

BAB II

LANDASAN TEORI

2.1 Sistem

Menurut Nugraha dan Sofyan dalam jurnal (Yuliana, Zahrudin, dan Utari, 2018) “Sistem sebagai suatu entitas yang tersusun dari dua atau lebih komponen atau subsistem yang saling berkolaborasi untuk meraih tujuan bersama.”. Sedangkan, Menurut Tata Sutabri dalam jurnal (Abdurahman, Safi, dan Abdullah, 2018) “Sistem merupakan sekelompok elemen yang saling terkait secara erat, bekerja bersinergi untuk mencapai suatu tujuan tertentu.”

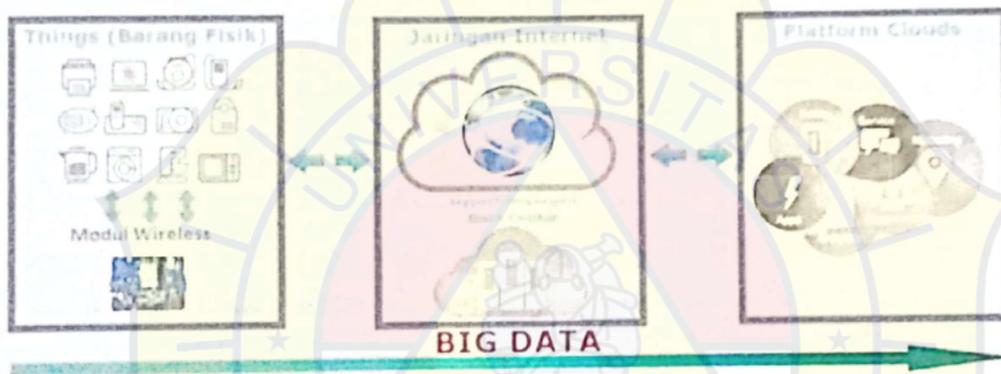
2.2 Parkir

Menurut Hobbs (1995), Parkir dapat dijelaskan sebagai tindakan meletakkan atau menyimpan kendaraan pada lokasi tertentu, dengan durasi tergantung pada penyelesaian kebutuhan dari pengemudi. Sedangkan Menurut Warpani (1990), Parkir dapat didefinisikan sebagai tindakan meletakkan kendaraan pada suatu area atau lokasi dengan durasi parkir tertentu. Dengan kata lain, kebutuhan parkir secara sederhana dapat diartikan sebagai kebutuhan akan ruang untuk memarkir sejumlah kendaraan selama jangka waktu tertentu

2.3 Internet Of Things (IOT)

Menurut Internet of Things (IoT), atau yang juga dikenal sebagai IoT, adalah suatu konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang terus-menerus terhubung. Konsep ini memungkinkan penghubungan mesin, peralatan, dan objek fisik lainnya melalui sensor jaringan dan aktuator untuk mengumpulkan data dan mengelola kinerjanya sendiri. Hal ini memungkinkan mesin untuk bekerja sama dan bahkan mengambil tindakan berdasarkan informasi baru yang diperoleh secara mandiri.

Gambar Arsitektur IOT:



Gambar 2.1. Arsitektur IOT

Berikut adalah gambaran umum tentang arsitektur Internet of Things (IoT) yang dapat diterapkan dalam sebuah sistem parkir:

1. **Sensor Parkir:** Sensor parkir dipasang di setiap slot parkir untuk mendeteksi ketersediaan tempat parkir. Sensor ini bisa berupa sensor magnetik, sensor ultrasonik, atau sensor lainnya yang dapat mendeteksi keberadaan kendaraan.
2. **Node Pengumpul Data (Data Gathering Node):** Node ini bertugas mengumpulkan data dari sensor parkir secara berkala atau berdasarkan peristiwa tertentu, seperti deteksi kendaraan yang baru masuk atau keluar.

dari slot parkir. Node ini juga bisa bertindak sebagai titik akses jaringan untuk mengirimkan data ke server.

3. **Jaringan Komunikasi:** Data yang dikumpulkan oleh node pengumpul data perlu dikirim ke server pusat untuk diproses lebih lanjut. Ini bisa dilakukan melalui berbagai jaringan komunikasi, seperti jaringan Wi-Fi, jaringan seluler (3G/4G/5G), atau protokol komunikasi khusus seperti LoRaWAN atau NB-IoT.
4. **Gateway IoT:** Gateway IoT berfungsi sebagai perantara antara node pengumpul data dan server pusat. Gateway ini bertanggung jawab untuk mengumpulkan data dari node, meneruskannya ke server, dan mengelola protokol komunikasi antara perangkat IoT dan infrastruktur jaringan.
5. **Server Pusat:** Server pusat merupakan pusat pengolahan dan manajemen data dari seluruh sistem IoT. Di sini, data dari sensor parkir diolah, disimpan, dan dianalisis untuk menghasilkan informasi yang berguna, seperti ketersediaan tempat parkir, laporan penggunaan, dan analisis tren.
6. **Aplikasi Seluler dan Web:** Aplikasi seluler dan web berfungsi sebagai antarmuka pengguna untuk berinteraksi dengan sistem parkir. Melalui aplikasi ini, pengguna dapat melihat informasi tentang ketersediaan tempat parkir, melakukan pemesanan atau pembayaran tempat parkir, dan menerima notifikasi terkait dengan status parkir.
7. **Sistem Manajemen Parkir:** Sistem manajemen parkir berperan dalam mengatur operasi parkir secara keseluruhan. Ini termasuk manajemen tarif parkir, pengelolaan pemesanan, pelaporan transaksi, dan integrasi dengan sistem pembayaran.

8. **Keamanan dan Privasi:** Lapisan keamanan dan privasi perlu diterapkan di setiap titik dalam arsitektur IoT, termasuk enkripsi data, autentikasi pengguna, dan perlindungan terhadap serangan siber.

2.4 Web

Menurut Agus Hariyanto (2015), situs web dapat diartikan sebagai serangkaian halaman yang menampilkan berbagai informasi berupa teks, gambar, animasi, suara, dan video. Hal ini mencakup elemen-elemen statis maupun dinamis yang membentuk suatu struktur terkait satu sama lain, dengan setiap halaman terhubung melalui jaringan hyperlink. Sedangkan Menurut Robi Abdulloh (2015), Web dapat dianggap sebagai kumpulan halaman-halaman yang memuat informasi dalam format data digital, termasuk teks, gambar, video, audio, dan animasi lainnya. Semua konten tersebut dapat diakses melalui koneksi internet.

2.5 Mobile

Menurut Pressman dan Bruce (2021), Aplikasi mobile adalah aplikasi yang didesain khusus untuk platform mobile seperti iOS, Android atau Windows Mobile. Aplikasi seluler dalam banyak kasus memiliki antarmuka pengguna dengan mekanisme interaksi unik yang disediakan oleh platform seluler, interoperabilitas dengan sumber daya berbasis web yang menyediakan akses ke berbagai informasi terkait aplikasi, dan kemampuan pemrosesan lokal. mengumpulkan, menganalisis, dan memformat data dengan cara yang paling sesuai untuk platform seluler. Selain itu, aplikasi seluler menyediakan fungsi penyimpanan persisten untuk platform.

2.6 Xampp

Menurut Jubilee Enterprise XAMPP merupakan aplikasi web server yang menggabungkan Apache, MySQL, dan phpMyAdmin. Nama XAMPP sendiri berasal dari singkatan X (cross-platform), Apache Server, MySQL, dan Python. Dengan adanya huruf "X" yang menandakan cross-platform, XAMPP dapat diinstal pada berbagai sistem operasi, termasuk Windows, Linux, MacOS, dan Solaris.

2.7 Basis Data

Menurut Utomo, Database atau basis data adalah sekelompok data yang saling terkait. Data ini umumnya diatur dalam bentuk tabel yang memiliki hubungan satu sama lain. Setiap tabel memiliki field atau kolom yang digunakan untuk menyimpan informasi tertentu. Dengan memanfaatkan database, dapat dilakukan pengorganisasian dan pengelolaan data secara lebih efisien, serta melakukan kueri untuk mendapatkan informasi yang dibutuhkan.

2.8 Mysql

Menurut Kurniawan Rulianto, MySQL merupakan sistem basis data atau penyimpanan informasi yang mendukung skrip PHP. Selain itu, MySQL memiliki kueri atau bahasa SQL (Structured Query Language) yang sederhana dan menggunakan karakter escape yang serupa dengan PHP. Selain kelebihan tersebut, MySQL juga dikenal sebagai basis data paling cepat pada saat ini.

2.9 UML

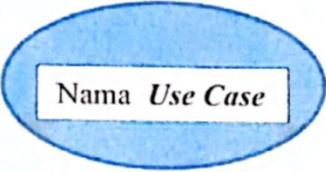
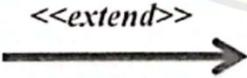
Unified Modeling Language (UML) adalah alat atau model yang digunakan untuk merancang pengembangan perangkat lunak berbasis object-oriented. UML bukan hanya membantu dalam merancang blueprint sistem, tetapi juga memberikan

standar penulisan yang komprehensif. Standar tersebut mencakup konsep proses bisnis, penulisan kelas-kelas dalam bahasa pemrograman yang spesifik, skema database, dan komponen yang diperlukan dalam sistem perangkat lunak. Dengan mengadopsi UML, pengembang aplikasi mampu secara jelas dan sistematis menggambarkan arsitektur dan desain perangkat lunak yang diinginkan, sehingga memfasilitasi kolaborasi tim dan meningkatkan pemahaman tentang sistem yang akan dikembangkan. Penggunaan UML juga memberikan kerangka kerja yang kokoh untuk mengelola kompleksitas proyek pengembangan perangkat lunak, yang pada akhirnya mendukung pencapaian tujuan proyek dengan lebih efisien.

2.9.1 Use Case Diagram

Use Case dibawah ini menjelaskan interaksi apa saja yang dapat dilakukan oleh user dan admin. Diantaranya User melakukan Parkir dengan melakukan scan Qr code lalul memasuki tempat Parkir jika sudah keluar dari parkir dengan melakukan scan Qr code. Lalu Admin diharuskan login terlebih dahulu pada website untuk verifikasi, lalu dapat melakukan manajemen user, manajemen data Parkir Serta manajemen harga parkir, dan logout.

Tabel 2.1. Diagram Use Case

Simbol	Deskripsi
<p data-bbox="323 302 446 331"><i>Use Case</i></p> 	<p data-bbox="794 302 1230 611">Fungsionalitas yang disediakan sistem sebagai unit-unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>Use Case</i></p>
<p data-bbox="323 649 486 678">Aktor/Actor</p> 	<p data-bbox="794 649 1222 1037">Mempresentasikan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem. <i>Actor</i> hanya berinteraksi dengan use case tetapi tidak memiliki kontrol atas <i>Use Case</i></p>
<p data-bbox="316 1075 598 1104">Asosiasi / association</p> 	<p data-bbox="794 1075 1241 1328">Komunikasi antara aktor dan <i>Use Case</i> yang berpartisipasi pada <i>Use Case</i> atau <i>Use Case</i> memiliki interaksi dengan aktor</p>
<p data-bbox="311 1366 542 1395">Ekstensi / extend</p> 	<p data-bbox="794 1366 1241 1865">Hubungan ekstensi antar <i>Use Case</i> berarti bahwa suatu <i>Use Case</i> merupakan tambahan kegunaan dari <i>Use Case</i> yang lain jika kondisi atau syarat tertentu dipenuhi. Jika prosedur dari <i>Use Case</i> merupakan alternatif untuk</p>

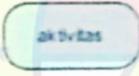
	menjelaskan use-case lain. <i>Use Case</i> akan dikerjakan apabila salah satu syarat terpenuhi
<p>Generalisasi / <i>generalization</i></p> 	<p>Generalisasi pada aktor dan use case digunakan untuk menyederhanakan model dengan cara menarik keluar sifat-sifat pada aktor maupun use case yang sejenis. • Untuk mengetahui kapan dibutuhkan generalisasi berdasarkan pada tujuan: 1. Mekanisme berbeda dengan satu tujuan yang sama (generalisasi use case). 2. Agen berbeda dengan satu tujuan yang sama (generalisasi aktor).</p>

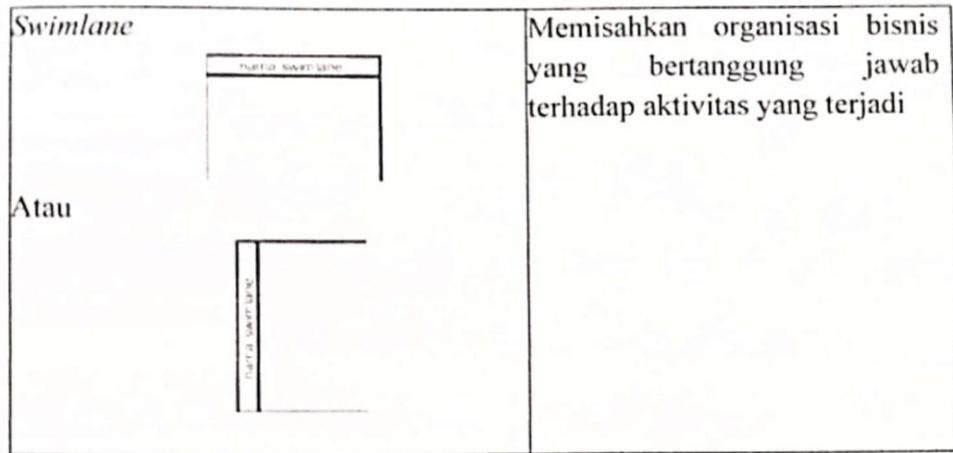
2.9.2 Activity Diagram

Menurut Rosa dan Shalahuddin (2014), Activity Diagram adalah suatu teknik pemodelan yang digunakan untuk mengilustrasikan rangkaian kerja atau kegiatan dari sebuah sistem atau proses bisnis dalam perangkat lunak. Perlu ditekankan bahwa diagram aktivitas mencerminkan kegiatan yang dilakukan oleh sistem, bukan oleh aktor. Diagram ini memberikan gambaran tentang urutan kegiatan dalam sistem dan bagaimana kegiatan-kegiatan tersebut saling terhubung

berdasarkan urutan waktu atau kronologis. Selain itu, diagram ini menggunakan simbol-simbol khusus untuk merepresentasikan elemen-elemen berbeda dalam sistem. Melalui penggunaan Diagram Aktivitas, kita dapat dengan jelas memvisualisasikan langkah-langkah yang terlibat dalam sistem dan memahami interaksi di antara kegiatan-kegiatan tersebut. Berbagai simbol pada diagram ini membantu menggambarkan dengan lebih jelas komponen-komponen yang ada dalam sistem.. Berikut adalah simbol-simbol yang ada pada diagram aktivitas (Rosa dan Shalahuddin, 2014:162):

Tabel 2.2. Simbol-simbol Activity Diagram

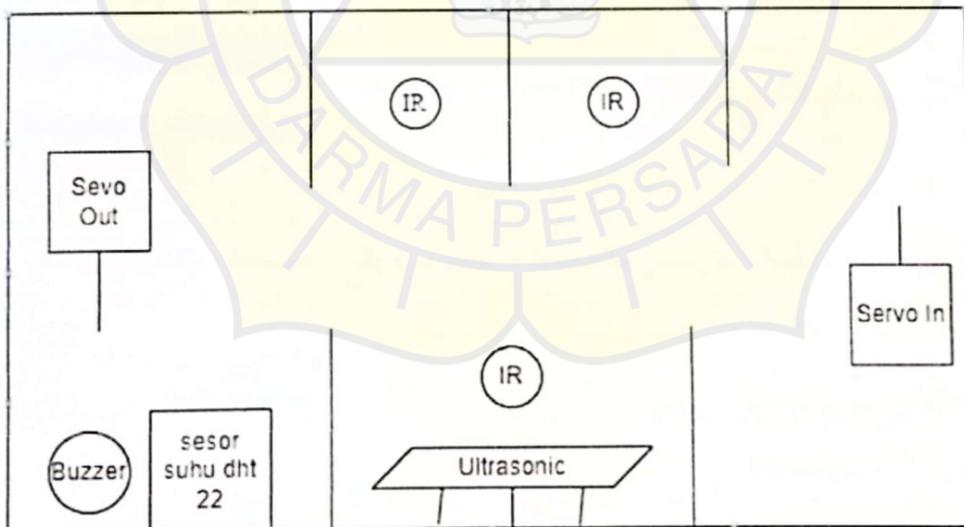
Simbol	Definisi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.



2. 10 Arduino

Arduino adalah nama keluarga papan mikrokontroler yang awalnya diproduksi oleh perusahaan proyek cerdas. Salah satu pencipta adalah Massimo Banzi. Papan ini adalah perangkat keras "sumber terbuka" sehingga siapa pun dapat membuatnya. Arduino dirancang untuk memudahkan pengujian atau implementasi berbagai perangkat mikrokontroler.

Berikut adalah Prototype dari sistem Parkir IOT:



Gambar 2.2. Skema Prototype Sistem parkir IOT

Berikut sensor-sensor dan aktuator yang di gunakan:

Aktuator adalah sebuah perangkat atau mekanisme yang bertugas untuk mengubah energi listrik, hidraulis, atau pneumatik menjadi gerakan atau tindakan fisik. Aktuator sering digunakan dalam berbagai sistem otomatis, seperti sistem kontrol industri, kendaraan, dan robotika. Contoh aktuator termasuk motor listrik, silinder hidrolik, katup pneumatik, dan solenoid. Fungsi utama aktuator adalah untuk menghasilkan gerakan atau tindakan berdasarkan sinyal kontrol yang diberikan kepadanya

1. Servo



Gambar 2.3. Servo

Servo adalah aktuator yang digunakan untuk menghasilkan gerakan presisi berdasarkan sinyal kontrol yang diberikan kepadanya.

2. Buzzer



Gambar 2.4. Buzzer

Buzzer adalah jenis aktuator suara yang menghasilkan suara tertentu ketika diberi daya.

3. Sensor Inframerah:



Gambar 2.5. Inframerah

Sensor inframerah digunakan untuk mendeteksi keberadaan kendaraan di area parkir. Mereka bekerja dengan mengirimkan sinar inframerah yang tidak terlihat dan kemudian mendeteksi pantulan kembali dari objek. Ketika objek terdeteksi, sistem dapat mengirimkan sinyal untuk menunjukkan bahwa slot parkir telah terisi.

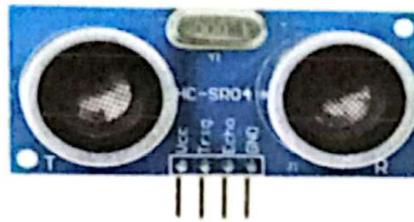
4. Sensor Suhu DHT22:



Gambar 2.6. Suhu DHT22

Sensor ini digunakan untuk mengukur suhu dan kelembaban di sekitar area parkir. Ini bisa membantu dalam memonitor kondisi lingkungan di mana kendaraan diparkir. Dalam beberapa kasus, suhu dan kelembaban dapat mempengaruhi sistem kendaraan atau keamanan, seperti mengaktifkan sistem pendingin atau mengurangi kelembaban untuk mencegah korosi.

5. Sensor Ultrasonik:



Gambar 2.7. Ultrasonik

Sensor ultrasonik sering digunakan untuk mengukur jarak antara kendaraan dan objek di sekitarnya, seperti tembok atau kendaraan lain. Mereka mengirimkan gelombang ultrasonik dan kemudian mengukur waktu yang dibutuhkan gelombang itu untuk kembali setelah memantul dari objek. Informasi ini digunakan untuk menentukan apakah kendaraan telah mencapai batas parkir atau untuk membantu dalam manuver parkir yang tepat. **Arduino Mega** adalah salah satu varian papan Arduino yang populer karena memiliki jumlah pin I/O digital dan analog yang besar, cocok untuk proyek-proyek yang membutuhkan lebih banyak koneksi. Berikut adalah spesifikasi :

Tipe: Arduino Mega 2560

Spesifikasi Umum:

Mikrokontroler: ATmega2560

Tegangan Operasional: 5V

Tegangan Input (rekomendasi): 7-12V

Tegangan Input (batas): 6-20V

Pin Digital I/O: 54 (diantaranya 15 dapat digunakan sebagai output PWM)

Pin Input Analog: 16

Arus DC per Pin I/O: 20 mA

Arus DC untuk Pin 3.3V: 50 mA

Memori Flash: 256 KB (8 KB digunakan untuk bootloader)

SRAM: 8 KB

EEPROM: 4 KB

Kecepatan Clock: 16 MHz

DataSheet Arduino Mega 2560

Arduino Mega 2560 adalah papan mikrokontroler berdasarkan ATmega2560 (datasheet). Ini memiliki 54 pin input / output digital (dimana 14 dapat digunakan sebagai output PWM), 16 input analog, 4 UART (port serial perangkat keras), osilator kristal 16 MHz, koneksi USB, colokan listrik, header ICSP, Dan tombol reset. Ini berisi semua yang dibutuhkan untuk mendukung mikrokontroler; Cukup hubungkan ke komputer dengan kabel USB atau nyalakan dengan adaptor AC keDC atau baterai untuk memulai. Mega kompatibel dengan kebanyakan perisai yang dirancang untuk Arduino Duemilanove atau Diecimila

Spesifikasi Hardware ASUS VivoBook dengan AMD Ryzen 3:

Prosesor: AMD Ryzen 3 (misalnya Ryzen 3 3200U atau Ryzen 3 3250U)

RAM: 4GB atau 8GB DDR4 (Beberapa model mungkin memiliki opsi untuk meningkatkan RAM)

Penyimpanan: SSD 128GB atau HDD 1TB (Beberapa model mungkin menawarkan kombinasi SSD dan HDD, atau penyimpanan yang berbeda)

Kartu Grafis: AMD Radeon Graphics (GPU terintegrasi)

Layar: Layar LED 14 inci atau 15,6 inci dengan resolusi HD atau Full HD

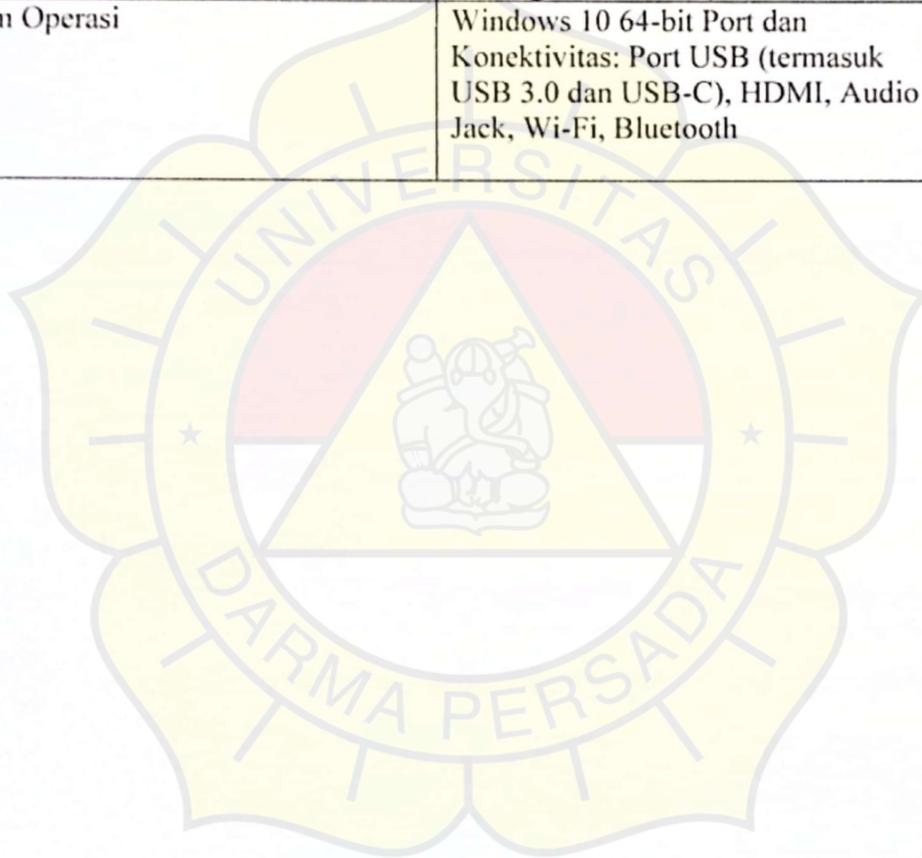
Sistem Operasi: Windows 10 64-bit Port dan Konektivitas: Port USB (termasuk

USB 3.0 dan USB-C), HDMI, Audio Jack, Wi-Fi, Bluetooth

Baterai: Baterai Lithium-ion dengan daya tahan yang bervariasi tergantung pada penggunaan.

Tabel 2.3. Komponen Spesifikasi Hardware yang di gunakan

Komponen	Spesifikasi
Prosesor	AMD Ryzen 3 (misalnya Ryzen 3 3200U atau Ryzen 3 3250U)
RAM	4GB atau 8GB DDR4 (Beberapa model mungkin memiliki opsi untuk meningkatkan RAM)
Sistem Operasi	Windows 10 64-bit Port dan Konektivitas: Port USB (termasuk USB 3.0 dan USB-C), HDMI, Audio Jack, Wi-Fi, Bluetooth





TEKNOLOGI INFORMASI
UNIVERSITAS DARMA PERSADA