

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

2.1.1 Sistem

Menurut Kristanto (2018:3) “Sistem dapat diartikan sebagai sekumpulan komponen (elemen atau unsur) yang saling berkaitan hingga (dapat) mempengaruhi (sesuai dengan fungsi masing-masing) satu sama lainnya dengan urutan/prosedur tertentu dalam mencapai suatu tujuan atau menghasilkan suatu output”.

Pengertian sistem menurut Meilinda (2018), Sistem merupakan kumpulan himpunan atau komponen dari unsur yang saling bergantung dan berinteraksi untuk sebuah tujuan yang bersifat sama.

Berdasarkan pengertian di atas dapat disimpulkan bahwa sistem adalah kumpulan dari komponen-komponen yang saling berkaitan satu dengan yang lain untuk mencapai tujuan dalam melaksanakan suatu kegiatan pokok perusahaan.

2.1.2 Informasi

Pengertian menurut Romney dan Steinbart (2015:4), Informasi adalah data yang telah di kelola dan di proses pengambilan keputusan. Sebagaimana perannya, penggunaan pembuatan keputusan yang lebih baik sebagaimana kuantitas dan kualitas dari peningkatan informasi.

Pengertian menurut Sutabri (2016) dalam buku Sistem Informasi Manajemen, Informasi adalah data yang telah diklasifikasi atau diolah atau interpretasi untuk digunakan dalam proses pengambilan keputusan.

Berdasarkan pengertian di atas, dapat disimpulkan bahwa konsep sistem informasi adalah data yang diolah untuk mengambil keputusan yang berguna bagi penggunanya.

2.1.3 Sistem Informasi

Menurut Muslihudin dan Oktafianto (2016:10) “Sistem informasi adalah hubungan antara data dan informasi, dan merupakan entitas penting yang merupakan sistem informasi”.

Berdasarkan pengertian di atas, dapat disimpulkan bahwa sistem informasi adalah suatu kegiatan atau kegiatan dari suatu sistem informasi yang dibentuk oleh serangkaian proses yang berupa komponen-komponen.

2.2 Sistem Parkir

Dalam undang-undang nomor 22 tahun 2009 tentang Lalu Lintas dan Angkutan Jalan, Bab I Ketentuan Umum, pada Pasal 1 angka 15 dan 16 tertulis sebagai berikut : Pasal 1 Dalam Undang-Undang ini yang dimaksud dengan: 15. Parkir adalah keadaan Kendaraan berhenti atau tidak bergerak untuk beberapa saat dan ditinggalkan pengemudinya; 16. Berhenti adalah keadaan Kendaraan tidak bergerak untuk sementara dan tidak ditinggalkan pengemudinya. Menurut Martadipura 2013 dalam Arief Budiman & Joko Triono (2016), terdapat dua jenis yaitu Parkir yang berada di badan jalan (on street parking) dan parkir yang berada luar badan jalan (off street parking). Pada On Street parking membutuhkan badan jalan untuk digunakan sebagai tempat parkir, contoh adalah parkir di depan pertokoan dimana kendaraan berhenti di pinggir jalan. Sedangkan untuk off street parking tidak membutuhkan badan jalan untuk sebagai tempat parkir melainkan dipersiapkan sebuah lahan khusus untuk menampung kendaraan parkir [10]. Dengan demikian dapat dikatakan bahwa parkir, suatu keadaan dimana suatu kendaraan yang bersifat sementara ditinggalkan oleh pengemudinya.

2.3 Retribusi Parkir

Dalam Peraturan Daerah Nomor 2 Tahun 2016 tentang Retribusi Pelayanan Parkir di Tepi Jalan Umum dan Tempat Parkir Khusus pada pasal 3 ayat 1, 2 dan 3 : menyebutkan bahwa (1) Objek Retribusi pelayanan parkir di tepi jalan umum adalah penyediaan pelayanan parkir di tepi jalan umum yang ditentukan oleh pemerintah Daerah sesuai dengan ketentuan peraturan Perundang-undangan; bahwa (2), Objek Retribusi khusus tempat parkir adalah pelayanan tempat khusus parkir yang telah disediakan, dimiliki dan/atau dikelola pemerintah Daerah; bahwa (3) Dikecualikan dari Objek Retribusi Tempat Khusus Parkir adalah pelayanan tempat parkir yang disediakan, dimiliki dan /atau dikelola oleh pemerintahan, BUMN, BUMD dan pihak swasta. Selanjutnya tarif retribusi parkir diatur pada pasal 11 ayat (1) menyatakan bahwa besarnya tariff retribusi parkir di tepi jalan umum untuk satu kali parkir ditetapkan (a) Kendaraan

tenpelan/gandingan Rp.10.000 / perkendaraan; (b) Kendaraan truck ukuran berat Rp. 8.000 /kendaraan; (c) Kendaraan truck mini dan sejenisnya Rp. 4.000 / kendaraan; (d) Kendaraan mobil sedan, mini bus, pick up dan kendaraan lainnya sejenisnya Rp. 3.000 / kendaraan; (e) Kendaraan Bajaj, Kaisar, Tossa, roda tiga dan sejenisnya Rp. 2.000.- perkendaraan; (f) Kendaraan bermotor jenis sepeda motor Rp.2.000 / kendaraan. Kemudian pasal 14 ayat (1) Pemungutan Retribusi dilaksanakan oleh petugas Dinas Perhubungan yang secara teknis dilaksanakan oleh UPTD (Unit Pelaksana Teknis Daerah). Ia berdasarkan potensi parkir masing-masing titik parkir.

2.4 Metode Prototype

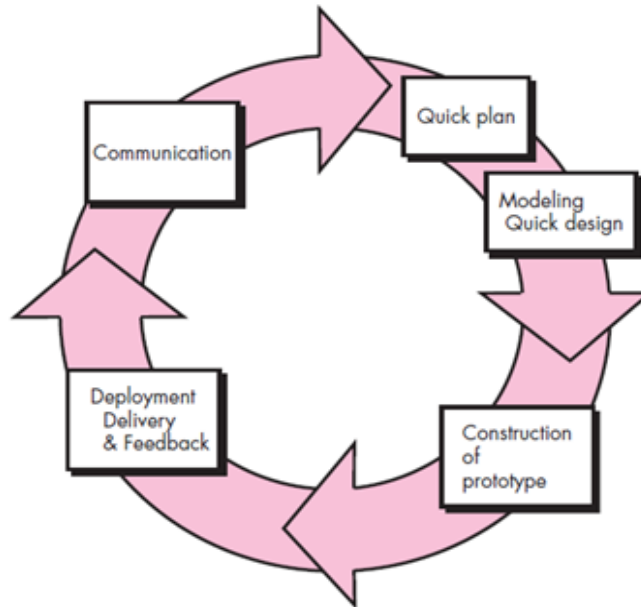
Prototype adalah proses pembuatan model sederhana yang memungkinkan pengguna memiliki gambaran tentang program serta melakukan pengujian awal. *Prototype* memberikan fasilitas bagi pengembang dan pengguna untuk saling berinteraksi selama proses pembuatan, sehingga pengembang dapat lebih mudah memodelkan perangkat lunak yang akan di buat. Metode ini cocok digunakan untuk mengembangkan sebuah perangkat lunak yang dikembangkan kembali. Metode ini dimulai dengan pengumpulan kebutuhan pengguna.

Kemudian membuat sebuah rancangan kilat yang selanjutnya akan dievaluasi kembali sebelum di produksi secara benar *Prototype*. Segala perubahan dapat terjadi pada saat *prototype* dibuat untuk memenuhi kebutuhan pengguna dan saat yang sama memungkinkan pengembangan untuk lebih memahami kebutuhan pengguna secara baik. (Pressman, 2012)

Tahapan-tahapan dalam metode *Prototype* :

1. Komunikasi (*Communication*) : pengumpulan data awal, yaitu komunikasi dengan klien dan *user* untuk menentukan kebutuhan.
2. Perencanaan Cepat (*Quick Plan*) : pembuaan perencanaan analisis terhadap kebutuhan pengguna.
3. Pemodelan Perancangan Cepat (*Modeling Quick Design*) : membuat rancangan desain program.
4. Pembentukan *Prototype* (*Construction of prototype*) : pembuatan aplikasi berdasarkan dari pemodelan desain yang telah dibuat.

5. Penyerahan Sistem dan Umpan Balik (*Development Delivery and Feedback*) : memproduksi perangkat ssecara benar sehinga dapat digunakan oleh pengguna.



Gambar 2.1 Diagram Prototype

Pada Gambar 2.1, Tahap pertama ialah *communication* dan pengumpulan data awal yaitu tahap suatu perencanaan yang dilakukan, mulai dari menciptakan dan melaksanakan proses untuk memastikan bahwa perencanaan tersebut berkualitas tinggi, terpercaya, efisiensi biaya. Tahap kedua adalah *quick plan* yaitu analisis terhadap kebutuhan pengguna. Tahap ketiga adalah *modelling quick design* yaitu pembuatan desain secara umum untuk selanjutnya dikembangkan kembali. Tahap keempat adalah *construction of prototype* adalah pembuatan perangkat *prototype* termasuk pengujian dan penyempurnaan. Tahap kelima adalah *deployment, delivery, and feedback* adalah tahap penyerahan sistem ke pengguna dan umpan balik.

2.5 Unified Modeling Language







Menurut (Rosa dan Shalahuddin 2018), “*Unified Modeling Language (UML)* adalah salah satu standar bahasa yang banyak digunakan dalam industri untuk mendefinisikan persyaratan, melakukan analisis dan desain, dan menggambarkan arsitektur dalam pemrograman berorientasi objek. UML adalah bahasa visualisasi

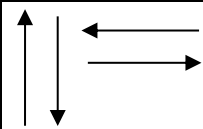
untuk pemodelan dan sistem komunikasi menggunakan diagram dan teks pendukung. UML hanya untuk pemodelan. Jenis-jenis Unified Modeling Language:

2.5.1 Flowchart

Flowchart atau bagan alur adalah diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu program. Setiap langkah digambarkan dalam bentuk diagram dan dihubungkan dengan garis atau arah panah. Flowchart berperan penting dalam memutuskan sebuah langkah atau fungsionalitas dari sebuah proyek pembuatan program yang melibatkan banyak orang sekaligus.

Tabel 2.1. Simbol-simbol Flowchart

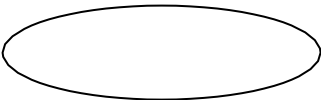
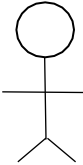
| No | Simbol | Nama | Fungsi |
|----|-------------------------------------------------------------------------------------|------------------|----------------------------------------------------------------------------------|
| 1. |  | Terminal | Digunakan untuk memulai atau mengakhiri program. |
| 2. |  | Input/Output | Digunakan untuk menyatakan input atau output tanpa melihat jenisnya. |
| 3. |  | Manual Operation | Digunakan untuk menunjukkan pengolahan yang tidak dilakukan oleh komputer. |
| 4. |  | Decision | Digunakan untuk memilih proses yang akan dilakukan berdasarkan kondisi tertentu. |
| 5. |  | Processing | Digunakan untuk menunjukkan pengolahan data yang dilakukan oleh komputer. |
| 6. |  | Disk Storage | Digunakan untuk menyatakan masukan dan keluaran yang berasal dari disk. |


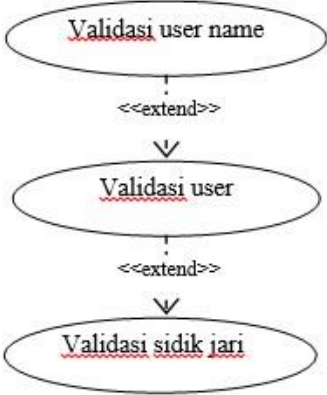

| | | | |
|----|-----------------------------------------------------------------------------------|------------------------------------------|---------------------------------------------------------------------------------------------------|
| 7. |  | Flow Direction Symbol/Connecting line | Berfungsi untuk menghubungkan simbol yang satu dengan yang lainnya, menyatakan arus suatu proses. |
|----|-----------------------------------------------------------------------------------|------------------------------------------|---------------------------------------------------------------------------------------------------|

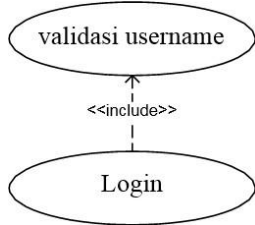
2.5.2 Use case Model

Sukanto dan Shalahuddin (2018:155), “*Use case* atau diagram *Use case* adalah suatu pemodelan perilaku (*behavior*) dari suatu sistem informasi yang akan dibuat. Sebuah *Use case* menggambarkan interaksi antara satu atau lebih aktor dan sistem informasi yang akan dibuat. Secara dasar, *Use case* digunakan untuk mengetahui fungsi apa saja yang ada dalam sistem informasi dan siapa yang memiliki akses ke fungsi tersebut”. Berikut adalah simbol-simbol yang ada pada diagram *Use case* :

Tabel 2.2. Simbol-simbol Use case Diagram

| No. | Simbol | Deskripsi |
|-----|-----------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | <i>Use case</i>  | Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor, biasanya dinyatakan dengan menggunakan kata kerja diawal frase nama <i>use case</i> . |
| 2. | <i>Aktor/actor</i>  | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang, biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. |

| | | |
|----|----------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 3. | Asosiasi/ <i>association</i>  | Komunikasi antara aktor dan <i>Use case</i> yang berpartisipasi pada <i>Use case</i> atau <i>Use case</i> memiliki interaksi dengan aktor. |
| 4. | Exstensi/ <i>extend</i> <<extend>> | <p>Relasi antara <i>Use case</i> tambahan dengan <i>use case</i>, bahkan jika tidak ada <i>Use case</i> tambahan, <i>Use case</i> yang ditambahkan dapat eksis secara independen, mirip dengan prinsip pewarisan dalam pemrograman berorientasi objek, biasanya <i>Use case</i> tambahan memiliki nama yang sama dengan yang ditambahkan kasus penggunaan, misalnya</p>  <p>Arah panah mengarah pada <i>Use case</i> yang ditambahkan, biasanya <i>Use case</i> yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p> |
| 5. | Generalisasi/ <i>generalization</i>  | Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua <i>buah Use case</i> dimana fungsi yang satu adalah fungsi yang |

| | | |
|----|------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | <p>lebih umum dari lainnya, misalnya :</p> <p>arah panah mengarah pada <i>Use case</i> yang menjadi generalisasinya (umum)</p> |
| 6. | <p>Menggunakan / <i>include</i> / <i>uses</i></p> <p><<include>></p> <p>—————→</p> <p><<uses>></p> <p>—————→</p> | <p>Relasi antara <i>Use case</i> tambahan ke <i>Use case</i> di mana <i>Use case</i> yang ditambahkan membutuhkan <i>Use case</i> ini untuk menjalankan fungsinya atau sebagai kondisi eksekusinya.:</p> <ul style="list-style-type: none"> - <i>Include</i> berarti <i>Use case</i> yang ditambahkan akan selalu di panggil saat <i>Use case</i> tambahan dijalankan, missal pada kasus berikut :  <pre> graph BT Login([Login]) -.-> <<include>> Validasi([validasi username]) </pre> <ul style="list-style-type: none"> - <i>Include</i> berarti <i>Use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>Use case</i> yang di tambahkan telah dijalankan sebelum <i>Use case</i> tambahan di jalankan, misal pada kasus berikut : Kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada |



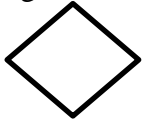


| | | |
|--|--|------------------------------------------------|
| | | pertimbangan dan interpretasi yang dibutuhkan. |
|--|--|------------------------------------------------|

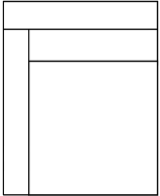
Sumber : Sukamto dan Shalahuddin (2018:156-158)

2.5.3 Activity Diagram

Sukamto dan Shalahuddin (2018:161), “*Activity* diagram atau diagram aktivitas menggambarkan alur kerja (*workflow*) atau aktivitas dari suatu sistem atau proses bisnis atau menu dalam perangkat lunak. Diagram aktivitas menggambarkan aktivitas sistem daripada apa yang dilakukan aktor, sehingga sistem dapat melakukan aktivitas”. Berikut adalah simbol- simbol yang ada pada diagram aktivitas

Tabel 2.3. Simbol-simbol Activity Diagram

| No. | Simbol | Deskripsi |
|-----|---------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| 1. | Status awal  | Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal. |
| 2. | Aktivitas  | Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja. |
| 3. | Percabangan/ <i>decision</i>  | Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu. |
| 4. | Penggabungan/ <i>join</i>  | Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu. |
| 5. | Status akhir  | Status akhir yang dilakukan oleh sistem, sebuah diagram aktivitas memiliki sebuah status akhir. |

| | | |
|----|-----------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|
| 6. | Swimlane  | Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi. |
|----|-----------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|



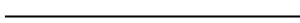
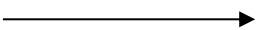
Sumber : Rosa dan Shalahuddin (2018:162-163)



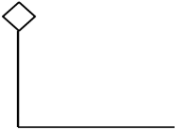
2.5.4 Class Diagram

Sukanto dan Shalahuddin (2018:141), “diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem”. Kelas memiliki apa yang disebut atribut dan *method* atau operasi. Berikut penjelasan atribut dan *method* :

1. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.
2. Operasi atau *method* adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

Tabel 2.4. Simbol-simbol Class Diagram

| No. | Simbol | Deskripsi |
|-----|--------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | Kelas  | Kelas pada struktur system |
| 2. | Halaman/Halaman  | Sama dengan konsep Halaman dalam pemrograman berorientasi objek |
| 3. | Asosiasi/ <i>association</i>  | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> |
| 4. | Asosiasi berarah/ <i>directed association</i>  | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> |

| | | |
|----|------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| 5. | Generalisasi  | Relasi antar kelas dengan makna generalisasi – spesialisasi (umum - khusus) |
| 6. | Kebergantungan/ <i>dependensi</i>  | Relasi antar kelas dengan makna kebergantungan antar kelas |
| 7. | Agregasi/ <i>aggregation</i>  | Relasi antar kelas dengan makna semua-bagian (<i>whole-part</i>) |

Sumber : Sukamto dan Shalahuddin (2018:146-147)

2.5.5 Kamus Data

Sukamto, Ariani Rosa, dan M. Shalahuddin (2018), mengemukakan bahwa kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada *DFD*. Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) data keluaran (*output*) dapat dipahami secara umum.

Tabel 2.5. Simbol-simbol Kamus Data

| No | Simbol | Keterangan |
|----|------------------|-------------------------------------------|
| 1. | = | Disusun atau terdiri dari |
| 2. | + | Dan |
| 3. | [] | Baik...atau... |
| 4. | { } ⁿ | ⁿ kali diulang/bernilai banyak |
| 5. | () | Data opsional |
| 6. | *...* | Batas komentar |

2.6 HyperText Preprocessor (PHP)

Menurut Sidik (2017:4), "PHP secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly*

yang dieksekusi di server web, dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML, dikenal juga sebagai bahasa pemrograman *server side*."

Menurut Harianto,dkk (2019:13), "PHP merupakan software Open-Source yang disebar dan dilisensikan secara gratis serta dapat didownload secara bebas dari situs resminya".

Berdasarkan pengertian di atas, bahwa dapat disimpulkan PHP adalah bahasa pemrograman berupa data yang dikelola melalui sebuah sistem untuk input, output untuk membuat website.

2.7 XAMPP

XAMPP adalah perangkat lunak gratis, yang mendukung beberapa sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya sebagai server yang berdiri sendiri (localhost), yang terdiri dari program Apache HTTP Server, database MySQL, dan penerjemah bahasa yang ditulis dalam pemrograman PHP dan Perl. Nama XAMP adalah singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU General Public License dan gratis, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis (Mearaj et al., 2019).

2.8 Database

Menurut Setiawan (2017), Basis data atau database adalah kumpulan tabel yang berisi sejumlah besar data, serta kumpulan baris (field) dan kolom (columns). Disini kita akan menyimpan data dari project yang kita buat. Pamungkas (2017) basis Data ialah suatu kumpulan data yang saling terhubung dan disimpan secara bersamaan pada sebuah media, yang diorganisasikan berdasarkan suatu skema atau struktur tertentu dengan software agar bias digunakan untuk manipulasi kegunaan tertentu.

2.9 MySQL (My Structure Query Language)

MySQL (My Structured Query Language) atau yang biasa dibaca mai-se-kuel adalah sebuah program pembuat dan pengelola database atau yang sering disebut

dengan DBMS (DataBase Management System), sifat dari DBMS ini adalah Open Source. MySQL sebenarnya produk yang berjalan pada platform Linux, dengan adanya perkembangan dan banyaknya pengguna, serta lisensi dari database ini adalah Open Source, maka para pengembang kemudian merilis versi Windows. Selain itu MySQL juga merupakan program pengakses database yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi Multi User (Banyak Pengguna).

Kelebihan lain dari MySQL adalah menggunakan bahasa query (permintaan) standard SQL (Structured Query Language). Sebagai sebuah program penghasil database, MySQL tidak mungkin berjalan sendiri tanpa adanya sebuah aplikasi pengguna (interface) yang berguna sebagai program aplikasi pengakses database yang dihasilkan. MySQL dapat didukung oleh hampir semua program aplikasi baik yang Open Source seperti PHP maupun yang tidak Open Source yang ada pada platform windows seperti Visual Basic, Delphi dan lainnya.(Fajar Hariadi, 2013).

2.10 HTML

Sulistiono (2018), HTML (Hypertext Markup Language) adalah bahasa markup yang digunakan untuk membuat suatu halaman website, menampilkan berbagai macam informasi seperti gambar, teks, suara, dan video yang ada pada web internet, ditulis dalam sebuah berkas format ASCII supaya dapat menghasilkan sebuah tampilan wujud yang terintegrasi.

2.11 JavaScript

Sulistiono (2018), JavaScript merupakan suatu bahasa yang digunakan untuk membuat sebuah program yang agar dokumen HTML yang ditampilkan dalam browser agar menjadi lebih interaktif dan tidak sekedar indah saja. JavaScript juga memberikan beberapa fungsionalitas ke suatu halaman web.

2.12 Visual Studio Code

Visual Studio Code (VS Code) adalah sebuah teks *editor* yang ringan dan handal dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*. Teks *editor* ini secara

langsung mendukung bahasa pemrograman *JavaScript*, *Typescript*, dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang *via marketplace* Visual Studio Code (seperti *C++*, *C#*, *Python*, *Go*, *Java*, dst). (Permana & Romadhon, 2019).

2.13 Penelitian Terkait

Tabel 2.6. Penelitian Terkait

| No | Judul | Peneliti Sebelumnya |
|----|-------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1. | Sistem Informasi Peramalan Pendapatan Retribusi Parkir Dinas Perhubungan Kota Palembang | peramalan pendapatan retribusi parkir ini yaitu untuk membantu UPTD perparkiran Dinas Perhubungan kota Palembang untuk mengoptimalkan pendapatan asli daerahnya. Sistem yang dibangun dapat membantu dalam mengelola perizinan parkir, mengelola transaksi retribusi parkir dan dapat melakukan peramalan pendapatan retribusi parkir pada tahun selanjutnya. Data retribusi parkir akan dijadikan titik awal dalam peramalan pendapatan pada tahun berikutnya. Statistika akan dilakukan untuk menganalisa peramalan pendapatan retribusi dengan melakukan analisa menggunakan data berkala dengan metode kuadrat terkecil. Tidak akan terjadi redundansi kegiatan saat pengelolaan retribusi parkir dibantu dengan sebuah system, mengetahui lokasi mana yang belum membayar retribusi parkir, serta dapat meramalkan pendapatan retribusi parkir selanjutnya |
| 2. | Perancangan Sistem Informasi Parkir Pada Dinas Perhubungan Komunikasi Dan Informatika Kabupaten | Sasaran pembangunan daerah kabupaten Pacitan melalui Dinas Perhubungan Komunikasi dan Informatika, diarahkan kepada upaya dalam penyelenggaraan pelayanan yaitu jasa transportasi, komunikasi dan informatika sehingga dapat |

| | | |
|----|-----------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | Pacitan | mewujudkan masyarakat yang demokratis dan sejahtera. Tujuan dari penelitian ini adalah merancang sistem yang dapat digunakan untuk mengelola data parkir yang ada di Dinas Perhubungan dan Informatika pada kabupaten Pacitan. |
| 3. | Sistem Informasi Pengelolaan Tenaga Parkir Dan Retribusi Parkir Pada Dinas Perhubungan Komunikasi Dan Informatika Kabupaten Kudus | Menghasilkan aplikasi perangkat lunak untuk memudahkan proses pengolahan dan pencarian data agar lebih cepat sebelum dibangun aplikasi, disamping itu untuk memudahkan dalam mengelola laporan pendapatan parkir. Hasil dari rancang bangun ini adalah sebuah aplikasi berbasis desktop |