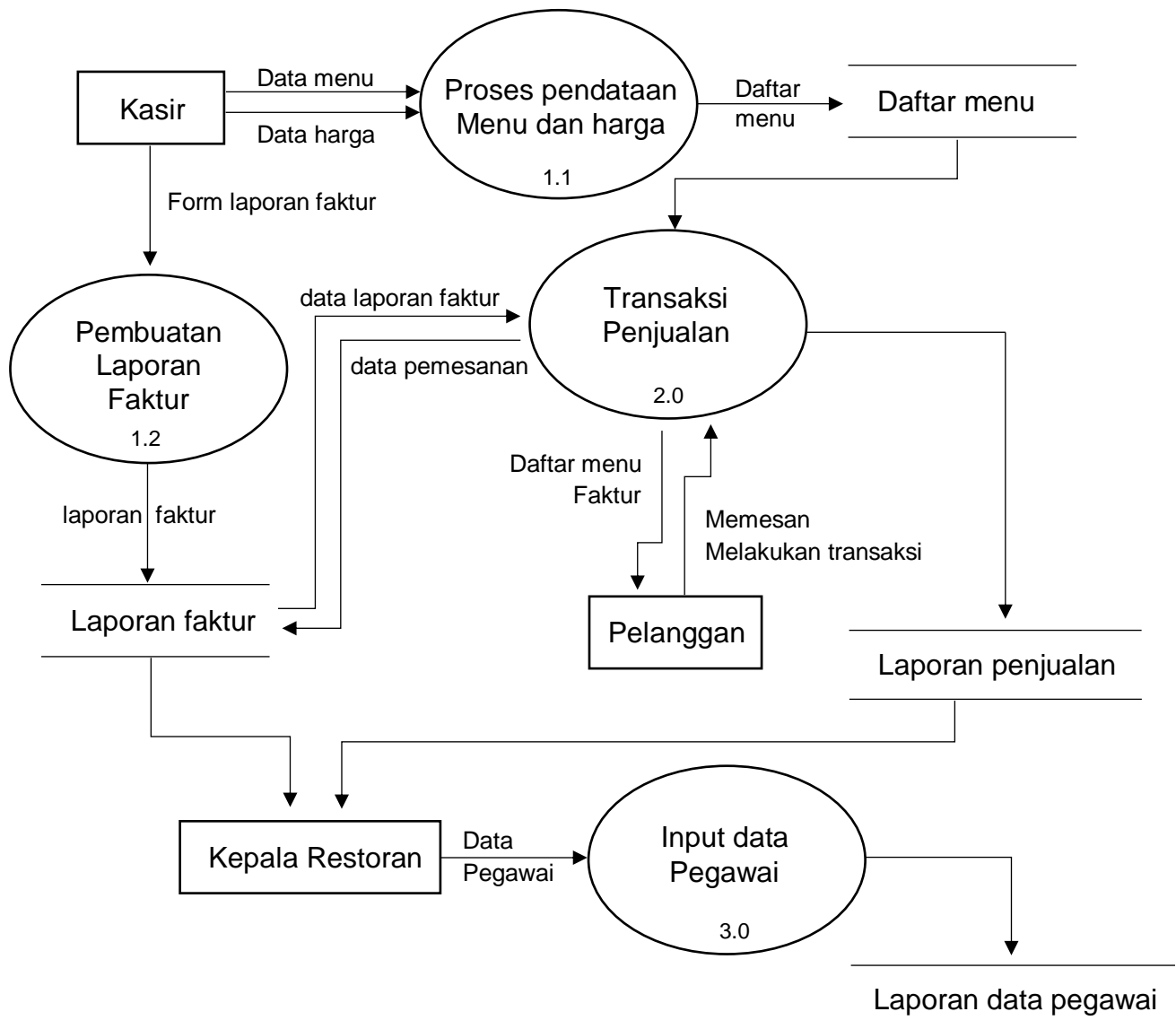
1. DFD Level 0



DFD Level 0

2. Data Flow Diagram Level 1 Proses 1, Level 1 Proses 2,
Level 2 Proses 0, Level 3 Proses 0

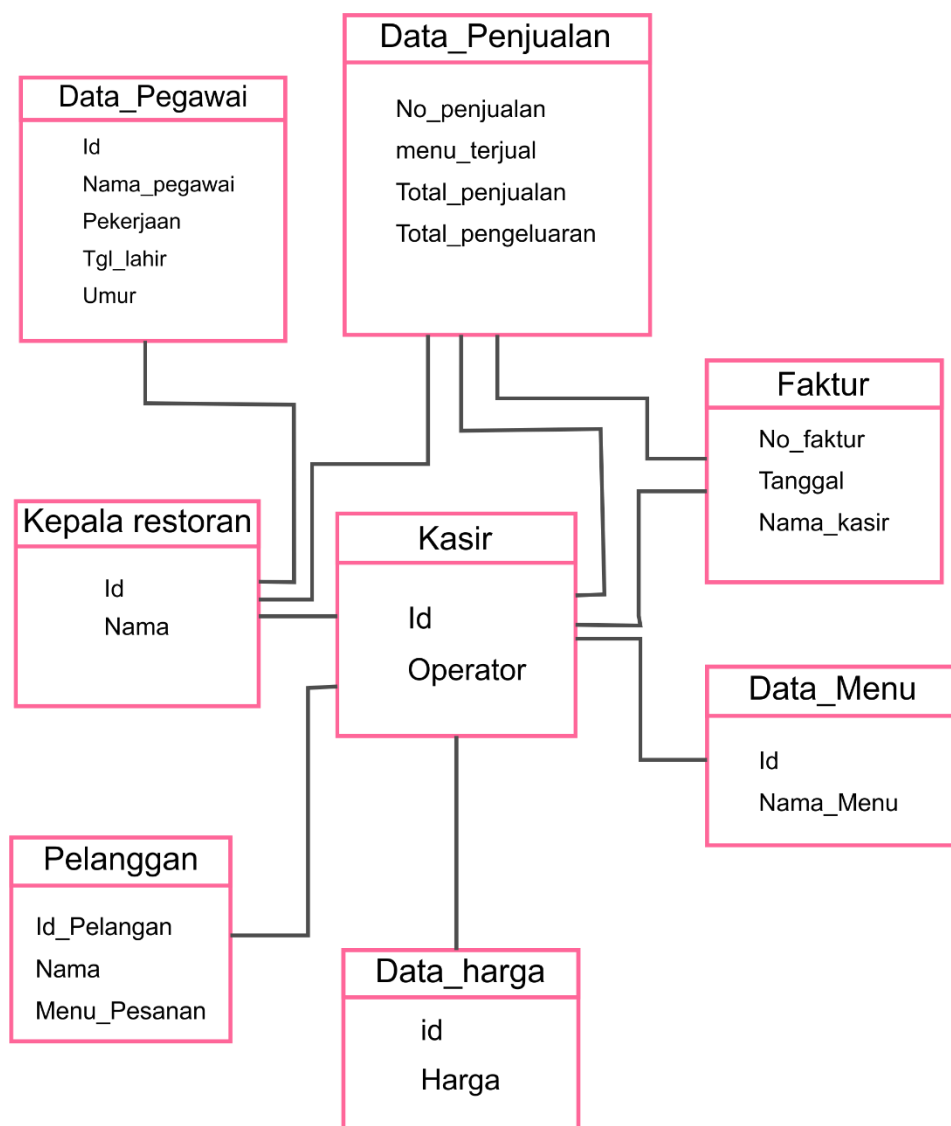
Simbol 5



Data Flow Diagrams

3.4.5 Relasi Tabel

Pada perancangan sistem aplikasi Kasir yang disusun dalam proposal ini, hubungan antar tabel atau relasi tabel yang digunakan yaitu seperti dibawah ini :



Gambar 3
Relasi tabel

3.4.6 Struktur Tabel

1. Tabel pelanggan

Pada tabel pelanggan menyimpan data yang diperlukan dari pelanggan seperti nama dan menu pesanan pelanggan. *Primary key* yang dipakai pada tabel pelanggan ini yaitu *Id_pelanggan*.

Tabel 3
Tabel pelanggan

Nama Field	Tipe Data	Keterangan
Id_pelanggan	<i>Int(11)</i>	<i>Primary Key</i>
Nama	<i>Varchar(40)</i>	<i>Not Null</i>
Menu_pesanan	<i>Varchar(40)</i>	

2. Tabel kasir

Pada tabel kasir menyimpan data yang diperlukan untuk kasir guna keperluan transaksi penjualan, seperti data menu, data harga, dan data pesanan. *Primary key* yang dipakai pada tabel pelanggan ini yaitu *Id_kasir*.

Tabel 4
Tabel kasir

Nama Field	Tipe Data	Keterangan
Id_kasir	<i>Int(11)</i>	<i>Primary Key</i>

operator	<i>Varchar(40)</i>	<i>Not Null</i>
Data_menu	<i>Varchar(40)</i>	
Data_harga	<i>Varchar(40)</i>	
Data_pesanan	<i>Varchar(40)</i>	

3. Tabel Kepala restoran

Pada tabel kepala restoran menyimpan data yang diperlukan kepala restoran untuk keperluan mengelola restoran, seperti nama kepala restoran. *Primary key* yang dipakai pada tabel pelanggan ini yaitu Id.

Tabel 5

Tabel kepala restoran

Nama Field	Tipe Data	Keterangan
Id	<i>Int(11)</i>	<i>Primary Key</i>
Nama	<i>Varchar(40)</i>	<i>Not Null</i>

4. Tabel Harga

Pada tabel harga menyimpan data yang diperlukan tentang harga menu yang ada pada restoran untuk keperluan transaksi penjualan, seperti nama harga. *Primary key* yang dipakai pada tabel pelanggan ini yaitu Id_harga.

Tabel 6
Tabel harga

Nama Field	Tipe Data	Keterangan
Id_harga	<i>Int(11)</i>	<i>Primary Key</i>
Nama_Harga	<i>Varchar(40)</i>	<i>Not Null</i>

5. Tabel menu

Pada tabel menu menyimpan data yang diperlukan tentang data menu yang ada pada restoran untuk keperluan transaksi penjualan, seperti nama menu. *Primary key* yang dipakai pada tabel pelanggan ini yaitu Id_menu.

Tabel 7
Tabel menu

Nama Field	Tipe Data	Keterangan
Id_menu	<i>Int(11)</i>	<i>Primary Key</i>
Nama_menu	<i>Varchar(40)</i>	<i>Not Null</i>

6. Tabel faktur

Pada tabel faktur menyimpan data yang diperlukan tentang data pelanggan sehingga bisa menghasilkan faktur untuk output transaksi penjualan, seperti tanggal, nama kasir. *Primary key* yang dipakai pada tabel pelanggan ini yaitu no_faktur.

Tabel 8
Tabel faktur

Nama Field	Tipe Data	Keterangan
No_faktur	<i>Int(11)</i>	<i>Primary Key</i>
tanggal	<i>Date</i>	<i>Not Null</i>
Nama_kasir	<i>Varchar(40)</i>	

7. Tabel penjualan

Pada tabel menu menyimpan data yang diperlukan tentang data menu yang ada pada restoran untuk keperluan transaksi penjualan, seperti nama menu. *Primary key* yang dipakai pada tabel pelanggan ini yaitu *Id_menu*.

Tabel 9
Tabel penjualan

Nama Field	Tipe Data	Keterangan
No_penjualan	<i>Int(11)</i>	<i>Primary Key</i>
Menu_terjual	<i>Varchar(40)</i>	<i>Not Null</i>
Total_penjualan	<i>Varchar(40)</i>	
Total_pengeluaran	<i>Varchar(40)</i>	

8. Tabel pegawai

Pada tabel pegawai menyimpan data yang diperlukan tentang data pegawai yang ada pada restoran, seperti nama

pegawai, pekerjaan, tanggal lahir dan umur. *Primary key* yang dipakai pada tabel pelanggan ini yaitu *Id_pegawai*.

Tabel 10
Tabel pegawai

Nama Field	Tipe Data	Keterangan
Id_pegawai	<i>Int(11)</i>	<i>Primary Key</i>
Nama_pegawai	<i>Varchar(40)</i>	<i>Not Null</i>
pekerjaan	<i>Varchar(40)</i>	
Tgl_lahir	<i>Date</i>	
Umur	<i>Int</i>	

3.5 Instrumen Penelitian

3.5.1 Perangkat Keras

Dalam melakukan penelitian ini, perangkat keras atau *hardware* yang digunakan yaitu : Laptop dengan merek *Lenovo ideapad 320*, dengan prosesor *Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz 1.99 GHz*, RAM 4,00 GB, dan *System* tipe 64-bit *operating system, x64-based processor*.

3.5.2 Perangkat Lunak

Dalam melakukan penelitian ini, perangkat lunak atau *software* yang digunakan adalah :

- 1) *Firefox* sebagai *preview* pustaka *google scholar* mencari beberapa informasi yang terkait dengan penulisan proposal.
- 2) *Java* sebagai bahasa pemrograman yang digunakan untuk menjalankan salah satu program untuk pengembangan aplikasi.
- 3) *Netbeans* sebagai *software* yang digunakan untuk merancang aplikasi sesuai dengan penulisan proposal ini.
- 4) *Mendeley* sebagai aplikasi untuk mengutip jurnal.
- 5) *Windows 10 pro* sebagai sistem operasi di laptop.
- 6) *Microsoft Word* sebagai *software* yang digunakan untuk menulis penyusunan proposal.

3.6 Jadwal Penelitian

Tabel 11

Jadwal penelitian

No	Kegiatan	Tahun 2021				
		Oktober			November	
		Senin	Sabtu	Minggu	Sabtu	Minggu
1.	Penyusunan dan Pengajuan Judul					
2.	Acc Judul					
3.	Pembuatan Proposal					
4.	Presentasi Proposal					
5.	Revisi Proposal					
6.	Pengumpulan Proposal					

BAB IV

HASIL DAN ANALISA

4.1 Hasil Penelitian

Setelah melewati perancangan sistem dan berakhir dengan pembuatan program yang sesungguhnya, maka hasil yang dicapai oleh penulis adalah sebuah aplikasi kasir sederhana dengan menggunakan bahasa pemrograman Java dengan aplikasi Netbeans. Dari hasil pembuatan aplikasi dan *coding* yang telah yang dibuat, maka dapat disimpulkan bahwa aplikasi berjalan dengan baik sesuai dengan konfigurasi yang diinginkan. Simulasi pada program aplikasi kasir yang telah dibuat ini, bermanfaat untuk membuat transaksi penjualan di sebuah restoran dapat berjalan dengan sistem yang baik dan secara terkomputerisasi.

Dari hasil penelitian ini dapat diuraikan tentang cara membuat dan menjalankan program dengan menggunakan aplikasi Netbeans dengan bahasa pemrograman Java.

4.1.1 Pembuatan Aplikasi

Aplikasi kasir sederhana ini, didesain dengan konsep OOP (*Object Oriented Programming*) menggunakan tiga *class* yang saling berhubungan yaitu *class* kasir_menu, *class* kasir_harga, dan *class* kasir_kembalian. *Class* kasir_menu berhubungan

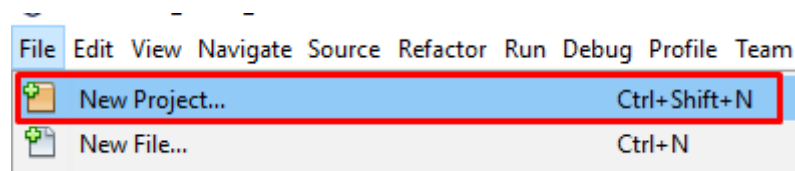
dengan *class* kasir_kembalian. *Class* kasir_harga berhubungan dengan seluruh *class* yang ada. *Class* kasir_kembalian berhubungan dengan *class* kasir_harga. Ketiga *class* saling berhubungan, jika salah satu *class* ada yang hilang atau *error* maka program tidak dapat dieksekusi. Berikut uraiannya.

Adapun perancangan aplikasi ada penelitian ini terbagi menjadi lima tahap antara lain :

1. Pembuatan Projek

Tahapan dalam pembuatan projek uraiannya sebagai berikut :

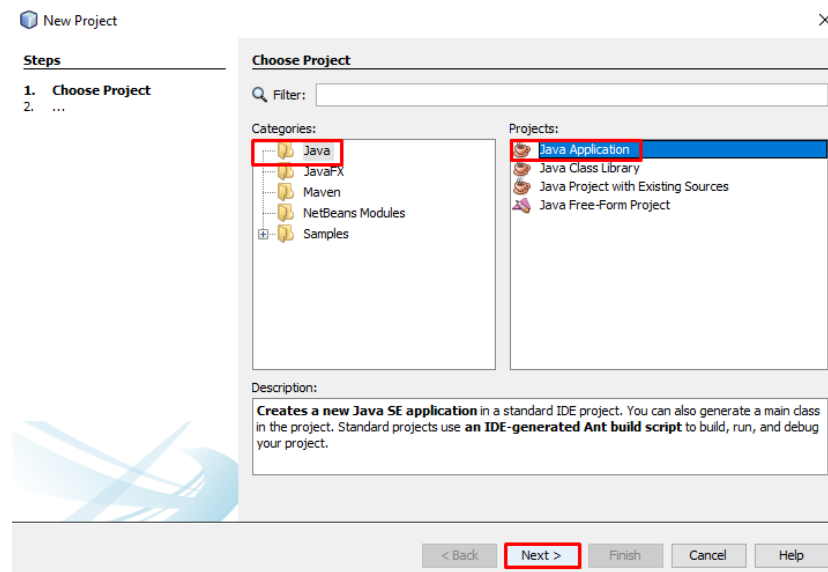
a) Pilih menu file kemudian pilih new projek, seperti gambar di bawah ini



Gambar 4

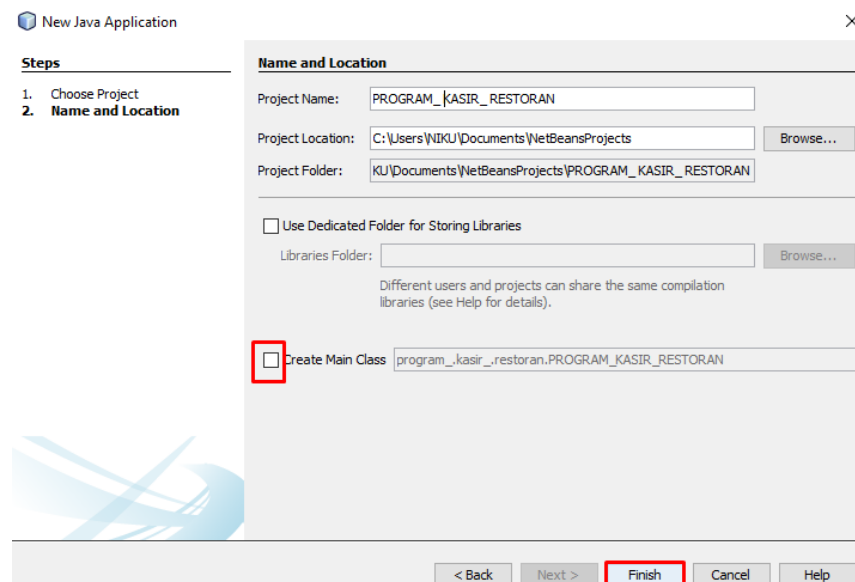
Membuat projek baru

b) Setelah itu pilih java lalu pilih java application kemudian klik next



Gambar 5
Memilih proyek

- c) Setelah itu ketikkan nama proyek yang sesuai, jangan lupa hilangkan ceklis yang ada pada create main class, lalu klik finish

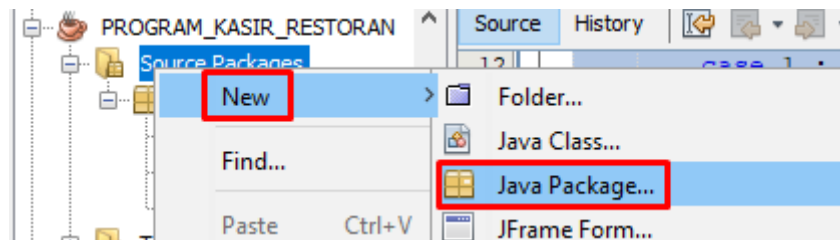


Gambar 6
Membuat nama proyek

2. Pembuatan Package

Tahapan pada pembuatan package dalam program kasir restoran yaitu sebagai berikut :

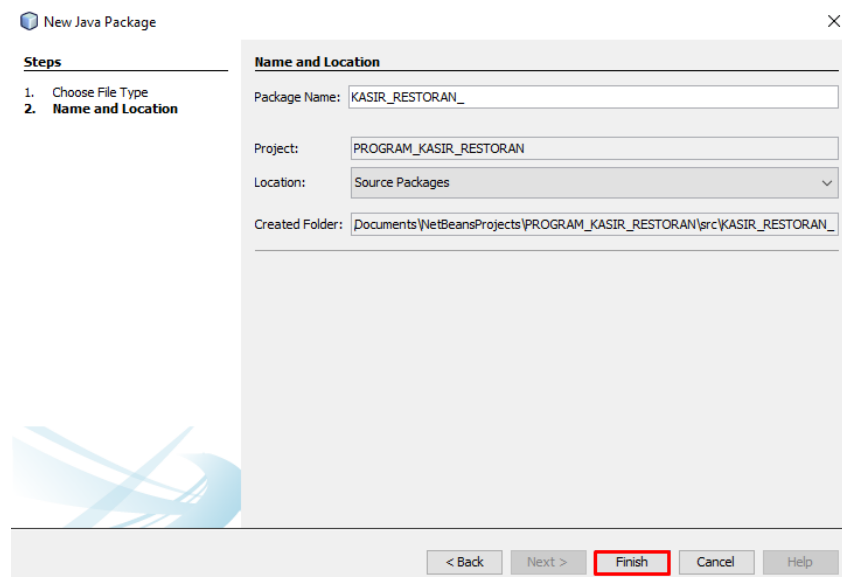
- a. Klik kanan pada source package pilih new kemudian klik java package



Gambar 7

Membuat package

- b. Lalu ketikkan nama package yang sesuai kemudian klik finish



Gambar 8

Memberi nama package

3. Pembuatan Class

- a. Pada pembuatan program kasir restoran, menggunakan tiga class, berikut uraiannya :

1) *Class Kasir_Menu* (kelas master dengan *method*)

Pada *class* ini merupakan *class* utama yang menjalankan program. Penjelasannya sebagai berikut :

- a) *Class kasir_menu* terdiri dari detail menu dan harga yang ada dalam program sesuai daftar menu dan daftar harga di sebuah restoran

- b) *Class kasir_menu* berhubungan dengan *class kasir_kembalian*, dimana ketika menjalankan program, menu dan harga secara otomatis akan di akumulasi saat proses transaksi dilakukan

- c) *Class* ini akan secara terus menerus ditampilkan ketika pengguna masih memilih menu yang diinginkan, dan program akan berakhir pada proses transaksi penjualan di lakukan atau telah melakukan akumulasi seluruh pembayaran

- d) *Class kasir_menu* menggunakan kondisi *if* dan *else* serta menggabungkan dengan kondisi *do & while*

2) *Class Kasir_Harga*

Class kasir_harga merupakan *class* yang berisi deklarasi harga dari setiap menu yang ada. Penjelasan sebagai berikut:

- a) *Class* kasir_harga memiliki fungsi khusus yaitu *class* ini dapat terbaca di seluruh *class* dengan syarat terdapat pada *package* yang sama
- b) Pada *class* kasir_menu, *class* kasir_harga ini hanya dipanggil bukan di hubungkan langsung karena *class* ini dapat terhubung keseluruhan *class*
- c) *Class* kasir_harga menggunakan kondisi pilihan *switch case* dan berhubungan dengan menu yang ada di *class* kasir_menu
- d) *Class* ini berfungsi untuk memilih pilihan yang telah disediakan sistem, jika pilihan yang dipilih tidak ada dalam sistem maka *class* kasir_harga akan menampilkan kode *error* dan hanya dapat memilih pilihan yang ada saja

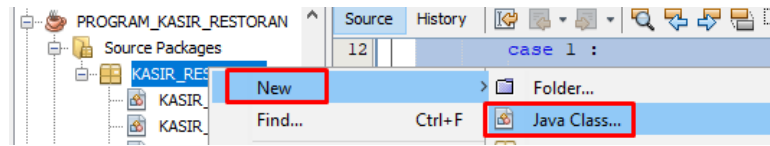
3) *Class* Kasir_Kembalian

Class kasir_kembalian adalah *class* yang berfungsi untuk mengakumulasi seluruh proses transaksi yang terjadi dalam program ketika dijalankan. Penjelasan sebagai berikut :

- a) *Class* kasir_kembalian berhubungan langsung dengan *class* kasir_harga yang digunakan pada proses akumulasi transaksinya
- b) *Class* ini memiliki beberapa objek untuk dapat menyelesaikan akumulasi transaksi dengan bantuan *class_harga* dan *class_menu*
- c) *Class* kasir_kembalian menggunakan fungsi *if else* pada setiap proses dalam objek yang ada
- d) *Class* kasir_kembalian berfungsi pada proses pembayaran atau melakukan transaksi pada bagian total bayar
- e) *Class* ini memiliki keistimewaan dimana pada saat proses pembayaran, ketika pengguna/kasir menginputkan jumlah uang yang kurang dari total harga, maka sistem akan menampilkan kode untuk melunasi pembayaran dan sistem tidak akan berhenti sampai proses pelunasan selesai.

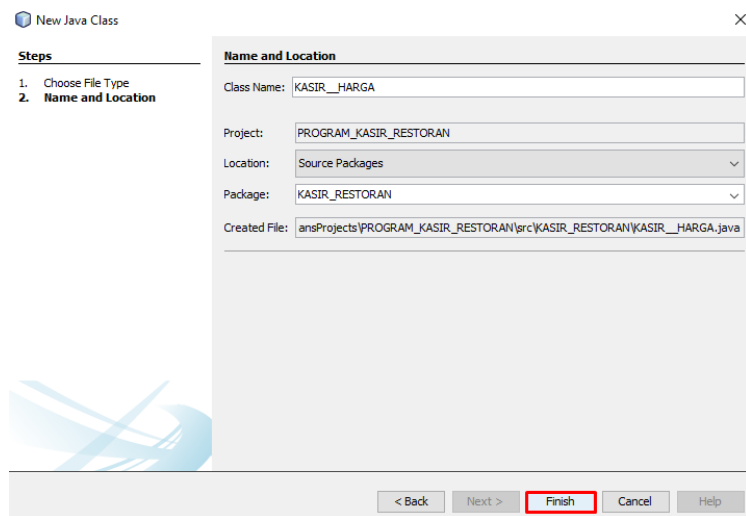
Tahapan dalam pembuatan ketiga *class* diatas pada program kasir restoran yaitu sebagai berikut :

- a) Klik kanan pada source package yang sudah dibuat kemudian pilih new, lalu pilih java class



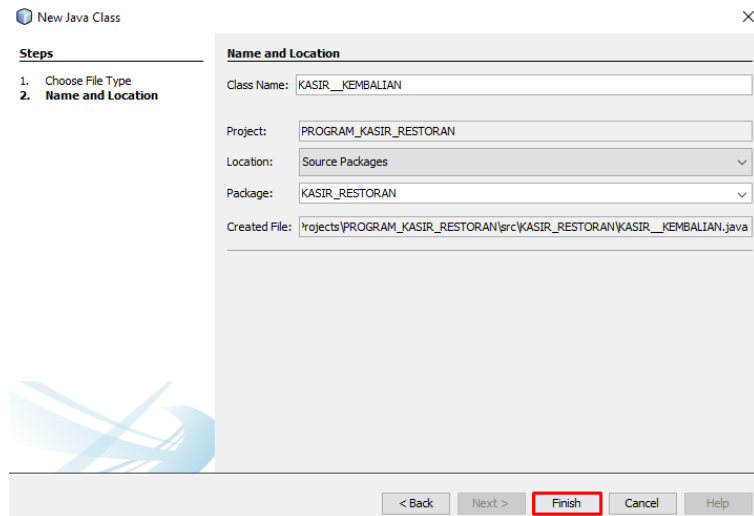
Gambar 9
Membuat class

b) Lalu berikan nama class yang sesuai kemudian klik finish



Gambar 10
Memberi nama class

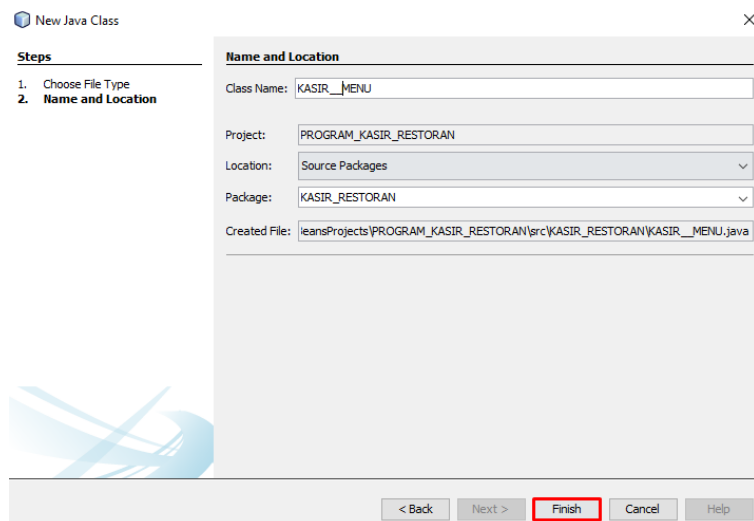
c) Kemudian buat lagi class seperti contoh diatas dengan nama yang berbeda



Gambar 11

Membuat class kedua

- d) Lalu buat class ketiga dengan cara yang sama tapi nama yang berbeda



Gambar 12

Membuat class ketiga

4. Source code pada program

Berikut adalah source code yang digunakan dalam perancangan program kasir restoran :

a. Source code class kasir_harga

```

package KASIR_RESTORAN;
public class KASIR_HARGA {
    public double byr, kembalian, kekurangan, total;
    protected int jumlah, pilihan;
    static final String selamat = "      ~> [RESTORAN NICE]
<~";
    double hitungtotal (int jml){
        int harga;
        switch(pilihan){
            case 0 :
                System.exit(0);
                break;
            case 1 :
                harga = 15000;
                total = total + (harga*jumlah);
                break;
            case 2 :
                harga = 10000;
                total = total + (harga*jumlah);
                break;
            case 3 :
                harga = 20000;
                total = total + (harga*jumlah);
                break;
            case 4 :
                harga = 5000;
                total = total + (harga*jumlah);
                break;
            case 5 :
                harga = 7000;
                total = total + (harga*jumlah);
                break;
            case 6 :
                harga = 10000;
                total = total + (harga*jumlah);
                break;
            case 9 :
                break;
            default :
                System.out.println("ERROR : Input yang Anda
masukkan salah");
                break;
        }
        return total;
    }
}

```

```
}
```

b. Source code class kasir_kembalian

```
package KASIR_RESTORAN;
public class KASIR_KEMBALIAN extends KASIR_HARGA {
    void viewtotal() {
        System.out.println("|=====| PEMBAYARAN
]=====|");
        System.out.println("Total : Rp." +total);
    }
    public double hitungkembalian (double bayar) {
        byr = bayar;
        if (byr > total){
            kembalian = bayar-total;
            return kembalian;
        }
        else {
            kekurangan = (bayar-total)*(-1);
            return kekurangan;
        }
    }
    void viewkembalian(){
        if (byr >= total){
            System.out.println("Kembalian : Rp." +kembalian);
            System.out.println("|=====| TERIMA KASIH
]=====|");
        }
        else {
            System.out.println("Silahkan membayar kembali,
Uang Anda masih kurang Rp." +kekurangan);
        }
    }
    void pelunasan(double bayar){
        byr = bayar;
        System.out.println("Pembayaran ulang : Rp." +byr);
        if (kekurangan > byr) {
            kekurangan = kekurangan - byr;
            System.out.println("Silahkan membayar kembali,
uang Anda masih kurang Rp." + kekurangan);
        }
        else if (kekurangan < byr){
            kembalian = byr - kekurangan;
            kekurangan = 0.0;
            System.out.println("Kembalian : Rp." + kembalian);
        }
    }
}
```

```

        System.out.println("|===== [ TERIMA KASIH
]=====|");
    }
    else if (kekurangan == byr){
        kekurangan = kekurangan + byr;
        System.out.println("|===== [ TERIMA KASIH
]=====|");
    }
}

```

c. Source code class kasir_menu

```

package KASIR_RESTORAN;
import java.util.Scanner;
public class KASIR_MENU extends KASIR_KEMBALIAN{
    Scanner input = new Scanner (System.in);
    public KASIR_MENU (){
        System.out.println(KASIR_HARGA.selamat);
        do {
            System.out.printf("%-10s %-10s\n", "| ===== MENU
===== |", "===== Harga ===== |");
            System.out.printf("%-10s %-10s\n", "| 1. Nasi Goreng
", "| Rp. 15000      |");
            System.out.printf("%-10s %-10s\n", "| 2. Nasi Kuning
", "| Rp. 10000      |");
            System.out.printf("%-10s %-10s\n", "| 3. Nasi
Padang ", "| Rp. 20000      |");
            System.out.printf("%-10s %-10s\n", "| 4. Es Teh    ",
"| Rp. 5000      |");
            System.out.printf("%-10s %-10s\n", "| 5. Es Jeruk   ",
"| Rp. 7000      |");
            System.out.printf("%-10s %-10s\n", "| 6. Es Boba    ",
"| Rp. 10000     |");
            System.out.printf("%-10s %-10s\n", "|
===== ", "===== |");
            System.out.printf("%-10s %-10s\n", "| 9. Selesai dan
Hitung Pembayaran", "|");
            System.out.printf("%-10s %-10s\n", "| 0. Keluar
Program      ", "|");
            System.out.printf("%-10s %-10s\n", "|
===== ", "===== |");
            System.out.print("Masukkan Pilihan : ");
            pilihan = input.nextInt();
            if (pilihan >= 1 && pilihan <=6) {
                System.out.print("Masukkan Jumlah Beli : ");
                jumlah = input.nextInt();
            }
        } while (true);
    }
}

```

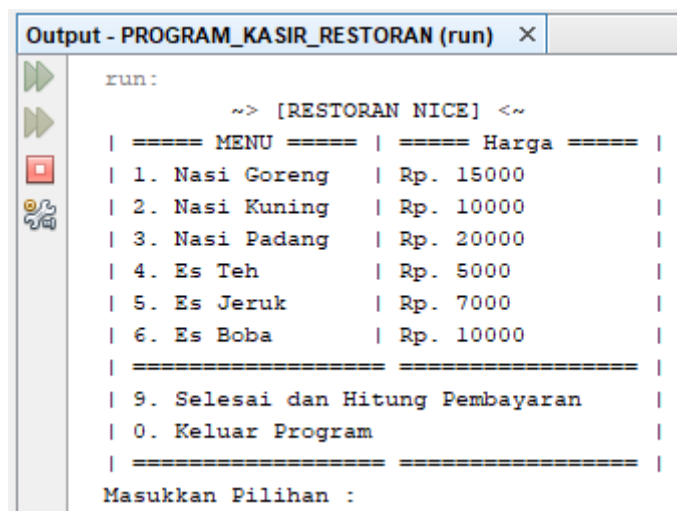
```

    }
    hitungtotal(jumlah);
} while (pilihan != 9);
viewtotal();
System.out.print("Bayar : Rp.");
hitungkembalian(input.nextDouble());
viewkembalian ();
while (total != byr && byr < total){
    if (kekurangan != 0){
        System.out.print("Bayar : Rp.");
        pelunasan (input.nextDouble());
    }
    else {
        break;
    }
}
}
public static void main(String[] args) {
    new KASIR_MENU();
}

```

5. Output program

Berikut adalah gambar pada program yang telah dijalankan/dieksekusi.



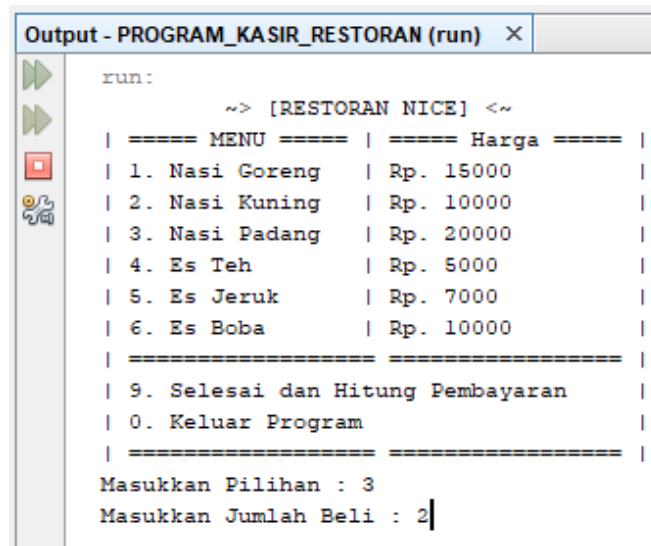
```

Output - PROGRAM_KASIR_RESTORAN (run) x
run:
    ~> [RESTORAN NICE] <~
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng         | Rp. 15000                |
| 2. Nasi Kuning         | Rp. 10000                |
| 3. Nasi Padang         | Rp. 20000                |
| 4. Es Teh              | Rp. 5000                 |
| 5. Es Jeruk            | Rp. 7000                 |
| 6. Es Boba             | Rp. 10000                |
| =====               | =====                 |
| 9. Selesai dan Hitung  | Pembayaran               |
| 0. Keluar Program      |                           |
| =====               | =====                 |
Masukkan Pilihan :

```

Gambar 13

Output ketika dijalankan



```

run:
    ~> [RESTORAN NICE] <~
    | ===== MENU ===== | ===== Harga ===== |
    | 1. Nasi Goreng      | Rp. 15000 |
    | 2. Nasi Kuning      | Rp. 10000 |
    | 3. Nasi Padang      | Rp. 20000 |
    | 4. Es Teh           | Rp. 5000  |
    | 5. Es Jeruk         | Rp. 7000  |
    | 6. Es Boba          | Rp. 10000 |
    | ===== |
    | 9. Selesai dan Hitung Pembayaran |
    | 0. Keluar Program |
    | ===== |
    Masukkan Pilihan : 3
    Masukkan Jumlah Beli : 2
  
```

Gambar 14

Menambahkan data ke dalam program



```

run:
    ~> [RESTORAN NICE] <~
    | ===== MENU ===== | ===== Harga ===== |
    | 1. Nasi Goreng      | Rp. 15000 |
    | 2. Nasi Kuning      | Rp. 10000 |
    | 3. Nasi Padang      | Rp. 20000 |
    | 4. Es Teh           | Rp. 5000  |
    | 5. Es Jeruk         | Rp. 7000  |
    | 6. Es Boba          | Rp. 10000 |
    | ===== |
    | 9. Selesai dan Hitung Pembayaran |
    | 0. Keluar Program |
    | ===== |
    Masukkan Pilihan : 3
    Masukkan Jumlah Beli : 2
    | ===== MENU ===== | ===== Harga ===== |
    | 1. Nasi Goreng      | Rp. 15000 |
    | 2. Nasi Kuning      | Rp. 10000 |
    | 3. Nasi Padang      | Rp. 20000 |
    | 4. Es Teh           | Rp. 5000  |
    | 5. Es Jeruk         | Rp. 7000  |
    | 6. Es Boba          | Rp. 10000 |
    | ===== |
    | 9. Selesai dan Hitung Pembayaran |
    | 0. Keluar Program |
    | ===== |
    Masukkan Pilihan : 9
  
```

Gambar 15

Memilih proses transaksi


```

Output - PROGRAM_KASIR_RESTORAN (run) X
run:
~> [RESTORAN NICE] <~
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 3
Masukkan Jumlah Beli : 2
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 9
|===== [ PEMBAYARAN ] =====|
Total : Rp.40000.0
Bayar : Rp.50000

```

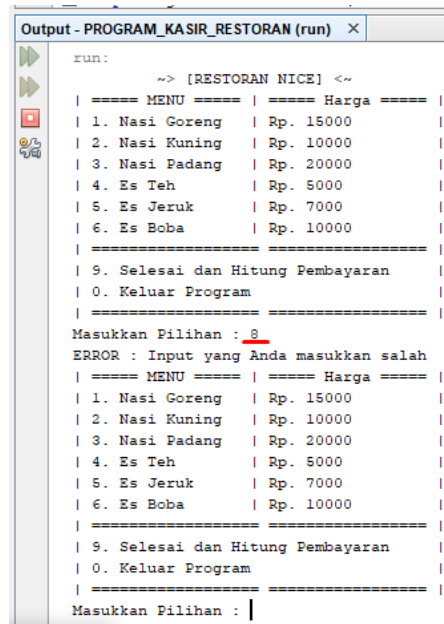
Gambar 16
Melakukan pembayaran

```

Output - PROGRAM_KASIR_RESTORAN (run) X
run:
~> [RESTORAN NICE] <~
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 3
Masukkan Jumlah Beli : 2
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 9
|===== [ PEMBAYARAN ] =====|
Total : Rp.40000.0
Bayar : Rp.50000
Kembalian : Rp.10000.0
|===== [ TERIMA KASIH ] =====|
BUILD SUCCESSFUL (total time: 2 minutes 28 seconds)

```

Gambar 17
Output selesai transaksi



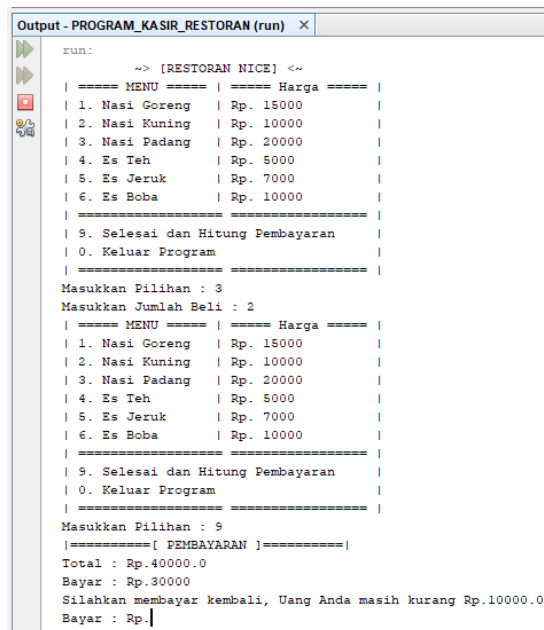
```

Output - PROGRAM_KASIR_RESTORAN (run) X
run:
~> [RESTORAN NICE] <~
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 8
ERROR : Input yang Anda masukkan salah
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : |

```

Gambar 18

Memilih data yang tidak ada dalam sistem



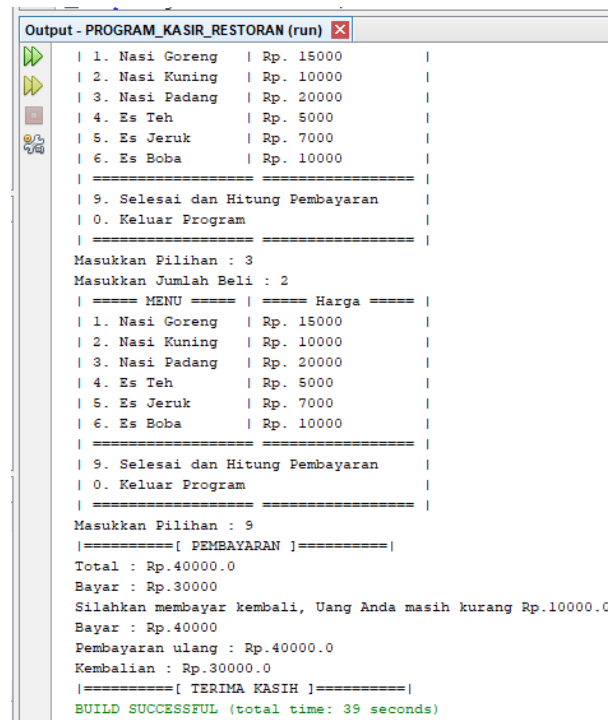
```

Output - PROGRAM_KASIR_RESTORAN (run) X
run:
~> [RESTORAN NICE] <~
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 3
Masukkan Jumlah Beli : 2
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 9
|===== [ PEMBAYARAN ] =====|
Total : Rp.40000.0
Bayar : Rp.30000
Silahkan membayar kembali, Uang Anda masih kurang Rp.10000.0
Bayar : Rp.|

```

Gambar 19

Menginputkan jumlah yang kurang



```

Output - PROGRAM_KASIR_RESTORAN (run)
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 3
Masukkan Jumlah Beli : 2
| ===== MENU ===== | ===== Harga ===== |
| 1. Nasi Goreng | Rp. 15000 |
| 2. Nasi Kuning | Rp. 10000 |
| 3. Nasi Padang | Rp. 20000 |
| 4. Es Teh | Rp. 5000 |
| 5. Es Jeruk | Rp. 7000 |
| 6. Es Boba | Rp. 10000 |
| ===== |
| 9. Selesai dan Hitung Pembayaran |
| 0. Keluar Program |
| ===== |
Masukkan Pilihan : 9
|===== [ PEMBAYARAN ] =====|
Total : Rp.40000.0
Bayar : Rp.30000
Silahkan membayar kembali, Uang Anda masih kurang Rp.10000.0
Bayar : Rp.40000
Pembayaran ulang : Rp.40000.0
Kembalian : Rp.30000.0
|===== [ TERIMA KASIH ] =====|
BUILD SUCCESSFUL (total time: 39 seconds)

```

Gambar 20

Melakukan pembayaran ulang

4.1.2 Pengorepasian Aplikasi

Pengoperasian aplikasi dilakukan untuk mengetahui apakah aplikasi yang dirancang sudah sesuai dengan rancangan yang dibuat atau tidak. Aplikasi dioperasikan dengan menggunakan perintah run untuk menjalankan program pada netbeans, adapun tahapan pengujian yaitu sebagai berikut :

1. Klik kanan pada class kasir_menu, lalu pilih run
2. Setelah itu akan muncul kotak output yang menampilkan beberapa pilihan

3. Ketikkan nomor pilihan yang ingin dipilih sesuai yang tertera dalam program
4. Masukkan jumlah beli yang diinginkan
5. Maka akan muncul menu pilihan lagi, pilih no 9 untuk mengeksekusi bagian transaksi
6. Maka akan muncul total yang tertera, lalu masukkan jumlah pembayaran
7. Selanjutnya akan muncul kembalian dari total uang yang diinputkan dan program telah selesai
8. Apabila jumlah uang yang diinputkan kurang maka program akan menampilkan dialog untuk menambahkan uang

4.1.3 Pengujian Aplikasi

Setelah dilakukan pengujian aplikasi pada virtual box, maka program kasir restoran dapat dijalankan dengan baik tanpa terjadi error.

4.2 Pembahasan

Dari hasil uji coba program dan pengamatan pada proses merancang sistem yang dibuat penulis, maka penulis berpendapat bahwa program kasir sederhana ini dapat dilakukan agar mengurangi kekeliruan pada proses transaksi yang melibatkan uang dalam jumlah kecil maupun besar. Sedangkan dari hasil perancangan sistem ini

penulis berharap ada yang dapat menerapkan sistem ini ke dalam aplikasi sesungguhnya dan dapat digunakan selayaknya dalam kehidupan sehari-hari.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah menyelesaikan tugas ini dan berdasarkan hasil uji coba program yang penulis lakukan, maka dapat diambil beberapa kesimpulan antara lain :

- 1) Aplikasi kasir sederhana memberi manfaat lebih untuk sebuah restoran, warung makan, *café*, toko yang difungsikan dibagian kasir untuk mengolah penjualan dan keuangan agar lebih teratur dan efisien
- 2) Program aplikasi kasir ini sangat berguna untuk digunakan di bagian penjualan karena pada saat proses transaksi dan akumulasi harga dapat lebih akurat, teliti dan terpercaya karena sistem ini dilengkapi dengan kalkulasi secara terkomputerisasi

5.2 Saran

Beberapa saran yang dapat digunakan untuk pengembangan dari aplikasi kasir ini adalah sebagai berikut :

- ii) Untuk pengembangan aplikasi ini peneliti selanjutnya dapat menggunakan penelitian ini untuk membuat sistem aplikasi kasir lebih sempurna dari sebelumnya.

- iii) Peneliti selanjutnya dapat menambahkan program ini ke dalam sistem operasi dengan konsep installer agar lebih mudah dalam penggunaannya
- iv) Dari hasil penelitian ini dapat dibuatkan draft atau modul penggunaan aplikasi secara detail dan terperinci agar memudahkan penggunanya dalam mengoperasikan aplikasi

DAFTAR PUSTAKA

- Aplikasi, Efektivitas, and Kasir Meng. "Related Papers." *Over The Rim*, 2017, pp. 191–99, doi:10.2307/j.ctt46nrzt.12.
- Ilham Akhsanu Ridlo. "Pedoman Pembuatan Flowchart." *Academia.Edu*, 2017, p. 14, https://www.academia.edu/34767055/Pedoman_Pembuatan_Flowchart.
- Juansyah, Andi. "Pembangunan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System (A-GPS) Dengan Platform Android." *Jurnal Ilmiah Komputer Dan Informatika (KOMPUTA)*, vol. 1, no. 1, 2015, pp. 1–8.
- Kurniawan, Hamid, et al. "Penerapan Metode Waterfall Dalam Perancangan Sistem Informasi Penggajian Pada Smk Bina Karya Karawang." *Jurnal Interkom: Jurnal Publikasi Ilmiah Bidang Teknologi Informasi Dan Komunikasi*, vol. 14, no. 4, 2021, pp. 13–23, doi:10.35969/interkom.v14i4.78.
- Lumbantoruan, Eka Pratiwi, and Paidi Hidayat. *No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析Title*. 2013, pp. 14–27.
- Martasubrata, Muhamad F., and Yudi Priyadi. "Analisis Kesiapan UMKM Dalam Mengadopsi E-SCM Melalui Kolaborasi Technology Acceptance Model Dan Data Flow Diagram Di UMKM Clothing Line Lokal Bandung." *Sosiohumanitas*, vol. 21, no. 2, 2020, pp. 108–15, doi:10.36555/sosiohumanitas.v21i2.1249.
- Ninla Elmawati Falabiba, et al. "済無No Title No Title No Title." *Paper Knowledge . Toward a Media History of Documents*, vol. 5, no. 2, 2014, pp. 40–51.
- Puspitasari, Dwi, et al. "Normalisasi Tabel Pada Basisdata Relasional." *Jurnal Prosiding SENTIA | ISSN: 2085-2347*, vol. 8, no. 1, 2016, pp. 340–45.
- Rachmawati, Rina. "Peranan Bauran Pemasaran (Marketing Mix) Terhadap Peningkatan Penjualan." *Jurnal Kompetensi Teknik*, vol. 2, no. 2, 2011, pp. 143–50.
- Rahmawati, Annisa, et al. "Pembuatan Sistem Informasi Rental Mobil

Dengan Menggunakan Java Dan Mysql.” *Jurnal Teknologi Dan Sistem Komputer*, vol. 3, no. 3, 2015, p. 335, doi:10.14710/jtsiskom.3.3.2015.335-342.

Wellem, Erman. “Generator Melodi Berdasarkan Skala Dan Akord Menggunakan Algoritma Genetika.” *Jurnal Informatika*, 2015.