

# Perancangan *Website* Sebagai Sarana untuk Mempertemukan *Supplier* dengan *Dropshipper*

Taufik Hidayat<sup>1</sup>, Yulindon<sup>2</sup>, Rahmat Hidayat<sup>3</sup>

<sup>1,3</sup> Teknologi Informasi, Politeknik Negeri Padang, Padang, Indonesia  
taufik1799hidayat@gmail.com

rahmat@pnp.ac.id

<sup>2</sup> Teknik Elektro, Politeknik Negeri Padang, Padang, Indonesia  
yulindon@pnp.ac.id

Diterima 2 Desember 2019

Disetujui 15 Juni 2020

**Abstract**—People use their loaf to gain benefit in Industrial Revolution 4.0. One of which is to sell stuffs by online. The advantage of selling online is lowerrunning costs. Dropship is one of categorizes in online business. It defines as a reseller or retailer who sells stuffs without storing . Many suppliers find it difficult to obtain a dropshipper. So, to solve this problem, one needs to obtain a website. The programming language is involved in website making such as PHP and MySQL as DBMS. In addition, it also needs a model to design software; one of the software development models is Waterfall. It is a sequential or consecutive model from analysis to maintenance stage.

**Index Terms**—CodeIgniter, Dropshipping, PHP, Website

## I. PENDAHULUAN

Media sosial umumnya digunakan sebagai tempat untuk berkomunikasi atau tempat menyampaikan informasi. Saat ini media sosial dimanfaatkan oleh sebagian orang untuk menghasilkan uang, dengan cara berjualan. Mereka berjualan hanya bermodal foto tanpa memiliki stok barang atau disebut juga dengan *dropshipping*.

Sebagian toko ingin memberikan layanan *dropshipping*, tetapi masih belum memiliki *website* sehingga menyulitkan mereka dalam mencari *dropshipper*. Beberapa toko memanfaatkan Facebook untuk mencari *dropshipper*, cara ini masih kurang efektif, karena tidak semua orang menerima informasi tersebut.

Untuk *website* yang mempertemukan antara *supplier* dengan *dropshipper* khususnya Indonesia masih belum ada. Yang ada hanya satu *website* untuk satu PT, contoh dropshipaja.com *website* ini mencari sendiri *dropshipper* mereka untuk menjual produk-produk yang dimiliki oleh PT tersebut. Dan masih banyak lagi *website-website* lain yang merekrut *dropshipper* dengan cara ini.

Untuk mengatasi persolan yang telah dipaparkan sebelumnya, *paper* ini akan memberikan solusi yaitu dengan memanfaatkan *website* sebagai sarana untuk mempertemukan pemilik toko dengan *dropshipper*. Paper ini hanya memberikan rancangan sistem yang akan dibuat. Terdiri dari *use case diagram*, *activity diagram*, kebutuhan *fungsi*ional dan *non-fungsi*ional serta desain *user interface*.

## II. LANDASAN TEORI

### A. Website

*Website* adalah sebuah perangkat lunak yang berisi dokumen-dokumen multimedia (gambar, teks, video, animasi dan suara) di dalamnya, yang menggunakan protokol HTTP (*Hypertext Transfer Protocol*) dan untuk mengaksesnya diperlukan sebuah aplikasi yang disebut *browser* [1].

### B. PHP

PHP atau disebut juga dengan *Hypertext Preprocessor* adalah bahasa scripting *server-side* untuk membuat *website* dinamis. Beberapa kelebihan PHP dibandingkan dengan bahasa pemrograman yang lain adalah mudah dipelajari dan dapat berjalan di sistem operasi apapun. PHP juga termasuk bahasa pemrograman yang populer karena memiliki fungsi built-in lengkap dan proses eksekusi cepat [2]. Maksud dari *server-side* adalah semua proses dilakukan di dalam *server* sebelum dikirimkan ke *browser* pengguna. Hasil yang diterima oleh pengguna adalah file yang berformat HTML [3].

### C. MySQL

MySQL adalah *database server* yang berfungsi untuk mengelola data sebelum dikirimkan ke pengguna. Oleh karena pemrosesan terjadi di *server*, maka pengguna dapat mengakses *database* dimana saja dan kapan saja dengan syarat komputer terhubung ke *server* [4].

#### D. CodeIgniter

CodeIgniter adalah salah satu *framework* PHP yang memiliki ukuran yang kecil sehingga membuat waktu eksekusinya menjadi lebih cepat dibandingkan dengan *framework* lain. CodeIgniter juga menggunakan konsep MVC (*Model*, *View* dan *Controller*) untuk mempermudah dan mempercepat dalam pembuatan *website* [5].

#### E. Dropshipping

*Dropshipping* atau disebut juga dengan *dropship*, merupakan salah satu bentuk jual beli dimana *reseller* atau *retailer* (pengecer) melakukan penjualan tanpa menyetok barang terlebih dahulu. Maksudnya yaitu *dropshipper* (orang yang melakukan *dropshipping*) mempromosikan suatu produk melalui sosial media atau *marketplace* seperti tokopedia atau bukalapak dengan bermodalkan foto. Jika ada penjualan maka uang yang diterima oleh *dropshipper* akan diteruskan ke *supplier*, kemudian barang dikirim oleh *supplier* ke pembeli dengan nama *dropshipper*. Keuntungan dari *dropship* yaitu tidak perlu stok barang, dan harga produk ditentukan oleh *dropshipper*.

#### F. UML

UML (*Unified Modeling Language*) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. UML terdiri dari diagram dan teks-teks [6].

#### G. Use Case Diagram

*Use case diagram* merupakan pemodelan yang menggambarkan kelakuan sistem yang akan dibuat. Fungsi dari *use case diagram* adalah untuk mengetahui fungsi apa saja yang terdapat dalam sistem dan siapa saja yang berhak untuk mengaksesnya [6].

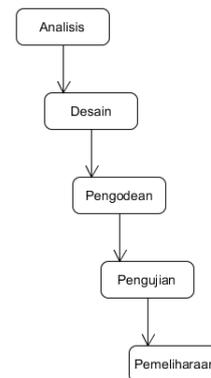
#### H. Activity Diagram

*Activity diagram* atau diagram aktivitas berfungsi untuk menggambarkan aliran kerja dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. *Activity diagram* hanya menggambarkan aktivitas sistem bukan aktivitas *actor* [6].

### III. METODE PENELITIAN

Untuk metode perancangan system perangkat lunak, penulis menggunakan metode *waterfall*. Metode *waterfall* atau disebut juga dengan *sekuensial linear* merupakan metode pengembangan perangkat lunak yang bersifat *sekuensial* atau terurut dimulai dari analisis, desain, pengodean, pengujian dan tahap pemeliharaan [6]. Sifat dari metode *waterfall* yaitu *sekuensial* maksudnya adalah proses pada tahap kedua tidak bisa dilakukan sebelum proses tahap satu selesai, sehingga metode ini sangat mudah untuk dipahami.

Berikut adalah gambar metode *waterfall*:

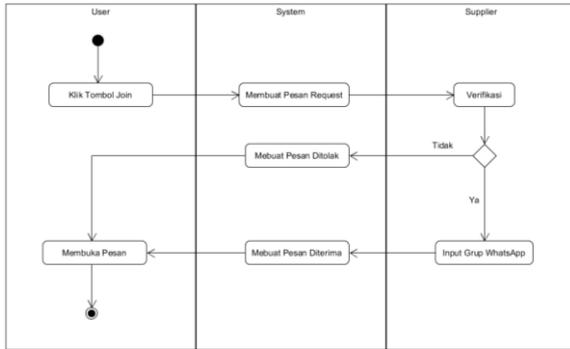


Gambar 1. Metode *waterfall*

1. Analisis kebutuhan perangkat lunak  
Aktivitas analisis kebutuhan perangkat lunak bertujuan untuk memahami perangkat lunak seperti apa yang dibutuhkan oleh pengguna. Hasil dari analisis kebutuhan perangkat lunak, nantinya akan didokumentasikan [6].
2. Desain  
Desain perangkat lunak adalah proses merubah kebutuhan perangkat lunak yang telah didapat sebelumnya menjadi sebuah desain agar dapat diimplementasikan menjadi program. Tahap ini juga perlu didokumentasikan [6].
3. Pembuatan kode program  
Tahap ini merupakan tahapan merubah desain perangkat lunak menjadi kode program, sesuai dengan apa yang telah didesain sebelumnya [6].
4. Pengujian  
Tujuan dari pengujian adalah untuk memastikan bahwa *output* yang dihasilkan oleh program sesuai dengan apa yang diharapkan serta untuk mengurangi terjadinya kesalahan atau *error* [6].
5. Pemeliharaan  
Terkadang program mengalami permasalahan setelah diterima oleh pengguna atau tidak terdeteksi pada saat pengujian. Itulah alasan kenapa ada tahapan pemeliharaan, tujuannya adalah untuk memperbaiki jika terjadi kesalahan ketika perangkat lunak telah diterima oleh pengguna [6].

IV. HASIL DAN PEMBAHASAN

A. Desain Sistem



Gambar 2. Desain sistem

Berdasarkan Gambar 2 terlihat bahwa sistem ini dimulai ketika *user* menekan tombol join yang ada di halaman cari *supplier*, kemudian sistem mengirimkan pesan kepada *supplier* yang dituju yang berisikan informasi tentang *user* tersebut. Selanjutnya *supplier* melakukan verifikasi berdasarkan informasi yang dikirim oleh sistem. Jika *user* tersebut diterima maka *supplier* harus mengisi *link* grup whatsapp, yang digunakan untuk mengumpulkan para *dropshipper*. Kemudian sistem mengirimkan pesan kepada *user* bahwa ia diterima sebagai *dropshipper* di toko tersebut. Jika *user* yang mendaftar tadi tidak diterima maka sistem akan mengirimkan pesan yang berisikan informasi penolakan.

B. Kebutuhan Fungsional

Kebutuhan *fungsional* adalah kebutuhan yang berisi layanan apa saja yang terdapat pada sistem. Untuk mengetahui kebutuhan *fungsional* dari sistem yang penulis rancang bisa dilihat pada Tabel 1 dan Tabel 2. Disini terdapat dua kebutuhan *fungsional* yaitu untuk pengguna dan *supplier*.

Tabel 1. Kebutuhan *fungsional* pengguna

NO	Kode Fungsi	Nama Fungsi	Deskripsi
1	SIDO-P-001	Login	Fungsi yang digunakan oleh pengguna untuk masuk ke sistem.
2	SIDO-P-002	Blog	Fungsi yang digunakan oleh pengguna untuk melihat artikel yang berhubungan dengan <i>dropship</i> .
3	SIDO-P-003	Informasi Pencarian Dropshipper	Fungsi yang digunakan oleh pengguna untuk melihat informasi toko-toko mana saja yang sedang mencari <i>dropshipper</i> .
4	SIDO-P-004	Join	Fungsi yang digunakan oleh pengguna untuk bergabung menjadi

			anggota <i>dropshipper</i> disebuah toko.
--	--	--	---

Tabel 2. Kebutuhan *fungsional* supplier

No	Kode Fungsi	Nama Fungsi	Deskripsi
1	SIDO-S-001	Login	Fungsi yang digunakan oleh <i>supplier</i> untuk masuk ke dalam sistem. Tujuannya supaya <i>supplier</i> bisa membuat posting pencarian <i>dropshipper</i> .
2	SIDO-S-002	Blog	Fungsi yang digunakan oleh <i>supplier</i> untuk melihat artikel yang berhubungan dengan <i>dropship</i> .
3	SIDO-S-003	Informasi Pencarian Dropshipper	Fungsi yang digunakan oleh <i>supplier</i> untuk melihat toko-toko mana saja yang sedang membuka lowongan <i>dropshipper</i> .
4	SIDO-S-004	Rekrut Dropshipper	Fungsi yang digunakan oleh <i>supplier</i> untuk merekrut <i>dropshipper</i> .
5	SIDO-S-005	Posting Pencarian Dropshipper	Fungsi yang digunakan oleh <i>supplier</i> untuk memberikan informasi, bahwa mereka sedang mencari <i>dropshipper</i> .

C. Kebutuhan Non-Fungsional

Kebutuhan *non-fungsional* adalah kebutuhan yang berisi perilaku yang terdapat pada sistem. Untuk mengetahui kebutuhan *non-fungsional* dari sistem yang penulis rancang bisa dilihat pada Tabel 3.

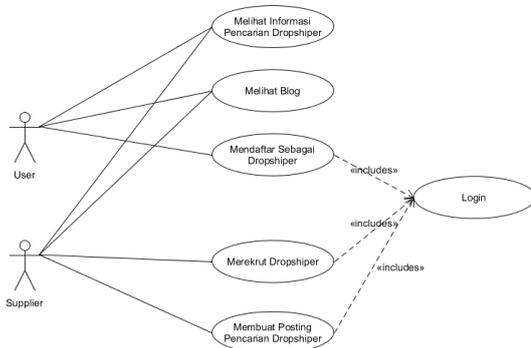
Tabel 3. Kebutuhan non-fungsional

No	Kode Fungsi	Nama Fungsi	Deskripsi
1	SIDO-001	Portability	Sistem dapat dijalankan di sistem operasi apapun, baik itu windows maupun linux. Serta dapat diakses menggunakan berbagai <i>browser</i> seperti mozilla atau google chrome
2	SIDO-002	Availability	Sistem dapat diakses kapan saja, dimana saja dan oleh siapa saja baik itu <i>user</i> maupun <i>reseller</i> , dengan syarat terkoneksi dengan internet.
3	SIDO-003	Realibility	Sistem bisa berkerja 24 jam secara penuh dan bisa diupdate kapan saja.
4	SIDO-004	Security	Semua data yang ada dapat terjaga dengan baik.

D. Use Case Diagram

Pada Gambar 3 terlihat bahwa sistem ini memiliki 2 jenis pengguna yaitu *user* atau pengguna biasa dan *supplier*. *User* bisa melihat blog, melihat informasi pencarian *dropshipper*, dan untuk mendaftar sebagai *dropshipper*, *user* harus *login* terlebih dahulu.

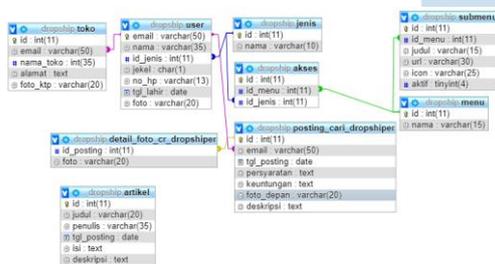
Sedangkan *supplier* bisa melihat blog, melihat informasi pencarian *dropshipper*, untuk merekrut *dropshipper* serta membuat postingan pencarian *dropshipper*, *supplier* harus *login* terlebih dahulu.



Gambar 3. Use case sistem

E. Desain Database

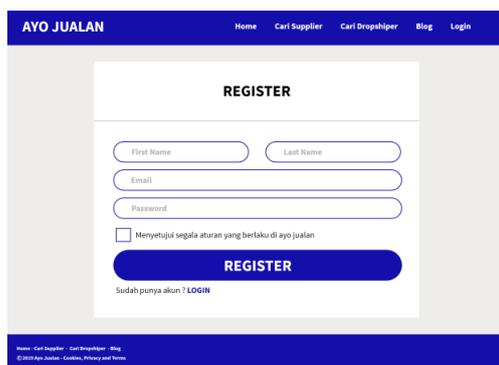
Pada Gambar 4 terlihat bahwa sistem yang akan dibuat terdiri dari beberapa tabel yang saling berelasi.



Gambar 4. Desain database

F. Rancangan Tampilan

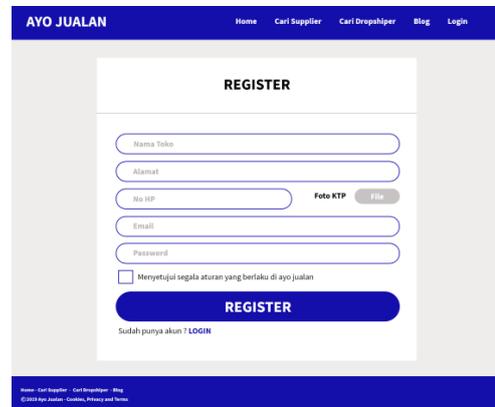
F.1 Halaman Pendaftaran Pengguna



Gambar 5. Halaman pendaftaran pengguna

Gambar 5 merupakan halaman pendaftaran yang digunakan oleh pengguna untuk mendaftar. Terdiri dari kolom nama lengkap, *email*, dan *password* yang harus diisi untuk bisa mengakses sistem.

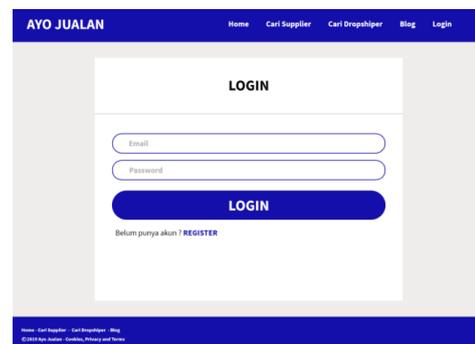
F.2 Halaman Pendaftaran Supplier



Gambar 6. Halaman pendaftaran supplier

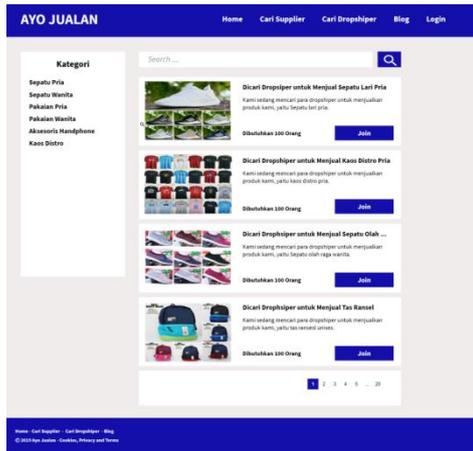
Untuk tampilan halaman pendaftaran *supplier* bisa dilihat pada Gambar 6. Formnya sama seperti form pendaftaran *user*, tetapi yang membedakan pada halaman pendaftaran *supplier* ada kolom untuk pengisian nama, alamat dan nomor telepon toko. Disini *supplier* juga wajib mengirimkan foto KTP.

F.3 Halaman Login

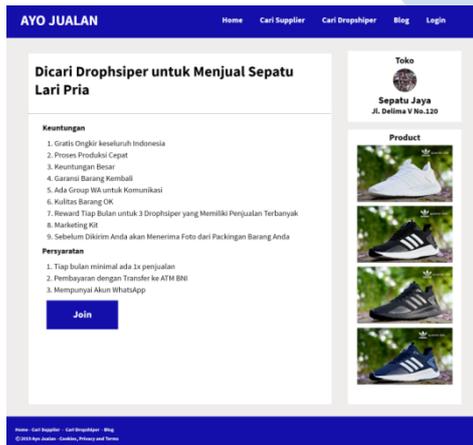


Gambar 7. Halaman login

Halaman *login* digunakan untuk masuk ke sistem. Berbeda dengan halaman pendaftaran yang dibagi dua, satu untuk *supplier* dan satu lagi untuk pengguna. Sedangkan halaman *login* digunakan oleh pengguna maupun *supplier*. Untuk tampilan halaman *login* bisa dilihat pada Gambar 7.

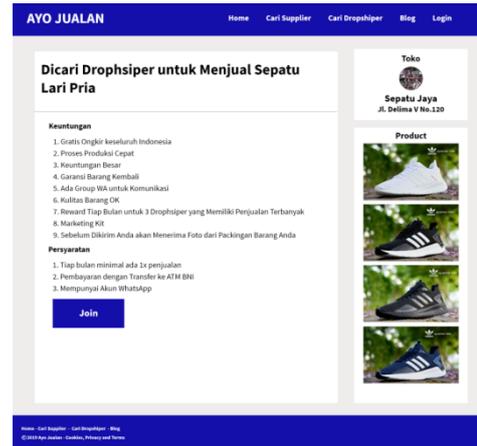
F.4 Halaman Cari *Supplier*Gambar 8. Halaman cari *supplier*

Untuk tampilan dari halaman cari *supplier* bisa dilihat pada Gambar 8. Halaman cari *supplier* digunakan oleh pengguna untuk mendaftar menjadi *dropshipper* di sebuah toko atau sekedar mencari informasi.

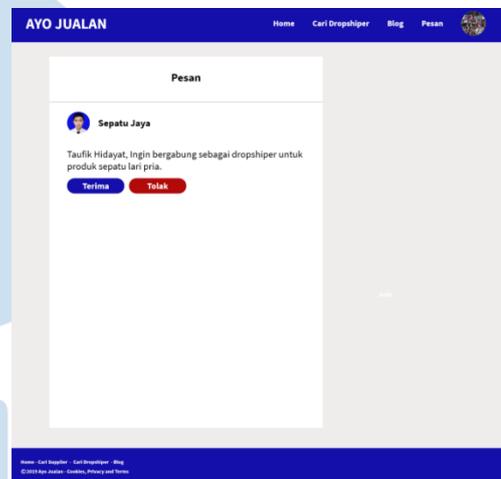
F.5 Halaman Detail Cari *Supplier*Gambar 9. Halaman *detail* cari *supplier*

Halaman *detail* cari *supplier* muncul ketika judul yang ada di halaman cari *supplier* diklik. Coba perhatikan Gambar 9, di sana terdapat judul, nama toko, tombol join serta produk-produk yang akan mereka tawarkan. Tombol join berfungsi untuk mendaftar menjadi *dropshipper* di toko tersebut.

## F.6 Halaman Pesan



Gambar 10. Halaman pesan pengguna

Gambar 11. Halaman pesan *supplier*

Halaman pesan digunakan oleh pengguna maupun *supplier*, namun memiliki tampilan dan fungsi yang berbeda. seperti Gambar 10, merupakan halaman pesan untuk pengguna. Halaman pesan pengguna berisi informasi penolakan atau diterima menjadi *dropshipper* di sebuah toko. Sedangkan Gambar 11, merupakan halaman pesan untuk *supplier*. Sedikit berbeda dengan halaman pesan pengguna, di sini terdapat tombol terima atau tolak, yang berfungsi untuk menerima atau menolak lamaran *dropshipper*. Jika diterima, maka akan tampil sebuah kolom untuk pengisian alamat grup WhatsApp.

### F.7 Halaman Posting Pencarian *Dropshipper*

Gambar 12. Halaman *posting* pencarian *dropshipper*

Halaman *posting* pencarian *dropshipper*, digunakan oleh *supplier* untuk mencari *dropshipper* dengan cara membuat semacam artikel supaya pengguna tertarik untuk mendaftar. Pada Gambar 12 terdapat beberapa kolom yang harus diisi seperti, persyaratan, keuntungan yang didapat oleh *dropshipper* serta foto-foto barang yang akan dijual.

### F.8 Halaman Blog

Gambar 13. Halaman blog

Gambar 13, memperlihatkan tampilan dari halaman blog. Pada halaman blog terdapat artikel-artikel yang berisi informasi yang berhubungan dengan *dropshipping*.

## V. SIMPULAN

### A. Kesimpulan

Berdasarkan hasil perancangan *website* sebagai media untuk mempertemukan *supplier* dengan *dropshipper*, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Sistem yang dirancang hanya sebatas untuk mempertemukan antara *supplier* dengan *dropshipper*. Sehingga untuk manajemen *dropshipper* diserahkan sepenuhnya kepada *supplier* namun kami menyarankan untuk menggunakan grup WhatsApp.
2. Dalam pembuatan *website* diperlukan bahasa pemrograman, disini penulis memilih bahasa pemrograman PHP dan untuk pengelolaan *database* penulis menggunakan MySQL. Untuk mempercepat pembuatan *website* maka diperlukan *framework* atau kerangka kerja. *Framework* yang digunakan adalah CodeIgniter, karena memiliki waktu eksekusi yang cepat dibanding *framework* lain.

### B. Saran

1. Di masa mendatang, jika ada penelitian yang mengajukan judul penelitian yang sama, sangat disarankan untuk membuat sistem yang berbasis Android.
2. Jika *website* ini benar-benar dibuat, diharapkan untuk memperkuat bagian keamanan sistem karena ada beberapa data yang penting seperti foto KTP dari *supplier*.

## DAFTAR PUSTAKA

- [1] A. M. Rudiyanto, *Pemrograman Web Dinamis Menggunakan Php dan Mysql*. Yogyakarta: CV. Andi Offset, 2011.
- [2] Angga Wibowo, *16 Aplikasi PHP Gratis untuk Pengembangan Situs Web*. Yogyakarta: Andi, 2007.
- [3] V. Lusiana, "Sistem Informasi Promosi Batik Tradisional Semarang Berbasis Web," *J. Din. Inform.*, vol. 1, no. 1, pp. 26–35, 2009.
- [4] S. Haris, *Manajemen Database MySQL menggunakan MySQL-Front*. Jakarta, 2003.
- [5] Dedi, Triono, and W. Muhajiroh, "Perancangan Sistem E-Commerce Batik Sopiyan Hadi Berbasis Web," *Sisfotek Glob.*, vol. 7, no. 1, pp. 124–129, 2017.
- [6] A. Rosa and M. Shalahuddin, *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Bandung: Informatika, 2016.