

BAB II

TINJAUAN PUSTAKA

2.1 Surat Tanda Daftar Budidaya(STD-B)

Surat Tanda Daftar Budidaya (STD-B) merupakan pendataan dan pendaftaran pekebun dengan luasan kurang dari 25 ha oleh pemerintah untuk 137 komoditas perkebunan, termasuk kelapa sawit, proses penerbitan didahului dengan pendataan, verifikasi dan validasi lapangan atas lahan milik pekebun yang mengajukan permohonan. STD-B juga menjadi bukti administrasi legal untuk mendorong peningkatan mutu kelapa sawit karena mencantumkan posisi lahan petani, kualitas bibit sampai pada hasil panen. STD-B ini akan menjadi modal bagi petani dalam menjual hasil panen maupun mengembangkan usaha[21].

PEMERINTAH KABUPATEN PASER
DINAS PERKEBUNAN DAN PETERNAKAN
 KOMPLEK PERANTORAN GENTUNG TENANG
 Gedung FL2 Raw.1-30, Kasuma Bangsa MAS-Kode 6027811
 TANJA PASER

**SURAT TANDA DAFTAR USAHA PERKEBUNAN UNTUK BUDIDAYA
(STD-B)**

Kabupaten/Kota : Paser
 Kecamatan : Kecamatan Kuaro
 Nomor : 525/53/STD-B/Bid.Bur/X/2021

A. Keterangan Pemilik :

1. N a m a : KUSWANTO
2. Tempat, Tanggal Lahir : Semarang, 16 Oktober 1982
3. Nomor KTP : 6401051610820001
4. A l a m a t : Desa Rangan, RT. 07
Kecamatan Kuaro

B. Data Kebun :

1. Lokasi/Titik Koordinat/Polygon Kebun :
 Desa Rangan, Kecamatan Kuaro
 a. Koordinat UTM_1 : X = 599640 Y = 9800079
 b. Koordinat UTM_2 : X = 599638 Y = 9800099
 c. Koordinat UTM_3 : X = 600067 Y = 9800093
 d. Koordinat UTM_4 : X = 600065 Y = 9800038
2. Status Kepemilikan Lahan : Sertifikat Hak Milik
3. N o m o r : 16040505.1.00138
4. Luas Lahan Kebun : 20.000 M2
5. Luas Lahan yang Ditanami : 20.000 M2
6. Jenis Tanaman : Kelapa Sawit
7. Produksi per Hektar per Tahun : -
8. Asal Benih : Penangkaran Benih Bersertifikat
9. Jumlah Pohon : 270 Pohon
10. Pola Tanam : Monokultur
11. Jenis Pupuk : Organik dan Non Organik
12. Mitra Pengolahan : Perusahaan Perkebunan
13. Jenis Tanah : Mineral
14. Tahun Tanam : Tahun 2021/2022
15. Usaha Lain di Lahan Kebun : -

STD-B ini tidak berlaku apabila terjadi perubahan terhadap informasi tersebut di atas.
 Tana Paser, 11 Oktober 2021

a.n. BUPATI PASER,
 Kepala Dinas Perkebunan Dan Peternakan

Diko Bawono, S.P., M.Si.
 NIP. 19750113 199803 1 003

Gambar 2. 1 Surat Tanda Daftar Budidaya

Status masa berlakunya STD-B tersebut, selama usaha budidaya tanaman perkebunan masih dilaksanakan oleh Pekebun yang namanya tertera, sehingga menjadi tidak berlaku lagi apabila terjadi perubahan atas pemilik, perubahan jenis tanaman dan perubahan luas kebun, tanahnya musnah dan/atau tidak diusahakan sesuai peruntukannya[22].

2.2 Sistem Informasi Geografis (SIG)

Sistem Informasi Geografis adalah sekumpulan terorganisir dari perangkat keras komputer, perangkat lunak, data geografis, dan personel yang dirancang untuk mengumpulkan, menyimpan, memperbaiki, menganalisis, dan menampilkan informasi berbasis geografis. Sebagai alat, Sistem Informasi Geografis berfungsi untuk mengambil, menyimpan, dan menampilkan data yang berkaitan dengan realitas geografis di bumi[23].

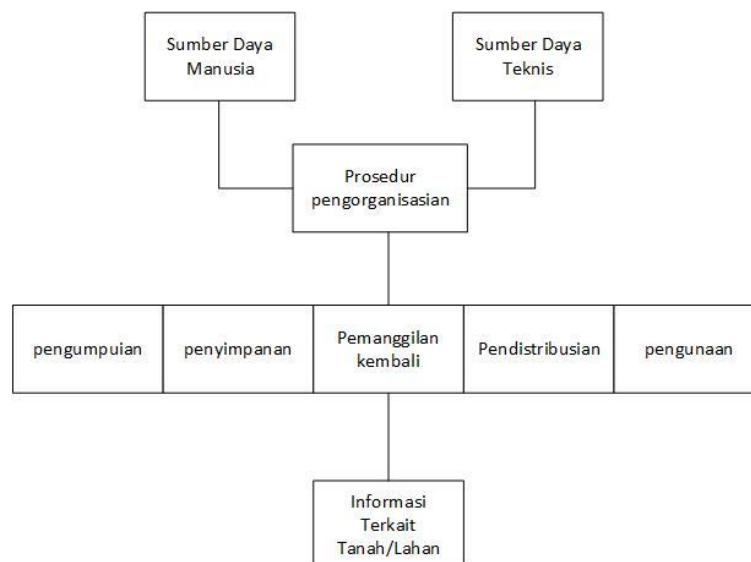


Gambar 2. 2 Sistem Informasi Geografis

Adapun turunan khusus dari Sistem Informasi Geografis (SIG) yaitu Sistem Informasi Pertanahan (SIP), yang juga dikenal sebagai *Land-related Information System* atau *Land-Record Information System* (LRIS). SIP mengacu pada informasi tentang kepemilikan tanah, menggantikan istilah SIG dalam konteks ini, dirancang sebagai sumber informasi rinci tentang kepemilikan tanah, data ini direkam dalam peta dengan skala 1:1,000 hingga 1:10,000, dan disimpan dalam sistem berbasis computer[24].

Konsep Sistem Informasi Pertanahan (SIP) meliputi komponen-komponen sumberdaya manusia, sumberdaya teknis, dan prosedur pengorganisasian yang bekerja sama untuk menghasilkan informasi terkait tanah guna memenuhi kebutuhan manajerial, untuk mencapai hasil yang diinginkan, diperlukan pengumpulan, penyimpanan, pemanggilan kembali, serta pemrosesan/analisis data sehingga informasi dapat dihasilkan dan didistribusikan terhadap informasi[25].

Berikut contoh gambar pada konsep sistem informasi pertanahan(SIP) pada gambar berikut



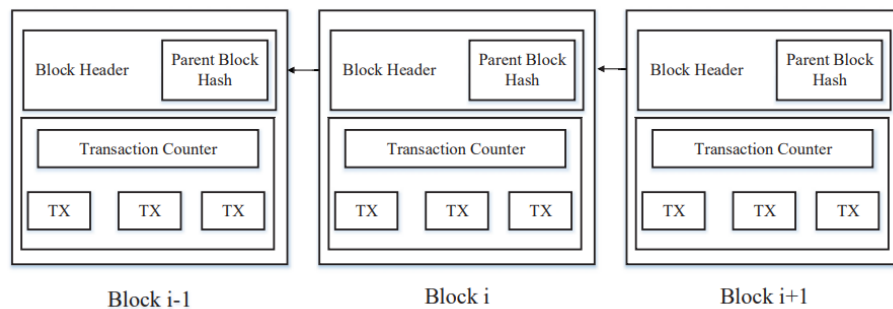
Gambar 2. 3 konsep sistem informasi pertanahan

SIP mencakup data spasial dan data non-spasial, yang keduanya penting dalam menggambarkan dan memahami kondisi lahan secara komprehensif. Data spasial mencakup informasi geografis dan visual, seperti model 3D dan peta geo-spasial, yang membantu mencegah terjadinya sertifikat ganda. Sementara itu, data non-spasial menyediakan deskripsi detail mengenai hak milik dan batas-batas properti serta perubahan tutupan lahan. Kombinasi kedua jenis data ini memungkinkan SIP untuk digunakan dalam berbagai aplikasi praktis, seperti pengelolaan properti, pengambilan keputusan berbasis spasial, dan pemantauan perubahan tutupan lahan, yang semuanya berkontribusi pada pengelolaan lahan yang lebih efisien dan terencana[26].

2.3 Blockchain

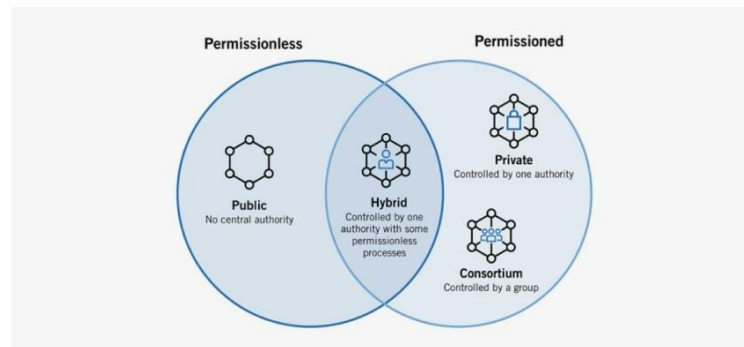
Blockchain adalah teknologi yang berkembang pesat dalam era revolusi industri 4.0, terutama dikenal karena fitur immutability-nya yang mengamankan data agar tidak dapat diubah oleh pengguna[27]. Awalnya diciptakan untuk

cryptocurrency bitcoin oleh Satoshi Nakamoto pada tahun 2008, blockchain menggunakan jaringan peer-to-peer di mana setiap node berfungsi sebagai buku besar bersama yang terdistribusi. Data yang tersimpan pada blok tidak dapat diubah, kecuali ada persetujuan mayoritas pengguna jaringan. Setiap perubahan data dicatat sebagai blok baru, memastikan bahwa tidak ada blok sebelumnya yang dihapus, menciptakan sistem yang transparan dan aman[28].



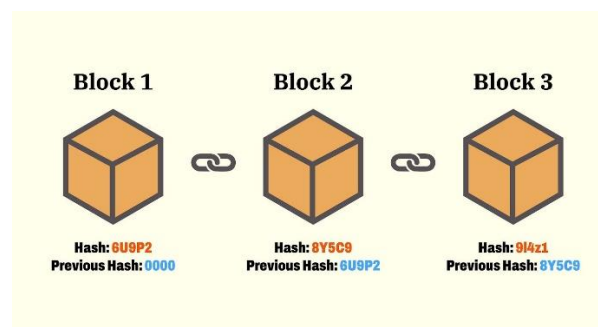
Gambar 2. 4 Ilustrasi *Blockchain*

Blockchain menggunakan buku besar (ledger) untuk mencatat dan menyimpan transaksi yang dibagikan dengan pihak-pihak terkait, sehingga efisiensi kinerja sangat penting[29]. Karakteristik utama blockchain meliputi desentralisasi, yang menghilangkan kebutuhan akan pihak ketiga dalam proses validasi transaksi. Ini meningkatkan kecepatan dan mengurangi biaya sistem. Persistensi memastikan bahwa data tidak mudah diubah atau dihapus, memberikan keamanan tinggi terhadap serangan dunia maya. Anonimitas melindungi identitas pengguna dengan memungkinkan interaksi melalui alamat pribadi yang tidak mengungkapkan identitas asli. Auditabilitas memudahkan pelacakan dan verifikasi transaksi, memastikan transparansi dalam setiap proses[30].



Gambar 2. 5 Jenis *Blockchain*

Blockchain dibagi menjadi tiga jenis: *public*, *private*, dan *hybrid*, *public blockchain*, seperti Bitcoin dan Ethereum, terbuka untuk semua orang dan bersifat open source, memberikan keamanan yang tinggi tetapi menghadapi tantangan dalam kecepatan dan penyimpanan karena skalanya yang besar, *private blockchain* lebih terbatas, hanya dapat diakses oleh anggota tertentu dan dikendalikan lebih ketat. Ini menawarkan kecepatan lebih tinggi dan biaya lebih rendah, *hybrid blockchain* menggabungkan kedua jenis sebelumnya, dengan aturan akses tertentu dan kontrol yang lebih besar, serta biaya rendah dan kapasitas penyimpanan yang besar, menjadikannya pilihan fleksibel untuk berbagai aplikasi[31].

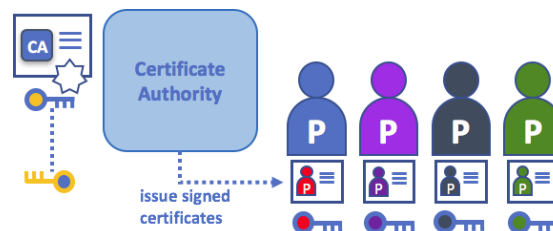


Gambar 2. 6 Struktur *Blockchain*

Struktur blockchain terdiri dari tiga komponen utama: blok, rantai, dan jaringan, blok berisi daftar transaksi yang dicatat ke dalam buku besar selama periode tertentu, rantai atau *chain* menyambungkan blok-blok ini menggunakan hash, yang memastikan bahwa setiap perubahan pada blok mudah terdeteksi, meningkatkan kepercayaan pada sistem[32].

2.4 Certificate Authority

Certificate Authority (CA) adalah sebuah sistem yang digunakan untuk mengelola sertifikat keamanan yang diperlukan dalam jaringan *blockchain Hyperledger Fabric*. Setiap organisasi memiliki CA tersendiri yang digunakan untuk mengelola identitas dan sertifikat keamanan organisasi tersebut, CA ini sangat penting karena mereka menentukan izin yang tepat atas sumber daya dan akses ke informasi yang dimiliki oleh para aktor dalam jaringan *blockchain*[33].



Gambar 2. 7 Ilustrasi *Certificate Authority*

Agar identitas dapat diverifikasi, identitas tersebut harus berasal dari otoritas terpercaya. *Membership Service Providers* (MSP) adalah otoritas terpercaya di *Hyperledger Fabric*. Lebih khusus lagi, MSP adalah komponen yang mendefinisikan aturan yang mengatur identitas yang valid untuk organisasi ini [34].



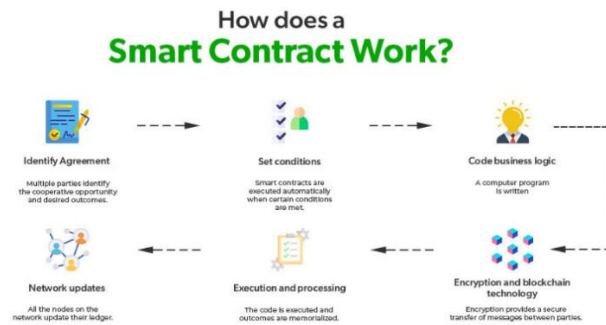
Gambar 2. 8 contoh sederhana CA dan MSP

Contoh sederhana mengenai CA Dan MSP, bayangkan berjalan-jalan ke supermarket untuk membeli makanan. Lihat tanda di kasir bahwa hanya Visa,

Mastercard, dan AMEX diterima, jika mencoba menggunakan kartu pembayaran alternatif, seperti "ImagineCard", kartu tersebut tidak akan diterima walaupun memiliki cukup dana, memiliki kartu kredit yang valid saja tidak cukup; toko juga harus menerimanya. CA dan MSP bekerja dengan cara yang sama: CA menyediakan daftar identitas, dan MSP menentukan anggota organisasi tertentu yang berpartisipasi dalam jaringan, otoritas CA dan MSP melakukan tugas yang serupa. MSP, seperti daftar penyedia kartu yang diterima oleh toko, menentukan aktor tepercaya dari jaringan pembayaran toko dan mengubah identitas yang dapat diverifikasi menjadi anggota jaringan blockchain. Di sisi lain, CA membagikan berbagai jenis identitas yang dapat diverifikasi [35].

2.5 Smart Contract

Smart contract atau disebut juga chaincode dalam teknologi blockchain melibatkan beberapa tahapan, termasuk mengidentifikasi perjanjian, menentukan kondisi, mengkodekan logika bisnis, memperbarui jaringan, melaksanakan dan memproses, serta enkripsi. Tahap pertama dalam pembuatan smart contract adalah mengidentifikasi perjanjian yang ingin dibuat oleh pihak-pihak yang terlibat dalam transaksi. Perjanjian ini dapat berupa perjanjian pembelian, sewa, atau jenis perjanjian lainnya. Perjanjian tersebut harus jelas dan ringkas untuk memastikan bahwa kedua belah pihak memiliki pemahaman yang sama tentang apa yang diharapkan. Setelah tahap identifikasi perjanjian, pihak-pihak yang terlibat menetapkan kondisi untuk perjanjian tersebut. Hal ini mencakup menentukan syarat dan ketentuan perjanjian, seperti harga, tanggal pengiriman, dan informasi relevan lainnya. Kondisi ini akan menjadi dasar dari smart contract. Setelah kondisi ditetapkan, logika bisnis dari perjanjian tersebut dikodekan ke dalam smart contract menggunakan bahasa pemrograman seperti Hyperledger Fabric. Kode tersebut menentukan aturan dan proses yang akan digunakan untuk mengeksekusi smart contract[36].



Gambar 2. 9 Ilustrasi *Smart Contract*

Selanjutnya, smart contract dienkripsi menggunakan kriptografi untuk memastikan keamanannya dan agar tidak dapat diubah. Enkripsi ini penting karena memastikan bahwa isi kontrak hanya dapat dilihat oleh pihak yang terlibat dalam perjanjian. Setelah smart contract tersedia dalam jaringan, akan dieksekusi secara otomatis ketika kondisi yang ditentukan terpenuhi. Misalnya, jika smart contract adalah untuk perjanjian pembelian, akan dieksekusi ketika pembeli mengirim jumlah cryptocurrency yang ditentukan ke dompet penjual. Smart contract kemudian memproses transaksi secara otomatis, mentransfer cryptocurrency dari pembeli ke penjual. Setelah tahap eksekusi dan proses, smart contract diunggah ke jaringan blockchain. Jaringan akan diperbarui untuk memasukkan smart contract baru, sehingga tersedia untuk semua peserta dalam jaringan. Smart contract didistribusikan ke seluruh node dalam jaringan, memastikan bahwa semua pihak memiliki akses ke informasi yang sama[37].

2.6 Hyperledger fabric

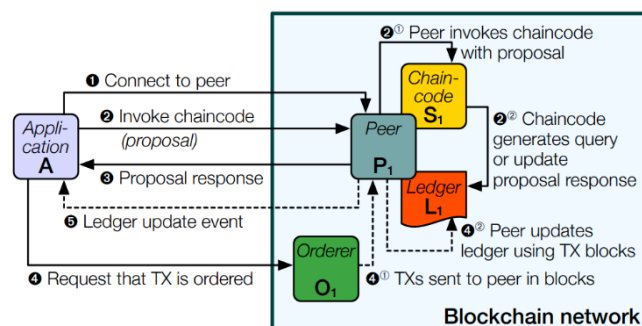
Hyperledger Fabric adalah sebuah platform sumber terbuka yang digunakan untuk membangun solusi buku besar terdistribusi. Dengan arsitektur modularnya, platform ini memberikan tingkat kerahasiaan, fleksibilitas, ketahanan, dan skalabilitas yang tinggi, sehingga memungkinkan solusi yang dikembangkan dengan Fabric dapat disesuaikan dengan berbagai industri. Fabric merupakan

kerangka kerja blockchain yang bersifat privat dan rahasia yang dikelola oleh Linux Foundation[38].



Gambar 2. 10 Logo Hyperledger Fabric

Selanjutnya Hyperledger Fabric (HF) adalah sebuah blockchain yang mendukung *smart contract*, dikenal sebagai *chaincode*. *Chaincode* di HF dapat diimplementasikan dalam tiga bahasa pemrograman tujuan umum: Go, Java, dan Node.js. HF menggunakan kontainer Docker untuk memfasilitasi penyebaran *chaincode*[39].



Gambar 2. 11 Ilustrasi Jaringan Hyperledger Fabric

Jaringan Hyperledger Fabric terdiri dari tiga jenis node: Klien/aplikasi (*clients*), rekan (*peers*) dan pemesan (*orderers*). Sebelum fungsi *chaincode* dapat dipanggil (*invoked*), *chaincode* harus diinstal dan diletakkan di sistem file serta diinstall pada *peer*. *Clients* dapat mengirim permintaan transaksi (*transaction proposal*) ke rekan (*peers*) untuk memanggil fungsi *chaincode* (Gambar 2-❶ dan Gambar 2-❷). Dalam fase pertama yaitu mengeksekusi atau pengesahan, *peer* mengeksekusi fungsi *chaincode* (Gambar 2-❷❶ dan Gambar 2-❷❶) dan mengirimkan respons kembali ke *clients* (Gambar 2-❸). Tanggapan transaksi, yang

ditandatangani oleh *peer*, berisi pesan respons eksekusi, *readset*, dan *writeset*. Setelah *client* mengumpulkan cukup tanggapan sesuai dengan kebijakan pengesahan, *client* mengirimkannya ke pemesan (Gambar 2-4), yang menempatkan transaksi ke dalam blok dan mendistribusikan blok tersebut ke rekan-rekannya (Gambar 2-4①). Pada fase ketiga yaitu validasi, *peers* memeriksa apakah kebijakan pengesahan terpenuhi dan memastikan tidak ada konflik baca-tulis antara transaksi yang berbeda. Pada fase terakhir akhirnya memasukkan transaksi ke dalam buku besar (Gambar 2-4②)[40].

2.7 Hyperledger Caliper

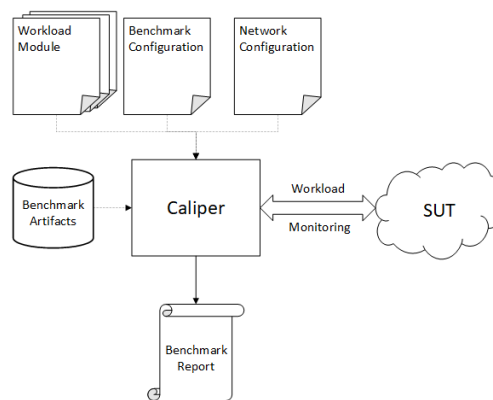
Hyperledger Caliper adalah alat ukur performa yang open-source dan otomatis untuk mengukur kinerja solusi berbasis blockchain. Alat ini dirancang untuk mengukur berbagai indikator kinerja, seperti latensi, throughput, dan penggunaan CPU. Dengan fitur-fitur ini, Hyperledger Caliper sangat berguna untuk mengukur kinerja berbagai solusi blockchain, memberikan data yang akurat dan dapat diandalkan bagi pengembang dan peneliti. Keberagaman, otomatisasi, dan dukungan multi-platform membuat alat ini sangat bermanfaat bagi mereka yang ingin mengoptimalkan kinerja aplikasi blockchain mereka[41].



Gambar 2. 12 Hyperledger Caliper

Salah satu kasus penggunaannya adalah dalam sistem manajemen pendaftaran sertifikat tanah berbasis blockchain, menunjukkan efektivitasnya dalam mengukur dan meningkatkan kinerja aplikasi yang berfokus pada desentralisasi data dan peningkatan transparansi dalam manajemen tanah [42]. Hyperledger Caliper telah diterapkan untuk menganalisis kinerja Hyperledger

Fabric, dengan fokus pada bagaimana parameter konfigurasi seperti batas waktu batch dan ukuran blok mempengaruhi kinerja keseluruhan sistem, ini penting untuk memahami bagaimana penyesuaian parameter dapat mengoptimalkan throughput dan latensi jaringan. Selain itu, Hyperledger Caliper membantu dalam menentukan konfigurasi optimal yang dapat meningkatkan efisiensi dan kinerja jaringan secara keseluruhan. Penggunaan alat ini dalam berbagai platform blockchain menunjukkan fleksibilitas dan kemampuannya dalam memberikan wawasan kinerja yang mendalam, yang penting bagi pengembangan dan pengelolaan aplikasi blockchain yang efisien [43].



Gambar 2. 13 Ilustrasi Kerja Hyperledger Valiper

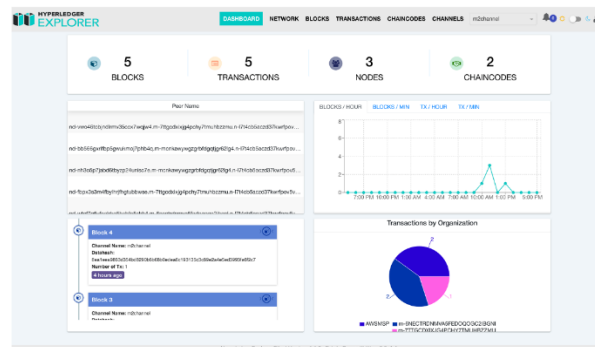
2.8 Hyperledger Explorer

Hyperledger Explorer adalah aplikasi penting yang dirancang untuk memvisualisasikan dan memantau data blockchain berbasis Hyperledger Fabric. Dengan fitur visualisasi yang canggih, pengguna dapat melihat data blockchain dalam bentuk tabel, grafik, atau chart, sehingga memudahkan pemantauan transaksi dan status blok. Fitur pencarian dan filter yang tersedia memungkinkan pengguna untuk menemukan data tertentu berdasarkan kriteria spesifik seperti tanggal, ID transaksi, atau ID blok, meningkatkan efisiensi dalam mengelola dan mengakses informasi yang relevan, fitur export juga menjadi keunggulan, karena pengguna dapat mengeksport data dalam format CSV, JSON, atau Excel untuk analisis lebih lanjut [44].



Gambar 2. 14 Hyperledger Explorer

Selain itu, Hyperledger Explorer dilengkapi dengan fitur keamanan yang kuat untuk mengamankan data blockchain, termasuk pengaturan hak akses dan level keamanan. Integrasi dengan aplikasi bisnis dan analisis lainnya memastikan bahwa data blockchain dapat digunakan secara maksimal dalam berbagai konteks. Fitur monitoring dan reporting membantu pengguna memantau status blockchain secara real-time dan membuat laporan yang komprehensif tentang transaksi dan blok. Dengan semua fitur ini, Hyperledger Explorer menawarkan solusi yang komprehensif dan efisien untuk pengelolaan data blockchain, menjadikannya alat yang sangat berharga bagi organisasi yang menggunakan Hyperledger Fabric[45].



Gambar 2. 15 Contoh Tampilan Hyperledger Explorer

2.9 InterPlanetary File System (IPFS)

InterPlanetary File System (IPFS) adalah sebuah protokol jaringan inovatif yang dirancang untuk memungkinkan berbagi dan mengakses file secara langsung melalui jaringan peer-to-peer (P2P), tanpa memerlukan server sentral. Dalam sistem ini, setiap file yang diunggah diberi hash unik yang berfungsi sebagai

identifikasi file tersebut, memungkinkan akses yang cepat dan aman. IPFS menggabungkan konsep penyimpanan terdistribusi di mana setiap node dalam jaringan dapat berfungsi sebagai penyimpan file. Ini membuat file yang diunggah dapat diakses dari mana saja dalam jaringan, memastikan bahwa data tetap tersedia meskipun beberapa node mengalami kegagalan. Selain itu, penggunaan hash untuk pengalamatan konten memastikan bahwa file yang diakses adalah versi yang tepat dan tidak berubah, meningkatkan integritas dan keamanan data[46].



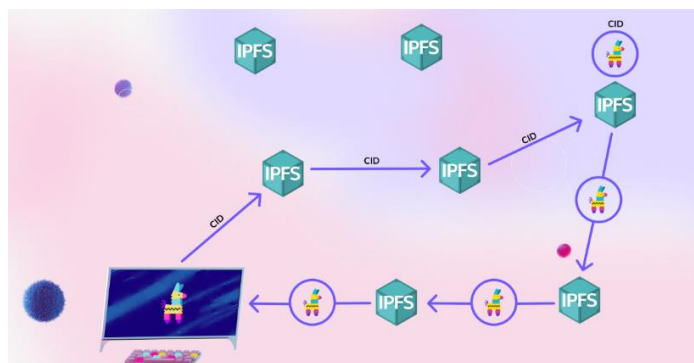
Gambar 2. 16 InterPlanetary File System

Tujuan utama IPFS adalah untuk meningkatkan keamanan, ketersediaan, dan efisiensi dalam berbagi dan mengakses file. Dengan menggunakan hash unik untuk identifikasi file, IPFS memastikan bahwa data yang diakses adalah asli dan tidak dimodifikasi, mengurangi risiko keamanan yang biasa terjadi pada sistem terpusat. Ketersediaan ditingkatkan melalui penyimpanan terdistribusi, yang menghindari ketergantungan pada satu titik kegagalan dan memungkinkan akses file yang lebih cepat dan andal dari berbagai lokasi dalam jaringan. Efisiensi juga ditingkatkan karena IPFS menghilangkan kebutuhan untuk mengunggah dan mengunduh file dari server sentral, mengurangi latensi dan biaya bandwidth. Contoh penggunaan IPFS meliputi berbagi file antar pengguna, mengakses data tanpa bergantung pada server sentral, dan mengembangkan aplikasi berbasis P2P yang memanfaatkan kekuatan jaringan terdistribusi untuk performa dan keamanan yang lebih baik[47].

2.10 Pinata

Pinata adalah sebuah platform inovatif yang memanfaatkan teknologi *blockchain* untuk memungkinkan pengguna mengunggah file ke jaringan InterPlanetary File System (IPFS), dalam prosesnya, file yang diunggah melalui

Pinata akan dikonversi menjadi kode hash unik atau yang biasa disebut *content identifier* (CID), yang dapat diakses di jaringan IPFS, pinata memastikan bahwa data yang disimpan terdesentralisasi, artinya data tidak bergantung pada satu server atau entitas pusat. Ini meningkatkan ketahanan data terhadap kegagalan sistem dan serangan siber. Dengan demikian, Pinata tidak hanya menawarkan platform penyimpanan data yang aman dan andal, tetapi juga memungkinkan pengguna untuk menikmati keuntungan dari jaringan terdistribusi IPFS[48].



Gambar 2. 17 Ilustrasi Pinata

Cara kerja Pinata cukup sederhana dan efisien. Pertama, pengguna mengunggah file ke platform Pinata, di mana file tersebut diubah menjadi kode hash unik(CID). Hash ini kemudian digunakan untuk menyimpan file di jaringan IPFS, yang terdiri dari sejumlah node yang berkomunikasi satu sama lain untuk menyimpan dan mengakses data, ketika pengguna ingin mengakses file, mereka cukup memasukkan kode hash unik(CID) tersebut, keuntungan utama dari Pinata termasuk desentralisasi, yang mengurangi ketergantungan pada pihak ketiga, serta keamanan yang ditingkatkan melalui penggunaan teknologi blockchain. Selain itu, proses pengunggahan dan pengaksesan file yang mudah dan efisien menjadikan Pinata solusi praktis untuk kebutuhan penyimpanan data modern[49].

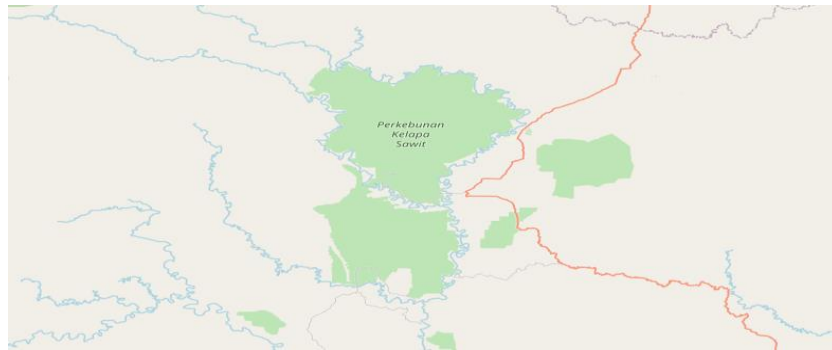
2.11 OpenStreetMap

OpenStreetMap (OSM) adalah proyek berbasis komunitas yang bertujuan untuk mengumpulkan dan mengembangkan data geografis yang dapat diakses secara bebas dan terbuka. Didirikan pada tahun 2004 oleh Steve Coast, OSM kini menjadi salah satu sumber data geografis terbesar dan paling populer di dunia. Pengguna OSM dapat mengumpulkan data geografis seperti jalan, bangunan, dan fitur lainnya menggunakan perangkat lunak seperti JOSM dan iD, yang kemudian disimpan dalam format XML. Data yang dikumpulkan terus dikembangkan dan diperbarui oleh pengguna lainnya melalui perangkat lunak yang memungkinkan perbaikan dan pembaruan data[50].



Gambar 2. 18 OpenStreetMap

Data OSM dapat dimanfaatkan oleh berbagai aplikasi dan layanan, seperti navigasi, perencanaan, dan analisis, serta dapat diakses secara bebas oleh siapa saja yang membutuhkannya. OSM menjadi platform yang memberdayakan pengguna untuk berpartisipasi dalam pengembangan data geografis dan menyediakan akses terbuka bagi siapa saja yang membutuhkan informasi tersebut, tampilan default pada peta menampilkan jalan, bangunan, sungai, dan fitur geografis lainnya dengan gaya kartografi yang bersih dan informatif. Tampilan ini paling sering digunakan untuk navigasi umum dan referensi geografis[51]. Adapun contoh terkait Tampilan default Maps dapat dilihat pada Gambar 2. 19 Tampilan default Maps



Gambar 2. 19 Tampilan default Maps

2.12 Mapbox

Mapbox adalah platform pemetaan dan lokasi berbasis cloud yang memungkinkan pengembang untuk menambahkan peta, data geospasial, dan navigasi ke aplikasi mereka. Salah satu fitur utama dari Mapbox adalah peta satelitnya, yang menawarkan gambar resolusi tinggi dari permukaan bumi yang diambil dari satelit. Gambar-gambar ini memberikan pandangan yang sangat rinci tentang berbagai fitur permukaan bumi, seperti bangunan, jalan, vegetasi, dan lainnya. Dengan menggunakan Mapbox, pengembang dapat memanfaatkan lapisan peta satelit untuk berbagai aplikasi, termasuk pelacakan aset, analisis lingkungan, dan navigasi[52].



Gambar 2. 20 Tampilan Peta Satelit

Peta satelit dari Mapbox juga dapat dikombinasikan dengan data lain untuk membuat visualisasi yang lebih kaya dan informatif. Misalnya, pengembang dapat menambahkan data vektor di atas peta satelit untuk menampilkan rute perjalanan, lokasi POI (Points of Interest), atau informasi real-time seperti lalu lintas. Mapbox menyediakan API yang mudah digunakan dan berbagai alat untuk kustomisasi peta, sehingga pengguna dapat menyesuaikan tampilan dan perilaku peta sesuai dengan kebutuhan aplikasi mereka. Dukungan untuk berbagai platform seperti web, iOS, dan Android membuat Mapbox menjadi pilihan yang fleksibel dan kuat untuk pengembangan aplikasi berbasis peta[53].

2.13 React Leaflet

React Leaflet adalah sebuah pustaka sumber terbuka yang menyediakan kumpulan komponen React untuk peta Leaflet, sebuah perpustakaan pemetaan sumber terbuka yang kuat yang memungkinkan pengembang menciptakan peta interaktif untuk aplikasi web. Dengan React Leaflet, integrasi peta Leaflet ke dalam aplikasi React menjadi lebih mudah[54].



Gambar 2. 21 React Leaflet

Komponen yang terdapat pada React Leaflet adalah:

1. Marker

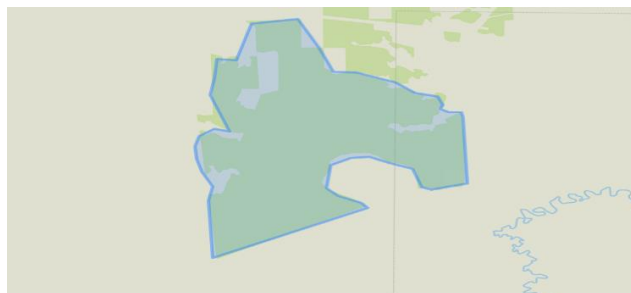
Marker pada maps adalah fitur yang umum digunakan dalam aplikasi peta digital untuk menunjukkan lokasi tertentu. Marker dapat berupa simbol yang digunakan untuk mengidentifikasi lokasi tersebut.



Gambar 2. 22 Tampilan Marker

2. Polygon

Polygon digunakan untuk memvisualisasikan wilayah atau area tertentu pada peta digital dan membantu pengguna dalam menemukan informasi yang dibutuhkan. Polygon dapat berupa poligon yang terdiri dari beberapa titik koordinat yang disambungkan untuk membentuk suatu area.



Gambar 2. 23 Tampilan Polygon

3. PopUp

Popup digunakan untuk memvisualisasikan informasi tambahan tentang lokasi tertentu dan membantu pengguna dalam menemukan informasi yang dibutuhkan. Popup dapat berupa gambar, atau teks yang digunakan untuk mengidentifikasi lokasi tersebut dan membantu pengguna dalam menemukan informasi yang dibutuhkan.

Gambar 2. 24 Tampilan *PopUp*

2.14 Node.js

Node.js adalah platform JavaScript yang digunakan untuk menjalankan program JavaScript di luar browser. Kelebihan utama Node.js adalah model I/O non-blocking yang membuatnya sangat efisien dalam menangani permintaan yang banyak. Selain itu, Node.js memungkinkan penggunaan JavaScript untuk menulis aplikasi server-side dan mengakses fitur sistem operasi seperti file I/O dan jaringan. Cara kerja Node.js menggunakan konsep non-blocking dan single-threaded, di mana permintaan klien diterima dan ditambahkan ke dalam Event Queue, kemudian diproses melalui Event Loop. Jika permintaan tersebut memerlukan akses ke sumber daya eksternal, sebuah thread dari Thread Pool akan menyelesaikannya sebelum mengirimkan respons kembali ke klien[55].



Gambar 2. 25 NodeJS

Kelebihan Node.js meliputi pemrosesan cepat berkat penggunaan JavaScript Engine V8 dari Google Chrome, dan mekanisme single-threaded yang membantu

server merespon secara asynchronous, membuatnya lebih scalable dibandingkan server tradisional. Node.js juga dilengkapi dengan library yang sangat lengkap, memudahkan pengembang untuk mengembangkan aplikasi. Selain itu, Node.js dapat menangani banyak permintaan secara serentak dan berjalan pada berbagai sistem operasi seperti Linux, Mac OS X, dan Windows tanpa perlu mengubah kode program. Contoh penggunaan Node.js dapat dilihat pada aplikasi LinkedIn yang mengurangi jumlah server dan meningkatkan performa, Uber yang mengembangkan aplikasi transportasi online yang efisien, dan PayPal yang memfasilitasi pengembangan server-side dan client-side secara praktis[56].

2.15 Typescript

TypeScript merupakan sebuah superset dari JavaScript yang dikembangkan oleh Microsoft, bertujuan untuk memperluas fitur-fitur bahasa JavaScript dengan menambahkan tipe data yang ketat dan dukungan untuk pemrograman berorientasi objek (OOP) serta modul. Perbedaan utama antara TypeScript dan JavaScript terletak pada penggunaan tipe data yang ketat, di mana TypeScript memungkinkan penggunaan tipe data yang lebih ketat untuk mencegah kesalahan penulisan kode saat kompilasi. Hal ini berbeda dengan JavaScript yang tidak memiliki tipe data yang ketat dan hanya mendeteksi kesalahan saat runtime. Dengan demikian, TypeScript menjadi pilihan yang lebih cocok untuk pengembangan aplikasi yang besar dan kompleks, sementara JavaScript lebih sesuai untuk pengembangan aplikasi yang lebih kecil[57].



Gambar 2. 26 TyperScript

2.16 Express

Express.js adalah framework web app untuk Node.js yang ditulis dengan bahasa pemrograman JavaScript. Framework open source ini dibuat oleh TJ Holowaychuk pada tahun 2010 lalu. Express.js adalah framework back end. Artinya, ia bertanggung jawab untuk mengatur fungsionalitas website, seperti pengelolaan routing dan session, permintaan HTTP, penanganan error, serta pertukaran data di server[58].

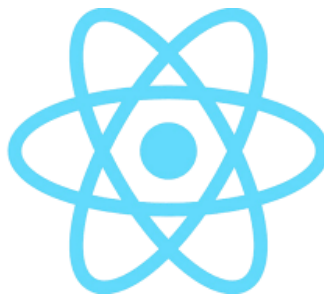


Gambar 2. 27 *Express*

Berbeda dengan kebanyakan framework yang punya aturan ketat mengenai penulisan kode, seperti ketika Anda belajar AngularJS, framework Laravel, dan Django, Express.js ini memiliki aturan coding yang lebih fleksibel. Pasalnya, Express.js di JavaScript adalah framework yang menggunakan pendekatan Unopinionated dalam proses pengembangan. Artinya, pengguna punya kebebasan dalam menentukan metode yang akan digunakan untuk mengeksekusi suatu perintah [59].

2.17 ReactJS

ReactJS adalah library JavaScript yang diciptakan oleh Facebook untuk membangun antarmuka pengguna yang interaktif dan stateful. Dengan pendekatan berbasis komponen, ReactJS memungkinkan pengembang untuk membagi UI menjadi bagian-bagian kecil yang dapat digunakan kembali, membuatnya sangat cocok untuk aplikasi web dengan antarmuka yang kompleks dan memerlukan performa tinggi. Salah satu keunggulan utama dari ReactJS adalah kemudahan belajar dan penggunaannya, terutama bagi pengembang yang sudah menguasai dasar-dasar JavaScript. Penulisan komponen di ReactJS memadukan objek JavaScript dengan HTML, memberikan pengalaman pengembangan yang efisien dan efektif. Selain itu, fitur seperti Virtual DOM dan Single Way Data Flow memastikan bahwa aplikasi yang dibangun dengan ReactJS memiliki performa optimal dan mudah di-debug[60].



Gambar 2. 28 ReactJS

Keunggulan lain dari ReactJS adalah dukungannya terhadap SEO dan komponen yang dapat digunakan kembali. Dengan Virtual DOM, ReactJS mampu melakukan rendering yang lebih cepat dan efisien, sementara optimasi SEO yang ditawarkan memastikan aplikasi web memiliki kecepatan halaman yang baik, yang berkontribusi positif terhadap ranking mesin pencari. ReactJS juga mempermudah proses pengembangan dengan komponen siap pakai yang dapat digunakan kembali, sehingga mempercepat waktu development dan mengurangi usaha yang diperlukan untuk memelihara aplikasi. Namun, seperti teknologi lainnya, ReactJS juga memiliki kekurangan, seperti kebutuhan untuk memahami konsep JSX dan mengatasi kurva belajar yang terkait dengan state management dan tools pendukung lainnya[61].

2.18 Mysql

MySQL yang dibaca “MY-ES-KYOO-EL” merupakan sistem manajemen *database* yang bersifat *open-source* yang menggunakan perintah dasar atau bahasa pemrograman yang berupa *structured query language* (SQL) yang cukup populer di dunia teknologi. MySQL berguna sebagai database. Saat ini, relational database management system (RDBMS) MySQL telah dipakai lebih dari 66 juta pengguna di seluruh belahan dunia, adapun MySQL memiliki dua bentuk lisensi, yaitu *Free Software* dan *Shareware* atau perangkat lunak bermilik yang penggunaannya terbatas. Jadi, MySQL adalah database server gratis dengan lisensi GNU *General Public License* (GPL) yang bisa dipakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada[62].



Gambar 2. 29 MYSQL

Dalam istilah pemrograman, SQL sendiri menjadi bahasa yang dipakai di dalam pengambilan data pada *relational database* atau *database* yang terstruktur. Dengan kata lain, MySQL merupakan *database management system* yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan *database serve*[63].

2.19 CouchDB

CouchDB merupakan salah satu database NoSQL berbasis dokumen yang masuk dalam pembinaan Apache Foundation. Bersama dengan Cassandra, Hadoop, Tomcat, Lucene, CouchDB tinggal serumah dalam dunia open source. Salah satu

database NoSQL yang memiliki konsep sama dengan CouchDB adalah MongoDB[64].



Gambar 2. 30 CouchDB

CouchDB juga menjadi salah satu pionir database NoSQL sebelum digandrungi oleh developer di masa kini. CouchDB dibangun menggunakan bahasa pemrograman Erlang yang mengandalkan pada reliabilitas dan konkurensi. CouchDB juga menjadi salah satu basis dalam pengembangan IBM Cloudant, sebuah solusi database berbasis cloud yang ditawarkan IBM kepada enterprise[65].

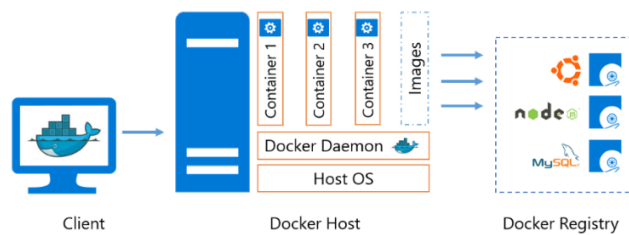
2.20 Docker

Docker adalah platform perangkat lunak yang memungkinkan pengembang untuk membuat, menguji, dan menerapkan aplikasi dengan cepat dan mudah. Dengan Docker, pengembang dapat mengemas aplikasi dan semua dependensinya ke dalam unit standar yang disebut kontainer. Kontainer ini berisi semua yang diperlukan untuk aplikasi tersebut agar dapat berjalan, termasuk pustaka, alat sistem, kode, dan waktu proses[66].



Gambar 2. 31 docker

Salah satu fitur utama Docker adalah kemampuannya untuk menciptakan ruang isolasi yang disebut kontainer. Kontainer ini berfungsi sebagai lingkungan terisolasi di mana aplikasi dapat dijalankan tanpa mempengaruhi lingkungan sistem yang lain. Ini memungkinkan aplikasi untuk berjalan secara konsisten di berbagai lingkungan, dari lingkungan pengembangan lokal hingga lingkungan produksi di cloud[67].

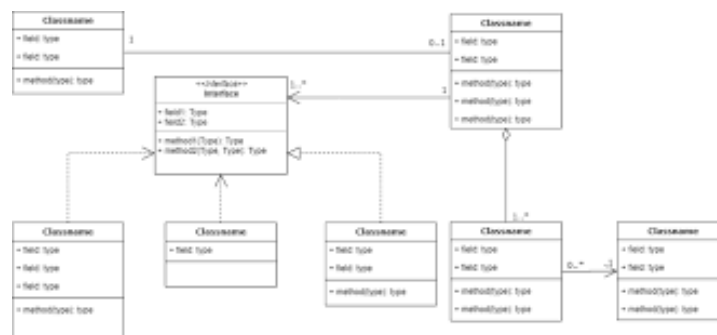


Gambar 2. 32 Ilustrasi Docker

Dengan Docker, pengembang dapat dengan mudah membuat, menginstal, dan menjalankan aplikasi kompleks di berbagai platform tanpa perlu khawatir tentang perbedaan konfigurasi atau dependensi sistem. Ini mempercepat siklus pengembangan dan penyebaran aplikasi, serta memungkinkan aplikasi untuk berjalan dengan konsisten di berbagai lingkungan[68].

2.21 Unified Modeling Language (UML)

Unified Modelling Language (UML) adalah suatu metode dalam pemodelan secara visual yang digunakan sebagai sarana perancangan sistem berorientasi objek. Awal mulanya, UML diciptakan oleh Object Management Group dengan versi awal 1.0 pada bulan Januari 1997. UML juga dapat didefinisikan sebagai suatu bahasa standar visualisasi, perancangan, dan pendokumentasian sistem, atau dikenal juga sebagai bahasa standar penulisan blueprint sebuah software[69].

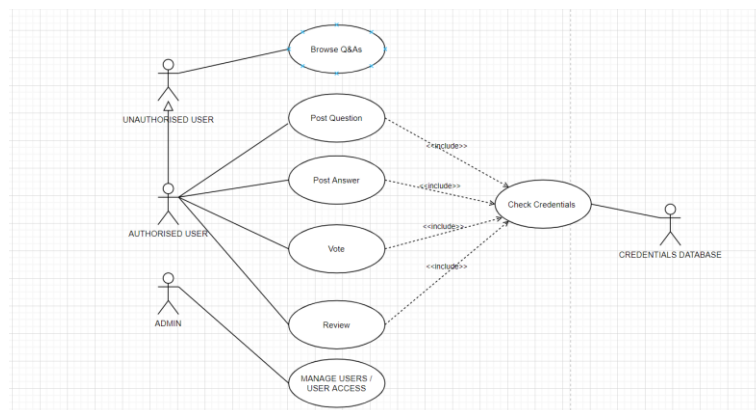


Gambar 2. 33 *Unified Modelling Language*

UML diharapkan mampu mempermudah pengembangan piranti lunak serta memenuhi semua kebutuhan pengguna dengan efektif, lengkap, dan tepat. Hal itu termasuk faktor-faktor scalability, robustness, security, dan sebagainya, terdapat beberapa diagram yang biasanya digunakan untuk memodelkan analisis fungsional dalam rangka pengembangan perangkat lunak. Berikut diantaranya diagram yang umum digunakan [70]:

a). Use Case Diagram

Use case diagram adalah satu dari berbagai jenis diagram UML (Unified Modelling Language) yang menggambarkan hubungan interaksi antara sistem dan aktor. Use Case dapat mendeskripsikan tipe interaksi antara si pengguna sistem dengan sistemnya.

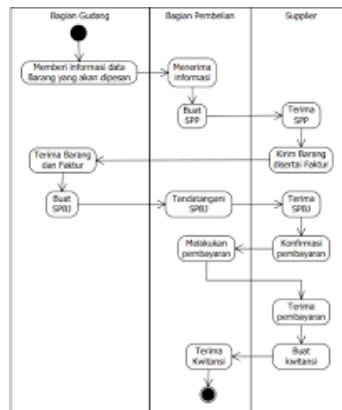


Gambar 2. 34 use case

Jadi, secara umum bisa didefinisikan bahwa pengertian use case diagram adalah teknik guna menunjukkan hubungan antara pengguna sistem dengan sistem itu sendiri. Hasilnya berupa skema sederhana untuk memudahkan user membaca dan memahami informasi yang diberikan.

b). Activity Diagram

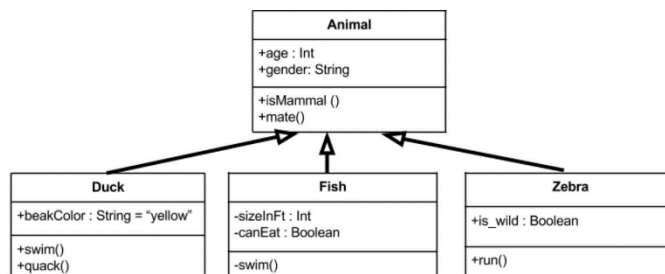
Activity Diagram, dalam bahasa Indonesia diagram aktivitas, yaitu diagram yang dapat memodelkan proses-proses yang terjadi pada sebuah sistem. Runtutan proses dari suatu sistem digambarkan secara vertikal. *Activity diagram* merupakan pengembangan dari Use Case yang memiliki alur aktivitas. Alur atau aktivitas berupa bisa berupa runtutan menu-menu atau proses bisnis yang terdapat di dalam sistem tersebut.



Gambar 2. 35 Activity Diagram

c.) Class Diagram

Class diagram atau diagram kelas adalah salah satu jenis diagram struktur pada UML yang menggambarkan dengan jelas struktur serta deskripsi *class*, atribut, metode, dan hubungan dari setiap objek. Ia bersifat statis, dalam artian diagram kelas bukan menjelaskan apa yang terjadi jika kelas-kelasnya berhubungan, melainkan menjelaskan hubungan apa yang terjadi.

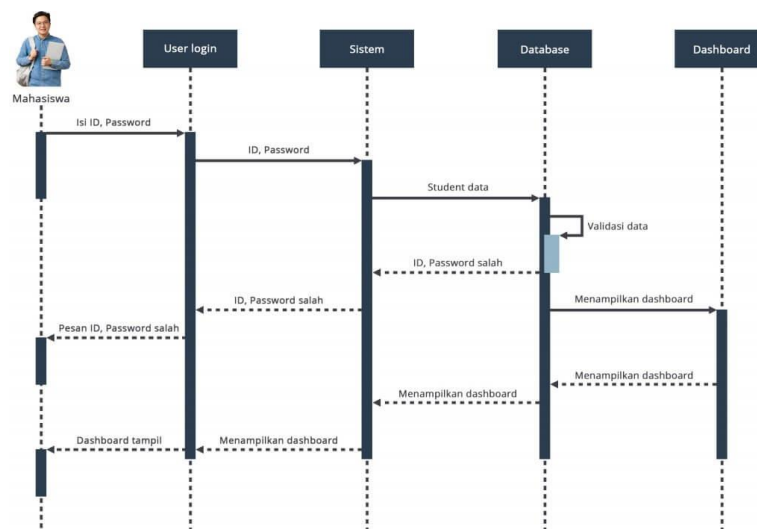


Gambar 2. 36 Class diagram

Desain model dari diagram kelas ini sendiri dibagi menjadi dua bagian. Bagian pertama merupakan penjabaran dari database. Bagian kedua merupakan bagian dari modul MVC, yang memiliki *class interface*, *class control*, dan *class entity*.

d). Sequence Diagram

Sequence diagram atau diagram urutan adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci. Selain itu sequence diagram juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaannya. Objek-objek yang berhubungan dengan berjalannya proses operasi biasanya diurutkan dari kiri ke kanan.



Gambar 2. 37 Sequence Diagram

Diagram urutan dapat digunakan untuk menggambarkan serangkaian langkah yang dilakukan sebagai respon dari sebuah peristiwa untuk menghasilkan suatu output tertentu. Sequence diagram berhubungan dan berkaitan erat dengan use case diagram, di mana satu *use case diagram* akan menjadi satu diagram sequence.