

APLIKASI BERBASIS WEB UNTUK PENGELOLAAN PENJUALAN KATERING (STUDI KASUS DI WIWIN CATERING, BANDUNG)

Defina Damayanti¹, Renny Sukawati², Irman Hariman³.

^{1,2,3} Universitas Telkom, Bandung

definadmt@student.telkomuniversity.ac.id¹, rennys@tass.telkomuniversity.ac.id²,

irmanhariman@telkomuniversity.ac.id³

Abstrak

Wiwin Catering adalah perusahaan manufaktur yang telah memiliki izin usaha yang bergerak dibidang catering yang bertempat di Permata Kopo Blok CA no. 73 Rt. 08 Rw. 06 Sayati Kecamatan Margahayu, Kabupaten Bandung. Transaksi yang terjadi di *Wiwin Catering* ini adalah transaksi pembelian, pembebanan dan penjualan. Terdapat berbagai masalah dalam proses pencatatan transaksinya, terutama pencatatan penjualan secara tunai dengan membayar uang muka terlebih dahulu sebesar 40%, dan sisanya 60% akan di bayar ketika catering telah diterima oleh pelanggan, juga pencatatan pembebanan yang dikeluarkan setiap bulannya, serta laporan akuntansi. Untuk mengatasi permasalahan tersebut dibangun sebuah aplikasi berbasis web yang dapat mengelola pencatatan penjualan *cash* atau uang muka penjualan, transaksi pembebanan dan menghasilkan laporan akuntansi seperti jurnal umum, buku besar dan laporan laba rugi. Aplikasi berbasis web ini dibangun dengan metode berorientasi objek dan pengembangan model *Software Development Life Cycle* (SDLC) serta menggunakan Bahasa pemrograman PHP, *Framework CodeIgniter*.

Kata Kunci— Penjualan, Katering, *Framework CodeIgniter*, Aplikasi Berbasis Web.

Abstract

Wiwin Catering is a manufacturing company that has a business license engaged in catering located at Permata Kopo Blok CA no. 73 Rt. 08 Rw. 06 Sayati Margahayu Subdistrict, Bandung Regency. Transactions that occur in Wiwin Catering is a transaction of purchase, loading and selling. There are various problems in the process of recording transactions, especially the recording of sales in cash by paying a down payment in advance of 40%, and the remaining 60% will be paid when the catering has been received by the customer, as well as the recording of charges issued each month, as well as accounting reports. To solve the problem, a web-based application is built that can manage the sale of cash or down payment sales, loading transactions and generate accounting reports such as general journals, general ledgers and profit and loss statements. This web-based application is built with object-oriented methods and develops a Software Development Life Cycle (SDLC) model and uses php programming language, CodeIgniter Framework.

Keywords— Sales, Catering, *CodeIgniter Framework*, Web-Based Applications.

I. PENDAHULUAN

Dengan globalisasi dewasa ini, menyebabkan persaingan antar perusahaan semakin banyak. Salah satunya yaitu di bidang usaha makanan. Penyedia jasa catering makin bertambah dari waktu ke waktu. Peningkatan usaha catering memang sangat beralasan selain meningkatnya jumlah penduduk yang juga meningkatkan jumlah orang yang membutuhkan layanan jasa ini. Selain peningkatan penduduk, dengan perubahan gaya hidup penduduk kota besar yang ingin lebih praktis maka makin bertambahlah konsumen bisnis catering ini. Perubahan bisnis catering dari waktu ke waktu tidak hanya ditandai dengan peningkatan jumlah pemilik usaha, bahkan perubahan dalam variasi makanan, variasi

desain penataan, dan fasilitas lainnya juga mulai terlihat dahulu catering hanya memasak makanan umum yang banyak disukai dalam bentuk prasmanan saja, semakin lama berkembang seperti adanya fasilitas gubuk, variasi makanan dan ada pula catering yang hanya menyediakan menu khusus seperti catering vegetarian dan sebagainya. [1].

Wiwin Catering adalah perusahaan manufaktur yang telah memiliki izin usaha yang bergerak dibidang catering boxes dan prasmanan yang bertempat di Permata Kopo Blok CA no. 73 Rt. 08 Rw. 06 Sayati Kecamatan Margahayu, Kabupaten Bandung. *Wiwin Catering* ini berdiri sejak tahun 2016 dan

memiliki 4 pegawai di tahun 2020. Rata-rata penjualan perbulan pada katering ini kurang lebih Rp. 25.000.000. transaksi yang terjadi di katering ini pembelian, pembebanan dan penjualan.

Transaksi penjualan di *Wiwini Catering* ini dapat dilakukan secara tunai, dengan kebijakan awal untuk sistem pembayarannya dapat berupa *Cash* membayar uang muka terlebih dahulu sebesar 40%, dan sisanya 60% akan di bayar ketika katering telah di terima oleh pelanggan.

Kendala dalam proses oprasional di *Wiwini Catering* adalah pada saat pencatatan transaksi dan laporan. Karena pada saat pencatatan transaksi penjualan dan pembebanan *wiwini catering* masih menggunakan pencatatan manual didalam buku, berkas-berkasnyapun masih berceceran juga sering kali hilang. Bahkan terkadang transaksi tidak dicatat sama sekali, akibatnya pemilik tidak memiliki *history* transaksi dan laporan yang akurat di setiap bulannya.

Maka dari itu, *Wiwini Catering* ini ingin memperbaiki permasalahan tersebut, dengan memiliki pencatatan atas transaksi yang terjadi dan membuat laporan di setiap bulannya. Dari permasalahan di atas, maka dapat diusulkan aplikasi yang dapat mengatasi masalah-masalah tersebut, seperti mampu mencatat transaksi penjualan dan pembebanan.

TABEL 1. STUDY LITERATURE

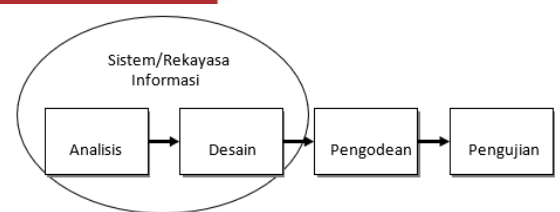
Judul Proyek Akhir	Aplikasi Penjualan Dan Pengelolaan Piutang Berbasis Web (Studi Kasus di Konveksi Al-Fath Inc, Bandung) [2]
Penulis	Diesta Kalyana Putri Budiman
Fungsionalitas	<ol style="list-style-type: none"> 1. Transaksi Penjualan Tunai dan Kredit 2. Pencatatan piutang 3. Penilaian resiko piutang 4. Jurnal umum 5. Buku Besar 6. Laporan laba rugi
Metode Akuntansi yang digunakan	Metode pencatatan persediaan menggunakan metode perpetual
Mengidentifikasi kekurangan dan kelebihan	Kekurangan : Aplikasi ini tidak menghasilkan laporan penjualan
Keterkaitan dengan tema	Proyek ini mengambil tema yang sama yaitu transaksi penjualan dengan uang muka terlebih dahulu

Perbedaan	Aplikasi yang dibuat tidak membuat pencatatan piutang dan penilaian resiko piutang
-----------	--

II. METODE PENELITIAN

A. Metode Pengembangan Perangkat Lunak

Pada proyek akhir ini, metodologi yang digunakan adalah *Software Development Life Cycle* (SDLC) dengan model *Waterfall*. Model SDLC air terjun (*waterfall*) sering disebut juga model sekuensial linear (*sequential linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (*support*) [3].



Gambar 1 Metode Waterfall

Bentuk tahapan-tahapan yang terdapat pada model waterfall yaitu sebagai berikut :

a) Analisis

Proses pengumpulan kebutuhan dilakukan secara intensif untuk memspezifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user [3].

1. Wawancara

Pengumpulan data telah dilakukan pada tanggal 16 September 2020. Metode ini dilakukan dengan berdialog langsung dengan *interview* dengan Ibu Wiwin Winori selaku pemilik dari *Wiwini Catering* yang bertujuan agar dapat menggali kebutuhan *user* secara lebih jelas.

2. Observasi

Observasi yang dilakukan setelah melakukan wawancara pada tanggal 16 September 2020 secara langsung dengan meminta informasi mengenai kegiatan transaksi di perusahaan seperti transaksi khusus penjualan.

b) Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean [3]. Dengan melakukan pemodelan aplikasi berbasis objek *oriented* dengan *Unified Modelling Language* (UML) yang

terdiri dari *case diagram*, *activity diagram*, *sequence diagram* dan *class diagram*. Dan untuk mendesain basis data menggunakan *Entity Relationship Diagram (ERD)*.

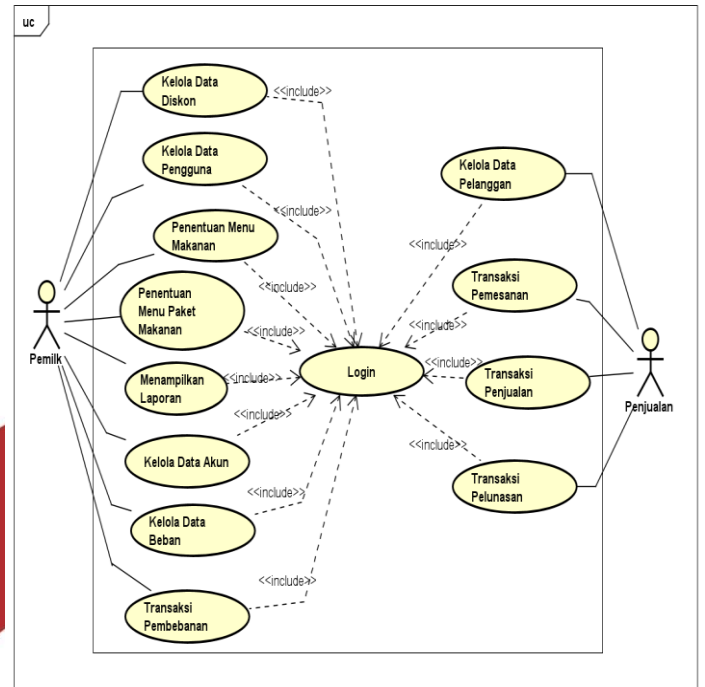
c) Pengkodean

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah di buat pada tahap desain [3]. Perancangan sistem yang akan dibuat menggunakan bahasa pemrograman *Hypertext Preprocessor (PHP)*, *CodeIgniter (CI)*, *Hypertext Macro Language (HTML)*, dan basis data menggunakan *MySQL*.

d) Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan [3]. Penggunaan *Black Box* dalam pengujian aplikasi akan memberikan penjelasan tentang kesesuaian harapan dalam pembuatan program.

e) Pendukung (*Support*) atau Pemeliharaan (*Maintenance*).



Gambar 0-1 Usecase Diagram

B. Metodologi yang Digunakan

Metodologi yang penulis gunakan dalam menyusun proyek akhir ini adalah Metodologi Berorientasi Objek, dikarenakan konsep pemrograman yang penulis gunakan adalah pemrograman berorientasi objek. Dan juga semua data dan fungsi di dalam proyek akhir ini dibungkus dalam kelas-kelas atau objek-objek.

III. HASIL

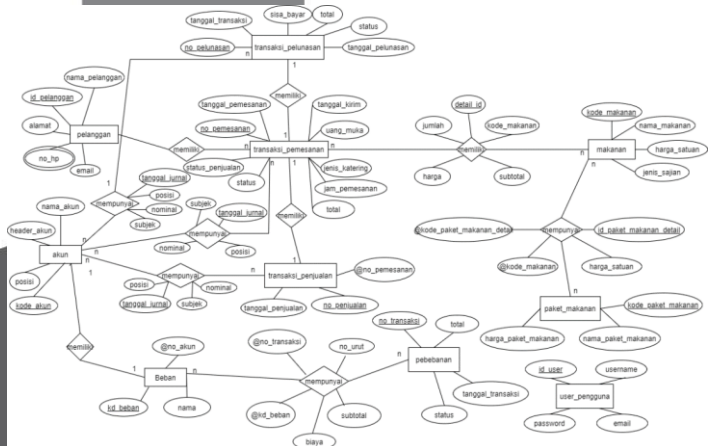
Pada bagian ini menjelaskan tentang sistem yang di gunakan untuk membuat perancangan aplikasi yang akan di terapkan di Hoax Cuiiiih Clothing.

A. Usecase Diagram

Pemodelan kebutuhan digambarkan berbentuk diagram use case sebagai berikut:

B. Entity Relationship Diagram

Penggambaran *Entity Relationship Diagram (ERDiagram)* sebagai berikut :



Gambar 0-1 Entity Relationship Diagram (ERD)

IV. IMPLEMENTASI DAN PENGUJIAN

Implementasi merupakan gambaran sistem yang dibangun. Terdapat dua implementasi yaitu, implementasi *database* dan implementasi *user interface*. Terdapat dua implementasi pada sistem aplikasi ini, yaitu implementasi basis data dan implementasi proses. Dan hasil dari perancangan sistem yang dibuat dengan melakukan coding menggunakan bahasa php,

html, css, dan javascript. Serta menggunakan framework codeigniter dan untuk basis data menggunakan MySQL. Pengujian ini menggunakan black box testing dan pengujian manual. Pengujian manual dibuat untuk melihat kesesuaian data yang diisi manual dengan data yang diisikan pada aplikasi.

A. Implementasi Data

Berikut ini adalah database yang digunakan dalam membangun aplikasi ini dengan nama catering.sql. Adapun tampilan dari database tersebut sebagai berikut :

Table	Action	Rows	Type	Collation	Size	Overhead
akun	Browse Structure Search Insert Empty Drop	11	InnoDB	latin1_swedish_ci	16 K B	-
beban	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K B	-
detail_pemesanan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K B	-
diskon	Browse Structure Search Insert Empty Drop	2	InnoDB	utf8mb4_general_ci	16 K B	-
jurnal	Browse Structure Search Insert Empty Drop	23	InnoDB	latin1_swedish_ci	16 K B	-
makanan	Browse Structure Search Insert Empty Drop	26	InnoDB	latin1_swedish_ci	16 K B	-
paket_makanan	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 K B	-
paket_makanan_detail	Browse Structure Search Insert Empty Drop	23	InnoDB	utf8mb4_general_ci	16 K B	-
pelanggan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K B	-
pembelian	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K B	-
transaksi_pelunasan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K B	-
transaksi_pemesanan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K B	-
transaksi_penjualan	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	16 K B	-
user_pengguna	Browse Structure Search Insert Empty Drop	2	InnoDB	latin1_swedish_ci	16 K B	-

Gambar 0-1 Implementasi Database MySQL

Berikut ini merupakan proses yang dibuat berdasarkan perancangan antarmuka yang telah dibuat sebelumnya.

a. Master Data Akun

Berikut ini merupakan implementasi proses menambah, dan menampilkan akun. Pada implementasi tampilan semua master data dan memiliki tombol Tambah Akun yang berfungsi untuk menambahkan data akun seperti berikut :

The 'Tambah Akun' form contains the following fields:

- No. Akun: Masukkan No. Akun
- Nama Akun: Masukkan Nama Akun
- Header Akun: Masukkan Header Akun
- Posisi Normal: -- Pilih --

Buttons: Simpan, Batal

Gambar 0-2 Implementasi Proses Menambah Data Akun

Berikut merupakan tampilan daftar akun yang sudah dimasukkan. Tampilan berikut ini berfungsi untuk memberikan informasi akun yang dimiliki oleh perusahaan:

No	No. Akun	Nama Akun	Header Akun	Posisi Normal	Aksi
1	1.1.1	Kas	1	D	✓
2	1.1.2	Piutang Dagang	1	D	✓
3	1.1.3	Persediaan	1	D	✓
4	2.1.1	Utang Muka Penjualan	2	K	✓
5	3.1.1	Mobil	3	K	✓
6	4.1.1	Penjualan	4	K	✓
7	5.1.1	Harga Pokok Penjualan	5	D	✓

Gambar 0-3 Implementasi Proses Aplikasi Menampilkan Data Akun

Tombol ubah pada tampilan Data Akun berfungsi untuk mengubah data yang telah dimasukkan sebelumnya. Ketika pengguna

mengklik tombol ubah, maka akan muncul tampilan seperti berikut ini:

The 'Ubah Akun' form contains the following fields:

- No. Akun: 1.1.1
- Nama Akun: Kas
- Header Akun: 1
- Posisi Normal: -- Pilih --

Buttons: Simpan, Batal

Gambar 0-4 Implementasi Proses Aplikasi Mengubah Data Akun

b. Master Data Makanan

Berikut ini merupakan implementasi proses menambah, menampilkan dan mengubah data makanan. Pada implementasi tampilan semua master data memiliki tombol tambah data yang berfungsi untuk menambahkan data baru seperti berikut ini:

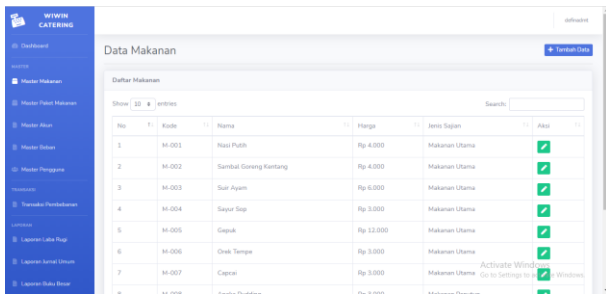
The 'Tambah Makanan' form contains the following fields:

- Kode: M-001
- Nama: Masukkan Nama
- Harga: Masukkan Harga
- Jenis Sajian: -- Pilih --

Buttons: Simpan, Batal

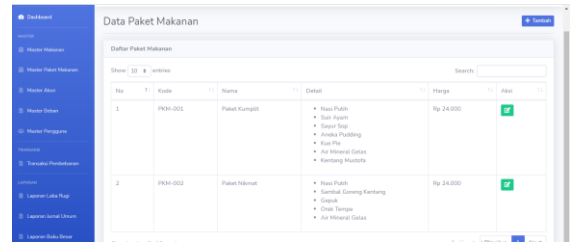
Gambar 0-5 Implementasi Proses Aplikasi Menambah Data Makanan

Berikut ini merupakan tampilan data makanan yang sudah dimasukkan. Tampilan berikut ini berfungsi untuk memberi informasi mengenai detail makanan yang akan dijual, yaitu: nama makanan, harga, jenis sajian:



Gambar 0-6 Implementasi Proses Aplikasi Menampilkan Data Makanan

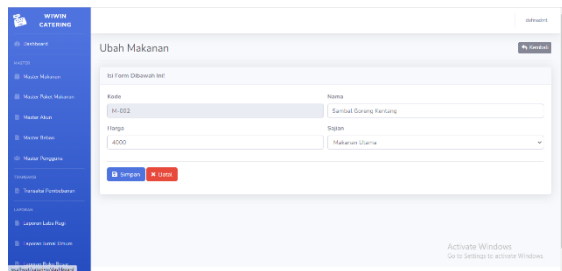
informasi mengenai detail paket makanan yang akan dijual, yaitu: kode paket, nama paket, detail paket dan harga paket:



Gambar 0-9 Implementasi Proses Aplikasi Menampilkan Data Paket Makanan

Tombol ubah pada tampilan data paket makanan berfungsi untuk mengubah data yang telah dimasukkan sebelumnya, namun data yang dapat diubah nama paket makanan dan item makanan. Ketika pengguna mengklik tombol ubah, maka akan muncul tampilan sebagai berikut:

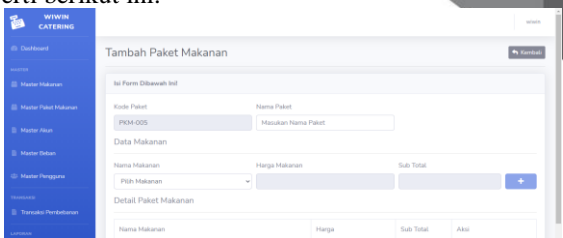
Tombol ubah pada tampilan data makanan berfungsi untuk mengubah data yang telah dimasukkan sebelumnya, namun data yang dapat diubah nama makanan, jenis hidangan, dan harga. Ketika pengguna mengklik tombol ubah, maka akan muncul tampilan sebagai berikut:



Gambar 0-7 Implementasi Proses Aplikasi Menambah Data Makanan

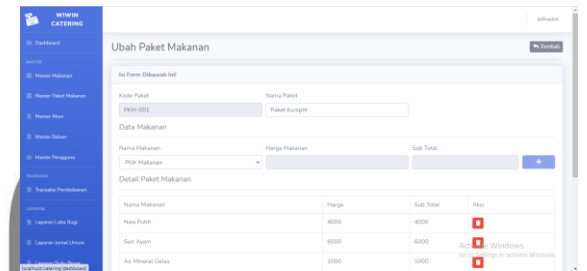
c. Master Data Paket Makanan

Berikut ini merupakan implementasi proses menambah, menampilkan dan mengubah data paket makanan. Pada implementasi tampilan semua master data memiliki tombol tambah data yang berfungsi untuk menambahkan data baru seperti berikut ini:



Gambar 0-8 Implementasi Proses Aplikasi Menambah Data Paket Makanan

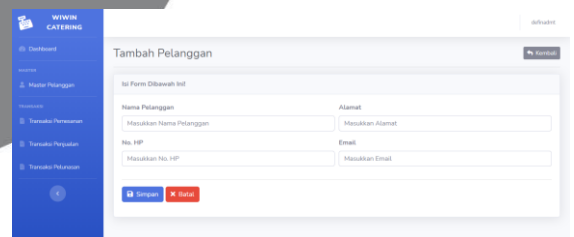
Berikut ini merupakan tampilan data makanan yang sudah dimasukkan. Tampilan berikut ini berfungsi untuk memberi



Gambar 0-10 Implementasi Proses Aplikasi Mengubah Data Paket Makanan

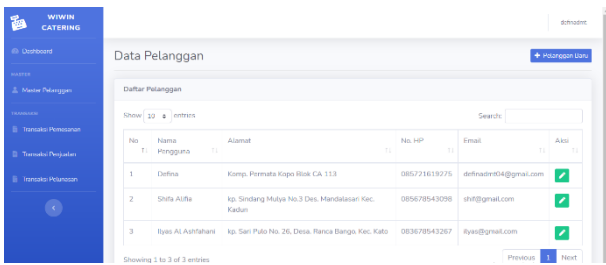
d. Master Data Pelanggan

Berikut ini merupakan implementasi proses menambah, menampilkan dan mengubah data pelanggan. Pada implementasi tampilan semua master data memiliki tombol tambah data yang berfungsi untuk menambahkan data baru sebagai berikut:

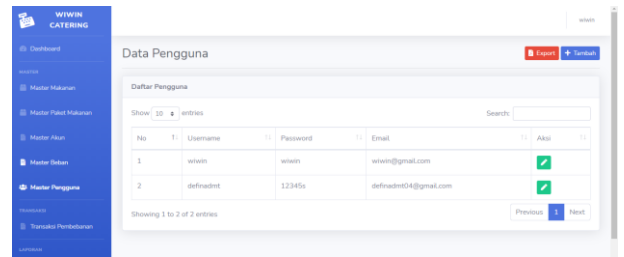


Gambar 0-11 Implementasi Proses Aplikasi Menambah Data Pelanggan

Berikut di bawah ini merupakan tampilan daftar data pelanggan yang sudah dimasukkan. Tampilan ini berfungsi untuk memberi informasi mengenai pelanggan yaitu: nama pelanggan, alamat, no. HP, dan email.



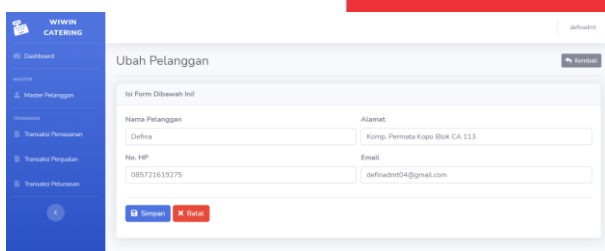
Gambar 0-12 Implementasi Proses Aplikasi Menampilkan Data Pelanggan



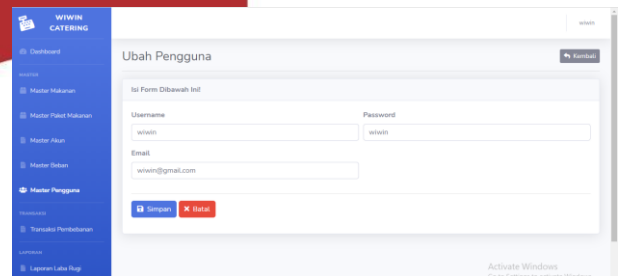
Gambar 0-15 Implementasi Proses Aplikasi Menampilkan Data Pengguna

Tombol ubah pada tampilan data pelanggan berfungsi untuk mengubah data yang telah dimasukkan sebelumnya, namun data yang dapat diubah hanya data nama pelanggan, alamat, no. HP, dan email. Ketika pengguna mengklik tombol ubah, maka akan muncul tampilan *form* ubah data pelanggan seperti berikut ini:

Tombol ubah pada tampilan data pelanggan berfungsi untuk mengubah data yang telah dimasukkan sebelumnya. Ketika pengguna mengklik tombol ubah, maka akan muncul tampilan *form* ubah data pelanggan seperti berikut ini:



Gambar 0-13 Implementasi Proses Aplikasi Mengubah Data Pelanggan



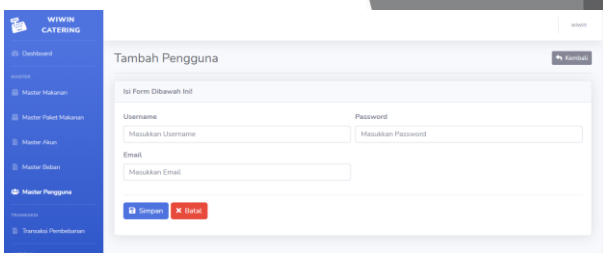
Gambar 0-16 Implementasi Proses Aplikasi Mengubah Data Pengguna

e. Master Data Pengguna

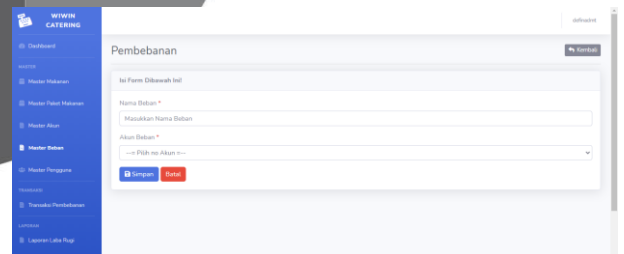
Berikut ini merupakan implementasi proses menambah, menampilkan dan mengubah data pengguna. Pada implementasi tampilan semua master data memiliki tombol tambah data yang berfungsi untuk menambahkan data baru sebagai berikut:

f. Master Data Beban

Berikut ini merupakan implementasi proses menambah, menampilkan dan mengubah data beban. Pada implementasi tampilan semua master data memiliki tombol tambah data yang berfungsi untuk menambahkan data baru sebagai berikut:



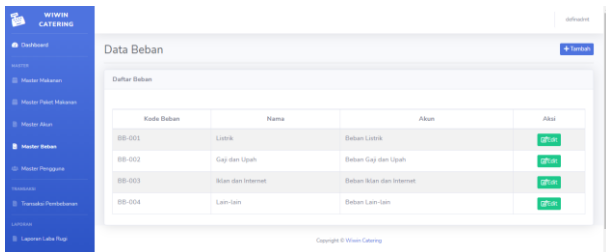
Gambar 0-14 Implementasi Proses Aplikasi Menambah Data Pengguna



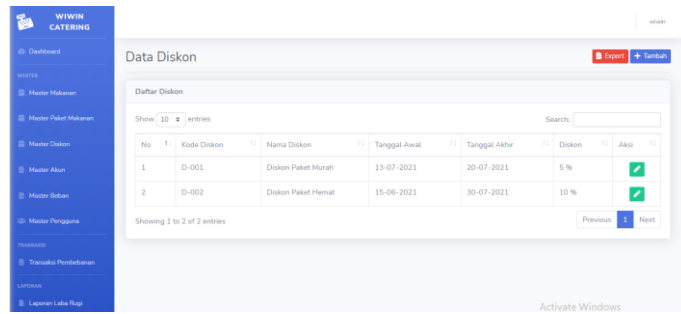
Gambar 0-17 Implementasi Proses Aplikasi Menambah Data Beban

Berikut di bawah ini merupakan tampilan daftar data pengguna yang sudah dimasukkan. Tampilan ini berfungsi untuk memberi informasi mengenai pengguna yaitu: username, password dan email.

Berikut di bawah ini merupakan tampilan daftar data beban yang sudah dimasukkan. Tampilan ini berfungsi untuk memberi informasi mengenai pengguna yaitu: kode beban, nama beban, jenis, dan akun



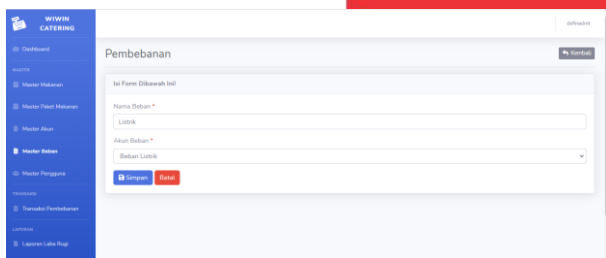
Gambar 0-18 Implementasi Proses Aplikasi Menampilkan Data Beban



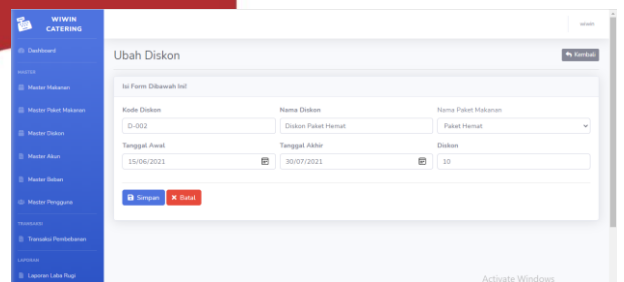
Gambar 0-21 Implementasi Proses Aplikasi Menampilkan Data Diskon

Tombol ubah/edit pada tampilan data pelanggan berfungsi untuk mengubah data yang telah dimasukkan sebelumnya. Ketika pengguna mengklik tombol ubah, maka akan muncul tampilan form ubah data beban seperti berikut ini:

Tombol ubah/edit pada tampilan data pelanggan berfungsi untuk mengubah data yang telah dimasukkan sebelumnya. Ketika pengguna mengklik tombol ubah, maka akan muncul tampilan form ubah data diskon seperti berikut ini:



Gambar 0-19 Implementasi Proses Aplikasi Mengubah Data Beban



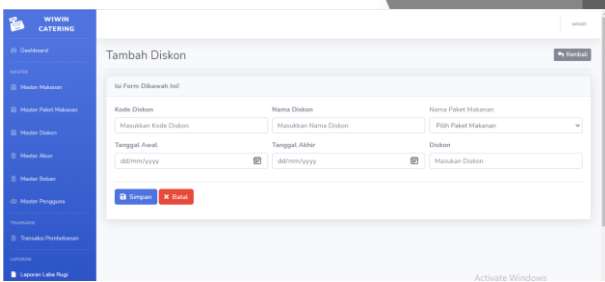
Gambar 0-22 Implementasi Proses Aplikasi Mengubah Data Diskon

g. Master Diskon

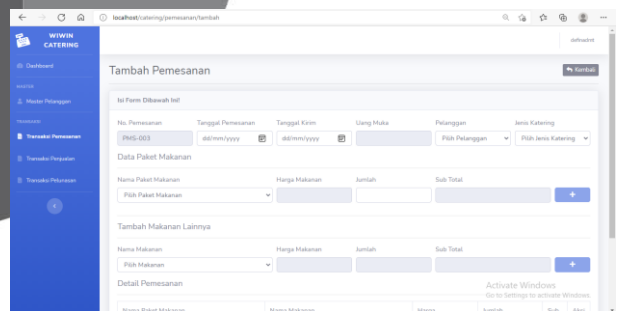
Berikut ini merupakan implementasi proses menambah, menampilkan dan mengubah data diskon. Pada implementasi tampilan semua master data memiliki tombol tambah data yang berfungsi untuk menambahkan data baru sebagai berikut:

h. Transaksi Pemesanan

Berikut ini merupakan implementasi proses menambah dan menampilkan transaksi pemesanan. Pada implementasi tampilan transaksi pemesanan memiliki form untuk menambahkan data baru seperti berikut:



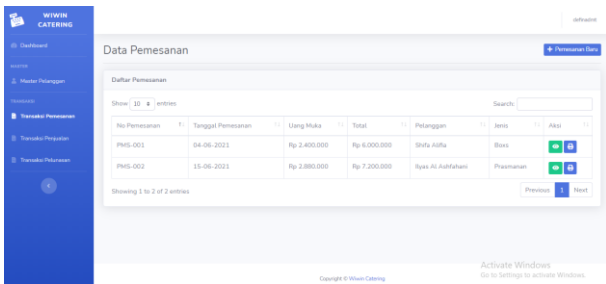
Gambar 0-20 Implementasi Proses Aplikasi Menambah Data Diskon



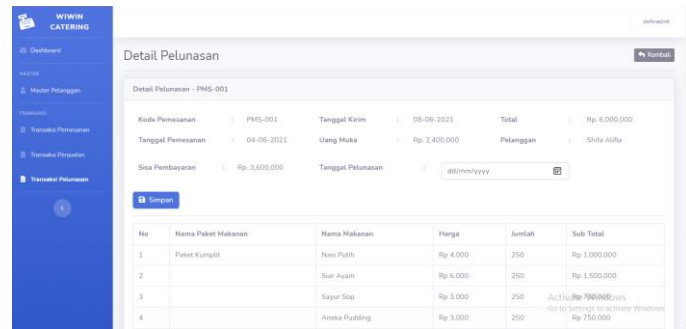
Gambar 0-23 Implementasi Proses Aplikasi Menambah Data Transaksi Pemesanan

Berikut di bawah ini merupakan tampilan daftar data beban yang sudah dimasukkan. Tampilan ini berfungsi untuk memberi informasi mengenai pengguna yaitu: kode diskon, nama diskon, tanggal awal, tanggal akhir dan diskon.

Berikut di bawah ini merupakan tampilan daftar data pembelian yang sudah dimasukkan. Tampilan ini berfungsi untuk memberi informasi mengenai pemesanan yaitu: no pemesanan, tanggal pemesanan, uang muka, total, nama pelanggan, dan jenis catering.



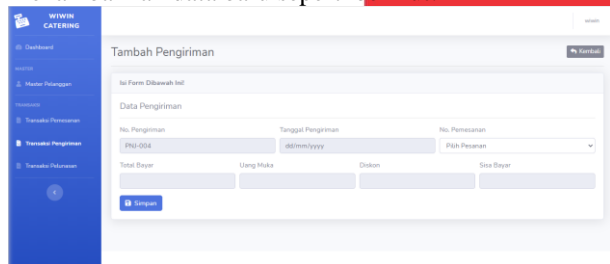
Gambar 0-24 Implementasi Proses Aplikasi Menampilkan Data Transaksi Pemesanan



Gambar 0-27 Implementasi Proses Aplikasi Menambah Data Transaksi Pelunasan

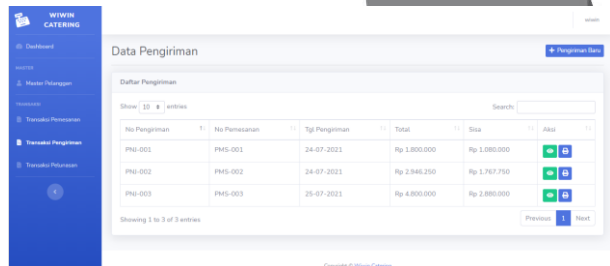
i. Transaksi Pengiriman

Berikut ini merupakan implementasi proses menambah dan menampilkan transaksi penjualan. Pada implementasi tampilan transaksi pembebanan memiliki form untuk menambahkan data baru seperti berikut:



Gambar 0-25 Implementasi Proses Aplikasi Menambah Data Transaksi Pengiriman

Di bawah ini merupakan tampilan Data Transaksi Penjualan. Tampilan berikut ini berfungsi untuk memberi informasi mengenai transaksi yang terjadi yaitu nomor penjualan, nomor pemesanan, tanggal penjualan, total, dan sisa:

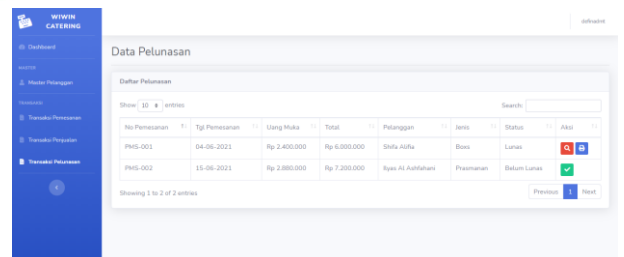


Gambar 0-26 Implementasi Proses Aplikasi Menampilkan Data Transaksi Pengiriman

j. Transaksi Pelunasan

Berikut ini merupakan implementasi proses menambah dan menampilkan transaksi pelunasan. Transaksi pelunasan diambil dari seluruh transaksi penjualan. Pada implementasi tampilan transaksi penjualan memiliki form untuk menambahkan data baru seperti berikut:

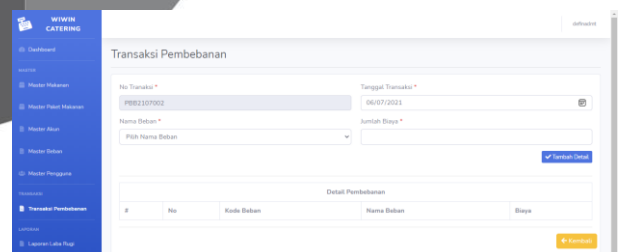
Di bawah ini merupakan tampilan Data Transaksi Pelunasan. Tampilan berikut ini berfungsi untuk memberi informasi mengenai transaksi yang terjadi yaitu nomor pemesanan, tanggal pemesanan, uang muka, total, pelanggan, jenis, dan status :



Gambar 0-28 Implementasi Proses Aplikasi Menampilkan Data Transaksi Pelunasan

k. Transaksi Pembebanan

Berikut ini merupakan implementasi proses menambah dan menampilkan transaksi pembebanan. Pada implementasi tampilan transaksi pembebanan memiliki form untuk menambahkan data baru seperti berikut:



Gambar 0-29 Implementasi Proses Aplikasi Menambah Data Transaksi Pembebanan

Di bawah ini merupakan tampilan Data Transaksi Pembebanan. Tampilan berikut ini berfungsi untuk memberi informasi mengenai transaksi yang terjadi yaitu nomor transaksi, tanggal dan total :

#	No	No Transaksi	Tanggal	Total	Status	Aksi
1		PBE2107001	29-06-2021	Rp. 825.000,00	Salvato	[icon]
Total				Rp. 825.000,00		

Gambar 0-30 Implementasi Proses Aplikasi Menampilkan Data Transaksi Pembebanan

Tanggal	Keterangan	Ref	Debit	Kredit	Debit	Kredit
	Saldo Awal				Rp. 0,00	
04-06-2021	Uang Muka Penjualan	2.1.1		Rp. 2.400.000,00		Rp. 2.400.000,00
15-06-2021	Uang Muka Penjualan	2.1.1		Rp. 2.880.000,00		Rp. 5.280.000,00
08-06-2021	Uang Muka Penjualan	2.1.1	Rp. 2.400.000,00			Rp. 2.880.000,00
19-06-2021	Uang Muka Penjualan	2.1.1	Rp. 2.880.000,00			Rp. 0,00
	Saldo Akhir				Rp. 0,00	

Gambar 0-34 Implementasi Proses Aplikasi Menampilkan Buku Besar

l. Jurnal Umum

Berikut ini merupakan implementasi proses menampilkan data jurnal umum. Pada implementasi tampilan jurnal umum berfungsi untuk menampilkan proses pencatatan transaksi yang sudah terjadi dari transaksi pemesanan, transaksi penjualan, transaksi pelunasan dan transaksi pembebanan:

No transaksi	Tanggal	Keterangan	Ref	Debit	Kredit
PMS-001	2021-06-04	Putang Dagang	1.1.2	Rp. 3.600.000,00	
PMS-001	2021-06-04	Kas	1.1.1		Rp. 2.400.000,00
PMS-001	2021-06-04	Penjualan	4.1.1		Rp. 3.000.000,00
PMS-001	2021-06-04	Uang Muka Penjualan	2.1.1		Rp. 2.400.000,00

Gambar 0-31 Implementasi Proses Aplikasi Menampilkan Jurnal Umum dari Transaksi Pemesanan

PMS-001	2021-06-08	Uang Muka Penjualan	2.1.1		Rp. 2.400.000,00
PMS-001	2021-06-08	Penjualan	4.1.1		Rp. 2.400.000,00

Gambar 0-32 Implementasi Proses Aplikasi Menampilkan Jurnal Umum dari Transaksi Pengiriman

PLN-001	2021-06-08	Kas	1.1.1	Rp. 3.600.000,00	
PLN-001	2021-06-08	Putang Dagang	1.1.2		Rp. 3.600.000,00

Gambar 0-33 Implementasi Proses Aplikasi Menampilkan Jurnal Umum dari Transaksi Pelunasan

m. Buku Besar

Berikut ini merupakan implementasi proses menampilkan buku besar. Pada implementasi tampilan buku besar berfungsi untuk menampilkan proses pencatatan transaksi yang sudah terjadi dari transaksi penjualan dan transaksi pelunasan piutang yang kemudian dikelompokkan berdasarkan kode akun dan nama akunnya masing-masing.

j. Laporan Laba Rugi

Berikut ini merupakan implementasi proses menampilkan laporan Laba Rugi. Pada implementasi tampilan laporan laba rugi berfungsi untuk mengetahui laba atau rugi disetiap periode bulannya.

Pendapatan	Saldo
Penjualan	Rp. 13.200.000,00
Perjualan	Rp. 5.280.000,00
Merge Piutang Penjualan	Rp. 7.920.000,00
Labanya	Rp. 7.920.000,00
Beban Operasional	Rp. 600.000,00
Beban Gaji dan Ujrah	Rp. 150.000,00
Beban Listrik	Rp. 75.000,00
Beban Kiri dan Internet	Rp. 825.000,00
Total Beban	Rp. 7.995.000,00
Labanya	Rp. 7.995.000,00

Gambar 0-35 Implementasi Proses Aplikasi Menampilkan Laporan Laba Rugi

IV. KESIMPULAN

Sehubungan dengan pembangunan aplikasi dan pengujian yang telah dilakukan pada proyek akhir ini, maka dapat disimpulkan bahwa: Aplikasi dalam Proyek Akhir mampu menangani pencatatan penjualan secara tunai dengan uang muka terlebih dahulu mampu menangani pencatatan pembebanan, mampu menampilkan Jurnal Umum dan Buku Besar, mampu menghasilkan Laporan Laba Rugi.

IV. REFERENSI

[1] W. Ayodya, Kursus Singkat Usaha Aneka Katering Laris Manis, Jakarta: Gramedia, 2008.
 [2] D. K. Budiman, Aplikasi Penjualan dan Pengelolaan Piutang Berbasis Web, Bandung: Telkom University, 2020.
 [3] R. Sukanto and M. Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek, Bandung: Informatika Bandung, 2013.

