

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Sebelumnya

Penelitian yang telah dilakukan oleh Budiansyah[18] pada tahun 2019, mengenai pengembangan modul Himpunan Alumni dalam aplikasi Alumni IPB *Connect* dengan menggunakan arsitektur REST API. Memiliki tujuan sebagai upaya untuk meningkatkan peran Himpunan Alumni IPB dalam menjalin hubungan yang berkelanjutan antara alumni dan almamater. Metode pengembangan yang digunakan adalah *Prototyping* dengan menggunakan arsitektur REST API. Bahasa pemrograman utama yang digunakan adalah Node.js dengan *framework* Express.js, serta didukung oleh basis data MongoDB. Pengembangan difokuskan pada aplikasi *server-side* yang menghasilkan *web services* untuk aplikasi *client-side*. Hasil dari penelitian ini adalah implementasi API yang dapat diakses melalui api.stage.ipb.ac.id, sesuai dengan kebutuhan pengembang aplikasi *client-side*. API ini memfasilitasi manipulasi data terkait fakultas, program studi, lowongan kerja, event, dan anggota, dan telah berhasil diintegrasikan dengan aplikasi *front-end* sesuai yang diharapkan.

Penelitian selanjutnya dilakukan oleh Enstayn, dkk[19] pada tahun 2019, mengenai perancangan Sistem Informasi Manajemen Berbasis Web pada Pt Unicorn Intertranz. Penelitian ini memiliki tujuan merancang sistem informasi manajemen untuk PT. Unicorn Intertranz. Metode *Prototyping* digunakan dalam penelitian ini untuk memudahkan klien dalam evaluasi sistem secara bertahap. Tahapan penelitian meliputi pengumpulan data, terjemahan diagram UML, pengkodean, pengujian, dan implementasi. Desain sistem juga memperhatikan komunikasi data untuk dokumen kebutuhan pengguna. *Quick Modelling* digunakan untuk menerjemahkan data ke dalam model sistem sebelum pengkodean dilakukan. Pengujian sistem dilakukan dengan tujuan menemukan

kesalahan sebelum diserahkan kepada klien. Subsistem seperti *Delivery Order* dan Operasional memiliki fungsi dan prototipe spesifik yang diuji menggunakan metode *Black Box Testing*. Hasil penelitian menunjukkan bahwa sistem informasi manajemen yang dirancang berhasil memenuhi kebutuhan perusahaan dengan efektif dan efisien. Studi ini juga merujuk pada penelitian terdahulu tentang prototipe perangkat lunak dan sistem informasi manajemen.

Penelitian oleh Muhajir, dkk[20] pada tahun 2022, berfokus pengembangan sistem informasi presensi di SMAN 13 Pangkep dilakukan dengan metode pengembangan *prototype*. Penelitian bertujuan untuk memudahkan pekerjaan instruktur dan siswa di SMAN 13 Pangkep dengan mengembangkan sistem informasi presensi berbasis *mobile* Android dan web. Proses pengembangan dengan menggunakan metode *Prototype* melibatkan analisis kebutuhan, pemodelan *use case*, desain prototipe, desain arsitektur perangkat lunak, dan implementasi program. Uji coba sistem meliputi validasi, kompatibilitas, dan uji kegunaan dengan skor validasi mencapai 100% dan kegunaan mencapai 87,5 menggunakan metode SUS. Sistem ini berhasil memenuhi semua kebutuhan fungsional dan ramah pengguna bagi guru, siswa, dan bimbingan konseling.

Penelitian yang dilakukan oleh Muhammad, dkk.[21], terfokus pada Pengembangan *Backend* Server Berbasis Arsitektur REST API untuk meningkatkan efisiensi dalam transfer dana antar dompet digital, seiring dengan pergeseran paradigma dari pembayaran tunai menuju sistem non-tunai yang semakin dominan. Pengembangan ini menggunakan *framework* NestJS sebagai infrastruktur utama dan basis data MySQL untuk menyimpan dan mengelola informasi transaksi. Pengujian API dilakukan dengan menggunakan Postman, sementara proses pengembangan mengikuti metodologi *Extreme Programming* (XP) yang mengedepankan iterasi cepat dan responsif terhadap perubahan kebutuhan. Uji coba sistem menggunakan metode *BlackBox Testing* untuk memverifikasi fungsionalitas keseluruhan tanpa memperhatikan struktur

internalnya. Hasil dari penelitian ini adalah implementasi berhasil dari sistem REST API yang mampu mengelola transfer dana antar dompet digital secara efektif. Sistem ini dirancang untuk memenuhi kebutuhan pengembang aplikasi *front-end* dan *mobile*, memastikan integrasi yang lancar dan responsif.

Terakhir adalah penelitian yang telah dilaksanakan oleh Firdaus & Ramadhan pada tahun 2021[22]. Penelitian ini bertujuan untuk mengatasi permasalahan serius kerentanan pangan di Indonesia pada tahun 2018 dengan mengembangkan sistem Patriot Pangan untuk deteksi dini. Tujuan utamanya adalah untuk memanfaatkan partisipasi aktif masyarakat dalam melaporkan kondisi pangan yang rentan untuk meningkatkan respons dan penanganan yang lebih cepat. Metode pengembangan yang digunakan adalah *Scrum* yang terdiri dari lima sprint yang masing-masing berlangsung selama dua minggu. Artefak metodologis termasuk *product backlog*, *sprint backlog*, *use case diagram*, dan *class diagram*. Penelitian tersebut berhasil mengembangkan backend berbasis REST API menggunakan Node.js dengan *framework* Express.js dan MongoDB sebagai *databasenya*. *Backend* ini memungkinkan aplikasi E-Participation melaporkan kerentanan pangan dan mengumpulkan data terkait bulanan, sementara *E-Initiative* mengelola kampanye dan donasi. Hasilnya mencakup penerapan 23 API yang telah diuji yang bertujuan untuk memfasilitasi deteksi dini kerentanan pangan dan secara efektif mendukung *crowdfunding* untuk mitigasinya.

Tabel 2. 1 Penelitian Sebelumnya

No	Peneliti	Judul Penelitian	Tahun Terbit	Tujuan Penelitian	Metode Penelitian	Hasil yang diperoleh
1	Budiansyah.[18]	Pengembangan Modul <i>Back End</i> Himpunan Alumni Pada Aplikasi Ipb Connect	2019	Tujuan dari penelitian ini adalah untuk meningkatkan peran Asosiasi Alumni IPB melalui pengembangan modul <i>Back End</i> pada aplikasi IPB Connect.	Prototyping	Hasil pengembangan modul <i>Back End</i> pada aplikasi IPB <i>Connect</i> adalah API yang dapat diakses di api.stage.ipb.ac.id . API ini dibuat untuk meningkatkan peran Asosiasi Alumni IPB dengan menyediakan <i>platform</i> untuk keterlibatan dan interaksi alumni dengan aplikasi tersebut.
2	Kustanto & Prillysca Chernovita.[19].	Perancangan Sistem Informasi Manajemen Berbasis Web	2019	Tujuan dari penelitian yang dibahas ini adalah untuk merancang sistem informasi manajemen	Prototyping	Hasil penelitian tentang perancangan sistem informasi manajemen berbasis web untuk PT Unicorn Intertranz mencakup penciptaan sebuah sistem yang

No	Peneliti	Judul Penelitian	Tahun Terbit	Tujuan Penelitian	Metode Penelitian	Hasil yang diperoleh
		Studi Kasus : Pt Unicorn Intertranz		berbasis web untuk PT Unicorn Intertranz, sebuah perusahaan yang bergerak di bidang layanan pengiriman barang untuk mendukung kegiatan bisnis dan manajemen data perusahaan.		dapat digunakan untuk mengelola semua proses bisnis perusahaan, mulai dari penerimaan Surat Perintah Pengiriman (DO) hingga faktur, manajemen keuangan, dan pemantauan operasional.
3	Muhajir dkk.[20]	Pengembangan Sistem Informasi Presensi Sekolah Menggunakan Metode	2022	Tujuan dari penelitian ini adalah untuk memudahkan pekerjaan instruktur dan siswa di SMAN 13	Prototyping	Implementasi sistem informasi presensi yang memenuhi kebutuhan fungsional dan ramah pengguna bagi guru, siswa, dan bimbingan konseling di SMAN 13

No	Peneliti	Judul Penelitian	Tahun Terbit	Tujuan Penelitian	Metode Penelitian	Hasil yang diperoleh
		<i>Prototype</i> (Studi Kasus: Sman 13 Pangkep)		Pangkep dengan mengembangkan sistem informasi presensi berbasis <i>mobile</i> Android dan web.		Pangkep. Hasil menunjukkan validasi 100%, kompatibilitas di berbagai perangkat dan browser, dan skor kegunaan sebesar 87,5, menunjukkan efektivitas dan kemudahan penggunaan sistem.
4	Muhammad, dkk[21].	Pengembangan <i>Backend</i> Server Berbasis Arsitektur REST API pada Sistem Transfer Dompot Digital	2023	Tujuan penelitian ini adalah untuk mengembangkan sistem transfer dana antar dompet digital yang dapat diakses oleh pengembang frontend dan mobile, sehingga	Extreme Programming (XP)	Hasil dari penelitian ini adalah implementasi sukses dari sistem <i>server-side</i> REST API yang memfasilitasi transfer dana antar dompet digital. Sistem ini dapat diintegrasikan dengan mudah oleh pengembang <i>front-end</i> dan <i>mobile</i> , serta telah diuji secara berkala untuk memastikan kinerja yang

No	Peneliti	Judul Penelitian	Tahun Terbit	Tujuan Penelitian	Metode Penelitian	Hasil yang diperoleh
				mempermudah integrasi dan penggunaan oleh pengguna akhir.		optimal dan manajemen kesalahan yang efisien.
5	Firdaus & Ramadhan.[22].	Pengembangan <i>Back End</i> Berbasis REST API pada Sistem E-Partisipasi dan E-Inisiatif Patriot Pangan	2021	Mengembangkan backend berbasis REST API yang dapat digunakan oleh aplikasi E-Partisipasi, E-Inisiatif, dan web Patriot Pangan.	Scrum	Hasil penelitian ini berhasil mengembangkan backend berbasis REST API menggunakan bahasa pemrograman JavaScript dan framework Express.js, yang digunakan oleh aplikasi E-Partisipasi, E-Inisiatif, dan web Patriot Pangan.

2.2 Dasar Teori

2.2.1 Sistem Informasi Manajemen(SIM)

Sistem Informasi Manajemen adalah sebuah sistem yang dirancang untuk mengumpulkan, memproses, menyimpan, dan mengelola informasi yang relevan dengan kegiatan manajemen dalam suatu organisasi. Sistem Informasi Manajemen memiliki peran yang krusial dalam membantu manajer untuk mengambil keputusan yang tepat[23]. Dengan adanya SIM, manajer dapat mengakses informasi yang akurat dan relevan untuk mendukung perencanaan, pengorganisasian, pengelolaan, dan pengendalian dalam organisasi. SIM juga merangkum data dari berbagai sumber internal dan eksternal, sehingga membantu dalam pemantauan kinerja, identifikasi masalah, dan peluang perbaikan. Sistem Informasi Manajemen juga membantu dalam mengotomatisasi proses bisnis, meningkatkan efisiensi operasional.

2.2.2 Application Programming Interface (API)

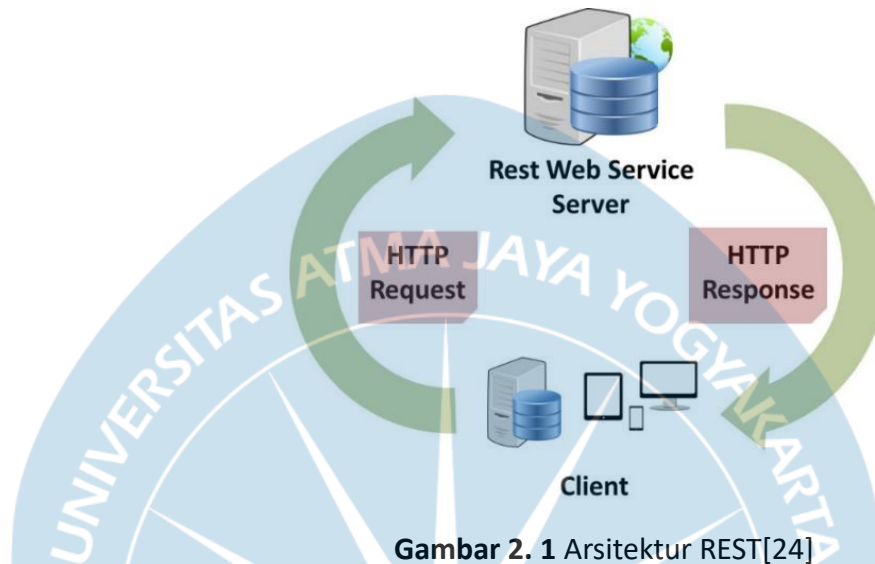
Application Programming Interface (API) atau Antarmuka Pemrograman Aplikasi adalah antarmuka yang dibuat oleh pengembang sistem sehingga sebagian atau seluruh fungsi sistem dapat diakses secara terprogram. API juga umumnya dianggap sebagai serangkaian teknik yang jelas untuk menciptakan komunikasi antara komponen perangkat lunak yang berbeda. Fungsi API adalah untuk memfasilitasi penggunaan teknologi tertentu ketika membangun perangkat lunak atau aplikasi untuk pengembang[24]. Ini memainkan peran yang krusial dalam pengembangan perangkat lunak modern. Mereka bertindak sebagai penghubung antara berbagai komponen perangkat lunak, memfasilitasi komunikasi dan interaksi di antara mereka. Hal ini memungkinkan integrasi sistem yang lebih

kompleks dan meningkatkan efisiensi pengembangan perangkat lunak secara keseluruhan[25].

2.2.3 Representational State Transfer (REST)

Representational State Transfer (REST) adalah sebuah arsitektur perangkat lunak yang menjadi landasan bagi pembangunan sistem berbasis web yang fleksibel, terdistribusi, dan mudah dikelola[26]. REST didasarkan pada sejumlah prinsip, termasuk penggunaan protokol HTTP untuk komunikasi antara klien dan server, pengidentifikasian sumber daya menggunakan URI (*Uniform Resource Identifier*), serta penggunaan metode HTTP seperti GET, POST, PUT, dan DELETE untuk berinteraksi dengan sumber daya tersebut. Prinsip desain REST juga mencakup konsep HATEOAS (*Hypermedia as the Engine of Application State*), yang memungkinkan klien untuk melakukan navigasi aplikasi secara dinamis melalui tautan-tautan yang diberikan oleh server[27][28]. Terdapat sejumlah keuntungan dalam menggunakan REST, termasuk fleksibilitas dalam merancang antarmuka aplikasi, kemampuan untuk meningkatkan skalabilitas dengan memisahkan klien dan server, serta kemampuan untuk menggunakan berbagai bahasa pemrograman dan teknologi. Namun, implementasi REST juga memiliki tantangan tersendiri, seperti kompleksitas dalam merancang *endpoint* API yang intuitif dan efisien, serta masalah keamanan terkait manajemen otentikasi dan otorisasi[29].

REST



Gambar 2. 1 Arsitektur REST[24]

2.2.4 Back-End

Back-End memainkan peran penting dalam pengembangan perangkat lunak dengan menangani logika, fungsionalitas, pemrosesan data, manajemen sumber daya, dan integrasi antarmuka pengguna[30]. Sebagai tulang punggung sistem, back-end memastikan kohesi dan keandalan operasi aplikasi tanpa interaksi langsung dengan pengguna. Dikenal juga sebagai *server-side*, back-end adalah tempat di mana data diproses, ditambahkan, diubah, atau dihapus. Back-end, yang fokus pada keamanan, desain sistem, dan manajemen data, sangat dibutuhkan dalam pengembangan sistem atau aplikasi dinamis dengan data yang selalu berubah-ubah[31]. Pengembangan back-end merujuk pada bagian dari pengembangan web yang berfokus pada pembuatan dan pengelolaan sisi server dari aplikasi. Ini melibatkan proses-proses yang kompleks dalam merancang, membangun, dan mengelola komponen-komponen server dari aplikasi web[32]. Bagian ini didedikasikan untuk menangani penyimpanan

data, pemrosesan, dan manajemen, memastikan komunikasi yang lancar antara antarmuka pengguna dan basis data atau server yang mendasarinya. Ini melibatkan implementasi algoritma-algoritma yang kuat, optimalisasi kinerja, dan penguatan langkah-langkah keamanan untuk menjaga integritas dan efisiensi dari seluruh sistem[33]. Tugas tersebut memerlukan pemahaman yang mendalam dalam mengelola logika dan proses data secara efisien, selain itu juga melibatkan kemampuan untuk berkomunikasi secara efektif antara antarmuka pengguna (*front-end*) dengan basis data (*database*) di belakang layar[34].

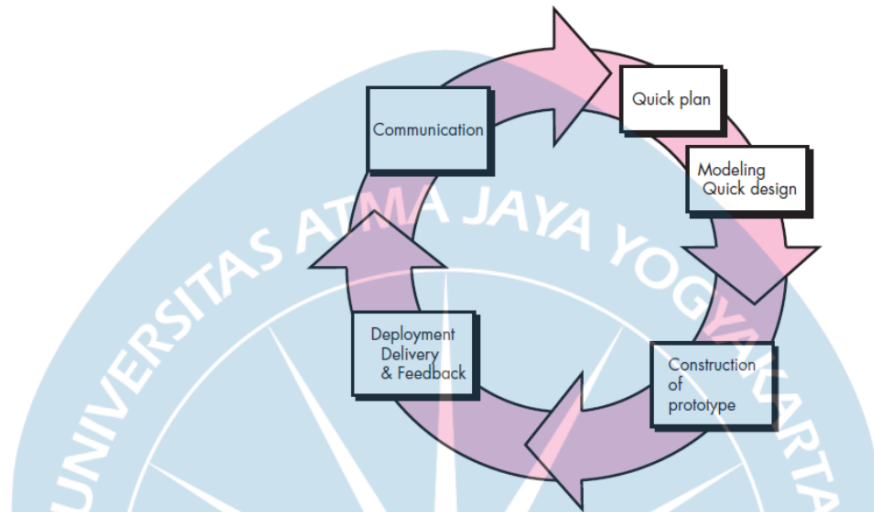
2.2.5 Entity Relationship Diagram(ERD)

Diagram Hubungan Entitas adalah representasi visual dari hubungan antar entitas dalam *database*. Diagram ini terdiri dari entitas yang direpresentasikan sebagai persegi panjang, dan hubungan antar entitas yang direpresentasikan sebagai garis yang menghubungkan persegi panjang tersebut[35]. Tujuan dari *Entity-Relationship Diagram* adalah untuk memberikan gambaran yang jelas dan ringkas tentang struktur dan hubungan dalam sistem *database*. Saat membuat ERD, penting untuk mengidentifikasi entitas dan atributnya, serta hubungan antar entitas[36].

2.2.6 Metode Prototyping

Prototyping merupakan metode yang umum digunakan dalam pengembangan sistem karena dapat mewakili kebutuhan pengguna ketika mereka tidak dapat mendefinisikan kebutuhannya. Metode ini menghasilkan prototipe sebagai versi awal sistem, yang digunakan untuk menyajikan konsep, bereksperimen dengan desain, dan mengidentifikasi kemungkinan masalah serta solusinya[37]. Oleh karena itu, pembuatan prototipe melibatkan interaksi pengguna untuk mengevaluasi apakah prototipe perlu disesuaikan berdasarkan permintaan dan kebutuhan pengguna[38]. Selain itu, prototipe juga merupakan metode yang efektif untuk menerima umpan

balik tentang sistem yang diusulkan serta menjelaskan bagaimana sistem tersebut tersedia untuk memenuhi kebutuhan informasi pengguna[39].



Gambar 2. 2 Metode Pengembangan Prototyping[32]

Tahapan metode *prototyping* ditunjukkan pada **Gambar 1.3**. Proses *prototyping* mencakup beberapa tahap utama yaitu:

1. Komunikasi (*Communication*)

Pada tahap ini, penulis berinteraksi secara langsung dengan klien untuk memahami kebutuhan awal dan menetapkan tujuan umum proyek. Melalui wawancara dan diskusi, tim mengumpulkan wawasan mendalam tentang kebutuhan dan harapan klien, dengan tujuan mengatasi isu-isu inti dan mendefinisikan solusi perangkat lunak yang diinginkan. Hasil dari tahap ini adalah dokumen perencanaan awal yang menguraikan ruang lingkup proyek, tujuan, dan pemahaman awal tentang persyaratan fungsional dan non-fungsional.

2. Modeling Cepat (*Quick Plan*)

Pengembang membuat model cepat dari perangkat lunak yang dapat digunakan oleh klien. Model cepat ini berfungsi sebagai representasi awal dari solusi perangkat lunak, membantu visualisasi konsep inti dan fungsionalitas yang akan disediakan.

3. Pemodelan Perancangan Cepat (*Quick Design Modeling*)

Tahap ini melibatkan pengembangan desain yang lebih rinci berdasarkan model cepat yang telah dibuat sebelumnya. Penulis merancang struktur perangkat lunak dengan lebih detail, termasuk arsitektur sistem, antarmuka pengguna, dan logika bisnis yang akan diimplementasikan.

4. Pengembangan Prototipe (*Construction of Prototype*)

Pengembang membangun prototipe yang dapat digunakan oleh klien untuk memberikan komentar dan umpan balik. Prototipe ini merupakan implementasi awal dari perangkat lunak yang mencakup fungsionalitas dasar yang direncanakan. Klien dapat berinteraksi dengan prototipe untuk memahami lebih baik cara kerja solusi yang diusulkan.

5. Penerapan, Pengiriman, dan Umpan Balik (*Deployment, Delivery, dan Feedback*)

Proses iterasi dimulai dengan umpan balik dari klien tentang prototipe. Penulis mengevaluasi umpan balik ini dan melakukan penyesuaian atau perbaikan yang diperlukan untuk memenuhi kebutuhan dan harapan klien. Setelah penyesuaian dilakukan, prototipe yang diperbarui atau versi berikutnya dari perangkat lunak dan disampaikan kepada klien untuk evaluasi lebih lanjut. Proses ini berulang secara iteratif sampai klien puas dengan solusi perangkat lunak yang dikembangkan dan siap digunakan secara penuh.

Metode *prototype* memiliki beberapa manfaat, sebagai berikut

- Mengurangi Risiko

Metode ini memungkinkan pengembang untuk memantau dan mengurangi risiko yang diambil dalam proses pengembangan perangkat lunak.

- Mempercepat Proses

Prototype memungkinkan pengembang untuk membuat perangkat lunak lebih cepat dan efektif.

- Pengembangan Berkualitas

Metode ini memungkinkan pengembang untuk membuat perangkat lunak yang lebih berkualitas, karena klien dapat memberikan feedback secara langsung[37].

2.2.7 Unified Modeling Language(UML)

Dalam desain sistem, diagram UML digunakan untuk mewakili secara visual struktur dan perilaku suatu sistem. Diagram UML memberikan notasi standar untuk mewakili berbagai aspek sistem, seperti kelas, objek, hubungan, dan interaksi[40]. UML memiliki berbagai jenis diagram yang masing-masing memiliki fokus dan tujuan yang dapat disesuaikan dengan kondisi pengembang. Diagram *Use Case* menggambarkan interaksi antara pengguna (aktor) dan sistem, serta fungsi-fungsi utama yang disediakan oleh sistem. Diagram ini berguna untuk mengidentifikasi persyaratan fungsional dan memperjelas batasan sistem. *Sequence Diagram* atau Diagram Urutan memodelkan interaksi antar objek dalam urutan waktu tertentu, menunjukkan bagaimana objek berkomunikasi satu sama lain melalui pesan. Diagram ini membantu dalam memahami alur kontrol dan urutan eksekusi dalam suatu proses atau skenario[40][41]. Diagram ini membantu dalam memahami dan mengkomunikasikan desain sistem kepada pemangku kepentingan. Selain itu, diagram UML memungkinkan dokumentasi

keputusan desain sistem utama dan berfungsi sebagai cetak biru untuk tahap implementasi[42].

2.2.8 Pengujian Fungsional

Pengujian fungsional berfokus pada jaminan kualitas perangkat lunak dengan membuat kasus uji sesuai dengan spesifikasi yang telah ditentukan. Dalam metode ini, fungsi-fungsi perangkat lunak diuji melalui pemberian data masukan dan analisis hasil keluarannya. Metode ini lebih menitikberatkan pada hasil akhir dibandingkan dengan struktur internal program yang jarang diperhatikan[43]. Metode ini digunakan untuk mengidentifikasi kesalahan atau *bug* pada sistem dengan menguji fungsionalitasnya tanpa mengetahui kode internalnya. Hasil dari pengujian fungsional kemudian dibandingkan dengan keluaran yang diharapkan untuk menentukan apakah sistem berfungsi sebagaimana mestinya[44].