

RANCANG BANGUN APLIKASI PEMESANAN DAN TRANSAKSI (STUDI KASUS: FST UNIVERSITAS TRILOGI)

Silvester Dian Handy Permana¹⁾, Muhammad Rizal Fauzi²⁾

¹ Teknik Informatika, ² Sistem Informasi, Fakultas Telematika, Universitas Trilogi
Jalan Kampus Trilogi / STEKPI No. 1, Kalibata, Jakarta Selatan 12760
Telp : (021) 7980011, Fax : (021) 7981352
E-mail : handy@universitas-trilogi.ac.id¹⁾

Abstract

FST Corner (Food Science Technology) is a café corner, located on Universitas Trilogi, South Jakarta. In conducting its business, FST Corner Cafe services still use paper-based system. It is difficult for employees serving customers in the cafe, especially at rush hour. To serve the guests, waiters take orders in the receipt. The payment system is still manually caused need long time for calculation. Cafe manager often in hurry for calculating payments due to its numerous customers in the café. Not a few customers who did not buy because of long waiting time. This reduces the income that should be received by this Café. Therefore, it is necessary to build a system that can assist in the calculation and recording of work orders. Data collection methods used were observation, interview and literature study. While the system development method used is SDLC (System Development Live Cycle) Waterfall Model. This application has a feature to see a list of orders, ordering menu to customers digitally, calculates bill payments, as well as a cafe manager can help in reviewing the financial statements.

Abstrak

FST Corner (Food Science Technology) merupakan sebuah sudut cafe yang berlokasi di Kampus Universitas Trilogi, Jakarta Selatan. Dalam menjalankan usahanya, pelayanan Cafe FST Corner masih menggunakan sistem paper-based. Hal tersebut menyulitkan karyawan dalam melayani tamu cafe, terutama pada jam-jam sibuk. Untuk melayani tamu yang datang, pelayan mencatat pesanan dalam struk / nota. Sistem pembayarannya masih manual yang menyebabkan perhitungannya cukup memakan waktu. Pengelola café sering kerepotan dalam mencatat dan menghitung pembayaran karena banyaknya pelanggan di Café ini. Tak sedikit juga pelanggan yang tidak jadi membeli karena waktu menunggu lama. Hal ini tentu saja mengurangi pendapatan yang harusnya diterima oleh Café ini. Oleh karena itu, perlu dibangun sistem yang dapat membantu pekerjaan dalam perhitungan dan pencatatan pesanan. Metode pengumpulan data yang digunakan adalah observasi, wawancara dan studi kepustakaan. Sedangkan metode pengembangan sistem yang digunakan adalah SDLC (System Development Live Cycle) Model Waterfall. Aplikasi ini memiliki fitur untuk melihat daftar pesanan, pemesanan menu untuk pelanggan secara digital, menghitung tagihan pembayaran, serta dapat membantu manager cafe dalam melihat laporan keuangan.

Kata kunci: *Cafe FST Corner, Information System, Aplikasi Pemesanan, Aplikasi Transaksi, SDLC*

1. PENDAHULUAN

Perkembangan teknologi yang semakin canggih menuntut layanan yang semakin mudah, cepat dan praktis. Gaya hidup masyarakat kini pun tidak lepas dari teknologi informasi. Contohnya penggunaan gadget yang menyediakan beberapa fitur yang digunakan oleh masyarakat. Pada penelitian ini, diambil studi kasus pada Cafe FST Corner yang berlokasi di Kampus Universitas Trilogi, Jakarta Selatan. Cafe ini merupakan cafe yang menyediakan berbagai olahan makanan hasil tangan mahasiswa.

Dalam menjalankan usahanya, pelayanan Cafe FST Corner masih menggunakan sistem paper-based yang artinya untuk melayani tamu yang datang, pelayan mencatat pesanan dalam struk / nota. Sistem pembayarannya masih manual yang menyebabkan perhitungannya cukup memakan waktu. Pengelola café sering kerepotan dalam mencatat dan menghitung pembayaran karena banyaknya pelanggan di Café ini. Tak sedikit juga pelanggan yang tidak jadi membeli karena waktu menunggu lama.

Dari wawancara singkat yang dilakukan oleh penulis kepada pelayan yang ada di café ini, setiap harinya sekitar 75% pelanggan yang sudah datang dan terlayani. Sisanya meninggalkan tempat dan mencari makanan ditempat lain. Hal ini sangat mengganggu pemasukan yang harusnya diterima oleh Café ini.

Cafe FST Corner juga nantinya akan berkembang menjadi lebih besar seiring dengan bertambahnya mahasiswa dan kebutuhan mahasiswa yang ada. Maka dari itu cafe ini membutuhkan sistem pelayanan yang baru agar proses pelayanan pada cafe ini menjadi lebih baik dan dapat melayani seluruh pelanggannya dengan lebih baik.

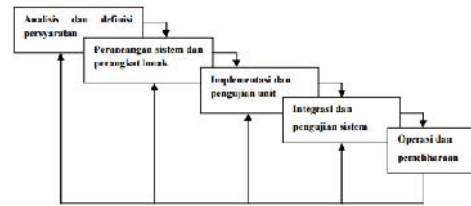
Berdasarkan beberapa masalah tersebut, maka dibuatlah sistem yang mampu untuk memberikan solusi dalam mengerjakan pelayanan transaksi dan pemesanan pada sebuah cafe dan juga membuat sistem pemesanan dan pembayaran berbasis web dimana penulis juga akan merancang database yang baru agar data lebih terstruktur dan bisa menghasilkan informasi yang lebih akurat. Kelebihan aplikasi adalah adanya pemesanan digital yang terletak pada setiap meja untuk melakukan pemilihan makanan yang ada pada cafe tersebut sehingga data pesanan yang masuk dapat dilihat oleh kasir dan pada bagian dapur/pelayan, auto-warning untuk notifikasi meja yang ingin melakukan pembayaran. Selain itu juga manager cafe dapat melihat laporan penjualan makanan dan minuman per periode yang diinginkan.

2. METODE PENELITIAN

Metode pengerjaan yang digunakan untuk membuat sistem ini sesuai dengan tahapan pengembangan perangkat lunak menggunakan model proses atau paradigma *waterfall*. *Waterfall model* memiliki tempat penting dalam rekayasa perangkat lunak. Bahkan paradigma ini merupakan paradigma rekayasa perangkat lunak yang paling luas dipakai dan yang paling tua. Alasan lain penggunaan metode *waterfall* model dalam pembuatan sistem informasi pada Café FST Corner adalah jumlah pengembang perangkat lunak yang sangat terbatas.

Disamping itu, tahapan pada *waterfall* model mengambil kegiatan dasar yang digunakan dalam hampir semua pengembangan perangkat lunak, sehingga dapat lebih mudah untuk dipahami terlebih bila hanya digunakan dalam mengembangkan perangkat lunak yang tidak begitu besar dan kompleks [1].

Adapun metode pengerjaan yang digunakan adalah metode *waterfall*.



Gambar 1. Model Waterfall

Berikut ini adalah tahap-tahap pengembangan dengan menggunakan metode *waterfall*:

a. Analisis dan Definisi Persyaratan (kebutuhan perangkat lunak)

Proses mengumpulkan informasi kebutuhan sistem/perangkat lunak melalui konsultasi dengan pengguna / *user system*. Proses ini mendefinisikan secara rinci mengenai fungsi-fungsi, batasan dan tujuan dari perangkat lunak sebagai spesifikasi sistem yang akan dibuat.

Pada tahap ini dilakukan dengan wawancara langsung kepada pemilik cafe untuk mengetahui data yang dibutuhkan.

b. Perancangan Sistem dan Perangkat Lunak

Proses perancangan sistem ini difokuskan pada empat atribut, yaitu struktur data, arsitektur perangkat lunak, representasi antarmuka, dan detail (algoritma) prosedural. Struktur data adalah representasi dari hubungan logis antara elemen-elemen data individual.

c. Implementasi

Pada tahap ini, membangun perangkat lunak yang direalisasikan sebagai serangkaian program atau unit program. Kemudian pengujian unit melibatkan verifikasi bahwa setiap unit program telah memenuhi spesifikasinya. Pada tahap ini adalah menerjemahkan desain yang telah dibuat kedalam code sehingga terbangun fungsionalitas yang diinginkan.

d. Integration

Bagian-bagian program diintegrasikan menjadi sebuah kesatuan sistem dan kemudian dilakukan pengujian. Dengan kata lain, pengujian ini ditujukan untuk menguji keterhubungan dari tiap-tiap fungsi perangkat lunak untuk menjamin bahwa persyaratan sistem telah terpenuhi.

Pada tahap ini dilakukan *System Testing* dan *User- Acceptance Testing* (UAT) untuk mengetahui apakah sistem dan fungsionalitas yang dibangun sudah sesuai dengan kebutuhan atau tidak.

e. Operation and Maintenance

Tahap ini biasanya memerlukan waktu yang paling lama. Tahapan ini menerapkan sistem (diinstall dan dipakai). Pemeliharaan mencakup koreksi dari beberapa kesalahan yang belum diketemukan pada tahapan

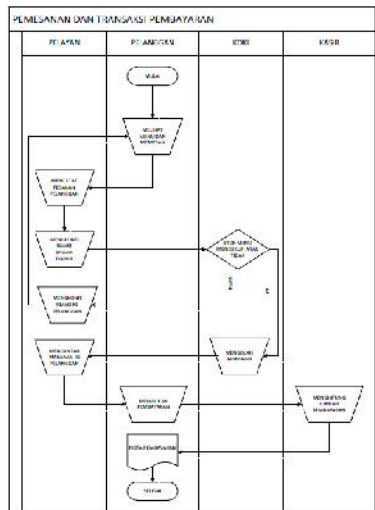
sebelumnya. Perbaikan atas implementasi unit sistem dan pengembangan pelayanan sistem merupakan bagian dari tahapan ini. Tidak menutup kemungkinan untuk persyaratan persyaratan baru ditambahkan dalam masa ini.

3. ANALISIS

Pada bagian analisis ini, penulis akan menganalisis dari berbagai aspek yaitu *Flowmap*, *Diagram Konteks*, dan *Data Flow Diagram*. Analisis *Flowmap* ini menganalisis mengenai aktifitas dalam pemesanan antara pelanggan, koki dan sistem baik yang saat ini berjalan maupun yang akan diusulkan. *Diagram Konteks* adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup sebuah sistem. *Diagram Konteks* yang dibuat ini melingkupi pemesanan dan pembayaran pada sistem yang dibuat. *Data Flow Diagram* adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, tersruktur dan jelas. *Data Flow Diagram* yang dibuat ini menjelaskan mengenai pemesanan dan pembayaran pada sistem yang dibuat [2].

3.1. Flowmap Saat Ini

Berikut adalah flowmap pemesanan dan pembayaran yang sedang berjalan..



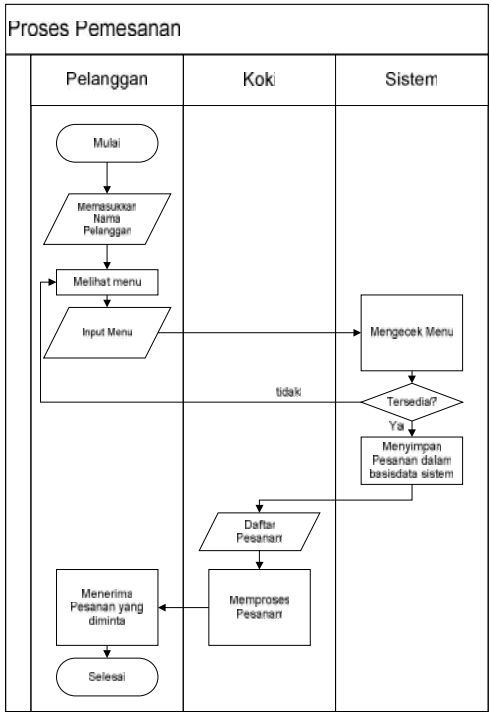
Gambar 2 Flowmap pemesanan dan pembayaran

Flowmap pemesanan dan pembayaran yang sedang berjalan saat ini dimulai dari pelanggan yang melihat menu dan memesan. Pelayan mencatat pesanan dan mengkonfirmasi koki yang berada di dapur. Koki mengecek menu yang dipesan apakah ada atau tidak. Jika ada maka dibuatkan, jika tidak maka mengkonfirmasi ke pelayan dan pelayan mengkonfirmasi kepada pemesan bahwa menu

sudah habis / tidak ada. Sesudah mengolah makanan yang dipesan, koki memberitahu pelayan agar dibawakan ke pelanggan dan pelanggan melakukan pembayaran. Kasir menghitung jumlah pembayaran dan memberikan nota pembayaran kepada pelanggan.

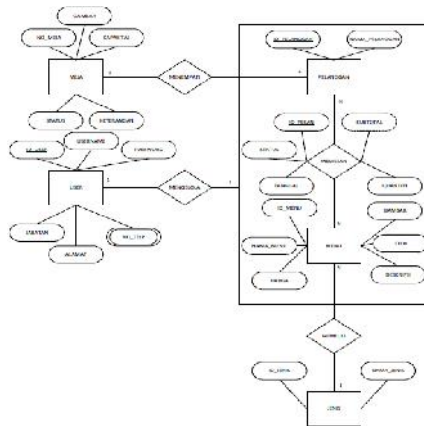
3.2. Flowmap yang Diusulkan

Berikut adalah *flowmap* sistem pemesanan yang baru:



Gambar 3 Flowmap Pemesanan

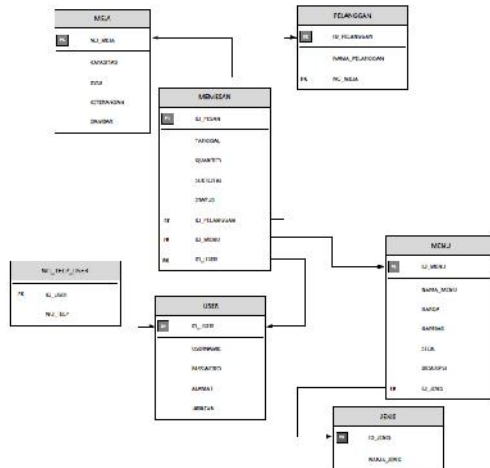
Pada menu pemesanan ini, Pelanggan memasukkan nama dan melihat menu yang ada. Pelanggan dapat memesan menu dari program ini. Setelah itu, sistem akan mengecek menu tersebut, apakah tersedia atau tidak. Apabila tidak tersedia, maka akan memberikan notifikasi kepada pelanggan bahwa menu tersebut tidak tersedia. Pelanggan dapat memesan menu lain dari daftar menu yang ada. Apabila tersedia, maka sistem akan menyimpan pemesanan yang dilakukan oleh pelanggan. Setelah itu, sistem akan maka akan mengirimkan notifikasi kepada koki untuk dibuatkan. Setelah dibuatkan, maka koki dapat memproses pesanan dan memberikan kepada pelanggan ketika pesanan sudah selesai diproses.



Gambar 8 Entity Relationship Diagram

4.2. Relasi Antar Tabel

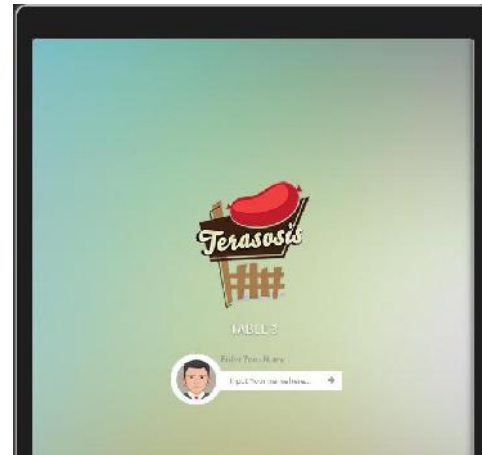
Berikut adalah Relasi Antar Tabel dari penelitian ini:



Gambar 9 Relasi Antar Tabel

4.3. Implementasi Halaman Login Pelanggan

Pembuatan halaman login ini merupakan suatu identifikasi yang berbeda antara suatu pengguna dengan pengguna yang lain [5]. Pelanggan harus melalui halaman ini untuk mengakses halaman menu. Pelanggan memasukkan nama lalu tekan enter. Menu ini tidak tersedia apabila menggunakan device yang sudah terletak di meja karena setiap device yang tersedia meja sudah mempunyai identifikasi berupa nomor meja masing-masing.



Gambar 10 Halaman Login Pelanggan

4.4. Implementasi Halaman Kategori Menu

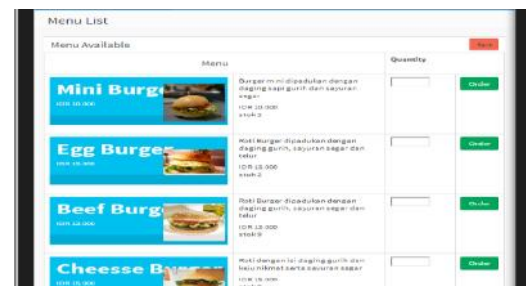
Setelah berhasil masuk (*login*), Pelanggan dapat memilih menu berdasarkan kategorinya.



Gambar 11 Halaman Kategori Menu

4.5. Implementasi Halaman Memesan Menu

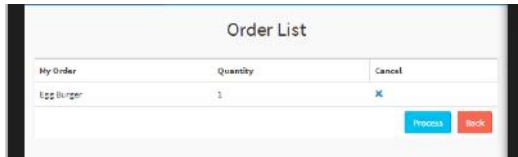
Ketika memilih kategori menu, maka akan muncul macam-macam menu berdasarkan kategori yang dipilih tadi. Pelanggan memasukkan jumlah pesanan pada field *quantity* kemudian tekan tombol order. Jika pelanggan ingin kembali kehalaman kategori menu, tekan tombol back. Namun jika pelanggan sudah selesai memesan, tekan tombol finish.



Gambar 12 Halaman Memesan Menu

4.6. Implementasi Halaman Daftar Pesanan Pelanggan

Setelah selesai memesan, maka akan muncul daftar pesanan pelanggan yang tadi telah dipesan. Pelanggan bisa tekan tombol *back* untuk memesan menu yang lain, namun jika pelanggan sudah selesai memesan, pelanggan bisa tekan tombol *process*.



Gambar 13 Halaman Daftar Pesanan Pelanggan

4.7. Implementasi Halaman Setelah Memesan

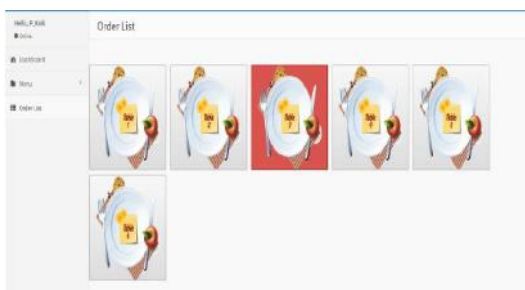
Setelah pelanggan tekan tombol *process* tadi, maka akan muncul daftar pesanan yang telah dipesan beserta harga dan totalnya. Penghitungan total bayar merupakan hasil kali antara jumlah yang dibayarkan dengan harga barang [6]. Pada halaman ini pelanggan bisa tekan tombol *More Order's* untuk memesan menu yang lainnya. Pelanggan juga bisa tekan tombol *billing* untuk mengetahui tagihan yang harus dibayar. Halaman setelah pemesanan dapat dilihat pada gambar 14.

No	My Order	Quantity	Price	Sub Total
1	Egg Burger	1	13,000	13,000
2	Cheese Burger	1	15,000	15,000
Total				IDR 28,000

Gambar 14 Halaman Setelah Memesan

4.8. Implementasi Halaman Bagian Dapur

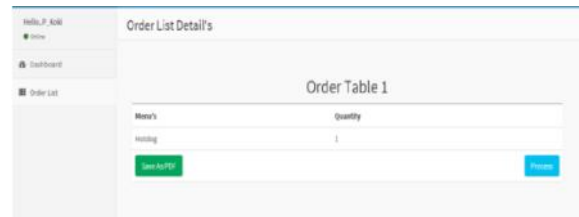
Pada halaman ini bagian dapur akan mengetahui jika ada pesanan yang masuk berdasarkan nomor mejanya. Jika kotak berwarna merah, maka ada pesanan masuk. Jika kotak berwarna putih maka tidak ada pesanan yang masuk. Halaman dapur dapat dilihat pada gambar 15.



Gambar 15 Halaman Bagian Dapur

4.9. Implementasi Halaman Daftar Pesanan Dapur

Setelah bagian dapur klik kotak berwarna merah, maka akan muncul daftar pesanan yang telah dipesan oleh pelanggan. Bagian dapur klik tombol *preview as PDF* untuk menyimpan sementara daftar pesanan pada meja tersebut. Lalu bagian dapur dapat klik tombol *process* untuk memproses pesanan tersebut.



Gambar 16 Halaman Daftar Pesanan Dapur

4.10. Implementasi Halaman Input Menu Baru

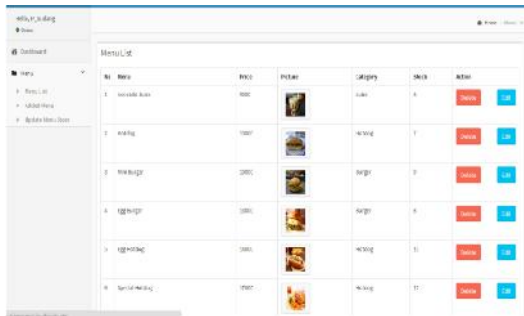
Pada halaman ini bagian gudang dapat menginputkan menu baru dengan mengisi data-data menu baru tersebut. Setelah itu tekan *save* untuk menyimpan data menu baru tersebut.



Gambar 17 Halaman Input Menu Baru

4.11. Implementasi Halaman List Menu Dapur

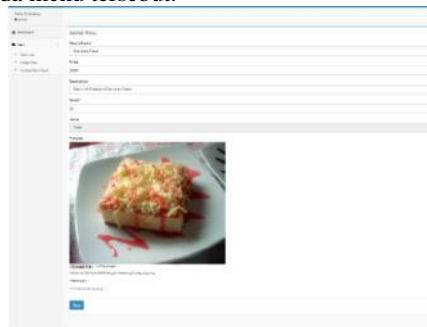
Pada halaman ini akan ada data-data semua menu yang ada. Bagian dapur dapat menghapus menu dengan klik tombol *delete*. Jika bagian gudang ingin meng-update menu, bagian dapur dapat klik tombol *edit*.



Gambar 18 Halaman List Menu Dapur

4.12. Implementas Halaman Update Menu

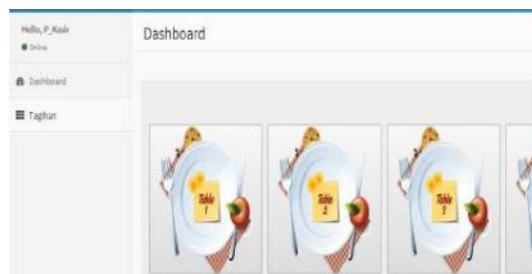
Pada halaman ini bagian dapur dapat meng-*update* menu yang ingin diubah. Bagian dapur hanya perlu merubah apa saja yang ingin diubah pada menu tersebut.



Gambar 19 Halaman Update Menu

4.13. Implementasi Halaman Bagian Kasir

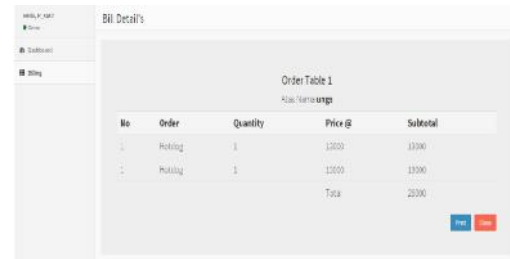
Pada halaman ini bagian kasir dapat melihat meja mana yang akan melakukan pembayaran.



Gambar 20 Halaman Bagian Kasir

4.14. Implementasi Halaman Daftar Pesanan Pada Kasir

Setelah kasir menekan kotak meja yang ingin melakukan pembayaran, maka akan muncul halaman yang berisi daftar tagihan biaya pesanan pelanggan. Kasir tekan tombol *print* untuk mencetak struk tagihan tersebut, setelah itu kasir tekan tombol *close* untuk menandakan bahwa transaksi pembayaran sudah selesai.



Gambar 21 Halaman Daftar Pesanan Pada Kasir

4.15. Implementasi Halaman Informasi Tagihan Pelanggan

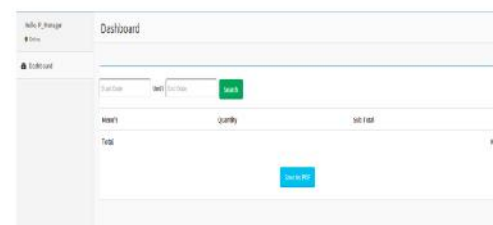
Setelah pelanggan tekan tombol *billing* tadi, maka akan muncul informasi tagihan pelanggan yang harus dibayar.



Gambar 22 Halaman Informasi Tagihan Pelanggan

4.16. Implementasi Halaman Manager

Pada halaman ini *manager* dapat melihat laporan penjualan berdasarkan tanggal. Manager bisa meng-*inputkan* tanggal awal dan tanggal akhir untuk melihat laporan penjualan diantara tanggal berikut. Lalu manager tekan tombol *search* dan akan muncul data menu apa saja yang terjual diantara tanggal yang di-*inputkan* tadi.



Gambar 23 Halaman Manager

5. PENGUJIAN

Pengujian yang dilakukan penulis ini menggunakan metode pengujian *black box*. Pengujian ini dilakukan dengan cara menguji aplikasi dari segi fungsionalitasnya, yaitu memberikan *inputan* berdasarkan suatu kondisi, lalu mengamati apakah keluaran yang dihasilkan sesuai dengan keluaran yang diharapkan sebelumnya dan memberikan kesimpulan dari hasil pengujian tersebut.

Pengujian terhadap sistem harus dilakukan untuk tercapainya sistem yang akurat dan tepat guna. Proses pengujian berfokus pada logika internal software, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional, yaitu mengarahkan pengujian untuk menemukan kesalahan – kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan [7].

Pada aplikasi ini, penulis melakukan pengujian terhadap 14 halaman yang tersedia. Dalam tahap pengujian ini, dibuat rencana pengujian aplikasi agar pengujian ini terukur. Terukur dari apa yang akan diuji, apa detail uji yang hendak dilakukan dan apa jenis uji yang diberikan. Rencana pengujian aplikasi dapat dilihat dalam tabel 1 berikut ini :

Tabel 1. Rencana Pengujian Aplikasi

Kelas Uji	Detail Uji	Jenis Uji
Login pelanggan	Pelanggan dapat login dengan inputan nama	<i>Black box</i>
Kategori menu	Sistem dapat menampilkan menu berdasarkan kategori	<i>Black box</i>
Halaman pemesanan	Sistem dapat menerima pemesanan dari pelanggan	<i>Black box</i>
Daftar pesanan	Sistem dapat memproses pesanan	<i>Black box</i>
Halaman setelah memesan	Sistem dapat menampilkan menu yang telah dipesan	<i>Black box</i>
Bagian Dapur	Sistem dapat menampilkan nomor meja yang memesan menu	<i>Black box</i>
Daftar pesanan dapur	Melihatkan daftar yang dipesan pelanggan	<i>Black box</i>
Input menu	Sistem dapat menginput menu	<i>Black box</i>
List menu dapur	Sistem dapat menghapus dan mengedit menu	<i>Black box</i>
Update menu	Sistem dapat mengupdate menu	<i>Black box</i>
Bagian kasir	Melihat meja yang melakukan pembayaran	<i>Black box</i>
Daftar pesanan pada kasir	Melihat pemesanan yang telah dilakukan	<i>Black box</i>

Kelas Uji	Detail Uji	Jenis Uji
	dan siap untuk dibayar	
Tagihan pelanggan	Memunculkan informasi tagihan pelanggan yang harus dibayar	<i>Black box</i>
Halaman Manager	Melihat laporan penjualan	<i>Black box</i>

Dari rencana yang didapat dari tabel 1 diatas, maka dapat dilakukan beberapa tahap pengujian sebagai berikut :

1. Pengujian Login Pelanggan

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Input nama	Nama yang diinputkan	Handal
Tombol login	Menampilkan halaman kategori menu	Handal

2. Pengujian Kategori Menu

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol kategori menu	Menampilkan menu yang ada dalam kategori tersebut.	Handal

3. Pengujian Halaman Pemesanan

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol order	Memesan menu yang dipilih	Handal
Menambahkan nilai <i>quantity</i>	Menu yang dipesan bertambah	Handal

4. Pengujian Daftar Pesanan

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol cancel	Menu yang dipesan menghilang	Handal
Tombol Proses	Memproses pesanan	Handal
Tombol back	Kembali ke halaman pemesanan	Handal

5. Pengujian Halaman setelah Pesanan

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol more orders	Kembali ke halaman pemesanan	Handal
Tombol billing	Memproses tagihan	Handal

6. Pengujian Bagian Dapur

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol meja	Melihat daftar pesanan yang dilakukan oleh meja ybs.	Handal

7. Pengujian Daftar Pesanan Dapur

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol proses	Memproses pesanan	Handal
Tombol Save As PDF	Menyimpan dalam bentuk PDF	Handal

8. Input menu baru

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Input data dalam <i>name, price, description, stock</i>	Data terinput	Handal
Tombol <i>choose category</i>	Memilih kategori menu yang bersangkutan	Handal
Tombol <i>pictures</i>	Memilih gambar menu yang bersangkutan	Handal
Tombol <i>save</i>	Menyimpan menu yang sudah diinputkan	Handal

9. Pengujian List Menu Dapur

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol Delete	Menghapus menu	Handal
Tombol Edit	Mengedit	Handal

	menu dengan menampilkan halaman update menu	
--	---	--

10. Update Menu

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Input data dalam <i>name, price, description, stock</i>	Data terinput	Handal
Tombol <i>choose category</i>	Memilih kategori menu yang bersangkutan	Handal
Tombol <i>pictures</i>	Memilih gambar menu yang bersangkutan	Handal
Tombol <i>save</i>	Menyimpan menu yang sudah diedit	Handal

11. Pengujian Bagian Kasir

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol meja	Melihat daftar pesanan yang dilakukan oleh meja ybs.	Handal

12. Pengujian Daftar Pesanan Pada Kasir

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Tombol Proses	Memproses pesanan	Handal
Tombol back	Kembali ke bagian kasir	Handal

13. Pengujian Halaman Tagihan Pelanggan

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
-	Menampilkan data yang telah dipesan oleh pelanggan ybs.	Handal

14. Pengujian Halaman Manager

Hasil uji		
Data Masukan	Hasil yang diharapkan	Kesimpulan Uji
Input data start date dan end date	Data dapat diinputkan	Handal
Tombol search	Menampilkan data sesuai dengan tanggal yang diinputkan	Handal
Tombol Save As PDF	Menyimpan dalam bentuk PDF	Handal

6. KESIMPULAN

Berdasarkan pembangunan dan pengujian aplikasi ini, maka dapat disimpulkan:

1. Aplikasi ini dapat membantu dalam memesan dan melihat menu di Cafe FST Corner Universitas Trilogi secara digital yang terletak pada setiap meja atau dapat memesan secara online menggunakan *smartphone* secara pribadi dengan menuliskan nama pemesan.
2. Aplikasi ini dapat membantu bagian kasir untuk menghitung tagihan pembayaran di Cafe FST Corner Universitas Trilogi. Selain itu aplikasi ini dapat membantu Manager cafe dalam melihat laporan penjualan per periode yang diinginkan.
3. Aplikasi ini dapat membantu menghubungkan antara pelanggan, kasir, dan bagian dapur.
4. Hasil pengujian aplikasi ini menunjukkan bahwa aplikasi ini sudah di uji dengan baik

dan dapat berjalan sebagaimana fungsinya. Semua kemungkinan yang dapat dilakukan pengguna sudah teruji dan dapat berfungsi dengan baik. Hasil yang diharapkan dari pengujian inipun memenuhi harapan dari pengembang aplikasi ini.

7. DAFTAR RUJUKAN

- [1] Sommerville, Ian (2009), Software Engineering Ninth Edition, Addison – Wesley, hal. 30. Rosa A. S. Rekayasa. Perangkat Lunak (Terstruktur dan Berorientasi Objek). Penerbit Modula, 2011
- [2] Jogiyo H, Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis, Yogyakarta: Andi Offset, 2005
- [3] Nugroho E, Ratnasari K, Ramadhani K and Putro B, Rekayasa Perangkat Lunak, Bandung: Politeknik Telkom, 2009.
- [4] Whitten, Jeffrey L. Bentley, Lonnie D. Dittman, Kevin C 2004, System Analysis and Design, Andi Offset, Yogyakarta.
- [5] Permana, Silvester Dian Handy, dkk.. Analisis dan Perancangan Sistem Informasi Toko Dika. Seminar Nasional Literasi Informasi (Senarai 2014). Medan. 2014:439-443
- [6] Stice, Kousen. Intermediate Accounting: Akutansi Keuangan 1 Edisi 16, Jakarta: Salemba Empat, 2007
- [7] Pandara, Eko, Stanley Karouw, Meicsy Najoan, 2014. Implementasi Proses Uji Sistem Informasi Admisi Pascasarjana Universitas Sam Ratulangi. E-journal Teknik Informatika, Volume 4, No. 2 (2014), ISSN : 2301-8364.