**BAB II**

**LANDASAN TEORI**

* 1. **Studi Literatur**

Jurnal karya Priyadi dan Dwi Repa yang berjudul pada Universitas Teknokrat Indonesia pada tahun 2018[1] menjelaskan latar belakang kenapa dibuatnya aplikasi *marketplace* jasa *printing*. beberapa perusahan jasa percetakan masih dilakukan secara konvensional, salah satunya Opick Percetakan, Penjualan masih dilakukan dengan cara pencatatan kedalam buku pemesanan, sehingga sering terjadi kesalahan dalam pesanan seperti penulisan pesanan yang salah, jadi tidak sesuai dengan pesanan pelanggan, tidak adanya informasi jika terjadi kesamaan data, seperti penulisan pesanan lebih dari satu kali atau pemesanan oleh orang yang sama, tidak ada informasi jika sudah dilakukan pencatatan pemesanan. Sehingga karyawan sering mengalami kesalahan dalam penulisan pemesanan, sering terjadi kehilangan berkas, seperti nota transaksi yang hilang atau tercecer karena lupa menaruhnya, dan pelanggan harus datang langsung ke tempat jasa percetakan sehingga membuang waktu untuk melakukan pemesanan. Hal tersebut membuat peneliti membuat sebuah aplikasi *marketplace* difokuskan untuk menyebarkan informasi jasa percetakan dengan menghadirkan layanan, fasilitas, menu serta memberikan kenyaman bagi pelanggan dalam mencari jasa percetakan yang ditawarkan.

Menurut jurnal Perancangan Aplikasi *Marketplace* Jasa Percetakan Berbasis *Website* oleh Rahmat Fauzi, Satrio Wibowo, Dela Youlina Putri pada tahun 2018[2], banyaknya permintaan mencetak dokumen yang tinggi di lingkungan kampus akan tetapi belum banyak yang menyediakan layanan antar langsung dari jasa printing tersebut atau jasa pemesanan secara *real time.* Sehingga dibuatlah aplikasi *marketplace* jasa percetakan yang dapat memberikan layanan pencarian beberapa percetakan, melihat perbandingan harga antar percetakan, efisiensi waktu tanpa harus mengantri.

Jurnal yang berjudul Perancangan Aplikasi *Marketplace Order* Baju Pintar menggunakan *Web Responsif* untuk Memudahkan *Customer* Mendesain Sesuai Selera oleh Ali Kumaidi pada tahun 2018[3] menjelaskan bahwa Penelitian ini memaparkan proses perancangan aplikasi *marketplace* yang memudahkan customer dalam memilih desain yang sesuai yang telah di sediakan oleh butik dan penjahit, memilih bahan yang disediakan oleh toko bahan serta menggunakan jasa penjahit yang sesuai dengan pilihan.Metode pengembangan menggunakan GRAPPLE (Guidelines for Rappid Application Enginering) dengan tujuan dapat menghasilkan sistem berorientasi objek dalam waktu yang singkat tanpa mengurangi kualitas sistem yang dibangun. Serta menggunakan bahasa pemrograman PHP dan basis data MySQL. Adapun pengembangan menggunakan web responsive dengan *bootstrap framework*. Untuk tahap analisa dan desain diagram menggunakan diagram UML.

Pada jurnal yang berjudul Perancangan Sistem Informasi *E-Marketplace* Bank Sampah Berbasis Web oleh Irma Kurnia Juliany, Muhammad Salamuddin, dan Yuni Kristina Dewi tahun 2018[4] menjelaskan saat ini sampah kering yang menjadi bahan baku utama kerajinan daur ulang bisa didapatkan oleh pengepul dari masyarakat yang menyetor sampah pada bank sampah. Namun, para pengepul harus mendatangi bank sampah untuk mendapatkan rincian stok dari sampah yang ingin dibelinya. Hal tersebut menyebabkan peneliti ingin membuat Perancangan Sistem Informasi *E*-*Marketplace* Bank Sampah Berbasis *Web* menggunakan *Unified Modeling Language* (UML) sebagai dasar permodelannya.untuk memudahkan proses transaksi antara bank sampah dan pengepul mulai dari pencarian barang, tawar-menawar harga hingga transaksi jual-beli. Sistem juga dapat menampilkan laporan persediaan sampah di masing- masing bank sampah juga statistik transaksi antara bank sampah dan pengepul sehingga kedua belah pihak dapat terbantu dan bank sampah dapat terus beroperasi secara maksimal untuk mengontrol penimbunan sampah yang terus bertambah.

Jurnal yang berjudul Rancang Bangun Sistem *E-Commerce Marketplace* pada Usaha Mikro Kecil dan Menengah Berbasis *Website* oleh Andre Juniar Bahruly pada tahun 2019 [5] menjelaskan bahwa permasalahan pada UMKM yang terjadi saat ini yaitu kurang sesuainya data barang yang telah dijual dengan data barang stok yang berada di Gudang. Konsumen yang akan melakukan pembelian pun sering di buat kecewa dikarenakan tidak mendapatkan informasi jumlah stok yang tersedia secara *real time*. Hal tersebuat membuat peneliti membuat sebuah sistem *marketplace* pada usaha mikro kecil dan menengah yang berissi tentang proses pendaftaran UMKM, proses pemesanan, penjualan, akses produk, proses update persediaan barang oleh vendor dan proses pengiriman

* 1. **Aplikasi Marketplace Jasa Percetakan**

Secara istilah pengertian aplikasi adalah suatu program yang siap untuk digunakan yang di buat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang dituju. Menurut kamus komputer eksekutif, aplikasi memiliki arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang siasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan. Aplikasi adalah kumpulan perintah program yang dibuat untuk melakukan pekerjaan pekerjaan tertentu[6]. Aplikasi adalah program yang dibuat oleh pemakai yang diajukan untuk melakukan tugas khusus.

*Marketplace* adalah aplikasi atau situs *web* yang memberi fasilitas jual beli *online* dari berbagai sumber. *Marketplace* memiliki fungsi yang sama dengan pasar tradisional, perbedaannya yaitu *marketplace* lebih terkomputerisasi dengan menggunakan bantuan sebuah jaringan dalam mendukung sebuah pasar agar dapat dilakukan secara efisien dalam menyediakan *update* informasi dan layanan jasa untuk penjual dan pembeli yang berbeda-beda[7].

Terdapat dua jenis *marketplace* yaitu:

1. Marketplace Murni

Marketplace murni merupakan situs marketlace yang hanya menyediakan lapak untuk berjualan dan fasilitas pembayaran. Penjual yang melakukan kerja sama akan diberikan keleluasan lebih banyak. Penjual harus menyediakan deskripsi dan foto produk secara mandiri. Penjual juga dapat menerima penawaran harga dari pembeli. Sebelum melakukan pembayaran, pembeli dapat melakukan penawaran harga kepada penjual. Setelah mendapatkan harga yang disepakati oleh kedua belah pihak, pembeli bisa mengirimkan sejumlah uang ke rekening yang disediakan *marketplace*.

1. Marketplace Konsinyasi

Merupakan jenis kerja sama *marketplace* titip barang. Dimana penjual hanya perlu menyediakan produk serta detail informasi produk ke pihak *marketplace.* sedangkan untuk foto produk, Gudang, dan pengiriman barang, serta fasilitas pembayaran akan disediakan oleh pihak *marketplace*.

*Marketplace* memiliki keuntungan baik bagi penjual maupun pembeli, yaitu [8]:

* + - 1. Manfaat pembeli

1. Memfasilitasi pencarian dan perbandingan produk dan informasi dalam hal kualitas atau harga sesuai dengan keadaan berbagai pemasok
2. Pembeli mendapatkan harga kompetitif antar pemasok secara global
3. Mengurangi biaya pengadaan atau biaya pembelian.
   * + 1. Keuntungan penjual
4. *E-Marketplace* mempermudah pencarian pembeli baru
5. Perluas penjualan dalam menjangkau pasar
6. Media promosi produk tanpa biaya promosi, transaksi, dan biaya penjualan yang rendah
7. Dapat digunakan oleh masyarakat umum
8. Memfasilitasi dalam memantau dan menganalisis permintaan-permintaan pasar serta tren pembeli.

Percetakan adalah sebuah proses industri untuk memproduksi massal tulisan dan gambar yang berada diatas sebuah media seperti kertas, kain, kayu dan sebagainya. Teknik percetakan umum lainnya meliputi sablon, cetak *relief*, *rotogruve,* dan percetakan berbasis digital seperti pita jarum, *inkjet*, dan laser.

Digital printing adalah proses cetak gambar yang sudah didesain menuju ke material atau media fisik. Usaha jasa percetakan sekarang dapat melakukan Teknik percetakan secara professional dengan menggunakan printer laser atau dengan inkjet yang dapat menghemat waktu dan lebih terjangkau [9].

Percetakan merupakan sebuah komunikasi massa yang dapat digunakan sejak tahun 1400 hingga awal 1900. Percetakan dalam masa sekarang dapat dikategorikan sebagai industri penting diberbagai negara, khususnya negara maju.

* 1. **Alat Bantu Perancangan Sistem**
     1. **UML (*Unified Modelling Language*)**

*Unified Modelling Language* (UML) adalah sebuah bahasa pemodelan standar dalam industri untuk visualisasi, merancang, dan dokumentasikan sistem piranti lunak. Pemodelan menggunakan UML merupakan metode pemodelan berorientasi objek dan berbasis visual. Karenanya, pemodelan UML merupakan pemodelan yang fokus pada pendefinisian struktur statis dan model sistem informasi yang dinamis daripada mendefinisikan data dan model proses.

UML menyediakan beberapa notasi dan artifak standar yang dapat digunakan sebagai alat komunikasi bagi para pelaku dalam proses analisis dan desain. Artifak dalam UML di definisikan sebagai informasi dalam proses pengembangan perangkat lunak. Dalam proses analisis dan desain, terdapat tiga artifak yang penting yaitu *Use Case Diagram, Sequence diagram, dan Class Diagram*. *Use Case Diagram* merupakan artifak dari proses analisa, sedangkan *Sequence* *Diagram* dan *Class Diagram* merupakan artifak dari proses desain [10].

1. Use Case Diagram (Diagram Use Case)

*Use Case Diagram* adalah pemodelan untuk kelakuan (behavior) sistem informasi yang dibuat dengan kata lain use case menjelaskan apa yang dilakukan oleh sistem yang akan dibangun dan siapa yang berinteraksi dengan sistem [11]. Symbol-simbol pada diagram *use case* diagram terdapat pada table 2.1.

**Tabel 2.1 Use Case diagram**

|  |  |
| --- | --- |
| Simbol | Keterangan |
| Use case | Fungsional yang disediakan sistem untuk menjelaskan urutan kegiatan yang dilakukan actor dan sistem untuk mencapai suatu tujuan tertentu, biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama use case. |
| Assosiasi atau *association* | interaksi antara aktor dan use case yang berpartisipasi pada use case atau use case yang memiliki interaksi dengan actor pada suatu sistem. |
| Include | Yaitu kelakuan yang harus terpenuhi agar sebuah *event* dapat terjadi, dimana pada kondisi ini sebuah *use case* adalah bagian dari *use case* lainnya |
| Extend | kelakuan yang hanya berjalan di bawah kondisi tertentu. Arah panah relasi pada use case mengarah pada use case yang lebih besar kontrolnya atau yang dipakai. |

1. Activity Diagram

*Activity diagram* sangat mirip dengan flowchart karena kita dapat memodelkan sebuah alur kerja dari satu aktivitas ke aktivitas lainnya atau dari satu aktivitas ke dalam keadaan sesaat. Diagram ini memodelkan alur kerja sebuah proses bisnis dan urutan aktivitas dalam suatu proses [12]. Simbol pada diagram *activity* diagram terdapat pada table 2.2.

**Tabel 2.2 Activity diagram**

|  |  |
| --- | --- |
| Simbol | Keterangan |
| *Initial node* | Mempresentasikan titik mulai aktivitas |
| Node aksi | Mempresentasikan suatu tugas atau aksi yang dilakukan oleh sistem perangkat lunak |
| *Swimlanes* | Menggambarkan objek mana yang bertanggungjawab untuk aktivitas tertentu. |
| *Fork* | Menggambarkan aktivitas yang dilakukan secara parallel |
| *Join* | 1. Menggambarkan kegiatan yang digabungkzan |
| Aliran control  (*Control Flow*) | Mempresentasikan adanya aliran control. Adanya Aliran proses suatu aktivitas |
| *Final node* | Mempresentasikan akhir dari aktivitas |

1. Sequence Diagram

*Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah scenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem [12]. Simbol-simbol pada diagram *sequence* terdapat pada table 2.3.

**Tabel 2.3 Sequence diagram**

|  |  |
| --- | --- |
| Simbol | Keterangan |
| Actor | Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi. Biasanya dinyatakan dengan kata benda di awal frase nama actor. |
| Garis Hidup atau lifeline | Menyatakan kehidupan suatu objek. |
| Waktu aktif | Menyatakan objek dalam keadaan aktif dan berinteraksi dengan pesan. atau menggambarkan tempat berakhirnya atau mulainya suatu pesan |
| Pesan (message *call*) | Menyatakan pengiriman pesan |
| Pesan (message *return*) | Menyatakan hasil atau pesan yang diterima dari pengiriman pesan |
| Boundary Class | Boundary terletak di antara system dengan dunia sekelilingnya. Semua form, laporan, antar muka ke perangkat atau pun ke sistem lainnya termasuk dalam kategori. |
| Control Class | Menggambarkan “perilaku mengatur”, mengkoordinasikan perilaku sistem dan dinamika dari suatu sistem, menangani tugas utama dan mengontrol alur kerja suatu system atau penanganan dari kesalahan. |
| Entity Class | Menggambarkan informasi yang harus disimpan oleh system (struktur data dari sebuah sistem). Dan bisa merupakan tabel dari sebuah struktur data dari databases. |

1. Class Diagram

*Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram ini menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaiman mereka saling berinteraksi untuk mencapai suatu tujuan [12]. Simbol pada diagram *class* terdapat pada tabel 2.4.

**Tabel 2.4 Class diagram**

|  |  |
| --- | --- |
| Simbol | Deskripsi |
| Kelas | Kelas pada struktur sistem |
| Antar muka / *interface* | Sama dengan konsep *interface* dalam pemrograman berorientasi objek. |
| Asosiasi / *association* | Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan *multiplicity*. |
| Asosiasi berarah / *directed association* | Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan *multiplicity.* |
| Generalisasi | Relasi antar kelas dengan makna generalisasi-spesifikasi (umum khusus) |
| Kebergantungan / *dependency* | Kebergantungan antar kelas |
| Agregasi / *aggregation* | Relasi antar kelas dengan makna semua bagian (*whole-part*). |

* + 1. **Entity Relationship Diagram ERD**

ERD adalah model data dengan diagram hubungan entitas yang terdiri atas kumpulan objek dasar yang disebut entitas dan hubungan objek-objek yang tersebut dengan menggunakan perangkat konseptual dengan menggunakan diagram. Setiap membuat ERD, yang ditentukan terlebih dahulu adalah derajat yaitu hubungan secara kompleks pada hubungan data. Entitas merupakan objek yang dibedakan dari objek yang lain oleh himpunan dari atribut. Adapun simbol-simbol yang dapat digunakan ERD dapat dilihat pada table 2.5.

**Tabel 2.5 Simbol Entity Relationship Diagram**

|  |  |  |
| --- | --- | --- |
| **Simbol** | **Nama** | **Keterangan** |
|  | Relasi | Menunjukan adanya hubungan diantara sejumlah entitas yang berbeda |
|  | Atribut | Mendeskripsikan karakter entitas (atribut yang berfungsi sebagai *key* diberi garis bawah) |
|  | Entitas | Suatu objek yang dapat diidentifikasi dalam lingkungan pemakai |
|  | Garis | Penghubung antar relasi dengan entitas, relasi, dan entitas dengan atribut. |

* 1. **Pengembangan Produk**
     1. **Framework Laravel**

Framework adalah sekumpulan script yang dapat membantu *developer/programmer* dalam menangani berbagai masalah dalam pemrograman, seperti koneksi ke *database*, pemanggilan variable, file, dan lain-lain sehingga pekerjaan developer lebih focus dan lebih cepat dalam membangun aplikasi [13].

Laravel adalah Framework pengembangan web MVC yang di tulis dalam bahasa PHP. Framework laravel dibuat oleh Taylor Otwel. Framework ini juga mengadopsi konsep Model-View-Controller, framework ini juga menyediakan librari-librari umum yang biasa diperlukan pada saat pengerjaan aplikasi. Laravel berada dibawah lisensi MIT [13].

Beberapa fitur yang dimiliki framework Laravel yaitu:

1. *Bundles* adalah sebuah fitur dengan sistem pengemasan modular dan berbagai bundle telah tersedia untuk digunakan dalam aplikasi.
2. *Eloquement* ORM yaitu penerapan PHP lanjutan dari pola “*active record*”, menyediakan metode internal untuk mengatasi kendaala hubungan antara obyek database.
3. *Restful controllers*, memberikan sebuah option pilihan untuk memisahkan logika dalam melayani HTTP GET dan permintaan POST.
4. *Reverse routing* merupakan hubungan antara link dan *Route*, sehingga saat ada perubahan pada route secara otomatis akan tersambung dengan link yang relevan. Ketika link yang dibuat dengan menggunakan nama-nama dari Route yang ada secara otomatis Laravel akan membuat URl yang sesuai.
5. *Application Logic* merupakan bagian dari aplikasi yang dikembangkan, baik menggunakan *Controllers* atau sebagai bagian dari deklarasi *Route*.
   * 1. **Database**

Basis data atau *Database* adalah kumpulan data yang disimpan secara sistematis didalam komputer dan dapat diolah atau di manipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Basis data meliputi spesifikasi berupa tipe data, struktur, dan batasan-batasan data yang akan disimpan. Proses mengambil dan memasukkan data dari dan ke media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (*database management system*). DBMS merupakan sistem perangkat lunak yang memungkinkan *user* untuk memelihara, mengontrol, mengakses data secara praktis dan efisien [14].

Bahasa basis data adalah suatu perantara bagi pemakai dengan basis data yang diatur dalam bahasa khusus yang ditetapkan oleh perusahaan. Berdasarkan fungsinya, bahasa basis data dibagi menjadi 3 komponen utama yaitu[15]:

* + - 1. *Data Definition Language* (DDL)

DDL adalah bagian dari SQL yang digunakan untuk mendefinisikan data dan objek *database*. Perintah yang tergolong DDL adalah *create, alter, dan drop*.

* + - 1. *Data Manipulation Language* (DML)

DML adalah bagian dari SQL yang dipergunakan untuk memanipulasi data dalam table/record dari table. Jenis perintah yang termasuk DML adalah *Select, Insert, Update,* dan *Delete.*

* + - 1. *Data Central Language* (DCL)

DCL adalah bagian dari SQL yang dipergunakan untuk mengontrol hak-hak akses dalam objek *database* SQL *server.* Jenis perintah ini adalah *grant dan revoke.*

* + 1. **MySQL**

MySQL merupakan salah satu perangkat lunak untuk sistem manajemen database SQL. MySQL diciptakan oleh David Axmark Allan Larson, dan Michael Widenius. MySQL merupakan perangkat database yang dapat digunakan untuk membuat aplikasi web dinamis seperti CMS. MySQL merupakan perangkat lunak gratis dibawah lisensi GNU General public License tetapi tidak cocok dengan penggunaan perangkat GPL[16].

MySQL sebenarnya merupakan turunan dari salah satu konsep utama dalam database, yaitu SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Beberapa keistimewaan MySQL, antara lain: portability, multiuser, security.

Beberapa kelebihan dari MySQL yaitu [17]:

1. Fleksibel

Mengembangkan aplikasi desktop maupun aplikasi web dengan menggunakan teknologi yang sangat bervariasi dapat menggunakan MySQL.

1. Performa tinggi

MySQL memiliki mesin query dengan performa mantap. Oleh karena itu, proses transaksional dapat melakukannya dengan sangat cepat.

1. Proteksi data yang handal

MySQL memiliki fitur mekanisme yang baik untuk menangani hal tersebut, yaitu dengan menyediakan fasilitas manajemen user, enkripsi data dan lain sebagainya.

* + 1. ***PHP* (*Hypertext Preprocessor*)**

*PHP (Hypertext Preprocessor),* yaitu bahasa pemrograman *web server-side* yang bersifat open source. PHP merupakan script yang terintegrasi dengan HTML dan berada pada server (*server side HTML embedded scripting*). PHP adalah script yang berarti halaman yang digunakan untuk membuat halaman *website* yang dinamis. Semua *script* PHP dieksekusi pada server dimana script tersebut dijalankan. *Web Server* yang mendukung PHP dapat ditemukan dimana-mana, mulai dari Apache, IIS, *Lighttpad* hingga Xitami dengan konfigurasi yang relatif mudah [18].

Keunggulan dari PHP adalah[19] :

1. PHP menggunakan bahasa skrip yang tidak melakukan sebuah kompilasi dalam penggunaannya.
2. PHP bekerja pada *web server* yang dibuat oleh Microsoft, seperti IIS atau PWS juga pada apache yang bersifat terbuka.
3. Perkembangan interpreter pada PHP jauh lebih cepat dan jauh lebih mudah, karena memiliki milis-milis dan pengembang yang dapat membantu pengembangannya.
4. PHP memiliki referensi yang begitu banyak sehingga sangat mudah untuk dipahami.
   * 1. **Webserver**

*Webserver* adalah perangkat lunak yang berfungsi sebagai penerima permintaan yang dikirimkan melalui *browser*, kemudian memberikan tanggapan permintaan dalam bentuk halaman situs web atau dalam dokumen HTML. Saat mengambil halaman *website, browser* mengirimkan HTTP *request* ke server yang kemudian diproses oleh *webserver* [13].

Pada webserver, HTTP *request* diproses dengan bantuan HTTP server. HTTP server merupakan perangkat lunak yang bertugas menerjemahkan URL (alamat situs web) dan HTTP (protocol yang digunakan *browser* untuk menampilkan halaman *website* [13].

**DAFTAR PUSTAKA**

[1] S. Hirsh *et al.*, “Aplikasi Marketplace Jasa Printing Berbasis Web (Studi kasus:Opick Percetakan),” *Africa Educ. Rev.*, vol. 15, no. 1, pp. 156–179, 2018, [Online].

[2] R. Fauzi, S. Wibowo, and D. Y. Putri, “Perancangan Aplikasi Berbasis Website,” *Fountain Informatics J.*, vol. 3, no. 1, p. 5, 2018.

[3] A. Khumaidi, “Perancangan Aplikasi Marketplace Order Baju Pintar Menggunakan Web Responsif Untuk Memudahkan Customer Mendesain Sesuai Selera,” *Ikhraith-Informatika*, vol. 2, no. 2, pp. 61–69, 2018, [Online].

[4] I. K. Juliany, M. Salamuddin, and Y. K. Dewi, “Perancangan Sistem Informasi E-Marketplace Bank Sampah Berbasis Web,” *Semin. Nas. Teknol. Inf. dan Multimed. 2018*, pp. 19–24, 2018.

[5] B. A. Juniar, “Rancang Bangun Sistem E-Commerce Marketplace pada Usaha Mikro Kecil dan Menengah Berbasis Website,” no. 2, pp. 1–13, 2019.

[6] A. Juansyah, “Pembangunan Aplikasi Child Tracker Berbasis Assisted – Global Positioning System ( A-GPS ) Dengan Platform Android,” *J. Ilm. Komput. dan Inform.*, vol. 1, no. 1, pp. 1–8, 2015, [Online].

[7] B. Hutauruk, B. D., Naibaho, J. F., & Rumahorbo, “Analisis Dan Perancangan Aplikasi Marketplace Cinderamata Khas Batak Berbasis Android,” *J. Method.*, vol. 3, no. 1, pp. 242–246, 2017.

[8] M. A. Akbar, S. N. Alam, J. Hutahaean, and J. Simarmata, *E-COMMERCE: Dasar Teori Dalam Bisnis Digital*. Yayasan Kita Menulis, 2020.

[9] A. A. Muflih, H. A. Ghazali, M. T. Sari, and P. N. Humaeroh, *Bisnis Ala Mahasiswa*. 2021.

[10] M. Muslihudin, A. Pramesta, and C. V. A. OFFSET, *Analisis dan Perancangan Sistem Informasi Menggunakan Model Terstruktur dan UML*. Penerbit Andi.

[11] A. Nugroho, *Rekayasa Perangkat Lunak Berorientasi Objek dengan Metode USDP*. Penerbit Andi.

[12] E. Triandini and I. G. Suardika, *Step by Step Desain Proyek Menggunakan UML*. Penerbit Andi.

[13] Y. Yudhanto and H. A. Prasetyo, *Panduan mudah belajar Framework Laravel*. Jakarta: PT Elex Media Komputindo, 2018.

[14] C. Tristianto, “Penggunaan Metode Waterfall untuk Pengembangan Sistem Monitoring dan Evaluasi Pembangunan Pedesaan,” pp. 8–22, 2018.

[15] M. M. Amin and S. Pringsewu, *Pemrograman Database Visual Basic 6 dan SQL Server 2000*. Penerbit Andi.

[16] W. Komputer, *Mastering CMS Programming with PHP & MySQL*. Bandung: Informatika Bandung, 2011.

[17] I. A. Watung and A. A. E. Sinsuw, “Perancangan Sistem Informasi Data Alumni Fakultas Teknik Unsrat Berbasis Web,” *J. Tek. Inform.*, vol. 3, no. 1, 2014, doi: 10.35793/jti.3.1.2014.3922.

[18] Anhar, *Panduan Menguasai PHP & MySQL Secara Otodidak*. Jakarta selatan: mediakita, 2017.

[19] Fridayanthie, E. Wida, and Tias Mahdiati, “Rancang Bangun Sistem Informasi Permintaan Atk Berbasis Intranet (Studi Kasus: Kejaksaan Negeri Rangkasbitung),” *J. Inf.*, vol. 4, pp. 126–137, 2016.