

IMPLEMENTASI SYSTEM MODELING LANGUAGE PADA PEMODELAN APLIKASI REPARATION

Oleh:

Sampurna Panjiwinata¹, Mohammad Fajar^{2*}, Hasniati³

^{1,2,3}Teknik Informatika, STMIK KхарISMA Makassar

e-mail: ¹sampurnapanjiwinata_20@kharisma.ac.id, ²fajar@kharisma.ac.id,
³hasniati@kharisma.ac.id

Abstrak: Pada proses pengembangan perangkat lunak, developer umumnya menggunakan Unified Modeling Language (UML) sebagai bahasa untuk memodelkan dan merancang sistem. Saat ini telah dikembangkan versi yang lebih luas dari UML yaitu System Modeling Language (SysML) yang menyediakan sejumlah perbaikan, salah satunya yaitu alokasi untuk spesifikasi kebutuhan melalui requirement diagram. Meskipun demikian, SysML masih banyak digunakan untuk memodelkan sistem-sistem tertanam dibanding pemakaiannya pada sistem informasi atau jenis aplikasi komputer yang lebih luas. Oleh karena itu, penelitian ini bertujuan untuk mengimplementasikan SysML pada sistem Reparation. Aplikasi Reparation merupakan aplikasi yang menyediakan jasa servis secara online, membantu masyarakat dalam mencari dan memesan jasa servis dengan mudah dan cepat tanpa harus ke bengkel. Wawancara dan studi literatur dilakukan untuk memperoleh kebutuhan fungsional dan non-fungsional sistem yang didefinisikan dalam tabel spesifikasi kebutuhan. Sementara pengujian sistem dilakukan untuk menguji sejumlah fitur contoh yang dipilih. Dari studi ini menunjukkan bahwa SysML dapat diimplementasikan pada pemodelan aplikasi Reparation. Diagram-diagram yang dihasilkan yaitu activity diagram, parametric diagram, requirement diagram, use case diagram, dan sequence diagram. Parametric diagram digunakan untuk menentukan batasan-batasan berupa atribut dari fungsionalitas pada sistem, sementara requirement diagram digunakan untuk menganalisis kebutuhan fungsional dan non-fungsional sistem seperti Daftar Akun, Pesan Jasa Servis, dan Perangkat Pengguna, Termasuk hubungan dari requirement diagram ke use case diagram dan parametric diagram dapat ditelusuri. Selain itu, pengujian black box menunjukkan fungsionalitas yang diuji seperti Pesan Jasa Servis dan Tambah Pesanan Jasa Servis berhasil dijalankan.

Kata kunci: SYSML, UML, Reparation, Requirement Diagram, Parametric Diagram

Abstract: In the software development process, developers commonly use the Unified Modeling Language (UML) as a language to model and design systems. Currently, an expanded version of UML, known as the System Modeling Language (SysML), has been developed, providing several enhancements, one of which is the allocation for specifying requirements through requirement diagrams. However, SysML is still predominantly used for modeling embedded systems rather than for broader computer information systems or applications. Therefore, this research aimed to implement SysML in the Reparation system. The Reparation application is a platform that provides online repair services, assisting the public in easily and quickly finding and booking repair services without having to visit a workshop. Interviews and literature studies were conducted to obtain functional and non-functional system requirements, which were defined in requirement specification tables. System testing was performed to test a selection of example features. This study demonstrated that SysML can be implemented in modeling the Reparation application. The diagrams generated include activity diagrams, parametric diagrams, requirement diagrams, use case diagrams, and sequence diagrams. Parametric diagrams were used to determine

* Corresponding author : Mohammad Fajar (fajar@kharisma.ac.id)

constraints in the form of attributes of system functionalities, while requirement diagrams were used to analyze the functional and non-functional requirements of the system, such as Account Registration, Service Booking, and User Devices. Including the relationships between requirement diagrams, use case diagrams, and parametric diagrams could be traced. Additionally, black-box testing demonstrated that functionalities such as Service Booking and Adding Service Orders could be successfully executed.

Keywords: SYSML, UML, Reparation, Requirement Diagram, Parametric Diagram

1. PENDAHULUAN

Dalam pembuatan sebuah sistem informasi, diperlukan tahapan-tahapan dan teknik tertentu agar sistem yang dikembangkan tersebut memiliki kualitas yang baik dan biaya pengembangan (*cost*) yang rendah. Tahapan-tahapan tersebut seringkali disebut dengan *System Development Life Cycle* (SDLC) yang merupakan tahapan yang dilakukan dalam membangun sistem informasi [1]. Secara umum, terdapat lima fase dalam SDLC yaitu Perencanaan, Analisis, Desain, Implementasi, dan *Maintenance* [2]. Pada pengembangan sistem informasi ataupun perangkat lunak, pemodelan sistem merupakan hal yang sangat penting. Pemodelan sistem dapat membantu para *developer* untuk memahami dan menyederhanakan sistem informasi yang dibangun. Dalam memodelkan sistem, secara umum para *developer* menggunakan *Unified Modeling Language* (UML). Namun, UML memiliki keterbatasan diantaranya adalah UML tidak dapat memodelkan sistem yang lebih kaku [3]. Oleh karena itu, Object Management Group (OMG) mengembangkan ekstensi dari UML yang disebut *System Modeling Language* (SysML).

System Modeling Language merupakan bahasa pemodelan yang digunakan untuk memodelkan sistem yang lebih kompleks [4], [5] seperti dari sisi *hardware*, *software*, proses, informasi, personil, dan fasilitas [3]. SysML juga mendukung analisis, desain, verifikasi, dan validasi sistem [6]. Diantara diagram yaitu *requirement diagram* dan *parametric diagram* [7]–[11].

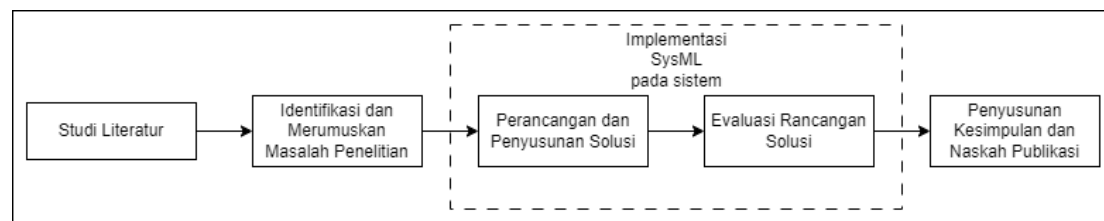
Reparation merupakan aplikasi yang menyediakan jasa servis secara *online*. Reparation dapat langsung diunduh pada Google Play Store melalui tautan berikut <https://play.google.com/store/apps/details?id=id.kharisma.studio.atelier>. Aplikasi *mobile* berbasis Android ini bertujuan untuk membantu memudahkan masyarakat dalam mencari dan memesan jasa servis dengan mudah dan cepat tanpa harus pergi ke bengkel. Reparation memiliki beberapa fitur seperti *Login*, Pesan Jasa Servis, dan Konsultasi. Reparation sendiri pada awalnya dibangun dan dikembangkan dengan pemodelan sistem yang sudah umum digunakan yaitu UML. Namun, pemodelan SysML merupakan pemodelan sistem yang masih baru dan belum begitu luas pemanfaatannya seperti halnya UML.

Selama ini, pemodelan SysML umumnya digunakan pada sistem-sistem tertanam (*embedded system*) seperti pada penelitian yang dilakukan oleh Kelvin Aurin pada jaringan nirkabel [12], Amal Ahmed Anda dan Daniel Amyot pada *cyber-physical systems* [13], Emre Canbulut, dkk pada sistem satelit sederhana [14]. Berbeda dengan penelitian yang dilakukan oleh David Hetherinton yang memodelkan SysML pada *game* simulasi pengeboran minyak

[15]. Terdapat juga penelitian yang dilakukan oleh Faïda Mhenni yang membuat model desain arsitektur pada *mechatronic systems* dengan SysML seperti teleskop [16]. Sementara itu, untuk pemodelan sistem informasi atau aplikasi masih kurang penggunaannya. Oleh karena itu, penelitian ini bertujuan untuk mengimplementasikan *System Modeling Language* (SysML) pada pemodelan sistem Reparation.

2. METODE PENELITIAN

Penelitian ini menggunakan sumber data primer dan sekunder dengan jenis data kualitatif. Sumber data pada penelitian ini terbagi menjadi dua, yaitu sumber data primer diperoleh dari hasil wawancara dan sumber data sekunder yang diperoleh dari jurnal-jurnal, artikel, buku, penelitian yang terdahulu. Selain itu, data kualitatif juga diperoleh dari pengujian sistem yang menjelaskan apakah pemetaan dari dua diagram baru pada SysML dapat dilakukan berupa pemetaan *requirement diagram* ke *use case diagram*, pemetaan *requirement diagram* ke *parametric diagram*, pemetaan *parametric diagram* ke baris kode serta apakah SysML dapat diimplementasikan pada aplikasi Reparation. Tahapan-tahapan penelitian disajikan pada Gambar 1.



Gambar 1. Tahapan Penelitian

Berikut ini merupakan tahapan-tahapan yang dilakukan dalam penelitian ini.

1. Studi Literatur

Studi literatur dilakukan untuk mendapatkan informasi awal terkait topik penelitian yang dipilih yaitu analisis dan desain sistem menggunakan SysML dalam buku dan makalah-makalah di jurnal, seperti “Pemodelan Sistem Jaringan Sensor Nirkabel Menggunakan *System Modeling Language*” [12], “*Application of Model-based Systems Engineering with SysML in A Small Satellite Project*” [14], “*SysML Requirements for Training Game Design*” [15], “*A Practical Guide to SysML: The Systems Modeling Language*” [7], “*SysML Distilled: A Brief Guide to the Systems Modeling Language*” [8], “*Systems Engineering with SysML/UML*” [9], “*Self-Adaptive System Verification based on SysML*” [17] dan lain sebagainya.

2. Identifikasi dan Merumuskan Masalah Penelitian

Pada tahapan ini, peneliti menyusun rumusan masalah yang menjadi fokus permasalahan yang akan dikaji dalam penelitian yaitu bagaimana mengimplementasikan *System Modeling Language* (SysML) pada pemodelan sistem Reparation.

3. Perancangan dan Penyusunan Solusi

Pada tahapan ini, dilakukan penyusunan solusi yang diusulkan sebagai solusi dari masalah yang telah dirumuskan yaitu dengan mengimplementasikan SysML pada pemodelan

sistem Reparation. Implementasi SysML dilakukan dengan menganalisis data hasil wawancara sebelumnya yang kemudian diolah menjadi tabel spesifikasi kebutuhan. Selanjutnya, hasil analisis kebutuhan yang diperoleh dari tabel digunakan dalam pembuatan beberapa diagram pada SysML diantaranya *requirement diagram*, *use case diagram*, *activity diagram*, dan *sequence diagram*. Setelah itu, dibuatkan desain dari sistem seperti arsitektur sistem, tampilan *user interface*, dan *parametric diagram*.

4. Evaluasi Rancangan Solusi

Pada tahapan ini, peneliti melakukan evaluasi terhadap rancangan solusi yang diusulkan. Evaluasi dilakukan dengan pengujian *black box* dimana peneliti akan menguji kode program dari beberapa fitur apakah berhasil dijalankan atau tidak. Evaluasi yang kedua dilakukan evaluasi ketertelusuran diagram dan kode, apakah diagram SysML yang dihasilkan dapat dipetakan ke diagram lain serta apakah dapat digunakan sebagai dasar pembuatan kode implementasi.

5. Penyusunan Kesimpulan dan Naskah Publikasi

Hasil evaluasi dijadikan dasar penyusunan kesimpulan penelitian. Setelah kesimpulan didapat, selanjutnya peneliti menyusun laporan akhir atau naskah publikasi penelitian.

3. HASIL DAN PEMBAHASAN

3.1. Deskripsi Data

Wawancara dilakukan untuk memperoleh kebutuhan-kebutuhan pengguna terhadap sistem. Wawancara tersebut melibatkan 20 responden yang terbagi menjadi 10 orang calon pelanggan dengan jumlah pertanyaan sebanyak 12 dan 10 orang calon mitra dengan jumlah pertanyaan sebanyak 15. Dari wawancara yang dilakukan terhadap calon pelanggan, dapat diketahui bahwa respons yang diberikan oleh calon pelanggan untuk status validasi "Ya" sebanyak 63,3% dimana pada *Riskiest Assumption 1* sebanyak 7 responden, *Riskiest Assumption 2* sebanyak 6 responden, dan *Riskiest Assumption 3* sebanyak 6 responden. Sementara untuk status validasi "Tidak" sebanyak 36,7% dimana pada *Riskiest Assumption 1* sebanyak 3 responden, *Riskiest Assumption 2* sebanyak 4 responden, dan *Riskiest Assumption 3* sebanyak 4 responden. Sementara dari wawancara terhadap calon mitra, dapat diketahui bahwa respons yang diberikan oleh calon mitra untuk status validasi "Ya" sebanyak 76,7% dimana pada *Riskiest Assumption 1* sebanyak 6 responden, *Riskiest Assumption 2* sebanyak 9 responden, dan *Riskiest Assumption 3* sebanyak 8 responden. Sementara untuk status validasi "Tidak" sebanyak 23,3% dimana pada *Riskiest Assumption 1* sebanyak 4 responden, *Riskiest Assumption 2* sebanyak 1 responden, dan *Riskiest Assumption 3* sebanyak 2 responden.

3.2. Pembahasan

Data hasil wawancara dianalisis dan diolah sehingga diperoleh data-data hasil penelitian yang berupa deskripsi, gambar, dan tabel. Data-data tersebut terbagi ke dalam dua tahap yaitu Analisis dan Desain. Adapun gambar yang dibuat berupa dua diagram baru pada SysML dan tiga diagram lain diantaranya *requirement diagram*, *use case diagram*, *activity diagram*, *sequence diagram*, dan *parametric diagram*.

3.2.1 Analisis

Pada tahap ini, analisis terhadap data hasil wawancara dilakukan lalu diperoleh data-data hasil penelitian berupa tabel spesifikasi kebutuhan, *requirement diagram*, *use case diagram*, *activity diagram*, dan *sequence diagram*. Data-data tersebut adalah sebagai berikut.

1. Tabel Spesifikasi Kebutuhan

Berdasarkan data hasil wawancara, diperoleh fungsionalitas atau fitur yang dibutuhkan oleh pengguna. Terdapat juga fungsionalitas atau fitur yang diperoleh setelah dilakukan diskusi dengan tim *Founder* yang terdiri dari tiga orang yaitu Henderson Ruslim, Renaldi Anggarkusuma, dan Sampurna Panjiwinata. Diskusi tersebut dilakukan untuk memperoleh fungsionalitas yang sebaiknya ada pada sistem. Tabel 1 memperlihatkan beberapa contoh spesifikasi kebutuhan yang didapatkan dari pengguna.

Tabel 1: Beberapa Contoh Spesifikasi Kebutuhan

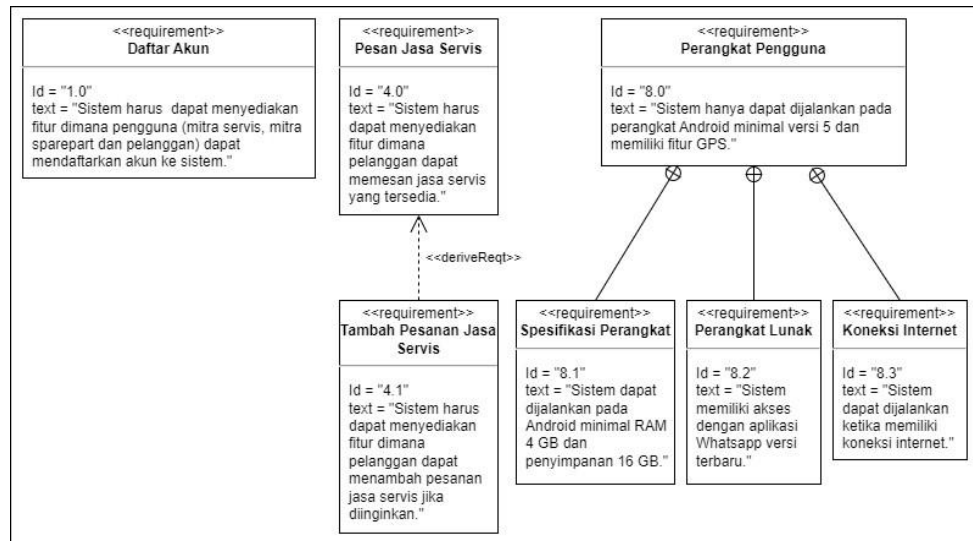
No.	Fungsionalitas	Aktor	Keterangan	Sumber
1	Pesan jasa servis	Pelanggan	Pelanggan dapat memesan jasa servis yang tersedia.	Hasil wawancara
2	Tambah pesanan jasa servis	Pelanggan	Pelanggan dapat menambah pesanan jasa servis jika diinginkan.	Diskusi dengan tim <i>Founder</i>
3	Konsultasi	Mitra servis, pelanggan	Pelanggan dapat melakukan konsultasi dengan mitra servis.	Hasil wawancara
4	Jual <i>sparepart</i> motor	Mitra <i>sparepart</i>	Mitra dapat menjual <i>sparepart</i> motor dengan menambah detail barang seperti nama, deskripsi, tipe, harga, dan gambar.	Hasil wawancara
5	Beli <i>sparepart</i> motor	Pelanggan	Pelanggan dapat membeli <i>sparepart</i> motor.	Hasil wawancara

Pada Tabel 1, dapat dilihat bahwa fungsionalitas Tambah Pesanan Jasa Servis diperoleh dari hasil diskusi dengan tim *Founder*. Sementara, fungsionalitas Pesan Jasa Servis, Konsultasi, Jual *Sparepart* Motor, dan Beli *Sparepart* Motor diperoleh dari hasil wawancara.

2. Requirement Diagram

Berdasarkan hasil analisis kebutuhan pada Tabel 1, diperoleh kebutuhan fungsional yang menjadi acuan dalam pembuatan *requirement diagram*. *Requirement diagram* yang merupakan diagram baru pada SysML memiliki fungsi yang hampir sama dengan tabel spesifikasi kebutuhan. Namun, *requirement diagram* dapat menganalisis kebutuhan fungsional dan non-fungsional, berbeda dengan tabel spesifikasi kebutuhan hanya mampu

menganalisis kebutuhan fungsional saja. Adapun kebutuhan non-fungsional yang diperoleh berupa perangkat pengguna. Contoh *requirement diagram* dapat dilihat pada Gambar 2.



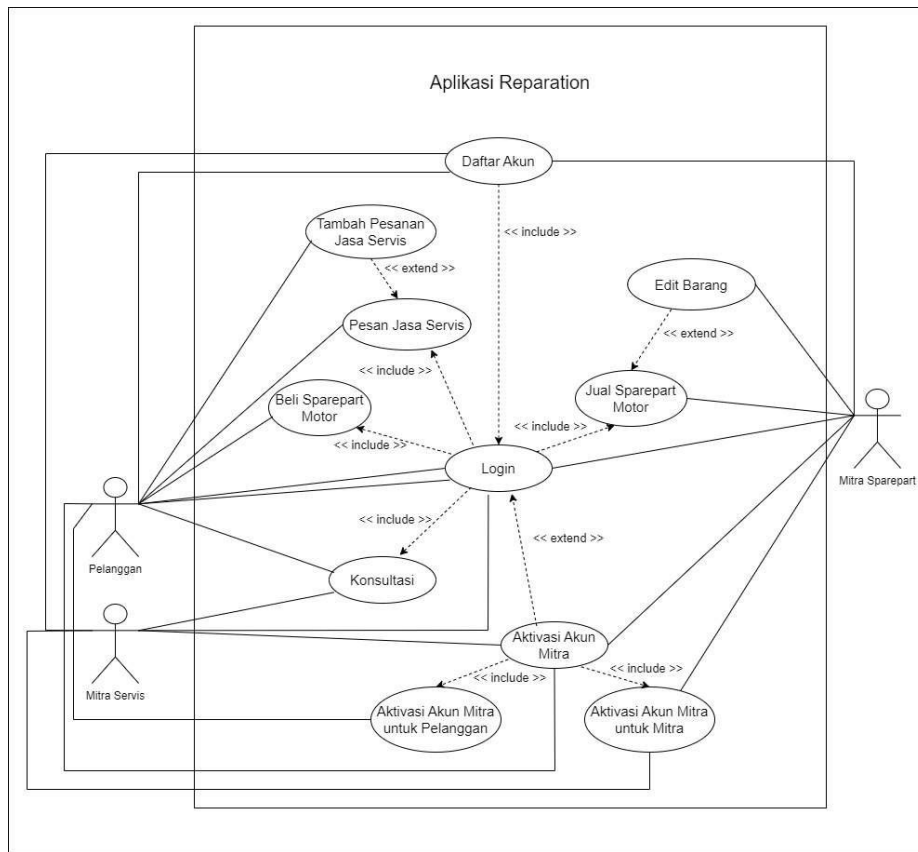
Gambar 2. Requirement Diagram

Kebutuhan fungsional untuk Pesan Jasa Servis memiliki ID 4.0 yang menunjukkan bahwa fitur Pesan Jasa Servis merupakan salah satu *requirement* induk (bukan merupakan bagian dari *requirement* yang lain). *Text* yang terdapat di dalamnya berfungsi untuk menjelaskan *requirement* Pesan Jasa Servis secara lebih spesifik. Sementara itu, fitur Tambah Pesanan Jasa Servis merupakan sebuah *requirement* baru yang diperoleh dari fitur Pesan Jasa Servis sehingga memiliki ID 4.1 dan dinotasikan dengan *dashed arrow pointing* dan tulisan *<<deriveReq>>*.

Adapun kebutuhan non-fungsional untuk Perangkat Pengguna dengan ID 8.0 menggunakan notasi yang berbeda yang disebut juga *Namespace Containment*. Hal ini menunjukkan bahwa *requirement* Perangkat Pengguna memiliki tiga *requirement* lain yang terkandung di dalamnya diantaranya Spesifikasi Perangkat, Perangkat Lunak, dan Koneksi Internet.

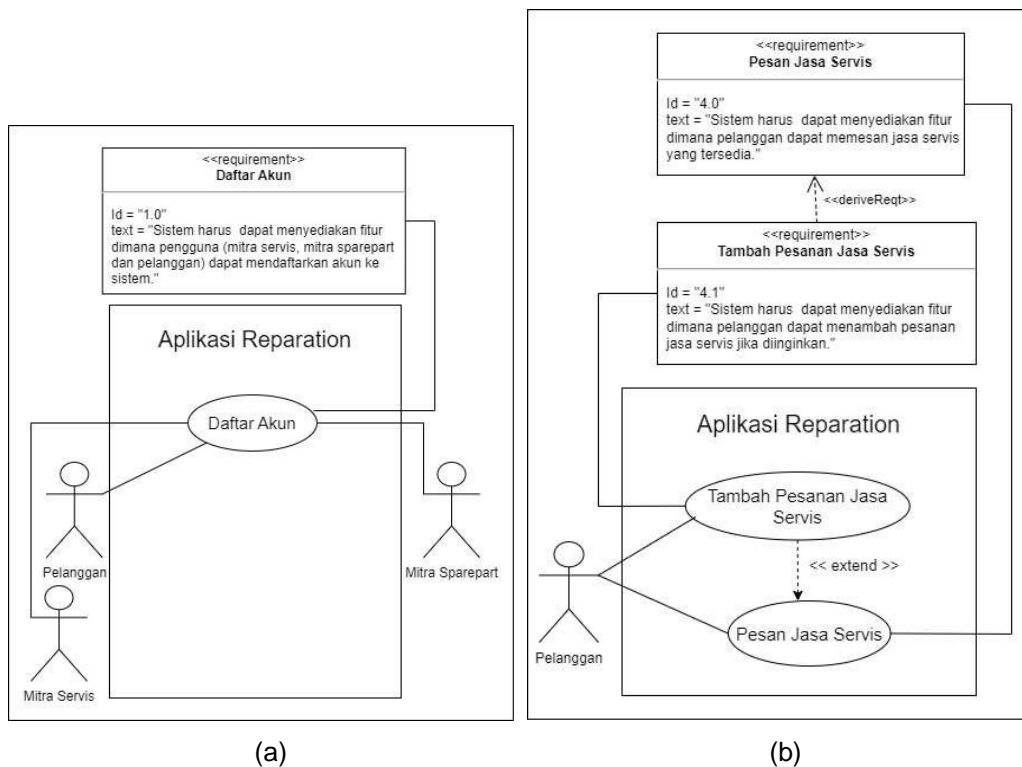
3. Use Case Diagram

Berdasarkan hasil analisis kebutuhan pada Tabel 1 dan Gambar 2, diperoleh fungsionalitas pada sistem yang kemudian digambarkan pada *use case diagram* dan dihubungkan dengan aktor-nya masing-masing. *Use case diagram* disajikan pada Gambar 3.



Gambar 3. Use Case Diagram

4. Pemetaan Requirement Diagram ke Use Case Diagram

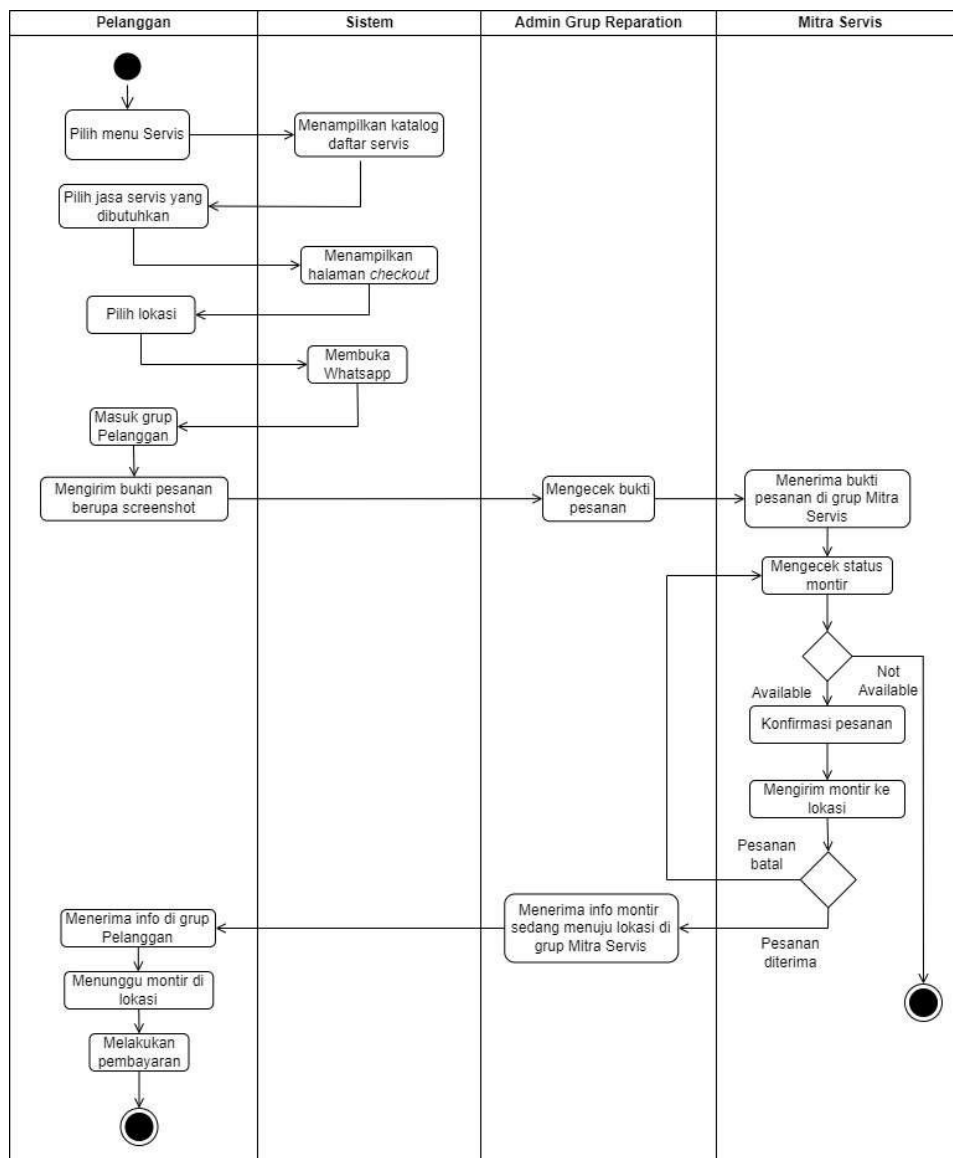


Gambar 4. Pemetaan Requirement Diagram ke Use Case Diagram

Berdasarkan Gambar 2 dan 3, maka pemetaan dilakukan dari *requirement diagram* ke *use case diagram* untuk memastikan ketertelusuran dari setiap kebutuhan fungsional yang terdapat pada Gambar 2 dengan fungsionalitas yang terdapat pada Gambar 3.

Pada Gambar 4 (a), dapat dilihat bahwa kebutuhan fungsional yaitu Daftar Akun berhasil dipetakan ke dalam *use case diagram* dengan aktor Pelanggan, Mitra Servis, dan Mitra *Sparepart*. Sementara pada Gambar 4 (b), dapat dilihat kebutuhan fungsional Pesan Jasa Servis dan Tambah Pesanan Jasa Servis juga berhasil dipetakan ke dalam *use case diagram* dengan aktor Pelanggan.

5. Activity Diagram



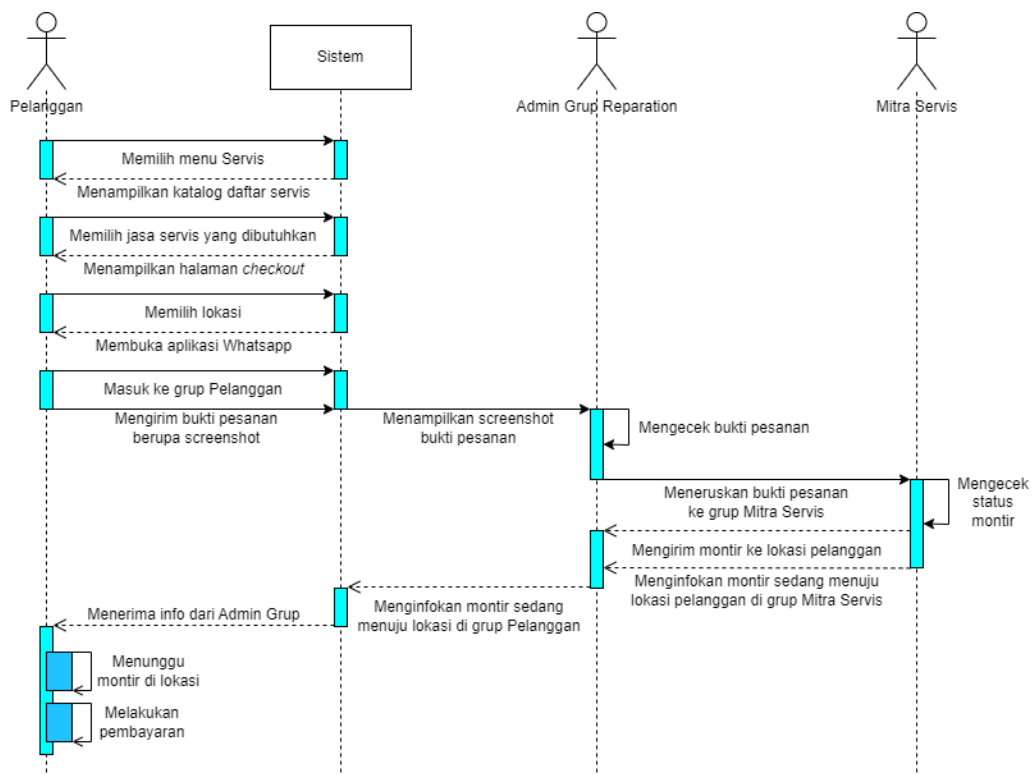
Gambar 5. Activity Diagram Pesan Jasa Servis

Berdasarkan Gambar 3, analisis sistem dengan pembuatan alur aktivitas dari setiap fungsionalitas yang ada pada sistem. Contoh *activity diagram* yang dibuat dapat dilihat pada Gambar 5.

Gambar 5 menjelaskan alur aktivitas pada fitur Pesan Jasa Servis yang dimulai dari pelanggan masuk ke menu Servis, sistem menampilkan katalog daftar servis lalu pelanggan memilih jasa servis yang dibutuhkan. Selanjutnya, sistem menampilkan halaman *checkout*, pelanggan memilih lokasi pelanggan berada. Lalu, sistem membuka aplikasi Whatsapp, pelanggan masuk ke grup Pelanggan, mengirim bukti pesanan berupa *screenshot* lalu bukti pesanan dikirim ke grup Mitra Servis dan mitra servis mengecek status montir yang *available*. Jika tidak ada montir yang *available*, maka pesanan diberikan kepada mitra servis lain yang memiliki montir *available*. Selanjutnya, mitra servis mengonfirmasi pesanan. Jika terdapat kendala yang dialami montir maka pesanan dibatalkan dan pesanan diberikan kepada mitra servis lain. Jika pesanan diterima, mitra servis mengirim montir ke lokasi dan memberi info pada Admin Grup Reparation. Selanjutnya, pelanggan menerima info, menunggu montir di lokasi, dan melakukan pembayaran berupa *cash* pada montir.

6. Sequence Diagram

Berdasarkan Gambar 3, analisis sistem dengan pembuatan alur urutan dari setiap fungsionalitas yang ada pada sistem. Contoh *sequence diagram* yang dibuat dapat dilihat pada Gambar 6.



Gambar 6. Sequence Diagram Pesan Jasa Servis

Pada Gambar 6, dapat dilihat bahwa alur urutan pada fitur Pesan Jasa Servis untuk Mitra dimulai dari pelanggan memilih menu Servis, sistem menampilkan katalog daftar servis lalu pelanggan memilih jasa servis yang dibutuhkan. Selanjutnya, sistem menampilkan halaman *checkout* lalu pelanggan memilih lokasi pelanggan berada. Sistem membuka

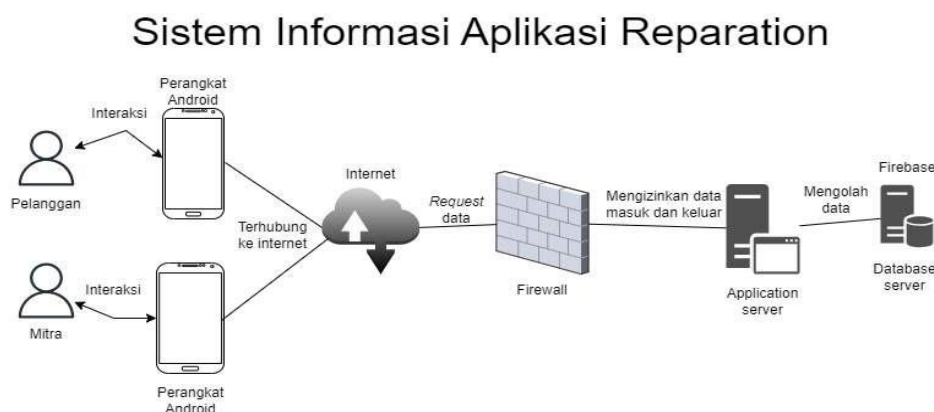
aplikasi Whatsapp, pelanggan masuk ke grup Pelanggan, dan mengirim bukti pesanan berupa *screenshot*. Lalu Admin Grup Reparation mengecek bukti pesanan dan meneruskannya ke grup Mitra Servis. Selanjutnya, mitra servis mengecek status montir. Lalu, mitra servis mengirim montir ke lokasi dan memberi info pada Admin Grup Reparation. Selanjutnya, Admin Grup Reparation meneruskan info di grup Pelanggan, pelanggan menerima info dari Admin Grup Reparation, menunggu montir di lokasi, dan melakukan pembayaran berupa *cash* pada montir.

3.2.2 Desain

Pada tahap ini, desain dari sistem dibuat lalu diperoleh data-data hasil penelitian seperti arsitektur sistem, tampilan *user interface*, dan *parametric diagram*.

1. Arsitektur Sistem

Pada Gambar 7, dapat dilihat bahwa aliran data dimulai dari perangkat Android pengguna yang terhubung ke internet lalu melakukan *request* data ke *application server* dengan melewati *firewall* terlebih dahulu. Lalu, *database server* yang mengolah data mengirim data keluar hingga ke perangkat pengguna.

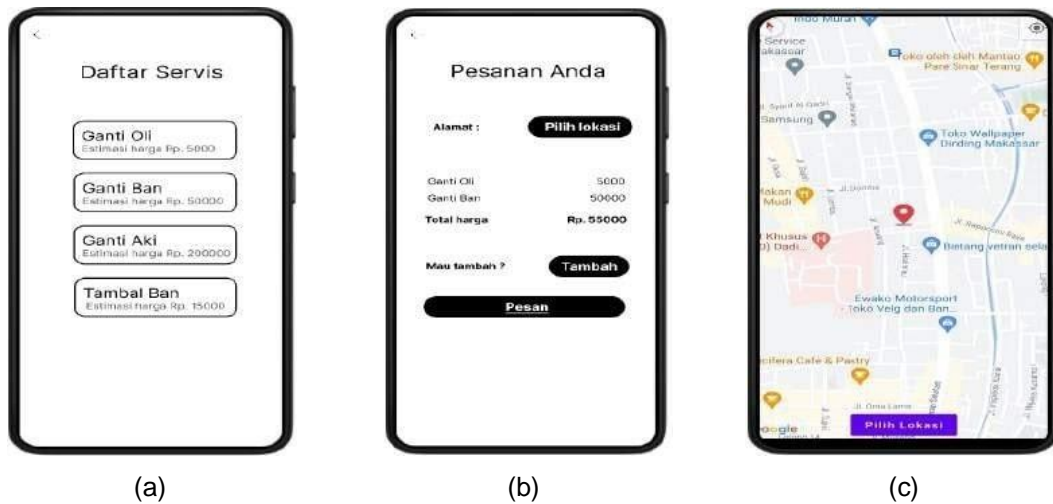


Gambar 7. Arsitektur Sistem

Adapun standar bagi pengguna untuk menggunakan aplikasi Reparation yaitu memiliki sistem operasi minimal Android versi 5 ke atas, *hardware* dengan RAM minimal 4 GB dan penyimpanan internal minimal 16 GB, *software* dengan aplikasi Whatsapp versi terbaru dan Google Play Store, serta untuk jaringan dengan jaringan Broadband atau WiFi minimal 5 Mbps. Sementara itu, standar *database server* yang digunakan yaitu Firebase.

2. Tampilan *User Interface*

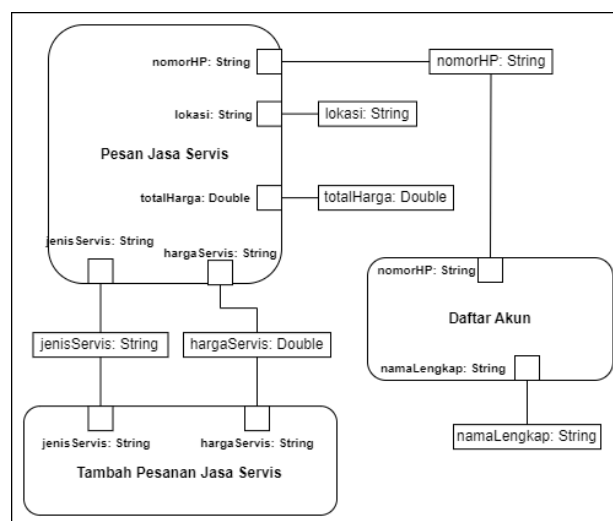
Berdasarkan hasil analisis kebutuhan pada Tabel 1 dan Gambar 2, desain *user interface* yang baru dibuat menggunakan aplikasi Figma. Desain baru tersebut dibuat karena *user interface* yang lama merupakan hasil pemodelan menggunakan pemodelan UML sementara *user interface* yang baru menggunakan pemodelan SysML. Hasil desain tampilan *user interface* dapat dilihat pada Gambar 8.



Gambar 8. Tampilan Halaman Fitur Pesan Jasa Servis dan Tambah Pesanan Jasa Servis

Pada Gambar 8 (a), dapat dilihat bahwa halaman Daftar Servis menampilkan jasa servis yang tersedia serta tanda panah yang berfungsi untuk kembali ke halaman Home. Dapat dilihat juga jasa servis yang ditampilkan lengkap dengan jenis dan harganya masing-masing. Pada Gambar 8 (b), dapat dilihat bahwa halaman Checkout memiliki tombol Pilih Lokasi yang berfungsi bagi pelanggan untuk menentukan lokasi dimana pelanggan berada. Selain itu, terdapat juga pesanan dengan harganya masing-masing. Terdapat juga total harga dari jasa servis yang dipesan serta tombol Tambah yang berfungsi untuk menambah pesanan jika diinginkan. Adapun tombol Pesan yang berfungsi menyimpan data ke database dan membuka aplikasi Whatsapp untuk masuk ke grup Pelanggan. Sementara pada Gambar 8 (c), dapat dilihat bahwa halaman Pilih Lokasi menampilkan peta Google Maps yang berfungsi bagi pelanggan untuk menentukan dimana lokasi pelanggan berada.

3. Parametric Diagram



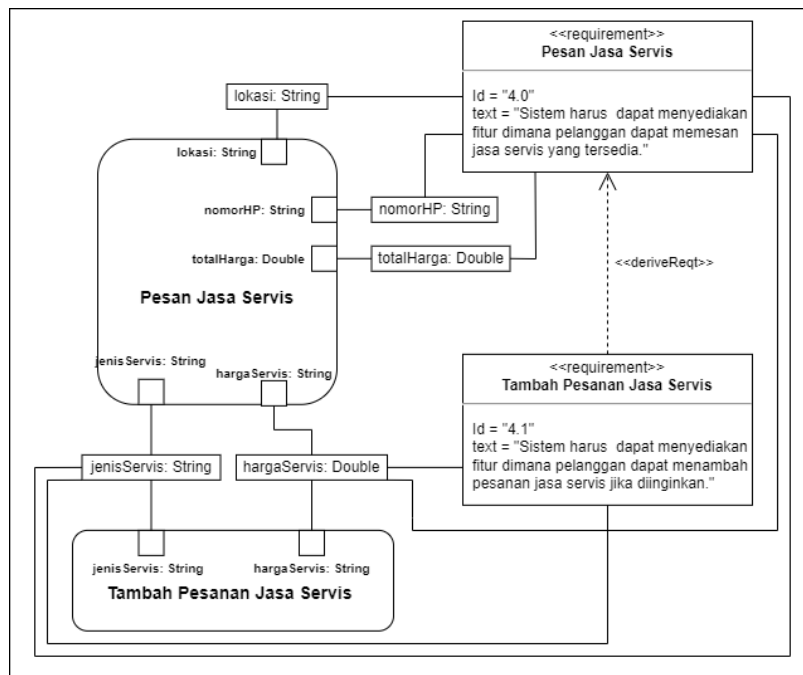
Gambar 9. Contoh Parametric Diagram

Contoh parametric diagram pada Gambar 9 menjelaskan batasan-batasan berdasarkan fungsionalitas yang ada pada sistem. Dapat dilihat bahwa batasan-batasan

tersebut berupa atribut atau parameter dari fungsionalitas yang ada pada sistem yang terhubung dengan *value property* melalui *port*-nya masing-masing. Dapat dilihat juga terdapat beberapa *value property* yang terhubung dengan dua *port* dari fungsionalitas yang berbeda.

4. Pemetaan *Requirement Diagram* ke *Parametric Diagram*

Berdasarkan Gambar 2 dan Gambar 9, dilakukan pemetaan dari *requirement diagram* ke *parametric diagram* untuk mengetahui ketertelusuran diagram. Pemetaan tersebut dapat dilihat pada Gambar 10. Gambar 10 menunjukkan hubungan yang lebih kompleks dikarenakan adanya ketergantungan antara beberapa atribut atau parameter yang terlibat dengan terpenuhinya suatu *requirement*.

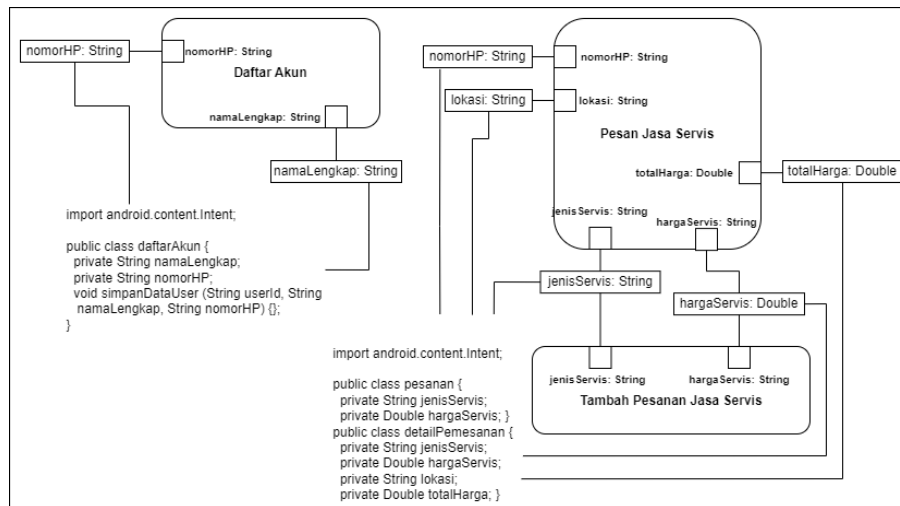


Gambar 10. Contoh Pemetaan *Requirement Diagram* ke *Parametric Diagram*

Berdasarkan Gambar 10, dapat dilihat bahwa setiap atribut yang ada berhasil dipetakan ke *requirement* Pesan Jasa Servis dan Tambah Pesan Jasa Servis. Adapun atribut-atribut yang terdapat pada fungsionalitas Pesan Jasa Servis adalah daftarServis, nomorHP, lokasi, totalHarga, jenisServis, dan hargaServis. Sementara atribut-atribut yang terdapat pada fungsionalitas Tambah Pesan Jasa Servis adalah daftarServis, jenisServis, dan hargaServis.

5. Pemetaan *Parametric Diagram* ke Baris Kode

Pada tahap ini, dilakukan pemetaan dari *parametric diagram* ke baris kode Java pada aplikasi Android berdasarkan *parametric diagram* yang telah dibuat pada Gambar 9. Sebagai contoh, pemetaan hanya dilakukan terhadap dua fungsionalitas yaitu Daftar Akun dan Pesan Jasa Servis. Contoh pemetaan tersebut dapat dilihat pada Gambar 11.

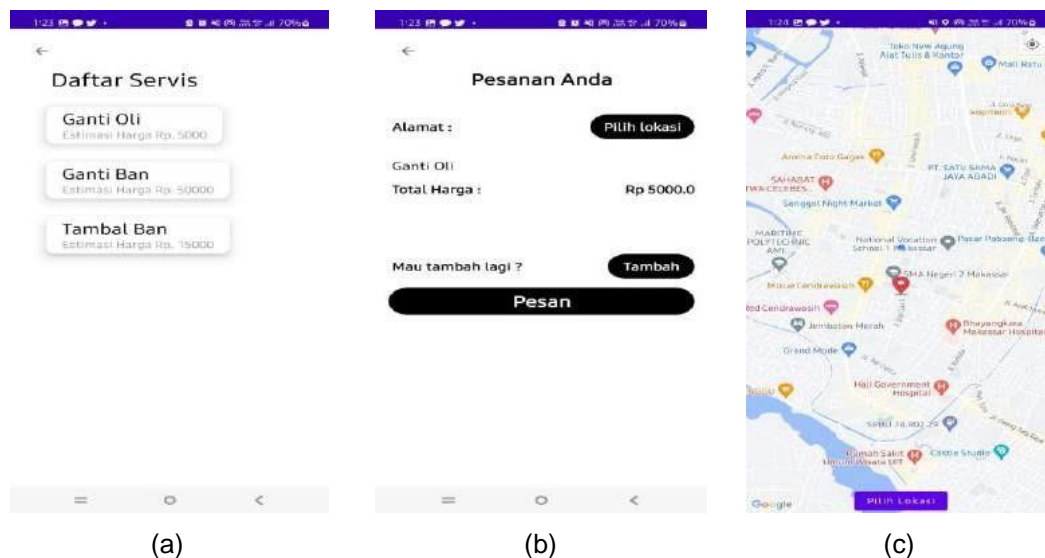


Gambar 11. Contoh Pemetaan *Parametric Diagram* ke Baris Kode

Berdasarkan Gambar 11, dapat dilihat bahwa *parametric diagram* berhasil menghasilkan potongan kode berdasarkan atribut atau parameter meskipun kode tersebut bukan merupakan kode sempurna yang dapat digunakan dalam implementasi sistem.

3.2.3 Pengujian *Black Box*

Pengujian *black box* dilakukan terhadap beberapa fungsionalitas yang dipilih pada aplikasi *Reparation*. Pengujian dilakukan dengan menjalankan aplikasi *Reparation* dan mengamati hasil *output* apakah berhasil atau mengandung kesalahan. Fitur contoh yang dipilih dan diuji yaitu *Pesan Jasa Servis* dan *Tambah Pesanan Jasa Servis*.



Gambar 12. Pengujian pada Fitur *Pesan Jasa Servis* dan *Tambah Pesanan Jasa Servis*

Gambar 12 menunjukkan bahwa pengujian *black box* untuk fitur *Pesan Jasa Servis* berhasil dilakukan dimana sistem menampilkan halaman *Daftar Servis*, *Checkout*, dan peta *Google Maps*.

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan disimpulkan, yaitu:

1. *System Modeling Language* (SysML) dapat diimplementasikan pada pemodelan dan perancangan aplikasi Reparation. Diagram-diagram yang dihasilkan yaitu *activity diagram*, *parametric diagram*, *requirement diagram*, *use case diagram*, dan *sequence diagram*. *Parametric diagram* digunakan untuk menentukan batasan-batasan berupa atribut dari fungsionalitas pada sistem. *Requirement diagram* digunakan untuk menganalisis kebutuhan fungsional dan non-fungsional pada sistem seperti Daftar Akun, Pesan Jasa Servis, Perangkat Pengguna, dan lainnya serta ketertelusuran dari *requirement diagram* ke *use case diagram* dan *parametric diagram*.
2. Hasil pengujian *black box* menunjukkan hasil *output* berhasil dijalankan dengan baik terhadap beberapa fungsionalitas yang diuji seperti Pesan Jasa Servis dan Tambah Pesanan Jasa Servis.

DAFTAR PUSTAKA

- [1] Y. Seby Dwanoko, "Implementasi Software Development Life Cycle (SDLC) dalam Penerapan Pembangunan Aplikasi Perangkat Lunak," *J. Teknol. Inf.*, vol. 7, no. 2, pp. 83–94, 2021, doi: 10.1137/19M1297300.
- [2] R. Inggi, Y. Prayudi, and B. Sugiantoro, "Penerapan System Development Life Cycle (SDLC) dalam Mengembangkan Framework Audio Forensik," *semantIK*, vol. 4, no. 2, pp. 193–200, 2018, doi: 10.5281/zenodo.2528444.
- [3] Lucidchart, "UML Diagrams vs SysML Diagrams," *lucidchart.com*. <https://www.lucidchart.com/blog/uml-diagrams-vs-sysml-diagrams#:~:text=SysML can model a wider,easier to learn and apply.> (accessed Jan. 23, 2023).
- [4] E. Huang, R. Ramamurthy, and L. F. Mcginnis, "System and Simulation Modeling using SysML," *Proc. 2007 Winter Simul. Conf.*, pp. 796–803, 2007, [Online]. Available: <https://ieeexplore.ieee.org/abstract/document/4419675>
- [5] A. Akl, T. Gayraud, and P. Berthou, "Key Factors in Designing In-Flight Entertainment Systems," *Recent Adv. Aircr. Technol.*, pp. 331–360, 2012, doi: 10.5772/37218.
- [6] C. J. J. Paredis *et al.*, "An overview of the SysML-Modelica transformation specification," *20th Annu. Int. Symp. Int. Counc. Syst. Eng. INCOSE 2010*, vol. 1, no. July, pp. 709–722, 2010, doi: 10.1002/j.2334-5837.2010.tb01099.x.
- [7] S. Friedenthal, A. Moore, and R. Steiner, *A Practical Guide to SysML*, 2nd ed. Morgan Kaufmann, 2012. doi: 10.1016/C2010-0-66331-0.
- [8] L. Delligatti, *SysML Distilled: A Brief Guide to the Systems Modeling Language*. Pearson Education, Inc., 2014. [Online]. Available: [https://app.ute.edu.ec/content/4915-114-4-1-6-19/SysML Distilled_ A Brief Guide - Lenny Delligatti.pdf](https://app.ute.edu.ec/content/4915-114-4-1-6-19/SysML%20Distilled_%20A%20Brief%20Guide%20-%20Lenny%20Delligatti.pdf)
- [9] T. Weilkiens, *Systems Engineering with SysML/UML*. Denise E. M. Penrose, 2007. [Online]. Available: [https://www.fullstackrobotics.xyz/Books/Tim Weilkiens - Systems Engineering with SysML and UML.pdf](https://www.fullstackrobotics.xyz/Books/Tim%20Weilkiens%20-%20Systems%20Engineering%20with%20SysML%20and%20UML.pdf)
- [10] A. Morkevicius, S. Gudas, and D. Silingas, "Model-Driven Quantitative Performance Analysis of Updm-Based Enterprise Architecture," *Inf. Technol. Control*, pp. 218–223,

2010.

- [11] D. Blouin, "Modeling Languages for Requirements Engineering and Quantitative Analysis of Embedded Systems," pp. 1–163, 2013, [Online]. Available: https://www.researchgate.net/figure/4-An-example-SysML-parametric-diagram-for-modeling-vehicle-dynamics-from-6_fig10_260433525
- [12] K. Aurin, M. Fajar, and A. Munir, "Pemodelan Jaringan Sensor Nirkabel Menggunakan System Modeling Language," *JTRISTE*, vol. 8, no. 1, pp. 9–20, 2021, [Online]. Available: <https://jurnal.kharisma.ac.id/jtriste/article/view/166/113>
- [13] A. A. Anda and D. Amyot, "Arithmetic Semantics of Feature and Goal Models for Adaptive Cyber-Physical Systems," 2019, p. 13. doi: 10.1109/RE.2019.00034.
- [14] E. Canbulut, V. Aydingül, and B. Yağlıoğlu, "Application of Model-based Systems Engineering with SysML in A Small Satellite Project," in *A/AC*, 2019, p. 20. [Online]. Available: https://www.researchgate.net/profile/Burak-Yaglioglu/publication/336130617_APPLICATION_OF_MODEL-BASED_SYSTEMS_ENGINEERING_WITH_SYSMML_IN_A_SMALL_SATELLITE_PROJECT/links/5d911b1092851c33e9488b8d/APPLICATION-OF-MODEL-BASED-SYSTEMS-ENGINEERING-WITH-SYSML-IN-A-SMALL-SATELLITE-PROJECT.pdf
- [15] D. Hetherinton, "SysML Requirements for Training Game Design," *2014 17th IEEE Int. Conf. Intell. Transp. Syst. ITSC 2014*, pp. 162–167, 2014, doi: 10.1109/ITSC.2014.6957684.
- [16] F. Mhenni, J. Choley, O. Penas, R. Plateaux, and M. Hammadi, "Advanced Engineering Informatics A SysML-based Methodology for Mechatronic Systems Architectural Design," *Adv. Eng. Informatics*, p. 14, 2014, doi: 10.1016/j.aei.2014.03.006.
- [17] S. M. Lee, S. Park, and Y. B. Park, "Self-adaptive system verification based on SysML," *ICEIC 2019 - Int. Conf. Electron. Information, Commun.*, pp. 1–3, 2019, doi: 10.23919/ELINFOCOM.2019.8706383.