

LAPORAN PRAKTIK INDUSTRI
IMPLEMENTASI APLIKASI
SERAH TERIMA DOKUMEN AKTIVASI BERBASIS
WEBSITE DI PT. INDONESIA COMNETS PLUS SURABAYA

Jl. Ketintang Baru No.1-3, Ketintang, Kec. Gayungan,
Kota Surabaya, Jawa Timur 60231
(Tanggal 25 Januari – 26 Februari 2021)



Oleh
M. Fakhruddin Budihardjo
NIM 18050974010

UNIVERSITAS NEGERI SURABAYA
FAKULTAS TEKNIK
JURUSAN TEKNIK INFORMATIKA
PRODI S1 PENDIDIKAN TEKNOLOGI INFORMASI
2021

LAPORAN PRAKTIK INDUSTRI
IMPLEMENTASI APLIKASI
SERAH TERIMA DOKUMEN AKTIVASI BERBASIS
WEBSITE DI PT. INDONESIA COMNETS PLUS SURABAYA

Jl. Ketintang Baru No.1-3, Ketintang, Kec. Gayungan,
Kota Surabaya, Jawa Timur 60231
(Tanggal 25 Januari – 26 Februari 2021)

Oleh
M. Fakhruddin Budihardjo
NIM 18050974010

UNIVERSITAS NEGERI SURABAYA
FAKULTAS TEKNIK
JURUSAN TEKNIK INFORMATIKA
PRODI S1 PENDIDIKAN TEKNOLOGI INFORMASI
2021

LEMBAR PERSETUJUAN

Laporan Praktik Industri / ~~Praktik Kerja Lapangan~~

Nama : M. Fakhruddin Budihardjo

NIM : 18050974010

Program Studi : S1 Pendidikan Teknologi Informasi

Judul : Implementasi Aplikasi Serah Terima Dokumen
Aktivasi Berbasis Website di PT. Indonesia
Comnets Plus Surabaya.

Telah memenuhi syarat untuk diuji dan diseminarkan sebagai
persyaratan lulus mata kuliah praktik industri.

Surabaya, ----
Menyetujui
Pembimbing / Penguji,

Prof. Dr. Ekohariadi, M.Pd.
NIP. 196004041987011001

LEMBAR PENGESAHAN

Laporan Praktik Industri/~~Praktik Kerja Lapangan~~ :

Judul : Implementasi Aplikasi Serah Terima Dokumen
Aktivasi Berbasis Website di PT. Indonesia
Comnets Plus Surabaya.

Nama Industri : PT. Indonesia Comnets Plus Surabaya.

Alamat Industri: Jl. Ketintang Baru No.1-3, Ketintang, Kec.
Gayungan, Kota Surabaya, Jawa Timur 60231

Yang dilaksanakan oleh mahasiswa Fakultas Teknik (FT)
Universitas Negeri Surabaya:

Nama : M. Fakhruddin Budihardjo

NIM : 18050974010

Program Studi : S1 Pendidikan Teknologi Informasi

Jurusan : Teknik Informatika

Telah diseminarkan / diuji dan dinyatakan lulus.

Surabaya,-----

Menyetujui
Penguji

Mengetahui/Menyetujui
Pembimbing Industri

Mengesahkan
Wakil Dekan I
Fakultas Teknik Unesa,

Menyetujui
Pembimbing/Penguji,

Drs. Edy Sulistiyo, M.Pd.
NIP. 196404201991031005

Prof. Dr. Ekohariadi, M.Pd.
NIP. 196004041987011001

KATA PENGANTAR

Puji syukur penulis haturkan kehadirat Allah SWT. Atas segala rahmat dan karunia-Nya. Sehingga penulis dapat menyelesaikan penyusunan laporan praktik industri tanpa suatu halangan yang berarti. Keberhasilan penyusunan laporan ini tidak lepas tanpa adanya pengarahan, bimbingan, dan bantuan dari berbagai pihak. Untuk itu penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT atas rahmat dan nikmatnya penulis beserta tim dapat menyelesaikan Praktek Industri dengan lancar.
2. Kedua orang tua yang senantiasa selalu mendukung dan mendo'akan penulis.
3. Yth. Bapak I Kadek Dwi Nuryana, S. T., M. Kom. selaku Ketua Jurusan Teknik Informatika.
4. Yth. Bapak Drs. Bambang Sujatmiko, M. T., selaku Ketua program Studi S1 Pendidikan Teknologi Informasi.
5. Bapak Titus Herman Efendi selaku Manager Bidang Operasi Pemeliharaan dan Aset SBU Regional Jawa Timur PT. Coments Plus (ICON+).
6. Rekan - rekan Praktik Industri (PI) setim penulis yang telah membantu mensukseskan Praktik Industri ini.
7. Teman-teman Prodi S1 Pendidikan Teknologi Informasi yang membantu menyelesaikan laporan ini.

Semoga Allah SWT membalas semua kebaikan mereka. Kemudian penyusun serahkan semuanya kepada Allah SWT, semoga yang penulis harapkan dapat terkabul, Aamiin.

Surabaya, 20 April 2021

Penulis

DAFTAR ISI

LAPORAN PRAKTIK INDUSTRI	i
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN.....	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	v
DAFTAR GAMBAR	vii
DAFTAR TABEL	viii
DAFTAR LAMPIRAN	ix
BAB I PENDAHULUAN.....	2
A. Latar Belakang.....	2
B. Tujuan.....	3
BAB II KAJIAN PUSTAKA	5
C. Website	5
D. Unsur-Unsur Website.....	6
E. PHP	8
F. XAMPP.....	10
G. HTML	11
H. CSS.....	11
I. <i>Framework</i>	12
J. Laravel.....	12
K. Bootstrap	13
L. Visual Studio Code	13
BAB III PELAKSANAAN PRAKTIK INDUSTRI DAN PEMBAHASAN.....	15

A. Profil Industri	15
a. Sejarah Perusahaan.....	15
b. Visi dan Misi ICON+	16
c. Lokasi, Gambaran Bangunan, dan Logo ICON +	16
d. Struktur Organisasi	21
B. Deskripsi Pelaksanaan Kegiatan.....	23
a. Identitas Perusahaan	23
b. Waktu Pelaksanaan	23
c. Faktor – Faktor Pendukung dan Penghambat.....	25
d. Keterlibatan Mahasiswa	25
C. Pembahasan	26
a. Implementasi Program	26
b. Source Code.....	34
BAB IV PENUTUP	61
A. Kesimpulan.....	61
B. Saran	62
DAFTAR PUSTAKA	63
LAMPIRAN.....	64

DAFTAR GAMBAR

Gambar 3.1. Pintu depan ICON+	19
Gambar 3.2. Tampilan Google Maps ICON+	19
Gambar 3.3. Logo ICON+	20
Gambar 3.4. Struktur Organisasi ICON+	22
Gambar 3.5. Struktur Organisasi Bidang Operasi, Pemeliharaan dan aset ICON+	22
Gambar 3.6. Halaman Awal dan Login	26
Gambar 3.7. Halaman Dashboard Utama 1	27
Gambar 3.8. Halaman Tabel User	27
Gambar 3.9. Kotak Dialog Tambah User	28
Gambar 3.10. Kotak Dialog Edit User	28
Gambar 3.11. Halaman Dashboard Utama 2	29
Gambar 3.12. Halaman Tabel Pelanggan 1	29
Gambar 3.13. Kotak Dialog Tambah Pelanggan	30
Gambar 3.14. Kotak Dialog Edit Pelanggan	30
Gambar 3.15. Kotak Dialog Hapus Pelanggan	31
Gambar 3.16. Halaman Dashboard Utama 3	31
Gambar 3.17. Halaman Tabel Pelanggan 2	32
Gambar 3.18. Halaman Dashboard Utama 4	33
Gambar 3.19. Halaman Tabel Pelanggan 3	33

DAFTAR TABEL

Tabel 3.1. Strategic Bussiness Unit ICON+	17
Tabel 3.2. Jadwal Pelaksanaan PI	23

DAFTAR LAMPIRAN

Lampiran 1 Lembar Nilai Praktik Industri.....	64
Lampiran 2 Surat Permohonan Ijin Praktik Industri	65
Lampiran 3 Surat Balasan	66
Lampiran 4 Lembar Bimbingan Praktik Industri	67
Lampiran 5 Dokumentasi Kegiatan Praktik Industri	68

BAB I

PENDAHULUAN

A. Latar Belakang

Kita telah memasuki era dimana teknologi dan informasi sangat dekat dan dibutuhkan oleh manusia, maka dari itu dibutuhkan sebuah strategi untuk pengembangan di bidang teknologi informasi yang dapat diaplikasikan ke segala bidang. Oleh karena itu banyak industri yang menjadikan teknologi sebagai kebutuhan dan juga perantara dalam memudahkan bisnis dalam industri. Perusahaan harus bertahan dalam dunia persaingan bisnis dengan menyiapkan sumber daya manusia yang berkualitas dan berdaya saing. Maka dari itu adanya Praktik industri dalam suatu Universitas diharapkan memberikan suatu sumbangan yang besar pada pengembangan teknologi informasi di berbagai bidang untuk industri, diharapkan juga mampu menambah wawasan mahasiswa dan juga salah satu langkah bagi mahasiswa untuk menerapkan ilmu yang didapat di perguruan tinggi.

Penulis memilih PT. Indonesia Comnets Plus (ICON +) Surabaya sebagai tempat melaksanakan Praktik Industri. Pada pelaksanaan Praktik Industri penulis diberikan penugasan yaitu mengimplementasikan website yang sedang penulis bangun yaitu SIVI (Serah Terima Dokumen dan Aktivasi).

Maka dari itu penulis membuat laporan yang berjudul **“IMPLEMENTASI APLIKASI SERAH TERIMA DOKUMEN AKTIVASI BERBASIS WEBSITE DI PT. INDONESIA COMNETS PLUS SURABAYA”**.

B. Tujuan

Adapun tujuan dari pengusulan praktik industri ini adalah sebagai berikut:

1. Tujuan Umum
 - a. Terwujudnya pola hubungan yang jelas dan terarah antara dunia perguruan tinggi dan pengguna outputnya.
 - b. Dunia usaha mampu mewujudkan kepedulian dan partisipasinya dalam ikut memberikan kontribusi pada sistem pendidikan nasional.
 - c. Membuka wawasan mahasiswa agar dapat mengetahui dan memahami sistem kerja di dunia industri sekaligus mampu mengadakan pendekatan, penyerapan dan pemecahan masalah yang berasosiasi dengan dunia kerja secara utuh.
 - d. Menumbuhkan dan menciptakan pola berpikir yang konstruktif yang berwawasan bagi mahasiswa dan dunia kerja.
2. Tujuan Khusus
 - a. Meningkatkan kemampuan mahasiswa atau calon sarjana dalam menghadapi kehidupan sosial di masyarakat secara baik dan harmonis.
 - b. Mendapatkan pengalaman kerja yang relevan sekaligus media adaptasi/pengenalan terhadap lingkungan kerja sebelum terjun ke dunia kerja yang sesungguhnya.
 - c. Mengenal dan mempelajari tentang permasalahan-permasalahan yang terjadi di dunia kerja, dan kemudian berusaha mencari penyelesaiannya berdasarkan ilmu yang didapat di bangku kuliah.
 - d. Mempersiapkan dan melatih profesionalisme di bidang Bidang Kerja Praktik.

BAB II

KAJIAN PUSTAKA

C. Website

Ketika kita menjelajah internet, seringkali kita diarahkan untuk membuka sebuah situs atau tempat yang mengandung berbagai jenis konten. Tempat atau situs tersebut dinamakan *website*. Website adalah media dengan beragam halaman yang terhubung (*hyperlink*) dan memuat suatu informasi berupa berkas baik secara tekstual, visual, maupun audio (Elgamar, 2020). Secara teknis, website merupakan sekumpulan *page* dalam sebuah domain atau subdomain tertentu yang ditempatkan pada sebuah *server web* dan membutuhkan jaringan Internet atau jaringan lokal untuk mengaksesnya. Dalam proses akses *website*, diperlukan protokol komunikasi jaringan yang disebut HTTP serta alamat akses *website* yang dinamakan URL. Seluruh website yang bisa diakses cara bebas berada dalam World Wide Web (WWW).

Website terbagi dalam 2 jenis. Jenis pertama adalah *website* statis, sebuah *website* dengan struktur sederhana dan memiliki fungsi sederhana pula dengan tipikal perbaruan tidak berkala. Sehingga perubahan isi *website* hanya dilakukan secara manual. Sedangkan jenis kedua adalah *website* dinamis, sebuah *website* dengan struktur kompleks dan memang dirancang sedemikian rupa dengan maksud dan tujuan yang luas serta mudah untuk diperbarui. *Website* jenis ini akan membutuhkan perbaruan secara berkala berdasarkan tujuan dibuatnya website tersebut. *Website* telah merambah berbagai bidang dengan masing-masing tujuannya, mulai dari *website e-commerce*, blog pribadi, jejaring sosial, konten visual dan berbagai macam bidang lainnya.

Untuk mengakses sebuah website, diperlukan sebuah perangkat lunak penjelajah web yang dinamakan peramban atau biasa disebut *browser*. Contoh peramban ini adalah Google

Chrome, Mozilla Firefox, Safari, Opera, dan berbagai peramban lainnya.

D. Unsur-Unsur Website

Sebuah website terdiri dari berbagai unsur yang membangun *website* itu sendiri. Setiap unsur tersebut saling tersinkronisasi sehingga terbentuklah sebuah *website* yang dikenal saat ini. Unsur-unsur tersebut adalah (Solichin, 2016):

1. Nama Domain (URL)

Nama domain diidentifikasi sebagai alamat unik sebuah website agar bisa diakses pengguna. Alamat unik inilah yang menjadi identitas utama website ketika pengguna mencari website yang dimaksud. Sebuah nama domain bisa ditentukan secara bebas, namun untuk memperolehnya diperlukan sejumlah biaya sebagai biaya sewa dalam jangka waktu tertentu kepada penyedia nama domain terkait. Domain bisa saja diperoleh secara gratis, namun penamaannya belum tentu bisa diubah sesuai dengan keinginan atau hanya mengikuti pakem penyedia domain tersebut. Beberapa domain juga bisa ber-ekstensi lokal dalam negeri. Misalnya seperti domain berakhiran “.go.id” adalah *website* yang dikhususkan untuk lembaga atau instansi Pemerintahan Republik Indonesia.

2. Rumah *website* (Web Hosting)

Untuk menyempurnakan fungsi dari sebuah *website*, diperlukan sebuah penyimpanan secara online agar sebuah *website* dapat memuat berkas yang akan ditampilkan di *website* tersebut. Penyimpanan tersebut disebut sebagai *web hosting*. Kapasitas penyimpanan pun juga bermacam-macam, disesuaikan dengan biaya sewa dalam jangka waktu tertentu dan kemampuan yang dipilih. Kapasitasnya bisa mulai dari ratusan MB (Mega Byte) hingga GB (Giga Byte), bahkan bisa lebih besar dari itu

tergantung jenis *website*. Proses penyewaan bisa dilakukan melalui perusahaan penyedia *hosting* dengan masing-masing kapabilitasnya. Namun jika masih dalam pengembangan, *website* dapat dibangun secara lokal menggunakan server desktop milik pengembang yang biasa disebut *localhost* (Enterprise, 2019).

3. Bahasa Skrip (*Scripts*)

Bahasa skrip atau bahasa program merupakan penerjemah atas instruksi yang dilakukan di *website* ke dalam bahasa mesin. Di masa kini, bahasa program sudah semakin banyak dan memiliki kelebihan masing-masing. Semakin banyak bahasa program yang disertakan, maka semakin dinamis pula sebuah *website* yang berdampak terhadap kemudahan pengelolaan *website*. Kualitas sebuah *website* juga bisa ditentukan melalui bahasa program yang digunakan dan kerumitan setiap fungsinya. Bahasa dasar sebuah *website* adalah HTML, dengan berbagai bahasa program penunjangnya seperti PHP, Javascript, Python dan berbagai bahasa program lainnya.

4. Desain Web

Desain *website* menentukan seberapa baik tampilan sebuah *website* bisa diterjemahkan oleh pengguna. Selain kebutuhan fungsional, desain web juga menjadi kebutuhan estetika sebuah *website* agar pengguna bisa merasakan kenyamanan ketika menggunakan sebuah *website*. Tampilan *website* adalah nilai jual yang paling pertama dilihat pada sebuah *website* sebelum memasuki bagian dalam *website*.

5. Program Transfer Data

Sebuah *website* memerlukan berbagai jenis file pendukung yang mana seluruh file tersebut diakses di komputer pembuat *website*. File pendukung *website* dapat

berupa teks, gambar, maupun video. Namun pengaksesan *file* tersebut hanya dapat dilakukan secara *offline*. Ketika sebuah *website* mengudara secara *online*, maka seluruh *file* tersebut harus disimpan dalam rumah *website* (*web hosting*) secara online pula. Hal tersebut membutuhkan fitur FTP (*File Transfer Protocol*) yang dapat memindahkan *file* di *website* ke dalam pusat data *web hosting*. Berbagai program FTP telah tersebar di internet baik berbayar maupun secara percuma-cuma.

6. Publikasi Web

Publikasi menjadi sebuah kewajiban bagi pengembang *website* jika ingin *website* tersebut dikenal secara umum dan dapat digunakan oleh berbagai orang di belahan dunia. Untuk melakukan publikasi, kita dapat melakukannya secara *offline* melalui sosialisasi, penyebaran pamflet, papan iklan dan sebagainya, ataupun secara online melalui iklan di mesin pencarian, iklan di situs ternama, atau mempromosikannya melalui kanal proyek freelancer dan sebagainya. Semakin baik publikasi, maka semakin banyak dan semakin luas pula cakupan pengguna *website* tersebut.

E. PHP

PHP merupakan singkatan dari *PHP: Hypertext Preprocessors*. Sedangkan PHP dalam singkatan tersebut berarti *Personal Home Page*. Sebagai salah satu bahasa pemrograman, PHP adalah bahasa pemrograman untuk *website* yang paling populer di kalangan programmer karena mudah digunakan dan tergolong *powerful* serta modern di masa kini. PHP *Development Team* selaku pengembang PHP telah merilis versi terkini yakni PHP 7.4. Bahasa ini dapat berjalan secara lintas platform. (Solichin, 2016) dalam bukunya “Pemrograman Web dengan PHP dan MySQL”, menyebutkan beberapa kelebihan PHP untuk pemrograman web, diantaranya :

1. Bisa digunakan secara gratis
2. Berlisensi GNU General Public License (GPL) berarti setiap versi PHP yang terdistribusi benar-benar gratis
3. Performa dapat diandalkan dan efisien
4. Mendukung berbagai jenis basis data.
5. Tersedia banyak library yang bisa digunakan
6. Dapat berjalan lintas platform
7. Kemudahan untuk mempelajari PHP

Secara fungsi, PHP digunakan sebagai bahasa skrip pemrograman dan bisa digunakan juga sebagai bahasa skrip *server-side*. Hal ini tentu mendukung pengembangan website dinamis daripada *website* statis yang hanya menggunakan HTML dan CSS. PHP disebut sebagai bahasa pemrograman skrip karena untuk menggunakannya, kita dapat menyisipkannya ke dalam HTML. Selain kegunaannya diatas, PHP juga bisa dibuat untuk membuat CMS atau Sistem Manajemen Konten yang berguna untuk mengelola konten sebuah *website*.

Terdapat beberapa sintaksis dasar dalam PHP, diantaranya:

1. Pembatas

Pembatas berfungsi untuk menentukan kode mana yang dapat dijalankan oleh PHP. Pembatas ini dibuka dengan “<?php” sebagai pembuka kode dan ditutup oleh “>”. diluar pembatas yang didefinisikan, maka PHP tidak dapat berjalan. Pembatas juga menjadi pemisah PHP dengan bahasa program lainnya.

2. Variabel

Penulisan variabel dalam PHP harus diawali dengan simbol *dollar* “\$” lalu diikuti oleh nama variabel dan diakhiri dengan titik koma “;”. Contohnya adalah \$variabel_satu.

3. Komentar

Bila ingin meninggalkan catatan pada kode program, kita bisa menggunakan “/**/”, “//”, dan # sebagai komentar yang mana teks tersebut tidak akan diterjemahkan ke dalam program.

4. Fungsi

Fungsi menjadi ekstensi tambahan yang tersedia untuk PHP. Hingga kini, berbagai fungsi telah ditemukan dengan berbagai penamaan yang berbeda-beda.

F. XAMPP

XAMPP merupakan program besutan Apache Friends yang berfungsi sebagai server lokal di komputer. XAMPP sendiri merupakan singkatan dari X yang artinya cross dan bermakna dapat digunakan sistem operasi manapun, A yang berarti Apache, M berarti MySQL, dan kedua P yang berarti PHP dan Perl. Program ini bersifat lintas platform sehingga bisa digunakan di berbagai sistem operasi (Solichin, 2016).

Kehadiran XAMPP sangat membantu pengembang *website* ketika ingin menjalankan *website* melalui *hosting* lokal (*localhost*). Dengan fungsinya sebagai *server* lokal, maka XAMPP dapat mengganti peran web *hosting* untuk sementara waktu selama *website* dalam proses pengembangan.

Perlu diketahui bahwasannya terdapat beberapa bagian penting dalam XAMPP, yakni:

1. Htdocs

Htdocs merupakan direktori *file* yang menyimpan segala *file* yang akan ditampilkan di *website*. Segala jenis *file* tersebut disatukan dalam 1 folder yang akan menjadi alamat akses *website* yang sedang dikembangkan.

2. Control Panel

Control Panel menjadi bagian penting dalam XAMPP yang berfungsi sebagai panel pengendali service yang akan kita jalankan atau dimatikan. Hal ini

disesuaikan dengan kebutuhan pengembang untuk menentukan mana yang dijalankan dan mana yang tidak.

3. PhpMyAdmin

Sebagaimana fungsi web *hosting*, PhpMyAdmin menjalankan peran sebagai pengelolaan konfigurasi MySQL. Untuk membukanya, kita dapat mengetikkan <http://localhost/phpmyadmin> di kolom teks *browser* kita.

G. HTML

HTML merupakan singkatan dari *HyperText Markup Language*. Sebuah bahasa markah yang digunakan untuk menampilkan informasi yang tersedia dalam *website* (Solichin, 2016). HTML menjadi standar bahasa yang digunakan ketika mengembangkan sebuah *website*. Kini versi HTML telah mencapai HTML 5, menjadikan *website* semakin dinamis dan interaktif. Keberadaannya juga dibantu oleh jenis bahasa lainnya seperti CSS terkait tampilan dan bahasa skrip seperti Javascript. Terdapat 2 tipe dalam HTML, yakni *block level* dan *inline tags*. Jika *block level* hanya menggunakan seluruh space yang tersedia serta membentuk *line* baru pada dokumen, maka *inline tags* hanya menggunakan spasi sesuai kebutuhan dan tidak ada pembuatan *line* baru.

H. CSS

Sebuah *website* belum lengkap jika belum terdapat CSS. Dengan kepanjangan *Cascading Sheet Style*, CSS menjadi bahasa desain web yang berguna untuk mengatur bagaimana tampilan sebuah *website*. CSS dibuat secara terpisah dari konten utama yang bisa meminimalisir kerumitan kode program dan memudahkan pembuatan tampilan sesuai dengan kebutuhan (Solichin, 2016).

I. Framework

Framework merupakan sebuah kerangka kerja yang biasa diterapkan dalam pengembangan perangkat lunak berbasis *desktop* maupun *website*. Sebuah *framework* tersusun dalam sebuah pola yang disebut MVC (*Model View Controller*), fungsinya adalah untuk memisahkan antara presentasi dengan logika. Dengan adanya *framework*, pengembang dapat dengan mudah mengelola komponen pemrograman secara tersusun dan terstruktur, sehingga pengembang dapat meningkatkan efisiensi waktu selama masa pengembangan perangkat lunak serta memudahkan pembagian kerja apabila pengembangan dikerjakan lebih dari 1 orang. *Framework* memiliki berbagai macam jenis yang digolongkan dalam beberapa basis aplikasi. Seperti Laravel pada PHP, Xamarin pada *Desktop* maupun *Mobile*, dan jenis-jenis *framework* populer lainnya.

J. Laravel

Laravel adalah salah satu jenis *framework* PHP yang cukup umum digunakan oleh pengembang perangkat lunak berbasis *website* khususnya di Indonesia. Laravel sendiri rilis pertama kali pada tahun 2011. Kini, laravel telah memasuki versi 8.0 yang rilis beberapa waktu lalu. Struktur yang dimiliki Laravel cukup sederhana, namun dapat membangun sebuah *website* yang powerful. Hal tersebutlah yang menyebabkan Laravel cukup populer di kalangan pengembang perangkat lunak, baik dalam proyek skala kecil maupun yang berskala besar. Terdapat 2 *tools* utama dari Laravel, yakni Composer dan Artisan. Composer serupa dengan *dependencies management* yang berisikan sekumpulan *dependencies* dan *library* yang disimpan dalam format json. Sedangkan Artisan adalah *command line interface* milik Laravel yang mencakup berbagai perintah guna membantu pengembangan *website* berbasis Laravel. Adapun fitur-fitur yang terdapat pada Laravel adalah *Blade Template Engine*, *Routing*, *Modularity*, *Testability*, *Query Builder* dan *ORM*, *Authentication*, *Schema Builder*, *Configuration*

Management, E-mail Class, Redis, dan Event and Command Bus (Yudhanto, et al., 2019).

K. Bootstrap

Bootstrap merupakan serangkaian kerangka tampilan website yang berbasis HTML, CSS, dan Javascript. Menurut (Hery Sulistiono, 2018) Bootstrap merupakan sebuah Pustaka yang bersifat *open-source* dan berupa *framework* dengan CSS dan Javascript sehingga dapat membuat sebuah web menjadi responsif. Dulunya, Bootstrap bernama Twitter Blueprint sebelum akhirnya dirilis ke publik dengan nama Bootstrap. Bisa dikatakan jika bootstrap adalah *framework* CSS yang berguna dalam pengembangan *front-end* sebuah *website*. Dengan bootstrap, maka kita akan mendapatkan kemudahan serta efisiensi ketika sedang membuat tampilan *website* yang responsif.

Agar dapat menggunakannya, kita perlu menginstallnya terlebih dahulu. Proses install bisa dilakukan secara *online* maupun *offline*. Tentunya setiap sistem operasi akan memiliki cara yang berbeda-beda.

L. Visual Studio Code

Untuk membangun sebuah *website*, diperlukan sebuah program berupa *text editor* yang berfungsi untuk mengolah kode program *website* yang sedang dikembangkan. Salah satu program *text editor* yang populer adalah Microsoft Visual Studio Code yang dikembangkan oleh Microsoft. Dirilis pertama kali pada tahun 2015 dan kini telah memasuki versi 1.52.1 untuk versi terbaru. Program ini tentunya mendukung berbagai jenis bahasa pemrograman seperti JavaScript, CSS, Java, HTML, Typescript, dan bahasa lainnya. Sehingga program ini dapat memudahkan proses pembuatan sebuah *website* yang memerlukan berbagai jenis bahasa program (Blazing, 2019).

BAB III

PELAKSANAAN PRAKTIK INDUSTRI DAN PEMBAHASAN

A. Profil Industri

a. Sejarah Perusahaan

PT. Indonesia Comnets Plus (ICON+) adalah perusahaan nasional yang bergerak di bidang penyelenggaraan jaringan dan jasa telekomunikasi. ICON+ merupakan anak perusahaan PT. PLN (Persero) yang didirikan pada 3 Oktober 2000. ICON+ bertujuan untuk mengoptimalkan pemanfaatan infrastruktur jaringan kelistrikan untuk telekomunikasi, yang semula hanya dimanfaatkan untuk memenuhi kebutuhan layanan jaringan telekomunikasi bagi operasi ketenagalistrikan. Seiring berjalannya waktu dan berkembangnya teknologi yang dimiliki, kini PT. Indonesia Comnets Plus (ICON+) mampu beroperasi sebagai penyedia utama jaringan telekomunikasi baik bagi PLN maupun sector public.

PT. Indonesia Comnets Plus (ICON+) mendukung perkembangan telekomunikasi dan teknologi informasi di Indonesia dengan memanfaatkan *Right of Ways* (RoW) jaringan kelistrikan PLN untuk menyediakan layanan jasa dan jaringan telekomunikasi berbasis *fiber optic*, baik bagi PLN maupun pelanggan *corporate* lainnya. Dalam hal ini ICON+ menyediakan jaringan *fiber optic* yang terbentang di pulau jawa, Bali, Nusa Tenggara, Sumatra, Sulawesi dan Kalimantan. Sejak tahun 1995, PLN telah Mengoperasikan jaringan *fiber optic*

terutama di pulau Jawa untuk memenuhi kebutuhan telekomunikasi internal PLN.

b. Visi dan Misi ICON+

Visi:

Menjadi penyedia solusi TIK tekemuka di Indonesia berbasis jaringan melalui pemanfaatan asset strategis.

Misi:

1. Memberikan layanan TIK yang terbaik di kelasnya kepada pelanggan guna meningkatkan nilai perusahaan.
2. Memenuhi kebutuhan dan harapan PLN secara proaktif dengan menyediakan solusi – solusi TIK yang inovatif dan memberikan nilai tambah.
3. Membangun organiasai pembelajar yang berkinerja tinggi untuk mendorong perusahaan mencapai bisnis yang unggul dan menjadi pilihan bagi talenta-talenta terbaik.
4. Memberikan kontribusi terhadap perkembangan telekomunikasi nasional.

c. Lokasi, Gambaran Bangunan, dan Logo ICON +

PT. Indonesia Comnets Plus (ICON+) memiliki kantor pusat yang terletak di Kawasan PLN Cawang, Jalan Mayjend Sutoyo No.1, Cililitan, Jakarta Timur, 13640. Sedangkan untuk alamat Korespondensi berada di Jalan KH Abdul Rochim No.1, Kuningan Barat, Mampang, Jakarta Selatan, 12710.

Untuk memenuhi kebutuhan pelanggan yang tersebar di seluruh pelosok Indonesia, diperlukan *Strategic Bussiness Unit* (SBU) di masing – masing

regional PT. Indonesia Comnets Plus (ICON+), yaitu sebagai table berikut:

No.	Strategic Business Unit Regional	Alamat
1.	Sumatera Utara	Jl. Jend Gatot Subroto No. 198, Medan Helvetia, Medan, Sumatera Utara 20123, Telepon (061) 8477983, Fax (061) 8457458.
2.	Sumatera	Jl. Soekarno Hatta, Komp Griya Satitri No. 4-5, Marpoyan Damai, Pekanbaru, Telepon (0751) 810012, Fax (0751) 34656.
3.	Sumatera Selatan	Jl. R. Sukamto No. 92 B-C, Plimbang 30114, Telepon (0711) 363963, Fax (0711) 378718.
4.	Jakarta & Banten	Kompleks PT PLN (Persero) Duren Tiga Gedung 12, Jl. Karang Ktes, Jakarta Selatan 12760, Telepon (021) 29532400.
5.	Jawa Barat	Jl. Supratman No. 58, Bandung, Jawa Barat 40121, Telepon (022) 7200262, Fax (022) 7203742.
6.	Jawa Tengah	Jl. Setia Budi No. 96, Srandol Kulon, Banyumanik, Semarang 50263, Telepon (024) 76403482.
7.	Jawa Timur	PLN Pikitring, Jl. Ketintang Baru I No. 1-3, Surabaya, Telepon (031) 8273399,

		Fax (031) 8286611.
8.	Bali & Nusa Tenggara	Area Perkantoran PT. PLN PUSHARLIS, Jl. Gunung Mandalawangi No. 15, tegal Harum, Denpasar, Bali 80119, Telepon (0361) 480710, Fax (0361) 480731.
9.	Kalimantan	Panin Tower Lantai 15, Jl. Jendral Sudirman, Klandasan Ilir, Balikpapan Kota, Kota Balikpapan, Kalimantan Timur 76113, Telepon (0542) 8501777.
10.	Sulawesi & Indonesia Timur	Jl. DR. Sam Ratulangi No. 134, Makasar 90125, Telepon (0411) 8919068, Fax (0411) 8919094.

Tabel 3. 1. Strategic Bussiness Unit ICON+

Lokasi Kegiatan Praktik Industri (PI) ini terletak di Kantor PT. Indonesia Comnets Plus (ICON+) SBU Regional Surabaya, yang beralamat di PLN Pikitring, Jalan Ketintang Baru 1 No. 1-3, Ketintang, Kecamatan Gayungan, Kota Surabaya, Jawa Timur 60231.



Gambar 3. 1. Pintu depan ICON+



Gambar 3. 2. Tampilan Google Maps ICON+

Logo PT. Indonesia Comnets Plus (ICON+) seperti berikut :



Gambar 3. 3. Logo ICON+

Etika profesi PT. Indonesia Comnets Plus tertera pada logo perusahaan. Berikut ini arti logo sekaligus etika profesi PT. Indonesia Comnets Plus:

1. Integrity
ICONERS (Sebutan untuk seluruh karyawan PT. Indonesia Comnets Plus) harus jujur, tulus dan dapat dipercaya dalam berpikir, berkata dan bertindak. Tanda garis coretan pada huruf "I" melambangkan ICONERS harus selalu berhubungan dengan Tuhan Yang Maha Esa.
2. Care
ICONERS selalu bersiap perduli, berempati dan responsive serta, memberikan pelayanan yang melebihi harapan *stake holder*.
3. Open Mind
ICONERS bersikap yang objektif dan komunatif untuk menghasilkan kinerja yang baik.

4. Innovation

ICONERS selalu mengerahkan kreativitas dalam segala hal untuk menghasilkan nilai tambah bagi *stake holder* (pemegang saham, pelanggan, karyawan, pemerintah).

5. Teamwork

ICONERS percaya bahwa kekuatan sinergi dapat membentuk tim pemenang. ICONERS bekerja sama dengan semangat gotong royong, saling mengisi dan saling menyemangati, berfikir menang – menang untuk memperoleh hasil yang maksimal.

6. Excellence

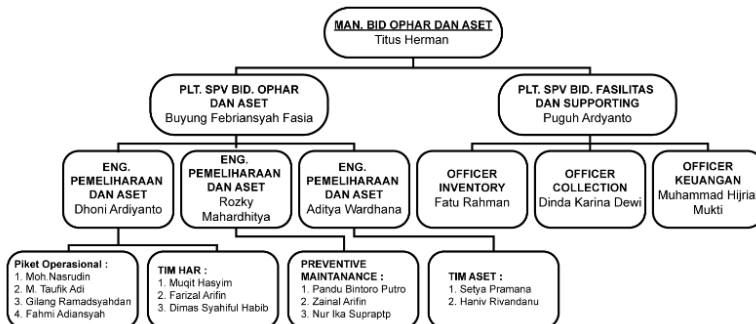
ICONERS bekerja dengan cerdas dan ulet untuk menghasilkan kualitas terbaik dalam mendukung keberhasilan perusahaan.

d. Struktur Organisasi

Dalam suatu perusahaan diperlukan struktur organisasi yang mengatur tugas dan wewenang serta tanggung jawab dari setiap bagian. Adapun struktur organisasi yang diperlukan oleh PT. Indonesia Comnets Plus adalah struktur organisasi dari staf dimana didalamnya terdapat pengawasan secara langsung dan spesialis dalam pekerjaan. Berikut adalah bagan struktur organisasi PT. ICON+ beserta nama setiap staffnya. Struktur organisasi PT. Indonesia Comnets Plus (ICON+) SBU Regional Surabaya dipimpin oleh seorang General Manager dan membawahi tiga Bidang yaitu, Bidang Operasi, Pemeliharaan dan Aset, Bidang Pembangunan dan Aktivasi, serta Bidang Pemasaran dan Penjualan.



Gambar 3. 4. Struktur Organisasi ICON+



Gambar 3. 5. Struktur Organisasi Bidang Operasi, Pemeliharaan dan aset
ICON+

B. Deskripsi Pelaksanaan Kegiatan

Adapun deskripsi pelaksanaan kegiatan Praktik Industri adalah sebagai berikut:

a. Identitas Perusahaan

Nama : PT. Indonesia Comnets Plus
Alamat : Jl. Ketintang Baru No.1-3, Ketintang,
Kec. Gayungan, Kota Surabaya,
Jawa Timur 60231.
Kontak : (031)8273399

b. Waktu Pelaksanaan

Kegiatan Praktik Industri berlangsung selama 1 bulan, dari tanggal 25 Januari – 26 Februari 2021. Dilaksanakan WHO dan WFH mengingat situasi dan kondisi, dan untuk mencegah penyebaran COVID-19.

No.	Hari,Tanggal	Uraian Kegiatan
1.	Senin 25 Januari 2021	Pengenalan Kantor dan diskusi project yang akan dikerjakan
2.	Selasa 26 Januari 2021	WFH (Work From Home) Diskusi Perencanaan Project melalui Grup WA
3.	Rabu 27 Januari 2021	WFH (Work From Home) Diskusi Perencanaan Project
4.	Kamis 28 Januari 2021	WFH (Work From Home) Brief Project
5.	Jumat 29 Januari 2021	WFH (Work From Home) Brief Project
6.	Senin 1 Febuari 2021	WFH (Work From Home) Pembagian Jobdisk
7.	Selasa 2 Febuari 2021	WFH (Work From Home) Mengerjakan Project
8.	Rabu 3 Febuari 2021	WFH (Work From Home) Mengerjakan Project
9.	Kamis 4 Febuari 2021	WFH (Work From Home) Mengerjakan Project
10.	Jumat	WFH (Work From Home)

	5 Febuari 2021	Mengerjakan Project
11.	Senin 8 Febuari 2021	WFH (Work From Home) Mengerjakan Project
12.	Selasa 9 Febuari 2021	WFH (Work From Home) Mengerjakan Project
13.	Rabu 10 Febuari 2021	WFH (Work From Home) Asistensi Project
14.	Kamis 11 Febuari 2021	WFH (Work From Home) Revisi Project
15.	Jumat 12 Febuari 2021	WFH (Work From Home) Revisi Project
16.	Senin 15 Febuari 2021	WFH (Work From Home) Asistensi Project
17.	Selasa 16 Febuari 2021	WFH (Work From Home) Revisi Project
18.	Rabu 17 Febuari 2021	WFH (Work From Home) Revisi Project
19.	Kamis 18 Febuari 2021	WFH (Work From Home) Asistensi Project
20.	Jumat 19 Febuari 2021	WFH (Work From Home) Proses Menyesuaikan Website tahap akhir
21.	Senin 22 Febuari 2021	WFH (Work From Home) Proses Menyesuaikan Website tahap akhir
22.	Selasa 23 Febuari 2021	WFH (Work From Home) Proses Menyesuaikan Website tahap akhir
23.	Rabu 24 Febuari 2021	WFH (Work From Home) Proses Menyesuaikan Website tahap akhir
24.	Kamis 25 Febuari 2021	Website sudah bisa digunakan
25.	Jumat 26 Febuari 2021	Menyerahkan Website yang telah dibuat

Tabel 3. 2. Jadwal Pelaksanaan PI

c. Faktor – Faktor Pendukung dan Penghambat

a. Faktor Pendukung

Selama melaksanakan Praktik Industri di PT. Comnets Plus, ada beberapa factor pendukung yang membantu kelancaran dalam melakukan kegiatan praktik industry, diantaranya sebagai berikut:

1. Adanya Kerjasama yang baik.
2. Kesediaan perusahaan dalam menerima kami untuk melaksanakan praktik industry di PT. Comnets Plus.

b. Faktor Penghambat

Berikut adalah factor penghambat dalam kegiatan praktik industry di PT. Comnets Plus:

1. Dikarenakan kondisi pandemi Covid-19 dan mengharuskan penulis untuk melaksanakan Praktik Industri Work from Home untuk mencegah penyebaran covid-19, sehingga penulis kurang merasakan pengalaman nyata bekerja dalam suatu perusahaan.

d. Keterlibatan Mahasiswa

Selama mahasiswa melakukan Praktik Industri, mahasiswa juga melakukan instruksi yang diberikan dari pihak industri. Hal tersebut meliputi pembagian tugas dalam kegiatan keseharian pada masing – masing bidang dan proyek bersama.

C. Pembahasan

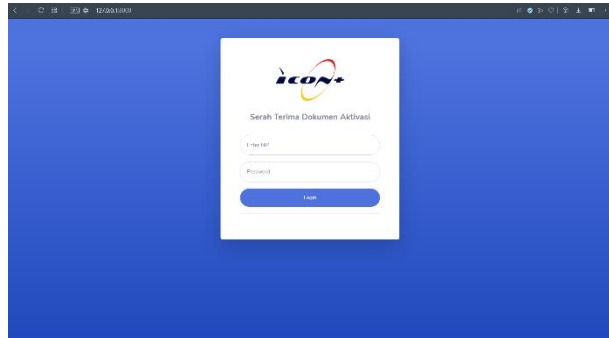
Pada Kegiatan Praktik Industri sejak tanggal 25 Januari 2021 hingga 26 Februari 2021 di PT. Indonesia Comnets Plus Surabaya (ICON+), Penulis diberikan penugasan berupa pembuatan website serah terima dokumen yang akan digunakan oleh perusahaan.

a. Implementasi Program

Tahap ini merupakan langkah-langkah dari pemakaian Aplikasi Serah Terima Dokumen Aktivasi. Adapun penjelasan yang akan dijabarkan pada sub sub dibawah ini.

1. Halaman Awal dan Login

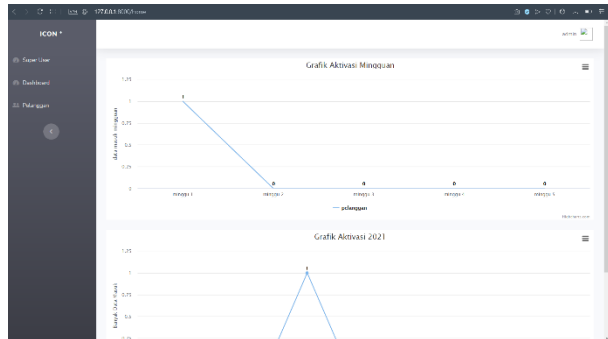
Setiap user yang ingin masuk kedalam web diminta untuk login terlebih dahulu agar mendapatkan akses kedalam web, dengan memasukkan NIP dan Password.



Gambar 3. 6. Halaman Awal dan Login

2. Halaman Dashboard Utama Super Admin

Setelah melakukan login, Role Super Admin akan diarahkan ke halaman dashboard utama, di dashboard ini user akan mendapatkan informasi terkait jumlah pelanggan yang masuk tiap minggu dan tiap bulannya



Gambar 3. 7. Halaman Dashboard Utama 1

3. Halaman Tabel User di Super Admin

Untuk Role Super Admin Memiliki akses untuk menambah, mengedit, dan menghapus User atau pengguna di Role lainnya, seperti Role Aktivasi, Pemeliharaan, dan Sales. Segala bentuk aktivitas pengolahan user tadi ada pada halaman ini

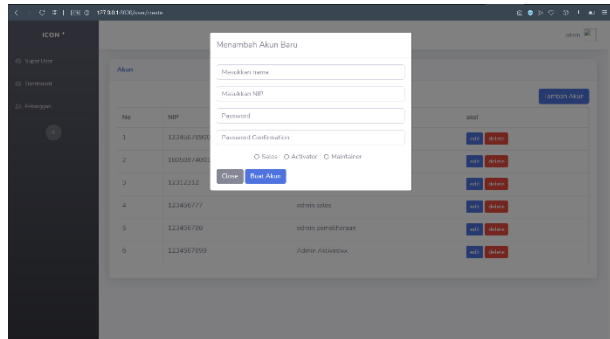
The screenshot shows a table titled 'Akses' with columns: 'No', 'NIP', 'Name', and 'aksi'. The table contains 6 rows of user data. Each row has two buttons in the 'aksi' column: 'edit' (blue) and 'hapus' (red). A 'Tambah Akses' button is located at the top right of the table.

No	NIP	Name	aksi
1	12345678900	elmer	<button>edit</button> <button>hapus</button>
2	10050070001	aktivasi dinas	<button>edit</button> <button>hapus</button>
3	1231212	reptok	<button>edit</button> <button>hapus</button>
4	123806777	aktivasi sales	<button>edit</button> <button>hapus</button>
5	123806788	aktivasi pemeliharaan	<button>edit</button> <button>hapus</button>
6	123806789	Aktifasi Aktivasi	<button>edit</button> <button>hapus</button>

Gambar 3. 8. Halaman Tabel User

4. Halaman Tambah User oleh Super Admin

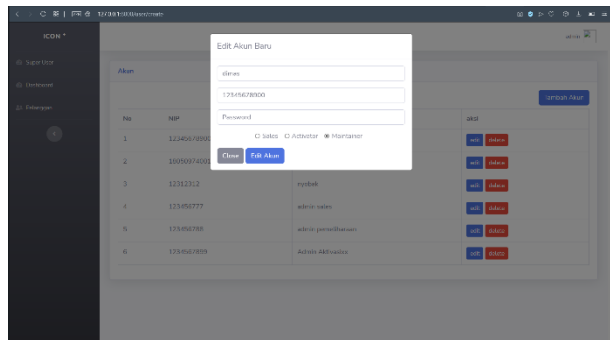
Berikut adalah tampilan halaman ketika melakukan tambah user yang dilakukan di role super admin



Gambar 3. 9. Kotak Dialog Tambah User

5. Halaman Edit User oleh Super Admin

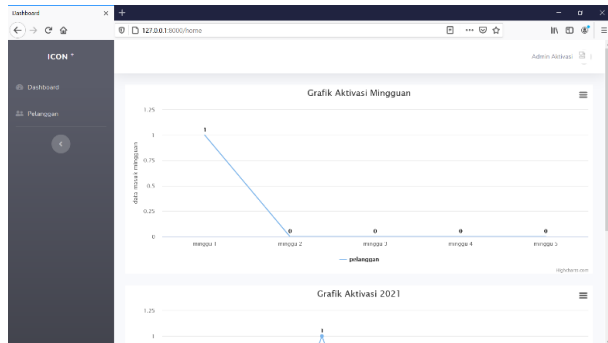
Berikut adalah tampilan halaman ketika melakukan edit user yang dilakukan di role super admin



Gambar 3. 10. Kotak Dialaog Edit User

6. Halaman Dashboard Utama Role Aktivasi

Setelah melakukan login, Role Aktivasi akan diarahkan ke halaman dashboard utama, di dashboard ini user akan mendapatkan informasi terkait jumlah pelanggan yang masuk tiap minggu dan tiap bulannya



Gambar 3. 11. Halaman Dashboard Utama 2

7. Halaman Tabel Pelanggan di Role Aktivasi

User mendapatkan informasi terkait pelanggan yang telah masuk, mulai dari Nomer ISO, Nama Pelanggan, Status Aktivasi, Alamat, Tanggal Aktivasi, penanggung Jawab, dan dokumen aktivasi. Pada role aktivasi ini, user dapat melakukan tambah, edit, dan hapus pelanggan, serta juga nantinya bisa melihat apakah dokumen yang telah diupload, sudah di cek oleh tim pemeliharaan sebagai bentuk simbolis serah terima dokumentasi.

Mark as read?	Nomor SO	Pelanggan	Status	Alamat	Tanggal Upload	Pelaksana	Aksi
<input type="checkbox"/>	0001	Galuhul	Aktif	UNESA Kertirang	20-May-2021	Dimas	Sedikit Download

Gambar 3. 12. Halaman Tabel Pelanggan 1

8. Halaman Tambah Pelanggan

Berikut adalah tampilan halaman ketika melakukan tambah pelanggan yang dilakukan di role aktivasi

The screenshot shows a web application interface with a sidebar on the left containing icons for Dashboard, Pelanggan, Mark as read?, and a plus icon. The main content area displays a modal dialog titled "Menambah Pelanggan Baru". The dialog contains the following fields: "Nomor SO" (text input), "Nama Pelanggan" (text input), "Alamat" (text input), "Status" (radio buttons for "Aktif" and "Non-Aktif", with "Non-Aktif" selected), "Pelaksanaan Tugas" (text input), and "Unggah Dokumen" (a "Choose File" button with the text "No file chosen" below it). At the bottom right of the dialog are "Tutup" and "Simpan" buttons. In the background, a sidebar menu is visible with "Pelanggan" selected, and a main content area showing a "Tambah Pelanggan" button and a file upload section with a "file?" label and a "Choose File" button.

Gambar 3. 13. Kotak Dialog Tambah Pelanggan

9. Halaman Edit Pelanggan

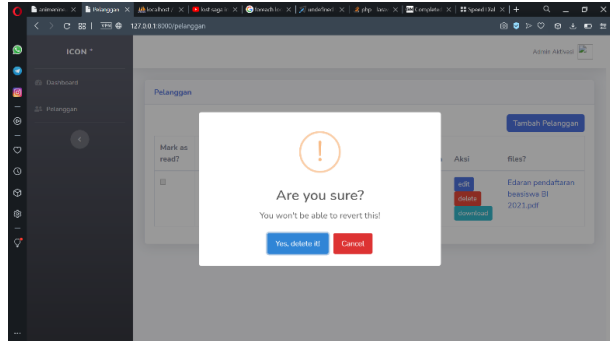
Berikut adalah tampilan halaman ketika melakukan edit pelanggan yang dilakukan di role aktivasi

The screenshot shows the same web application interface as before, but with a modal dialog titled "Edit Pelanggan". The dialog contains the following fields: "Nomor SO" (text input with value "0001"), "Nama Pelanggan" (text input with value "Sakhrul"), "Alamat" (text input with value "UNESA Kertelung"), "Status" (radio buttons for "Aktif" and "Non-Aktif", with "Aktif" selected), "Pelaksanaan Tugas" (text input with value "Dimas"), and "Unggah Dokumen" (a "Choose File" button with the text "No file chosen" below it). At the bottom right of the dialog are "Tutup" and "Simpan" buttons. The background shows the same sidebar and main content area as in the previous screenshot.

Gambar 3. 14. Kotak Dialaog Edit Pelanggan

10. Halaman Hapus Pelanggan

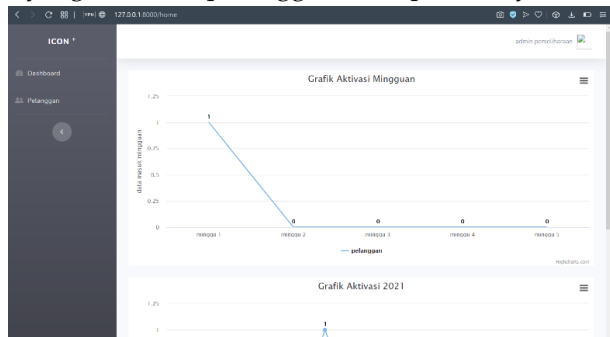
Berikut adalah tampilan kotak dialog ketika melakukan hapus pelanggan yang dilakukan di role aktivasi



Gambar 3. 15. Kotak Dialog Hapus Pelanggan

11. Halaman Dashboard Utama Role Pemeliharaan

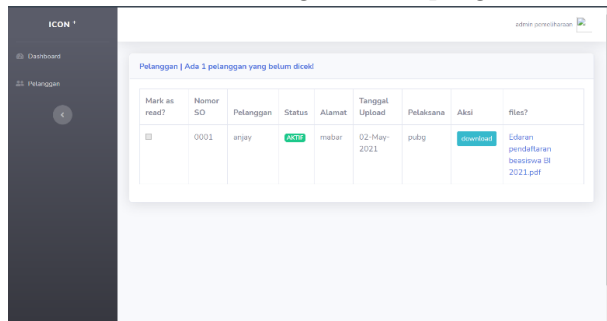
Setelah melakukan login, Role Pemeliharaan akan diarahkan ke halaman dashboard utama, di dashboard ini user akan mendapatkan informasi terkait jumlah pelanggan yang masuk tiap minggu dan tiap bulannya



Gambar 3. 16. Halaman Dashboard Utama 3

12. Halaman Tabel Pelanggan di Role Pemeliharaan

Pada halaman ini user mendapatkan informasi terkait pelanggan yang telah masuk, mulai dari Nomer ISO, Nama Pelanggan, Status Aktivasi, Alamat, Tanggal Aktivasi, penanggung Jawab, dan dokumen aktivasi. Pada role pemeliharaan, dapat melakukan download dokumen aktivasi sebagai bentuk pengawasan.



admin pemeliharaan

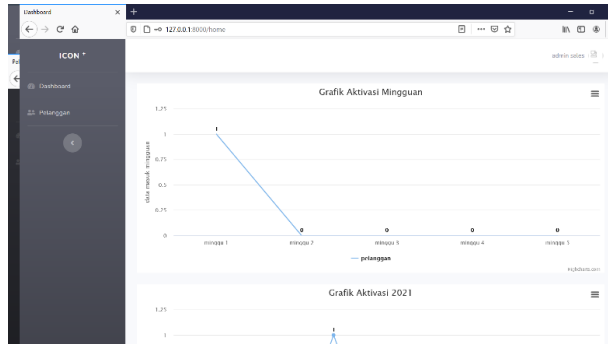
Pelanggan | Ada 1 pelanggan yang belum dibaca

Mark as read?	Nomer ISO	Pelanggan	Status	Alamat	Tanggal Upload	Pelaksana	Aksi	File?
<input type="checkbox"/>	0001	anjey	aktif	mabar	02-May-2021	pubg	download	Ederen pendaftaran keaktifan B 2021.pdf

Gambar 3. 17. Halaman Tabel Pelanggan 2

13. Halaman Dashboard Utama Role Sales

Setelah melakukan login, Role Sales akan diarahkan ke halaman dashboard utama, di dashboard ini user akan mendapatkan informasi terkait jumlah pelanggan yang masuk tiap minggu dan tiap bulannya



Gambar 3. 18. Halaman Dashboard Utama 4

14. Halaman Tabel Pelanggan di Role Sales

Pada halaman ini user mendapatkan informasi terkait pelanggan yang telah masuk, mulai dari Nomer ISO, Nama Pelanggan, Status Aktivasi, Alamat, Tanggal Aktivasi, penanggung Jawab, dan dokumen aktivasi.

The screenshot shows a web dashboard with a dark sidebar menu. The main content area displays a table titled 'Pelanggan'. The table has six columns: 'Nomer SO', 'Pelanggan', 'Status', 'Alamat', 'Tanggal Upload', and 'Pelaksana'. There is one data row visible.

Nomer SO	Pelanggan	Status	Alamat	Tanggal Upload	Pelaksana
0001	anjay	aktif	malbar	02-May-2021	pubg

Gambar 3. 19. Halaman Tabel Pelanggan 3

b. Source Code

1. Halaman Awal dan Login

```
@extends('layouts.auth')
@section('content')
<style>
    .center {
        display: block;
        margin-left: auto;
        margin-right: auto;
        width: 50%;
    }
</style>
<div class="row justify-content-center">
    <div class="col-xl-5 col-lg-12 col-md-9">
        <div class="card o-hidden border-0
shadow-lg my-5">
            <div class="card-body p-0">
                <!-- Nested Row within Card
Body -->
                    <div class="row">
                        <div class="col-lg-12">
                            <div class="p-5">
                                <div class="card
o-hidden border-0">
                                    
                                </div>
                                <br>
                                <div class="card
o-hidden border-0">
                                    <h5
class="text-center font-weight-bold">Serah
Terima Dokumen Aktivasi</h5>
                                </div>
                                <br>
                                <form class="user"
method="POST" action="{{ route('login') }}">
                                    @csrf
                                    <div
class="form-group">
                                        <input
type="text" class="form-control form-control-
user @error('nip') is-invalid @enderror"
name="nip" value="{{ old('nip') }}"

id="exampleInputEmail"                                aria-
describedby="emailHelp"
placeholder="Enter NIP">
                                    @error('nip')
```


2. Controller Super Admin

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Http\Requests\SuperuserRequest;
use App\Models\User;
use Illuminate\Support\Facades\Hash;

class UserController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        return view('superuser.adding');
    }

    /**
     * Show the form for creating a new
    resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        $users = User::orderBy('created_at',
'desc')->get()->except(auth()->user()->id);

        return view('superuser.superuser',
compact('users'));
    }

    /**
     * Store a newly created resource in
    storage.
     *
     * @param \Illuminate\Http\Request
    $request
     * @return \Illuminate\Http\Response
     */
    public function store(SuperuserRequest
$request)
    {
        $input = $request->all();
```

```

        $input['password']
Hash::make($request->password);

        $user = User::create($input);
        $user->assignRole($request->role);

        return redirect()-
>route('user.create')->with('success', 'Akun
telah ditambah');

    }

    /**
     * Display the specified resource.
     *
     * @param int $id
     * @return \Illuminate\Http\Response
     */
    public function show($id)
    {
        //
    }

    /**
     * Show the form for editing the specified
    resource.
     *
     * @param int $id
     * @return \Illuminate\Http\Response
     */
    public function edit($id)
    {
        $user = User::find($id);

        return view('superuser.edit',
compact('user'));
    }

    /**
     * Update the specified resource in
    storage.
     *
     * @param \Illuminate\Http\Request
    $request
     * @param int $id
     * @return \Illuminate\Http\Response
     */
    public function update(Request $request,
$id)
    {
        $user = User::find($id);

```

```

        $request->validate([
            'name' =>
'required|min:3|max:255',
            'nip' =>
'required|min:8|max:255|unique:users,nip, '.$u
ser->id,
            'password' => 'string|min:8',
            'role' => 'required',
        ]);

        // dd($request->name);
        $user->name = $request->name;
        $user->nip = $request->nip;
        if($request->password){
            $user->password =
Hash::make($request->password);
        }
        $user->save();

        $user->assignRole($request->role);

        return redirect()-
>route('user.create')->with('success', 'Data
akun berhasil di ubah');
    }

    /**
     * Remove the specified resource from
storage.
     *
     * @param int $id
     * @return \Illuminate\Http\Response
     */
    public function destroy($id)
    {
        User::destroy($id);
        return redirect()-
>route('user.create')->with('success', 'data
telah dihapus');
    }
}

```


3. Tampilan Tabel User

```
@extends('layouts.app')
@section('title')
    Pelanggan
@endsection
@section('content')
@if (session('success'))
    <div class="alert alert-success"
    role="alert">
        {{session('success')}}
    </div>
@endif

<div class="card shadow mb-4">
    <div class="card-header py-3">
        <h6 class="m-0 font-weight-bold text-
primary">Akun</h6>
    </div>
    <div class="card-body">
        <!-- Button trigger modal -->
        <div class="float-right mb-2">
            <button class="btn btn-primary
btn-modal" data-href="{{ route('user.index')
}}" data-container=".my-modal">Tambah
Akun</button>
        </div>

        <div class="table-responsive">
            <table class="table table-
bordered" id="datatable data-table"
width="100%" cellpadding="0">
                <thead>
                    <tr>
                        <th>No</th>
                        <th>NIP</th>
                        <th>Nama</th>
                        <th>aksi</th>
                    </tr>
                </thead>
                <tbody>
                    @foreach ($users as $key
=> $value)
                        <tr>
                            <td>{{ $loop-
iteration}}</td>
                            <td>{{ $value->nip ??
''}}</td>
                            <td>{{ $value->name ??
''}}</td>
                            <td>
```

```

                                <form
action="{{ route('user.destroy', [$value-
>id]) }}" method="POST">
                                @csrf

@method('delete')

                                <button
class="btn btn-primary btn-sm btn-modal" data-
href="{{ route('user.edit', [$value->id]) }}"
data-container=".my-modal">edit</button>
                                <button
class="btn btn-danger btn-sm btn-delete"
type="submit">delete</button>
                                </form>
                                </td>
                                </tr>
                                @endforeach
                                </tbody>
                                </table>
                                </div>
                                </div>
</div>
<div class="modal fade my-modal" tabindex="-1"
role="dialog"
labelledby="exampleModalLabel"
hidden="true"></div>
@endsection
@section('js')
<script
src="https://cdn.jsdelivr.net/npm/@popperjs/c
ore@2.9.2/dist/umd/popper.min.js"
integrity="sha384-
IQsoLX15PILFhosVNubq5LC7Qb9DXgDA9i+tQ8Zj3iwWA
wPtgFTxbJ8NT4GN1R8p"
crossorigin="anonymous"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5
.0.1/dist/js/bootstrap.min.js"
integrity="sha384-
Atwg2Pkwv9vp0ygtnlJAojH0nYbwNjLPhwyoVbhoPwBhj
QPR5VtM2+xf0Uwh9KtT"
crossorigin="anonymous"></script>
@endsection

```

4. Controller Dashboard Utama

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Pelanggan;
use Carbon\Carbon;

class HomeController extends Controller
{
    /**
     * Create a new controller instance.
     *
     * @return void
     */
    public function __construct()
    {
        $this->middleware('auth');
    }

    /**
     * Show the application dashboard.
     *
     * @return
     */
    public function index()
    {
        $pelangganm = Pelanggan::select('id',
        'created_at')
        -
        >whereYear('created_at', date('Y'))
        -
        >get()
        -
        >groupBy(function($date){

        return Carbon::parse($date->created_at)-
        >format('m');

        });

        $pelangganmcount = [];
        $pelangganArrm = [];
        foreach($pelangganm as $key =>
        $value) {
            $pelangganmcount[(int)$key] =
            count($value);
        }
    }
}
```

```

        $month = ['Jan', 'Feb', 'Mar', 'Apr',
'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct',
'Nov', 'Dec'];

        for($i = 1; $i <= 12; $i++) {
            if(!empty($pelangganmcount[$i])){
                $pelangganArrm[$i]['count'] =
$pelangganmcount[$i];
            } else {
                $pelangganArrm[$i]['count'] =
0;
            }
            $pelangganArrm[$i]['month'] =
$month[$i-1];
        }

        $datam = [];
        foreach($pelangganArrm as $key =>
$value) {
            $datam[$key] = $value['count'];
        }

        $tahun = Carbon::now()->year;
        // dd($tahun);

        $pelangganw = Pelanggan::select('id',
'created_at')
->whereYear('created_at', date('Y'))
->whereMonth('created_at', date('m'))
->get()
->groupBy(function($date){

return Carbon::parse($date->created_at)-
>format('W');
});

        $start = Carbon::now()-
>startOfMonth()->week-1;
        $this_month = Carbon::now()->month;

        $pelangganwcount = [];
        $pelangganArrw = [];
        foreach($pelangganw as $key =>
$value) {
            $pelangganwcount[(int)$key] =
count($value);
        }

```

```

        if($this_month == 2){
            for($i = $start; $i <= $start+3;
$i++){

if(!empty($pelangganwcount[$i])){

$pelangganArrw[$i]['count'] =
$pelangganwcount[$i];
            } else {

$pelangganArrw[$i]['count'] = 0;
            }
        }
    }else{
        for($i = $start; $i <= $start+4;
$i++){

if(!empty($pelangganwcount[$i])){

$pelangganArrw[$i]['count'] =
$pelangganwcount[$i];
            } else {

$pelangganArrw[$i]['count'] = 0;
            }
        }
    }
    $dataw = [];
    foreach ($pelangganArrw as $key =>
$value) {
        $dataw[$key] = $value['count'];
    }

    return view('home', compact('datam',
'dataw', 'tahun', 'start', 'this_month'));
}
}

```

5. Tampilan Dashboard Utama di Semua Role

```
@extends('layouts.app')
@section('title')
    Dashboard
@endsection

@section('content')
    <div id="container">

        </div>
    <br>
    <div id="chart-pelanggan">

        </div>
    @endsection

@section('js')
    <script
    src="https://code.highcharts.com/highcharts.s
    rc.js"></script>
    <script
    src="https://code.highcharts.com/modules/expo
    rting.js"></script>
    <script
    src="https://code.highcharts.com/modules/expo
    rt-data.js"></script>
    <script
    src="https://code.highcharts.com/modules/acce
    ssibility.js"></script>
    @if($this_month == 2)
    <script>
        Highcharts.chart('container', {
            title: {
                text: 'Grafik Aktivasi Mingguan'
            },
            xAxis: {
                categories: ['minggu 1', 'minggu
                2', 'minggu 3', 'minggu 4']
            },
            yAxis: {
                min: 0,
                title : { text : 'data masuk
                mingguan' }
            },
            plotOptions: {
                line: {
                    dataLabels: {
                        enabled: true
                    },
                    enableMouseTracking: false
                }
            }
        });
    </script>
    @endif
    </script>
endsection
```

```

    }
  },
  series: [{
    type: 'line',
    name: 'pelanggan',
    data: [!!! $dataw[$start] !!],
    {!!! $dataw[$start+1] !!}, {!!!
    $dataw[$start+2] !!}, {!!! $dataw[$start+3]
    !!}],
    marker: {
      enabled: false
    },
    // states: {
    //   hover: {
    //     lineWidth: 0
    //   }
    // },
    // enableMouseTracking: false
  // }, {
  //   type: 'scatter',
  //   name: 'Observations',
  //   data: [1, 1.5, 2.8, 3.5, 3.9,
4.2],
  //   marker: {
  //     radius: 4
  //   }
  // }
  ]});
</script>
@else
<script>
  Highcharts.chart('container', {
    title: {
      text: 'Grafik Aktivasi Mingguan'
    },
    xAxis: {
      categories: ['minggu 1', 'minggu
2', 'minggu 3', 'minggu 4', 'minggu 5']
    },
    yAxis: {
      min: 0,
      title : { text : 'data masuk
mingguan' }
    },
    plotOptions: {
      line: {
        dataLabels: {
          enabled: true
        },
        enableMouseTracking: false
      }
    }
  }

```

```

    },
    series: [{
        type: 'line',
        name: 'pelanggan',
        data: [!!! $dataw[$start] !!],
        {!! $dataw[$start+1] !!}, {!!
        $dataw[$start+2] !!}, {!! $dataw[$start+3]
        !!}, {!! $dataw[$start+4] !!}],
        marker: {
            enabled: false
        },
        // states: {
        //     hover: {
        //         lineWidth: 0
        //     }
        // },
        // enableMouseTracking: false
    // }, {
    //     type: 'scatter',
    //     name: 'Observations',
    //     data: [1, 1.5, 2.8, 3.5, 3.9,
4.2],
    //     marker: {
    //         radius: 4
    //     }
    // }
    ]
});
</script>
@endif
<script>
    Highcharts.chart('chart-pelanggan', {
        chart: {
            type: 'line'
        },
        title: {
            text: 'Grafik Aktivasi {{$tahun}}'
        },
        xAxis: {
            categories: ['Jan', 'Feb', 'Mar',
'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep',
'Oct', 'Nov', 'Dec']
        },
        yAxis: {
            min: 0,
            title: {
                text: 'Banyak Data Masuk'
            }
        },
        plotOptions: {
            line: {
                dataLabels: {

```



```

        enabled: true
    },
    enableMouseTracking: false
}
},
series: [{
    name: 'Pelanggan',
    data: [!!! $datam['1'] !!], {!!
$datam['2'] !!}, {!! $datam['3'] !!}, {!!
$datam['4'] !!}, {!! $datam['5'] !!}, {!!
$datam['6'] !!}, {!! $datam['7'] !!}, {!!
$datam['8'] !!}, {!! $datam['9'] !!}, {!!
$datam['10'] !!}, {!! $datam['11'] !!}, {!!
$datam['12'] !!}]
    }]
});
</script>
@endsection

```

6. Controller Data Pelanggan

```
<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use App\Models\Pelanggan;
use Illuminate\Support\Facades\File;
use File as FileO;
use ZipArchive;

class PelangganController extends Controller
{
    /**
     * Display a listing of the resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function index()
    {
        $pelanggan =
        Pelanggan::orderBy('updated_at', 'desc')->get();

        $uncheck =
        Pelanggan::where('readable', '!=', 'check')->get()->count();

        return view('pelanggan.index',
        compact('pelanggan', 'uncheck'));
    }

    /**
     * Show the form for creating a new
     resource.
     *
     * @return \Illuminate\Http\Response
     */
    public function create()
    {
        return view('pelanggan.create');
    }

    /**
     * Store a newly created resource in
     storage.
     *
     * @param \Illuminate\Http\Request
     $request
     * @return \Illuminate\Http\Response
     */
}
```

```

public function store(Request $request)
{
    // dd($request->all());
    $messages = [
        "file1.max" => "Maksimal 3 file."
    ];

    $request->validate([
        'nomor_so' => 'required',
        'nama' => 'required|min:3|max:255',
        'alamat' => 'required|min:3|max:255',
        'ptl' => 'required',
        'file1.*' => 'mimes:pdf|max:5000',
        'file1' => 'max:3',
    ], $messages);

    $input = $request->only(['nomor_so', 'nama', 'alamat', 'status', 'ptl', 'file1']);
    $input['activator_id'] = auth()->user()->id;

    if ($request->hasFile('file1')) {
        foreach ($request->file('file1') as $key => $pdf) {
            $name = substr($pdf->getClientOriginalName(), 0, -4).'.'.$pdf->getClientOriginalExtension();
            $pdf->move(public_path('asset/file'), $name);
            $data[] = $name;
        }
    }
    $input['file1'] = json_encode($data);

    // dd($input);
    Pelanggan::create($input);

    return redirect()->route('pelanggan.index')->with('status', 'Pelanggan berhasil ditambahkan');
}

/**
 * Display the specified resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */

```

```

public function show($id)
{
    //
}

/**
 * Show the form for editing the specified
resource.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function edit($id)
{
    $data = Pelanggan::find($id);
    return view('pelanggan.edit',
compact('data'));
}

/**
 * Update the specified resource in
storage.
 *
 * @param \Illuminate\Http\Request
$request
 * @param int $id
 * @return \Illuminate\Http\Response
 */
public function update(Request $request,
$id)
{
    $messages = [
        "file1.max" => "Maksimal 3 file."
    ];

    $request->validate([
        'nomor_so' => 'required',
        'nama' => 'required',
        'required|min:3|max:255',
        'alamat' => 'required',
        'required|min:3|max:255',
        'status' => 'required',
        'ptl' => 'required',
        'file1.*' => 'mimes:pdf|max:5000',
        'file1' => 'max:3',
    ], $messages);

    $pelanggan = Pelanggan::find($id);
    $input = $request->all();
    // dd($input);
}

```

```

        $sold_file1 = $pelanggan->file1;
        if($request->hasFile('file1')) {
            if($sold_file1 != null) {
                foreach
(json_decode($sold_file1) as $key => $pdf) {

File::delete('asset/file/'.$pdf);

                }
            }
            foreach ($request->file('file1')
as $key => $pdf) {
                $name = substr($pdf->getClientOriginalName(), 0, -4).'.'.$pdf->getClientOriginalExtension();
                $pdf->move(public_path('asset/file'), $name);
                $data[] = $name;
            }
            $input['file1'] =
json_encode($data);
            $input['readable'] = 'unchecked';
        } else {
            unset($input['file1']);
        }

        $pelanggan->update($input);

        return redirect()->route('pelanggan.index')->with('status',
'Data pelanggan berhasil di ubah');
    }

/**
 * Remove the specified resource from
storage.
 *
 * @param int $id
 * @return \Illuminate\Http\Response
 */
    public function destroy(Pelanggan
$pelanggan)
    {
        if($pelanggan->file1 != null) {
            foreach (json_decode($pelanggan->file1) as $key => $pdf) {

File::delete('asset/file/'.$pdf);

            }
        }

        $pelanggan->delete();
    }

```

```

        return redirect()-
>route('pelanggan.index')->with('status',
'Pelanggan berhasil dihapus');
    }

    public function getDownload(Pelanggan
$pelanggan)
    {
        $headers = ["Content-
Type"=>"application/zip"];
        $fileName = $pelanggan->nama.'-
'.rand().'zip';

        $zip = new ZipArchive;

        if ($zip->open(public_path($fileName),
ZipArchive::CREATE) === TRUE)
        {
            $files = [];

            foreach(json_decode($pelanggan-
>file1) as $key => $value){
                $files[] = public_path().
"/asset/file/" . $value;
            }

            foreach ($files as $key => $value)
            {
                $relativeNameInZipFile =
str_replace('full',$key,basename($value));
                $zip->addFile($value,
$relativeNameInZipFile);
            }

            $zip->close();
        }

        if(auth()->user()-
>hasRole('maintainer')){
            $pelanggan->readable = 'check';
            $pelanggan->save();
        }

        return response()-
>download(public_path($fileName), $fileName,
$headers);
    }
}

```

7. Tampilan Tabel Pelanggan di Semua Role

```
@extends('layouts.app')
@section('title')
    Pelanggan
@endsection
@section('content')
@if (session('status'))
    <div class="alert alert-success"
    role="alert">
        {{session('status')}}
    </div>
@endif

<div class="card shadow mb-4">
    <div class="card-header py-3">
        <h6 class="m-0 font-weight-bold text-
        primary">Pelanggan @role('maintainer')
        @if($unchecked == 0)| Seluruh pelanggan telah
        dicek! @else| Ada {{ $unchecked }} pelanggan yang
        belum dicek! @endif @endrole</h6>
    </div>
    <div class="card-body">
        <!-- Button trigger modal -->
        @role('activator')
        <div class="float-right mb-2">
            <button class="btn btn-primary
            btn-modal" data-href="{{
            route('pelanggan.create') }}" data-
            container=".my-modal">Tambah
            Pelanggan</button>
        </div>
        @endrole
        <div class="table-responsive">
            <table class="table table-
            bordered" id="datatable data-table"
            width="100%" cellpadding="0">
                <thead>
                    <tr>

@role('activator|maintainer')
                        <th>Mark as
read?</th>

                        @endrole
                        <th>Nomor SO</th>
                        <th>Pelanggan</th>
                        <th>Status</th>
                        <th>Alamat</th>
                        <th>Tanggal
Upload</th>

                        <th>Pelaksana</th>
                    </tr>
                </thead>
            </table>
        </div>
    </div>
</div>
```

```

@role('activator|maintainer')
    <th>Aksi</th>
    <th>files?</th>
    @endrole
</tr>
</thead>
<tbody>
    @foreach ($pelanggan as
$key => $value)
        <tr>

@role('activator|maintainer')
    <td>
        <input
id="readable" name="readable" data-id="{{
$value->id }}" type="checkbox" aria-
label="Checkbox for following text input"
@if($value->readable == 'check') checked
@endif disabled="disabled">
        </td>
    @endrole
    <td>{{ $value-
>nomor_so ?? ' ' }}</td>
    <td>{{ $value->nama
}}</td>
    <td><span
class="badge badge-{{ $value->status ==
'AKTIF' ? 'success' : 'warning' }}">{{
$value->status }}</span></td>
    <td>{{ $value->alamat
}}</td>
    <td>{{ $value-
>updated_at->format('d-M-Y') }}</td>
    <td>{{ $value->ptl
}}</td>
@role('activator|maintainer')
    <td>
@role('activator|maintainer')
        <form
action="{{route('pelanggan.destroy', [$value-
>id]) }}" method="POST">
            @csrf

@method('delete')
@role('activator')
        <button
class="btn btn-primary btn-sm btn-modal"
data-href="{{ route('pelanggan.edit',
[$value->id]) }}" data-container=".my-
modal">edit</button>

```



```

                                <button
class="btn btn-danger btn-sm btn-delete"
type="submit">delete</button>
                                @endrole

@role('activator|maintainer')
                                @if($value-
>file1 != null)
                                <a
onclick="reload('{{route('pelanggan.download'
, [$value->id])}}')" href="#" class="btn btn-
info btn-sm">download</a>
                                @endif
                                @endrole
                                </form>
                                @endrole
                                </td>
                                <td>
                                @if($value-
>file1)
                                @foreach
(json_decode($value->file1) as $pdf)
                                <a
href="{{asset('asset/file/'.$pdf)}}"
target="_blank">{{ $pdf }}</a>
                                @endforeach
                                @endif
                                </td>
                                @endrole
                                </tr>
                                @endforeach
                                </tbody>
                                </table>
                                </div>
                                </div>
                                </div>
                                <div class="modal fade my-modal" tabindex="-
1" role="dialog" aria-
labelledby="exampleModalLabel" aria-
hidden="true"></div>
                                @endsection
                                @section('js')
                                <script>
                                function reload(url)
                                {
                                window.open(url, '_blank');
                                window.focus();
                                location.reload();
                                }
                                </script>
                                @endsection

```

8. Fitur Tambah Pelanggan

```
<!-- Modal Tambah -->
<div class="modal-dialog" role="document">
  <div class="modal-content">
    <div class="modal-header">
      <h5 class="modal-title"
id="exampleModalLabel">Menambah Pelanggan
Baru</h5>
      <button class="close"
type="button" data-dismiss="modal" aria-
label="Close"><span aria-
hidden="true">x</span></button>
    </div>
    <div class="modal-body">
      <form action="{{
route('pelanggan.store') }}" method="POST"
enctype="multipart/form-data">
        @csrf
        <div class="mb-3">
          <label
for="exampleInputEmail1" class="form-
label">Nomor SO</label>
          <input type="text"
name="nomor_so" class="form-control"
required>
        </div>
        <div class="mb-3">
          <label
for="exampleInputEmail1" class="form-
label">Nama Pelanggan</label>
          <input type="text"
name="nama" class="form-control" required>
        </div>
        <div class="mb-3">
          <label
for="exampleInputEmail1" class="form-
label">Alamat</label>
          <input type="text"
name="alamat" class="form-control" required>
        </div>
        <label for="text"
class="form-label">Status</label>
        <div class="form-check">
          <input class="form-check-
input" type="radio" name="status"
id="flexRadioDefault1" value="AKTIF">
          <label class="form-check-
label" for="flexRadioDefault1">
            Aktif
          </label>
        </div>
      </form>
    </div>
  </div>
</div>
```

```

        </div>
        <div class="form-check">
            <input class="form-check-
input" type="radio" name="status"
id="flexRadioDefault2" checked
value="NONAKTIF">
                <label class="form-check-
label" for="flexRadioDefault2">
                    Non Aktif
                </label>
            </div>
            <div class="mb-3">
                <label
for="exampleInputEmail1" class="form-
label">Pelaksana Tugas</label>
                <input type="text"
name="ptl" class="form-control" id="ptl"
required>
            </div>
            <br>
            <div class="mb-3">
                <label for="formFile"
class="form-label">Unggah Dokumen</label>
                <input class="form-
control" type="file" id="formFile"
name="file1[]" multiple="true" required>
            </div>
            <div class="modal-
footer"><button class="btn btn-secondary"
type="button" data-
dismiss="modal">tutup</button><button
class="btn btn-primary"
type="submit">Simpan</button></div>
        </form>
    </div>
</div>
</div>

```

9. Fitur Edit Pelanggan

```
<!-- Modal Edit -->
<div class="modal-dialog" role="document">
  <div class="modal-content">
    <div class="modal-header">
      <h5 class="modal-title"
id="exampleModalLabel">Edit Pelanggan</h5>
      <button class="close"
type="button" data-dismiss="modal" aria-
label="Close"><span aria-
hidden="true">×</span></button>
    </div>
    <div class="modal-body">
      <form action="{{
route('pelanggan.update', [$data->id]) }}"
method="POST" enctype="multipart/form-data"
id="editForm">
        @csrf
        @method('PUT')
        <div class="mb-3">
          <label
for="exampleInputEmail1" class="form-
label">Nomor SO</label>
          <input type="text"
name="nomor_so" class="form-control"
id="nomor_so" value="{{ old('nomor_so') }}"
$data->nomor_so }}" required>
        </div>
        <div class="mb-3">
          <label
for="exampleInputEmail1" class="form-
label">Nama Pelanggan</label>
          <input type="text"
name="nama" class="form-control" id="nama"
value="{{ old('nama') }}" $data->nama }}"
required>
        </div>
        <div class="mb-3">
          <label
for="exampleInputEmail1" class="form-
label">Alamat</label>
          <input type="text"
name="alamat" id="alamat" class="form-
control" value="{{ old('alamat') }}" $data-
>alamat }}" required>
        </div>
        <label for="text"
class="form-label">Status</label>
        <div class="form-check">
```

```

        <input class="form-check-
input" type="radio" name="status"
id="stat_aktif" value="AKTIF" @if($data-
>status == 'AKTIF') checked @endif>
        <label class="form-check-
label" for="flexRadioDefault1">
            Aktif
        </label>
    </div>
    <div class="form-check">
        <input class="form-check-
input" type="radio" name="status"
id="stat_nonaktif" value="NONAKTIF"
@if($data->status == 'NONAKTIF') checked
@endif>
        <label class="form-check-
label" for="flexRadioDefault2">
            Non Aktif
        </label>
    </div>
    <div class="mb-3">
        <label
for="exampleInputEmail1" class="form-
label">Pelaksana Tugas</label>
        <input type="text"
name="ptl" class="form-control" id="ptl"
value="{ { old('ptl') ?? $data->ptl } }"
required>
    </div>
    <br>
    <div class="mb-3">
        <label for="formFile"
class="form-label">Unggah Dokumen</label>
        <input class="form-
control" type="file" id="formFile"
name="file1[]" multiple="true">
    </div>
    <div class="modal-footer">
        <button class="btn btn-
secondary" type="button" data-
dismiss="modal">tutup</button>
        <button class="btn btn-
primary" type="submit">Simpan</button>
    </div>
</form>
</div>
</div>
</div>

```


BAB IV PENUTUP

A. Kesimpulan

Setelah melaksanakan serangkaian kegiatan Praktik Industri di PT. Indonesia Comnets Plus Surabaya yang dilaksanakan 25 Januari – 26 Febuari 2021, Penulis mendapatkan penugasan mengimplementasikan website serah terima dokumen aktivasi yang kami buat. Penulis mendapatkan hasil sebagai berikut:

1. Penulis berhasil mengimplementasikan aplikasi Serah Terima Dokumen Aktivasi berbasis website.
2. Website Serah Terima Dokumen Aktivasi dapat memudahkan dalam penyerahan dokumen aktivasi yang dapat dilihat oleh berbagai divisi.
3. Ada 4 Tipe user yang diterapkan pada website ini, Setiap user memiliki tugas dan fungsinya masing masing. Berikut tugas dan fungsi masing masing tipe user:
 - a. Super Admin
Super Admin dapat mengelola user, baik menambahkan, edit, dan hapus user.
 - b. Tim Aktivasi
User dari tim aktivasi dapat mengupload dokumen pelanggan yang dapat dilihat oleh user yang lain, selain itu tim aktivasi dapat mengelola dokumen pelanggan, Mulai dari menambahkan, mengupdate, menghapus dokumen.
 - c. Tim Pemeliharaan
User tim pemeliharaan dapat melihat dokumen yang di tambahkan oleh tim aktivasi. Lalu setelah tim pemeliharaan melihat dokumen maka akan

mengirimkan notifikasi untuk tim aktivasi karena telah dilihat. Selain itu tim pemeliharaan memiliki fitur download dokumen yang tidak dimiliki user lain.

d. Tim Sales

User tim sales memiliki fitur melihat dokumen pelanggan yang di upload oleh tim aktivasi saja.

4. Pada dashboard terdapat grafik mingguan yang melaporkan banyak dokumen yang telah di upload.

B. Saran

Adapun beberapa saran yang dapat diberikan penulis sebagai pertimbangan pengembangan aplikasi, yaitu:

- 1.

DAFTAR PUSTAKA

- Blazing, A. (2019). *Pemrograman Windows dan Web Dengan C# & ASP.Net Core MVC Menggunakan Visual Studio 2017*. Klik Solusi Publisher.
- Elgamar. (2020, July). *Buku Ajar Konsep Dasar Pemrograman Website Dengan PHP*. Ahlimedia Book.
- Enterprise, J. (2019). *PHP untuk Programmer Pemula*. Elex Media Komputindo.
- Hery Sulistiono, S. M. (2018). *Coding Mudah dengan CodeIgniter, JQuery, Bootstrap, dan Datatable*. Elex Media Komputindo.
- Solichin, A. (2016). *Pemrograman Web dengan PHP dan MySQL*. Penerbit Budi Luhur.
- Yudhanto, Y., & Prasetyo, H. A. (2019). *Mudah Menguasai Framework Laravel*. Elex Media Komputindo.

LAMPIRAN

Lampiran 1 Lembar Nilai Praktik Industri

Lampiran 2 Surat Permohonan Ijin Praktik Industri

Lampiran 3 Surat Balasan

Lampiran 4 Lembar Bimbingan Praktik Insustri

Lampiran 5 Dokumentasi Kegiatan Praktik Industri