Jurnal MOSFET
Vol. 06 No. 02, Juli – Desember 2026, pp. 75-85, eISSN: 2775-5274
http://jurnal.umpar.ac.id/index.php/jmosfet
DOI: https://doi.org/10.31850/jmosfet.v6i2.4511

Merancang *Cardinality-Based Feature Model* and *Orthogonal Variability Model* menggunakan Pendekatan Rekayasa Lini Produk Perangkat Lunak

Ahmad Anshari[1*], A. Muhammad Syafar[2]

[1,2]*Teknik Informatika, Universitas Islam Negeri Alauddin Makassar, Indonesia*
**Email : ahmad.anshari@uin-alauddin.ac.id*

Abstract: Variability is the ability of a software system or artifact to be changed, adapted, or configured for reuse in SPLE products. As the amount of variability increases in SPLE, the complexity of managing changes and the evolution of that variability becomes a primary concern of this case study. Software designers seek simple and specialized methods in designing software to make it easily understood and implemented in its development, especially in large-scale productivity design. This study uses the Orthogonal Variability Model (OVM) and the Cardinality-based Feature Model (CBFM) for the analysis and design of the MANRE.ID application system and identifies the basic requirements for building the MANRE.ID application and then uses OVM to establish relationships and identify them with each other. In this study, the researchers propose that the integration of models between CBFM and OVM results in easier and more structured development of MANRE.ID.

Keywords: Software Product Line Engineering; Cardinality-based Feature Model; Orthogonal Variability Model

1. PENDAHULUAN

Software product line engineering (SPLE) telah terbukti memberdayakan industri untuk mengembangkan keragaman sistem serupa dengan biaya lebih rendah, dalam waktu yang lebih singkat, dan dengan kualitas yang lebih tinggi jika dibandingkan dengan pengembangan sistem tunggal (Pohl & Metzger, 2018). Lini produk perangkat lunak (juga kadang-kadang disebut keluarga produk perangkat lunak) adalah "sekumpulan sistem intensif perangkat lunak yang berbagi sekumpulan fitur yang dikelola dan umum yang memenuhi kebutuhan spesifik segmen pasar atau misi tertentu dan yang dikembangkan dari kesamaan kumpulan aset inti [artefak] dengan cara yang ditentukan" (Pohl & Metzger, 2014). *Feature Model* (FM) yang pertama kali diperkenalkan oleh Kang dengan *Feature Oriented Domain Analysis* (FODA) menjadi bagian inti dari penelitian dan pengembangan SPLE. FM telah diperluas dengan beberapa pendekatan, salah satunya adalah *Cardinality-based Feature Model* (CBFM). Pendekatan lain selain penggunaan notasi FM untuk menyatakan variabilitas adalah *Ortogonal Variability Model* (OVM) (Wahyudianto et.al, 2014).

Manre merupakan aplikasi berbasis mobile yang berjalan di dua perangkat yaitu Android dan iOS, yang menjadi wadah penjualan produk makanan dan minuman beserta sebagai tempat berekspresi para pemasak yang suka membagikan resepnya. Dalam pembuatan

aplikasi ini sangat membutuhkan banyak fitur, sehingga perlunya menentukan sebuah fitur-fitur menjadi point utama dalam pengembangan kedepannya. Maka dari itu disini peneliti melakukan Analisa untuk menarik sebuah model garis untuk menentukan dan mencari fitur-fitur yang akan digunakan dan juga akan lebih baik bagi developer untuk melakukan build aplikasi manre ini.

Dalam SPLE, pemodelan lini produk menjadi perhatian utama dengan representasi variabilitas sistem. Salah satu yang paling luas metode pemodelan yang digunakan adalah pemodelan variabilitas (M. Li et.al., 2019). Variabilitas memainkan peran kunci dalam membedakan anggota produk, kompleksitas mengelola peningkatan jumlah variabilitas tersebut menjadi perhatian utama yang perlu ditangani (Lee et.al., 2016). Teknik representasi variabilitas memungkinkan pemodelan variabilitas sistem. Namun, teknik ini sering kali mengandalkan pengalaman para ahli dan membutuhkan upaya besar di muka dari manusia untuk mengembangkan model lini produk melalui identifikasi manual variabilitas (J. Chavarriaga et.al., 2019).

Variabilitas lini produk adalah kuncinya, perhatian lintas sektor di SPLE. Variabilitas lini produk menentukan bagaimana aplikasi lini produk perangkat lunak dapat berbeda, misalnya, dalam hal properti, fitur yang ditawarkan, fungsi, atau kualitas yang ditawarkan. Apakah properti yang diberikan adalah variable umum) atau variabel untuk aplikasi lini produk perangkat lunak ditentukan oleh keputusan manajemen eksplisit, biasanya dibuat oleh manajemen produk. Variabilitas lini produk didokumentasikan dalam apa yang disebut model variabilitas. Manfaat SPLE yang dilaporkan termasuk peningkatan produktivitas sebanyak faktor 10, peningkatan kualitas sebanyak faktor 10, penurunan biaya sebanyak 60%, penurunan kebutuhan tenaga kerja sebanyak 87%, penurunan waktu ke pasar sebanyak 98%, dan kemampuan untuk pindah ke pasar baru dalam beberapa bulan, bukan tahun (Pohl & Metzger, 2018).

Cardinality-based feature model (CBFM) adalah model fitur hierarkis, di mana setiap fitur memiliki kardiak (Horcas & Pinto, 2020). Notasi kardinalitas kenampakan medan adalah [m..n], dengan interval dari m sampai n, dimana $m \in Z \wedge n \in Z \cup \{*\} \wedge 0 \leq m \wedge (m \leq n \vee n = *)$. Notasi tambahan lainnya adalah kardinalitas grup untuk setiap grup fitur, di mana fitur dapat diatur. Kardinalitas suatu kelompok mempunyai interval dari m ke n dengan notasi [m-n], dimana $m, n \in Z \wedge 0 \leq m \leq n \leq k$, dan dimana k adalah banyaknya kenampakan medan dalam kelompok (Wahyudianto et.al, 2014). Beberapa penulis mengusulkan untuk memperluas model fitur dasar dengan multiplikasi UML seperti bentuk [n, m] dengan n sebagai batas bawah dan m batas atas. Ini digunakan untuk membatasi jumlah sub-fitur yang dapat menjadi bagian dari produk setiap kali induk dipilih (Czarnecki et.al., 2004). Jika batas atasnya adalah m, fitur tersebut dapat dikloning sebanyak yang kita inginkan (selama batasan lainnya dipatuhi). Notasi ini berguna untuk produk yang dapat diperluas dengan jumlah komponen yang berubah-ubah. *Cardinality-based feature model* (CBFM) telah diusulkan dengan ekstensi notasional FODA dan FORM, dengan kardinalitas yang ditentukan dari setiap fitur yang dengan jelas menyajikan dependensi, hubungan, dan titik variasi model fitur SPL (A. Abbas et.al., 2017).

Untuk mendukung dokumentasi ortogonal variabilitas lini produk, variabilitas lini produk didokumentasikan dalam model khusus (Pohl & Metzger, 2014). Dokumentasi variabilitas lini produk dipisahkan dari dokumentasi artefak pengembangan perangkat lunak. Dengan demikian, variabilitas lini produk diperlakukan sebagai artefak lini produk kelas satu (Pohl & Metzger, 2014). Menghubungkan variabilitas lini produk yang didefinisikan dalam model variabilitas ortogonal dengan artefak perangkat lunak yang didefinisikan dalam model artefak, realisasi variabilitas lini produk dalam artefak perangkat lunak didokumentasikan (Marcello et.al., 2017). OVM berfokus pada dokumentasi variabilitas dalam model variabilitas terpisah daripada mengintegrasikan variabilitas secara langsung menjadi artefak pengembangan (Aitor, 2019).

Saat ini, model keputusan dan model fitur juga terutama digunakan dengan cara orthogonal. Variabilitas yang ditentukan dalam OVM terkait dengan artefak lain seperti model fitur, model kasus penggunaan, dan komponen model (Rabih et.al., 2017). OVM menggunakan titik variasi dan varian untuk menangkap variabilitas di antara sekumpulan produk. Mirip dengan model keputusan, OVM berfokus pada yang didokumentasikan variabilitas. Bahasa batasan digunakan untuk menentukan batasan variabilitas (Roos-Frantz et.al., 2011). Dalam OVM, batasan antar node didefinisikan secara grafis. Batasan dapat ditentukan antara Vs, VP, dan Vs dan VP dan mungkin merupakan pengecualian batasan atau batasan kebutuhan. Batasan pengecualian menentukan pengecualian bersama, misalnya, varian mengecualikan opsional VP berarti bahwa jika varian dipilih untuk produk tertentu, VP tidak boleh terikat, dan sebaliknya (Frantz & Segura, 2008).

Meskipun Cardinality-Based Feature Model (CBFM) dan Orthogonal Variability Model (OVM) telah banyak digunakan dalam Software Product Line Engineering (SPLE), sebagian besar penelitian masih berfokus pada penggunaan masing-masing model secara terpisah untuk merepresentasikan fitur maupun variabilitas sistem. Akibatnya, dokumentasi hubungan antara struktur fitur, *variation point*, dan *dependency* belum disajikan secara terpadu sehingga proses penelusuran kebutuhan dan konfigurasi produk menjadi kurang sistematis, khususnya pada aplikasi dengan jumlah fitur yang besar.

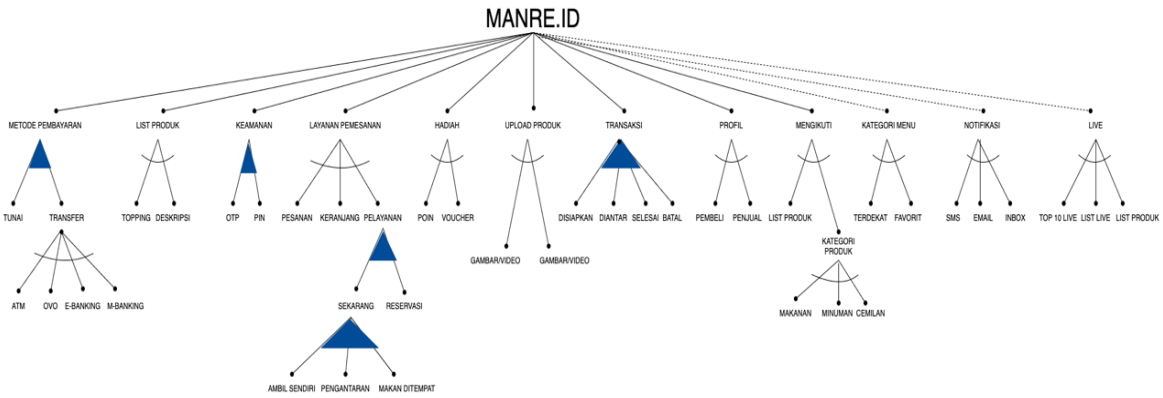
Berdasarkan permasalahan tersebut, penelitian ini mengusulkan suatu alur pemodelan yang mengintegrasikan *Cardinality-Based Feature Model* (CBFM), *Orthogonal Variability Model* (OVM), dan pemodelan dependency dalam proses domain engineering aplikasi Manre.id. Kontribusi penelitian ini tidak terletak pada pengembangan metode baru, melainkan pada penyusunan *workflow* pemodelan yang sistematis untuk mentransformasikan kebutuhan domain menjadi dokumentasi variabilitas yang terstruktur. Workflow tersebut diharapkan dapat memudahkan identifikasi hubungan antarfitur, meningkatkan keterlacakan kebutuhan, serta mendukung proses pengembangan aplikasi berbasis *Software Product Line Engineering*.

2. METODOLOGI PENELITIAN

Pada fase ini, peneliti mendeskripsikan dan mengimplementasikan diagram OVM yang menggambarkan detail yang terbentuk untuk studi kasus manre. OVM menggambarkan kasus ini terbentuk sebagai proses lebih dari model fitur. Saat peneliti merancang dan membangun semua persyaratan domain diagram dan menyertakannya dengan CBFM.

2.1. CBFM untuk Manre.id

Untuk memberikan visi yang komprehensif dan luas untuk fitur dan keterkaitan antara hubungan dalam teknologi (SPLE) dan untuk produksi (Produk Massal) untuk produk apa pun, Anda harus menggunakan (CBFM) untuk menunjukkan visi keseluruhan untuk produk ini dan melalui penggunaan hubungan timbal balik antara karakteristik yang diusulkan, kita tahu bahwa hubungan ini wajib atau opsional atau alternatif, menggunakan model ini (CBFM) akan memiliki visi yang jelas melalui kerja tim (Marcello et.al., 2017). Dalam model yang diusulkan manre peneliti menemukan satu set fitur wajib, alternatif dan opsional. Jumlah fitur mencapai 52 fitur beragam termasuk dalam model pembangunan sistem manre dan gambar 1 di bawah ini menggambarkan fitur dan hubungan, peneliti menemukan hubungan antara sistem dengan kategori menu, notifikasi, dan live yang bersifat opsional. Untuk fitur yang lain, sistem memiliki hubungan yang bersifat mandatory sehingga alurnya bersifat wajib.

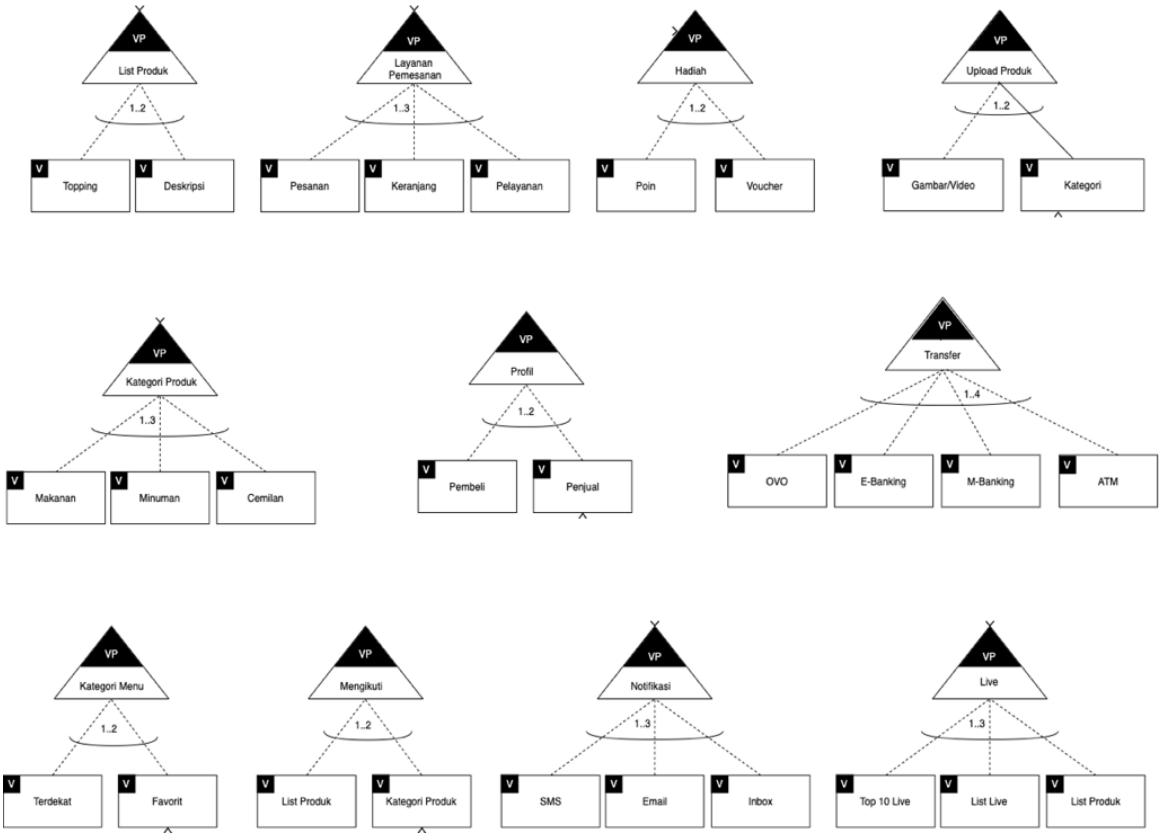


Gambar 1. CBFM pada Manre.id

CBFM merupakan representasi visual dalam bentuk diagram yang sangat banyak digunakan dalam (SPLE) dan input untuk menghasilkan aset yang baik dapat berupa produk lain serta dapat ditentukan Portofolio yang juga memberikan dokumentasi yang baik dan deskripsi desain arsitektur dan definisi masalah yang komprehensif dan memberikan visi tentang bagaimana menyelesaikannya dan kemungkinan menjelaskan bagaimana membangun kode atau bagian-bagiannya juga merupakan bagian utama yang membantu memberikan pandangan yang komprehensif persyaratan produk (Marcello et.al., 2017).

2.2. OVM untuk Manre.id

Orthogonal Variability Modeling (OVM) merupakan salah satu pendekatan untuk memodelkan variabilitas pada lini produk perangkat lunak (Roos-Frantz et.al., 2011). Diagram OVM berisi banyak kesederhanaan seperti bentuk *Variation Point* (VP) yang merepresentasikan fitur variasi dalam *Feature Model* juga berisi Varian sederhana (V). Langkah pertama adalah mendapatkan OVM dari CBFM tertentu. CBFM berisi fitur umum dan variabel dari SPLE. Fitur umum adalah fitur yang menjadi bagian dari semua produk di SPLE, dan fitur variabel adalah fitur yang menjadi bagian dari beberapa produk. Namun, OVM hanya mendokumentasikan variabilitas SPL. Oleh karena itu, untuk mengubah FM menjadi OVM, setiap variabilitas yang direpresentasikan dalam FM harus diubah menjadi titik variasi dalam OVM (Li et.al, 2015).

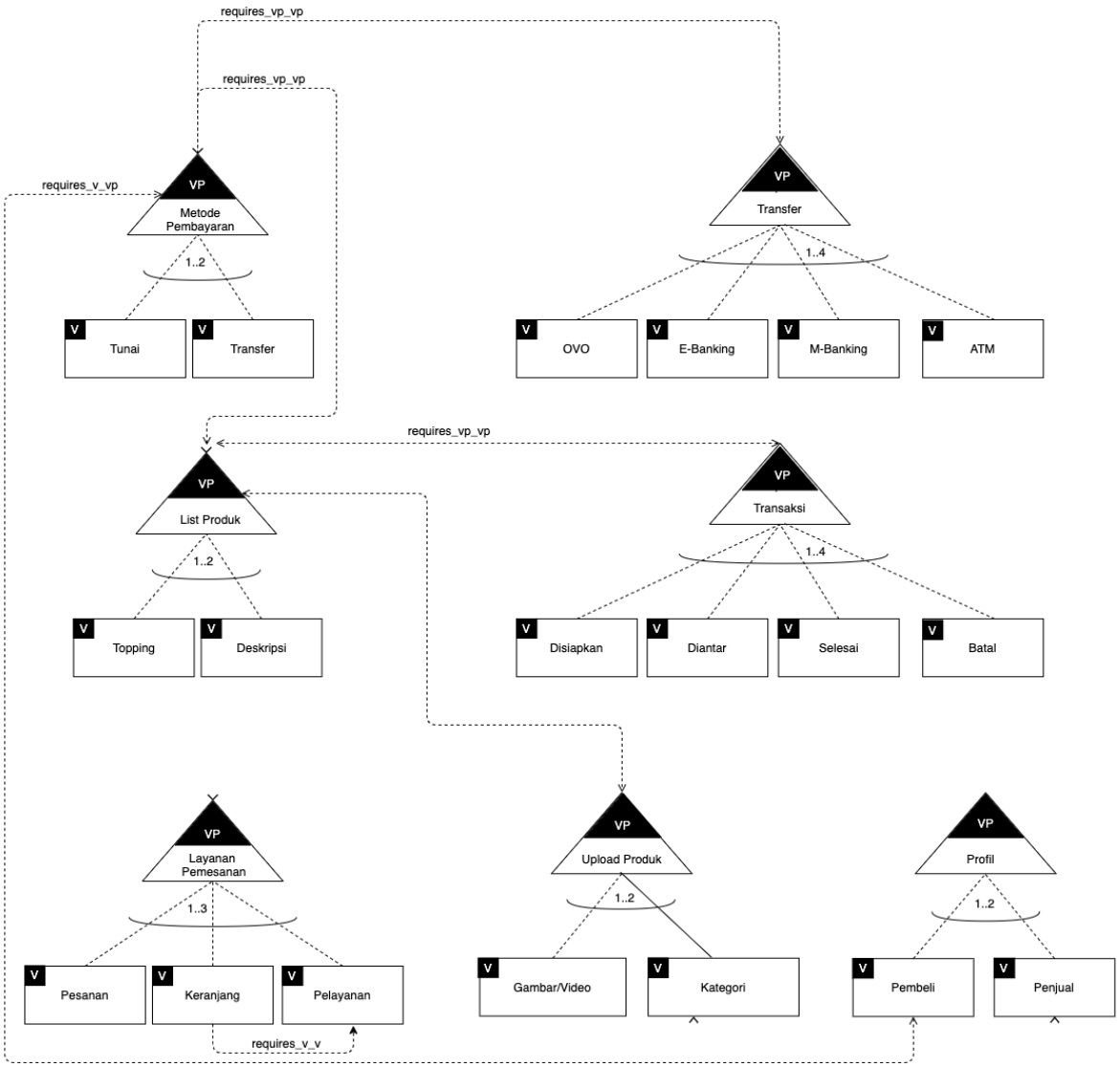


Gambar 2. OVM pada Manre.id

Gambar 2 mewakili Model Variabilitas Ortogonal untuk sistem aplikasi manre.id yang diusulkan. Angka tersebut berisi OVM untuk fitur utama yang terdiri dari model sistem aplikasi manre.id. Seperti yang kita lihat pada gambar ini misalnya OVM untuk fitur Kategori Produk berisi kategori produk sebagai *Variation Point* (VP) yang memiliki varian (V) makanan, minuman, dan cemilan. Hubungan diantara mereka adalah suatu keharusan dan hubungan antara VP dan V adalah hubungan XOR.

2.3. *Dependencies*

Dependencies antar fitur dapat diklasifikasikan menjadi dependensi statis dan dinamis. Dependensi statis mencerminkan hubungan fitur hierarki dan batasan statis di antara fitur di level yang sama. Dependensi statis mencakup dekomposisi, generalisasi, dan batasan statis. Batasan statis mencakup wajib dan dikecualikan, yang mencerminkan ketergantungan antara fitur sesama, terutama varian yang berbeda dalam satu titik variabel (Aitor, 2019)



Gambar 3. *Dependencies* pada Manre.id

Gambar 3 mewakili ketergantungan antara Metode Pembayaran (VP) dengan Transfer (VP) sebagai (VP-VP yang diperlukan), Pembeli (V) memiliki ketergantungan dengan metode pembayaran (VP) sehingga hubungan fitur kedua tersebut adalah (VP-VP).

Untuk hubungan yang memiliki ketergantungan variant dengan variant dapat dilihat pada keranjang dengan pelayanan (V-V).

3. HASIL DAN PEMBAHASAN

Perkembangan teknologi telah membuat masyarakat menghadapi banyak tantangan, khususnya di bidang bisnis (Marcello et.al., 2017). Di sisi lain, peneliti sedang melakukan studi untuk mengetahui inovasi baru yang sesuai dengan perkembangan masyarakat informasi (Marcello et.al., 2017). Penelitian ini telah merangkum studi tentang membangun model untuk sistem aplikasi manre.id yang sesuai dengan lingkungan, kebutuhan dan kebutuhan pasar berdasarkan metode SPLE. Proses planning yang dilakukan sebelum adanya usulan penggabungan model CBFM dengan OVM adakah pengembang aplikasi melakukan planning dengan cara manual dimana proses bisnis yang dilakukan tidak mengadopsi standarisasi dalam pengembangan rekayasa perangkat lunak. Dengan dihasilkannya model CBFM dan OVM untuk sistem aplikasi manre.id, pemangku kepentingan beserta pengembang perangkat lunak pada manre.id dapat lebih mudah memehami dasar daripada kebutuhan dan proses bisnis mereka.

Peneliti memulai penelitian dengan mempelajari studi sebelumnya tentang SPLE. Kemudian, peneliti merancang dan melakukan langkah-langkah dari Teknik Domain untuk menentukan Persyaratan Domain bagi model manre yang dirancang menggunakan dua diagram yakni CBFM dan OVM. Metode ini menghasilkan model untuk analisis dan desain domain engineering pada sistem aplikasi manre.id. Setelah peneliti memperlihatkan hasil model yang peneliti usulkan pada sistem aplikasi manre.id, peneliti mengemukakan bahwa dengan diterapkannya CBFM dan OVM, maka proses bisni yang akan dihasilkan dapat berjalan dengan lebih lancar dan mudah. Ketika terdapat kesalahan fitur ataupun hubungan variant dengan variation points pada saat implementasi, maka dengan melihat model CBFM dan OVM hal tersebut dapat diatasi. Peneliti mengukakan bahwa diperlukannya OVM model dalam rencana proses bisnis didasari oleh dokumentasi yang dibutuhkan oleh sistem aplikasi manre.id secara kompleks dan terstruktur seiring dengan meningkatkan kebutuhan fitur dari pelanggan maupun pemangku kepentingan aplikasi manre.id.

Hasil pemodelan menunjukkan bahwa penggunaan *Cardinality-Based Feature Model* (CBFM) mampu merepresentasikan struktur fitur aplikasi Manre.id secara hierarkis dengan membedakan fitur *mandatory, optional, dan alternative*. Representasi tersebut memberikan gambaran yang lebih jelas mengenai variasi produk yang dapat dikembangkan dalam lingkungan *Software Product Line Engineering* (SPLE). Dengan demikian, proses identifikasi kebutuhan domain menjadi lebih sistematis dibandingkan pendekatan konvensional yang umumnya mendokumentasikan kebutuhan dalam bentuk daftar spesifikasi tanpa menunjukkan hubungan antarfitur secara eksplisit.

Transformasi CBFM menjadi *Orthogonal Variability Model* (OVM) memungkinkan pemisahan antara struktur fitur dan dokumentasi variabilitas. Pemisahan ini memberikan keuntungan dalam proses pengelolaan perubahan, karena penambahan maupun

modifikasi fitur tidak selalu memerlukan perubahan pada keseluruhan model fitur. Dokumentasi variabilitas yang terpisah juga meningkatkan keterlacakan antara *variation point, variant*, dan kebutuhan sistem sehingga memudahkan pengembang dalam melakukan konfigurasi produk yang berbeda sesuai kebutuhan pengguna.

Selain itu, pemodelan *dependency* melengkapi representasi variabilitas dengan menunjukkan hubungan ketergantungan antar *variation point* maupun *variant*. Informasi ini menjadi penting pada tahap *application engineering* karena membantu pengembang mengidentifikasi fitur yang harus dipilih secara bersamaan maupun fitur yang tidak dapat digunakan secara bersamaan. Dengan adanya dokumentasi *dependency* tersebut, risiko terjadinya konfigurasi produk yang tidak valid dapat diminimalkan sejak tahap perancangan.

Jika dibandingkan dengan penggunaan *Feature Model* secara tunggal, integrasi CBFM, OVM, dan *dependency model* memberikan dokumentasi variabilitas yang lebih komprehensif. CBFM berperan dalam memodelkan struktur fitur beserta kardinalitasnya, sedangkan OVM memfokuskan dokumentasi pada aspek variabilitas. Kombinasi keduanya menghasilkan representasi yang lebih mudah dipahami serta lebih mendukung proses evolusi perangkat lunak ketika terjadi penambahan fitur maupun perubahan kebutuhan pada siklus pengembangan berikutnya.

Secara keseluruhan, hasil penelitian menunjukkan bahwa pendekatan pemodelan yang diusulkan tidak hanya menghasilkan dokumentasi kebutuhan yang lebih terstruktur, tetapi juga mendukung prinsip utama SPLE, yaitu *reuse, maintainability, dan variability management*. Oleh karena itu, workflow yang dihasilkan berpotensi menjadi acuan dalam proses *domain engineering* pada pengembangan aplikasi lain yang memiliki karakteristik variabilitas tinggi.

Untuk memastikan bahwa model yang dihasilkan telah merepresentasikan kebutuhan domain secara lengkap dan konsisten, dilakukan validasi internal melalui proses pemeriksaan ketercakupan dan konsistensi. Validasi dilakukan dengan memetakan seluruh fitur yang telah diidentifikasi pada tahap analisis kebutuhan ke dalam model CBFM, kemudian memastikan bahwa setiap fitur variabel berhasil ditransformasikan menjadi variation point atau variant pada OVM serta memiliki hubungan *dependency* yang sesuai. Selain itu, dilakukan pemeriksaan terhadap hubungan *mandatory, optional, alternative*, dan *exclusive-or* (XOR) untuk memastikan tidak terdapat inkonsistensi antar elemen model.

Tabel 1. Validasi Model

Kriteria Validasi	Hasil
Total fitur domain	100%
Fitur berhasil dimodelkan dalam CBFM	100%

Kriteria Validasi	Hasil
Fitur variabel berhasil dipetakan ke OVM	100%
Variation Point terdokumentasi	Ya
Varian terdokumentasi	Ya
Dependency tervalidasi	Ya
Inkonsistensi relasi	Tidak ditemukan

Hasil validasi menunjukkan bahwa seluruh fitur domain berhasil direpresentasikan dalam CBFM dan dipetakan secara konsisten ke dalam OVM. Seluruh *variation point, variant*, serta hubungan *dependency* berhasil terdokumentasi tanpa ditemukan konflik relasi maupun redundansi pada model. Hal ini menunjukkan bahwa *workflow* pemodelan yang diusulkan mampu menjaga konsistensi representasi variabilitas selama proses transformasi dari *feature model* menuju *orthogonal variability model*.

Temuan penelitian ini sejalan dengan Pohl dan Metzger (2014) yang menyatakan bahwa dokumentasi variabilitas merupakan aspek fundamental dalam *Software Product Line Engineering* untuk mendukung *reuse* dan konfigurasi produk. Selain itu, hasil penelitian ini juga mendukung Roos-Frantz *et al*. (2011) yang menunjukkan bahwa OVM memberikan dokumentasi variabilitas yang lebih eksplisit dibandingkan *Feature Model*. Berbeda dengan penelitian sebelumnya yang umumnya menggunakan CBFM atau OVM secara terpisah, penelitian ini mengintegrasikan kedua model tersebut dalam satu *workflow* pemodelan sehingga hubungan antara struktur fitur, *variation point*, dan *dependency* dapat direpresentasikan secara lebih sistematis.

4. KESIMPULAN

Merancang model standar untuk sistem aplikasi manre.id sesuai dengan standar SPLE. Model ini telah disiapkan untuk mengembangkan dan memproduksi secara besar-besaran sistem tersebut untuk dianggap sebagai lini produksi terintegrasi. Produksi aplikasi baru dengan cepat dan sangat bergantung pada kepuasan akan kebutuhan klien dan pasar, yang menjadikan opsi ini tantangan besar. Kemampuan merancang produk apa pun sesuai dengan standar SPLE dengan mudah, jika departemen produk tunduk pada spesifikasi SPLE. Peneliti menggunakan alat Rekayasa Perangkat Lunak untuk analisis dan perancangan untuk menghasilkan model CBFM dan OVM yang dapat digunakan dalam sistem aplikasi manre. Peneliti telah mengidentifikasi melalui model CBFM dan OVM untuk bagaimana membangun persyaratan bagi persyaratan domain persyaratan teknik domain dan seperti yang diharapkan menggunakannya untuk mengembangkan tahap rekayasa aplikasi sistem manre. Dengan ini, peneliti telah menghasilkan model baru bagi pengembang aplikasi manre.id untuk pengembangan dan pembaruan persyaratan dasar.

Penelitian ini memberikan kontribusi berupa penyusunan *workflow* pemodelan variabilitas yang mengintegrasikan *Cardinality-Based Feature Model* (CBFM), *Orthogonal Variability Model* (OVM), dan *dependency model* dalam satu proses domain engineering. *Workflow* tersebut diterapkan pada studi kasus aplikasi *marketplace* kuliner Manre.id untuk menghasilkan dokumentasi variabilitas yang lebih sistematis dan terstruktur. Selain itu, penelitian ini menunjukkan bahwa integrasi CBFM dan OVM memungkinkan keterlacakan hubungan antara fitur, *variation point*, dan *variant* secara lebih jelas, sehingga mendukung proses analisis kebutuhan, mempermudah konfigurasi produk, serta memfasilitasi pengembangan dan evolusi perangkat lunak berbasis *Software Product Line Engineering* (SPLE).

Pengembang yang bekerja di bidang rekayasa perangkat lunak arsitektur dan menggunakan perangkat lunak desain untuk menghasilkan produksi massal perlu menetapkan visi dan strategi umum untuk produk yang dibutuhkan dengan mengidentifikasi kebutuhan masyarakat akan produk ini dan ini didasarkan pada pengembangan atau penambahan beberapa karakteristik produk sebelumnya atau produk yang diperbarui. Di sini, setelah produksi kebutuhan dasar sistem manre sesuai dengan metode peneliti di atas melalui model CBFM dan OVM dengan menggunakan alat analisis dan perancangan, penelitian ini akan menemukan cara bagaimana menemukan produk yang tepat sesuai dengan kriteria standar. Peneliti mengusulkan kedepannya untuk mencari metode hybrid yang sesuai dengan model sistem aplikasi manre untuk pembuatan peta jalan yang menentukan kebutuhan dasar dan karakteristik scalable dalam model yang diusulkan dan dapat menggunakan hasil penelitian ini sebagai masukan nyata untuk dapatkan hasil dengan spesifikasi tinggi yang dibutuhkan.

DAFTAR PUSTAKA

Abbas, A., Siddiqui, I. F., Lee, S. U.-J., & Bashir, A. K. (2017). Binary Pattern for Nested Cardinality Constraints for Software Product Line of IoT-Based Feature Models. *IEEE Access*, *5*, 3971–3980. https://doi.org/10.1109/ACCESS.2017.2680470

Arrieta, A. (2019, September 9). Variability Modeling and Management of MATLAB/Simulink Models. *23rd International Systems and Software Product Line Conference* .

Business Process Variability Modeling: A Survey (March 2017).

Case Tool Support for Variability Management in Software Product Lines. ACM Comput. Surv (March 2017).

Chavarriaga, J., Casallas, R., Parra, C., Cecilia, M., & Ricardo, C. (2019, September 9). Nine Years of Courses on Software Product Lines at Universidad de los Andes, Colombia. *Software Product Life Cycle 19'*.

Czarnecki, K., Helsen, S., & Eisenecker, U. (2004, August). Staged configuration using feature models. *Proceedings of the Third International Conference on Software Product Lines (SPLC '04)*.

Frantz, F. R., & Segura, S. (2008). *Automated Analysis of Orthogonal Variability Models. A First Step*. Book.

Horcas, J. M., & Pinto, M. (2020). Automatic Evolution of Eco-Efficient Software Architectures with CVL Models. *IEEE Latin America Transactions*, *18*(07), 1238–1246. https://doi.org/10.1109/TLA.2020.9099765

Lee, J., & Hwang, S. (2016). *Variability Change Management Using the Orthogonal Variability Model-Based Traceability*. Wireless Pers Commun .

Li, M., Batmaz, F., Guan, L., Grigg, A., Ingham, M., & Bull, P. (2015). Model-based systems engineering with requirements variability for embedded real-time systems. *2015 IEEE International Model-Driven Requirements Engineering Workshop (MoDRE)*, 1–10. https://doi.org/10.1109/MoDRE.2015.7343874

Li, M., Grigg, A., Dickerson, C., Guan, L., & Ji, S. (2019). A Product Line Systems Engineering Process for Variability Identification and Reduction. *IEEE Systems Journal*, *13*(4), 3663–3674. https://doi.org/10.1109/JSYST.2019.2897628

Pohl, K., & Metzger, A. (2014, May 31). Software Product Line Engineering and Variability Management: Achievements and Challenges. *FOSE'14*.

Pohl, K., & Metzger, A. (2018). *Software Product Lines*. The Essence of Software Engineering.

Roos-Frantz, Fabricia, Benavides, David, Ruiz-Cortés, & Antonio. (2011). *Feature Model to Orthogonal Variability Model Transformation towards Interoperability between Tools.*

Wahyudianto, Budiardjo, E. K., & Zamzani, M. (2014). Feature Modeling and Variability Modeling Syntactic Notation Comparison and Mapping. *Journal of Computer and Communications*, *2*, 101–108.