SEKOLAH PASCASARJANA

BAB II
TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Pemerintahan memiliki dua cara untuk memanfaatkan Blokchain: (1) dari perspektif pemerintahan oleh Blokchain, di mana organisasi dapat mengadopsi teknologi dan menggunakannya untuk berbagai operasi, seperti mengatur transaksi dan menyediakan layanan; dan (2) dari perspektif tata kelola atau tata kelola Blokchain, di mana organisasi atau pemerintahan yang menentukan bagaimana Blokchain seharusnya digunakan dan bagaimana Blokchain seharusnya digunakan (Adeyemi dkk., 2020).

Selain dua contoh sebelumnya, teknologi blokchain juga dapat digunakan dalam manajemen proyek konstruksi. Setiap proyek konstruksi selalu melibatkan lebih dari satu bisnis yang bekerja sama, biasanya bekerja sama secara hirarkis atau satu sama lain, dan kegiatan manajemen proyek ini sangat kompleks (Bhushan dkk., 2021).

Smart contract, yang pada awalnya bertujuan untuk membuat transaksi lebih mudah, fleksibel, dan efisien, berpotensi menimbulkan masalah hukum, khususnya berkaitan dengan kepentingan pelanggan sebagai pengguna layanan pengiriman online. Oleh karena itu, penulis ingin memeriksa kesepakatan para pihak dalam smart contract dengan melihat konstruksi hukum kontrak di Indonesia (Machado & Westphall, 2021).

Smart contract adalah pengembangan berikutnya dari blokchain setelah cryptocurrency. Nick Szabo menggambarkan smart contract sebagai "an electronic transaction protocol that executes the term of a contract." Secara umum, tujuan adalah untuk memenuhi persyaratan perjanjian yang umum, seperti pembayaran, liens, kerahasiaan, dan bahkan pelaksanaan, sehingga mengurangi situasi yang tidak menyenangkan dan tidak sengaja dan mengurangi kebutuhan akan perantara yang dapat diandalkan. Ekonomi terkait mencakup pengurangan kerugian fraud, biaya arbitrasi dan penegakan hukum, dan biaya transaksi lainnya. (Szabo, 1994)

Smart contract "is a computer program which verifies and executes its terms upon the occurrence of predetermined events," kata Mark Giancaspro. "Smart contract" didefinisikan sebagai "A smart contract is a secure and unstoppable computer program representing an agreement that is automatically executable and enforceable" (Imran, 2018). Sebaliknya, smart contract tidak dapat diubah dan beroperasi sesuai dengan instruksi yang diprogramnya (Mark Giancaspro, 2017).

Menurut ketiga definisi smart contract, ciri-ciri smart contract dapat dieksekusi secara otomatis. Sifat self-executing menjamin keamanan data transaksi yang dilakukan dalam smart contract, membedakannya dari kontrak elektronik lainnya yang tersedia di internet. Selain itu, smart contract yang beroperasi pada blokchain tidak memungkinkan pihak ketiga atau intermediaries untuk memvalidasi transaksi. Sebaliknya, kode pemrograman smart contract secara otomatis melakukan kesepakatan pengguna jaringan setelah transaksi diverifikasi melalui penyelesaian protokol kriptografi (Punuh dkk., 2023).

2.2 Keaslian Penelitian

Keaslian pada penelitian ini berfokus pada pembuatan *smart contract* yang dengan jaringan *desentralisasi finance* (DeFi) untuk pembuatan token dalam jaringan *binance smart chain* serta menghubungkan dengan menggunakan metode NDN untuk pengembangan jaringan *Blokchain* dan untuk pengembangan sistem LMS menggunakan V model. Diperkuat penelitian sebelumnya yang linier membahas tentang Blokchain pada Tabel 2.1

Tabel 2.1 Keaslian Penelitian

No	Nama (tahun)	Judul	Hasil Penelitian
1	Yi-Ming Guo dkk. (2021)	*A Bibliometric Analysis and Visualization of Blokchain*	Hasilnya menunjukkan bahwa masa depan penelitian harus berkonsentrasi pada manajemen, teknologi *Blokchain*, energi, mesin belajar, dan rumah pintar. Oleh

Tabel 2.2 Keaslian Penelitian (lanjutan)

No	Nama (tahun)	Judul	Hasil Penelitian
			karena itu, makalah ini membantu para sarjana dan praktisi dalam mendapatkan pemahaman yang komprehensif tentang status *quo dan tren Blokchain riset.*
2	Na Shi dkk. (2021)	*A Blokchain-empowered AAA scheme in the large-scale HetNet*	Hasilnya menunjukkan bahwa skema tersebut tidak hanya aman tetapi juga desentralisasi, tanpa gangguan dan kepercayaan.
3	Dinh C. Nguyen ("2020")	*Blokchain for 5G and beyond networks: A state of the art survey*	Hasil penelitian dari upaya baru-baru ini mendemonstrasikan potensi besar *Blokchain* dalam berbagai praktik Area 5G, mulai dari peningkatan teknologi 5G (yaitu kombinasi *cloud Blokchain*, adopsi *Blokchain-SDN)* hingga layanan 5G (seperti sebagai pembongkaran data berbasis Blokchain, spektrum yang diberdayakan oleh Blokchain berbagi) dan 5G IoT (seperti kota pintar berbasis *Blokchain).*
4	Adetomike Adeyemia dkk. (2020)	*Blokchain technology applications in power distribution systems*	Keuntungan utama dari teknologi *Blokchain* termasuk kemampuannya untuk mengurangi biaya transaksi, meningkatkan ketahanan sistem, dan saat teknologi *Blokchain* matang, mereka terikat untuk mendapatkan peningkatan skalabilitas dan dikenakan biaya operasional yang lebih rendah. Pertumbuhan ini akan membuat teknologi *Blokchain* menjadi lebih vital bagi

Tabel 2.3 Keaslian Penelitian (lanjutan)

			pengembangan mode operasi terdesentralisasi yang optimal.
5	Khizra Ashaf dkk. (2020)	*Blokchain technology in Named Data Networks: A detailed survey*	*Blokchain* menyediakan solusi tersentralisasi dan terdistribusi untuk dipelihara catatan yang konsisten dan andal dalam jaringan yang tidak dapat diandalkan tanpa memerlukan otoritas terpusat. Namun, teknologi *Blokchain* over IP masih memiliki beberapa masalah serius seperti kurangnya efisiensi untuk akses hierarkis. Penggunaan teknologi *Blokchain* melalui NDN telah memecahkan masalah ini dengan menyediakan sistem terdesentralisasi dan menyederhanakan arsitektur.
6	Danson Kimani, dkk. (2020)	*Blokchain, business and the fourth industrial revolution: Whence, whither, wherefore and how?*	Hasil menunjukkan beberapa manfaat signifikan dari teknologi tentang bagaimana organisasi dapat memanfaatkannya untuk mengurangi operasional, transaksi dan biaya agensi. Kemampuan teknis unik dari *Blokchain* dapat mengubah budaya organisasi dan nasional dan membantu meningkatkan transparansi dan kepercayaan di dalam dunia maya. Dikombinasikan dengan teknologi penginderaan lapangan, AI dan *internet of things, Blokchain* memiliki potensi untuk menawarkan manfaat yang signifikan untuk perdagangan internasional, pasar modal, perpajakan,

Tabel 2.4 Keaslian Penelitian (lanjutan)

			audit, perbankan, dan perusahaan pemerintahan, antara lain.
7	Sebastien Ragot, dkk. (2020)	*IP lifecycle management using Blokchain and machine learning: Application to 3D printing datafiles*	Analisis dan statistik adalah tambahan digunakan untuk mencapai perbandingan yang berarti dan menarik kesimpulan dalam hal pelanggaran hak kekayaan intelektual atau keabsahan. Aplikasi dibuat untuk file data pencetakan 3D (batu bata seperti Lego).
8	Dana Yang, dkk. (2021)	*Selective Blokchain system for secure and efficient D2D communication*	Keandalan konsensus dan kinerja sistem ditingkatkan melalui perhitungan probabilitas partisipasi nyata dari para peserta oleh elemen komunikasi dan dengan menentukan apakah setiap *node* adalah partisipan yang egois. Akibatnya, pekerjaan kami mengusulkan algoritma konsensus yang cocok untuk komunikasi D2D dan sistem SBC *(Selective Blokchain)* mempertimbangkan karakteristik lingkungan.
9	Baozhuang Niu, dkk. (2021)	*Should multinational firms implement Blokchain to provide quality verification?*	Kami menemukan bahwa, rantai blok adalah pedang bermata dua untuk MNF. Ini meningkatkan keuntungan grosir MNF, tetapi mengurangi keuntungan ritel MNF dan manfaat perencanaan pajak. Akibatnya, MNF tidak akan berpartisipasi dalam Blokchain jika perbedaan pajak besar dan persaingan hilir sangat ketat.

Tabel 2.5 Keaslian Penelitian (lanjutan)

10	Lennart Ante (2021)	*'Smart Contracts on the Blokchain' – 'A Bibliometric Analysis and Review'*	Menggunakan smart contract, logika bisnis dapat diotomatisasi dan aset seperti uang menjadi dapat diprogram, yang dibuka sebelumnya potensi aplikasi yang tidak dapat diakses. Hingga saat ini, smart contract mengontrol miliaran nilai.

2.3. Landasan Teori

Untuk memulai, kami akan membahas teori dasar yang mendukung penelitian ini, dan kemudian kami akan melihat literatur terbaru tentang metode yang digunakan dalam penelitian ini, serta blockchain.

2.3.1 Blokchain

Pada dasarnya, blokchain adalah basis data transaksi yang terdistribusi pada berbagai node yang terhubung dalam suatu jaringan peer-to-peer. Blokchain adalah salah satu jenis teknologi pengelolaan basis data (DLT) yang beroperasi secara terdesentralisasi dan menggunakan protokol konsensus untuk mencapai kesepakatan tentang prosedur manajemen basis data. Meskipun demikian, ada beberapa hal yang membedakan blokchain dari DLT lainnya (Hamma-adama dkk., 2020).

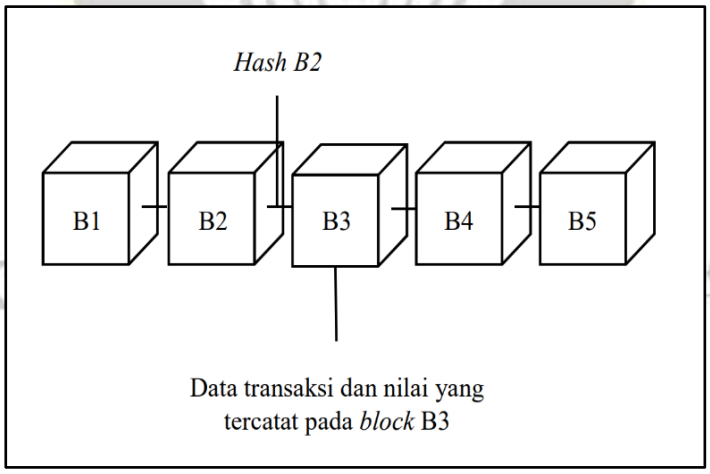
Pencatatan data dalam teknologi blokchain dilakukan dalam dua tahap:

1. Untuk melakukan transaksi data, suatu node akan menggunakan tanda pengenal digital dan kemudian mengumumkannya ke jaringan. Tanda pengenal digital adalah tanda pengenal yang digunakan oleh node tertentu dalam jaringan blokchain.
2. Pengumuman transaksi akan diterima oleh node lain di jaringan dan kemudian dimasukkan ke dalam blok baru.
3. Pembentukan blok akan dilakukan oleh node penerima menggunakan protokol konsensus yang ditetapkan, seperti Proof of Work, dan dikenal sebagai mining.

4. Setelah node penerima berhasil membentuk blok baru menggunakan protokol yang ditetapkan, blok baru ini akan diumumkan ke jaringan sehingga dapat dimasukkan ke rantai blok yang ada.

Pada sistem blok chain Bitcoin, suatu node melakukan transaksi mata uang digital (cryptocurrency) dengan tanda digitalnya. Selanjutnya, node tersebut akan mengumumkan transaksi tersebut ke jaringan (Imran, 2018). Setelah node lain menerima pengumuman transaksi yang terjadi, mekanisme protokol Proof of Work akan menggunakannya untuk membentuk blok. Mekanisme ini akan membentuk blok baru dengan menggunakan fungsi hash kriptografi seperti SHA-256 untuk menghubungkannya ke blok terakhir pada rantai blok yang ada. Menghitung nilai hash blok adalah cara pembentukannya. Seringkali, nilai hash ini disebut sebagai blok header atau blok hash.

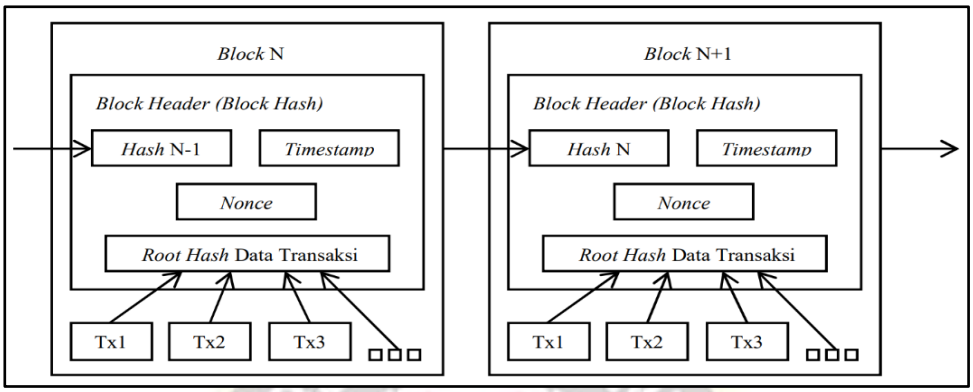
Nilai blok hash dibuat dengan fungsi hash dari nilai data transaksi yang tergabung dalam blok, serta nilai khusus seperti timestamp, nonce, dan blok hash dari blok terakhir pada rantai blok sebelumnya. Dengan menggunakan nilai blok hash terakhir sebagai nilai masukan atau input, hubungan antara satu blok dan blok lainnya akan terbentuk. Gambar 2.1 menunjukkan contoh blokchain.



Gambar 2.1 Ilustrasi Blokchain

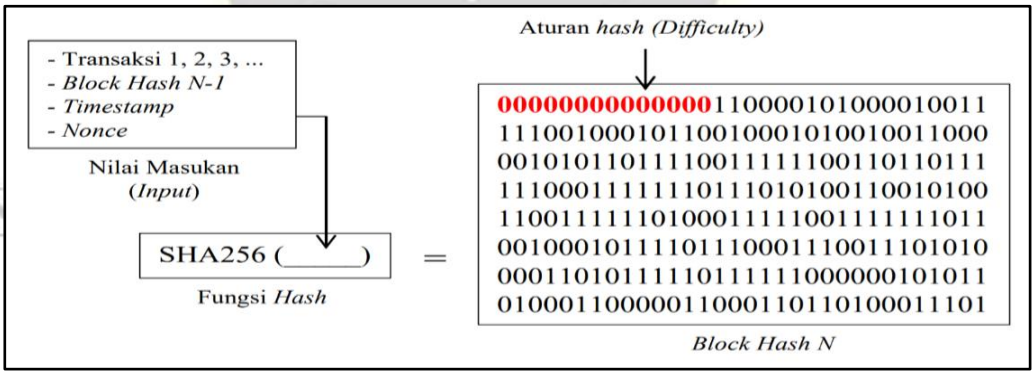
Protokol konsensus Proof of Work digunakan oleh jaringan blokchain untuk mencapai kesepakatan tentang cara membuat blok data yang sah. Jumlah komputasi

yang digunakan dalam proses pembentukan blok menentukan kesepakatan. Bukti pekerjaan ini adalah bukti penggunaan komputasi yang besar dalam proses pembentukan blok. seperti yang ditunjukkan pada gambar 2.2 berikut.



Gambar 2.2 Alur *Blok* pada Jaringan *Blokchain*

Nilai blok hash blok baru dalam mekanisme protokol Proof of Work harus memenuhi aturan kesulitan. Bit awal harus memiliki nilai kesulitan nol. Dalam komputasi blok hash, node membutuhkan nilai nonce, yang merupakan nilai masukan yang dinamis seperti nonce karena nilai masukan lain seperti data utama, timestamp, dan blok hash dari blok sebelumnya memiliki nilai yang tetap. Oleh karena itu, nilai masukan nonce sangat penting untuk perhitungan nilai blok hash yang sesuai berdasarkan kesulitan. Gambar 2.3 berikut menunjukkan ilustrasi komputasi blok hash.



Gambar. 2.3. Komputasi *Hash* pada *Blokchain*

Fungsi hash kriptografi identik dengan SHA-256 akan menghasilkan nilai hash yang sangat tidak terduga dan bersifat satu arah. Dengan kata lain, hash yang terbentuk tidak dapat dijelaskan untuk mendapatkan nilai input awal untuk

fungsinya. Hanya dengan menjalankan fungsi hash berulang kali menggunakan nonce yang berbeda, blok hash yang memenuhi kriteria kesulitan dapat dibuat. Node akan bersaing untuk membentuk blok dengan mencari nonce yang sesuai. Jaringan akan menganggap blok yang terbentuk dengan nilai nonce yang tepat karena proses ini membutuhkan banyak hitungan.

Secara umum, ketika suatu node berhasil membentuk blok, node biasanya akan diberikan cryptocurrency sebagai hadiah sesuai dengan protokol yang ditetapkan oleh blokchain. Namun, node dapat memilih untuk melakukannya atau tidak karena proses pembentukan blok membutuhkan banyak komputasi. Blok akan diumumkan kepada semua node dalam jaringan setelah miner menemukan nonce yang tepat.

Untuk melakukan verifikasi pembentukan blok dan dimasukkan ke dalam rantai blok yang sudah ada, setiap node akan menjalankan fungsi hash dengan nilai yang diberikan oleh miner. Menurut Daniel (2018), teknologi blockchain yang terus berkembang telah dikenal secara umum sebagai rangka kerja untuk membangun sistem terdesentralisasi. Setiap proyek yang mengembangkan sistem terdesentralisasi akan menggunakan blokchain untuk melakukan berbagai perubahan yang memenuhi persyaratan saat ini. Berbagai elemen termasuk perubahan ini, seperti jenis protokol konsensus yang digunakan, gagasan teknologi baru yang diterapkan, teknik pencatatan data, dan sebagainya.

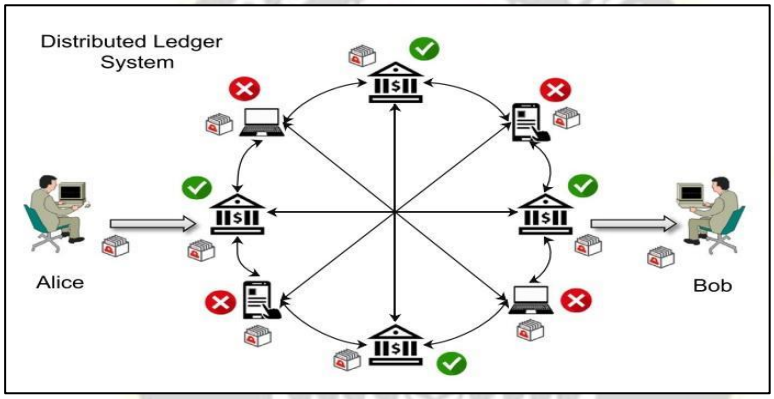
Menurut Sarmah (2018), banyak manfaat dapat dicapai jika struktur blokchain diterapkan pada suatu sistem. *Benefit* yang didapatkan antara lain sebagai berikut:

1. Sistem akan terdesentralisasi, yang berarti bahwa pengelolaan sistem dapat dilakukan tanpa adanya otoritas yang terpusat.
2. Pengguna akan memiliki wewenang untuk berpartisipasi dalam pengelolaan data saat ini.
3. Karena data tersebar di setiap node yang tergabung dalam jaringan sistem, data dapat diakses secara konsisten, lengkap, dan terkini.
4. Karena tidak ada otoritas yang terpusat, pengguna dapat mengakses data dengan mudah karena tidak ada otoritas

tidak memiliki otoritas yang terpusat

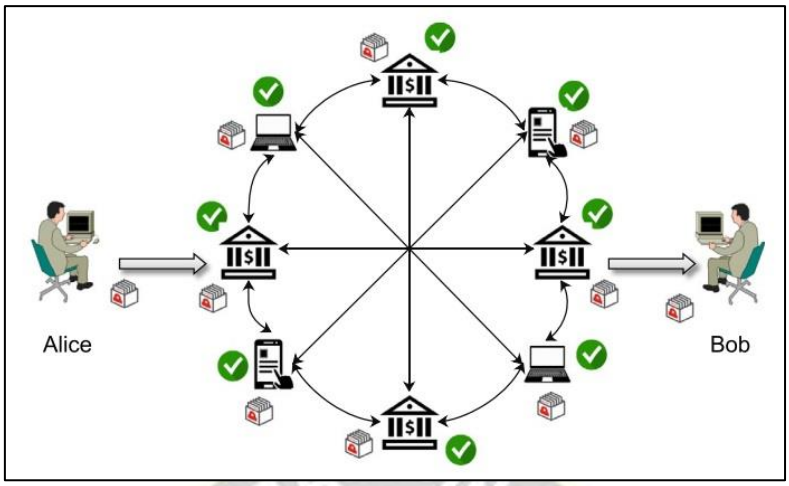
5. Setiap data disimpan dalam blokchain sehingga tidak dapat dimanipulasi.
6. Mekanisme yang digunakan oleh protokol blokchain memungkinkan data yang disimpan untuk dilindungi.
7. Karena sistem dibangun menggunakan jaringan peer-to-peer, ia memiliki kekebalan terhadap serangan cyber. Jaringan jaringan sistem

Menurut Dana dkk. (2018), Blockchain adalah buku besar digital yang terdiri dari semua transaksi yang ditandatangani secara kriptografis setelah validasi dan konsensus. Setelah validasi, setiap blok dihubungkan secara kriptografis dengan hash blok sebelumnya.



Gambar 2.4 *Distributed Ledger System Privat*

Blokchain, juga dikenal sebagai teknologi pembukuan terdistribusi, atau DLT, adalah ide yang memungkinkan setiap orang yang tergabung dalam jaringan terdistribusi untuk mengakses pembukuan. Konsep database terdistribusi yang sudah ada digunakan untuk mendukung konsep blockchain. Konsep ini muncul bersamaan dengan lahirnya bitcoin dan dimaksudkan untuk menyelesaikan masalah tidak adanya pihak ketiga seperti institusi pemerintah atau finansial yang dapat membangun kepercayaan antara pihak yang melakukan transaksi di lingkungan yang tidak aman (Bhushan dkk., 2021).



Gambar 2.5 *Distributed Ledger System Public*

Teknologi yang digunakan untuk basis data terdistribusi mirip dengan teknologi Blokchain secara konseptual. Information yang dicatat akan di save dan dibagikan kepada setiap anggota jaringan di basis data yang didistribusikan. Teknologi ini juga memungkinkan pihak ketiga untuk menghilangkan atau tidak ada mata uang kripto. Ide tentang pihak ketiga ini sudah lama ada, sekitar 30 tahun. Selain itu, dengan menggabungkan kunci publik kriptografi dan teknologi jaringan peer-to-peer, teknologi blokchain dapat mencegah pembelanjaan ganda atau lebih. Teknologi blokchain dapat didefinisikan secara literal sebagai kumpulan *part of infromation* yang terhubung satu sama lain dengan menggunakan fungsi enkripsi dan hash dari domain kriptografi (Chang dkk., 2020).

Karakteristik teknologi *Blokchain* berdasarkan dari penelitian sebelumnya adalah sebagai berikut :

1. Memiliki pembukuan yang tersebar atau tersebar di seluruh jaringan teman dan dapat diakses oleh semua anggota jaringan tersebut (Hughes dkk., 2019). Proses sebuah pembukuan adalah sebuah proses yang selalu diverifikasi dengan menerapkan konsensus yang telah disepakati oleh setiap simpul di dalam jaringan.
2. Informasi tetap tidak berubah dan aman dari perubahan karena ada proses verifikasi dan semua simpul memiliki nilai yang sama.
3. Transparansi untuk semua anggota, sehingga mereka dapat melihat informasi yang

tersimpan di Blokchain tetapi tidak dapat mengubahnya.

4. Memiliki Smart Contracts, sebuah metode atau media untuk menyimpan semua aturan dan kebijakan yang akan digunakan saat negosiasi ketentuan kontrak. Metode atau media ini akan secara otomatis disimpan dalam blokchain. Dengan demikian, dapat disimpulkan bahwa data yang disimpan melalui teknologi blokchain benar dan akurat.

2.3.2 Arsitektur *Blokchain* BSC

Arsitektur Smart Chain Binance mirip dengan Bitcoin. Menurut Persamaan 2.1. (m, n) = PoW(HN, Hn, d) (2.1), bukti pekerjaan dan informasi yang disimpan pada blok dibuat dengan algoritma BSC. Blok binance smart chain akan menyimpan state terbaru selain transaksi. Dalam persamaan ini, BSC disebut sebagai algoritma Proof of Work (Niu dkk., 2021).
Pada dasarnya, algoritma akan menemukan nilai mixHash dan nonce yang tepat berdasarkan masalah yang muncul selama proses pembentukan blok. Kedua nilai menunjukkan bahwa banyak proses komputasi telah terlibat dalam pembentukan blok.

2.3.3 *Smart Contract*

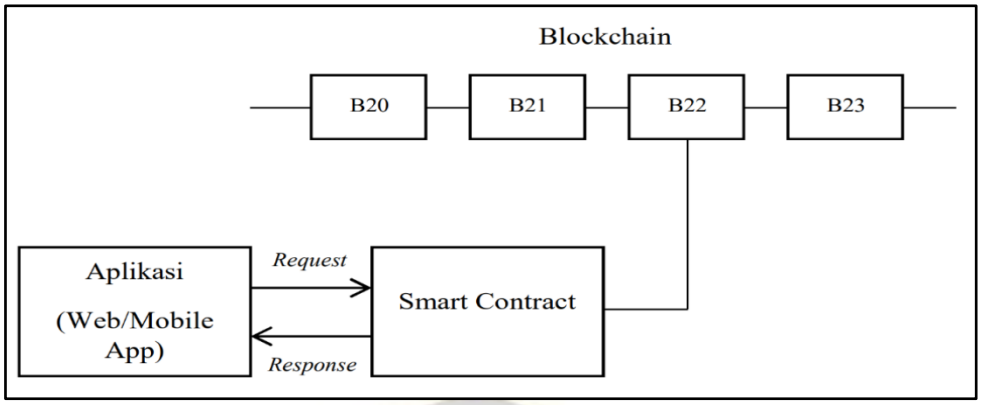
Perjanjian pintar adalah perjanjian elektronik yang dibuat dalam bentuk kode komputer yang dapat secara otomatis melaksanakan dan menerapkan syarat-syarat dalam perjanjian. Pada tahun 1994, pakar kriptografi, ilmuwan komputer, dan sarjana hukum Nick Szabo membuat smart contract, yang juga dikenal sebagai "smart contract". Perjanjian cerdas adalah kontrak yang mengikat antara beberapa pihak. Smart contract ini berbeda dari kontrak konvensional karena kodenya disimpan di blokchain.

Smart contract sering dikaitkan dengan teknologi blokchain, sebuah teknologi yang mengumpulkan dan mendistribusikan data dalam bentuk blok yang dihubungkan bersama dalam sebuah rantai, yang kemudian diverifikasi secara digital untuk memastikan bahwa data tersebut asli. Karena data yang dikumpulkan didistribusikan ke semua pihak yang menjalankan server, beberapa ahli berpendapat bahwa semua pengguna dapat mengakses blockchain.
Dalam kasus ini, Jerry Cuomo, Vice President of Blockchain Technology,

menyatakan bahwa smart contract merupakan kemajuan lebih lanjut dalam penerapan blokchain setelah cryptocurrency. Dia yakin bahwa smart contract dapat diterapkan pada banyak industri, seperti crowdfunding, layanan kesehatan, dan asuransi.

Sesuai dengan definisi di atas, Smart Contract adalah program komputer yang digunakan untuk otomatisasi persetujuan kedua belah pihak. Sebagai contoh, Nick Szabo mendefinisikan Smart Contract sebagai "sebuah janji atara kedua belah pihak yang diwakili oleh protokol digital", dan Max Raskin mendefinisikan Smart Contract sebagai "persetujuan yang diwakili oleh program komputer."
Smart contract dibuat untuk memastikan bahwa tidak ada kesalahan atau wanprestasi selama transaksi yang dapat membuat salah satu pihak dirugikan. Oleh karena itu, kodenya tidak dapat diubah dan permanen, jadi tidak ada yang dapat melanggar kontrak. Dalam kasus ini, smart contract dibedakan dari jenis kontrak elektronik lainnya yang tersedia di internet oleh sifat eksekusi otomatisnya.
Selain itu, smart contract menjamin kepercayaan kedua belah pihak karena dokumen saat ini akan ditulis dalam kode atau disebut enkripsi pada catatan sehingga tidak mungkin salah satu pihak mengatakan dokumen tersebut hilang. Selain itu, kriptografi dan enkripsi situs web akan menjaga dokumen yang ada aman dari peretas karena metode ini membutuhkan peretas cerdas yang tidak biasa untuk memecahkan kode.

Kedua teknologi tersebut digunakan dalam jaringan terdesentralisasi. Penggunaan smart contract memungkinkan pengembangan lebih lanjut dari teknologi blokchain. Pada awalnya, blokchain digunakan untuk melakukan tugas komputasi sederhana seperti pencatatan data transaksi, tetapi dengan integrasi smart contract, mereka sekarang dapat melakukan tugas komputasi yang lebih kompleks (Berdik dkk., 2021). Aplikasi desentralisasi menggunakan teknologi smart contract dan blokchain ini. Arsitekturnya dapat dilihat pada Gambar 2.6.

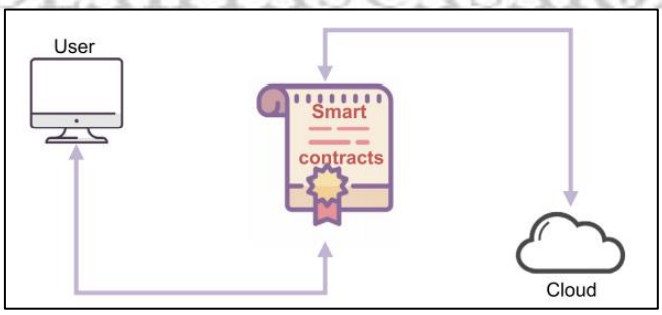


Gambar 2.6 Arsitektur Decentralized Application

Aplikasi terpusat terdiri dari tiga bagian utama: blokchain, aplikasi klien, dan smart contract. Aplikasi klien dapat berupa aplikasi web atau mobile yang memiliki kemampuan untuk berkomunikasi dengan smart contract melalui aplikasi programming interface (API). Smart contract sendiri bekerja seperti aplikasi server-side (back-end). Blokchain tersimpan di setiap node yang tergabung dalam jaringan tertentu (Lohmer & Lasch, 2020).

"Smart Contracts", yang didefinisikan oleh Nick Szabo sebagai "protokol transaksi terkomputerisasi yang menjalankan ketentuan perjanjian" (Szabo, 1994). Dalam naskah berikut, Nick Szabo menjelaskan tujuan umum Smart Contract:

"The general goals of smart contract design are to minimize malicious and unintentional exceptions, minimize the need for trusted intermediaries, and satisfy common contractual conditions (like payment terms, liens, confidentiality, and even enforcement)." Reducing fraud loss, arbitration and enforcement expenses, and other transaction costs are examples of related economic objectives.



Gambar 2.7 *Smart Contract*

Istilah "smart contract" digunakan ketika dua pihak melakukan kontrak secara otomatis di dalam sistem Blokchain. Setiap transaksi yang dilakukan dalam smart contract ini juga akan langsung diproses oleh Blokchain tanpa adanya pihak ketiga (Macrinici dkk., 2018).

Anda dapat melakukan transaksi yang dapat dipercaya tanpa pihak ketiga dengan smart contract. Lacak transaksi ini mustahil. Smart contract memiliki semua informasi tentang ketentuan kontrak dan melakukan semua yang dipikirkan secara otomatis. Bisnis jual beli e-commerce adalah salah satu contoh potensi penggunaan smart contract. Smart contract dapat mempercepat transaksi e-commerce. Ini dapat dicapai dengan menggunakan perjanjian pintar, yang menjamin bahwa perjanjian jual beli akan dilaksanakan (Macrinici dkk., 2018).

Smart contract memiliki lima (lima) jenis bentuk yang berbeda, masing-masing memiliki tujuan dan aplikasi unik. Kelima jenis smart contract ini adalah kontrak dasar token, kontrak penjualan massa, kontrak mintable, kontrak pengembalian, dan kontrak selesai. Empat jenis pertama dari kelima jenis smart contract ini sangat familiar digunakan dalam jual beli cryptocurrency.

Sedangkan *Terminable Contract* merupakan bentuk *smart contract* yang dapat digunakan untuk sistem *Blokchain* dalam jual beli barang online dan eksekusi program Blokchain dalam jasa keuangan.

2.3.4 *Binance Smart Chain*

Pada dasarnya, Binance Smart Chain adalah sistem pembayaran mata uang digital (cryptocurrency) yang terdesentralisasi, mirip dengan Bitcoin. Perbedaan utamanya dengan Bitcoin adalah bahwa sistemnya dibangun menggunakan bahasa pemrograman turing-complet, yang memungkinkan proses komputasional yang lebih kompleks pada struktur blokchain dibandingkan hanya mencatat data transaksional.

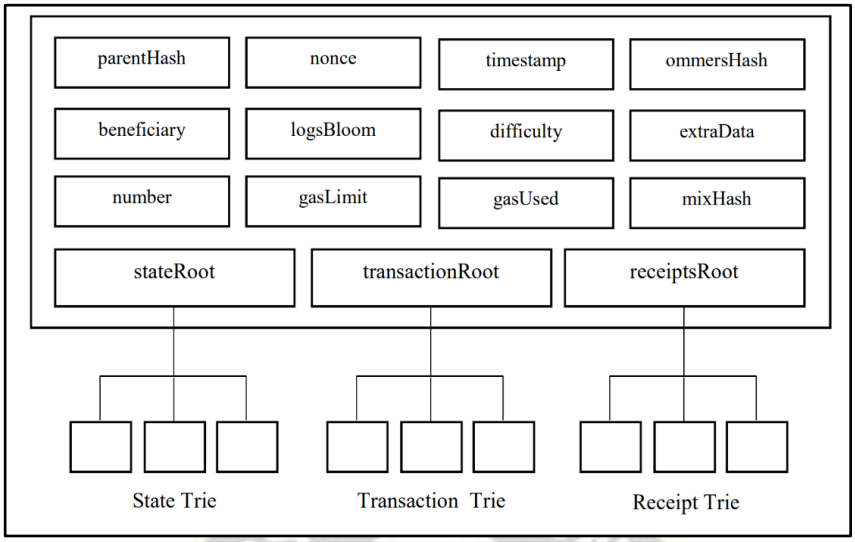
Seperti yang diketahui banyak orang, Binance Smart Chain adalah platform yang memungkinkan Anda membuat aplikasi yang bekerja secara decentralized. Mesin state berbasis transaksi pada dasarnya adalah Binance Smart ChainVirtual Machine (EVM). Suatu node dapat mengubah suatu state dengan mengirim transaksi yang mengandung input, yang memulai proses yang dikenal sebagai state

machine untuk mengubah suatu state. (Kasireddy, 2017). Binance Smart Chain menggambarkan setiap transaksi yang terjadi. Seperti Bitcoin, transaksi ini digabungkan dalam satu blok. Setiap blok terhubung ke blok sebelumnya. Protokol konsensus yang akan digunakan untuk membentuk blok adalah 'GHOST' (Greedy Heaviest Observed Subtree) (Kasireddy, 2017). Protokol 'GHOST' sebenarnya adalah protokol Proof of Work, tetapi dengan beberapa penyempurnaan. Protokol GHOST mengatasi masalah pembentukan blok tambahan secara bersamaan dengan blok yang tervalidasi. Orang yang menambang token dalam protokol GHOST tetap, yang membentuk blok stale, akan menerima hadiah. Ini digunakan karena pembentukan blok di binance smart chain dianggap lebih cepat dibandingkan dengan Bitcoin. Akibatnya, ada kemungkinan bahwa suatu stale blok akan terbentuk.

2.3.5 *Desentralisasi Finance* (DeFi)

Desentralisasi keuangan, juga disebut sebagai "desentralisasi keuangan", pada dasarnya dimulai dengan penemuan teknologi smart contract binance smart chain, yang merupakan sebutan untuk program komputer yang ditulis di atas blok binance smart chain. Blok binance smart chain terdiri dari tiga bagian, yaitu kumpulan blok header dari ommer, yang merupakan blok tetap yang dibuat secara bersamaan dengan blok yang tervalidasi, dan blok header sendiri, yang mengandung informasi transaksi yang tergabung dalam blok (Kasireddy, 2017).

Kecepatan transaksi sangat penting untuk memastikan mekanisme blokchain berjalan dengan baik karena kecepatan pembentukan blok binance smart chain hanya 15 detik dibandingkan dengan blokchain lain seperti Bitcoin (10 menit). Secara umum, protokol blockchain hanya memberikan hadiah kepada miner yang melakukan pembentukan blok yang tervalidasi, meningkatkan tingkat persaingan dalam proses pembentukan blok seiring dengan kecepatan pembentukan blok.



Gambar 2.8 Ilustrasi *Blok Header*

Dalam protokol binance smart chain, miner yang membentuk ommer juga akan mendapatkan hadiah, tetapi tidak sebanyak hadiah blok tervalidasi. Akibatnya, selama proses pembentukan blok, insentif miner dapat meningkat. Dengan bantuan protokol tertentu, blok binance smart chain berisi data khusus. Ini hampir sama dengan blok Bitcoin. Blok header akan dibuat dengan menggabungkan semua data yang ada. Blok header DeFi untuk suatu blok pada jaringan pintar binance digambarkan pada Gambar 2.8. Blok header berisi informasi berikut: (Hughes dkk., 2019).

1. *Parent Hash* : hash yang berasal dari *blok header* milik parent *blok*, yaitu *blok* terakhir sebelum *blok* terkait.
2. *Ommers Hash* : *hash* yang berasal dari daftar *ommer blok* terkait.
3. *Beneficiary* : *address* yang berasal dari *account* yang menerima biaya pembayaran dalam proses *mining blok* terkait.
4. *State Root* : *hash* yang berasal dari *state* yang tersimpan pada *blok* terkait. *Hash* ini terbentuk menggunakan struktur *Merkle Patricia Trie*.
5. *Transaction Root* : *hash* yang berasal dari transaksi yang tercatat pada *blok* terkait. Hash ini terbentuk menggunakan struktur *Merkle Patricia Trie*.
6. *Receipts Root* : hash yang berasal dari resi transaksi yang tercatat pada *blok* terkait. Hash ini terbentuk menggunakan struktur *Merkle Patricia Trie*.
7. Logs Bloom : struktur data bloom filter yang terdiri atas catatan informasi.

8. Difficulty : tingkat kesulitan yang berasal dari penyelesaian *blok* terkait.
9. Number : nilai penjumlahan yang berasal dari *blok* terkait.
10. Gas Limit : batas maksimum *gas* yang dapat digunakan dalam pemrosesan komputasi transaksi pada *blok* terkait.
11. *Gas Used* : jumlah total *gas* yang digunakan pada pemrosesan komputasi transaksi dari *blok* terkait.
12. *Timestamp* : *timestamp* yang berbasis unix dari pembentukan *blok* terkait.
13. *ExtraData* : data ekstra yang berhubungan dengan *blok* terkait.
14. *Mix Hash* : sebuah hash, yang apabila dikombinasikan dengan nonce, akan membuktikan bahwa blok terkait telah dibentuk melalui proses komputasi yang cukup.
15. *Nonce* : sebuah *hash*, yang apabila dikombinasikan dengan *Mix Hash*, akan membuktikan bahwa *blok* terkait telah dibentuk melalui proses komputasi yang cukup.

2.3.6 *Learning Management System*

Suatu perangkat lunak atau software yang dikenal sebagai Learning Management System (LMS) berfungsi untuk mengelola kebutuhan administrasi, mengajar, melakukan kegiatan, memdokumentasikan, melaporkan, dan menghubungkan LMS, materi pelatihan, dan kegiatan secara online (terhubung ke internet) (Bradley, 2021).

Media pembelajaran meningkatkan motivasi dan perhatian siswa untuk belajar, meningkatkan efisiensi dan efektivitas penyampaian informasi, dan membuat informasi lebih mudah dicerna. Dengan demikian, media pembelajaran memiliki peran penting dalam membantu siswa dalam proses pembelajaran. Oleh karena itu, sistem manajemen pembelajaran (LMS) dibangun untuk menjadi alat pembelajaran yang dapat diakses dan berbasis open source. Diharapkan dapat menarik perhatian siswa, memberikan motivasi, dan disesuaikan dengan minat siswa. Oleh karena itu, diharapkan informasi yang disampaikan melalui media tersebut dapat dipahami siswa dan diuji dalam proses belajar mengajar untuk mengetahui seberapa efektif penggunaan Sistem Manajemen Pembelajaran (LMS) (Zhong et al., 2018).

Administrasi, penyampaian materi, penilaian, pemantauan, dan komunikasi adalah semua bagian dari LMS. Materi-materi yang mencakup kompetensi pedagogik dan profesional yang digabungkan dengan paket multimedia (teks, animasi, video, dan suara) yang tersedia dalam LMS akan mempercepat penguasaan ilmu pengetahuan dan teknologi, yang akan memungkinkan peningkatan optimal kualitas pembelajaran. Dalam buku A Field Guide to Learning Management System (2009), Ryan K.Ellis menyatakan,

"A software program that automates the administration, monitoring, and reporting of training events is known as a learning management system.". Menurut Ryan K. Ellis, LMS adalah perangkat lunak atau software yang digunakan untuk menangani administrasi, dokumentasi, pencarian materi, laporan kegiatan, dan penyediaan materi pelatihan kegiatan belajar mengajar melalui internet. LMS digunakan untuk membuat bahan pembelajaran online berbasis web dan mengelola kegiatan dan hasil pembelajaran. Seringkali disebut sebagai platform pembelajaran konten (LMS) atau sistem manajemen konten pembelajaran (LCMS). LMS pada dasarnya adalah aplikasi yang berfungsi untuk mengotomatisasi dan memvirtualisasi proses belajar secara elektronik (Bradley, 2021).

2.3.7 Fitur *Learning Management System*

Fitur yang tersedia dalam LMS untuk institusi pendidikan adalah sebagai berikut:

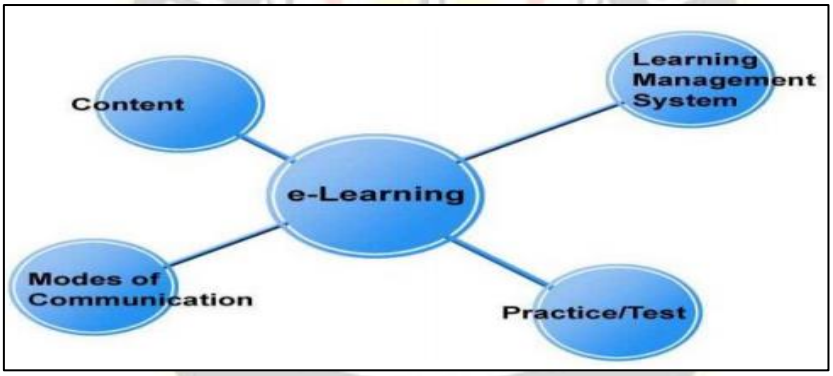
1. Pengelolaan hak akses pengguna (*user*)
2. Pengelolaan *courses*
3. Pengelolaan bahan ajar (*resource*)
4. Pengelolaan aktivitas
5. Pengelolaan nilai
6. Menampilkan nilai
7. Pengelolaan visualisasi LMS, sehingga bisa diakses dengan web browser.

Hari ini, tren terbesar di pasar "LMS" adalah kombinasi sistem ini dengan sistem manajemen bakat. Perangkat lunak manajemen bakat membantu mengelola, mengelola, menilai, mengembangkan, dan mempertahankan sumber daya penting organisasi. Lebih dari 70% perusahaan besar memiliki LMS pada tahun 2009, menurut penelitian Bersin. Hampir sepertiga dari perusahaan-perusahaan ini

mempertimbangkan untuk mengganti atau meng-“upgrade” sistem mereka dengan sistem manajemen bakat yang terintegrasi. Keunggulan LMS adalah program ini dapat membantu orang mengolah data administrasi dan pembelajaran melalui internet (Anggriawan, 2019).

2.3.8 *E - Learning*

"LMS" adalah istilah yang mengacu pada teknologi internet yang digunakan untuk mengirimkan berbagai produk yang memiliki kapasitas untuk meningkatkan pengetahuan dan keterampilan seseorang. Menurut perspektif ini, inti dari LMS adalah penggunaan teknologi internet atau tersedia secara online (Kumar dkk., 2020). Menurut Darin E. Hartley dalam Simanullang (2020), "LMS adalah sistem pendidikan yang menggunakan aplikasi elektronik untuk mendukung belajar mengajar dengan media internet, jaringan komputer, atau komputer sendiri."



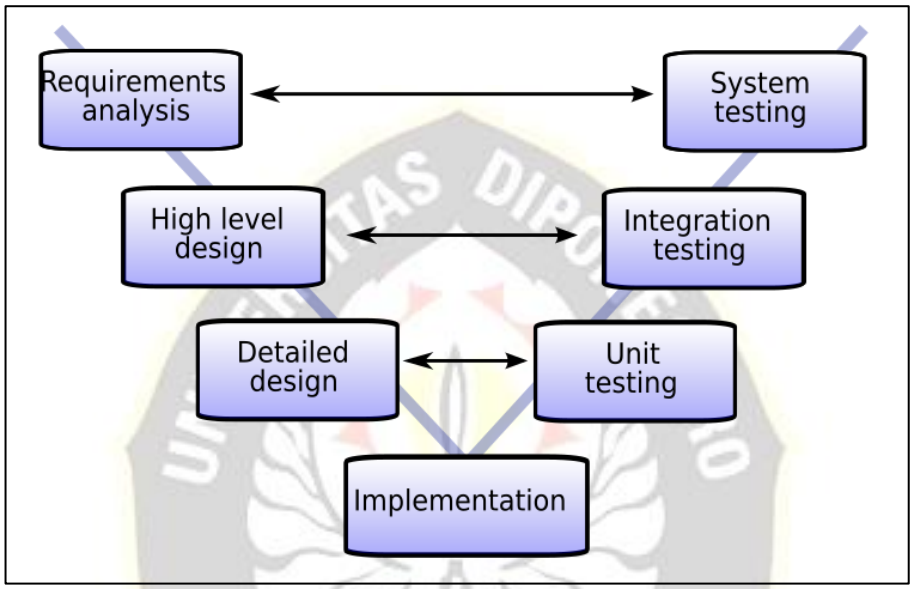
Gambar 2.9 LMS Ilustrasi

Berdasarkan uraian tentang definisi di atas, dapat disimpulkan bahwa LMS adalah proses dan kegiatan penerapan pembelajaran yang memanfaatkan media elektronik digital,baik berbasis internet yang bersifat online sebagai instrumen utamanya maupun berbasis *non-internet* seperti internet, tape video atau audio, penyiaran melalui satelit, televisi interaktif, dan CD-ROM (Ménard dkk., 2020).

2.3.9 Model V

Model V adalah perluasan dari model waterfall karena tahapnya mirip. V-Model menjalankan proses secara linear, tidak seperti model waterfall. Pemberitahuan kesalahan atau ketidaksesuaian dengan perancangan pengembangan software akan dikoreksi pada tahap sebelumnya dan kemudian,

yang disebut sebagai "cabang". Hubungan antara tahap pengujian dan pengembangan software ditunjukkan dalam Model Tahapan V (Zakaria & Fauziyah, 2022). Namun, meskipun model ini mirip dengan waterfall (Munar dan Sequeira, 2020), masing-masing tahap pengujian ditunjukkan pada gambar berikut:



Gambar 2.10 Model V

Berikut ini adalah uraian mengenai tahapan pada metode pengembangan yang digunakan, yaitu:

1. Analisis kebutuhan pengguna dan pengujian oleh pengguna: tahap ini hamper sama dengan model waterfall.
2. Spesifikasi kebutuhan pengguna dan pengujian sistem keseluruhan: Pada tahap ini, analis mulai merancang sistem berdasarkan dokumentasi kebutuhan pengguna yang telah dibuat sebelumnya. Sebagai hasilnya, analis membuat spesifikasi software yang mencakup organisasi sistem secara keseluruhan, struktur data, dan elemen lainnya. Pengujian sistem keseluruhan juga dilakukan untuk memastikan bahwa sistem memenuhi kebutuhan pengguna.
3. Desain dan pengujian arsitektural untuk unit integrasi. Hubungan antar interface, ketergantungan tabel dalam basis data, dan detail teknologi

yang digunakan adalah dasar pemilihan arsitektur, yang juga disebut sebagai desain Tingkat Tinggi. Pada titik ini, pengembang menguji setiap unit sistem yang memiliki input dan output. Seluruh sistem akan diuji lagi jika ada kesalahan.

4. Desain komponen design dan pengujian unit
   Perancangan dibagi menjadi modul-modul yang lebih kecil, yang juga disebut sebagai desain tingkat rendah. Untuk membuat coding lebih mudah bagi programmer, penjelasan yang cukup diberikan untuk setiap modul. Proses ini menghasilkan spesifikasi program, termasuk logika dan fungsi masing-masing modul, pesan kesalahan, dan proses input-output masing-masing modul, antara lain. Pengujian unit adalah pengujian yang dilakukan pada setiap unit setidaknya sekali untuk mengevaluasi alur sistem.
5. *Coding*
   Pada titik ini, setiap modul yang telah dibuat diprogram.

2.3.10 Pengujian Sistem

Pengujian perangkat lunak adalah serangkaian tindakan yang direncanakan dan sistematis yang digunakan untuk menguji atau mengevaluasi produk. Verifikasi adalah tahapan dari rekayasa perangkat lunak yang memastikan produk yang dibuat dari aktivitas pengembangan sesuai dengan spesifikasi yang ditentukan, dan validasi adalah tahapan dari rekayasa perangkat lunak yang memberikan penilaian produk yang tepat dan memenuhi keinginan pemangku kepentingan (Munar & Sequeira, 2020).

Perangkat lunak diuji dalam empat tahap untuk memastikan bahwa itu berjalan dengan benar. Sebagaimana tertera pada Tabel di bawah ini.

Tabel 2.6 Metode Pengujian

Teori Persamaan	Model-V	Teknik Pengujian	Aspek Uji
Verifikasi	Unit Testing	White Box	
	Integration Testing	Black Box	Functionality

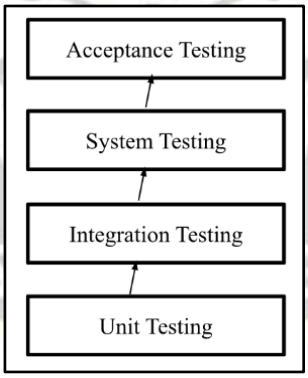
Teori Persamaan	Model-V	Teknik Pengujian	Aspek Uji
Validasi	System Testing Acceptance Testing	Instability Playability Test	Compatibility Playability

Tabel di atas menghubungkan metode pengujian sistem Model-V dengan teori Pressman. Model-V menguji aspek fungsionalitas, serta verifikasi bagian dari pengujian unit dan pengujian integrasi dari teknik pengujian yang digunakan. Selanjutnya, pengujian sistem dan penerimaan pengujian terdiri dari validasi, dan teknik pengujian Instability serta Playability digunakan. Pengujian meliputi aspek Compability dan Playability.

2.3.11 Fase Pengujian

Tabel 2.3 Fase Pengujian 'Model-V' menunjukkan gambar fase pengujian yang ada pada metode pengembangan sistem Model-V.

Tabel 2.7 Fase Uji 'Model-*V*'



Terdapat empat *fase* pengujian dalam metode pengembangan sistem Model-V adalah sebagai berikut :

1. *Unit Testing*

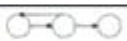
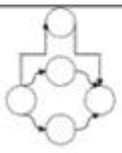
Proses pengujian perangkat lunak yang menguji masing-masing komponen dikenal sebagai pengujian unit. Tujuan dari pengujian unit adalah untuk memastikan bahwa setiap unit perangkat lunak menjalankan fungsinya sesuai desain. Seperti yang dikatakan Pressman, bagian pengujian berfokus pada verifikasi elemen terkecil dari perancangan perangkat lunak. Logika pemrosesan internal dan

struktur data di dalam komponen adalah fokus pengujian unit. Anda dapat menggunakan metode pengujian kotak putih atau fungsi untuk menguji unit.

Pengujian box putih menggunakan struktur kontrol dalam filosofi perancangan test case untuk membuat test case. Struktur kontrol ini dimasukkan ke dalam perancangan peringkat komponen. Membuat diagram alir, kompleksitas cyclomatic, dan jalur independen adalah bagian dari prosedur pengujian ini.(Pressman, 2020).

a) Notasi diagram alir

Notasi sederhana untuk representasi aliran kontrol disebut juga diagram alir (grafik program). Grafik alir menggambarkan aliran kontrol logika yang menggunakan notasi alir pada gambar di bawah ini:

Notasi	Arti
	Skema Sequence
	Skema If
	Skema While (...) DO (...)
	Skema Repeat (...) Until (...)
	Skema Case (...) Of

Gambar 2.11 Notasi Grafik Alir

Untuk menggambarkan *flow graph* dapat menggunakan simbol *flow graph* yang terbentuk lingkaran untuk mempresentasikan satu atau lebih *statement prosedural*. Anak panah pada flowgraph disebut *edges*. *Edge* harus berhenti pada suatu simbol, meskipun bila simpul tersebut tidak mempresentasikan *statement prosedural* (misalnya, lihat simpul untuk bangun *if-then-else).*

b) Jalur Independen

Jalur independen didefinisikan sebagai jalur yang melalui program

yang memasukkan setidaknya satu kumpulan pernyataan pemrosesan atau kondisi baru. Saat dinyatakan dengan flowgraph, jalur independen harus bergerak sepanjang setidaknya satu tepi yang tidak dilewatkan sebelum jalur tersebut ditentukan. Berikut contoh jalur independen;

Path 1 = 1-11

Path 2 = 1-2-3-4-5-10-1-1

Path 3 = 1-2-3-6-8-9-10-1-11

Path 4 = 1-2-3-6-7-9-10-1-11

Perhatikan bahwa masing-masing jalur harus memperkenalkan sebuah *edge* baru.Selanjutnya menentukanpondasi*cyclomatic complexity* adalah teori grafik, dan memberikan *metrics* perangkat lunak yang sangat berguna. *Cyclomatic complexity* dapat dihitung dalam salah satu dari tiga cara berikut:

a) Jumlah region grafik sesuai dengan *cyclomatic complexity*
b) *Cyclomatic complexity* V(G) untuk grafik alir dihitung dengan rumus: V(G) = E – N = 2
   Dimana : E = jumlah *edge* pada grafik alir
   : N = jumlah *node* pada grafik alir
c) *Cyclomatic complexity* V(G) juga dapat dihitung dengan rumus: V(G) = P + 1
   Dimana P = jumlah *predicate node* pada grafik alir

Pada unit testing, pengujian dilakukan dengan dengan memeriksa bagian kode program secara terpisah dari bagian yang lain. Pengujian dapat langsung dilakukan setiap kali sebuah kode unit (*event*, *procedure*, *function*) selesai dibuat. *Unit testing* dapat dilakukan dengan menggunakan metode *white box testing (functionality).*

2. *Integration Testing*

Menurut Pressman, pengujian integrasi adalah teknik untuk membangun arsitektur perangkat lunak yang juga melakukan pengujian untuk menemukan kesalahan antarmuka. Metode ini melibatkan penggabungan unit individu dan

pengujian sebagai kelompok, dengan tujuan untuk mengambil bagian yang diuji dan membuat struktur program yang telah ditentukan oleh perancangan.

Setelah pemeriksaan komponen dan unit selesai, langkah berikutnya adalah memastikan bahwa unit bekerja sebagai kombinasi, bukan lagi sebagai satu unit. Sebagai contoh, kita memiliki proses yang dilakukan oleh dua function, di mana satu function menggunakan hasil output dari yang lainnya. Kedua function ini bekerja dengan baik pada unit pengujian masing-masing.

Pada tahap pengujian integrasi, kita memeriksa apakah hasil interaksi kedua function bekerja sesuai dengan hasil yang diharapkan. Kita juga harus memastikan bahwa semua kondisi yang mungkin terjadi dari hasil interaksi antar unit menghasilkan output yang diharapkan. Pengujian black box (fungsi) dapat digunakan untuk menguji integrasi.

3. *System Testing*

Proses pengujian sistem, di mana perangkat lunak telah diselesaikan dan terintegrasi, menurut Pressman, terdiri dari berbagai tes untuk memastikan bahwa setiap komponen sistem telah terintegrasi dengan baik dan melakukan fungsi yang telah ditetapkan. Tujuan dari pengujian ini adalah untuk mengevaluasi kesesuaian sistem dengan persyaratan yang telah ditetapkan. Pengujian sistem juga mencakup pengujian aplikasi yang telah dikembangkan. Akibatnya, aplikasi harus berfungsi dengan baik dan menarik bagi pengguna. Untuk menguji sistem, Anda dapat menggunakan metode Blackbox atau tes yang tidak fungsional seperti pengujian konfigurasi, kompatibilitas, stres, dan kinerja, antara lain.

4. *Acceptance Testing*

Uji penerimaan, juga dikenal sebagai "uji penerimaan", adalah pemeriksaan formal yang menentukan apakah sistem memenuhi persyaratan penerimaan dan memastikan bahwa pengguna dapat menggunakannya. Tujuan pengujian ini adalah untuk menentukan kualitas perangkat lunak. Orang lain menambahkan pengujian penerimaan ke pengujian aplikasi. Pada tahap ini, pengguna akhir yang dipilih menguji aplikasi dan memberikan laporan masalah. Pengujian mensimulasikan penggunaan aplikasi nyata dalam

lingkungan nyata. Sebelum pengguna menyetujui dan menerima penerapan sistem aplikasi yang baru, proses ini adalah tahap terakhir. Akibatnya, masalah kecil seperti pengujian integrasi tidak diperhatikan pada tahap ini. Pengujian penerimaan juga melibatkan pengujian aplikasi secara keseluruhan, meskipun berbeda dengan siapa yang melakukannya.

Pada tahap ini, pengguna akhir yang dipilih menguji aplikasi dan memberikan laporan masalah. Pengujian mensimulasikan penggunaan aplikasi nyata dalam lingkungan nyata. Sebelum pengguna menyetujui dan menerima penerapan sistem aplikasi yang baru, proses ini adalah tahap terakhir. Oleh karena itu, masalah kecil seperti kesalahan pengetikan tidak akan ditangani pada tahap ini. Selama tes unit, komponen, dan integrasi, masalah kecil seharusnya ditangani. Usability, playability, atau metode Blackbox dapat digunakan untuk menguji acceptance.

Proses menguji perangkat lunak untuk verifikasi dan validasi dikenal sebagai pengujian perangkat lunak. Metode whitebox dan blackbox digunakan untuk verifikasi pada tahap unit pengujian dan integrasi. Selanjutnya, pada tahap validasi pengujian perangkat lunak, sistem pengujian dan penerimaan, dilakukan pengujian alpha untuk mengevaluasi kemampuan dan kesesuaian perangkat lunak.

Variabel penelitian ini adalah sebagai berikut:

a) Fungsionalitas adalah kemampuan perangkat lunak untuk melakukan tugas tertentu sesuai dengan kebutuhan pengguna dalam situasi tertentu;

b) Kompatibelitas adalah kemampuan perangkat lunak untuk beradaptasi dengan lingkungan baru; dan

c) Kemampuan untuk dimainkan adalah kemampuan perangkat lunak untuk digunakan dengan mudah dan menarik bagi pengguna ketika digunakan dalam konteks tertentu.