

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Penelitian terdahulu mengenai sistem informasi web digunakan sebagai referensi dalam pengembangan website e-commerce untuk Central Printing. Beberapa penelitian terdahulu memberikan landasan yang relevan, baik dari segi metode pengembangan, implementasi teknologi, maupun pemecahan masalah yang dihadapi.

Penelitian oleh (Prihandono & Amir, 2024) membahas implementasi teknologi informasi untuk meningkatkan efisiensi dan daya saing perusahaan. Penelitian memberikan wawasan tentang pentingnya mengelola teknologi informasi dalam mendukung pengambilan keputusan dan meningkatkan komunikasi dalam organisasi. Metodologi yang digunakan adalah *kualitatif* dengan pendekatan studi keputusan. Masalah yang diangkat adalah rendahnya efisiensi operasional dan daya saing perusahaan dalam menghadapi persaingan global. Tujuan dari penelitian ini adalah menganalisis dampak bahwa teknologi informasi mampu mendukung proses, meningkatkan komunikasi, dan mempercepat pengambilan keputusan, sehingga mendukung daya saing perusahaan. Hasil penelitian ini untuk dapat menghasilkan teknologi informasi yang mampu memfasilitasi proses, meningkatkan komunikasi, dan mempercepat pengambilan keputusan, sehingga meningkatkan daya saing.

Penelitian oleh (Putri & Suherman, 2024) berfokus pada strategi pemasaran berbasis *e-commerce* untuk meningkatkan akses pasar di industri percetakan. Penelitian ini menunjukkan pentingnya adaptasi teknologi dalam memperluas jangkauan geografis dan meningkatkan kepuasan pelanggan. Metodologi yang digunakan adalah *Kualitatif*. Masalah yang diangkat adalah terbatasnya jangkauan pasar pada pemasaran konvensional di perusahaan percetakan. Tujuan dari penelitian ini adalah mengembangkan strategi *e-commerce* yang efektif untuk meningkatkan daya pasar bisnis. Hasil penelitian menunjukkan bahwa penerapan *e-commerce* mampu memperluas akses pasar secara geografis dan meningkatkan

kepuasan pelanggan, meskipun tantangan seperti persaingan dan keamanan transaksi online tetap menjadi perhatian.

Penelitian oleh (Prilantana dkk. 2021) membahas pengembangan sistem *e-commerce* berbasis web untuk membantu proses pemasaran di Concordia Music Shop. Penelitian ini memberikan solusi digital untuk meningkatkan efektivitas operasional bisnis. Metodologi yang digunakan *Rapid Application Development (RAD)*. Masalah yang diangkat kurangnya platform digital untuk pemasaran dan transaksi produk *custom*. Tujuan dari penelitian ini adalah mengembangkan *e-commerce* untuk meningkatkan jangkauan pasar penjualan alat musik secara *online*. Hasil dari penelitian ini sistem *e-commerce* yang dirancang berhasil memenuhi kebutuhan bisnis dengan *user experience*, mendukung transaksi, dan pemesanan secara *online*.

Penelitian oleh (Putra & Suprianto, 2024) bertujuan untuk mengatasi kesalahan dalam pengelolaan data penjualan dan meningkatkan efisiensi di toko ritel. Penelitian ini relevan dalam memberikan solusi digital untuk manajemen data penjualan. Metodologi yang digunakan adalah *kuantitatif*. Masalah yang diangkat adalah pengelolaan data yang manual dan rentan kesalahan dalam proses penjualan. Tujuan dari penelitian ini adalah mengembangkan sistem berbasis web untuk meningkatkan akurasi dan efisiensi pengelolaan data transaksi penjualan. Hasil penelitian menunjukkan bahwa sistem berbasis web mampu mengurangi kesalahan pencatatan data dan meningkatkan efisiensi operasional toko. Hasil dari penelitian ini sistem berbasis web memudahkan pengelolaan data, mengurangi kesalahan manual, dan meningkatkan akurasi serta efisiensi operasional toko ritel.

Tabel 2.1 Perbandingan Penelitian Terdahulu

Penulis	Metodologi	Masalah	Tujuan	Hasil
(Prihandono & Amir, 2024)	<i>Kualitatif</i>	Rendahnya efisiensi organisasi dan daya saing perusahaan.	Menganalisis peran teknologi informasi dalam meningkatkan efisiensi organisasi dan daya saing perusahaan.	Teknologi informasi mampu mendukung proses, meningkatkan komunikasi, dan mempercepat pengambilan keputusan, sehingga meningkatkan daya saing.
(Putri & ...)	<i>Kualitatif</i>	Jangkauan pemasaran	Mengembangkan strategi	Strategi <i>e-commerce</i> efektif memperluas

Penulis	Metodologi	Masalah	Tujuan	Hasil
Suherman, (2024)		yang terbatas pada perusahaan percetakan.	pemasaran berbasis <i>e-commerce</i> untuk memperluas jangkauan pasar dan meningkatkan kepuasan pelanggan.	pasar secara geografis, meningkatkan aksesibilitas, dan kepuasan pelanggan meski tantangan persaingan tetap ada.
(Prilantana dkk. 2021)	<i>Rapid Application Development (RAD)</i>	Kurangnya platform digital untuk pemasaran produk gitar <i>custom</i> .	Mengembangkan sistem <i>e-commerce</i> berbasis web yang dapat melakukan <i>custom</i> pemesanan produk gitar, memperluas jangkauan pasar, dan meningkatkan peluang bisnis	Sistem <i>e-commerce</i> yang dirancang berhasil memenuhi kebutuhan bisnis dengan <i>user experience</i> , mendukung transaksi, dan pemesanan secara <i>online</i> .
(Putra & Suprianto, 2024)	<i>Kuantitatif</i>	Ketidaksiapan daya keuangan akibat pengelolaan manual pada toko ritel.	Mengembangkan sistem berbasis web untuk meningkatkan efisiensi pengelolaan data penjualan dan integrasi data di toko ritel.	Sistem berbasis web memudahkan pengelolaan data, mengurangi kesalahan manual, dan meningkatkan akurasi serta efisiensi operasional toko ritel.

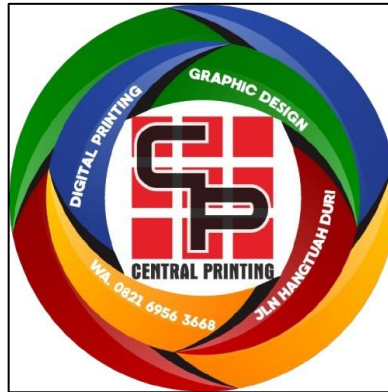
Penelitian oleh Deni Alfarouqi (2024) menggunakan metodologi *Rapid Solo Software Development (RSSD)* untuk mengatasi masalah pengelolaan pesanan, pencatatan transaksi yang tidak akurat, dan jangkauan pasar yang terbatas. Penelitian ini bertujuan mengembangkan sistem berbasis web yang terintegrasi untuk pengelolaan pesanan, pencatatan transaksi, dan pemasaran secara digital. Hasil dari penelitian ini adalah sistem informasi penjualan yang mendukung efisiensi operasional dan perluasan digital berbasis web.

Berdasarkan penelitian terdahulu yang telah dibahas, dapat disimpulkan bahwa terdapat berbagai solusi berbasis web untuk mendukung efisiensi operasional dan pemasaran. Namun, penelitian oleh (Prihandono & Amir, 2024) hanya fokus pada peningkatan efisiensi organisasi tanpa meningkatkan fitur penjualan *online*. Penelitian oleh (Putri & Suherman, 2024) menekankan strategi pemasaran berbasis *e-commerce* tetapi tidak mencakup pengelolaan transaksi dan pencatatan stok. Penelitian oleh (Prilantana dkk. 2021) mengembangkan sistem *e-commerce* berbasis web tetapi terbatas pada produk *custom* seperti alat musik. Penelitian oleh (Putra & Suprianto, 2024) berfokus pada pengelolaan data penjualan toko ritel tanpa perluasan fitur pemasaran digital. Dari semua penelitian terdahulu yang telah diuraikan, belum ada yang secara khusus mengintegrasikan pengelolaan pesanan, pencatatan transaksi, dan pemasaran digital berbasis web seperti yang dirancang dalam proyek ini. Keunggulan penelitian ini terletak pada penerapan metodologi *Rapid Solo Software Development (RSSD)*, yang memungkinkan pengembangan sistem secara efisien dan terfokus untuk memenuhi kebutuhan Central Printing. Penelitian ini diharapkan dapat memberikan kontribusi dalam meningkatkan efisiensi operasional, akurasi pencatatan, dan perluasan jangkauan pasar melalui platform *e-commerce* berbasis web.

2.2 Landasan Teori

2.2.1 Central Printing

Central Printing adalah sebuah perusahaan percetakan yang didirikan pada tahun 2009 dan berlokasi di Duri, Riau. Perusahaan ini memiliki spesialisasi dalam layanan cetak seperti buku, poster, kartu nama, brosur, dan katalog. Hingga saat ini, Central Printing telah mengembangkan bisnisnya dengan membuka empat cabang di berbagai wilayah. Selain menyediakan layanan percetakan, Central Printing juga menjadi mitra strategis bagi bisnis lokal, institusi pendidikan, dan individu yang membutuhkan produk cetak berkualitas. Perusahaan ini mengintegrasikan teknologi modern untuk menghasilkan produk cetak dengan hasil yang akurat dan efisien.



Gambar 2.1 Logo Central Printing

2.2.2 Percetakan

Percetakan merupakan aktivitas penggandaan teks maupun gambar secara massal dengan menggunakan metode dan teknologi tertentu untuk memenuhi kebutuhan dalam komunikasi visual. Perkembangan *digital printing* telah memberikan dampak besar terhadap industri percetakan melalui penyediaan hasil cetak berkualitas tinggi dan efisiensi biaya, khususnya untuk jumlah cetak skala kecil hingga menengah (Ritonga dkk., n.d.). Inovasi ini mendukung berbagai kebutuhan usaha, seperti produksi brosur, kartu nama, dan kemasan produk dengan desain yang dapat disesuaikan. Selain itu, industri percetakan memiliki peran penting dalam strategi pemasaran dan penyampaian pesan visual, di mana kecepatan dan kualitas hasil cetak menjadi aspek penting dalam menarik perhatian konsumen (Azhary & Cempena, 2024). Perkembangan teknologi tinta dan bahan cetak juga berkontribusi terhadap keberlanjutan industri dengan menghadirkan opsi material yang lebih ramah lingkungan.



Gambar 2.2 Printing

2.2.3 E-commerce

E-Commerce merupakan penerapan teknologi, perangkat lunak, dan proses bisnis yang terintegrasi guna mendukung pelaksanaan transaksi secara elektronik

antara pelaku usaha, konsumen, maupun komunitas tertentu. Dalam praktiknya, *e-commerce* tidak hanya terbatas pada aktivitas jual beli produk atau layanan, tetapi juga mencakup pertukaran informasi yang diperlukan untuk kelancaran transaksi digital. Sebagai bagian dari perkembangan teknologi informasi, *e-commerce* memungkinkan terjadinya interaksi antara penjual dan pembeli melalui jaringan internet tanpa memerlukan pertemuan secara langsung, sehingga memberikan kemudahan dan efisiensi dalam proses perdagangan (Asyifah dkk. 2023). Sistem yang dibangun pada penelitian ini menerapkan karakteristik tersebut melalui ketersediaan fitur pencarian dan filter produk, keranjang belanja, proses checkout, pengelolaan akun pengguna, laporan penjualan, serta pelacakan status pengiriman, sehingga tergolong sebagai website *e-commerce*.



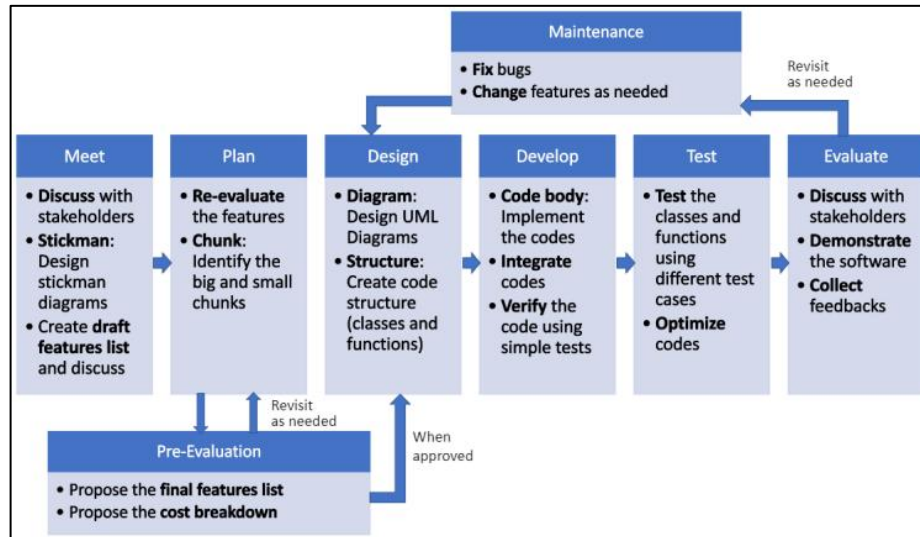
Gambar 2.3 *e-Commerce*

2.2.4 *Rapid Solo Software Development (RSSD)*

Rapid Solo Software Development (RSSD) adalah sebuah metodologi yang dirancang khusus untuk pengembang perangkat lunak individu guna mengatasi tantangan umum seperti keterbatasan sumber daya dan kurangnya kolaborasi dalam proses pengembangan. Metodologi ini dikembangkan oleh Kristo Radion Purba dan Rusyaizila Ramli dari University of Southampton Malaysia, serta mengadaptasi tahapan dasar dari pendekatan *Agile*, yakni pertemuan, perencanaan, perancangan, pengembangan, pengujian, dan evaluasi. *RSSD* menghadirkan keunggulan dalam hal efisiensi, modularitas, dan kemudahan peninjauan ulang (*revisitabilitas*). Efisiensi tercapai dengan menyederhanakan tahapan perencanaan, sementara modularitas memungkinkan kode disusun dalam bagian-bagian kecil yang berdiri sendiri. Kemampuan untuk ditinjau ulang memudahkan pengembang memahami kembali kode dan dokumentasi secara lebih sistematis. Dari hasil penerapan *RSSD* pada sepuluh proyek, diperoleh rata-rata tingkat kepuasan sebesar 4,19 dari 5,0,

yang menunjukkan bahwa metodologi ini dinilai efektif dalam membantu pengembangan perangkat lunak secara lebih terstruktur (Purba & Ramli, 2022).

Metode ini dipilih karena dirancang khusus untuk membantu pengembang tunggal (*solo developer*) dalam mengembangkan perangkat lunak secara efisien, modular, dan *revisitable*. Berikut merupakan gambar tahapannya:



Gambar 2.4 RSSD

1) *Meet* (Pertemuan Awal)

- Diskusi dengan *stakeholder* untuk memahami kebutuhan bisnis dan sistem yang akan dikembangkan.
- Menggunakan diagram *stickman* sebagai alat bantu visual untuk menggambarkan fitur-fitur utama yang diharapkan.
- Menyusun daftar awal fitur yang akan dikembangkan dan mengonfirmasinya dengan pemangku kepentingan.

2) *Plan & Pre-Evaluation* (Perencanaan & Pra-Evaluasi)

Pada tahap *Planning & Pre-Evaluation*, dilakukan perencanaan dan evaluasi awal fitur sistem untuk memastikan *relevansi* dan prioritasnya. Proses ini mencakup *Re-Evaluate* (meninjau kembali fitur yang telah ditentukan), *Chunking* (memecah fitur besar menjadi bagian lebih kecil untuk memudahkan pengembangan), dan *Pre-Evaluation* (menyusun daftar fitur final serta memperkirakan biaya dan sumber daya yang dibutuhkan).

- ***Re-Evaluate***: Evaluasi ini menunjukkan bahwa pengembangan sistem akan berfokus pada fitur-fitur kategori *Must Have* untuk memastikan

sistem dapat berfungsi dengan baik dan memenuhi kebutuhan utama pengguna. Fitur kategori *Should Have* akan dipertimbangkan untuk pengembangan tahap kedua, sedangkan fitur kategori *Could Have* dapat menjadi bagian dari peningkatan sistem jangka panjang. Dengan pendekatan ini, pengembangan sistem dapat dilakukan secara terstruktur dan efisien, sesuai dengan prioritas kebutuhan pengguna dan tujuan utama sistem.

- **Chunking:** Tahap ini bertujuan untuk membagi fitur utama menjadi bagian-bagian kecil yang lebih terstruktur dan dapat dikerjakan secara bertahap. Dengan *chunking*, pengembangan menjadi lebih efisien dan memungkinkan pengujian serta evaluasi setiap bagian secara independen sebelum integrasi ke dalam sistem utama.
- **Pre-Evaluation:** Menyusun proposal final fitur yang akan dikembangkan, serta melakukan estimasi biaya pengembangan.

3) *Design* (Perancangan)

Pada tahap *Design*, dilakukan perancangan sistem untuk mendefinisikan struktur, alur kerja, dan tampilan antarmuka pengguna. Proses ini mencakup pembuatan diagram seperti *Use Case Diagram*, *Entity Relationship Diagram (ERD)*, dan *Activity Diagram* untuk menggambarkan hubungan antar komponen serta alur proses dalam sistem. Selain itu, tahap ini juga mencakup penyusunan struktur sistem agar pengembangan berjalan lebih efisien dan terorganisir.

- **Diagram:** Mendesain *diagram UML* untuk memahami struktur sistem.
- **Structure:** Dilakukan perancangan arsitektur sistem dengan menentukan struktur kode yang terdiri dari kelas dan fungsi. Struktur ini dirancang agar sistem dapat dikembangkan secara modular dan mudah dikelola. Dengan pendekatan ini, setiap bagian dari sistem dapat dikembangkan dan diuji secara independen sebelum diintegrasikan secara keseluruhan. Struktur kode yang baik memungkinkan pengembang untuk lebih mudah melakukan perbaikan, pemeliharaan, dan pengembangan fitur baru di masa mendatang.

4) *Develop* (Pengembangan)

Setelah tahap *Design* selesai selanjutnya proses pengembangan kode dimulai. Pengembangan dilakukan berdasarkan desain sistem yang telah

dirancang, dengan mengikuti daftar fitur yang telah disepakati dalam *Final Feature List*. Proses ini dilakukan secara modular, memastikan bahwa setiap komponen dapat diuji dan diintegrasikan secara independen sebelum digabungkan menjadi satu sistem yang utuh.

- Implementasi Kode (*Code Body*)

Pada tahap ini, pengembang mulai menulis kode utama sesuai dengan desain yang telah disetujui. Setiap fitur dikembangkan secara bertahap hingga semua fungsi dalam sistem siap diimplementasikan. Pengembang juga harus memastikan bahwa kode yang ditulis sesuai dengan standar yang telah ditetapkan agar mudah dipahami, dipelihara, dan diintegrasikan.

- Integrasi Komponen (*Integrate*)

Setelah setiap modul dikembangkan, tahap integrasi dilakukan untuk memastikan semua komponen dapat bekerja sama dengan baik. Proses ini mencakup penyambungan modul, sinkronisasi data antar bagian sistem, serta validasi alur kerja untuk menghindari error atau konflik antar fitur. Jika ditemukan ketidaksesuaian atau bug, perbaikan akan dilakukan sebelum melanjutkan ke tahap berikutnya.

- Verifikasi (*Verify*)

Pada tahap ini, kode yang telah dikembangkan dan diintegrasikan akan diverifikasi untuk memastikan bahwa setiap bagian sistem bekerja sesuai dengan desain yang telah dirancang. Proses ini mencakup pemeriksaan struktur kode, koneksi antar modul, serta validasi alur sistem. Dengan verifikasi ini, diharapkan tidak ada kesalahan yang berlanjut ke tahap pengujian lebih lanjut, sehingga sistem lebih stabil dan siap untuk diuji secara menyeluruh.

5) *Test* (Pengujian)

- **Test Cases:** Sistem diuji menggunakan *Test Cases* yang telah dirancang sebelumnya. Pengujian dilakukan dengan metode *Black-Box* Testing, yang berfokus pada pemeriksaan apakah hasil keluaran sistem sesuai dengan masukan yang diberikan, tanpa melihat struktur kode internal.

- **Optimize:** Mengoptimalkan kode untuk meningkatkan efisiensi sistem.

6) *Evaluate* (Evaluasi)

Tahap evaluasi dilakukan untuk menilai hasil akhir sistem yang telah diuji dan dioptimalkan. Evaluasi ini bertujuan untuk memastikan bahwa sistem memenuhi standar yang diharapkan, baik dari segi fungsionalitas, performa, maupun kepuasan pengguna.

- **Discuss**

Pada tahap ini, dilakukan diskusi antara tim pengembang, pemangku kepentingan, dan pengguna untuk mengevaluasi hasil pengembangan sistem. Diskusi ini bertujuan untuk mengidentifikasi kelebihan, kekurangan, serta peluang perbaikan dalam sistem sebelum penerapan lebih lanjut.

- **Demonstrate**

Sistem yang telah dikembangkan akan diperlihatkan dalam bentuk demonstrasi kepada pengguna atau pihak terkait. Tujuan dari demonstrasi ini adalah untuk memastikan bahwa semua fitur berfungsi sebagaimana mestinya, serta untuk mendapatkan umpan balik awal sebelum sistem diterapkan secara luas.

- **Collect**

Pada tahap ini, dilakukan pengumpulan data dari hasil diskusi dan demonstrasi. Data yang dikumpulkan meliputi umpan balik pengguna, laporan bug yang masih ditemukan, serta saran perbaikan yang diperlukan untuk peningkatan sistem lebih lanjut. Informasi ini akan digunakan sebagai dasar untuk melakukan perbaikan dan pengembangan sistem di masa mendatang.

7) *Maintenance* (Pemeliharaan)

- **Fix Bugs:** Memperbaiki kesalahan atau *bug* dalam sistem setelah implementasi.
- **Change Features:** Melakukan perubahan atau penyesuaian fitur sesuai kebutuhan yang berkembang.

2.2.5 *Evaluasi Keberhasilan Implementasi Metode*

Evaluasi implementasi merupakan tahapan krusial untuk menilai sejauh mana perangkat lunak yang dikembangkan secara riil mampu memenuhi rancangan kebutuhan fungsional, estimasi waktu pengerjaan, serta standar kualitas yang telah ditetapkan sebelumnya. Menurut (Lestari dkk., 2026), pengujian pada batas prototipe atau desain semata tidak cukup untuk menjamin keberhasilan sistem saat dioperasikan pada kondisi nyata. Oleh karena itu, evaluasi implementasi bertujuan untuk mengukur tingkat realisasi sistem fungsional, mengidentifikasi hambatan teknis maupun operasional selama proses transisi desain ke sistem riil, serta memastikan bahwa metode pengembangan yang digunakan mampu menjaga efisiensi proses pengerjaan.

2.2.6 *Laravel*

Laravel merupakan *framework* pengembangan aplikasi web berbasis *PHP* yang bersifat *open-source* dan dapat digunakan secara gratis. Framework ini dikembangkan oleh Taylor Otwell dan dirancang untuk mendukung pengembangan aplikasi dengan menerapkan pola *Model-View-Controller (MVC)*. Struktur pola *MVC* pada *Laravel* memiliki perbedaan dibandingkan dengan penerapan umum pola tersebut. Salah satu perbedaannya terletak pada mekanisme *routing*, di mana *routing* berperan sebagai perantara antara permintaan (*request*) dari pengguna dan *controller*, sehingga *controller* tidak secara langsung menerima permintaan tersebut (Yudanto dkk. 2017).



Gambar 2.5 Laravel

2.2.7 *Database*

Pengelolaan data secara efisien dan sistematis menjadi kebutuhan yang esensial pada era digital, terutama seiring meningkatnya kompleksitas kebutuhan akan informasi yang akurat dan terintegrasi. Salah satu konsep penting dalam pengelolaan data adalah *basis data (database)*, yang merupakan kumpulan

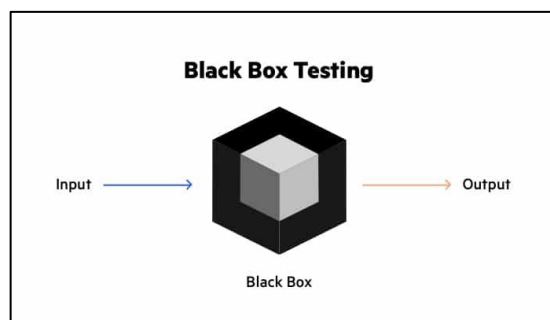
informasi yang tersimpan secara terstruktur di dalam sistem komputer. *Basis data* memungkinkan informasi disimpan secara sistematis dan dapat diakses menggunakan program komputer tertentu untuk memperoleh data yang dibutuhkan. Perangkat lunak yang berfungsi untuk mengelola dan mengeksekusi *kueri* terhadap *basis data* dikenal sebagai sistem manajemen *basis data* atau *database management system (DBMS)*. Pemahaman mengenai sistem *basis data* merupakan bagian dari kajian dalam bidang ilmu informasi (andaru, 2018).



Gambar 2.6 DATABASE

2.2.8 Blackbox Testing

Black box testing merupakan metode pengujian sistem yang berfokus pada penilaian terhadap spesifikasi fungsional tanpa memperhatikan struktur internal dari sistem yang diuji. Tujuan dari metode ini adalah untuk mendeteksi berbagai jenis kesalahan, seperti fungsi yang tidak berjalan semestinya, antarmuka yang bermasalah, struktur data yang tidak sesuai, kendala dalam akses *basis data*, performa sistem yang kurang optimal, serta gangguan dalam proses inisialisasi maupun terminasi. Beberapa teknik yang umum diterapkan dalam *black box testing* antara lain *equivalence partitioning*, *boundary value analysis*, dan *limit testing*, yang semuanya digunakan untuk menjamin bahwa seluruh fungsi sistem telah diuji secara menyeluruh (Maulana dkk. 2023).



Gambar 2.7 Blackbox Testing

2.2.9 System Usability Scale (SUS)

System Usability Scale (SUS) merupakan metode evaluasi yang umum digunakan untuk mengukur tingkat kemudahan penggunaan suatu antarmuka, sistem, atau aplikasi berdasarkan standar tertentu. Metode ini terdiri atas 10 butir pernyataan yang dinilai menggunakan skala Likert dari 1 sampai 5. Dalam proses penilaian, skor pada pernyataan bernomor ganjil dikurangi dengan angka 1, sedangkan skor pada pernyataan bernomor genap dikurangi dari angka 5. Selanjutnya, jumlah skor dari seluruh pernyataan dikalikan dengan faktor 2,5 guna memperoleh nilai akhir dengan rentang antara 0 hingga 100 (Chandra & Zul, 2024).

Pengguna diminta untuk memberikan penilaian terhadap sistem berdasarkan skala 1 hingga 5 terhadap sepuluh pernyataan yang telah disiapkan. Skala dan pernyataan yang digunakan disajikan pada tabel berikut:

Tabel 2.2 Skala Evaluasi Usability

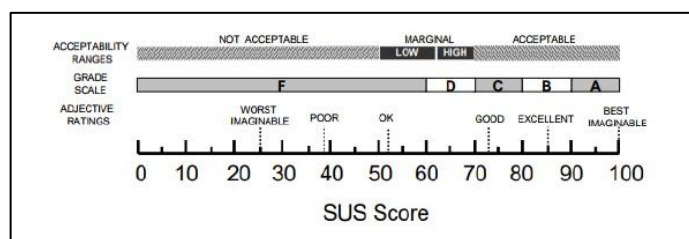
Skala	Penjelasan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Netral
4	Setuju
5	Sangat Setuju

Data yang dikumpulkan melalui SUS akan membantu dalam mengidentifikasi area yang perlu diperbaiki serta aspek yang telah berfungsi dengan baik.

Usability yang diperoleh. Berikut masing-masing kategorinya :

- $\geq 80 \rightarrow$ Website Sangat Baik (*Excellent usability*)
- $70 - 79 \rightarrow$ Website Baik (*Good usability*)
- $50 - 69 \rightarrow$ Website Cukup (*Average usability, perbaikan disarankan*)
- $< 50 \rightarrow$ Website Buruk (*Poor usability, perlu perbaikan signifikan*)

Hasil dari pengujian ini akan digunakan sebagai dasar dalam mengevaluasi kekurangan yang ada, mengidentifikasi berbagai masalah usability yang dihadapi oleh pengguna, serta menentukan langkah-langkah perbaikan yang diperlukan untuk meningkatkan kualitas pengalaman pengguna terhadap website di masa mendatang.



Gambar 2.8 System Usability Scale

2.2.10 Purposive Sampling

Purposive sampling merupakan teknik non-probability sampling yang menentukan sampel berdasarkan kriteria tertentu sesuai tujuan penelitian, bukan berdasarkan peluang acak. Salah satu variannya, maximum variation purposive sampling, mengarahkan pemilihan sampel untuk mencakup variasi karakteristik seluas mungkin dari populasi, sehingga data yang diperoleh mencerminkan keberagaman kondisi yang ada. Teknik ini banyak digunakan dalam penelitian evaluasi sistem informasi, termasuk pada evaluasi usability sistem informasi karir dan tracer study yang menentukan responden evaluasinya secara purposive (Melyani dkk., 2023).

Dalam penelitian ini, purposive sampling diterapkan pada seluruh penentuan informan kualitatif, yang berjumlah enam orang — satu informan kunci dari pihak internal Central Printing dan lima pelanggan. Kelima informan pelanggan dipilih berdasarkan variasi jenis pesanan yang mereka gunakan pada sistem, meliputi kebutuhan undangan, tugas kuliah, kemasan UMKM, kebutuhan kantor, dan komunitas, sehingga mencerminkan keberagaman use case sistem. Informan kunci dari pihak internal dipilih berdasarkan dua pertimbangan: menjabat rangkap sebagai admin sekaligus manager, sehingga memiliki wawasan pada aspek teknis maupun proses bisnis secara keseluruhan, dan memiliki masa kerja paling lama, sehingga mampu membandingkan kondisi operasional sebelum dan sesudah penerapan sistem. Pendekatan ini sejalan dengan penggunaan purposive sampling pada penelitian evaluasi usability sistem informasi sejenis (Melyani dkk., 2023).

2.2.11 Total Sampling (Sampling Jenuh)

Total sampling atau sampling jenuh adalah teknik penentuan sampel yang digunakan apabila seluruh anggota populasi dijadikan sampel penelitian, umumnya diterapkan pada populasi berjumlah kecil sehingga seluruh anggotanya dapat dilibatkan tanpa penarikan sampel acak maupun kriteria tambahan.

Dalam penelitian ini, teknik total sampling secara sadar tidak digunakan pada wawancara pihak admin, meskipun Central Printing memiliki lebih dari satu administrator. Pemilihan hanya satu informan admin dilakukan karena tujuan wawancara adalah menggali kedalaman informasi dari sumber yang paling information-rich, bukan memperoleh gambaran dari seluruh populasi admin. Dengan demikian, penelitian ini memilih pendekatan purposive dibandingkan total sampling untuk bagian kualitatifnya (Melyani dkk., 2023).

2.2.12 Pendekatan Mixed-Method dalam Evaluasi Sistem

Pendekatan mixed-method mengombinasikan data kuantitatif dan kualitatif secara bersamaan maupun berurutan dalam satu penelitian untuk memperoleh pemahaman yang lebih komprehensif dan mendalam. Data kuantitatif memberikan gambaran umum yang terukur dan dapat digeneralisasi, sedangkan data kualitatif memperdalam dan mentriangulasi temuan kuantitatif melalui eksplorasi konteks yang tidak terungkap oleh angka semata. Desain sequential/concurrent explanatory yang digunakan dalam pendekatan ini mengumpulkan data kualitatif untuk menjelaskan atau memperkuat hasil yang telah diperoleh dari data kuantitatif (Asrulla Asrulla, t.t.).

Dalam penelitian ini, pengujian kuantitatif menggunakan System Usability Scale dengan 30 responden dilakukan lebih dahulu untuk memperoleh gambaran umum tingkat usability sistem, kemudian diperkuat dan didalami melalui wawancara kualitatif terhadap enam informan untuk menggali kesan awal (first impression) yang tidak dapat terungkap hanya dari skor numerik (Asrulla Asrulla, t.t.).