

PENGEMBANGAN SISTEM INFORMASI *DIRECT DEBIT DONOR PROGRAMME* (DDDP) DENGAN PENDEKATAN *INCREMENTAL LIFE CYCLE MODEL* (STUDI KASUS LEMBAGA KONSERVASI LINGKUNGAN DI JAKARTA)

S. Suwarjono¹, Achmad Hindasyah², T. Tukiyat³

^{1,2,3}Pascasarjana Magister Teknik Informatika

Sekolah Tinggi Manajemen Informatika Dan Komputer Eresha

Jl. Raya Puspiptek, Buaran, Kec. Serpong, Kota Tangerang Selatan, Banten 15310

E-mail: bsuwarjono@gmail.com, ahindasyah@gmail.com, dimastuky@gmail.com

ABSTRACT

Artikel:

Diterima: 24 Agustus 2020

Direvisi: 31 Januari 2021

Diterbitkan: 06 September 2021

***Alamat Korespondensi:**

bsuwarjono@gmail.com

Direct Debit Donor Programme (DDDP) is a donation program for individuals who are interested in taking part in social activities by direct debit of donations through bank accounts or credit cards which are commonly referred to as supporters or donors. Management of a structured and integrated information system with a good database concept will support a more effective and efficient performance. This study aims to analyze and develop the DDDP information system as a solution for supporter donation data management in increasing time efficiency. Information system development is carried out in stages starting from the need for basic functions in accordance with the development stages with the Incremental model, namely Requirements, Specifications, Architecture Design, Code and Test. modeling system development using the Unified Modeling Language (UML). Data collection was carried out through observation, direct interviews with related parties, viewing documents that were related to the research being carried out and conducting literature reviews. Functional testing on the system uses the black box testing method with the test results not found any errors in the system. The result of this research is a supporter donation data management information system which is named DDDP Application System.

Keywords: *Incremental, Unified Modeling Language, Black Box Testing*

ABSTRAK

Direct Debit Donor Programme (DDDP) adalah program donasi untuk perorangan yang tertarik ikut andil dalam kegiatan sosial dengan cara mendebit langsung donasi melalui rekening bank atau kartu kredit yang biasa disebut sebagai *Supporter* atau *donatur*, Pengelolaan sistem informasi yang terstruktur dan terintegrasi dengan konsep database yang baik akan menunjang kinerja yang lebih efektif dan efisien. penelitian ini bertujuan untuk menganalisis dan mengembangkan sistem informasi DDDP sebagai solusi pengelolaan data donasi *Supporter* dalam meningkatkan efisiensi waktu. Pengembangan sistem informasi dilakukan secara bertahap dimulai dari kebutuhan fungsi-fungsi dasar sesuai dengan tahapan pengembangan dengan model *Incremental* yaitu *Requirement, Specification, Architecture Desain, Code* dan *Test*. pemodelan pengembangan sistem menggunakan *Unified Modeling Language* (UML). Pengumpulan data dilakukan melalui observasi, wawancara langsung dengan pihak terkait, melihat dokumen yang memiliki keterkaitan dengan penelitian yang sedang dilakukan serta melakukan tinjauan pustaka. Pengujian fungsional pada sistem menggunakan metode *Black Box Testing* dengan hasil pengujian tidak ditemukan adanya kesalahan pada sistem. Hasil dari penelitian ini adalah sistem informasi pengelolaan data donasi *supporter* yang diberi nama *DDDP Application System*.

Kata Kunci: *Incremental, Unified Modeling Language, Black Box Testing*

I. PENDAHULUAN

Era revolusi industri 4.0 telah didepan mata, teknologi digital berkembang pesat didukung dengan koneksi internet yang semakin cepat hampir disemua bidang sudah terhubung dengan internet. Teknologi big data sudah mulai dimanfaatkan banyak perusahaan ataupun lembaga besar dalam menggali informasi yang bermanfaat bagi perkembangan dan kemajuan perusahaan dalam menghadapi persaingan yang semakin ketat [1]. Penerapan teknologi data mining, *bussiness intelligence* untuk mendukung kebijakan bisnis mulai menjadi salah satu pilihan banyak perusahaan dan lembaga besar untuk meningkatkan pelayanan dan strategi bisnis [2].

Perusahaan ataupun lembaga swasta yang ingin menerapkan teknologi data mining ataupun *bussiness intelligence* dalam menghadapi era revolusi industri 4.0 yang menuntut kecepatan dalam mendapatkan informasi guna mendukung keputusan dan

kebijakan bisnis, haruslah melalui tahapan-tahapan yang harus dipenuhi seperti ketersediaan sumber data atau basis data yang baik. Penerapan basis data yang terpusat dan terintegrasi dengan semua *resources* yang ada tentu akan menjadi langka awal yang baik dalam menghadapi era revolusi industri 4.0.

Direct Debit Donor Programme (DDDP) adalah program donasi yang dirancang untuk masyarakat secara perorangan yang tertarik untuk ikut andil dalam kegiatan sosial baik di bidang konservasi lingkungan, kemanusiaan, penelitian dan lainnya dengan cara mendebit secara langsung dana yang ada di rekening bank atau kartu kredit [3]. Program ini banyak digunakan oleh lembaga non pemerintah dalam menggalang dan masyarakat untuk mendanai semua kegiatan yang dilakukan. Salah satu lembaga konservasi alam di jakarta tempat penelitian ini dilakukan menjalankan program DDDP dalam menggalang dana dari masyarakat secara perorangan untuk ikut andil

dalam membiayai kegiatan konservasi lingkungan yang dikelola langsung oleh divisi *Individu Fundraiser* dengan menjalankan prinsip menjaga kerahasiaan data *Supporter* dengan standar operasional prosedur yang ketat dan membatasi akses kepada data *Supporter* secara langsung. Dalam menjalankan kegiatan DDDP divisi *Individu Fundraiser* terbagi menjadi beberapa tim dalam pengelolaan data *Supporter* dengan pembagian tugas yang berbeda yaitu

1. *Fundraiser*, terjun kelapangan untuk memberikan edukasi dan mengajak masyarakat untuk ikut andil dalam kegiatan konservasi lingkungan dengan cara berdonasi dan menjadi *Supporter*.
2. *Settlement*, melakukan proses verifikasi data calon *Supporter* ke bank mitra dan melakukan proses pengajuan pendebitan donasi *Supporter* ke bank mitra.
3. *Contact center*, melakukan konfirmasi ke *Supporter* ataupun melayani keluhan dan permintaan dari *Supporter* yang berhubungan dengan *update* data, *cancel*, *upgrade*, *downgrade*, *hold* dan *refund* donasi.
4. *Delivery support*, menyiapkan data paket berupa souvenir, majalah dan laporan kegiatan ke *Supporter* sebagai apresiasi penghargaan karna telah ikut andil dalam kegiatan konservasi lingkungan dalam bentuk donasi.

khusus untuk tim *fundraiser* ada di beberapa kota besar Indonesia. Sistem informasi pengelolaan DDDP sebelumnya menggunakan jasa mitra selaku pihak ketiga namun mulai tahun 2016 sistem pengelolaan DDDP dilakukan secara mandiri dan setiap tim menggunakan sistem masing-masing yang mengakibatkan dibutuhkan waktu yang lebih lama dalam sinkronisasi data *Supporter* ataupun dalam membuat laporan dikarenakan belum tersedianya sistem yang terintegrasi dan terstruktur yang dapat digunakan bersama oleh semua tim dalam mengelola DDDP.

Melihat akan kebutuhan sistem informasi pengelolaan data *Supporter* yang terstruktur dan terintegrasi dalam menunjang kinerja dan meningkatkan efisiensi serta efektifitas pengelolaan data *Supporter* maka peneliti bermaksud mengembangkan sistem informasi DDDP. Metode pengembangan sistem informasi harus memperhatikan fungsionalitas dan kebutuhan bisnis dalam menunjang kinerja

masing-masing tim yang bisa berkembang secara dinamis mengikuti perkembangan teknologi dan zaman, untuk itu pengembangan sistem sebaiknya dikembangkan secara bertahap mulai dari kebutuhan fungsional yang paling pokok yaitu kebutuhan dasar yang harus terpenuhi dahulu kemudian terus dikembangkan sampai semua kebutuhan fungsional terpenuhi dan kedepannya tetap bisa dikembangkan lagi sesuai kebutuhan perkembangan inovasi dan strategi bisnis dalam pengelolaan data *Supporter*. Dalam rangka memudahkan masing-masing tim beradaptasi dengan sistem informasi baru serta meminimalisir ketidaksesuaian pada sistem yang dikembangkan maka metode yang tepat dalam pengembangan sistem informasi DDDP adalah *Incremental Life Cycle Model* [4]. Tahapan pengembangan model *incremental* secara umum ada empat tahapan yang biasa digunakan [5].

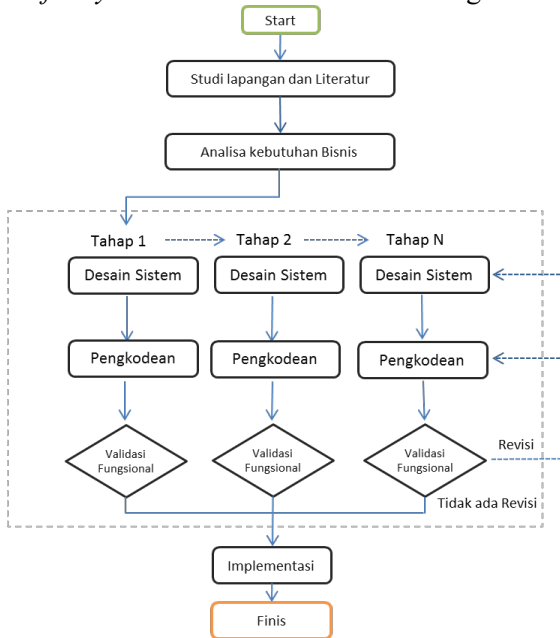
1. *Requirement*, yaitu melakukan analisis kebutuhan bisnis baik kebutuhan fungsional maupun non fungsional yang diperlukan dalam pengembangan sistem.
2. *Design and development*, yaitu melakukan proses desain dan pengembangan sistem dan jika ada modul baru yang akan ditambahkan maka akan dilakukan desain dan *development* kembali.
3. *Testing*, untuk melihat kesiapan dan respon dari sistem yang dikembangkan
4. *Implementation*, proses penggunaan sistem yang telah dikembangkan.

Tujuan dari penelitian ini adalah untuk menganalisa sistem informasi DDDP, merancang arsitektur desain pengembangan sistem informasi DDDP dan mengembangkan aplikasi sistem informasi DDDP yang bisa digunakan oleh tim divisi *Individu Fundraiser* sebagai solusi dalam pengelolaan data *Supporter* yang terintegrasi dan terstruktur dalam rangka meningkatkan kualitas laporan dan menunjang kinerja yang lebih baik serta berdampak pada peningkatan pelayanan dan kebijakan bisnis yang lebih efektif dan efisien [6].

II. METODOLOGI

Jenis penelitian pengembangan sistem informasi DDDP adalah *research and development* (R&D) yaitu menghasilkan

rancangan sistem dan perangkat lunak Sistem Informasi DDDP yang dikembangkan menggunakan tahapan atau langkah-langkah pengembangan perangkat lunak *Incremental life cycle model*. Gambaran kerangka kerja



Gambar 1. Kerangka Kerja Penelitian penelitian ini dapat dilihat pada Gambar 1.

III. HASIL DAN PEMBAHASAN

3.1 Analisa Kebutuhan Bisnis

Pada tahap ini akan dilakukan analisa bisnis yang ada di lembaga konservasi alam di Jakarta khususnya dibagian divisi *Individu Fundraiser* yang menjalankan atau membawahi *Direct Debit Donor Programme (DDDP)* dengan cara observasi dan wawancara langsung dengan semua pelaku yang berkaitan langsung dengan program DDDP untuk mengetahui letak permasalahan yang dihadapi dan melakukan analisa pada masalah tersebut agar bisa diidentifikasi secara benar untuk memudahkan dalam mencari solusi dalam menyelesaikan masalah-masalah yang dihadapi. Saat ini divisi *Individu Fundraiser* telah membuka cabang dan menempatkan tim *fundraiser* di beberapa kota besar Indonesia seperti DKI Jakarta, kota Bandung, Yogyakarta dan kota Surabaya. Dengan melakukan proses analisa secara menyeluruh terhadap pihak yang memiliki keterkaitan langsung maupun tidak langsung dalam rangka identifikasi kebutuhan

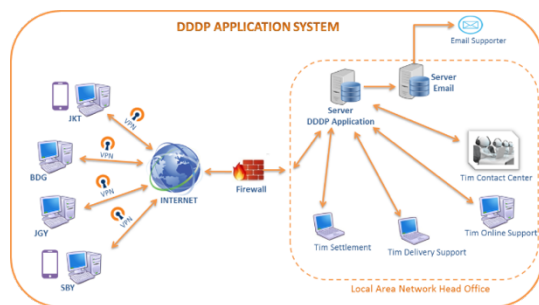
bisnis program DDDP agar dapat berjalan dengan baik dan memberikan solusi yang tepat dalam pengelolaan data *Supporter* oleh tim divisi *Individu Fundraiser* dengan tetap memperhatikan integritas dan tanggung jawab masing-masing anggota tim untuk menjaga kerahasiaan data *Supporter* agar tidak dapat disalahgunakan. Dalam pelaksanaan kegiatan pengelolaan data *Supporter* ada beberapa istilah kerja atau nama kegiatan yang digunakan seperti keterangan dalam tabel 1.

Tabel 1. Keterangan Istilah Kerja

| Nama Istilah Kegiatan | Keterangan |
|------------------------|--|
| <i>Registration</i> | Proses pendaftaran calon <i>Supporter</i> baru |
| Pengiriman Formulir | Pengiriman formulir asli calon <i>Supporter</i> dari kantor cabang ke kantor pusat di Jakarta |
| Verifikasi data | Proses verifikasi ke bank mitra terhadap kelengkapan data calon <i>Supporter</i> sesuai syarat bank mitra |
| <i>Submission bank</i> | Proses pengajuan pendebitan donasi <i>Supporter</i> aktif ke bank mitra untuk diproses |
| <i>Result bank</i> | Proses <i>update</i> data <i>history</i> donasi <i>Supporter</i> berdasarkan laporan pendebitan berhasil atau tidak yang dilakukan oleh bank mitra |
| <i>Update data</i> | Proses <i>update</i> data <i>Supporter</i> yang dilakukan oleh <i>contact center</i> |
| <i>Cancel data</i> | Proses permohonan berhenti berdonasi sebagai donator oleh <i>Supporter</i> yang diajukan ke <i>contact center</i> |
| <i>Downgrade</i> | Proses permohonan untuk menurunkan jumlah donasi oleh <i>Supporter</i> yang diajukan melalui <i>contact center</i> |
| <i>Upgrade</i> | Proses permohonan untuk menaikkan donasi oleh <i>Supporter</i> yang diajukan melalui <i>contact center</i> |
| <i>Hold</i> | Proses permohonan untuk berhenti sementara berdonasi oleh <i>Supporter</i> yang diajukan melalui <i>contact center</i> |

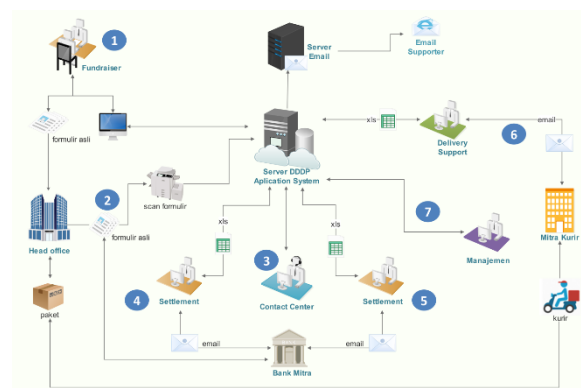
| | |
|--------------------|---|
| <i>Refund</i> | Proses permohonan pengembalian donasi yang telah terdebit oleh bank mitra ke rekening <i>Supporter</i> kembali yang di ajukan melalui contact center |
| <i>Delivery</i> | Proses pengiriman bingkisan berupa souvenir, majalah dan laporan ke <i>Supporter</i> sebagai bentuk apresiasi penghargaan ke <i>Supporter</i> karna telah ikut berdonasi untuk kegiatan konservasi lingkungan |
| <i>Email Blast</i> | Pengiriman email ke <i>Supporter</i> |

Kebutuhan kebutuhan fungsional yang diharapkan pada pengembangan sistem informasi DDDP adalah mengintegrasikan pengelolaan data *Supporter* oleh semua tim dalam satu kesatuan sistem aplikasi yang terintegrasi dan terstruktur yang diberi nama DDDP *Application System*. Dalam pengembangan dan implementasinya akan memanfaatkan sumber daya teknologi informasi yang telah ada dan dimiliki oleh lembaga yaitu Server Data, Server Email, Server VPN (*Virtual Private Network*), *IP Public* dan *Local Network*. Sehingga DDDP *Application System* terpusat pada satu server dan dapat diakses dari mana saja dan kapan saja menggunakan akses *Virtual Private Network* (VPN) untuk keamanan akses data melalui jaringan internet yang dapat digunakan oleh pihak yang berkepentingan baik menggunakan komputer ataupun smartphone [7]. Akses ke DDDP *Application System* tidak bisa sembarangan diakses oleh pihak yang tidak diijinkan karna harus memiliki akses VPN untuk terhubung jika berada di luar *Local Area Network Head Office*. Arsitektur rancangan Network DDDP *Application System* dapat dilihat pada gambar 2.



Gambar 2. Arsitektur Network DDDP *Application System*

Arsitektur pengembangan DDDP *Application System* dapat dilihat pada gambar 3.



Gambar 3. Architecture DDDP *Application System*

Untuk keterangan masing-masing proses dapat dijelaskan sebagai berikut:

1. Proses *registrasi* oleh tim fundraiser yaitu proses *input* dan *update* data calon *Supporter* menggunakan DDDP *Application System* melalui akses VPN sehingga data langsung tersimpan ke dalam database diserver *Head office* dan akan mengirim email konfirmasi otomatis ke calon *Supporter*.
2. Proses *Scan Hard Copy* adalah proses penyimpanan hasil *scan* dari form asli *Supporter* yang berbentuk *file PDF* dengan cara di *upload* ke DDDP *Application System* dan dihubungkan atau *link* langsung ke data *Supporter*.
3. Proses verifikasi kelengkapan data calon *Supporter* yang kurang lengkap atau kurang jelas maka akan dilakukan proses verifikasi data oleh tim *contact center*, selain itu juga melakukan proses *cancel*, *upgrade*, *downgrade*, *hold* dan *refund* donasi atas permintaan *Supporter*, langsung dilakukan melalui DDDP *Application System* dan secara otomatis email konfirmasi akan dikirimkan ke *Supporter*.
4. Proses verifikasi calon *supporter* ke bank mitra menggunakan DDDP *Application System* dengan cara mengexport data calon *Supporter* yang telah di *input* tim fundraiser sebelumnya dengan *file xls* dengan format sesuai standar bank mitra yang dituju, data hasil *export* tersebut dikirimkan menggunakan email sedangkan formulir asli akan diantar langsung ke bank mitra untuk

diverifikasi oleh pihak bank. hasil verifikasi oleh bank akan dilaporkan melalui email dalam bentuk file xls. data hasil verifikasi oleh bank akan di import ke *DDDP Application System* maka secara otomatis akan mengupdate status calon *Supporter* jika verifikasi berhasil maka status calon *Supporter* menjadi aktif dan jika tidak maka statusnya non aktif.

5. Proses *submmision* bank dan *result* bank yaitu mengexport data *Supporter* yang aktif dan telah masuk periode waktu untuk donasi melalui *DDDP Application System*. Format data dalam bentuk file xls disesuaikan dengan format masing masing bank mitra dan dikirimkan melalui email untuk diproses pendebitan donasi oleh pihak bank mitra. Hasil pendebitan akan dilaporkan pihak bank mitra (*result* bank) dalam bentuk file xls atau txt yang berisi data berhasil atau tidak didebit. Data yang dikirimkan oleh bank tersebut di *import* ke *DDDP Application System* untuk mengupdate data *history* donasi *Supporter*. Data *Supporter* yang berhasil di debit maka akan dikirim email konfirmasi kepada masing-masing *Supporter*.
6. Proses *delivery* paket yaitu mengambil data *Supporter* yang berhasil didebit dari *DDDP Application System* sesuai dengan kriteria tertentu untuk dikirimkan paket ke *Supporter*, data di *export* ke dalam file xls dan dikirim ke mitra atau vindow pengiriman paket untuk diproses pengirimannya, file dikirim melalui email untuk di data oleh pihak mitra. Kemudian *delivery support* akan menyiapkan paket yang akan diambil oleh kurir mitra. *DDDP Application System* akan mengirimkan email ke *Supporter* yang akan dikirim paket sebagai konfirmasi. Setelah proses pengiriman dilakukan maka vindow akan melaporkan hasil pengiriman paket dengan mengirimkan file laporan dalam bentuk file xls. data laporan hasil pengiriman akan di *import* ke *DDDP Application System* untuk mengupdate data *Supporter* yang menerima paket.
7. Proses laporan yaitu menampilkan laporan rekap donasi *Supporter* dan status terakhir *Supporter* berdasarkan priode waktu.

3.2 Perancangan Sistem

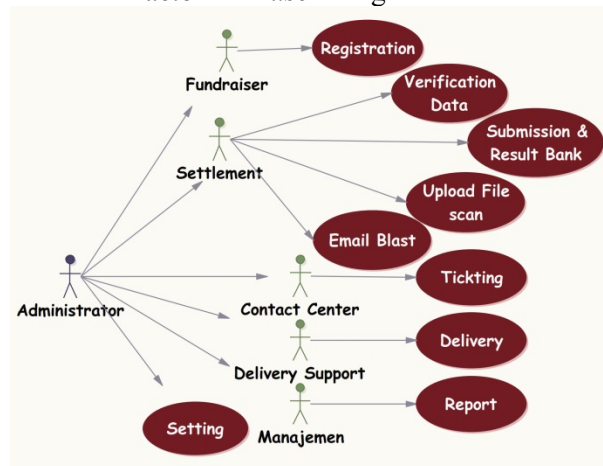
Dalam proses desain atau perancangan sistem informasi *DDDP* peneliti tidak melakukan banyak perubahan terhadap proses yang telah berjalan saat ini karna akan membutuhkan banyak penyesuaian oleh *user* nantinya yang menyebabkan mereka merasa berat untuk menggunakan sistem yang baru, untuk itu peneliti hanya berupaya untuk menterjemahkan sistem yang berjalan saat ini ke dalam sistem informasi baru yang menggunakan basis data yang tersentral dan terintegrasi yang digunakan bersama oleh semua tim dalam pengelolaan data *Supporter*, maka dari itu peneliti tidak banyak mengubah susunan *field-field* atau kolom yang telah digunakan dalam penyimpanan pengolahan data *Supporter* tapi berusaha mengkonversi ke dalam bentuk tabel-tabel basis data model relasional yang telah didukung oleh banyak *database manajemen system* (DBMS) karena memiliki banyak keuntungan dan kemudahan terutama dalam hal integrasi data yang sangat dibutuhkan dalam pengembangan sistem informasi *DDDP*. Pada penelitian ini DBMS yang digunakan sebagai media basis data adalah *Mysql 5.7* yang telah mendukung *Relational Database Management System* (RDBMS) dengan lisensi *GPL (General Public License)* dengan dukungan integrasi pada banyak bahasa pemrograman [8]. Dalam rancangan basis data *DDDP* peneliti melakukan penyesuaian terhadap *field-field* yang dibutuhkan dengan tetap melakukan proses normalisasi untuk mengelompokkan elemen data menjadi tabel-tabel dengan hubungan relasi pada entitasnya [9].

Rancangan pemodelan sistem menggunakan standar *Unified Modeling Language* (UML) sebagai bahasa grafis yang mudah dipahami dan mudah untuk digunakan dalam mendokumentasikan rancangan sistem informasi, UML memiliki beberapa standar komponen dalam membuat atau merancang desain sistem diantara komponen yang sering digunakan adalah *use case diagram*, *class diagram*, *activity diagram* dan *sequence diagram* [10]. Sedangkan komponen UML yang paling banyak digunakan dalam pengembangan sistem dengan keterlibatan *user* adalah komponen *use case narrative* [11]. dalam penelitian ini rancangan pemodelan yang dibuat ada 4 yaitu *use case*, *activity*, *sequence* dan *class diagram*. Pada pemodelan *use case*

diagram ditambahkan penjelasan dalam bentuk teks *specification* untuk menjelaskan lebih rinci gambaran rancangan proses DDDP dalam bentuk naratif. berikut gambaran *use case diagram* dan *use case specification*

3.2.1 Use Case Diagram

Perancangan *use case diagram* berfungsi untuk mendeskripsikan tipikal interaksi antara *actor* atau *user* dengan sistem.



Gambar 5. Use Case Diagram

3.2.2 Use Case Specification

Penjelasan hanya dengan *use case diagram* saja kadang belum cukup butuh penjelasan dalam bentuk naratif teks untuk menjelaskan fungsi dan tujuan yang ingin dicapai ketika *use case* dijalankan [12].

Tabel 2. Use case Specification

| | |
|----------------------------|--|
| <i>Use Case Name</i> | <i>Registration</i> |
| <i>Actor(s)</i> | TIM <i>Fundraiser</i> |
| <i>Summary Description</i> | Fasilitas untuk menginput dan mengupdate data calon <i>Supporter</i> baru yang ingin berdonasi |
| <i>Priority</i> | Harus ada |
| <i>Basic Path</i> | <ol style="list-style-type: none"> 1. <i>Login</i> ke aplikasi menggunakan <i>username</i> dan <i>password</i> masing masing 2. Masukan data calon <i>Supporter</i> sesuai dengan kolom yang disediakan 3. Data calon <i>Supporter</i> yang di <i>input</i> oleh <i>fundraiser</i> melalui <i>mock up mobile</i> harus divalidasi oleh <i>Admin city</i> 4. Data calon <i>Supporter</i> yang menggunakan <i>payment</i> atau berdonasi melalui bank <i>account</i> |

akan dilakukan verifikasi terlebih dahulu ke bank mitra sebelum data tersebut berstatus aktif

5. *Fundraiser* hanya dapat melihat daftar calon *Supporter* yang di *input* sendiri
6. *Admin city* dapat melihat data calon *Supporter* sesuai dengan wilayah masing masing

Business Rule

1. Harus terhubung ke jaringan *server DDDP Application system* bisa menggunakan media *internet* dengan akses *VPN*
2. Harus memiliki *username* dan *password*
3. Data yang telah berhasil di *input* oleh *Fundraiser* tidak dapat di *update* atau di *edit* oleh yang bersangkutan tapi yang bisa mengupdate adalah *Admin city*
4. *Admin city* hanya diijinkan mengupdate data calon *Supporter* yang telah di *input* selama 2 hari setelah data berhasil disimpan.
5. *DDDP Application System* harus terkoneksi ke *server email* secara langsung melalui *SMTP* untuk proses kirim *email* konfirmasi

| | |
|----------------------------|--|
| <i>Use Case Name</i> | <i>Verification Data</i> |
| <i>Actor(s)</i> | Tim <i>Settlement</i> |
| <i>Summary Description</i> | Melakukan proses verifikasi data calon <i>Supporter</i> yang berdonasi menggunakan bank <i>account</i> ke bank mitra kemudian memproses laporan hasil dari bank mitra ke sistem untuk mengupdate status calon <i>Supporter</i> |
| <i>Priority</i> | Harus ada |
| <i>Basic Path</i> | <ol style="list-style-type: none"> 1. <i>User settlement Supporter</i> harus <i>login</i> terlebih dahulu 2. Mengambil data calon <i>Supporter</i> yang berdonasi dengan <i>account</i> bank kemudian mengirim data tersebut ke bank mitra dalam bentuk rekap file <i>xls</i> atau <i>txt</i> 3. Laporan hasil verifikasi bank mitra yang diberikan dalam bentuk rekap file <i>xls</i> atau <i>txt</i> kemudian di <i>upload</i> ke <i>system</i> untuk mengupdate status calon <i>Supporter</i> baru |
| <i>Business Rule</i> | <ol style="list-style-type: none"> 1. Harus terhubung ke jaringan <i>server DDDP Application system</i> bisa menggunakan media <i>internet</i> dengan akses <i>VPN</i> . 2. Harus memiliki <i>username</i> dan <i>password</i> |

| | |
|----------------------------|--|
| | 3. File yang diupload harus format xls atau txt |
| <i>Use Case Name</i> | <i>Submission & Result Bank</i> |
| <i>Actor(s)</i> | Tim Settlement |
| <i>Summary Description</i> | Melakukan proses <i>submission</i> bank atau pengajuan pendebitan donasi ke bank mitra sesuai dengan jumlah <i>Supporter</i> aktif setiap bulannya kemudian melakukan proses <i>result</i> bank atau <i>update</i> status donasi <i>Supporter</i> berdasarkan hasil laporan pendebitan donasi oleh bank mitra |
| <i>Priority</i> | Harus ada |
| <i>Basic Path</i> | <ol style="list-style-type: none"> 1. <i>User settlement Supporter</i> harus login terlebih dahulu 2. Mengambil data calon <i>Supporter</i> yang aktif untuk berdonasi dan mengirimkan rekap data tersebut ke bank mitra dalam bentuk file xls atau txt sesuai format yang diberikan oleh bank mitra 3. File laporan hasil pendebitan yang dilakukan oleh bank mitra dalam bentuk rekap data file xls atau txt diupload ke sistem untuk mengupdate <i>history</i> donasi <i>Supporter</i> |
| <i>Business Rule</i> | <ol style="list-style-type: none"> 1. Harus terhubung ke jaringan server DDDP <i>Application system</i> bisa menggunakan media <i>internet</i> dengan akses VPN 2. Harus memiliki <i>username</i> dan <i>password</i> 3. File yang diupload harus format xls atau txt |
| <i>Use Case Name</i> | <i>Upload File Scan</i> |
| <i>Actor(s)</i> | Tim Settlement |
| <i>Summary Description</i> | Proses penyimpanan file <i>scan</i> formulir <i>Supporter</i> ke dalam sistem agar dapat digunakan dalam proses pelayanan <i>Supporter</i> dan bukti dokumentasi jika dibutuhkan dengan dimasukan kedalam sistem maka akan mudah dicari kembali |
| <i>Priority</i> | Harus ada |
| <i>Basic Path</i> | <ol style="list-style-type: none"> 1. <i>User admin Supporter</i> harus login terlebih dahulu 2. <i>Upload</i> file <i>scan</i> formulir ke sistem dengan nama file sama dengan <i>serial no Supporter</i> |
| <i>Business Rule</i> | <ol style="list-style-type: none"> 1. Harus terhubung ke jaringan server DDDP <i>Application system</i> |

| | |
|----------------------------|---|
| | bisa menggunakan media <i>internet</i> dengan akses VPN |
| | 2. Harus memiliki <i>username</i> dan <i>password</i> |
| | 3. File harus dalam bentuk pdf dengan nama sesuai dengan <i>serial no Supporter</i> |
| <i>Use Case Name</i> | <i>Email Blast</i> |
| <i>Actor(s)</i> | Tim Settlement |
| <i>Summary Description</i> | Melakukan proses <i>email blast</i> kepada <i>Supporter</i> yang memiliki email sesuai dengan kebutuhan dan kebijakan dari manajemen, data <i>Supporter</i> yang akan dikirim email bisa berdasarkan hasil pendebitan donasi, pengiriman paket atau yang lainnya sesuai situasi dan kondisi |
| <i>Priority</i> | Dianjurkan Ada |
| <i>Basic Path</i> | <ol style="list-style-type: none"> 1. <i>User</i> harus login terlebih dahulu 2. Memilih jenis data <i>Supporter</i> yang akan menerima email bast berdasarkan status kebutuhan saat mengirim email bast, bisa berdasarkan status pendebitan donasi, penerima paket barang souvenir ataupun hal yang lainnya |
| <i>Business Rule</i> | <ol style="list-style-type: none"> 1. Harus terhubung ke jaringan server DDDP <i>Application system</i> bisa menggunakan media <i>internet</i> dengan akses VPN 2. Harus memiliki <i>username</i> dan <i>password</i> 3. DDDP <i>Application System</i> harus terkoneksi ke <i>server email</i> secara langsung melalui SMTP untuk proses kirim <i>email</i> ke <i>Supporter</i> |
| <i>Use Case Name</i> | <i>Ticketing</i> |
| <i>Actor(s)</i> | Tim Contact Center (<i>Call Center, Supporter Service</i>) |
| <i>Summary Description</i> | Melayani proses <i>ticketing</i> kepada <i>Supporter</i> seperti <i>update</i> data <i>Supporter</i> , <i>update credit card</i> , proses <i>cancel</i> dan <i>hold</i> donasi, <i>downgrade</i> dan <i>upgrade</i> donasi, proses <i>refund</i> donasi dan keluhan atau informasi lainnya dari para <i>Supporter</i> |
| <i>Priority</i> | Harus ada |
| <i>Basic Path</i> | <ol style="list-style-type: none"> 1. <i>User</i> harus login terlebih dahulu 2. Memilih jenis tiket apakah <i>inbound</i> atau <i>outbound</i> 3. Memilih kasus sesuai dengan jenis pelayanan ke <i>Supporter</i> (<i>update</i> data, <i>cancel</i> dan <i>hold</i> donasi, <i>upgrade</i> atau <i>downgrade</i>) |

| | |
|----------------------------|--|
| | donasi, <i>update credit card</i> , <i>refund</i> donasi dan lainnya) |
| | 4. Untuk proses <i>cancel</i> donasi, <i>hold</i> donasi, <i>upgrade</i> dan <i>downgrade</i> donasi serta <i>refund</i> donasi harus divalidasi oleh <i>Supporter service</i> |
| <i>Business Rule</i> | 1. Harus terhubung ke jaringan server DDDP Application system berada (bisa menggunakan media internet atau VPN) 2. Harus memiliki username dan password 3. DDDP Application System harus terkoneksi ke server email secara langsung melalui SMTP untuk proses kirim email konfirmasi otomatis ke <i>Supporter</i> 4. DDDP Application System harus terkoneksi ke server email secara langsung melalui SMTP untuk proses kirim email konfirmasi ke <i>Supporter</i> |
| <i>Use Case Name</i> | <i>Delivery</i> |
| <i>Actor(s)</i> | Tim <i>Delivery Support</i> |
| <i>Summary Description</i> | Proses pengolahan data <i>Supporter</i> yang akan menerima paket souvenir sebagai penghargaan kepada <i>Supporter</i> yang telah aktif berdonasi dalam kegiatan program konservasi lingkungan, mengelola stok data barang paket souvenir agar tersedia sesuai dengan jumlah kebutuhan yang akan dikirim ke <i>Supporter</i> |
| <i>Priority</i> | Dianjurkan ada |
| <i>Basic Path</i> | 1. <i>User</i> harus <i>login</i> terlebih dahulu 2. Melakukan pemilihan data <i>Supporter</i> yang akan menerima paket barang souvenir atau majalah kegiatan konservasi tahunan berdasarkan data <i>Supporter</i> aktif yang berhasil didebit donasinya melalui proses <i>submission</i> bank 3. Data <i>Supporter</i> yang akan menerima paket barang harus divalidasi oleh <i>Supporter delivery</i> karna secara otomatis akan mengurangi stok barang yang ada digudang 4. Mengirimkan data <i>Supporter</i> yang akan menerima paket barang ke perusahaan kurir yang menjadi mitra dalam bentuk rekap file excel dan menyiapkan barang-barang paket tersebut agar siap diambil oleh perusahaan mitra jasa pengiriman paket |

| | |
|----------------------------|---|
| | 5. Menerima laporan hasil pengiriman paket barang oleh perusahaan mitra tentang status pengiriman melalui dalam bentuk rekap data file xls |
| | 6. <i>Upload</i> file laporan tersebut ke sistem untuk mengupdate status pengiriman paket barang sebagai <i>history</i> data penerimaan paket oleh <i>Supporter</i> |
| <i>Business Rule</i> | 1. Harus terhubung ke jaringan server DDDP Application system bisa menggunakan media internet dengan akses VPN 2. Harus memiliki <i>username</i> dan <i>password</i> 3. Format file xls |
| <i>Use Case Name</i> | <i>Report</i> |
| <i>Actor(s)</i> | Tim Manajemen |
| <i>Summary Description</i> | Menampilkan beberapa bentuk laporan status <i>Supporter</i> dari proses <i>registrasi</i> , proses <i>submission</i> bank dan <i>tickting</i> serta laporan rekap donasi berdasarkan hasil dari proses pendebitan donasi <i>Supporter</i> . Harus ada |
| <i>Priority</i> | Harus ada |
| <i>Basic Path</i> | 1. <i>User</i> harus <i>login</i> terlebih dahulu 2. Memilih jenis laporan yang diinginkan 3. Memilih periode data yang diinginkan sesuai kebutuhan |
| <i>Business Rule</i> | 1. Harus terhubung ke jaringan server DDDP Application system bisa menggunakan media internet dengan akses VPN 2. Harus memiliki <i>username</i> dan <i>password</i> |
| <i>Use Case Name</i> | <i>Setting</i> |
| <i>Actor(s)</i> | <i>Administrator</i> |
| <i>Summary Description</i> | Melakukan manajemen <i>user</i> yang akan mengakses ke sistem, menginput dan mengupdate data pendukung yang akan digunakan untuk memudahkan proses <i>input</i> data dalam formulir <i>registrasi</i> atau <i>form</i> yang lain dalam bentuk list data yang dipilih oleh user sehingga tidak perlu mengentikan data tersebut pada kolom tapi tinggal dipilih saja karna sudah dalam bentuk list data seperti <i>list</i> propinsi, kota, kecamatan, <i>source</i> , <i>campign</i> , <i>vanue</i> , <i>fundraiser</i> dan bank |
| <i>Priority</i> | Harus ada |
| <i>Basic Path</i> | 1. <i>User</i> harus <i>login</i> terlebih dahulu |

2. Memilih menu *setting* yang ada untuk proses *insert*, *update* dan *delete* data

Business Rule

1. Harus terhubung ke jaringan server DDDP Application system bisa menggunakan media *internet* dengan akses VPN
2. Harus memiliki *username* dan *password*

3.3 Pengkodean Sistem

Pengkodean sistem adalah tahap menterjemahkan desain sistem kedalam bahasa pemrograman yang dikenal komputer sehingga bisa menampilkan dan melakukan proses pengolahan data sesuai dengan yang diharapkan. Dalam penelitian ini sistem dikembangkan menggunakan bahasa program *Hypertext Preprocessor* (PHP) yang bersifat *server-side* yaitu pemrosesan data dilakukan disisi server dan berbasis web serta mudah dikombinasikan dengan *Relational Database Management System* (RDBMS) Mysql [13]. untuk memudahkan penggunaan sistem oleh *user* maka *interface* atau antarmuka sistem dikembangkan menggunakan *framework Easyui* yang telah mendukung HTML5 dan memiliki komponen-komponen antarmuka yang siap pakai sehingga memudahkan dalam proses pengembangan antarmuka sistem yang sedang dikembangkan [14].

Proses pengkodean dilakukan dalam 5 tahapan sesuai dengan tahapan pengembangan *incremental life cycle model* yaitu adanya pembagian dan pemilihan modul yang akan dikembangkan terlebih dahulu berdasarkan desain sistem yang telah dibuat kemudian dilanjutkan ke modul berikutnya sampai semua modul yang dibutuhkan selesai semua, dari desain tersebut maka pengembangan sistem dibagi kedalam 5 tahapan pengembangan dengan urutan modul yang dikembangkan sebagai berikut :

1. Tahap pertama pengembangan modul *Registration* dan *Verification* serta *Settings*
2. Tahap kedua modul *Submission Bank*, *Email Blast* dan *Upload Scan File*
3. Tahap ketiga modul *Ticketing*
4. Tahap keempat modul *Delivery*
5. Tahap kelima modul *Report* atau Laporan

Tampilan salah satu modul antarmuka pada sistem yaitu *form* registrasi calon *Supporter*

hasil pengembangan sistem informasi DDDP dapat dilihat pada gambar 6.

Gambar 6. Tampilan Form Registrasi

3.4 Black Box Testing

Black box testing merupakan pengujian yang didasarkan pada detail aplikasi seperti fungsi-fungsi yang ada pada aplikasi, tampilan aplikasi dan kesesuaian alur fungsi dengan bisnis proses yang diinginkan oleh pengguna [15]. Kegiatan yang dilakukan saat tester diantaranya adalah:

1. Membuat *test case* untuk menguji fungsi-fungsi yang ada pada aplikasi.
2. Membuat *test case* untuk menguji kesesuaian alur kerja suatu fungsi di aplikasi dengan *requirement/spesifikasi* yang dibutuhkan pengguna.
3. Mencari *bugs/error* dari tampilan (*interface*) aplikasi.

Pengujian *black box testing* tidak melakukan pengujian dari segi kode program tapi lebih berfokus terhadap fungsional program [16]. Pengujian fungsional terhadap DDDP Application System yang telah dikembangkan menggunakan metode *black box testing* dengan menggunakan secara langsung aplikasinya, pengujian fungsional ini dilakukan untuk menguji kesesuaian fungsi dan modul yang ada dengan spesifikasi yang telah dibuat dalam rancangan sistem. Dalam pengujian validasi fungsional pada penelitian ini dilakukan pada 4 aspek pengujian secara umum yaitu :

1. Pengujian *Interface* atau antarmuka sistem yang bertujuan untuk mengetahui fungsionalitas dari elemen-elemen *interface* yang terdapat pada sistem.
2. Pengujian dasar sistem yang bertujuan untuk mengetahui kerja dari fungsi-fungsi dasar sistem apakah sudah berjalan sesuai dengan kebutuhan apa belum.

3. Pengujian *form handle* sistem yang bertujuan untuk melihat kemampuan sistem dalam menangani inputan yang diberikan oleh *user*.
4. Pengujian keamanan sistem yang bertujuan untuk melihat atau mengetahui aspek keamanan yang dimiliki oleh sistem.

Tabel 3. Pengujian Interface Sistem

| Tes Kasus | Hasil Yang diharapkan | Status |
|---------------------------------------|--|----------|
| Pengujian <i>button submit/simpan</i> | Sistem akan memproses <i>form</i> | Berhasil |
| Pengujian <i>button search/cari</i> | Sistem akan memproses pencarian data | Berhasil |
| Pengujian <i>button cancel /close</i> | Sistem akan keluar dari <i>interface form</i> dan kembali ke halaman utama <i>module</i> yang sedang aktif | Berhasil |
| Pengujian <i>button edit/update</i> | Sistem akan masuk ke <i>form edit</i> data | Berhasil |
| Pengujian <i>button delete/hapus</i> | Sistem akan menjalankan proses menghapus data yang dipilih | Berhasil |
| Pengujian menu | Sistem akan masuk ke modul yang sesuai dengan menu yang dipilih | Berhasil |

Tabel 4. Pengujian Fungsi Dasar Sistem

| Tes Kasus | Hasil Yang diharapkan | Status |
|--|--|----------|
| Pengujian fungsi tampilan data | Data akan ditampilkan sesuai modul yang dipilih | Berhasil |
| Pengujian fungsi <i>upload</i> data tipe file PDF | File akan masuk ke dalam file <i>server</i> dan meta data file akan dimasukkan kedalam basis data | Berhasil |
| Pengujian fungsi <i>upload</i> data tipe excel (format 97-2003) | File masuk kedalam file temporer <i>server</i> dan meta data file akan memasukan ke dalam basis data | Berhasil |
| Pengujian <i>export</i> data ke dalam tipe file excel (format 97-2003) | File dalam bentuk file excel berhasil di download dari sistem | Berhasil |
| Pengujian fungsi ubah data | Data akan berubah sesuai dengan input user | Berhasil |
| Pengujian hapus data | Data yang dipilih akan terhapus dari basis data | Berhasil |

Pengujian *Email* terkirim dari sistem ke *email* tujuan *balst* Berhasil

Tabel 5. Pengujian Form Handle Sistem

| Tes Kasus | Hasil Yang diharapkan | Status |
|--|---|----------|
| Pengisian <i>form</i> dengan cara yang benar atau data benar | Sistem akan memproses <i>form</i> tersebut | Berhasil |
| Pengisian <i>form</i> salah atau data salah | Sistem tidak akan memproses <i>form</i> dan memberikan <i>feedback</i> kepada <i>user</i> | Berhasil |
| <i>Form</i> kosong lalu di <i>submit</i> | Sistem tidak akan memproses dan memberikan <i>feedback</i> kepada <i>user</i> | Berhasil |

Tabel 6. Pengujian Keamanan Sistem

| Tes Kasus | Hasil Yang diharapkan | Status |
|---|---|----------|
| <i>User</i> tidak melakukan <i>login</i> dengan benar | <i>User</i> tidak akan berhasil masuk ke sistem | Berhasil |
| <i>User</i> masuk kehalaman yang bukan haknya | <i>User</i> tidak akan masuk ke dalam halaman sistem | Berhasil |
| Tampilan menu sesuai dengan hak akses <i>user</i> | Menu muncul sesuai dengan hak akses yang dimiliki <i>user</i> | Berhasil |

IV. PENUTUP

Kesimpulan dari penelitian ini adalah :

1. Berdasarkan hasil pengujian fungsional menggunakan metode *Black Box Testing* tidak ditemukan adanya kesalahan pada sistem, maka *DDDP Application System* yang dikembangkan pada penelitian ini telah layak digunakan oleh tim divisi *Individu Fundraiser* dalam membantu pengelolaan data supporter.
2. Dengan adanya dokumen desain rancangan sistem yang telah dibuat akan membantu dan memudahkan dalam pengembangan sistem informasi *DDDP* kedepannya baik oleh peneliti sendiri ataupun pihak lain yang berkepentingan dalam pengembangan sistem.

Penelitian pengembangan sistem informasi *DDDP* ini belum sepenuhnya sempurna masih bisa dikembangkan kembali dengan menambahkan modul-modul yang baru

ataupun yang lainnya guna mendukung kinerja tim *Individu Fundraiser*.

DAFTAR PUSTAKA

- [1] C. Sundari, "Revolusi Industri 4.0 Merupakan Peluang Dan Tantangan Bisnis Bagi Generasi Milenial Di Indonesia," *Pros. Semin. Nas. DAN CALL Pap.*, no. Fintech dan E-Commerce untuk Mendorong Pertumbuhan UMKM dan Industri Kreatif, pp. 555–563, 2019.
- [2] H. Hamdan, "Industri 4.0: Pengaruh Revolusi Industri Pada Kewirausahaan Demi Kemandirian Ekonomi," *J. Nusant. Apl. Manaj. Bisnis*, vol. 3, no. 2, p. 1, 2018, doi: 10.29407/nusamba.v3i2.12142.
- [3] F. from the Heart, "FAQ Direct Debit Donor Programme," *foodfromtheheart.sg*, 2019. <https://www.foodfromtheheart.sg/faq/ddp> (accessed Jan. 12, 2020).
- [4] R. S. Pressman, *The Incremental Model in Software Engineering, A Practitioner's Approach*. New York: McGraw-Hill series in computer science, 2001.
- [5] A. Singh, "A Simulation Model For Incremental Software Development Life Cycle Model," *Int. J. Adv. Res. Comput. Sci.*, 2017, doi: 10.26483/ijarcs.v8i7.4136.
- [6] I. Irawati, S. Salju, and H. Hapid, "Pengaruh Penggunaan Sistem Informasi Manajemen Terhadap Kualitas Laporan Keuangan Pada Pt. Telkom Kota Palopo," *J. Manaj. STIE Muhammadiyah Palopo*, vol. 3, no. 2, pp. 6–12, 2019, doi: 10.35906/jm001.v3i2.302.
- [7] M. Maryanto, M. Maisyaroh, and B. Santoso, "Metode Internet Protocol Security (IPSec) Dengan Virtual Private Network (VPN) Untuk Komunikasi Data," *PIKSEL Penelit. Ilmu Komput. Sist. Embed. Log.*, vol. 6, no. 2, pp. 179–188, 2018, doi: 10.33558/piksel.v6i2.1508.
- [8] I. Warman And R. Ramdaniansyah, "Analisis Perbandingan Kinerja Query Database Management System (Dbms) Antara Mysql 5.7.16 Dan Mariadb 10.1," *J. Teknoif*, vol. 6, no. 1, pp. 32–41, 2018, doi: 10.21063/jtif.2018.v6.1.32-41.
- [9] A. Kurnianti, Angguningtyas, and R. G. Isnanda, "Perancangan Database Pada Sistem Asessmen Dan Pemetaan Hasil Asessmen Berbasis Tag Sebagai Pembantu Penyusunan Strategi Pembelajaran," *Semesta Tek.*, vol. 20, no. 2, pp. 106–115, 2017.
- [10] A. Hendini, "Pemodelan Uml Sistem Informasi Monitoring Penjualan Dan Stok Barang," *J. Khatulistiwa Inform.*, vol. 2, no. 9, pp. 107–116, 2016, doi: 10.1017/CBO9781107415324.004.
- [11] B. Dobing and J. Parsons, "How UML is used," *Commun. ACM*, vol. 49, no. 5, pp. 109–113, 2006, doi: 10.1145/1125944.1125949.
- [12] Munawar, *Analisis Perancangan Sistem Berorientasi Objek dengan UML (Unified Modeling Language)*. Bandung: Informatika Bandung, 2018.
- [13] A. Firman, H. F. Wowor, X. Najooan, J. Teknik, E. Fakultas, and T. Unsrat, "Sistem Informasi Perpustakaan Online Berbasis Web," *E-Journal Tek. Elektro Dan Komput.*, vol. 5, no. 2, pp. 29–36, 2016.
- [14] A. Bari and K. Kasmawi, "Sistem Informasi Manajemen Inventory Secara Online Menggunakan Framework Easyui," *INOVTEK Polbeng - Seri Inform.*, vol. 1, no. 1, p. 78, 2016, doi: 10.35314/isi.v1i1.141.
- [15] L. J. Siagian, *Otomatisasi Pengujian Perangkat Lunak (Software Test Automation)*. Yogyakarta: Deepublish, 2018.
- [16] W. N. Cholifah, Y. Yulianingsih, and S. M. Sagita, "Pengujian Black Box Testing pada Aplikasi Action & Strategy Berbasis Android dengan Teknologi Phonegap," *STRING (Satuan Tulisan Ris. dan Inov. Teknol.)*, vol. 3, no. 2, p. 206, 2018, doi: 10.30998/string.v3i2.3048.