

BAB 2 LANDASAN TEORI

2.1 Profile Mamigaya

Mamigaya merupakan sebuah brand dari toko yang menjual berbagai model baju menyusui yang di produksi di sebuah konveksi di daerah bandung jawa barat. Mamigaya telah dirintis sejak tahun 2012. Bisnis baju menyusui ini dijalankan secara *online* melalui berbagai aplikasi *marketplace* seperti Shopee, Tokopedia, Lazada, Facebook, dan juga memasarkannya secara *offline* yang beroperasi di rumah [4].

Baju menyusui yang di jual Mamigaya memiliki berbagai model dan kategori mulai dari *basic* seperti Elsha, Pappi, Sofie. Lalu ada maxidress seperti Izza Maxi Series, Abaya Series, Aulia Maxi Series, Aisyah Gradasi Merah, Reina Denim Square, Midi Kania, dan masih banyak lagi [4]. Berdasarkan hasil dari wawancara yang telah dilakukan dengan Mamigaya, berbagai macam model baju menyusui yang di jual oleh Mamigaya di buat dengan bahan yang nyaman dan mengikuti gaya ibu muda sehingga baju menyusui tersebut dapat digunakan juga untuk bekerja, dan produk baju menyusui Mamigaya dipasarkan dengan harga terjangkau kisaran harga antara Rp50.000 hingga Rp200.000.

Customer Mamigaya berasal dari banyak kalangan, Mamigayapun saat ini pemasarannya sudah mencakup hampir semua wilayah di Indonesia, dan sudah merambah ke manca negara seperti Singapura, Malaysia, Belanda, dan Jerman yang tertarik dengan model mamigaya [4]. Saat ini Mamigaya sendiri memiliki 13,9 ribu pengikut pada salah satu *marketplace* yaitu Shopee. Pada aplikasi Instagram Mamigaya memiliki 51,2 ribu pengikut dan postingan foto produk sebanyak 1.739 yang di posting pada akun pertama, serta memiliki 23,8 ribu pengikut dan postingan foto produk sebanyak 2.108 yang di posting pada akun kedua.

2.2 Baju Menyusui

Baju menyusui adalah pakaian yang dirancang khusus yang dimaksudkan untuk membuat menyusui menjadi lebih mudah dan nyaman. Baju menyusui adalah pakaian yang dirancang khusus yang dimaksudkan untuk membuat menyusui menjadi lebih mudah dan nyaman. Pakaian menyusui dirancang untuk membuat menyusui lebih mudah dan lebih tersembunyi, karena sering kali memiliki bukaan

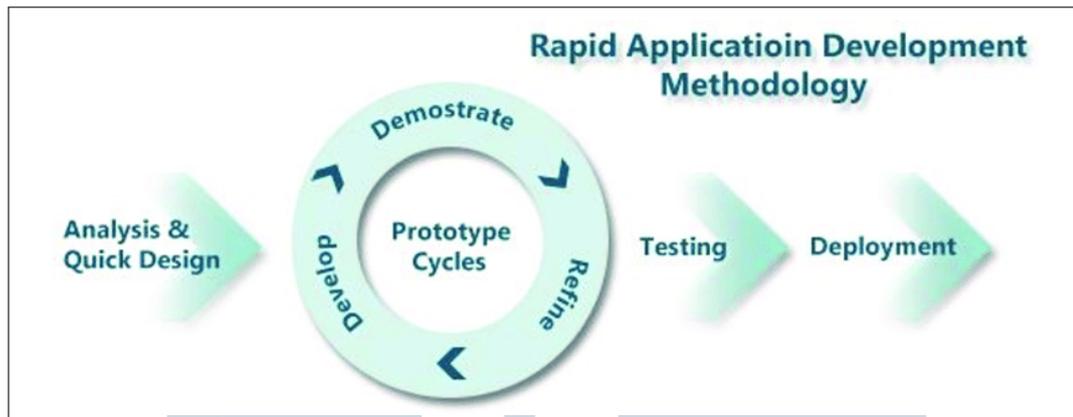
yang tersembunyi untuk memudahkan akses ke payudara. Pakaian menyusui sering menampilkan pakaian berlapis, seperti tanktop, bra menyusui, dan atasan menyusui, untuk memberikan akses dan perlindungan yang lebih baik.

Dengan *website e-commerce*, pengguna dapat menghemat waktu dan uang dengan menghindari pergi ke toko. Ini juga memungkinkan pengguna untuk meningkatkan pelayanan pelanggan dan meningkatkan loyalitas mereka. Dalam jurnal yang diterbitkan oleh institut pengembangan manajemen dan teknologi di India, dinyatakan bahwa *website e-commerce* memberi manfaat untuk semua orang yang terlibat dalam proses jual beli, termasuk produsen, pedagang, dan konsumen [9].

2.3 Rapid Application Development

Metode *rapid application development* merupakan suatu pendekatan berorientasi objek yang menghasilkan sebuah sistem dengan tujuan mempersingkat waktu pengerjaan aplikasi dan proses agar sesegera mungkin memberdayakan sistem perangkat lunak tersebut secara tepat dan cepat [10]. RAD merupakan pengembangan dan bagian dari SDLC adalah metodologi rekayasa perangkat lunak yang berfokus pada pembuatan aplikasi dengan cepat dan efisien. Pendekatan ini berfokus pada siklus pengembangan yang lebih pendek, komponen yang dapat digunakan kembali, dan alat pengembangan tingkat tinggi. RAD juga menekankan pengembangan berulang, memungkinkan proses yang lebih fleksibel dan responsif terhadap perubahan kebutuhan user.

Untuk merancang suatu sistem informasi yang membutuhkan waktu setidaknya minimal 180 hari, harus melawati tahapan yang memakan waktu cukup panjang, maka dengan menerapkan metode *rapid application development* hanya membutuhkan waktu 30-90 hari untuk menyelesaikan sistem perangkat lunak tersebut[10]. Metode ini menekankan keterlibatan pengguna dalam proses analisis dan perencanaan sehingga dapat memenuhi kebutuhan pengguna dengan baik sehingga dapat membuat kepuasan pengguna sistem meningkat.



Gambar 2.1. Diagram RAD

[11]

2.3.1 Analysis Quick Design

Fase pertama RAD adalah memahami persyaratan sistem (*analysis quick design*). Membutuhkan pengguna akhir tingkat tinggi atau berpengalaman luas untuk menentukan apa fungsi sistem yang seharusnya. Sehingga menjadi diskusi terstruktur tentang masalah bisnis yang perlu dipecahkan. Fase proses ini termasuk memutuskan bahasa pemrograman dan basis data apa yang perlu digunakan. Skrip PHP dan *database* MySQL digunakan sebagai alat pengembangan untuk mengembangkan prototipe.

PHP sendiri merupakan bahasa *scripting* yang awalnya dirancang untuk menghasilkan halaman web yang dinamis. Hal ini berevolusi untuk menyertakan kemampuan antarmuka baris perintah dan dapat digunakan dalam aplikasi grafis. Pada fase ini juga, struktur keseluruhan perangkat lunak ditentukan. Penting untuk memahami persyaratan sistem sebelum melanjutkan ke pengembangan prototipe[11].

2.3.2 Prototype Cycles

Fase kedua merupakan pengulangan dari fase pengembangan *prototype* yaitu *development*, *demonstration*, *design*. Hal ini termasuk membuat desain fisik untuk *database* dan terutama berfokus pada penerjemahan desain ke dalam kode pemrograman. Kode untuk menghubungkan dari bahasa pemrograman ke MySQL *database management system* (DBMS) yang dibuat[11].

- *Demonstrate* Tahap ini adalah tahap dimana *prototype* pertama dibuat dan

diperlihatkan kepada pengguna untuk mendapatkan *feedback* awal. *Prototype* pertama ini dibuat untuk menguji ide secara kasar, dan untuk menentukan fitur apa saja yang perlu ditingkatkan atau diubah.

- *Refine* Tahap ini melibatkan penyempurnaan *prototype* setelah *feedback* awal diterima. *Prototype* akan diperbaiki dan dimodifikasi berdasarkan umpan balik dari pengguna, sehingga *prototype* menjadi lebih baik dan sesuai dengan kebutuhan pengguna.
- *Develop* Tahap ini adalah tahap dimana *prototype* yang telah disempurnakan dikembangkan menjadi produk akhir. Tahap ini melibatkan pemrograman dan pengujian terhadap aplikasi hingga aplikasi siap untuk dirilis.

2.3.3 Testing

Tahap ini merupakan tahap uji coba aplikasi yang telah dibangun. Dalam tahap ini, aplikasi diuji secara menyeluruh untuk memastikan bahwa ia berfungsi dengan baik dan sesuai dengan kebutuhan pengguna. Uji coba meliputi pengujian fungsionalitas aplikasi dan pengujian kepuasan pengguna. Setelah uji coba selesai, aplikasi dapat ditingkatkan dan diperbaiki sesuai dengan hasil pengujian. Proses selanjutnya melibatkan peningkatan dan koreksi kesalahan *prototype*. Tahap ini diulang sampai *prototype* memenuhi tujuan penelitian.

2.3.4 Deployment

Tahap ini adalah tahap akhir dari pengembangan perangkat lunak. Dalam tahap ini, aplikasi dikembangkan dan disempurnakan sesuai dengan persyaratan dan umpan balik pengguna. Setelah aplikasi selesai dikembangkan, aplikasi dapat dirilis dan didistribusikan ke pengguna akhir. Namun, pengembangan aplikasi belum berakhir pada tahap ini, karena biasanya perlu dilakukan pemeliharaan dan peningkatan selama masa pakai aplikasi.

RAD menggunakan sejumlah alat dan teknik untuk merampingkan proses pengembangan. Ini termasuk pembuatan *prototype*, pengembangan berulang, dan pengembangan berbasis komponen. *Prototype* memungkinkan pengembang untuk menguji versi awal aplikasi dengan cepat, sehingga *feedback* user dapat diintegrasikan di awal proses. Pengembangan berulang memungkinkan pengembang untuk membuat versi dasar aplikasi dan kemudian secara bertahap

menambahkan lebih banyak fitur dan fungsionalitas. Pengembangan berbasis komponen memungkinkan pengembang untuk menggunakan kembali komponen dan modul dalam aplikasi yang berbeda, membuat proses pengembangan menjadi lebih cepat dan lebih efisien.

Secara keseluruhan, RAD adalah metodologi yang efisien dan efektif untuk pengembangan perangkat lunak. Ini memungkinkan pengembang untuk membuat aplikasi dengan cepat dan menanggapi kebutuhan pelanggan [12]. Dalam metode RAD memiliki karakteristik yang berbeda, kelebihan dan kekurangan dari setiap metode SDLC lainnya, sehingga pemilihan metode harus sesuai dengan penelitian yang dilakukan.

Tabel 2.1. Metode *Software Development Life Cycle*

SDLC	Karakteristik	Kelebihan	kekurangan
<i>Rapid Application Development (RAD)</i>	Metode RAD berfokus pada siklus pengembangan sistem yang sangat singkat dan oleh karena itu disebut sebagai perkembangan linier. Waktu pengembangan sistem metode RAD kurang lebih 30 hingga 90 hari	Kebutuhan aplikasi bisa berubah sewaktu-waktu. Aplikasi dikembangkan berdasarkan kebutuhan dan keinginan user. Memperkecil kemungkinan error dan hal buruk lainnya. Waktu pengembangan aplikasi bisa lebih cepat dan efektif.	Hanya cocok untuk proyek yang waktunya singkat. Hanya cocok digunakan untuk mengembangkan aplikasi secara modular (fokus ke suatu fitur untuk dijadikan module terpisah). Sulit diterapkan untuk mengembangkan aplikasi besar.

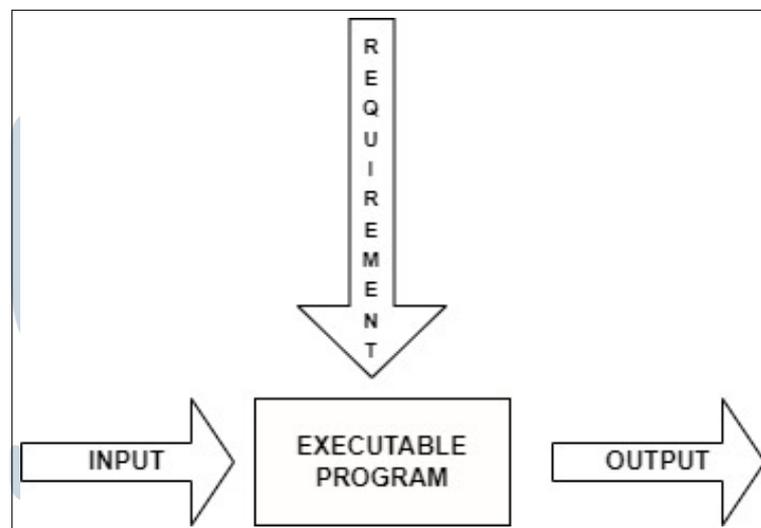
UNIVERSITAS
MULTIMEDIA
NUSANTARA

SDLC	Karakteristik	Kelebihan	kekurangan
Prototype	Melibatkan pembuatan model awal dari sistem yang akan dikembangkan. Prototipe digunakan untuk mendapatkan pemahaman yang lebih baik tentang persyaratan dan ekspektasi pengguna. Setelah umpan balik diterima, prototipe dapat dimodifikasi dan iterasi berulang dilakukan hingga mencapai solusi yang diinginkan.	Interaksi langsung dengan pengguna akhir dan pemangku kepentingan untuk memahami persyaratan dengan lebih baik. Dapat membantu dalam mengurangi risiko pengembangan dan meningkatkan kepuasan pengguna dengan menghasilkan solusi yang lebih sesuai dengan harapan.	Kesulitan dalam menangani persyaratan yang kompleks atau ketat. Proses iterasi berulang dapat memakan waktu dan sumber daya yang signifikan, terutama jika tidak terdapat kendali yang baik atas perubahan persyaratan.
V-Shaped	Pengembangan perangkat lunak yang linear dan berurutan. Setiap tahap pengembangan (seperti perencanaan, analisis, desain, pengujian, dll.) diikuti oleh tahap verifikasi yang sesuai. Pendekatan ini menekankan pentingnya perencanaan yang matang sebelum memasuki tahap pengembangan yang sebenarnya.	Pendekatannya yang terstruktur dan terdokumentasi dengan baik. Tahap verifikasi yang kuat membantu memastikan bahwa setiap tahap pengembangan diuji dengan baik dan sesuai dengan persyaratan yang ditetapkan sebelumnya.	Kurang fleksibel dan sulit beradaptasi jika terjadi perubahan persyaratan atau kesalahan yang terdeteksi di tahap akhir pengembangan. Proses pengujian yang terjadi di akhir siklus juga bisa menjadi risiko jika terdapat masalah yang signifikan yang hanya terungkap pada tahap tersebut.

Berdasarkan dari perbandingan tabel 2.1 dapat di simpulkan bahwa metode RAD cenderung lebih cocok pada penelitian ini dibandingkan metode *Prototype* dan *V-shape*. Hal ini dapat disimpulkan karena metode RAD fokus pada siklus pengembangan sistem yang sangat singkat antara 30 hingga 90 hari, serta memperkecil kemungkinan error saat berubah sewaktu waktu.

2.4 Blackbox Testing

Blackbox Testing adalah teknik pengujian perangkat lunak. Dia digunakan untuk menentukan fungsionalitas aplikasi. Fokus utama *Blackbox Testing* adalah masukan yang tersedia untuk sebuah aplikasi dan *output* yang diharapkan untuk setiap nilai *input*. Metode pengujian ini didasarkan pada kebutuhan perangkat lunak dan spesifikasi. Ini adalah teknik pengujian perangkat lunak dimana cara kerja sistem internal yang diuji tidak diketahui oleh penguji. Hal ini juga disebut pengujian dan perilaku berbasis spesifikasi pengujian. Teknik ini dinamakan demikian karena dalam pengujian ini, tester tidak perlu tahu tentang kode internal pelaksanaan aplikasi. Pengujian ini menangani keduanya valid dan input yang tidak valid sesuai dengan pelanggan persyaratan [13].



Gambar 2.2. Representation of Black Box Testing

[13].

2.5 Skala Likert

Metode *skala likert* merupakan metode yang digunakan untuk mengukur tingkat kepuasan pengguna menggunakan *skala likert* [14]. Pada penelitian ini skala *likert* digunakan untuk mengukur pendapat *user* terhadap *website* penjualan baju menyusui yang telah dibuat. Berikut merupakan tingkat skala beserta *score* yang digunakan:

Tabel 2.2. Table Skala Score untuk Jawaban Kuisisioner .

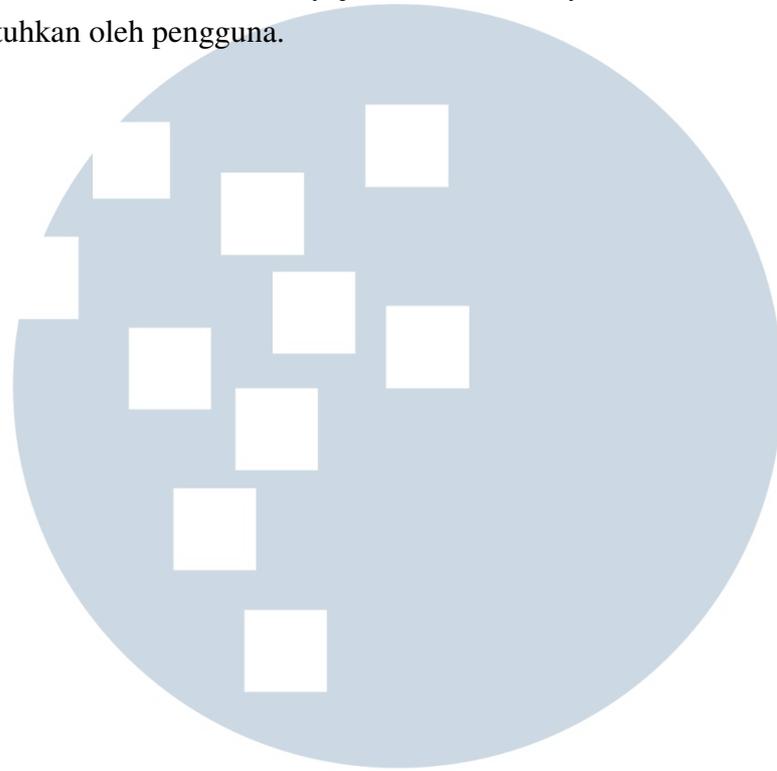
No	Pilihan Jawaban	Score
1	Sangat Setuju	5
2	Setuju	4
3	Netral	3
4	Tidak Setuju	2
5	Sangat Tidak Setuju	1

2.6 End-User Computing Satisfaction (EUCS)

End User Computing Satisfaction merupakan penilaian secara keseluruhan dari penggunaan sistem informasi yang didasarkan pengalaman mereka dalam menggunakan sistem tersebut [15]. *End User Computing Satisfaction* memiliki 5 aspek yang berisi isi (*content*), keakuratan (*accuracy*), bentuk (*format*), kemudahan penggunaan (*ease of use*) dan ketepatan waktu (*timeliness*).

1. Isi (*Content*), mengukur kepuasan pengguna dengan isi dari sistem. Untuk mengisi sistem biasanya terdiri dari fungsi dan modul yang dapat digunakan oleh pengguna sistem, serta informasi yang dihasilkan oleh sistem.
2. Akurasi (*Accuracy*), mengukur kepuasan pengguna berdasarkan akurasi data ketika sistem menerima *input*, sistem memprosesnya sebagai informasi.
3. Bentuk (*Format*), mengukur kepuasan pengguna dalam hal tampilan dan estetika antarmuka pengguna sistem.
4. Kemudahan pengguna (*Ease of Use*), mengukur kepuasan pengguna dengan situs kemudahan penggunaan atau kemudahan penggunaan saat menggunakan sistem, pemasukan data, pengolahan data dan pencarian data yang diperlukan.

5. Ketepatan (*timeliness*), mengukur kepuasan pengguna dengan situs ketepatan waktu sistem dalam menyajikan atau menyediakan informasi yang dibutuhkan oleh pengguna.



UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA