Jurnal Integrasi
Vol. 13 No. 2, October 2021, 142-146
e-ISSN: 2548-9828

Article History
Received September 2021
Accepted October 2021

Sistem S*witch Monitoring* Berbasis Web

Sandi Prasetyaningsih[1], Ahmad Kussyairi[2],
[1]Politeknik Negeri Batam
Multimedia and Networking Engineering Program
Jln. Ahmad Yani, Batam Centre, Batam 29461, Indonesia
E-mail: sandi@polibatam.ac.id, kussyairi@gmail.com

Abstrak

Dokumentasi switch pada PT ABC masih dilakukan secara manual oleh departemen IT. Sistem dokumentasi data yang dilakukan secara manual ini dapat menimbulkan beberapa masalah, antara lain munculnya data berulang. Selain itu, monitoring yang dilakukan secara manual akan memakan waktu jika pihak tertentu ingin melakukan pencarian data. Metode *waterfall* digunakan dalam pengembangan sistem monitoring switch berbasis web. Ada lima tahapan yang digunakan dalam pengembangan dengan metode *waterfall*, yaitu analisis, desain, pengkodean, pengujian, dan penerapan program serta pemeliharaan. Pengujian dari sistem monitoring menggunakan metode *black box* yang dilakukan pada sistem dengan melihat spesifikasi fungsional tanpa menguji desain dan pengkodean. Hasil perancangan, implementasi, dan pengembangan, sistem *switch monitoring* berbasis web berhasil dibuat dengan memiliki beberapa fitur seperti menambah *switch, edit switch, hapus switch, detail switch*, dan manajemen *user.*

Kata kunci: *Switch*, *Monitoring*, Metode *Waterfall*, Metode *Black box*

Abstract

Switch documentation monitoring at PT ABC is executed manually by IT department. This current system occurs some problems such us data redundant. Moreover, manual documentation also takes time as well as it brings user spend more times when they are trying to look for certain data. *Waterfall* method is used to generate switch monitoring on website based. There are five steps that are using to develop the system, such as analyzing, designing coding, testing, and implementing code as well as maintaining. During the testing, black box method is utilizing to obtain certain specification of the system. The result of the design, implementation, dan evolvement of switch system monitoring by using web is successfully implemented. This system has several functions that can be used by user, namely add switch, edit switch, and delete switch, detail switch, and user management control.

Keywords: Integrasi, margin

1. Pendahuluan

PT ABC merupakan perusahaan yang bergerak dibidang semikonduktor pembuatan dan perakitan *Integrated Circuit* (IC). Perusahaan ini didirikan pada tahun 2002 dan merupakan cabang dari perusahaan PT ABC yang berada di Munich, Jerman. Departemen IT pada perusahaan ini terdiri dari dua bagian, yaitu IT *Office* dan IT *Client Service*. Departemen IT *Office* merupakan departemen yang bertugas menangani masalah data *center*, jaringan dan perangkat jaringan lainnya.

Dalam melakukan proses bisnisnya, departemen IT berhadapan dengan beberapa kendala. Salah satunya adalah pengolahan data *monitoring switch* dan pelaporan data *switch* yang masih dilakukan secara manual. Pendataan dilakukan dengan mencatat data laporan dari label *switch*, ip, dan *port* yang digunakan pada suatu *device* menggunakan *Microsoft Excel*, sehingga dalam melakukan pendataan sering mengalami terjadinya *redundancy* data (data ganda). Hal ini tentu saja akan menimbulkan masalah-masalah lain jika terus dibiarkan. Salah satu akibat dari data yang teredudansi adalah data yang ada menjadi tidak konsisten [1]. Selain itu, sistem pengolahan data yang masih manual akan membuat sistem *monitoring* tidak efektif dan efisien. Sebagai contoh, jika diperlukan data dalam waktu cepat, hanya pihak yang menyimpan data yang dapat melihat sedangkan kondisi yang terjadi dalam keadaa darurat.

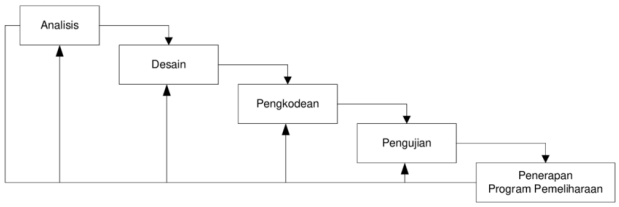
Berdasarkan permasalahan yang telah disebutkan di atas, penelitian ini akan fokus pada perancangan *monitoring* data *switch* berbasis web.

2. Metode Penelitian

Dalam melakukan penelitian, metode yang digunakan untuk perancangan sistem adalah metode SDLC; *waterfall*. Pada metode perancangan *waterfall*, tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Sebagai contoh tahap desain harus menunggu selesainya tahap sebelumnya yaitu tahap *requirement* [2]. Metode ini menyediakan beberapa pendekatan alur hidup perangkat lunak secara berurutan dimulai dari proses analisa, desain, pengkodean, pengujian, dan penerapan program pemeliharaan [3].

Menurut [4], metode *waterfall* memiliki beberapa kelebihan dibandingan metode pengembangan perangkat lunak lainnya, diantara lain:

1. Kualitas produk yang dihasilkan lebih baik karena tahapan eksekusinya dilakukan secara bertahap.
2. Proses pengembangan dilakukan dalam fase satu per satu sehingga dapat meminimalis terjadinya kesalahan.
3. Dokumen pengembangan sistem terorganisir karena setiap Langkah harus diselesaikan terlebih dahulu sebelum masuk keproses berikutnya.



Gambar 1: Ilustrasi Metode *Waterfall*

Gambar 1 merupakan ilustrasi dari tahapan metode *waterfall*. Setiap prosesnya memiliki peran yang sangat penting dalam pembuatan sebuah perangkat lunak.

A. Analisa Perangkat Lunak

Pada tahan ini, pembuat sistem harus sudah melakukan identifikasi kebutuhan perangkat lunak yang dibutuhkan untuk pengembangan dan *user.*

B. Desain

Desain perangkat lunak fokus pada desain pembuatan program yang meliputi deskripsi umum sistem, perancangan sistem, dan rancangan antarmuka. Pada proses desain ini, pengembang mencoba mengkonversi kebutuhan analisis perangkat lunak menjadi sebuah program yang akan digunakan pada tahap selanjutnya.

C. Pengkodean

Setelah melakukan identifikasi desain perangkat lunak, proses yang dilakukan selanjutnya adalah mengubah desain menjadi bahasa pemograman. Dalam proses pengkodean akan banyak ditemukan *trial* dan *error* karena penyesuaian program dan kebutuhan perangkat lunak.

C. Pengujian

Untuk memastikan apakah program yang dibuat sudah sesuai dengan kebutuhan perangkat lunak atau belum, dibutuhkan sebuah proses pengujian untuk semua fungsi yang terdapat pada program. Proses pengujian ini biasanya akan dilakukan lebih dari satu kali sampai dengan program dapat berjalan sesuai kebutuhan.

Pengujian pada sebuah perangkat lunak sangatlah penting. Berdasarkan [5], ada beberapa tujuan dari pengujian perangkat lunak, yaitu:

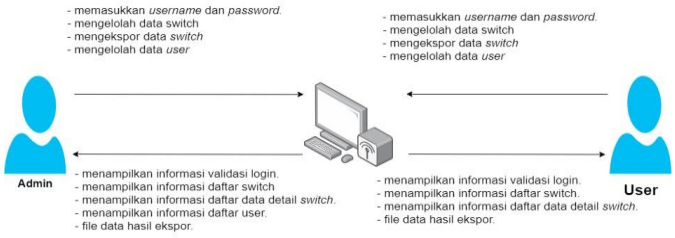
1. Kesalahan pada perangkat lunak dapat dilihat pada saat program dijalankan.
2. Pengujian dilakukan untuk mengetahui kesalahan yang belum teridentifikasi.
3. Pengujian dikatakan berhasil jika dapat memunculkan kesalahan yang belum teridentifikasi.

D. Penerapan program dan pemeliharaan

Ketika sudah digunakan oleh *user, error* mungkin saja akan ditemukan karena variasi *user* dan tidak terdeteksi pada saat pengujian. Proses pemeliharaan perangkat lunak dapat dimulai dari proses analisa; dilakukan beberapa perubahan tapi tidak untuk membuat sebuah perangkat lunak yang baru.

3. Hasil dan Pembahasan

3.1 Deskripsi Umum Sistem

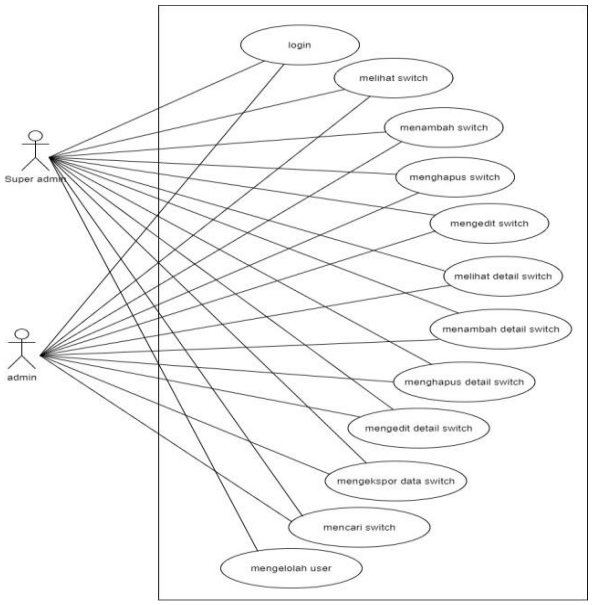


Gambar 2: Deskripsi Umum Sistem

Gambar 2 menjelaskan tentang sistem *network port monitoring* menangani pengelolaan data *switch*. Terdapat dua level *user,* yaitu *admin* dan *user*. *Admin* memiliki hak akses dalam mengelolah seluruh data yang meliputi data *switch*, dan data *user*. Sedangkan *user* hanya dapat mengelolah data *switch*. *Admin* dan *user* dapat melakukan ekspor file ke dalam bentuk

excel. Sistem ini hanya dikelolah oleh *admin* dan *user* dalam me*monitoring* data *switch*.

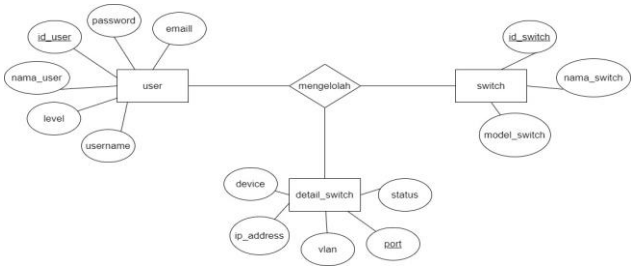
3.2 Perancangan Sistem



Gambar 3: Perancangan Sistem

Sistem networking monitoring berbasis web pada PT ABC memiliki dua *user* seperti yang disajikan pada gambar 3. *Super admin* harus melakukan *login* terlebih dahulu agar dapat menjalankan beberapa fungsi seperti melihat *switch*, menambah *switch*, menghapus *switch*, mengedit *switch*, melihat detail *switch*, menambah detail *switch*, menghapus detail *switch*, menghapus detail *switch*, mengedit detail *switch*, mengekspor detail *switch*, mencari *switch*, dan mengelola *user*. *Admin* memiliki fitur yang hamper sama dengan *super admin* kecuali tidak bisa melakukan pengelolaan *user*.

3.3 Perancangan Basis Data



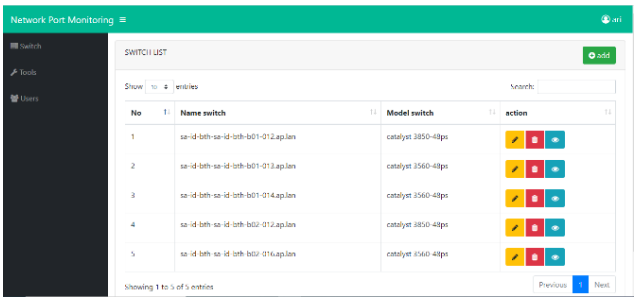
Gambar 4: Perancangan Basis Data

Pada perancangan basis data yang disajikan pada gambar 4, ada tiga entitas yang berbeda, yaitu *user*, detail_switch dan *switch*. Masing-masing entitas memiliki *primary key* yang berbeda, yaitu id_user, id_switch, dan port.

3.4 User Interface

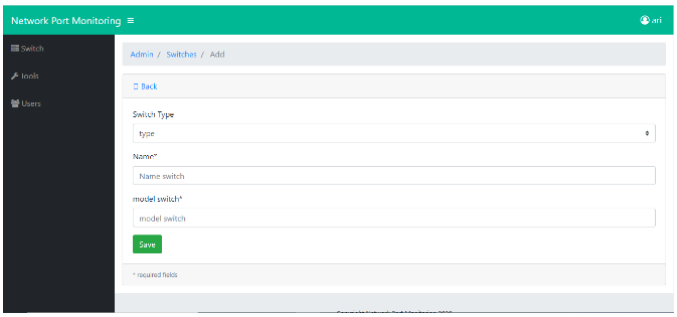
Gambar 5: Halaman Login

Pada halaman *login*, *admin* dan *user* menggunakan halaman yang sama. Pengguna harus memasukkan *username* dan *password* yang sebelumnya sudah didaftarkan ke sistem. Kemudian menekan *sign in* untuk masuk kesistem. Desain antarmuka *login* dapat dilihat pada gambar 5.



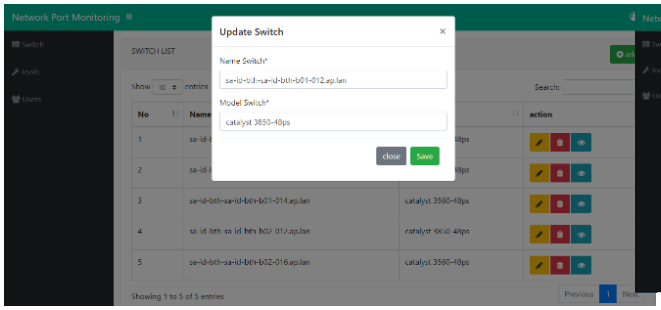
Gambar 6: Halaman List Switch

Pada halaman *list switch*, pengguna baik itu *admin* ataupun *user* dapat melihat data-data dari *switch* yang data halamannya berisikan nama *switch*, *model switch*, dan beberapa *action*. Desain antarmuka *list switch* dapat dilihat pada gambar 6.



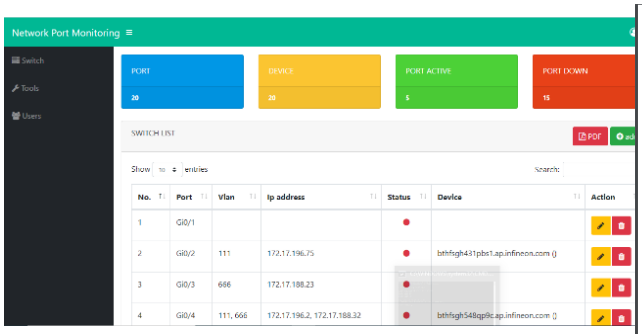
Gambar 7: Halaman Tambah Switch

Pada halaman tambah *switch* pengguna baik itu *admin* ataupun *user* dapat menambahkan data switch dengan memasukkan nama *switch* dan model *switch*. Desain antarmuka dapat dilihat pada gambar 7.



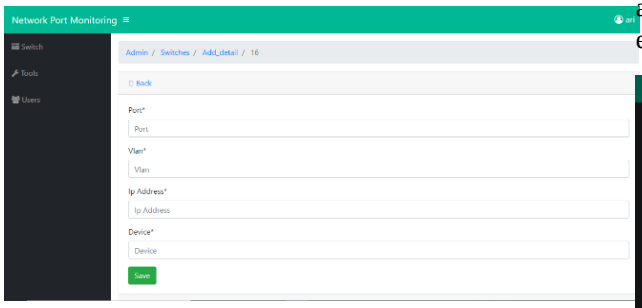
Gambar 8: Halaman Edit Switch

Pada halaman edit *switch*, pengguna baik itu *admin* ataupun *user* dapat melalukan *update* data pada data pada *list switch* dengan memasukkan data baru. Desain antarmuka halaman *edit switch* dapat dilihat pada gambar 8.



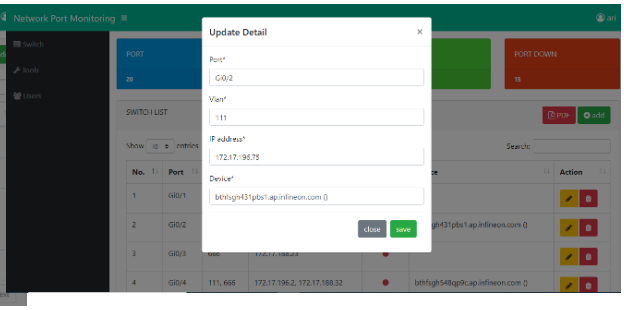
Gambar 9: Halaman Detail Switch

Pada halaman detail *switch*, pengguna baik itu *admin* ataupun *user* dapat melihat data *port* dari *switch*. halaman yang berisikan informasi dari *switch* yang berisikan beberapa *action*. Desain antarmuka dapat dilihat pada gambar 9.



Gambar 10: Halaman Tambah Detail Switch

Halaman tambah detail *switch, pengguna* baik itu *admin* ataupun *user* dapat menambah data detail *switch* dengan memasukkan form data pada halaman detail *switch.* Desain antarmuka halaman tambah detail *switch* dapat dilihat pada gambar 10.



Gambar 11: Halaman Edit Detail Switch

Pada halaman edit detail *switch*, pengguna baik itu *admin* ataupun *user* dapat meng-*update* data *port switch* dengan memasukkan data baru dan menyimpannya. Desain antarmuka halaman *edit detail switch* dapat dilihat pada gambar 11.

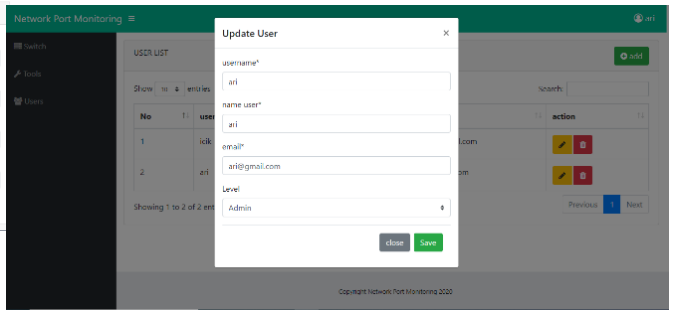
Nama Switch : sa-id-bth-sa-id-bth-b01-012.ap.lan

Model Switch : catalyst 3850-48ps

No	Port	Vlan	Ip Address	status	device
1	Gi0/1			tidak terhubung	
2	Gi0/2	111	172.17.196.75	tidak terhubung	bthfsgh431pbs1.ap.infineon.com ()
3	Gi0/3	666	172.17.188.23	tidak terhubung	
4	Gi0/4	111, 666	172.17.196.2, 172.17.188.32	tidak terhubung	bthfsgh548qp9c.ap.infineon.com ()
5	Gi0/5			tidak terhubung	
6	Gi0/6	111	172.17.196.52	tidak terhubung	bthemsps010b.ifxcep.net ()
7	Gi0/7	111	172.17.196.29	tidak terhubung	bthfsgh139qj09.ap.infineon.com ()
8	Gi0/8	111	172.17.196.51	terhubung	bthfsgh520rhqb.ap.infineon.com ()
9	Gi0/9	185	172.17.185.20	terhubung	bthwsgh114rj5v.ap.infineon.com ()
10	Gi0/10	184, 666	172.17.184.9, 172.17.188.28	tidak terhubung	bthfsgh114rj5t.ap.infineon.com ()
11	Gi0/11			tidak terhubung	
12	Gi0/12	111		tidak terhubung	bthemsps010c.ifxcep.net ()
13	Gi0/13	185	172.17.185.28	tidak terhubung	bthwsgh139qj0k.ap.infineon.com ()
14	Gi0/14	111	172.17.196.81	terhubung	bthemsps011b.ifxcep.net ()
15	Gi0/15	111	172.17.196.31	tidak terhubung	bthfsgh215txxq.ap.infineon.com ()
16	Gi0/16	111	172.17.196.44	tidak terhubung	bthemsps012a.ifxcep.net ()
17	Gi0/17	185	172.17.185.36	tidak terhubung	bthemsps012b.ifxcep.net ()

Gambar 12: Ekspor Switch

Ekspor data dapat dilakukan oleh admin maupun *user.* Dengan menekan ikon pdf pada detail *switch* maka akan menyajikan data dalam bentuk pdf. Desain hasil ekspor detail *switch* dapat dilihat pada Gambar 12.



Gambar 13: Halaman Mengelola User

Pada halaman mengelola *user, admin* dapat meng-*update* langsung data dari *user* dengan memasukkan data dan menyimpannya. Desain antarmuka halaman edit *user* dapat dilihat pada gambar 13.

3.5 Pengujian Sistem

TABLE 1

PENGUJIAN SISTEM

No	Nama Proses	Skenario Uji	Target	Validasi
1	Login	Username dan *password* benar	*Login* berhasil dan diarahkan ke halaman beranda	✓
		Username dan *password* salah	*Login* gagal dan menampilkan pesan kesalahan	✓
2	Melihat data *switch*	Membuka menu *switch*	Data *switch* berhasil di tampilkan	✓
3	Menambah data *switch*	Mengisi semua form dengan benar	Data *switch* berhasil disimpan	✓
		Salah satu form tidak terisi	Data *switch* gagal disimpan, muncul pesan kesalahan	✓
		Data *switch* sama	Data gagal disimpan, muncul pesan kesalahan	✓
4	Mengedit data *switch*	Mengubah data pada form	Data *switch* berhasil disimpan	✓
5	Melihat data detail_*switch*	Aktor memilih menu *switch*	Data detail *switch* berhasil ditampilkan	✓
6	Menambah data detail_*switch*	Mengisi form dengan benar	Data detail *switch* berhasil disimpan	✓
		Mengisi data pada port yang sama	Data gagal disimpan, muncun pesan kesalahan	✓
7	Mengedit data detail_*switch*	Mengubah data pada form	Data detail *switch* berhasil disimpan	✓
8	Mengekspor data *switch*	Aktor memilih salah satu data *switch* -> membuka detail *switch*	Data *switch* di *export* ke pdf	✓
9	Mengedit data *user*	Mengubah data pada form dengan isian benar	Data *user* berhasil disimpan	✓
		Menguba data pada form dengan isian salah	Data *user* gagal disimpan, muncul pesan kesalahan	✓
		Menghapus salah satu data *user*	Data *user* berhasil di hapus	✓

Pengujian sistem dilakukan setelah proses desain dan pengkodean selesai. Pengujian dilakukan untuk mengecek apakah sistem yang dibuat sudah sesuai dengan yang dibutuhkan atau belum. Hal ini dapat dilihat dari *input* dari *user* dan *output* yang dihasilkan oleh sistem. Metode *black box* digunakan dalam pengujian ini karena metode *black box* dilakukan pada sistem dengan melihat spesifikasi fungsional tanpa menguji desain dan pengkodean [6].

4. Kesimpulan

Berdasarkan hasil perancangan dan impelementasi *sistem network monitoring* dapat diambil beberapa kesimpulan, yaitu:

1. Sistem *network monitoring* berbasis web sudah berhasil dirancang dan diimplementasikan dibuktikan dengan pengujian sistem dengan menggunakan metode *black box.*
2. Sistem *network monitoring* telah mengimplementasi beberapa fitur mulai dari menambah, edit dan hapus switch, detail switch, dan manajemen user.

Daftar Pustaka

[1] Suharyanto, C.E., Chandra, J.E. and Gunawan, F.E., 2017. Perancangan Sistem Informasi Penggajian Terintegrasi Berbasis Web (Studi Kasus di Rumah Sakit St. Elisabeth). Jurnal Nasional Teknologi dan Sistem Informasi, 3(2), pp.225-232.

[2] Bangun Sistem Informasi Manajemen Arsip Rumah Sakit Bedah Surabaya Berbasis Web. Jurnal Sistem Informasi. No. 2. Vol. 3. 2338-137X.

[3] Sukamto, Ariani Rosa dan Shalahuddin, Muhamad. 2018. Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek. Informatika, Bandung.

[4] Wahid, A.A., 2020. Analisis Metode *Waterfall* Untuk Pengembangan Sistem Informasi. Jurnal Ilmu-Ilmu Informatika Dan Manajemen STMIK, October, pp.1-5.

[5] Rouf, A., 2012. Pengujian perangkat lunak dengan menggunakan metode white box dan black box. HIMSYATECH, 8(1).

[6] Nur, H., 2019. Penggunaan Metode *Waterfall* Dalam Rancang Bangun Sistem Informasi Penjualan. Generation Journal, 3(1), pp.1-10.