

## Pengembangan Sistem Informasi Manajemen Pemesanan Pakaian dan Keuangan Berbasis Web (Studi Pada: YP Screenprinting)

Fitra Wahyuda<sup>1</sup>, Satrio Agung Wicaksono<sup>2</sup>, Tri Afirianto<sup>3</sup>

Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>fitrawahyuda@student.ub.ac.id, <sup>2</sup>satrio @ub.ac.id, <sup>3</sup>tri.afirianto@ub.ac.id

### Abstrak

YP Screenprinting merupakan salah satu usaha kecil menengah (UKM) di Kabupaten Bojonegoro yang bergerak di bidang konveksi. Pesanan yang dikerjakan sebagian besar berupa kaos yang berasal dari dalam maupun luar kota. Proses pencatatan pesanan di Yp Screenprinting masih dilakukan secara manual dan pengeluaran untuk pembelian bahan baku tidak dicatat. Sehingga hal tersebut menimbulkan permasalahan seperti sulitnya mencari data pesanan atau pengeluaran dan pemilik usaha tidak bisa mengetahui berapa keuntungan atau kerugian yang dialami secara pasti. Berdasarkan permasalahan tersebut dibuatlah sebuah solusi berupa sistem informasi berbasis web untuk membantu mempermudah proses pengelolaan pesanan dan pengeluaran. Pembuatan sistem informasi dilakukan menggunakan *system development life cycle* (SDLC) berupa metode *Waterfall*. Pada tahap *requirements definition* menghasilkan 22 kebutuhan fungsional. Pada tahap *software design* menggunakan pendekatan *object oriented analysis design* (OOAD) dan hasilnya digambarkan menggunakan *unified modelling language* (UML) yang mana menghasilkan 3 kelas kontroler dan 4 kelas model. Pada tahap *implementation* menggunakan pola *model view controller* (MVC) sebagai arsitektur sistem. Terakhir pada tahap *system testing* diperoleh hasil 100% *valid* pada *validation testing* dan hasil 88,75% pada *user acceptance Testing* (UAT). Berdasarkan hasil pengujian tersebut bisa dikatakan bahwa sistem informasi yang sudah dibuat diterima dengan baik oleh pengguna.

**Kata kunci:** *sistem informasi, manajemen pemesanan, Waterfall, validation testing, UAT*

### Abstract

*YP Screenprinting is one of the small and medium businesses (SMB) in Bojonegoro Regency which is engaged in convection. Most of the orders placed in the form of t-shirts are from inside and outside the city. The process of recording orders at Yp Screenprinting is still done manually and expenses for purchasing raw materials are not recorded. So that it causes problems such as difficulty in finding data orders or expenses and business owners can not know how much profit or loss is experienced with certainty. Based on these problems a solution is made in the form of a web-based information system to help simplify the process of managing orders and expenses. Making information systems is done using the system development life cycle (SDLC) in the form of the Waterfall method. In the requirements definition stage it produces 22 functional requirements. The software design stage uses the object oriented analysis design (OOAD) approach and the results are described using a unified modeling language (UML) which produces 3 controller classes and 4 model classes. At the implementation stage using the model view controller (MVC) as the system architecture. Finally in the system testing stage, the results are 100% valid in validation testing and 88.75% results in user acceptance testing. Based on the test results it can be said that the information system that has been made is well received by the user.*

**Keywords:** *information system, booking management, Waterfall, validation testing, UAT*

## 1. PENDAHULUAN

Informasi pada perusahaan merupakan hal

yang sangat penting karena salah satu penunjang kelancaran kinerja perusahaan. Perusahaan dapat berkembang dengan cepat dan terus maju karena adanya perbaikan kinerja yang disebabkan oleh pemanfaatan informasi yang dimiliki oleh perusahaan tersebut (Aknovia, 2018). Oleh sebab itu, apabila suatu perusahaan tidak mau beradaptasi dengan perkembangan teknologi, maka perusahaan tersebut akan sulit untuk bersaing dan berkembang dengan kompetitornya, bahkan bisa mengakibatkan kalah saing dan harus menutup usahanya. Namun, apabila sebuah perusahaan dapat mengelola informasi yang dimiliki, maka perusahaan tersebut akan bisa berkembang dan bahkan bisa menguasai pasar. Untuk mengelola informasi tersebut bisa dilakukan secara manual atau menggunakan sistem. Namun, pengelolaan informasi menggunakan suatu sistem cenderung lebih efektif dan efisien.

YP Screenprinting merupakan salah satu usaha kecil menengah (UKM) di Kabupaten Bojonegoro yang bergerak di bidang konveksi yang memproduksi kaos, jersey, dll. UKM ini menerima pesanan dari dalam maupun luar kota. Berdasarkan hasil wawancara dengan mas Yoppy selaku pemilik YP Screenprinting, proses pencatatan pesanan masih dilakukan secara manual. Bahkan, untuk pengeluaran pembelian bahan baku tidak dicatat. Hal tersebut menimbulkan beberapa permasalahan seperti pemilik usaha sulit untuk melihat data pesanan karena harus mencari secara manual, terutama untuk pesanan yang sudah lama. Selain itu, pemilik usaha juga merasa kesulitan melihat total pemasukan dan pengeluaran. Hal tersebut karena tidak pernah dibuatnya laporan pemasukan dan pengeluaran bulanan, sehingga untuk keuntungan atau kerugian sulit diketahui secara pasti. Hingga saat ini, pemilik usaha hanya menggunakan perkiraan berdasarkan total produksi dalam sehari untuk mengetahui keuntungan atau kerugian yang didapat.

Dalam penelitian yang dilakukan oleh Laar, konjaang, dan Tankia (2015) mengungkapkan bahwa dengan menggunakan sistem manajemen berbasis komputer dalam sebuah perusahaan menghasilkan beberapa manfaat, yaitu sistem yang memiliki antarmuka bagus akan mudah digunakan, apabila dikelola dengan baik akan mempermudah pembuatan laporan, sistem dapat membantu manajemen penjualan dengan melacak dan menyimpan data

yang relevan agar menjadi efektif, dan apabila perusahaan tersebut berkembang, sistem tersebut juga bisa dikembangkan lagi sesuai dengan kebutuhan perusahaan.

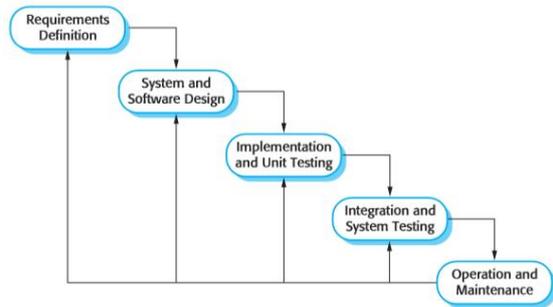
Berdasarkan masalah tersebut diperlukan sebuah sistem informasi untuk membantu proses pencatatan serta pengelolaan pesanan dan pengeluaran. Sehingga data tersebut akan mudah dicari, serta dapat membantu pemilik usaha mengetahui keuntungan atau kerugian yang didapat. Dalam mengembangkan sistem informasi tersebut akan cocok menggunakan pendekatan *system development life cycle* (SDLC) berupa metode *Waterfall*. Metode *Waterfall* adalah metode yang terstruktur dan cocok digunakan ketika sistem memiliki kebutuhan pengguna yang spesifik dan tidak banyak berubah selama pengembangan sistem (Sommerville, 2011).

## 2. LANDASAN PUSTAKA

YP Screenprinting dalam sehari dapat memproduksi 200-250 kaos dan memiliki 14 pegawai. Pesanan yang dikerjakan oleh YP Screenprinting kebanyakan berasal dari komunitas pecinta motor dan perkumpulan anak muda baik dalam maupun luar kota seperti Jambi, Lampung, Lombok, dll. Untuk membuat jersey lebih cepat, YP Screenprinting menggunakan sebuah mesin print untuk mencetak motif jersey. Akan tetapi, dalam proses pencatatan pesanan yang masuk masih dilakukan secara manual, bahkan pengeluaran yang dilakukan tidak dicatat.

Sistem informasi adalah sekumpulan komponen yang saling terkait untuk mengumpulkan, memproses, menyimpan, dan membagikan informasi guna mendukung pengambilan keputusan dan kendali di dalam sebuah organisasi (Laudon dan Laudon, 2014). Sehingga dengan menggunakan sistem informasi, pengguna dapat mengelola data atau informasi yang dimiliki dengan lebih baik. Langkah-langkah dalam mengembangkan sebuah sistem informasi sering disebut dengan *system development life cycle* (SDLC). Salah satu metode dalam SDLC adalah *Waterfall*. Menurut Sommerville (2011), metode *Waterfall* adalah metode yang terstruktur yang terdiri dari beberapa tahapan. Setiap tahapan akan menghasilkan dokumen yang harus disetujui dan apabila proses saat ini belum selesai maka tidak bisa lanjut ke tahap selanjutnya dan tidak bisa kembali ke tahap sebelumnya kecuali

sudah sampai pada tahap terakhir. Sehingga metode *Waterfall* cocok digunakan dalam pengembangan sistem informasi yang sudah memiliki tujuan yang jelas. Metode *Waterfall* dapat dilihat pada Gambar 1.



Gambar 1. Metode *Waterfall*  
 Sumber : (Sommerville, 2011)

Tahap pertama adalah *requirements definition* yang merupakan fase untuk menggali kebutuhan pengguna. Tahap kedua adalah *system and software design* yang merupakan fase untuk membuat rancangan sistem yang akan dibuat berdasarkan kebutuhan pengguna yang sudah diperoleh. Salah satu pendekatan yang dapat digunakan dalam merancang sistem adalah *object oriented analysis design* (OOAD) dan hasilnya nanti dapat digambarkan menggunakan *unified modelling language* (UML). Tahap ketiga adalah *implementation and unit testing* yang merupakan tahap untuk mengubah rancangan sistem yang sudah dibuat menjadi suatu unit program dan memastikan setiap unit sesuai dengan spesifikasinya. Salah satu pola yang dapat digunakan adalah pola *model view controller* (MVC). Menurut Laudon dan Laudon (2014), MVC merupakan cara untuk memisahkan sistem menjadi 3 komponen, yaitu *model* untuk mengelola data yang dibutuhkan sistem, *view* merupakan tampilan yang disajikan ke pengguna, dan *controller* untuk mengatur proses yang terjadi pada sistem. Tahap keempat adalah *integration and unit testing* yang merupakan tahap untuk menggabungkan setiap unit program kemudian mengujinya sebagai sebuah sistem yang utuh. Pengujian yang dapat digunakan adalah *validation testing* yaitu menguji fitur-fitur yang ada pada sistem sudah berjalan sesuai dengan harapan atau belum dengan menjalankan beberapa kasus uji (Sommerville, 2011). Selain itu ada juga *user acceptance testing* (UAT) yaitu proses perangkat lunak diserahkan kepada pengguna untuk digunakan selama periode waktu tertentu kemudian pengguna menyetujui

atau menolak produk perangkat lunak tersebut (Eriksson, 2016). Salah satu cara mengukur UAT adalah menggunakan skala *likert*. Menurut Rahman, Tolle, dan Dewi (2018) untuk menghitung nilai skala *likert* dapat menggunakan Persamaan (1). Kemudian hasil yang diperoleh dikategorikan menurut interpretasi seperti pada Tabel 1.

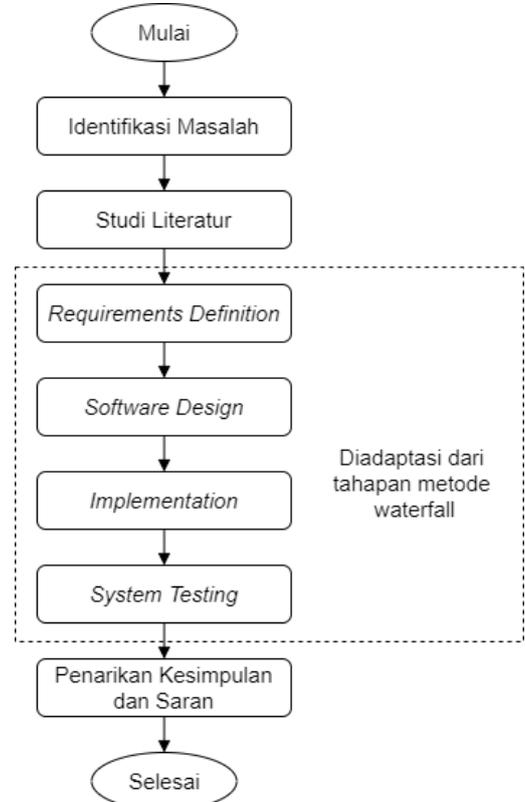
$$Index(\%) = \left(\frac{A}{B \times N}\right) \times 100\% \tag{1}$$

A = total skor yang diperoleh dari semua responden  
 B = maksimum poin yang bisa diperoleh dari setiap responden  
 N = jumlah responden

Tabel 1. Interpretasi skor likert

Skor Likert	Interpretasi skor dengan interval = 20	Kategori
1	0% – 19,99%	Sangat Tidak Setuju
2	20% – 39,99%	Tidak Setuju
3	40% – 59,99%	Netral
4	60% – 79,99%	Setuju
5	80% – 100%	Sangat Setuju

### 3. METODOLOGI



Gambar 2. Metodologi Penelitian

Penelitian ini menggunakan pendekatan SDLC yang mengadaptasi metode *Waterfall*

pada tahap pertama sampai keempat. Sehingga tahapan-tahapan pada penelitian ini digambarkan pada Gambar 2.

Identifikasi masalah dilakukan dengan melakukan wawancara terhadap pemilik YP Screenprinting. berdasarkan wawancara tersebut, diperoleh permasalahan mengenai pemilik usaha yang sulit mencari data pesanan atau pengeluaran dan tidak bisa mengetahui keuntungan atau kerugian yang dialami secara pasti karena hanya menggunakan perkiraan jumlah produksi dalam sehari.

Studi literatur dilakukan mencari referensi dari jurnal, buku, laporan penelitian, website atau artikel guna mendukung landasan teori pada penelitian in ini

*Requirements definition* dimulai dengan melakukan wawancara dengan pemilik YP Screenprinting untuk mengetahui proses bisnis saat ini (*As-Is*) yang kemudian dianalisis dan dibuatkan proses bisnis usulan (*To-Be*) untuk mengetahui siapa saja pengguna sistem dan dan apa saja kebutuhan pengguna tersebut.

*Software design* dilakukan untuk membuat rancangan sistem dengan cara menganalisa kebutuhan pengguna yang sudah diperoleh pada tahap *requirements definition* menggunakan pendekatan OOAD kemudian hasilnya digambarkan menggunakan UML yang nantinya akan menghasilkan beberapa diagram.

*Implementation* dibuat menggunakan HTML, CSS, Javascript, PHP, dan basis data menggunakan MySQL dengan menerapkan pola MVC. Sehingga nantinya dapat menghasilkan sistem informasi yang dapat digunakan dan sesuai dengan hasil rancangan yang sudah dibuat pada tahap *software design*.

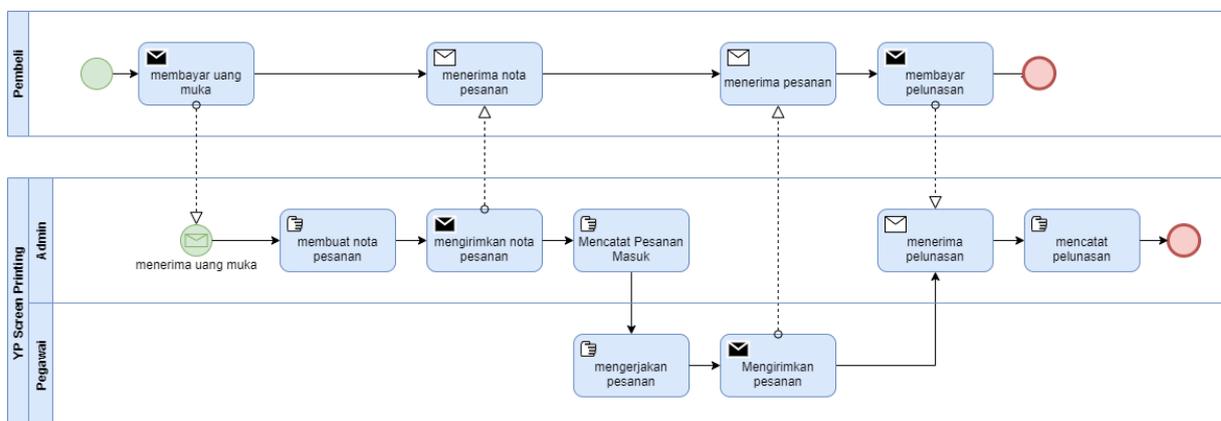
*System testing* dilakukan menggunakan

*validation testing*. Setelah itu sistem diberikan ke pengguna untuk melakukan *user acceptance testing (UAT)* dengan memberikan beberapa pertanyaan berupa kuesioner yang jawabannya menggunakan skala likert 5 poin kemudian dihitung menggunakan Persamaan (1).

Kesimpulan dibuat dari hasil penelitian ini untuk menjawab rumusan masalah. saran bertujuan memberikan saran perbaikan untuk penelitian selanjutnya.

#### 4. HASIL DAN PEMBAHASAN

Tahap *Requirements definition* dilakukan melalui proses wawancara dengan pemilik YP Screenprinting. Hasil wawancara tersebut dianalisis untuk mengetahui proses bisnis yang ada di sana saat ini dan kebutuhan pengguna. Berdasarkan hasil wawancara tersebut dibuat gambaran proses bisnis saat ini (*As-Is*). Terdapat 3 proses bisnis saat ini (*As-Is*) yang dibuat yaitu proses pencatatan pesanan, proses pencatatan pengeluaran yang terdiri dari pembelian bahan baku dan gaji pegawai, dan proses pencarian data pesanan atau pengeluaran. Proses bisnis pencatatan pesanan saat ini (*As-Is*) dapat dilihat pada Gambar 3. Proses bisnis tersebut melibatkan pembeli, admin (pemilik usaha dan admin), dan pegawai. Pada proses tersebut terdapat aktivitas mencatat pesanan yang masih dilakukan secara manual menggunakan buku catatan. Karena hal tersebut, maka proses pencarian lebih sulit karena harus mencari secara manual dan pembuatan laporan keuangan juga akan menjadi sulit juga karena harus menyalin total harga pesanan yang masuk dan total biaya pengeluaran yang dilakukan.

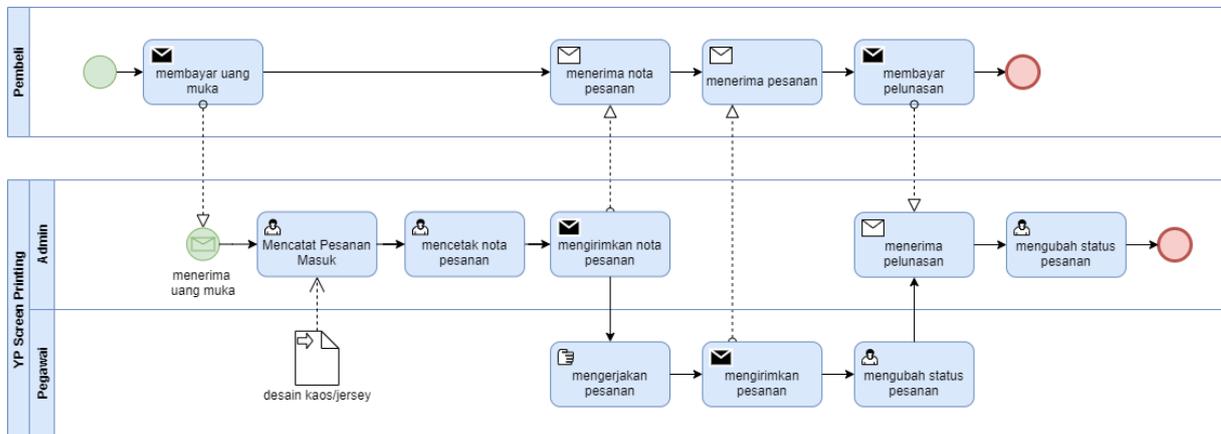


Gambar 3. Proses bisnis saat ini (*As-Is*) pada pencatatan pesanan

Karena ada beberapa masalah pada proses bisnis *As-Is*, maka memerlukan proses bisnis

usulan (*To-Be*) yang melibatkan sistem informasi yang akan dibuat. Proses bisnis usulan berjumlah 4 yaitu proses pencatatan pesanan, proses pencatatan pengeluaran yang terdiri dari pembelian bahan baku dan gaji pegawai, proses pencarian data pesanan atau

pengeluaran, dan proses pembuatan laporan keuangan. Proses bisnis usulan pada pencatatan pesanan dapat dilihat pada Gambar 4. Proses bisnis tersebut melibatkan pembeli, admin (pemilik usaha dan admin), dan pegawai.



Gambar 4. Potongan Proses bisnis usulan (*To-Be*) pada pencatatan pesanan

Pada proses bisnis usulan (*To-be*) tersebut (Gambar 4) terdapat beberapa perubahan, salah satunya adalah aktivitas mencatat pesanan yang awalnya dilakukan secara manual diubah menjadi menggunakan sistem informasi. Proses bisnis usulan (*To-be*) tersebut, selanjutnya dianalisis untuk menentukan saja pengguna sistem informasi yang akan dibuat. Berdasarkan hasil analisis diperoleh 3 pengguna dari sistem informasi yang akan dibuat yaitu pemilik usaha, admin dan pegawai. Selanjutnya membuat daftar aktivitas yang melibatkan sistem informasi dari semua proses bisnis usulan (*To-be*). Berdasarkan semua proses bisnis usulan (*To-be*) terdapat 6 aktivitas yang melibatkan sistem informasi yang akan dibuat. Beberapa aktivitas tersebut dapat dilihat pada Tabel 2.

Tabel 2. Beberapa Aktivitas yang Melibatkan Sistem Informasi

Kode Aktivitas	Nama aktivitas
KA-01	Mencatat pesanan masuk
KA-02	Mencetak nota pesanan
KA-03	Mengubah status pesanan

Daftar aktivitas selanjutnya dianalisis untuk memperoleh kebutuhan pengguna. Berdasarkan hasil analisis tersebut diperoleh 8 kebutuhan pengguna. Salah satu kebutuhan pengguna dapat dilihat pada Tabel 3.

Tabel 3. Salah Satu Kebutuhan Pengguna

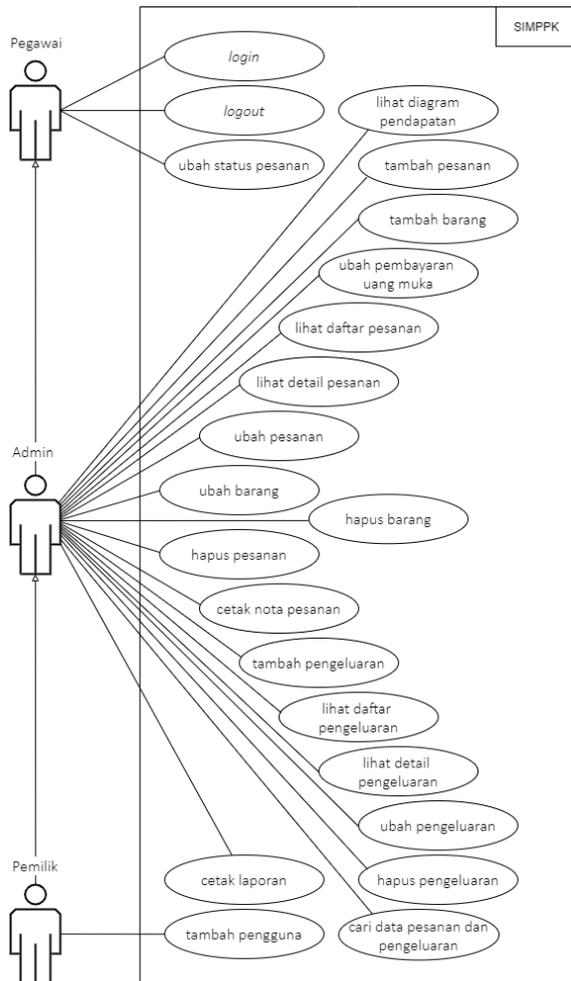
Kode Aktivitas	Kode Kebutuhan	Nama Kebutuhan	Pengguna
KA-01	SIMPP-K-01	Mengelola pesanan	Admin & pemilik usaha

Kebutuhan pengguna selanjutnya dianalisis untuk memperoleh kebutuhan fungsional (fitur) yang nanti akan dibuat pada sistem informasi. Berdasarkan hasil analisis tersebut diperoleh 22 kebutuhan fungsional. Beberapa kebutuhan fungsional dapat dilihat pada Tabel 4.

Tabel 4. Beberapa Kebutuhan Fungsional

Kode Kebutuhan	Kode Fitur	Nama Fitur
SIMPP-K-01	SIMPP-F-01	Tambah pesanan
	SIMPP-F-02	Lihat daftar pesanan
	SIMPP-F-03	Ubah pesanan
	SIMPP-F-04	Hapus pesanan

Kebutuhan fungsional yang sudah didapat selanjutnya digambarkan menggunakan *use case diagram* agar lebih mudah dipahami. *Use case diagram* dapat dilihat pada Gambar 5.



Gambar 5. Use Case Diagram

Setiap *use case* selanjutnya dijabarkan dalam *use case description*. *Use case description* tambah pesanan dapat dilihat pada Tabel 5.

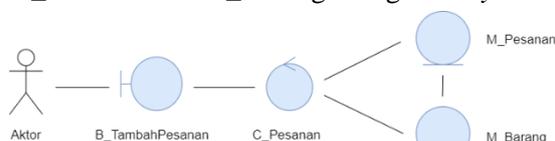
Tabel 5. Use Case Description Tambah Pesanan

<b>Use case name</b>	Tambah Pesanan
<b>Related requirements</b>	Kebutuhan SIMPP-K-01
<b>Goal in context</b>	Aktor dapat menambah data pesanan pelanggan
<b>Preconditions</b>	Aktor berada di halaman tambah pesanan
<b>Successful end condition</b>	Aktor berhasil menambah data pesanan pelanggan
<b>Failed end condition</b>	Aktor gagal menambah data pesanan pelanggan
<b>Primary actors</b>	Admin dan pemilik usaha
<b>Secondary actors</b>	-

<b>Trigger</b>	Aktor ingin menambah data pesanan pelanggan
<b>Main flow</b>	<ol style="list-style-type: none"> <li>1. Aktor memasukkan data pesanan dan data barang yang diperlukan</li> <li>2. Aktor menekan tombol simpan</li> <li>3. Sistem mengecek data yang dimasukkan</li> <li>4. Sistem membuat nomor pesanan dan nomor barang</li> <li>5. Sistem menyimpan data pesanan</li> <li>6. Sistem menampilkan pesan berhasil menambah pesanan</li> </ol>
<b>Extension</b>	<p>Apabila aktor tidak mengisi kolom yang diperlukan</p> <ol style="list-style-type: none"> <li>2.1. Sistem menampilkan pesan ada kolom yang tidak diisi</li> </ol> <p>Apabila tanggal pemesanan yang dimasukkan lebih baru dibanding tanggal batas pengerjaan</p> <ol style="list-style-type: none"> <li>4.1. Sistem menampilkan pesan gagal menyimpan data karena tanggal pemesanan tidak valid</li> </ol> <p>Apabila jumlah atau harga yang dimasukkan aktor bukan angka</p> <ol style="list-style-type: none"> <li>4.2. Sistem menampilkan pesan gagal menyimpan data karena jumlah atau harga bukan angka</li> </ol> <p>Apabila desain yang diunggah aktor tidak sesuai ketentuan</p> <ol style="list-style-type: none"> <li>4.3. Sistem menampilkan pesan gagal menyimpan data karena desain yang diunggah tidak sesuai ketentuan</li> </ol>

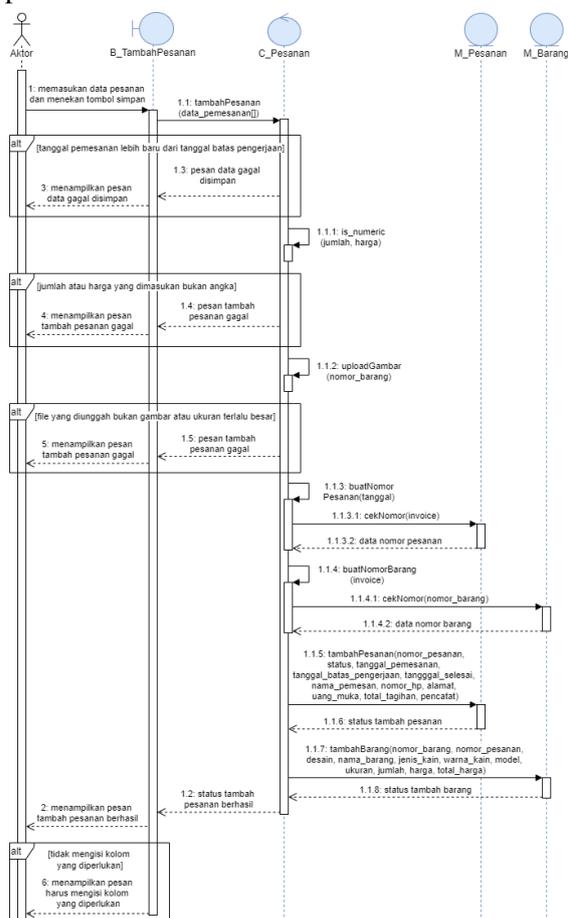
Setelah membuat *use case description*, selanjutnya masuk tahap *software design*. Tahap ini dimulai dengan menganalisa semua *use case* untuk mengetahui *use case* mana yang dapat membantu mempermudah pengelolaan pesanan, pengelolaan pengeluaran, dan pembuatan laporan keuangan. Dari hasil analisis tersebut memperoleh 17 *use case* yang menyelesaikan permasalahan, yaitu tambah pesanan, tambah barang, ubah pembayaran uang muka, lihat daftar pesanan, lihat detail pesanan, ubah pesanan, ubah barang, hapus barang, hapus pesanan, ubah status pesanan, tambah pengeluaran, lihat daftar pengeluaran, lihat detail pengeluaran, ubah pengeluaran, hapus pengeluaran, pencarian pesanan dan pengeluaran, dan cetak laporan.

Selanjutnya 17 *use case* tersebut akan diimplementasikan. Namun perlu dibuat rancangannya terlebih dulu. Langkah pertama pada tahap perancangan adalah membuat *class analysis* dari setiap *use case* untuk menentukan kelas apa saja yang akan terlibat dalam *use case* tersebut. *Class analysis* akan memperoleh kelas *boundary* yang menjadi tampilan antarmuka, kelas *controller* yang mengatur logika untuk menjalankan *use case*, dan kelas *entity* yang mengatur data yang disimpan. *Class analysis* untuk *use case* tambah pesanan dapat dilihat pada Gambar 5 yang mana memiliki kelas B\_TambahPesanan sebagai *boundary*, kelas C\_Pesanan sebagai *controller*, dan kelas M\_Pesanan dan M\_Barang sebagai *entity*.



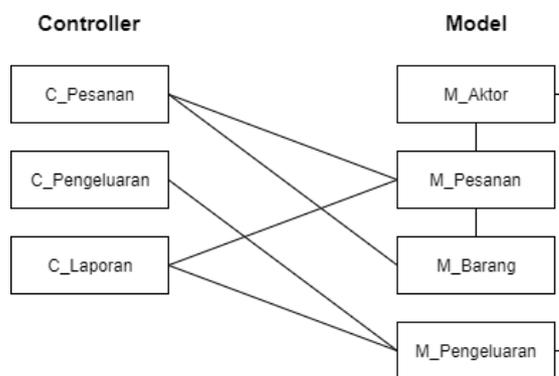
Gambar 5. Class Analysis Tambah Pesanan

Hasil *class analysis* selanjutnya dibuat menjadi *sequence diagram* yang dapat dilihat pada Gambar 6.

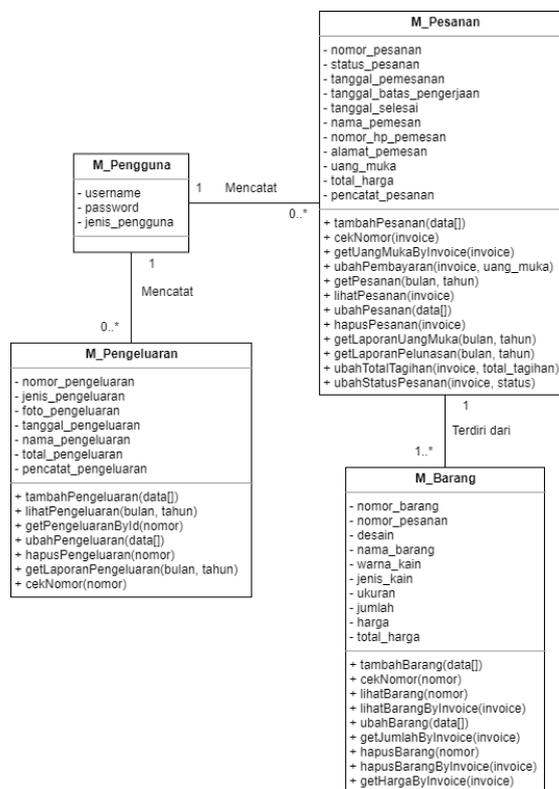


Gambar 6. sequence diagram Tambah Pesanan

*Sequence diagram* tambah pesanan menggambarkan bagaimana alur dan interaksi antar kelas ketika menjalankan *use case* tambah pesanan dengan cara menunjukkan *method* apa saja yang dipanggil dan data apa saja yang dikirimkan pada pemanggilan *method* tersebut. Alur tersebut dibuat berdasarkan *use case description* yang telah dibuat sebelumnya. langkah selanjutnya adalah membuat *class diagram* untuk menggambarkan kelas-kelas yang akan diimplementasikan pada tahap selanjutnya. Hubungan kelas *controller* (berasal dari kelas *controller*) dan kelas *model* (berasal dari kelas *entity*) dapat dilihat pada Gambar 7.

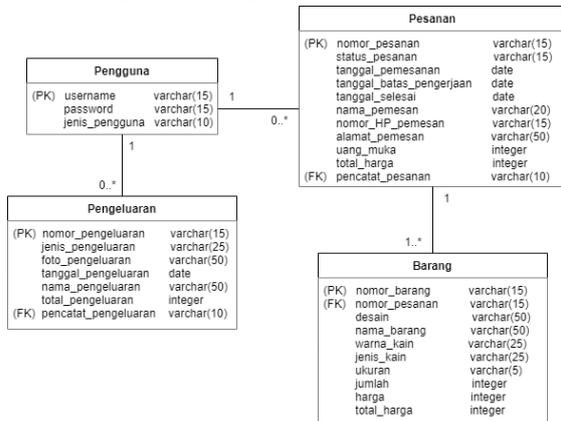


Gambar 7. Hubungan Kelas Controller dan Kelas Model



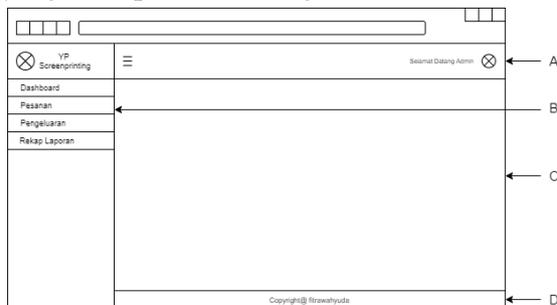
Gambar 8. Class Diagram Model

Gambar 8 merupakan *class diagram model* yang nantinya dibuat sebagai acuan dalam membuat *semantic data models* yang merupakan desain dari database yang akan dibuat. Hal itu dikarenakan pendekatan yang dipakai untuk merancang sistem pada penelitian ini adalah pendekatan OOAD. *Semantic data models* dapat dilihat pada Gambar 9.



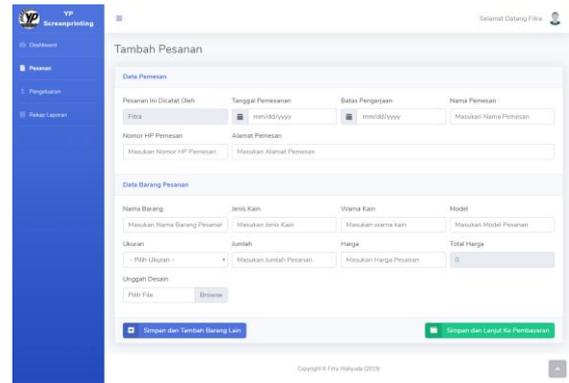
Gambar 9. *Semantic Data Models*

Setelah itu membuat desain antarmuka yang dapat dilihat pada Gambar 10. Bagian A merupakan *topbar*, bagian B merupakan *sidebar* yang berisi menu menu utama dari sistem, bagian C merupakan bagian utama dari sistem yang nantinya berisi form atau data yang ditampilkan, dan bagian D adalah *footer*.



Gambar 10. *Desain Antarmuka Sistem*

Setelah membuat rancangan sistem, selanjutnya masuk tahap *implementation* untuk mengubah rancangan yang dibuat menjadi sistem yang bisa digunakan. Implementasi pada penelitian ini menggunakan pola MVC menggunakan HTML, CSS, Javascript, dan PHP. Struktur kelas *model* dan *controller* dibuat berdasarkan *class diagram*, alur dan interaksi antara kelas *model* dan kelas *controller* dibuat berdasarkan *sequence diagram*. dan *database* dibuat berdasarkan *semantic data models*. Hasil implementasi antarmuka halaman tambah pesanan dapat dilihat pada Gambar 11.



Gambar 11. *Implementasi Antarmuka Halaman Tambah Pesanan*

Selanjutnya masuk ke tahap *system testing* atau pengujian. Pengujian yang pertama dilakukan adalah *validation testing* untuk mengetahui apakah 17 *use case* atau fitur yang telah diimplementasikan sudah berjalan sesuai harapan atau belum. Berdasarkan *Validation testing* tersebut diperoleh hasil 100% sesuai yang berarti bahwa fitur yang dibuat sudah berjalan dengan baik dan sesuai harapan. Selanjutnya sistem diberikan ke pengguna dan diuji menggunakan *user acceptance testing* (UAT) dengan cara membiarkan pengguna menggunakan sistem yang sudah dibuat secara menyeluruh, kemudian memberikan beberapa pertanyaan berupa kuesioner. Penilaian kuesioner tersebut memakai skala likert 5 poin yang terdiri dari sangat tidak setuju (1), tidak setuju (2), netral (3), setuju (4), sangat setuju (5). Hasil pengujian UAT dapat dilihat pada Tabel 6.

Tabel 6. *Hasil User Acceptance Testing (UAT)*

No	Pertanyaan	Pengguna	
		Pemilik usaha	Admin
1.	Sistem yang sudah dibuat dapat mempermudah pengelolaan data pesanan	4	4
2.	Form yang digunakan untuk memasukkan data pesanan sudah mencakup data yang selama ini dicatat secara manual	5	5
3.	Sistem yang sudah dibuat dapat mempermudah pemantauan status pengerjaan pesanan	4	4

4.	Sistem yang sudah dibuat dapat mempermudah pengelolaan data pengeluaran	4	4
5.	Form yang digunakan untuk memasukkan data pengeluaran sudah sesuai dengan kebutuhan	4	4
6.	Sistem yang sudah dibuat dapat mempercepat proses pencarian data pesanan maupun data pengeluaran	5	5
7.	Sistem yang sudah dibuat dapat mempermudah pembuatan laporan keuangan	5	5
8.	Sistem yang sudah dibuat mudah digunakan	4	5
<b>Jumlah</b>		35	36
<b>Total</b>		71	

$$\begin{aligned}
 \text{Index}(\%) &= \left( \frac{A}{B \times N} \right) \times 100\% \\
 &= \frac{71}{40 \times 2} \times 100\% \\
 &= \frac{71}{80} \times 100\% \\
 &= 88,75\%
 \end{aligned}$$

Perhitungan UAT menggunakan Persamaan (1) memperoleh hasil sebesar 88,75%. Berdasarkan kategori pada Tabel 1, hasil perhitungan UAT tersebut menunjukkan bahwa pengguna sangat setuju dengan pertanyaan yang diajukan. Sehingga bisa dikatakan bahwa secara keseluruhan sistem informasi yang dibuat diterima dengan baik oleh pengguna dan mempermudah permasalahan yang mereka hadapi.

### 5. KESIMPULAN

Untuk mempermudah pengelolaan pesanan, sistem yang sudah dibuat memiliki fitur tambah pesanan, tambah barang, ubah pembayaran uang muka, lihat daftar pesanan, lihat detail pesanan, ubah pesanan, ubah barang, hapus barang, hapus pesanan, ubah status pesanan, dan pencarian. Sedangkan untuk

mempermudah pengelolaan pengeluaran, sistem yang sudah dibuat memiliki fitur tambah pengeluaran, lihat daftar pengeluaran, lihat detail pengeluaran, ubah pengeluaran, hapus pengeluaran, dan pencarian. Fitur-fitur tersebut sudah diuji dengan *validation testing* untuk memastikan bahwa fitur-fitur tersebut berjalan dengan baik. Hasil *validation testing* dari semua fitur tersebut memperoleh hasil sesuai harapan.

Untuk mempermudah pembuatan laporan keuangan bulanan, sistem yang sudah dibuat mempunyai fitur cetak laporan agar dapat membuat laporan pemasukan dan pengeluaran bulanan dalam bentuk pdf. Fitur tersebut sudah diuji dengan *validation testing* dan memperoleh hasil sesuai harapan.

Sistem yang sudah dibuat secara umum mempermudah permasalahan yang ada di YP Screenprinting. Hal itu dibuktikan dengan hasil *user acceptance testing* (UAT) yang memperoleh hasil sebesar 88,75%.

Saran untuk penelitian selanjutnya adalah memindahkan sistem yang sudah dibuat saat ini kedalam sebuah *framework* agar mempermudah proses perawatan, proses pengecekan data yang dimasukkan pengguna dapat dikembangkan lagi agar sistem bisa langsung memberi pemberitahuan kepada pengguna apabila data yang dimasukkan tidak sesuai ketentuan, dan menambahkan beberapa fitur lagi kedepannya.

### 6. DAFTAR REFERENSI

Aknovia, 2018. *Peranan Penting Sistem Informasi Dalam Perusahaan*. [online] Tersedia di <<https://garudacyber.co.id/artikel/223-peranan-penting-sistem-informasi-dalam-perusahaan>> [diakses 25 Agustus 2019]

Eriksson, U., 2016. *The Ultimate Guide To User Acceptance Testing (UAT) Template Design*. [online] Tersedia di <<https://reqtest.com/testing-blog/ultimate-guide-to-user-acceptance-testing/>> [diakses 19 September 2019]

Rahman, M.H., Tolle, H., Dewi, R.K., 2018. *Pengembangan Sistem Informasi Pelayanan Ibu Hamil Pada Platform Android Berbasis Lokasi (Studi Kasus: Puskesmas Karangploso Kabupaten Malang)*, [e-journal] 2(11). Tersedia melalui: Jurnal Pengembangan Teknologi

Informasi dan Ilmu Komputer. <<http://j-ptiik.ub.ac.id>> [diakses 09 April 2020]

- Laar, D.S., Konjaang, J.K., Tankia, B.A., 2015. *Design and Development of a Sales Management System for SMEs in Northern Ghana*. [e-journal] 2(5). Tersedia Melalui: International Journal of Innovative Research in Advanced Engineering (IJIRAE). <<https://www.ijirae.com>> [diakses 18 September 2019]
- Laudon, K.C. & Laudon, J.P., 2014. *Management Information Systems : Managing the Digital Firm Thirteenth*. Harlow: Pearson Education Limited
- Sommerville, I., 2011. *Software engineering*. 9th ed. Boston: Pearson Education