

# Configurable Information System (CiS) Untuk Membantu UMKM Dalam Meningkatkan Fleksibilitas Sistem Penjualan dan Pembelian Dengan Tree-based Feature Model

Ellysa Tjandra <sup>1\*</sup>

<sup>1</sup> Program Studi Teknik Informatika, Universitas Surabaya, Jawa Timur  
Email: <sup>1\*</sup>ellysa@staff.ubaya.ac.id

(Naskah masuk: 9 Mar 2022, direvisi: 21 Mei 2022, 13 Jun 2022, diterima: 16 Jun 2022)

## Abstrak

*Configurable Information System (CiS)* atau yang sering disebut juga Sistem Informasi Terkonfigurasi adalah suatu sistem yang dapat diatur atau dikonfigurasi sesuai dengan kebutuhan. Dengan menerapkan sistem yang dapat dikonfigurasi maka perubahan kebutuhan organisasi akan dapat terakomodasi. Penelitian ini bertujuan menyediakan CiS yang memiliki fitur-fitur spesifik untuk penjualan pembelian dilengkapi dengan *tree-based feature model* untuk mengatur batasan sistem dengan tujuan menjaga integritas sistem, sehingga dapat dilakukan pengaturan terhadap fitur-fitur yang wajib ada (*mandatory*) dan fitur-fitur yang tidak wajib ada atau pilihan (*optional*), dengan demikian *user* dapat lebih leluasa mengatur fitur-fitur yang digunakan namun dengan batasan yang sesuai. Penelitian ini bertujuan untuk membantu organisasi berskala mikro, kecil, dan menengah (UMKM) berupa toko *retail* dan grosir dengan menyediakan CiS dengan ruang lingkup penjualan dan pembelian barang. Dengan adanya CiS maka pihak UMKM akan dapat lebih fleksibel dan leluasa mengatur fitur-fitur sesuai kebutuhan organisasi, sehingga kebutuhan organisasi dapat terakomodasi. Selain itu, dengan *tree-based feature model* maka UMKM dapat lebih mudah memilih dan mengatur fitur yang disediakan. Konfigurasi fitur yang disediakan dalam CiS ini antara lain fitur order penjualan, order pembelian, fitur pembuatan faktur, fitur konversi barang, *setting* metode pencatatan persediaan, *setting* diskon, *setting* bonus, *setting* PPN, dan *setting* metode pencatatan HPP. Ujicoba sistem CiS dilakukan dengan tiga metode, yaitu unit *testing*, *integration testing*, dan simulasi konfigurasi terhadap empat UMKM yang bergerak di bidang penjualan dan pembelian. Simulasi dilakukan dengan menjalankan skenario proses bisnis terhadap pengelola empat UMKM yang memiliki proses bisnis berbeda. Hasil proses ujicoba menunjukkan bahwa *tree-based feature model* dalam CiS yang dibuat dapat mengakomodasi kebutuhan proses bisnis yang berbeda dari keempat UMKM dengan beberapa saran pengembangan, dengan tingkat ketepatan fitur antara fitur yang diusulkan dengan fitur riil yang dibutuhkan sebesar 93,33%, sehingga dapat ditarik kesimpulan bahwa CiS yang disediakan dapat mengakomodasi beragam kebutuhan proses bisnis dalam UMKM. Dengan adanya CiS maka sistem dapat lebih fleksibel, perubahan sistem juga dapat diatur secara mandiri oleh pengguna, dan juga dapat mengurangi tingkat ketergantungan pada pengembang perangkat lunak.

**Kata Kunci:** *Configurable Information System, CiS, Feature Model, Penjualan, Pembelian, UMKM*

## *Configurable Information System (CiS) to Help MSMEs in Increasing the Flexibility of the Sales and Purchasing System using Tree-based Feature Model*

### *Abstract*

*A Configurable Information System (CiS), or what is often called a Configurable Information System, is a system that can be set or configured according to needs. By implementing a configurable system, the changing needs of the organization will be accommodated. This study aims to provide CiS, which has specific features for sale and purchase, equipped with a tree-based feature model to set system boundaries to maintain system integrity, so that mandatory and mandatory features can be arranged. With a tree-based feature model, users can more freely manage the features used but with appropriate limitations. This research aims to help micro, small and medium-scale organizations (MSMEs), including retail and wholesale stores, by providing CiS*

with the scope of selling and buying goods. With CiS, MSMEs will be able to be more flexible and free to arrange features according to the needs of the organization so that the needs of the organization can be accommodated. In addition, with the tree-based feature model, MSMEs can more easily select and manage the features provided. Configuration features provided in this CiS include sales order features, purchase orders, invoice creation features, item conversion features, inventory recording method settings, discount settings, bonus settings, VAT settings, and HPP recording method settings. The CiS system is tested using unit testing, integration testing, and configuration simulation on four MSMEs engaged in sales and purchases. The simulation is performed by running a business process scenario for the managers of four MSMEs with different business processes. The testing results show that the tree-based feature model in CiS that is made can accommodate the various business process needs of the four MSMEs. There are several development suggestions, with the level of feature accuracy between the proposed features and the real required features at 93.33%, so it can be concluded that the tree-based model provided in this research can accommodate MSMEs' business processes, thus increasing the flexibility of the system. With this CiS, system changes can also be managed independently by the user and reduce the level of dependence on software developers.

**Keywords:** Configurable Information System, CiS, Feature Model, Selling, Purchasing, MSME

## I. PENDAHULUAN

Dengan semakin berkembangnya teknologi, semakin banyak organisasi yang membutuhkan sistem berbasis komputer untuk membantu jalannya proses bisnis yang terjadi dalam organisasi. Menurut penelitian terdahulu, terdapat beberapa opsi yang dapat dipilih organisasi dalam mengembangkan sistem, yang pertama adalah dengan membeli perangkat lunak *off-the-shelf* atau perangkat lunak yang dijual secara masal, namun opsi ini memiliki keterbatasan, karena dibuat secara general, sehingga kurang dapat mengakomodasi kebutuhan organisasi secara lebih spesifik [1]. Opsi kedua yang dapat dipilih adalah dengan membangun perangkat lunak secara kustom (*customized system*), dimana organisasi meminta pengembang perangkat lunak untuk membuat sistem secara khusus untuk organisasi tersebut [2], [3]. Dengan membuat secara kustom, maka kebutuhan spesifik dalam organisasi dapat lebih terakomodasi, namun jika terdapat perubahan-perubahan pada proses bisnis dalam organisasi maka sistem yang ada tidak dapat disesuaikan oleh organisasi secara mandiri, sehingga tingkat ketergantungan pada pengembang perangkat lunak menjadi tinggi [4]–[6]. Opsi ketiga, yang dapat dipilih oleh organisasi adalah dengan membangun sebuah *Configurable System*.

*Configurable System* (CS) adalah sistem yang dapat dikonfigurasi dan diatur oleh pengguna untuk dapat memenuhi kebutuhan yang berbeda-beda [7]–[9]. Dengan menggunakan sistem terkonfigurasi maka pengguna dapat mengontrol fungsionalitas sistem sehingga dapat mengakomodasi perubahan kebutuhan yang mungkin timbul dalam organisasi [9], [10]. Dalam hal ini organisasi dapat lebih mudah menentukan dan mengatur fitur-fitur yang wajib ada dalam sistem (*mandatory feature*) dan fitur-fitur yang tidak wajib ada atau bersifat pilihan dalam sistem (*optional feature*), sehingga hal ini dapat meningkatkan fleksibilitas dari sistem [6], [9]. *Configurable System* diperlukan karena juga dapat mengurangi ketergantungan dengan pengembang *software*, karena setiap organisasi akan semakin berkembang, sehingga harus ditunjang oleh perubahan kebutuhan sistem [11], [12]

*Configurable System* yang diterapkan pada sistem informasi suatu organisasi disebut dengan istilah *Configurable Information System* (CiS) [13], [14]. Konfigurasi yang dimaksud dalam hal ini merupakan runtutan proses untuk menyesuaikan sistem informasi yang dengan kebutuhan perusahaan [1]. Dengan adanya CiS maka sistem akan bersifat fleksibel sehingga mudah untuk diubah sesuai dengan preferensi dari penggunaannya.

Beberapa penelitian terkait *Configurable System* telah dilakukan, terutama di organisasi-organisasi yang bergerak di bidang bisnis dan industri, dimana yang dikonfigurasi adalah fitur-fitur yang disediakan di dalam sistem informasi yang digunakan oleh organisasi, seperti proyek ERP [2], [14], [15] namun penelitian tersebut membahas ERP secara umum, sehingga kurang spesifik sesuai kebutuhan UMKM yang bergerak di bidang penjualan dan pembelian. Terdapat penelitian yang telah membahas secara spesifik, seperti konfigurasi pada sistem penggajian/payroll [16], [17], *bioinformatics* [5], sistem informasi penjualan pembelian [13], namun pada penelitian-penelitian ini tidak dilengkapi dengan *tree-based notation* untuk menyusun *feature model* untuk menggambarkan kategorisasi fitur (*mandatory/optional*), padahal *feature model* dalam *tree-based notation* ini sangat penting untuk menunjukkan batasan fitur mana yang boleh dan tidak boleh dipilih dalam konfigurasi sistem [5], [8].

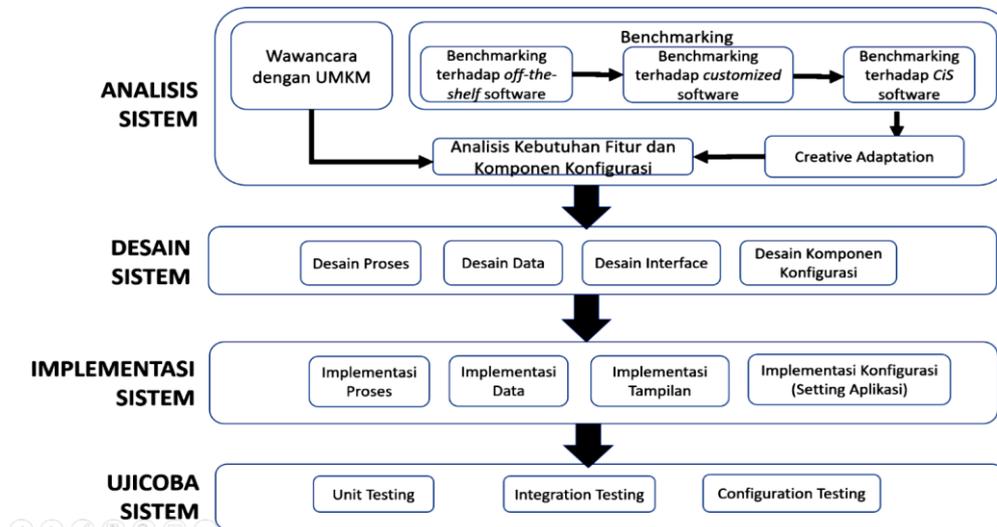
Penelitian ini bertujuan untuk menyediakan CiS yang dibuat spesifik untuk membantu organisasi berskala mikro, kecil, dan menengah (UMKM) yang bergerak di bidang usaha penjualan dan pembelian barang dengan dilengkapi *feature model* dalam bentuk *tree-based notation* sehingga dapat meningkatkan fleksibilitas sistem namun tetap memiliki batasan sehingga integritas sistem tetap terjaga. Dengan CiS maka pihak UMKM akan dapat lebih mudah mengatur fitur-fitur sesuai kebutuhan organisasi secara lebih mandiri, sehingga perubahan sistem akan menjadi lebih efisien karena dapat mengurangi tingkat ketergantungan pada pihak pengembang perangkat lunak [18].

**II. METODE PENELITIAN**

Penelitian ini dilakukan melalui beberapa tahap, menggunakan model *Waterfall* sebagai acuan. Meskipun model *Waterfall* ini sudah cukup lama ditemukan, namun dewasa ini masih banyak penelitian yang menggunakan

model ini, terutama jika semua tahapan dalam penelitian dilakukan secara berurutan dan faktor waktu pembuatan sistem tidak menjadi tujuan utama [17], [19]–[23]. Tahapan-tahapan metode penelitian ini terlihat pada Gambar 1.

Adapun tahapan yang digunakan dalam penelitian ini meliputi:



Gambar 1. Metode Penelitian

**1. Analisis Sistem**

Pada tahap analisis sistem dilakukan wawancara dengan empat UMKM di bidang penjualan dan pembelian retail bahan kebutuhan pokok untuk mengetahui kebutuhan proses bisnis yang berjalan dalam organisasi, terutama menyangkut perubahan atau penyesuaian yang mungkin terjadi dalam proses bisnis dalam organisasi. Secara paralel dilakukan pula *benchmarking* terhadap tiga perangkat lunak off-the-shelf yang cukup terkenal di kalangan UMKM, yaitu *Odoo*, *Accurate*, *MYOB*. Setelah itu dilakukan pula *benchmarking* terhadap perangkat lunak *customized* yang telah berjalan di empat UMKM, dan terakhir dilakukan *benchmarking* terhadap perangkat lunak CiS yang telah dihasilkan oleh penelitian-penelitian terdahulu. Dari hasil *benchmarking*, dilakukan *creative adaptation* untuk menentukan fitur-fitur yang akan diadopsi ke dalam CiS yang akan dibuat, termasuk di dalamnya komponen konfigurasi yang dibutuhkan.

**2. Desain Sistem**

Setelah kebutuhan fitur dan komponen konfigurasi telah dihasilkan dari tahapan analisis, dilakukan desain proses menggunakan *Business Process Model Notation* (BPMN), desain data dalam format *Entity Relationship Diagram* (ER-Diagram), desain *user interface* dalam bentuk *mock-up* tampilan aplikasi, serta desain komponen konfigurasi yang akan diakomodasi dalam CiS yang akan dibuat dalam bentuk *tree-based variability model* [17] untuk menggambarkan fitur-fitur yang wajib ada (*mandatory*) atau pilihan (*optional*).

**3. Implementasi Sistem**

Tahapan selanjutnya adalah pembuatan CiS untuk sistem penjualan dan pembelian yang dibuat berdasarkan hasil desain pada tahap sebelumnya. CiS yang dihasilkan dibuat dalam bentuk aplikasi *web*, sehingga dapat digunakan di lokasi yang berbeda-beda (terutama jika UMKM terletak di beberapa lokasi).

**4. Ujicoba Sistem**

Setelah CiS selesai dibuat, tahap selanjutnya adalah ujicoba CiS pada empat UMKM. Ujicoba sistem dilakukan terhadap modul-modul yang disediakan dalam aplikasi secara mandiri (*unit testing*), dilanjutkan dengan pengujian terhadap keseluruhan modul dalam aplikasi (*integration testing*), dengan mencoba beberapa skenario alternatif proses bisnis berbeda-beda yang mungkin terjadi dalam tiap UMKM (*configuration testing*).

**III. HASIL DAN PEMBAHASAN**

Berdasarkan hasil *benchmarking* dan wawancara yang dilakukan pada tahap analisis sistem, dihasilkan daftar kebutuhan fitur serta komponen konfigurasi seperti terlihat pada Tabel 1. Dari hasil analisis kebutuhan diperoleh total 45 fitur dalam CiS model yang digunakan dalam penelitian ini.

Jika suatu fitur/komponen konfigurasi bersifat *mandatory*, maka artinya fitur tersebut tidak dapat di-non-aktifkan dari sistem, namun jika bersifat *optional*, maka fitur tersebut dapat dipilih atau di-non-aktifkan dari sistem. Jika suatu fitur/komponen di-non-aktifkan dari sistem, maka fitur tersebut tidak akan dapat dilihat atau diakses oleh pengguna.

Untuk lebih mudah menggambarkan sifat dari tiap fitur yang ada maka perlu disusun *feature model* untuk menggambarkan fitur-fitur yang wajib ada (*mandatory*) atau pilihan (*optional*) dalam notasi *tree-based notation*. Contoh hasil desain yang

menggambarkan sifat fitur yang dibutuhkan dapat dilihat pada Gambar 2, dimana simbol  menyatakan bahwa fitur tersebut bersifat *mandatory* (tidak bisa di-non-aktifkan), dan

Tabel 1. Fitur Dan Komponen Konfigurasi Proses Bisnis Penjualan dan Pembelian pada UMKM

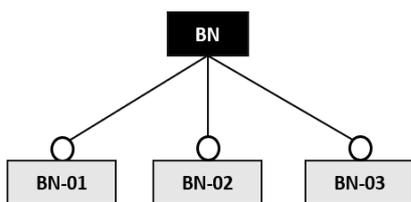
Kode	Fitur	Sifat (O: <i>Optional</i> , M: <i>Mandatory</i> )	Kode	Komponen Konfigurasi	Sifat
PO	Order Pembelian	O	PO-01	Uang muka pembelian	O
			PO-02	Jatuh tempo pengiriman pembelian	O
SO	Order Penjualan	O	SO-01	Uang muka penjualan	O
			SO-02	Jatuh tempo pengiriman penjualan	O
PN	Nota/Faktur Beli	M	PN-01	Jatuh tempo pembayaran pembelian	O
SN	Nota/Faktur Jual	M	SN-01	Jatuh tempo pembayaran pembelian	O
PR	Penerimaan Pembelian	M	PR-01	Barang diambil ke pemasok	M
			PR-02	Barang dikirim oleh pemasok	O
			PR-03	Barang dikirim oleh jasa pengiriman	O
SR	Pengiriman Penjualan	M	SR-01	Barang diambil oleh pelanggan	M
			SR-02	Barang dikirim oleh UMKM	O
			SR-03	Barang dikirim oleh jasa pengiriman	O
INV	Pencatatan Stok	M	INV-01	Kartu Stok	M
			INV-02	Surat Jalan (Keluar-Masuk Barang)	O
			INV-03	Penyesuaian ( <i>Adjustment</i> )	M
			INV-04	<i>Stock Opname</i>	M
			INV-05	<i>Perpetual</i>	O
			INV-06	Periodik	O
COGS	Perhitungan HPP	O	COGS-01	FIFO	O
			COGS-02	LIFO	O
			COGS-03	<i>Average</i>	O
PAY	Pembayaran	M	PAY-01	<i>Cash/Tunai</i>	M
			PAY-02	Transfer	O
			PAY-03	Kartu Debit	O
			PAY-04	Kartu Kredit	O
			PAY-05	Kredit - Mundur	O
			PAY-06	Kredit - Cicilan	O
TAX	PPN	O	TAX-01	PPN Pembelian	O
			TAX-02	PPN Penjualan	O
DISC	Diskon	O	DISC-01	Diskon Langsung	O
			DISC-02	Diskon Pelunasan	O
BN	Bonus	O	BN-01	<i>Cashback</i>	O
			BN-02	<i>Voucher</i>	O
			BN-03	Barang	O
CONV	Konversi Satuan Barang	O	CONV-1	Konversi Satuan Barang	O
ACC	Akuntansi	O	ACC-01	Jurnal	M
			ACC-02	<i>General Ledger</i> (Buku Besar)	M
			ACC-03	<i>General Financial Statement</i> (Laba Rugi ( <i>Income Statement</i> ), Perubahan Ekuitas ( <i>Statement of Owner's Equity</i> ), Neraca ( <i>Balance Sheet</i> ), Arus Kas ( <i>Cash Flow</i> ))	M
AM	Pengaturan Hak Akses	M	AM-01	Administrator	M
			AM-02	Pemilik	O
			AM-03	Manajer/ <i>Supervisor</i>	O
			AM-04	Pegawai Pembelian	O
			AM-05	Pegawai Penjualan	O
			AM-06	Pegawai Administrasi	O
			AM-07	Pegawai Gudang	O

simbol  menyatakan bahwa fitur tersebut bersifat *optional* (dapat di-non-aktifkan atau tidak digunakan).

Simbol  digunakan untuk membedakan fitur yang memiliki bentuk nyata (memiliki bentuk nyata berupa

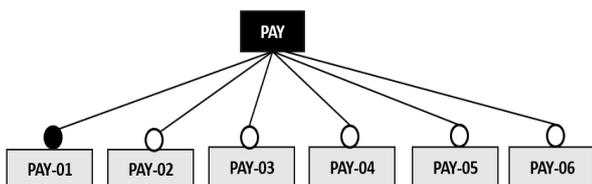
tampilan aplikasi (*concrete feature*)), sedangkan simbol  artinya fitur tersebut hanya digunakan untuk membentuk struktur atau konsep saja (tidak memiliki bentuk tampilan aplikasi (*abstract feature*)) [17]. Fitur-fitur dalam CiS model yang digunakan dalam penelitian ini terdiri dari 10 fitur *mandatory* dan 35 fitur *optional*.

Contoh *feature model* untuk fitur bonus dapat dilihat pada Gambar 2, dimana fitur bonus memiliki tiga komponen konfigurasi yang semuanya bersifat *optional*, yaitu bonus *cashback* (BN-01), *voucher* (BN-02), dan barang (BN-03). Ketiga komponen konfigurasi tersebut merupakan *concrete feature*, karena jika dikonfigurasi akan langsung berefek pada tampilan aplikasi, sedangkan fitur bonus hanya digunakan untuk mengkategorikan atau membentuk struktur *tree* secara konsep saja.



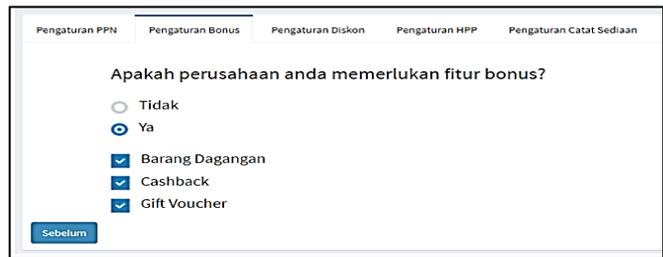
Gambar 2. Contoh *Tree-Based Variability Feature Model* Untuk Fitur Bonus

Contoh lain untuk *feature model* dapat dilihat juga pada Gambar 3, dimana di model tersebut digambarkan fitur pembayaran (PAY) yang terdiri dari enam komponen konfigurasi, yaitu *cash/tunai* (PAY-01), transfer (PAY-02), kartu debit (PAY-03), kartu kredit (PAY-04), kredit-mundur (PAY-05), dan kredit-cicilan (PAY-06). Untuk fitur pembayaran ini, komponen *cash/tunai* wajib ada dalam sistem, sehingga bersifat *mandatory*, sedangkan fitur lainnya bersifat *optional*.

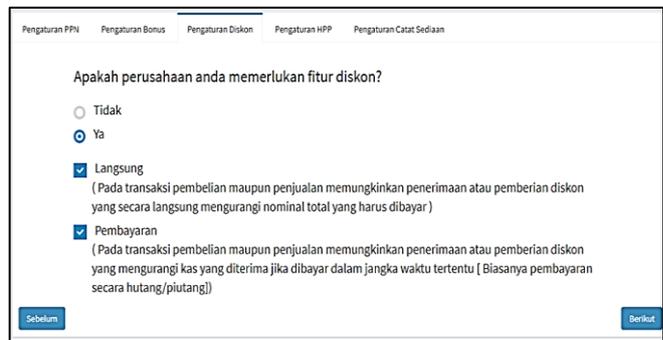


Gambar 3. Contoh *Tree-Based Variability Feature Model* Untuk Fitur Pembayaran

Pertama-tama ujicoba dilakukan dengan mekanisme unit *testing* terhadap tiap modul yang memuat fitur yang disediakan secara independen, sesuai daftar fitur yang tercantum pada Tabel 1, kemudian dilanjutkan dengan *integration testing* untuk menguji relasi antar fitur. Setelah *unit testing* dan *integration testing* selesai dilakukan dan hasilnya telah sesuai, maka selanjutnya dilakukan pengujian terhadap konfigurasi komponen terhadap tiap fitur yang disediakan dalam sistem. Contoh pengujian terhadap konfigurasi komponen, yaitu konfigurasi bonus, terlihat pada Gambar 4, sedangkan contoh konfigurasi terhadap komponen diskon terlihat pada Gambar 5.

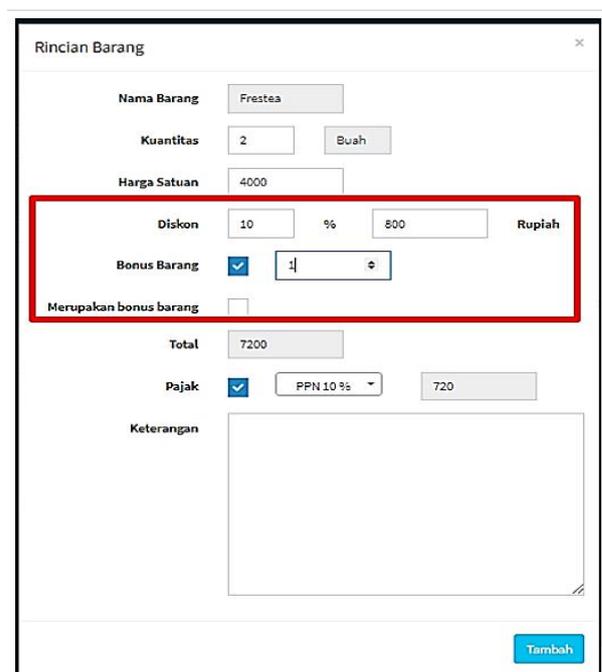


Gambar 4. Contoh Konfigurasi Komponen Bonus [24]



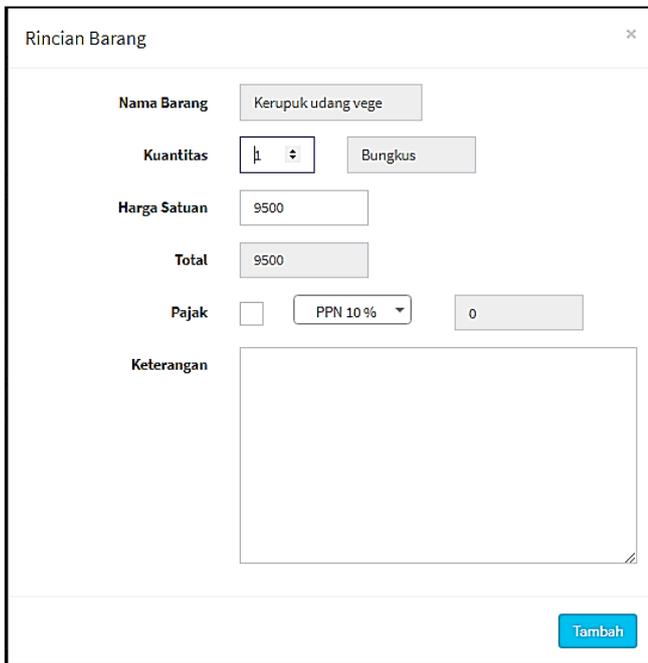
Gambar 5. Contoh Konfigurasi Komponen Diskon [24]

Setelah dilakukan konfigurasi terhadap komponen fitur yang ada, maka perbedaan hasil konfigurasi jika pengguna memilih komponen bonus dan diskon akan terlihat pada tampilan aplikasi (lihat Gambar 6) [24], dimana terdapat inputan terkait bonus dan diskon. Jika *user* memutuskan tidak memilih komponen bonus dan diskon maka masukan terkait bonus dan diskon tidak akan muncul pada tampilan aplikasi, seperti terlihat pada Gambar 7.



Gambar 6. Contoh Tampilan Hasil Konfigurasi (Menggunakan Bonus Dan Diskon) [24]

Untuk memastikan bahwa konfigurasi fitur-fitur yang disediakan dalam sistem telah sesuai maka dilakukan simulasi konfigurasi fitur terhadap pemilik sekaligus pengelola empat UMKM yang bergerak di bidang penjualan dan pembelian, yaitu toko *retail* dan grosir barang kebutuhan pokok, makanan beku, dan *fashion*. Konfigurasi fitur dilakukan oleh pengelola/pemilik UMKM karena pengelola/pemilik menguasai dan menentukan semua proses bisnis yang berjalan dalam UMKM.



Gambar 7. Contoh Tampilan Hasil Konfigurasi (Tanpa Bonus Dan Diskon) [24]

Dalam proses ujicoba ini tiap pemilik/pengelola UMKM melakukan konfigurasi fitur yang disediakan dalam sistem sesuai dengan kebutuhan dan proses yang berjalan di tiap UMKM. Daftar UMKM yang diujicoba dapat dilihat pada Tabel 2.

Tabel 2. Profil UMKM

UMKM	Bidang Usaha	Jumlah Karyawan
UMKM-1	Retail Barang Kebutuhan Pokok	4
UMKM-2	Retail Aksesoris Fashion	9
UMKM-3	Grosir & Retail Frozen Food	11
UMKM-4	Grosir Barang Kebutuhan Pokok	21

Hasil simulasi dari 45 fitur yang disediakan diujikan kepada UMKM dan hasilnya dapat dilihat pada Tabel 3. Simbol  $\surd$  menunjukkan bahwa fitur telah berjalan dengan baik, sedangkan Simbol  $\times$  menunjukkan bahwa terdapat kekurangan/perluannya tambahan/perbaikan pada fitur, dan

simbol - menunjukkan bahwa fitur tidak dibutuhkan oleh UMKM.

Dari hasil ujicoba pada Tabel 3 ditemukan beberapa temuan dan masukan, antara lain:

- Dari 45 fitur yang disusun dalam model, terdapat tiga kategorisasi fitur yang kurang tepat, yaitu fitur Kartu Stok, Penyesuaian, *Stock Opname* yang dimasukkan dalam kategori M (*Mandatory*) namun ternyata ada UMKM yang tidak menggunakan fitur tersebut, sehingga harusnya dikategorikan sebagai O (*Optional*), sehingga dapat disimpulkan persentase ketepatan fitur antara fitur yang diusulkan dengan fitur riil yang dibutuhkan sebesar 93,33%.
- Fitur laporan dan jenis hak akses pengguna sebaiknya juga dapat dikonfigurasi sesuai kebutuhan spesifik UMKM (saat ini fitur tersebut telah disediakan namun belum dapat dikonfigurasi).
- Terdapat UMKM yang menganggap fitur pembayaran berupa transfer kurang lengkap, harusnya dilengkapi dengan metode pembayaran dalam bentuk dompet digital (seperti *GoPay*, *Shopee Pay*, *Ovo*, dan lainnya)
- Terdapat UMKM yang berencana memproduksi sendiri barang dagangannya, namun saat ini belum dapat diakomodasi oleh sistem karena belum tersedia modul-modul manufaktur, sehingga menyarankan untuk menambah sistem dengan modul manufaktur.

Dari hasil ini dapat disimpulkan bahwa persentase ketepatan antara fitur yang direncanakan dengan fitur riil yang dibutuhkan sebesar 93,33%, sehingga dapat disimpulkan bahwa CiS yang dibuat dengan *tree-based feature model* dalam penelitian ini telah berhasil memfasilitasi empat UMKM untuk dapat menggunakan aplikasi serta mengkonfigurasi komponen fitur yang ada secara mandiri tanpa bantuan pengembang aplikasi. Dengan menggunakan CiS ini maka UMKM dapat mengurangi ketergantungan pada pihak pengembang aplikasi, dan dapat meningkatkan fleksibilitas sistem, sehingga dapat mengakomodasi perubahan kebutuhan yang terjadi dalam UMKM sesuai preferensi masing-masing UMKM.

#### IV. KESIMPULAN

CiS untuk pembelian penjualan yang dihasilkan oleh penelitian ini telah dapat memfasilitasi beberapa UMKM dalam melakukan konfigurasi secara mandiri pada sistem pembelian dan penjualan berdasarkan proses bisnis yang berlangsung dalam UMKM, serta dapat dengan mudah diatur sesuai perubahan kebutuhan proses bisnis dalam UMKM, dengan persentase ketepatan fitur sebesar 93.33%, sehingga CiS dapat membantu UMKM dalam meningkatkan fleksibilitas penggunaan sistem dan mengurangi tingkat ketergantungan pada pihak pengembang aplikasi. Sistem juga telah dapat digunakan sesuai kebutuhan atau preferensi setiap UMKM yang berbeda-beda, namun penentuan kategori fitur (*Optional/Mandatory*) harus dikaji lebih lanjut dengan lebih banyak melibatkan jumlah dan jenis UMKM agar lebih sesuai dengan kebutuhan UMKM yang berbeda-beda.

Untuk pengembangan ke depannya, penelitian mengenai CiS dapat diperluas dengan menambahkan fitur-fitur yang saat ini belum dapat dikonfigurasi namun diperlukan UMKM, yaitu fitur laporan dan jenis hak akses pengguna, penambahan

metode pembayaran dalam bentuk dompet digital, dan juga tidak tertutup kemungkinan bisa dikembangkan lagi dengan penambahan modul-modul lain seperti manufaktur, jasa, dan lainnya.

Tabel 3. Hasil Ujicoba Simulasi Konfigurasi Fitur

Kode	Komponen Konfigurasi	Sifat	UMKM-1	UMKM-2	UMKM-3	UMKM-4
PO-01	Uang muka pembelian	O	-	√	√	√
PO-02	Jatuh tempo pengiriman pembelian	O	-	√	√	√
SO-01	Uang muka penjualan	O	-	-	√	√
SO-02	Jatuh tempo pengiriman penjualan	O	-	-	√	√
PN-01	Jatuh tempo pembayaran pembelian	O	-	√	√	√
SN-01	Jatuh tempo pembayaran penjualan	O	-	-	√	√
PR-01	Barang diambil ke pemasok	M	√	√	√	-
PR-02	Barang dikirim oleh pemasok	O	√	√	√	√
PR-03	Barang dikirim oleh jasa pengiriman	O	-	√	√	√
SR-01	Barang diambil oleh pelanggan	M	√	√	√	√
SR-02	Barang dikirim oleh UMKM	O	-	√	√	√
SR-03	Barang dikirim oleh jasa pengiriman	O	-	√	√	√
<b>INV-01</b>	<b>Kartu Stok</b>	<b>M</b>	<b>-</b>	<b>√</b>	<b>√</b>	<b>√</b>
INV-02	Surat Jalan (Keluar-Masuk Barang)	O	-	√	√	√
<b>INV-03</b>	<b>Penyesuaian (Adjustment)</b>	<b>M</b>	<b>-</b>	<b>√</b>	<b>√</b>	<b>√</b>
<b>INV-04</b>	<b>Stock Opname</b>	<b>M</b>	<b>-</b>	<b>√</b>	<b>√</b>	<b>√</b>
INV-05	Perpetual	O	-	√	√	√
INV-06	Periodik	O	-	-	√	√
COGS-01	FIFO	O	-	-	-	√
COGS-02	LIFO	O	-	-	-	-
COGS-03	Average	O	-	√	√	√
PAY-01	Cash/Tunai	M	√	√	√	√
<b>PAY-02</b>	<b>Transfer</b>	<b>O</b>	<b>√</b>	<b>√</b>	<b>X</b>	<b>X</b>
PAY-03	Kartu Debit	O	-	√	√	√
PAY-04	Kartu Kredit	O	-	√	√	√
PAY-05	Kredit - Mundur	O	-	-	√	√
PAY-06	Kredit - Cicilan	O	-	-	-	√
TAX-01	PPN Pembelian	O	-	-	√	√
TAX-02	PPN Penjualan	O	-	-	√	√
DISC-01	Diskon Langsung	O	√	√	√	√
DISC-02	Diskon Pelunasan	O	-	-	√	-
BN-01	Cashback	O	-	-	√	-
BN-02	Voucher	O	-	√	-	-
BN-03	Barang	O	√	-	√	√
CONV-1	Konversi Satuan Barang	O	√	-	√	√
ACC-01	Jurnal	M	-	√	√	√
ACC-02	General Ledger (Buku Besar)	M	-	√	√	√
ACC-03	General Financial Statement	M	-	√	√	√
AM-01	Administrator	M	√	√	√	√
AM-02	Pemilik	O	√	√	-	√
AM-03	Manajer/Supervisor	O	-	-	√	-
AM-04	Pegawai Pembelian	O	-	-	√	√
AM-05	Pegawai Penjualan	O	-	-	√	√
AM-06	Pegawai Administrasi	O	-	√	√	√

REFERENSI

[1] P. Soffer, B. Golany, and D. Dori, "ERP modeling: A comprehensive approach," *Information Systems*, vol. 28, no. 6, pp. 673–690, 2003.  
 [2] J. Wolters, U. Y. Eseryel, and D. Eseryel, "Identifying the critical success factors for low customized ERP

system implementations in smes," *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.  
 [3] T. Randall, C. Terwiesch, and K. T. Ulrich, "Principles for user design of customized products," *California Management Review*, vol. 47, no. 4, pp. 68–85, 2005.

- [4] S. Mühlbauer, S. Apel, and N. Siegmund, "Identifying software performance changes across variants and versions," *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, 2020.
- [5] M. Cashman, M. B. Cohen, P. Ranjan, and R. W. Cottingham, "Navigating the maze: The impact of configurability in bioinformatics software," *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*, 2018.
- [6] L. Luthmann, "Specification and Analysis of Software Systems with Configurable Real-Time Behavior," thesis, Darmstadt, 2020.
- [7] J. Dorn, S. Apel, and N. Siegmund, "Mastering uncertainty in performance estimations of Configurable Software Systems," *Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering*, 2020.
- [8] X. Qu, "Testing of configurable systems," *Advances in Computers*, pp. 141–162, 2013.
- [9] S. Zhang and M. D. Ernst, "Which configuration option should I change?," *Proceedings of the 36th International Conference on Software Engineering*, 2014.
- [10] Y. Rogozov, J. Lipko, and O. Schevchenko, "The control in the structure of Configurable information system," *2014 IEEE 8th International Conference on Application of Information and Communication Technologies (AICT)*, 2014.
- [11] S. Nadi, T. Berger, C. Kastner, and K. Czarnecki, "Where do configuration constraints stem from? an extraction approach and an empirical study," *IEEE Transactions on Software Engineering*, vol. 41, no. 8, pp. 820–841, 2015.
- [12] S. W. Zhang and Z. H. Li, "A configurable platform of Application System and its end-user-oriented configuration-developing pattern," *Advanced Materials Research*, vol. 219-220, pp. 1415–1418, 2011.
- [13] A. Q. Gill and E. Chew, "Configuration information system architecture: Insights from Applied Action Design Research," *Information & Management*, vol. 56, no. 4, pp. 507–525, 2019.
- [14] M. Pozzebon and A. Pinsonneault, "Implementation of Configurable Information Systems: Negotiations between Global Principles and Local Contexts," *International Conference on Information Systems (ICIS) 2004 Proceedings*, 2004.
- [15] A. von Rhein, "Analysis Strategies for Configurable Systems," thesis, Universität Passau, Passau, 2016.
- [16] S. Limanto, Ellysa Tjandra, and Dionisius Dwi Putra, "Pembuatan configurable payroll software Untuk Meningkatkan Keleluasaan Saat Pengembangan Sistem penggajian," *Teknika*, vol. 10, no. 1, pp. 8–17, 2021.
- [17] Ricki Sastra, Numan Musyaffa, and Bayu Supriadi, "Perancangan Sistem informasi penggajian menggunakan model waterfall pada pt. Medina," *Jurnal Teknologi Informasi Universitas Lambung Mangkurat (JTIULM)*, vol. 4, no. 2, pp. 71–78, 2019.
- [18] W. M. van der Aalst, A. Dreiling, F. Gottschalk, M. Rosemann, and M. H. Jansen-Vullers, "Configurable process models as a basis for reference modeling," *Business Process Management Workshops*, pp. 512–518, 2006.
- [19] M. A. Cusumano and S. A. Smith, "Beyond the Waterfall: Software Development at Microsoft," *Compet. Age Digit. Converg.*, pp. 371–411, 1997.
- [20] G. W. Sasmito, "Penerapan Metode Waterfall Pada Desain Sistem Informasi Geografis Industri Kabupaten Tegal," *JURNAL INFORMATIKA : JURNAL PENGEMBANGAN IT*, vol. 2, no. 1, pp. 6–12, 2017.
- [21] N. Hidayati and S. Sismadi, "Application of waterfall model in development of Work Training Acceptance System," *INTENSIF: Jurnal Ilmiah Penelitian dan Penerapan Teknologi Sistem Informasi*, vol. 4, no. 1, pp. 75–89, 2020.
- [22] Y. Handrianto and B. Sanjaya, "Model waterfall dalam Rancang Bangun sistem informasi Pemesanan Produk Dan Outlet Berbasis web," *Jurnal Inovasi Informatika*, vol. 5, no. 2, pp. 153–160, 2020.
- [23] O. Irnawati and I. Darwati, "Penerapan Model Waterfall Dalam Analisis Perancangan sistem informasi inventarisasi berbasis web," *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, vol. 6, no. 2, pp. 109–116, 2020.
- [24] Eron, "Pembuatan Configurable Enterprise System untuk Penjualan Pembelian," thesis, 2019.