

## BAB III

### PELAKSANAAN KERJA MAGANG

#### 3.1 Kedudukan dan Koordinasi

Dalam melaksanakan program magang yang dipersembahkan oleh Orange Media Group Pte. Ltd., peran yang dipegang oleh peserta berfokus pada posisi magang sebagai bagian dari tim pengembangan web atau *web developer internship* yang dibimbing dan diarahkan oleh supervisor, yaitu Miss Nicole. Peran ini menuntut serangkaian tugas yang beragam dan tanggung jawab yang luas dalam hal pengembangan *website* perusahaan, yaitu melakukan desain ulang, menerapkan antarmuka baru pada situs *web* perusahaan, menciptakan situs *web* baru, menambahkan fitur-fitur tambahan pada situs, serta melakukan pemeliharaan pada aspek *front-end* maupun *back-end* dari situs milik perusahaan.

Semua tugas tersebut tidak hanya dijalankan semata untuk memenuhi kebutuhan operasional perusahaan, namun juga merupakan bagian integral dari strategi bisnis yang bertujuan untuk memperbarui, meningkatkan, menjaga relevansi situs web perusahaan, dan berkompetisi untuk mengalahkan perusahaan kompetitor. Sebagai contoh, proses desain ulang situs web diperlukan untuk menghidupkan kembali penampilan situs yang mungkin telah ketinggalan zaman, pembentukan situs web baru diperlukan untuk mengakomodasi konten-konten terbaru dari perusahaan dan mitra kerjanya guna meningkatkan daya saing untuk melawan kompetitor perusahaan, penambahan fitur-fitur baru pada situs *web* menjadi penting sebagai sarana untuk memperluas jangkauan promosi perusahaan, serta pemeliharaan rutin pada kedua sisi, baik *front-end* maupun *back-end*, merupakan sebuah kunci untuk menjaga kelancaran dan keamanan operasional situs *web* perusahaan.

Dalam menjalankan setiap tugas, peserta magang tidak hanya bertindak sebagai individu yang bekerja secara mandiri, tetapi juga diberikan dukungan penuh dari supervisi yang bersedia mengarahkan tugas dan tanggung jawab peserta magang, serta berkolaborasi dengan divisi UI/UX, yang memiliki keahlian khusus dalam

desain antarmuka pengguna dan pengalaman pengguna juga menjadi bagian integral dari proses pengembangan. Divisi UI/UX bukan hanya bertanggung jawab untuk merancang tampilan situs *web*, tetapi juga bertugas untuk menyusun konten-konten yang akan ditampilkan di dalamnya. Dengan demikian, kolaborasi antara peserta magang dan divisi UI/UX menjadi penting dalam memastikan bahwa setiap aspek dari pengembangan situs web ini terintegrasi secara mulus dan memenuhi standar yang telah ditetapkan oleh perusahaan.

Proses pengembangan *website* dalam hal ini menggunakan berbagai bahasa pemrograman, antara lain HTML, CSS, JavaScript, dan PHP. Alasan penggunaan bahasa pemrograman tersebut adalah bahasa pemrograman tersebut terkenal, mudah dipahami, dan mudah untuk diintegrasikan ke dalam berbagai proyek perusahaan. HTML serta CSS berperan untuk tampilan *front-end* dari *website*. JavaScript diperlukan apabila terdapat sebuah animasi yang diperlukan dalam sebuah *webpage*, misalnya pembuatan animasi *scroll website* dan *banner slider*. PHP dalam hal ini diperlukan untuk pemrograman *back-end*. *Website* yang dibuat dalam hal ini harus memenuhi salah satu syarat penting, yaitu mudah dipahami baik dari sisi *front-end* (untuk *user* yang nantinya berkunjung ke *website* perusahaan), maupun *back-end* (untuk beberapa karyawan internal perusahaan yang memiliki akses ke *back-end website*). Adapun divisi yang akan menggunakan *website* ini untuk kepentingannya masing-masing adalah sebagai berikut.

1. CEO: CEO dari perusahaan memiliki hak untuk memantau data yang dimasukkan ke dalam *website*.
2. Investor: Investor perusahaan memiliki hak untuk memantau data yang dimasukkan ke dalam *website*.
3. *Journalist*: *Journalist* memiliki hak untuk menambahkan konten berita ke dalam *website* perusahaan.
4. *Marketing / Sales*: *Marketing / Sales* memiliki hak untuk menambahkan konten ke dalam *website*.
5. *Advertising*: *Advertising* memiliki hak untuk memasukan iklan dan *upcoming events* perusahaan ke dalam *website*.

6. IT: IT yang dalam hal ini termasuk UI/UX memiliki hak untuk memantau data yang dimasukkan, mencegah terjadinya *error*, melakukan *quality assurance*, memasukan, dan mengubah data di dalam *website*.

Berdasarkan poin-poin di atas, dapat dilihat bahwa terdapat banyak divisi yang memerlukan *website* untuk aktivitas divisinya. Penting untuk diingat juga bahwa tidak semua anggota dalam divisi-divisi tersebut paham bahasa pemrograman, *front-end*, *back-end*, dan sebagainya. Maka, diperlukan sebuah *website* sederhana yang dapat digunakan oleh mereka untuk keperluannya masing-masing.

Dalam pelaksanaan kerja magang, diperlukan sebuah aplikasi komunikasi untuk berkoordinasi antara satu karyawan dan karyawan lainnya mengingat Sebagian besar operasional perusahaan ini dilaksanakan secara *online* atau WFH. Oleh karena itu, untuk membantu komunikasi antara satu karyawan dan karyawan lainnya digunakanlah sebuah aplikasi bernama Ring Central. Ring Central adalah sebuah aplikasi komunikasi yang dapat digunakan melalui *website*, install *software*-nya, dan di *handphone*. Dalam menjalankan operasional perusahaannya, Ring Central merupakan sebuah aplikasi yang dipakai untuk berkomunikasi antara satu karyawan dan karyawan lainnya dalam bentuk *chat* atau *text messages*. Ring Central sendiri memiliki sebuah fitur untuk mengetahui apakah karyawan tersebut sedang *online* atau *offline*. Hal ini digunakan oleh atasan untuk memantau apakah karyawannya sedang bekerja atau tidak. Selain Ring Central, perusahaan memiliki sebuah *website* yang digunakan untuk *video call*, yaitu [hi.industrialmeet.com](https://hi.industrialmeet.com). *Website* [hi.industrialmeet.com](https://hi.industrialmeet.com) digunakan apabila karyawan ingin melakukan *video call* antara satu karyawan dan karyawan lainnya.

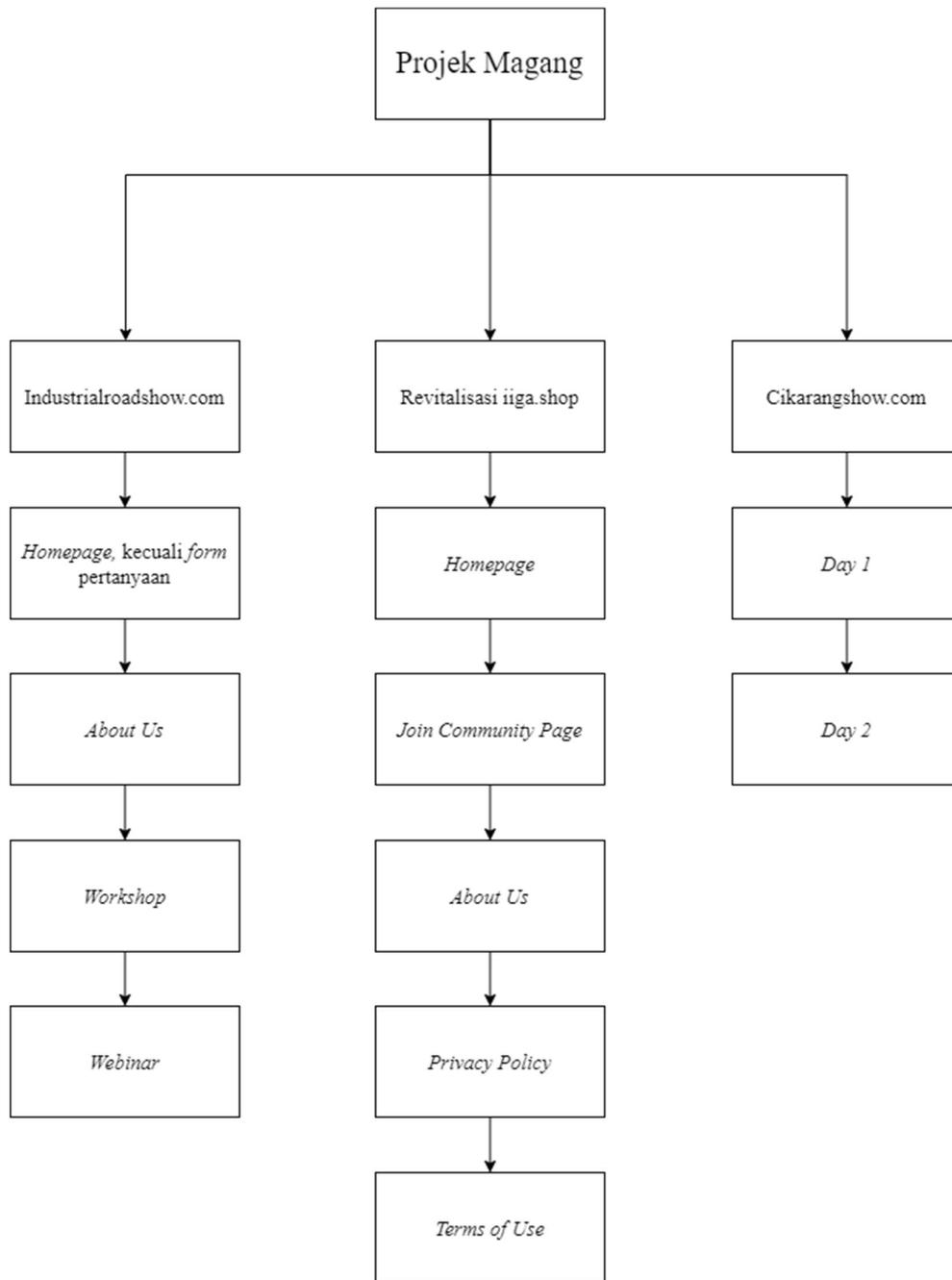
### 3.2 Tugas dan Uraian Kerja Magang

Dalam melaksanakan kerja magang sebagai *web developer intern* di Orange Media Group, peserta memiliki beberapa deskripsi pekerjaan yang dikerjakan selama program kerja magang. Berikut ini merupakan deskripsi pekerjaannya.

1. Melakukan perancangan atau desain ulang dari *website* perusahaan.
2. Melakukan penambahan fitur berupa *email marketing automation*.

3. Membuat dan merancang sebuah *website* baru untuk keperluan perusahaan.
4. Melakukan *data scrapping*.
5. Melakukan *quality assurance*.
6. Menangani *bugs* dan *error* di *website* perusahaan.
7. Membuat RSS Feeds atau iFrame yang dapat di-*embed* ke *website* lain.
8. Memberikan *update progress* terhadap perubahan yang terjadi di *website* milik perusahaan.





Gambar 3.1 Bagan Tugas yang dikerjakan Peserta Magang Dalam Projek Perusahaan

Gambar 3.1 merupakan sebuah bagan yang menunjukkan tugas dan tanggung jawab peserta magang pada saat mengerjakan proyek besar perusahaan.

Terdapat tiga buah projek besar perusahaan yang berhubungan dengan *website*, yaitu Industrialroadshow.com, revitalisasi Iiga.shop, dan Cikarangshow.com. Bagan-bagan di bagan nama projek merupakan bagan yang menunjukkan hal-hal yang dikerjakan peserta magang selama masa magang.

Berikut ini merupakan kegiatan dan tugas yang telah dilakukan oleh peserta magang selama empat bulan program kerja magang.

No.	Kegiatan	Tanggal Mulai	Tanggal Selesai
1	<i>Onboarding internship</i> , penjelasan <i>scope</i> projek, penjelasan tugas, penjelasan Trackabi, Ring Central, serta Roundcube Webmail, dan pengenalan divisi.	2 Januari 2024	9 Januari 2024
2	<i>Setup tools</i> , mempelajari <i>tools</i> , mempelajari maksud serta tujuan projek, mempelajari relevansi projek terhadap bisnis perusahaan, dan menentukan cara bagaimana menyelesaikan masalah perusahaan yang berhubungan dengan <i>website</i> .	10 Januari 2024	26 Januari 2024
3	Mempelajari <i>back-end website</i> yang menggunakan PHP.	29 Januari 2024	5 Februari 2024
4	Mencari cara untuk <i>data scrapping</i> dari berbagai <i>website</i> yang datanya akan dipakai untuk dimasukan ke dalam <i>database website</i> .	6 Februari 2024	13 Februari

5	Membuat <i>website</i> perusahaan yang digunakan untuk menampilkan <i>timeline</i> kegiatan perusahaan selama sepuluh tahun ( <i>industrialroadshow.com</i> ).	14 Februari 2024	22 Februari 2024
6	Merancang ulang <i>website</i> <i>iiga.shop</i> .	23 Februari 2024	7 Maret 2024
7	Melakukan penambahan fitur <i>email marketing automation</i> dalam <i>website</i> <i>iiga.news</i> dan menggunakannya untuk promosi <i>event</i> perusahaan.	11 Maret 2024	20 Maret 2024
8	<i>Maintenance</i> dan <i>quality assurance</i> untuk <i>iiga.news</i> .	21 Maret 2024	29 Maret 2024
9	Merancang <i>website</i> baru untuk acara perusahaan ( <i>cikarangroadshow.com</i> ).	1 April 2024	12 April 2024
10	Mempelajari tentang RSS Feeds dan <i>iFrame</i> .	15 April 2024	19 April 2024
11	Membahas dan mempelajari untuk pembuatan proyek baru <i>website</i> perusahaan yang menggabungkan berita-berita dari perusahaan Orange Media Group dan juga <i>partner</i> bisnisnya.	22 April 2024	30 April 2024

Tabel 3.1 Tabel Kegiatan Magang

Tabel 3.1 merupakan sebuah tabel yang berisi mengenai kegiatan yang telah dilalui oleh peserta magang selama periode magang. Berikut ini merupakan rincian mengenai kegiatan-kegiatan tersebut.

### **3.2.1 Onboarding Internship, Penjelasan Scope Projek, Penjelasan Tugas, Penjelasan Trackabi serta Ring Central, dan Pengenalan Divisi.**

Selama minggu pertama Januari, tepatnya pada tanggal 2 Januari 2024 sampai 9 Januari 2024, terdapat sejumlah kegiatan yang dilakukan oleh peserta magang. Berikut ini merupakan uraian kegiatan tersebut.

#### **3.2.1.1 Onboarding Internship**

Proses kerja magang di Orange Media Group Pte, Ltd dimulai pada tanggal 2 Januari 2024. Tanggal tersebut menandai langkah awal bagi peserta magang dalam menjalankan perannya sebagai bagian dari program *web developer internship* di perusahaan ini. Sebagai titik awal dari perjalanan magang, hari itu menjadi momen penting yang menandai dimulainya keterlibatan peserta magang dalam proyek-proyek pengembangan web yang akan mereka jalani selama masa magang mereka. Dengan langkah pertama ini, peserta magang diperkenalkan dengan lingkungan kerja, tim, dan tugas-tugas yang akan mereka hadapi selama periode magang.

Sebelum memasuki hari pertama magang, peserta magang menerima kontak dari salah satu karyawan senior perusahaan, yaitu Kak F.S, melalui aplikasi WhatsApp. Dengan penuh antusiasme, Kak F.S memberikan ucapan selamat atas diterimanya peserta magang untuk bergabung dengan Orange Media Group Pte. Ltd dalam posisi *web developer intern*. Tak hanya itu, Kak F.S juga memberikan informasi penting berupa *link* menuju *platform* Ring Central, Roundcube Webmail, serta email, dan kata sandi resmi perusahaan yang akan digunakan oleh peserta. Ring Central sendiri merupakan sebuah *platform* yang berbentuk baik *website* maupun *software* yang digunakan untuk melakukan *chat* bagi karyawan internal perusahaan, sedangkan Roundcube Webmail merupakan

sebuah *platform email* yang digunakan oleh perusahaan (hampir mirip seperti Gmail).

Selanjutnya, Kak F.S meminta peserta magang untuk segera melakukan *login* ke Ring Central dan Roundcube Webmail dengan menggunakan *email* dan *password* perusahaan yang baru saja diberikan. Hal ini dilakukan untuk memeriksa apakah akun tersebut dapat digunakan dengan baik atau ada masalah. Kak F.S menyarankan agar setelah berhasil masuk, peserta segera melakukan *update password*. Hal ini karena kata *password* yang diberikan cenderung sederhana dan mudah ditebak oleh pihak lain. Dengan format yang mudah terbaca seperti 'NamaKaryawan'88\$\$, keamanan akun menjadi rentan terhadap penyalahgunaan. Oleh karena itu, Kak F.S menekankan pentingnya untuk mengganti *password* sesegera mungkin guna meningkatkan keamanan akun. Peserta magang pun melakukan hal yang diminta. Pertama-tama peserta melakukan *login* ke Roundcube Webmail dan kemudian melakukan *update password* di Roundcube Webmail. Kemudian, peserta melakukan *login* ke Ring Central dan melakukan *update password* di Ring Central.

Setelah berhasil masuk ke Ring Central dan Roundcube Webmail, serta mengganti kata sandi, peserta magang disambut oleh *supervisor*, Miss Nicole. Dengan penuh keramahan, Miss Nicole menyampaikan informasi bahwa besok akan diadakan pertemuan penting secara *online* pada jam 8.00 WIB sebagai awal dari hari pertama magang. Hal ini menjadi kesempatan bagi peserta magang untuk berkenalan lebih jauh dengan dirinya dan mendapatkan arahan awal mengenai tugas dan tanggung jawab peserta selama masa magang.

Pada hari pertama, peserta magang diundang ke dalam sebuah pertemuan melalui *video call* oleh supervisor, Miss Nicole Chong. Dalam pertemuan tersebut, Miss Nicole Chong, selaku *supervisor*, memperkenalkan dirinya, termasuk nama, panggilan, dan jabatannya di perusahaan Orange Media Group Pte, Ltd. Begitu juga dengan peserta magang. Peserta magang diberi kesempatan untuk memperkenalkan diri, menyebutkan nama, panggilan, tempat kuliah, pemahaman tentang profesi sebagai seorang *web developer*, dan hal-hal terkait lainnya.

Setelah proses perkenalan selesai, Miss Nicole Chong memaparkan peraturan-peraturan perusahaan yang harus diperhatikan oleh seluruh anggota tim, termasuk para peserta magang. Salah satu aturan yang ditekankan adalah mengenai jam kerja magang. Pelaksanaan kerja magang dimulai pada pukul 08.00 WIB dan berakhir pada pukul 17.00 WIB, dengan waktu istirahat selama satu jam (total delapan jam kerja). Peserta magang dan juga para karyawannya dipersilakan mengambil waktu istirahat selama satu jam. Mereka dibebaskan untuk mengambil istirahat pada pukul berapa saja asalkan sudah di atas jam 12.00 WIB dan waktu istirahat hanya satu jam. Berdasarkan peraturan tersebut, peserta magang mengambil keputusan untuk mengambil jam istirahat pada pukul 12.00 WIB hingga 13.00 WIB dan terdapat beberapa hari tertentu di mana peserta akan mulai istirahat pada pukul 13.00 WIB hingga 14.00 WIB atau bahkan *skip break* dan keluar pada pukul 16.00 WIB.

Peraturan lain yang dijelaskan adalah mengenai hak khusus yang dimiliki oleh peserta magang sebagai *web developer intern*, yaitu akses ke data perusahaan dan *back-end* dari *website* perusahaan. Namun, dengan hak tersebut juga datang tanggung jawab besar untuk menjaga kerahasiaan data perusahaan, termasuk

akun, nama akun, dan *password*. Peserta magang diberi kepercayaan untuk tidak menyebarluaskan akun, nama akun, dan *password* perusahaan kepada siapapun kecuali diizinkan oleh *supervisor*. Selain itu, peserta magang juga tidak diizinkan untuk menghapus data dan mengirimkan data tanpa izin dari *supervisor*. Apabila terdapat data yang bocor atau bahkan hilang, peserta magang wajib bertanggung jawab atas hal tersebut dengan cara memperbaiki seperti semula. Hal ini dikarenakan apabila data bocor atau hilang perusahaan dapat dikenakan sanksi baik dari investor maupun pemerintah Singapura yang bertugas untuk melindungi data-data dari perusahaan. Sanksi tersebut dapat berupa sebuah denda besar yang angkanya dapat mencapai 10.000 SGD atau sekitar seratus dua puluh juta rupiah.

Selanjutnya, peserta magang diberi arahan dan aturan terkait *testing* di *back-end website* perusahaan. Peserta diinstruksikan untuk membuat halaman baru yang bersifat *private*, yang hanya dapat diakses oleh admin atau orang-orang khusus yang memiliki *link* akses ke halaman *private* tersebut. Tujuannya adalah untuk mencegah gangguan pada *website* perusahaan yang sedang berjalan, serta agar perbaikan atau perubahan yang sedang dilakukan tidak terlihat oleh pengguna. Apabila terjadi kesalahan yang menyebabkan kerusakan pada halaman utama *website*, peserta magang bertanggung jawab untuk memperbaikinya segera.

Kemudian, peserta magang juga diminta untuk mengaktifkan *software* Trackabi dan Ring Central, yang merupakan bagian penting dari infrastruktur kerja tim. Trackabi digunakan untuk melacak aktivitas kerja karyawan yang terdapat hubungan juga dengan gaji karyawan nantinya, sedangkan Ring Central digunakan untuk komunikasi internal perusahaan.

Terakhir, peserta magang diberi kebijakan untuk mengajukan izin tidak masuk, terlambat, atau hanya masuk setengah hari dengan alasan yang jelas, seperti sakit atau urusan keluarga, dengan syarat semua izin harus disampaikan dan disetujui oleh atasan. Hal ini bertujuan untuk memastikan transparansi dan komunikasi yang efektif dalam tim serta memberikan gaji yang adil kepada para karyawan.

Dengan demikian, hal-hal tersebutlah yang dilakukan selama *onboarding* internship yang merupakan hari pertama sekaligus hari perkenalan peserta magang.

### 3.2.1.2 Penjelasan *Scope* Proyek

Setelah menjelaskan peraturan dan regulasi perusahaan, Miss Nicole melanjutkan dengan membahas ruang lingkup proyek atau *scope* dari proyek yang akan dikerjakan. Beliau menjelaskan bahwa tujuan utama dari proyek ini adalah untuk memperbarui tampilan *front-end* dari *website* iiga.shop yang telah ada selama lebih dari 20 tahun. Secara singkat, iiga.shop merupakan situs *e-commerce* milik perusahaan yang dikenal dengan singkatan dari Industrial Guide Asia Shop. Situs ini berfungsi sebagai *platform* untuk menampilkan berbagai informasi perusahaan, termasuk produk, majalah *online*, *company listing*, *events*, *job listing*, berita milik perusahaan, dan lain-lain. Miss Nicole menjelaskan bahwa awalnya, iiga.shop hanya menggunakan Phpmyadmin dengan *front-end* sederhana atau bahkan seadanya saja. Namun, seiring berjalannya waktu, mereka menemukan cara untuk menerapkan *template front-end* yang lebih modern. Dengan adanya *template* tersebut, tampilan dari *website* ini meningkat menjadi lebih indah untuk dipandang. Sampai saat ini, *template* tersebut masih digunakan, artinya *template* tersebut telah digunakan selama lebih

dari sepuluh tahun tanpa perubahan yang signifikan dari segi tampilan *front-end*. Beliau berkata bahwa tampilan dari *iiga.shop* yang sekarang masih mudah dipahami, tetapi terlihat ketinggalan zaman dengan nuansa seperti situs web tahun 90-an. Oleh karena itu, proyek untuk merevitalisasi tampilan *front-end* dari *website* *iiga.shop* dilaksanakan.

Berdasarkan permasalahan yang telah dijelaskan di paragraf sebelumnya, hal yang menjadi perhatian adalah tidak adanya perubahan dalam tampilan *front-end* selama lebih dari sepuluh tahun, yang membuat *website* ini terasa membosankan. Oleh karena itu, proyek ini dilaksanakan untuk memberikan perubahan pada tampilan situs. *Scope* dari proyek ini mencakup kemampuan peserta magang untuk memperbarui tampilan halaman yang ada di dalam *website* tersebut, terutama halaman utama (*homepage*). Hal ini dikarenakan halaman *homepage* merupakan halaman yang paling krusial karena halaman tersebut merupakan halaman awal yang akan dilihat oleh *user* ketika mereka mengakses *website* *iiga.shop*. Oleh karena itu, fokus utamanya adalah merevitalisasi halaman *homepage*. Miss Nicole menjelaskan bahwa jika ada waktu yang cukup, peserta magang juga akan diminta untuk memperbarui halaman lainnya selain *homepage*, sebagai bagian dari upaya untuk memodernisasi keseluruhan tampilan *front-end* dari *website* ini. Perlu diingat juga bahwa proyek ini hanya untuk merevitalisasi *front-end* dari *website* saja tanpa mengganggu data yang ada di dalam *database website* tersebut. Oleh karena itu, data di dalam *database* dilarang keras untuk diubah, ditambahkan, dan dihapus tanpa seizin dari *supervisor*.

### 3.2.1.3 Penjelasan Tugas

Setelah menyampaikan gambaran lengkap mengenai peraturan perusahaan dan *scope* dari proyek yang akan dijalankan, Miss Nicole, sebagai *supervisor*, melanjutkan dengan menjelaskan secara terperinci berbagai tugas dan tanggung jawab yang akan dilaksanakan oleh peserta magang selama masa kerjanya sebagai *web developer intern*. Meskipun fokus utama peserta magang adalah merevitalisasi tampilan *front-end* dari *website* iiga.shop, namun peserta magang juga akan bertanggung jawab atas serangkaian tugas tambahan yang akan memperluas cakupan pengalaman mereka dalam dunia pengembangan web dan teknologi informasi secara keseluruhan.

Selain dari tugas utama, yaitu merevitalisasi tampilan *front-end* iiga.shop, peserta magang akan terlibat dalam pembuatan desain, penambahan fitur-fitur baru yang memperkaya pengalaman pengguna, modifikasi data, dan *data entry*. Selain itu, mereka juga akan diajak untuk memeriksa berbagai opsi *Software as a Service* (SaaS) dan alat otomatisasi lain yang dapat mendukung kemajuan bisnis perusahaan serta pengembangan *website* yang mereka kerjakan. Oleh karena itu, terkadang peserta akan diminta untuk melakukan *testing* dan *review* terhadap *tools* SaaS dan *automation tools* yang beliau temukan. Adapun tujuan dari *review* SaaS dan *automation tools* adalah supaya beliau mengetahui apa kelebihan dan kekurangan dari SaaS atau *automation tools* tersebut. Beliau perlu mengetahui bahwa apakah SaaS atau *automation tools* yang beliau temukan sesuai dengan tujuan bisnisnya dan sesuai dengan keperluan pengembangan *website*. Oleh karena itu, penting bagi beliau untuk mengetahui *review* dari sudut pandang seseorang yang paham akan IT. Alasan lain mengapa beliau meminta *review* adalah untuk memastikan bahwa terdapat karyawannya (peserta magang) yang sudah berpengalaman memakai SaaS atau *automation tools*

tersebut sehingga apabila suatu saat beliau berlangganan, peserta magang dapat menjelaskan cara menggunakannya ke karyawan lain. Alasan lain beliau memerlukan *review* adalah harga untuk berlangganan Saas dan *automation tools* tidaklah murah. Oleh karena itu, beliau memerlukan *review* yang matang mengenai kecocokan *tools* tersebut terhadap kebutuhannya, kelebihan, dan juga kekurangannya supaya beliau tidak sia-sia berlangganan Saas atau *automation tools* tersebut. Semua tugas yang dilakukan oleh peserta magang bertujuan untuk memberikan peserta magang pengalaman yang komprehensif dalam berbagai aspek teknologi informasi, memperluas wawasan mereka dan meningkatkan keterampilan dalam dunia IT.

Selanjutnya, Miss Nicole juga menekankan bahwa selain dari tanggung jawab langsung di tempat kerja, tujuan utama dari program magang ini adalah untuk memberikan kesempatan kepada peserta magang untuk terus belajar dan meningkatkan diri mereka sendiri. Oleh karena itu, meskipun mungkin ada beberapa tugas yang berada di luar jangkauan pengetahuan peserta magang atau mungkin tidak langsung terkait dengan bidang IT, mereka tetap akan ditugaskan untuk melaksanakannya. Hal ini bertujuan untuk melatih peserta magang dalam kemampuan adaptasi dan memberikan mereka kesempatan untuk mengembangkan keahlian di berbagai bidang, yang pada akhirnya akan membuat mereka menjadi profesional yang lebih berkualitas dan berpengetahuan luas.

#### **3.2.1.4 Penjelasan Trackabi, Ring Central, dan Roundcube Webmail**

Setelah menyampaikan informasi mengenai peraturan perusahaan, *scope* projek, dan tanggung jawab peserta magang, Miss Nicole, sebagai supervisor, meminta salah satu karyawannya,

yaitu Kak F.A (nama disamarkan), untuk menjelaskan mengenai dua *software* yang akan digunakan selama kerja magang di Orange Media Group Pte, Ltd, yaitu Trackabi dan Ring Central.

Pertama-tama *software* Trackabi. Sebagaimana telah dijelaskan di bab I bahwa Trackabi adalah sebuah perangkat lunak yang berfungsi seperti *timer* untuk menghitung jam kerja setiap karyawan di perusahaan. Kak F.A menjelaskan bahwa Trackabi akan digunakan setiap hari sebagai alat untuk memantau waktu masuk dan keluar karyawan. Trackabi akan diaktifkan saat karyawan memulai kerja (biasanya pada pukul 8.00 WIB) dan wajib dinonaktifkan saat jam kerja selesai (biasanya pada pukul 17.00 WIB). Namun, Trackabi tidak perlu diaktifkan jika karyawan bekerja secara *offline*. Hal ini dikarenakan *supervisor* dapat memantau secara langsung pekerjaan karyawannya. Kemudian, setelah menjelaskan apa itu Trackabi, Kak F.A meminta peserta magang untuk mengunduh *software* Trackabi melalui situs web resminya, trackabi.com. Kak F.A menegaskan dan memastikan bahwa peserta magang bukan menggunakan timer yang ada di situs web tersebut, melainkan mengunduh *software* Trackabi. Hal ini penting karena penggunaan timer yang diunduh memungkinkan akses penuh ke fitur-fitur yang diperlukan oleh perusahaan untuk melacak aktivitas karyawan.

Proses unduhan Trackabi tidak selalu mulus, dan Kak F.A memberikan petunjuk mengenai beberapa masalah yang mungkin timbul, seperti kesulitan membuka aplikasi setelah instalasi. Hal tersebut benar saja terjadi pada peserta magang sehingga peserta magang harus meng-*uninstall* terlebih dahulu *software* Trackabi dan kemudian meng-*install* kembali. Setelah berhasil mengunduh dan menginstal, peserta magang harus *login* menggunakan *email* dan *password* yang diberikan oleh perusahaan, bukan menggunakan

*email* pribadi. Hal ini dilakukan untuk memungkinkan perusahaan memantau aktivitas karyawan melalui *email* perusahaan yang disediakan.

Selain itu, Kak F.A juga memberikan aturan penggunaan Trackabi yang jelas. Pertama, Trackabi harus diaktifkan saat jam kerja dimulai (normalnya pukul 8.00 WIB) dan dinonaktifkan setelah delapan jam kerja (biasanya pada pukul 17.00 WIB). Trackabi juga wajib dinonaktifkan pada waktu istirahat selama satu jam. Oleh karena itu, apabila peserta magang terlambat lebih dari pukul 8.00, peserta magang wajib menyesuaikan bagaimana caranya agar *timer* tersebut mencapai delapan jam kerja pada hari itu. Adapun caranya adalah dengan melebihkan jam kerja terhadap jam istirahat atau melebihkan jam kerja terhadap jam selesai bekerja. Sebagai contoh, apabila karyawan baru menyalakan *timer* pada pukul 8.05 (yang berarti terlambat 5 menit), ia wajib memilih antara beristirahat selama 55 menit saja atau selesai bekerja pada pukul 17.05. Kak F.A lebih menyarankan untuk peserta mengurangi waktu istirahatnya dari pada selesai lebih lama apabila terjadi keterlambatan dan harus menyesuaikan *timer*. Terdapat sebuah regulasi lain yang ia jelaskan, yaitu *timer* Trackabi akan otomatis *pause* apabila *software* mendeteksi tidak adanya pergerakan *mouse* selama dua menit. Tidak hanya itu, *timer* tersebut akan mundur selama 5 menit sehingga peserta wajib menyesuainya lagi. Sebagai contoh, ketika *timer* menunjukkan waktu 5 jam 30 menit, tetapi selama dua menit *software* tidak mendeteksi adanya pergerakan *mouse*, *timer* tersebut akan *pause* dan mundur ke 5 jam 25 menit yang seharusnya 5 jam 32 menit. Oleh karena itu, penting untuk menjaga pergerakan *mouse* supaya tidak terjadi hal tersebut. Apabila peserta ingin ke toilet atau ada urusan yang membutuhkan waktu lebih dari dua menit, Kak F.A menyarankan untuk mem-

*pause* saja *timer* Trackabi supaya tidak mundur 5 menit dan kemudian peserta wajib menyesuaikan kembali waktu *timer* Trackabi supaya mencukupi delapan jam kerja dalam sehari. Jika *timer* tidak mencapai delapan jam kerja, karyawan akan mendapatkan pemberitahuan melalui *email* perusahaan yang memberikan peringatan bahwa waktu kerja Anda untuk hari tersebut tidak sampai delapan jam kerja.

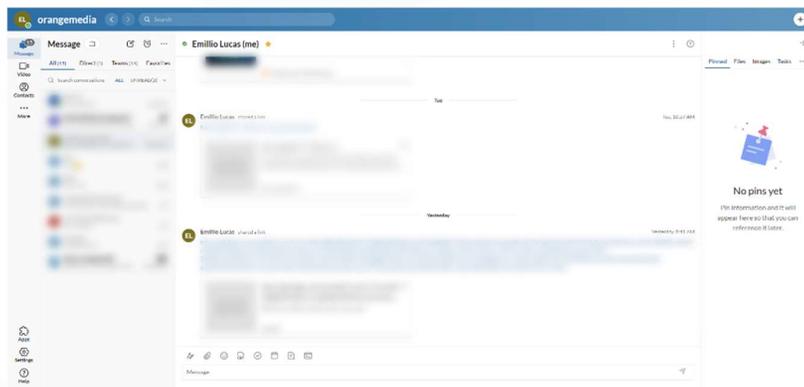
Berikutnya, Trackabi juga melakukan *screenshot* layar setiap lima menit sehingga perusahaan bisa *men-track* aktivitas karyawan dalam perusahaan tersebut. Kak. F.A menjelaskan bahwa sebenarnya tidak apa-apa bagi karyawan untuk melakukan aktivitas lain di *device* masing-masing mengingat bahwa hal tersebut merupakan hak dari masing-masing karyawan juga. Hanya saja, karyawan tetap harus memperhatikan pekerjaan dan tidak terlalu banyak membuka hal-hal di luar pekerjaan mengingat semuanya terpantau melalui Trackabi.

Kemudian, Kak F.A juga menjelaskan bahwa Trackabi terhubung langsung dengan Madam B (nama disamarkan), yang merupakan investor dari perusahaan. Madam B bertugas untuk memberikan gaji para karyawan Orange Media Group Pte. Ltd. Beliau menggunakan Trackabi untuk memantau jam kerja karyawan dan mengatur gaji mereka. Jika karyawan tidak mencapai delapan jam kerja, karyawan tersebut akan dihubungi untuk dimintai keterangan mengenai mengapa jam kerjanya tidak sampai delapan jam kerja. Apabila tidak ada alasan yang jelas, gaji mereka akan dipotong.

Terakhir, Kak F.A menegaskan pentingnya melaporkan masalah atau *bug* pada Trackabi kepada *supervisor* apabila ditemukan. Hal ini dikarenakan hal tersebut dapat memengaruhi

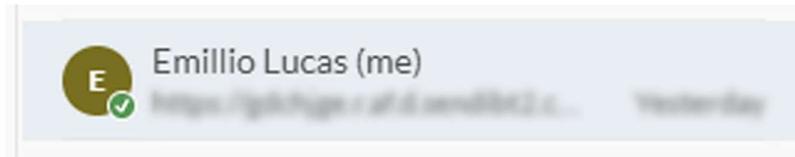
penggajian karyawan dan hanya *supervisor* yang dapat menghubungi langsung investor perusahaan untuk memberikan keterangan bahwa Trackabi peserta magang bermasalah sehingga gaji peserta magang tidak dipotong karena sudah ada keterangan yang jelas.

Selanjutnya adalah *software* Ring Central. Seperti yang sudah dijelaskan di bagian-bagian sebelumnya bahwa Ring Central merupakan sebuah *software* yang digunakan untuk melakukan *chat* sesama karyawan. *Software* ini pun digunakan oleh *supervisor* peserta magang untuk memantau apakah peserta magang *online* atau tidak. Pertama-tama, Kak F.A menjelaskan dengan meminta peserta magang membuka *website* Ring Central, yaitu [app.ringcentral.com](http://app.ringcentral.com). Setelah itu, ia meminta untuk melakukan *login* dengan menggunakan *email* dan *password* perusahaan (hal ini sudah dilakukan oleh peserta sehari sebelumnya).



Gambar 3.2 Tampilan Ring Central

Gambar 3.2 merupakan tampilan dari Ring Central dalam versi *website*. Berdasarkan gambar tersebut, dapat dilihat bahwa Ring Central berbentuk seperti *website chatting* yang berisi karyawan-karyawan internal perusahaan.

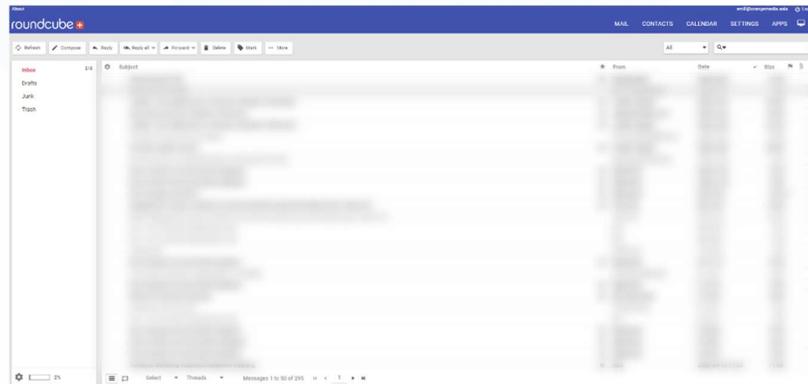


Gambar 3.3 Akun Ring Central Peserta Magang

Gambar 3.3 merupakan gambar akun peserta magang di Ring Central. Dalam hal ini, Kak F.A menjelaskan bahwa terdapat *icon* centang berwarna hijau disebelah *profile picture* akun. Hal tersebut merupakan status yang menjadi penanda bahwa karyawan sedang *online* dan hal tersebutlah yang dipantau oleh *supervisor* untuk peserta magang selain melalui Trackabi. Apabila peserta magang menutup *tab* Ring Central atau bahkan meninggalkan *device* selama lebih dari sepuluh menit, status gambar centang hijau akan berubah menjadi lingkaran yang berwarna putih yang menandakan karyawan tersebut sedang *offline*. Namun, Kak F.A menjelaskan bahwa terdapat sebuah masalah apabila menggunakan Ring Central versi *website*, yaitu meskipun karyawan masih bekerja dan tidak meninggalkan *device*-nya, terkadang statusnya bisa secara tiba-tiba *offline* dengan sendirinya. Hal tersebut diceritakan berdasarkan pengalamannya yang setiap sepuluh menit ditanya oleh *supervisor* sedang apa karena Ring Central miliknya berstatus *offline*. Hal ini kemungkinan besar disebabkan oleh karyawan yang tidak membuka *tab* Ring Central dalam jangka waktu yang lama sehingga Ring Central akan *offline* secara tiba-tiba. Untuk menghindari hal tersebut, Kak F.A menyarankan untuk peserta *download software* Ring Central tersebut supaya berbentuk aplikasi dari pada berbentuk *website*. Hal tersebut akan meminimalisir terjadinya status yang *error* tersebut. Kemudian, Kak F.A melanjutkan dengan membahas bahwa Ring Central dapat digunakan untuk membuat *group* yang memungkinkan seseorang

dapat berkomunikasi kepada lebih dari dua karyawan. Ring Central juga akan memberikan notifikasi berupa Windows Notification apabila terdapat *chat* yang masuk.

Terakhir, terdapat satu *platform* lagi yang sering digunakan oleh perusahaan. *Platform* tersebut adalah Roundcube Webmail. Seperti yang sudah dijelaskan di bagian sebelumnya bahwa Roundcube Webmail merupakan sebuah *platform email* yang dipakai oleh perusahaan. Kak F.A menjelaskan bahwa apabila terdapat suatu hal yang memerlukan *email*, gunakanlah *email* perusahaan dan *email* tersebut bisa dibuka hanya di Roundcube Webmail dan tidak bisa di Gmail biasa.



Gambar 3.4 Tampilan Roundcube Webmail

Gambar 3.4 merupakan gambar tampilan dari Roundcube Webmail. Berdasarkan gambar tersebut, dapat dilihat bahwa tampilannya menyerupai Gmail pada umumnya. Roundcube Webmail digunakan untuk *email* operasional perusahaan. Oleh karena itu *login* untuk masuk ke Roundcube Webmail wajib menggunakan *email* karyawan perusahaan. Dengan adanya Roundcube Webmail, segala hal yang berurusan dengan *email* di dalam operasional perusahaan wajib menggunakan Roundcube Webmail dan bukan Gmail biasa ataupun *email* pribadi karyawan.

### 3.2.1.5 Pengenalan Divisi

Setelah selesai *meeting*, *supervisor* peserta magang, yaitu Miss Nicole, mengundang para karyawan magang lainnya yang juga baru bergabung pada hari yang sama. Beliau memperkenalkan ke dua divisi yang berhubungan erat nantinya dengan pengembangan *website*, yaitu divisi UI/UX (yang termasuk divisi IT juga sama seperti peserta magang) dan juga divisi *sales / marketing*. Kedua divisi tersebut merupakan dua divisi yang berperan penting dalam pengembangan *website* nantinya. Tujuan diperkenalkan adalah supaya mereka saling kenal satu sama lain antar satu divisi dan divisi lainnya.

Divisi UI/UX merupakan divisi yang bertugas untuk melakukan desain, seperti mendesain majalah, poster produk, tiket acara, *banner*, hingga halaman *website*. Oleh karena itu, divisi *web developer internship* sangat berhubungan erat dengan divisi UI/UX. Hal ini dikarenakan divisi UI/UX adalah yang melakukan desain *website* yang nantinya akan diimplementasikan oleh divisi *website*. Mereka juga beranggotakan karyawan yang paham akan hal-hal IT sehingga mereka juga memahami bagaimana cara membuat *website*. Oleh karena itu, divisi *website* penting untuk menjalin komunikasi dan hubungan yang baik dengan divisi UI/UX karena kedua belah pihak harus bisa bekerja sama dengan baik agar proses revitalisasi *website* berjalan dengan baik.

Divisi lain yang berhubungan erat dengan divisi *website* adalah *sales / marketing*. Divisi *sales / marketing* merupakan divisi yang menjadi tulang punggung perusahaan ini. Tugas utama dari divisi *sales / marketing* adalah untuk menjual produk, merekrut peserta acara, merekrut perusahaan untuk dating ke *event*, memasukan berita ke dalam majalah dan juga ke dalam *website*.

Hubungan antara divisi *sales / marketing* dan divisi *website* adalah data-data yang diperlukan untuk pengembangan *website*, seperti berita, gambar, harga produk, info produk, dan sebagainya berasal dari divisi *sales / marketing* sehingga penting bagi divisi *website* untuk menjalin hubungan dan komunikasi yang baik dengan *sales / marketing* guna melaksanakan masing-masing tugas dengan baik.

Proses pengenalan divisi memakan waktu seminggu karena sangat banyak peserta-peserta magang lainnya yang juga baru direkrut oleh perusahaan. Mereka hadir pada tanggal yang berbeda-beda selama minggu pertama Januari 2024. Asal mereka juga berbeda-beda. Ada yang dari Bekasi, Riau, Bandung dan sebagainya bahkan ada juga yang berasal dari luar negeri, seperti Singapore, Malaysia, Jepang, dan sebagainya. Dengan demikian, pengenalan divisi ini dilakukan demi para karyawan dan peserta magang dapat mengenali satu sama lain supaya mereka dapat bekerja sama dengan baik.

### **3.2.2 *Setup tools* serta Mempelajari *tools*, Mempelajari Relevansi Projek terhadap Bisnis Perusahaan, dan Menentukan Bagaimana Cara Menyelesaikan Masalah Perusahaan yang Berhubungan dengan *Website*.**

Masuk minggu berikutnya, peserta magang mulai memasuki tahap persiapan untuk melaksanakan tugas-tugasnya. Beberapa hal yang dilakukan antara lain adalah *setup* serta mempelajari *tools*, memahami relevansi projek terhadap bisnis perusahaan, dan berdiskusi untuk memikirkan bagaimana menyelesaikan masalah dalam perusahaan yang berhubungan dengan IT atau *website*. Berikut ini merupakan uraian kegiatannya.

#### **3.2.2.1 *Setup* dan Mempelajari *Tools***

Dalam menjalankan tugasnya, baik itu untuk menangani proyek utama maupun tugas-tugas sampingan yang mungkin muncul, peserta magang memiliki akses ke sejumlah alat yang dapat mempermudah pekerjaannya atau bahkan dapat diintegrasikan ke dalam pengembangan *website*. Salah satu alat yang menjadi pijakan utama bagi pengembang *website* adalah software Visual Studio Code. Visual Studio Code bukan hanya sekadar *text editor* biasa. Visual Studio Code merupakan *platform* yang komprehensif untuk pengembangan *website* dengan segudang fitur yang mendukung produktivitas pengguna. Dalam hal ini, peserta magang dapat memanfaatkan Visual Studio Code untuk menulis kode dalam berbagai bahasa pemrograman, mulai dari HTML, CSS, Javascript Python, Vue.js, dan masih banyak lagi. Kemampuan ini membuat Visual Studio Code menjadi alat yang serbaguna dan dapat digunakan untuk berbagai jenis proyek pengembangan *website* selama program magang berlangsung. Antarmuka pengguna yang intuitif serta fitur pengeditan kode yang kuat menjadikan Visual Studio Code pilihan yang tepat bagi peserta magang dalam melaksanakan pekerjaannya. Selain itu, dukungan untuk ekstensi memungkinkan pengguna untuk menyesuaikan lingkungan pengembangan mereka sesuai dengan kebutuhan spesifik proyek. Dengan berbagai ekstensi yang tersedia, peserta magang dapat memperluas fungsionalitas Visual Studio Code untuk memenuhi kebutuhan khusus mereka, mulai dari pengembangan *front-end* hingga pengelolaan *database* atau *back-end*.

Dalam hal ini, peserta magang diminta oleh *supervisor* untuk *men-download software* Visual Studio Code melalui *website*-nya, yaitu [code.visualstudio.com](https://code.visualstudio.com). Setelah itu, beliau meminta untuk *meng-install software* tersebut. Setelah selesai *install*, beliau meminta untuk peserta magang melakukan instalasi *extension* yang

diperlukan untuk proyek pengerjaan pengembangan *website*. *Extention* tersebut, antara lain CodeSandBox, EsLint, Jupyter serta Python, Locofy.ai, Prettier – Code formater, dan Vue – Official. Berikut ini merupakan penjelasan *extension* tersebut mengenai apa fungsinya dan dipakai untuk apa saja *extension* tersebut.

1. CodeSandBox merupakan sebuah *extension* yang berfungsi untuk mengubah sebuah desain, terutama desain dalam Figma menjadi kodingan HTML, CSS, JavaScript, TypeScript, Vue.Js, React.Js, dan sebagainya. Hal ini sangat bermanfaat bagi *web developer* dan juga divisi UI/UX. Hal ini dikarenakan *web developer* dapat dengan mudah nantinya mengintegrasikan *code* yang di-*generate* dari CodeSandBox berdasarkan desain yang telah dibuat oleh divisi UI/UX. Namun, terdapat sejumlah kekurangan, yaitu kodingan yang di-*generate* melalui CodeSandBox tidaklah *responsive* apabila langsung diintegrasikan ke dalam *website*. Oleh karena itu, perlu disesuaikan lagi kodingannya supaya tampilannya *responsive*.
2. EsLint merupakan sebuah *extension* yang digunakan untuk melakukan *debugging* terhadap kodingan JavaScript. *Extension* ini akan sangat membantu peserta magang karena sifat dari *extenssion* ini adalah memberikan laporan terhadap *error* yang terjadi dalam kodingan JavaScript sehingga peserta magang dapat dengan mudah memperbaikinya.
3. Jupyter dan Python merupakan sebuah *extension* yang diperlukan oleh Visual Studio Code untuk bahasa pemrograman Python. Dalam hal ini, kedua *extension* tersebut berfungsi untuk menjalankan pemrograman berbahasa Python. Python dalam magang ini diperlukan untuk beberapa hal, seperti *data scrapping*, menghilangkan

data *null*, *formatting* data, menyeragamkan data, dan sebagainya.

4. Locofy.ai merupakan sebuah *extension* yang hampir sama dengan CodeSandBox, yaitu mengubah desain di Figma menjadi kodingan HTML, CSS, TypeScript, JavaScript, Vue.js, React.js, dan sebagainya. Hanya saja Locofy.ai memiliki sebuah AI yang berfungsi untuk *generate code* dengan lebih akurat. Namun, Locofy.ai jarang digunakan karena sistemnya hanya *free trial*. Perusahaan dalam hal ini lebih menekankan untuk menggunakan CodeSandBox karena meskipun berbayar, tetapi tidak *trial*. CodeSandBox dalam hal ini bisa memproduksi empat desain maksimal untuk *free* dan untuk menambahkannya, dapat menghapus salah satu *generate code* yang sudah tidak terpakai. Oleh karena itu, Locofy.ai jarang digunakan dan perusahaan lebih menekankan untuk menggunakan CodeSandBox.
5. Prettier – Code formater merupakan sebuah *extension* yang digunakan untuk melakukan *formatting* kodingan di Visual Studio Code. Hal ini bertujuan untuk menunjukkan kodingan yang lebih rapi sehingga tidak membingungkan baik peserta magang, maupun karyawan lain yang melihat kodingannya nanti.
6. Vue – Official merupakan sebuah *extension* yang digunakan menjalankan dan membuat *website* dengan menggunakan Vue.js. *Extension* ini digunakan apabila terdapat suatu hal yang memerlukan Vue.js.

Selain, Visual Studio Code dan juga *extension* yang diminta untuk di-*install*, terdapat sejumlah *extension* lain yang diminta untuk di-*install* juga di *browser*. Dalam hal ini peserta magang menggunakan *browser* Google Chrome sehingga *download* dan

*install* harus melalui Chrome Web Store. Beberapa hal yang harus di-*install*, antara lain Allow Copy +, Touch VPN, Vue.js devtools, dan Enable Right Click for Google Chrome. Berikut ini merupakan penjelasan *extension* tersebut mengenai apa fungsinya dan dipakai untuk apa saja *extension* tersebut.

1. Allow Copy + merupakan *extension browser* yang berfungsi untuk melakukan *copy* terhadap konten di salah satu *website*. Hal ini dibutuhkan karena salah satu *partner* bisnis perusahaan memiliki *website* yang *men-disable* atau melarang *user* untuk meng-*copy* konten yang ada dalam *website*-nya sedangkan perusahaan memerlukannya untuk keperluan *testing* data dan sudah minta izin. Meskipun sudah meminta izin dan disetujui, *partner* tersebut masih belum mengubah regulasi *website*-nya sehingga *supervisor* meminta peserta magang untuk men-*download* dan *install* Allow Copy + dengan tujuan tersebut.
2. Touch VPN merupakan sebuah *virtual private network* (VPN) yang berfungsi untuk meng-*unblock website* yang tidak bisa diakses entah disebabkan oleh apapun. Oleh karena itu, *supervisor* meminta untuk men-*download* dan *install* sebuah VPN yang nantinya akan diperlukan apabila terdapat sebuah *website* yang tidak bisa diakses karena diblokir, rusak, dan sebagainya.
3. Vue.js devtools merupakan sebuah *extension* yang berfungsi untuk melakukan *debugging* terhadap *website* yang dibuat menggunakan Vue.js. Hal ini akan sangat bermanfaat karena apabila terdapat *error*, Vue.js devtools dapat melakukan laporan mengenai *error*-nya di mana, *error*-nya apa, solusinya dan sebagainya.

4. Enable Right Click for Google Chrome merupakan pasangan *extension* untuk Allow Copy +. *Extension* ini berfungsi untuk mengaktifkan fitur *right click* untuk *website* yang tidak mengizinkan *right click* di *website*-nya. Dalam hal ini, kasusnya sama seperti sebelumnya bahwa terdapat *website partner* perusahaan yang tidak mengizinkan *user* untuk melakukan *right click* untuk meng-*copy* konten dari *website* tersebut. Tentu, dalam hal ini perusahaan telah meminta izin untuk mengizinkan *user* melakukan *right click* dan *copy* salah satu konten di dalam *website* tersebut guna melakukan *testing* data. Namun, karena regulasinya masih sama dan tidak di-*enable* fitur *right click* di *website* tersebut, *supervisor* meminta peserta magang untuk men-*download* dan *install extension* tersebut untuk bisa meng-*copy* konten dari *website partner* perusahaan ini.

Hal lain yang dilakukan pada tahapan *setup* dan mempelajari *tools* adalah melakukan *login* menggunakan *email* perusahaan ke beberapa *tools* lain, seperti *database* Phpmyadmin dan Namecheap. Phpmyadmin merupakan sebuah *tools* yang berisikan *database* perusahaan. Isi *database*-nya merupakan data-data perusahaan yang sangat krusial sehingga peserta magang dilarang keras melakukan perubahan, penambahan, dan menghapus tanpa seizin *supervisor*. Phpmyadmin sendiri merupakan *back-end* dari iiga.shop, salah satu *website e-commerce* perusahaan. Namecheap merupakan sebuah *website* yang menyediakan layanan *hosting website* perusahaan. Namecheap berisikan *hosting-an website* perusahaan. Isi dari *website* ini sangatlah berbahaya karena apabila terjadi kesalahan atau tidak sengaja mengklik sesuatu, dapat menyebabkan salah satu *website* perusahaan menjadi *down*. Oleh karena itu, *supervisor* peserta magang lagi-lagi memperingati untuk tidak melakukan perubahan,

penambahan, dan menghapus data tanpa seizin *supervisor* dan lebih baik untuk tidak membuka *tab* tersebut apabila tidak dibutuhkan.

### **3.2.2.2 Mempelajari Relevansi Projek terhadap Bisnis Perusahaan**

Pada minggu ketiga dan keempat bulan Januari, para peserta magang yang berada di wilayah Jakarta, Bekasi, dan Yogyakarta diundang untuk bertemu secara langsung dengan *supervisor*. Pertemuan pertama diadakan di Bekasi dan yang kedua di Jakarta. Tujuan dari pertemuan ini adalah untuk memperkenalkan para peserta magang dengan karyawan lain yang berasal dari berbagai daerah di Indonesia, memungkinkan mereka untuk saling mengenal lebih baik, lebih personal dan interaktif. Pertemuan ini juga memberikan kesempatan untuk memperdalam pemahaman mereka tentang struktur dan operasional perusahaan.

Dalam pertemuan tersebut, *supervisor* yang juga menjabat sebagai *marketing director*, memberikan penjelasan mendalam tentang pentingnya projek revitalisasi *website* yang sedang berlangsung. Beliau menekankan bahwa perusahaan kini berada di ambang perubahan besar, sebuah transformasi yang disebut dengan *business re-engineering*. Perubahan ini dimotivasi oleh kebutuhan untuk menyesuaikan diri dengan era digital yang kian mendominasi setiap aspek bisnis.

Dalam melaksanakan perubahan tersebut, tentu akan ada banyak hal yang harus dipersiapkan. Salah satu hal yang harus dipersiapkan adalah adanya media *website* yang dapat digunakan untuk promosi perusahaan dan produk-produknya. Pada kenyataannya, meski perusahaan telah memiliki *website* sejak lama, situs tersebut terkesan terlantar dan kurang terawat karena minimnya pembaruan dan pengelolaan yang efektif. Dalam dunia yang serba

cepat dan berorientasi digital saat ini, keberadaan *website* yang dinamis dan menarik menjadi krusial untuk menarik perhatian dan retensi pelanggan. Oleh karena itu, fokus utama dari proyek revitalisasi *website* adalah untuk menghidupkan kembali situs tersebut dengan mengintegrasikan teknologi terkini, menyertakan konten yang relevan dan menarik, serta menampilkan produk-produk unggulan perusahaan dengan cara yang lebih inovatif dan menarik. Hal ini bertujuan agar banyak orang yang lebih berminat untuk mengunjungi *website* perusahaan.

Selain itu, perusahaan juga merencanakan untuk meningkatkan upaya pemasaran digitalnya yang tidak hanya terbatas pada *website*, tetapi juga meliputi kampanye periklanan di berbagai *platform* media sosial, seperti Facebook, Youtube, Instagram, dan Twitter. Dengan memanfaatkan berbagai kanal digital, perusahaan bertujuan untuk menciptakan sebuah ekosistem *online* yang tidak hanya meningkatkan visibilitas produk-produknya, tetapi juga memfasilitasi interaksi yang lebih efektif dan menarik dengan konsumen. Hal ini merupakan langkah strategis dalam upaya perusahaan untuk tetap relevan dan kompetitif di pasar yang dinamis.

Dengan implementasi strategi-strategi tersebut, perusahaan berharap dapat memulihkan diri dari kerugian yang signifikan di mana sebelumnya sempat mencatat penurunan pendapatan hingga hampir 70%. Melalui revitalisasi *website* dan peningkatan fokus pada pemasaran digital, ada potensi besar untuk tidak hanya mengembalikan keuntungan, tetapi juga untuk meningkatkan pengakuan dan reputasi perusahaan di pasar yang lebih luas. Langkah ini diharapkan dapat membuka jalan bagi perusahaan untuk mengakses pasar baru dan menarik lebih banyak klien, sekaligus memperkuat posisinya sebagai pemimpin dalam industri. Dengan cara ini, perusahaan dapat memastikan keberlanjutan dan pertumbuhan

yang berkelanjutan di masa depan, mengubah tantangan menjadi peluang melalui inovasi, dan adaptasi yang strategis.

### **3.2.2.3 Menentukan Bagaimana Cara Menyelesaikan Masalah Perusahaan yang Berhubungan dengan *Website*.**

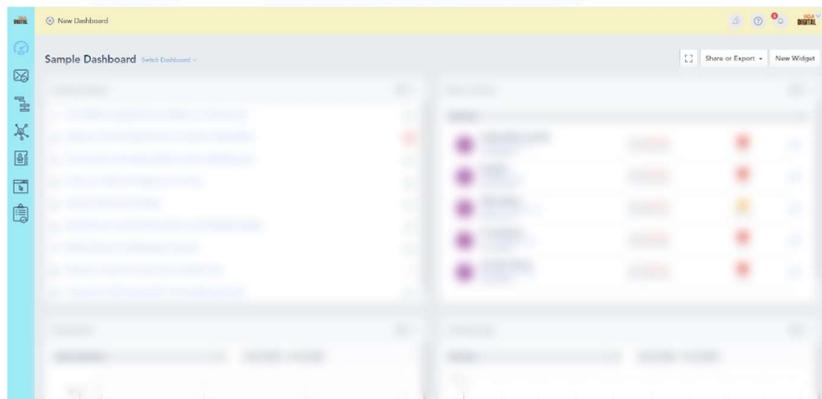
Seperti yang sudah dijelaskan di bagian-bagian sebelumnya bahwa perusahaan memiliki sebuah *website e-commerce* yang bernama *iiga.shop*. *iiga.shop* sudah ada sejak lama dan sudah memiliki banyak sekali perusahaan yang menjual produknya di *website* tersebut. Produk-produk yang dijual dapat berupa produk barang hingga jasa di antaranya adalah barang kecil (pipa ledeng, meteran, obeng, dan sebagainya) hingga barang besar (kapal, mesin pabrik, mobil *pickup*, dan sebagainya), dan berbagai jasa, seperti jasa renovasi rumah, ekspor barang menggunakan *container*, sewa kendaraan, dan sebagainya.

Perusahaan dalam hal ini ingin memikirkan bagaimana cara untuk menggaer lebih banyak perusahaan yang bermitra ke *iiga.shop* supaya perusahaan dapat bekerja sama dengan berbagai *partner* bisnis lainnya dari berbagai negara. Dalam membahas masalah ini, divisi *web developer* tidak bekerja sendirian, tetapi dibantu oleh divisi *sales / marketing* yang memang lebih ahli dalam bidang tersebut.

Terdapat berbagai pendapat mengenai solusi pada masalah tersebut, seperti mempromosikan *iiga.shop* melalui iklan di media sosial, menggaet kembali *partner-partner* bisnis perusahaan ini yang sebelumnya pernah bekerja sama dengan perusahaan, dan masih banyak lagi. Namun, terdapat satu solusi yang terbaik menurut *supervisor* adalah dengan mencari informasi sebuah perusahaan (terutama *email* perusahaan tersebut) yang kemudian *email* perusahaan tersebut akan dikumpulkan. Setelah dikumpulkan, *email-email* perusahaan tersebut akan dipakai untuk *email marketing automation*

yang berisikan ajakan perusahaan-perusahaan tersebut untuk menjual produk barang atau jasanya di iiga.shop. Hal tersebut disetujui oleh *supervisor* karena beberapa hal. Pertama, melakukan promosi melalui *email* dinilai lebih efektif dari pada menggunakan media sosial. Hal ini dikarenakan perusahaan dapat mempromosikan langsung ke perusahaan tersebut melalui *email*. Hal ini akan mengurangi biaya yang harus dibayarkan perusahaan apabila ingin melakukan promosi melalui media sosial. Kedua, perusahaan dapat menentukan sendiri siapa saja perusahaan yang menjadi target *market*-nya sehingga perusahaan tidak perlu membuang biaya yang lebih karena mempromosikan *website e-commerce*-nya ke perusahaan yang bukan target *market*-nya. Perusahaan ini lebih berfokus pada perusahaan-perusahaan industri teknologi, seperti pabrik mesin, robotik, otomotif, dan sebagainya sehingga perusahaan-perusahaan selain industri teknologi bukanlah target *market* perusahaan sehingga perusahaan tidak perlu melakukan promosi ke perusahaan-perusahaan tersebut. Hal ini akan mempersingkat waktu dan memotong biaya promosi. Ketiga, perusahaan memiliki sebuah *tools* untuk melakukan *email marketing automation* sehingga perusahaan tidak perlu membuang banyak waktu untuk mengirimkan *email* satu per satu ke setiap perusahaan. Hal tersebut tentunya dinilai tidak efektif karena membuang banyak waktu dan akan melelahkan divisi *sales / marketing* yang bertugas melakukan hal tersebut. Oleh karena itu, perusahaan *email marketing automation tools* akan digunakan dan diperlukan dalam hal ini untuk membantu mempromosikan iiga.shop. Caranya adalah dengan divisi *sales / marketing* akan menghadiri berbagai pameran pada tahun 2024 ini dan tugas mereka adalah meminta kartu nama setiap perusahaan yang hadir dalam *event* pameran tersebut. Mereka bertugas untuk mendapatkan kartu nama perusahaan sebanyak-banyaknya dari negara manapun kecuali China.

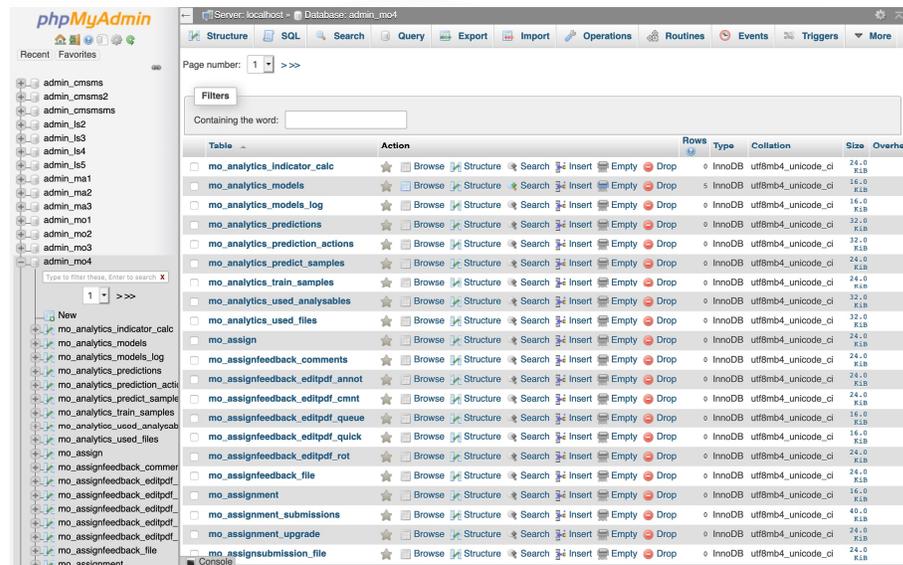
Perusahaan-perusahaan China dalam hal ini bukan merupakan target negara perusahaan ini dikarenakan terdapat sebuah regulasi di negara China yang menyebabkan mereka wajib berjualan di *e-commerce* negaranya sendiri. Oleh karena itu, tidak ada gunanya perusahaan mempromosikan *iiga.shop* yang merupakan *e-commerce* milik perusahaan Singapura ke perusahaan China karena mereka tidak mungkin akan bermitra ke *iiga.shop*. Kemudian, setelah mendapatkan kartu nama perusahaan-perusahaan, divisi *sales / marketing* bertugas meng-*input email*, nama, dan nomor telepon perusahaan ke dalam Google Sheets yang nantinya akan diberikan ke divisi IT (*web developer*) yang bertugas untuk meng-*upload* data-data perusahaan tersebut ke *tools email marketing automation* yang dapat melakukan *blast email* ke perusahaan-perusahaan tersebut. Nama *tools* tersebut adalah VBout.



Gambar 3.5 *Dashboard* VBout

Gambar 3.5 merupakan *dashboard* VBout yang merupakan tampilan awal VBout setelah *user* melakukan *login*. VBout merupakan sebuah *tools* yang digunakan oleh perusahaan untuk melakukan *email marketing automation*. VBout sangat berguna untuk perusahaan melakukan *blasting email* untuk melakukan promosi produknya.

### 3.3.3 Mempelajari PHP.



Gambar 3.6 Dashboard PHP [7]

*Hypertext Preprocessor* atau yang kerap disebut dengan PHP merupakan sebuah *database* yang dapat digunakan untuk pengembangan aplikasi *mobile*, *website*, dan masih banyak lagi. Gambar 3.6 merupakan gambar yang menunjukkan *dashboard* dari PhpMyAdmin (perlu diketahui bahwa gambar tersebut bukanlah *database* milik perusahaan sehingga aman apabila tidak disensor), yaitu sebuah tampilan antarmuka berbasis *web* yang digunakan untuk mengelola *database* dengan bahasa pemrograman MySQL. PhpMyAdmin memungkinkan pengguna untuk melakukan berbagai tugas administrative, seperti membuat, mengedit, menghapus, mengelola tabel kueri *database*, dan sebagainya. PhpMyAdmin menjadi salah satu *tools* yang sangat populer di kalangan pengembang *web* untuk mengelola *database* dengan bahasa pemrograman MySQL secara efisien dan mudah.

Dalam konteks *website* perusahaan, *iiga.shop*, *back-end* dari *iiga.shop* menggunakan PhpMyAdmin. Terdapat sejumlah alasan mengapa

perusahaan memilih untuk menggunakan PhpMyAdmin sebagai *back-end* dari iiga.shop. Alasan tersebut adalah sebagai berikut.

1. PhpMyAdmin merupakan sebuah *back-end* yang sudah ada sejak lama sehingga banyak yang memahami penggunaan dari PhpMyAdmin. Dengan populernya PhpMyAdmin menyebabkan banyak sekali *tutorial* di internet yang dapat digunakan oleh petugas IT perusahaan sehingga apabila terdapat sebuah masalah, petugas IT dapat mencarinya di internet dengan cepat dan mencari solusinya dengan cepat juga sehingga *website* tersebut tidak *down*.
2. PhpMyAdmin merupakan *tools* yang gratis dan *open-source*. Hal tersebut sangat bermanfaat bagi perusahaan karena perusahaan dapat memotong biaya untuk pengembangan *website*.
3. Alasan berikutnya adalah PhpMyAdmin dapat menampung banyak sekali data sehingga sangat bermanfaat bagi perusahaan untuk menggunakannya dalam jangka panjang. Sudah sejak *website* ini berdiri sampai sekarang, perusahaan masih menggunakan *database* PhpMyAdmin. Hal tersebut membuktikan bahwa *database* tersebut mampu menampung banyak sekali data mengingat iiga.shop sudah berdiri sekitar sepuluh tahun lalu dan *database*-nya masih tetap sama. Tentu, selama sepuluh tahun, data yang dimasukkan ke dalam *website* iiga.shop terus-menerus bertambah dan berubah. Namun, PhpMyAdmin tetap stabil dan tidak ada masalah selama lebih dari sepuluh tahun ia digunakan sebagai *back-end* dari *website* ini. Hal tersebut menjadikan PhpMyAdmin pilihan yang terbaik bagi perusahaan.

Setelah menjelaskan mengenai PhpMyAdmin, *supervisor* memberikan *login email* dan *password* kepada peserta magang agar mereka dapat mengakses *database* PhpMyAdmin milik perusahaan. Namun, *supervisor* juga memberikan peringatan penting untuk tidak sembarangan mengklik dalam PhpMyAdmin karena sifatnya yang sensitif. Pengalaman *supervisor* sendiri menjadi pelajaran berharga, di mana sebuah klik yang

tidak disengaja pada waktu yang lalu menyebabkan *website down* dan banyak data yang hilang. Kejadian tersebut memaksa mereka untuk melakukan pemulihan data secara manual dan memberikan permintaan maaf kepada perusahaan yang terdampak. Selain itu, terdapat peringatan penting lainnya, yaitu peserta magang diperbolehkan untuk mencoba, tetapi dilarang melakukan *publish*. Oleh karena itu, peserta wajib melakukan *testing* atau coba-coba di halaman *website* yang *private* yang tidak bisa diakses oleh *user* luar selain karyawan internal perusahaan. Apabila seandainya terdapat *maintance* dalam *website* iiga.shop yang disebabkan oleh berbagai hal, divisi *web developer* wajib memperbaikinya secepatnya. Pada intinya, selama jam kerja, *website* iiga.shop wajib bekerja dengan normal dan apabila terdapat perubahan yang harus dilakukan, entah perubahan data, penambahan data, dan sebagainya sebaiknya dilakukan di luar jam kerja, seperti di atas pukul 17.00 WIB atau pada hari libur.

Berdasarkan pengalaman dan peringatan tersebut, peserta magang diberikan pembelajaran tentang penggunaan PhpMyAdmin secara hati-hati dan cermat. Mereka diajarkan bagaimana cara menambah (*create*), memperbarui (*update*), menghapus (*delete*), mengunduh data, serta melakukan pencarian data dan mengetahui lokasi-lokasi dari data tersebut (*searching*). Pembelajaran ini menjadi penting karena akan membantu dalam merevitalisasi tampilan iiga.shop. Peserta juga diajarkan cara memanggil data dan memasukkan data menggunakan metode GET dan POST. GET digunakan untuk menampilkan data di *front-end*, sementara POST digunakan untuk memasukan dan menyimpan data ke dalam *database*. Dengan pemahaman yang matang tentang penggunaan PhpMyAdmin dan pemrosesan data melalui GET dan POST, peserta magang diharapkan dapat berkontribusi secara efektif dalam pengembangan dan perbaikan tampilan iiga.shop. Lebih dari sekadar mengakses database, mereka diajarkan untuk bertanggung jawab atas setiap tindakan yang mereka lakukan dalam lingkungan sensitif seperti PhpMyAdmin.

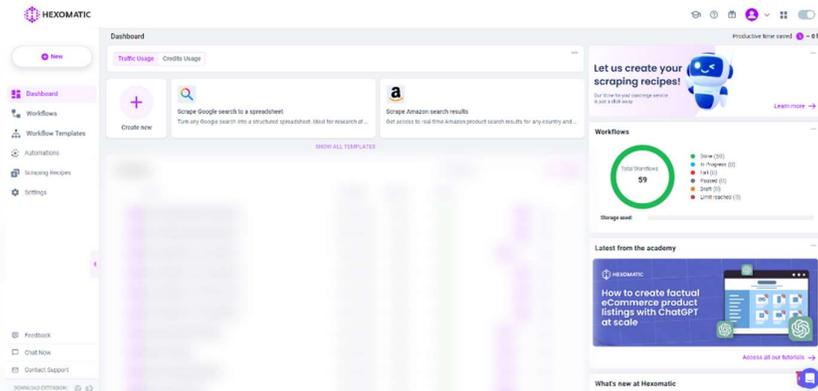
### 3.3.4 Data Scrapping

*Data scrapping* merupakan sebuah tindakan untuk mengambil atau men-*copy* dan *paste* data dari sebuah *website* atau sumber digital lainnya ke dalam *format file* .xlsx atau .csv. *Data scrapping* dapat dilakukan dengan cara menggunakan *tools* atau dengan menggunakan kodingan khusus. Dalam konteks perusahaan ini, data yang diambil dari sumber-sumber tertentu digunakan untuk keperluan divisi *sales / marketing* dan juga dibutuhkan untuk dimasukkan ke dalam *back-end website* nantinya. Sebenarnya tindakan ini dapat dikatakan sebagai tindakan yang ilegal karena mengambil data dari sumber lain untuk dijadikan data pribadi, tetapi lagi-lagi perusahaan telah meminta izin kepada perusahaan yang memiliki *website* tersebut untuk diambil datanya untuk keperluan perusahaan ini.

Pertama-tama, *supervisor* memerintahkan peserta magang untuk melakukan *data scrapping*. Beliau menjelaskan bahwa *data scrapping* merupakan sebuah proses untuk mengambil data dari *website* yang beliau akan berikan. Beliau juga memberikan sebuah *tools*, yaitu Hexomatic yang berfungsi untuk melakukan *data scrapping*.

#### 3.3.4.1 Hexomatic

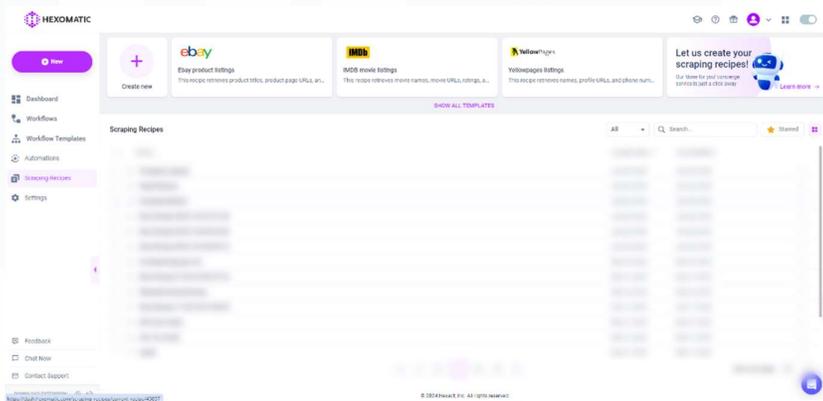
Hexomatic merupakan sebuah *tools* berupa *website* yang berfungsi untuk melakukan *data scrapping*. Perusahaan sudah menggunakan *tools* tersebut sudah cukup lama sehingga perusahaan familiar dengan penggunaannya. Selain itu, perusahaan juga memiliki akun *premium* yang paling tinggi sehingga semua fitur bisa diakses yang dapat memudahkan peserta magang untuk melakukan *data scrapping*. Berikut ini merupakan tampilan *dashboard* dari Hexomatic.



Gambar 3.7 *Dashboard* Hexomatic

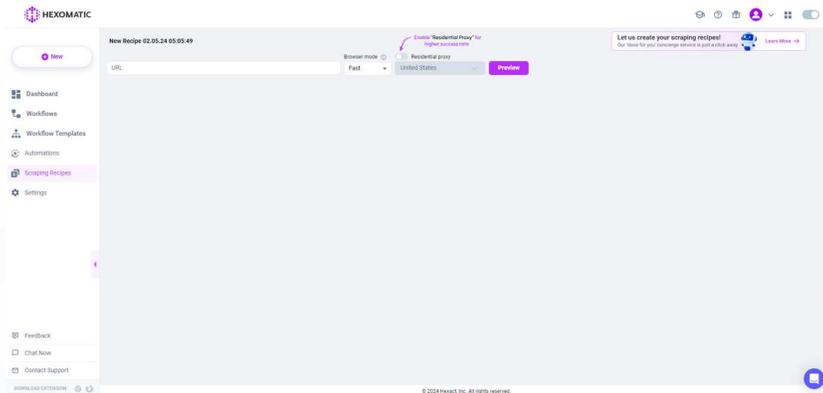
Gambar 3.7 merupakan tampilan *dashboard* dari Hexomatic. Bagian yang disensor merupakan bagian projek *data scrapping* yang sudah pernah dilakukan. Hal tersebut dilarang oleh perusahaan untuk ditunjukkan sehingga perlu disensor. Dalam hal ini, *supervisor* meminta peserta magang untuk mencoba-coba terlebih dahulu dan juga melihat *tutorial* yang disediakan oleh Hexomatic untuk melakukan *data scrapping*. Hal ini dikarenakan meskipun *supervisor* peserta magang sudah pernah menggunakan *tools* ini sebelumnya dan *tools* ini juga sudah ada sejak lama, tetapi *tools* ini jarang digunakan sehingga *supervisor* mengatakan bahwa *tools* ini bisa melakukan *data scrapping*, tetapi caranya bagaimana beliau juga sudah lupa karena sudah lama tidak digunakan dan juga sudah banyak *update* yang menyebabkan tampilan dari *website* Hexomatic sendiri sudah berubah drastis sejak terakhir kali *supervisor* menggunakannya. Hal tersebut merupakan sebuah hambatan yang dialami oleh peserta magang karena peserta magang harus mempelajarinya secara mandiri dan mencoba-cobanya sendiri. Belum lagi, *tutorial* yang disediakan oleh Hexomatic juga tidaklah lengkap atau dengan kata lain tidak sesuai dengan tugas yang diberikan kepada peserta magang dan juga *tutorial* yang diberikan oleh Hexomatic tidak *update* sehingga tampilan *website* Hexomatic masih menggunakan tampilan yang lama sehingga

membingungkan peserta magang. *Tutorial* yang diberikan oleh Hexomatic pun terakhir di-*upload* sekitar dua tahun lalu sehingga menyulitkan peserta magang untuk menyesuaikan dengan tampilan Hexomatic yang terbaru. Namun, dengan tekad untuk mempelajari lebih dalam dan mencoba-coba sendiri, peserta magang dapat mencari tahu bagaimana melakukan *data scrapping* dengan menggunakan Hexomatic. Berikut ini merupakan langkah penggunaannya.



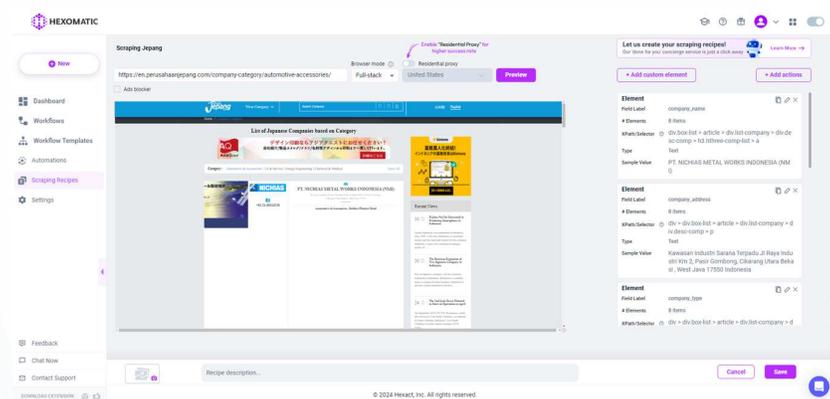
Gambar 3.8 Tampilan Bagian *Scrapping Recipe* dari Hexomatic

Pertama-tama, kita pergi ke menu *Scrapping Recipe* seperti di gambar 3.8. Halaman *Scrapping Recipe* merupakan halaman yang berfungsi untuk menentukan bagian mana saja dari *website* yang mau di-*copy*. Langkah selanjutnya adalah kita klik *Create new* dan kemudian kita akan masuk ke halaman sebagai berikut.



Gambar 3.9 Halaman *Create new* dalam menu *Scraping Reciepe*

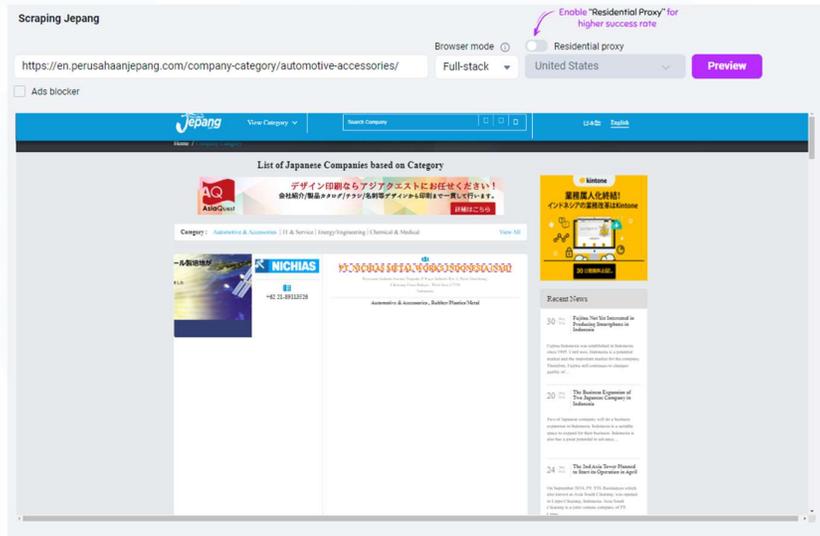
Gambar 3.9 merupakan halaman ketika *user* klik *Create new* di menu *Scraping Reciepe*. Bergasarkan gambar 3.8, dapat dilihat bahwa halaman tersebut terlihat kosong. Hal pertama yang harus dilakukan *user* adalah meng-*copy* dan *paste link* URL *website* yang akan di *scrape* datanya di *section* yang bertuliskan URL. Setelah itu, klik *Preview* dan hasilnya adalah sebagai berikut.



Gambar 3.10 Tampilan *Preview* dari *Website*

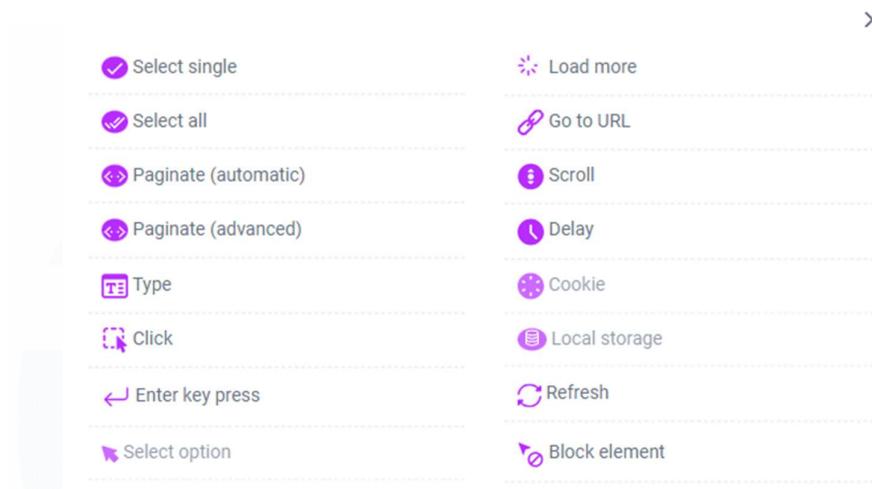
Gambar 3.10 merupakan tampilan apabila *user* sudah melakukan *paste* URL dan mengklik tombol *Preview*. Tampilan dari *website* yang akan di-*scrape* akan muncul seperti gambar 3.9. Jangan khawatir apabila tampilan *website* tidak sesuai dengan yang aslinya

karena memang bisa saja berbeda, tetapi yang paling penting adalah bagian data-data yang akan di-*scrape* terlihat sehingga dapat diseleksi. Langkah berikutnya adalah klik salah satu *elements* yang akan dilakukan *scrapping*. Dalam hal ini, diambil contoh salah satunya adalah nama perusahaan berdasarkan *website* ini.



Gambar 3.11 Contoh Pengambilan Data dari *Website*

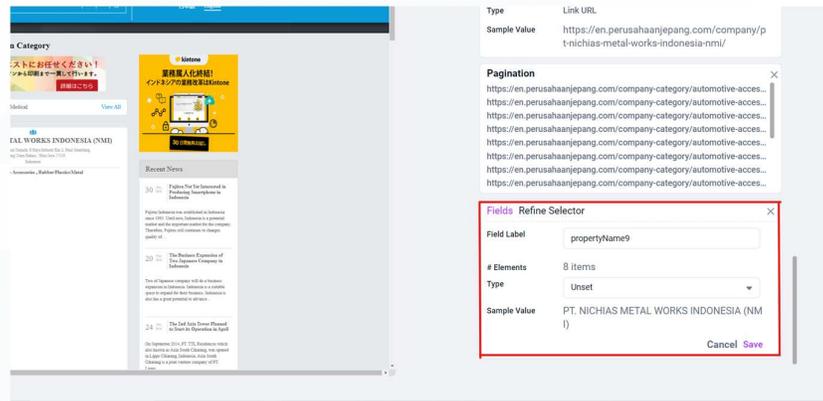
Gambar 3.11 merupakan contoh bagaimana cara mengambil salah satu data dari *website* tersebut. *User* hanya tinggal melakukan klik terhadap salah satu *element website* tersebut. Dalam contoh di gambar 3.11, *user* mengklik nama perusahaan. Kemudian, setelah *user* klik salah satu *element* tersebut, akan muncul *popup* sebagai berikut.



Gambar 3.12 *Popup* Setelah *User* Klik Salah Satu *Element Website*

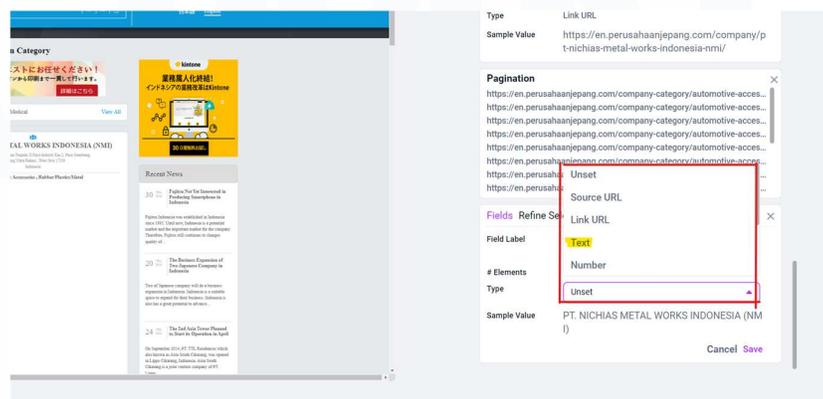
Gambar 3.12 menunjukkan apa yang terjadi setelah *user* mengklik salah satu *element* dari *website* tersebut yang dalam hal ini adalah nama perusahaan sebagai contoh. Berdasarkan gambar tersebut, dapat dilihat bahwa terdapat banyak pilihan. Pilihan-pilihan tersebut merupakan pilihan yang berfungsi untuk memberi tahu Hexomatic apa yang harus dilakukan oleh Hexomatic ketika *user* mengklik salah satu *element* dalam *website* tersebut. Sebagai contoh, pilihan *Select all* merupakan pilihan yang digunakan untuk menyeleksi *element* yang sama dalam satu halaman *website*. Dalam hal ini, fungsi tersebut yang digunakan karena *website* yang *supervisor* berikan berbentuk *list* nama-nama perusahaan sehingga dalam satu halaman terdapat banyak nama-nama perusahaan (tidak hanya satu seperti gambar sebelumnya dikarenakan tampilan *website* asli dan *Preview* Hexomatic berbeda). Oleh karena itu, untuk mendapatkan semua nama-nama perusahaan dalam satu halaman *website* tersebut, dipilahlah fungsi *Select all*. Oleh karena itu, secara otomatis Hexomatic akan mendeteksi *element* dan *div class* yang sama

dengan nama perusahaan yang pertama kali diklik oleh *user*. Setelah itu, akan muncul seperti gambar berikut.



Gambar 3.13 Tampilan Setelah Klik *Select All*

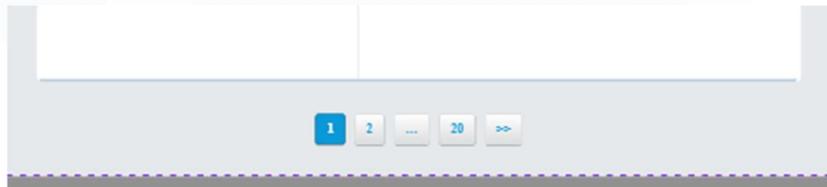
Gambar 3.13 merupakan tampilan setelah *user* klik *Select all*. Berdasarkan gambar tersebut, muncul sebuah *box* yang ditandai berwarna merah. *Box* tersebut berisikan sebuah *settings* yang dapat membuat Hexomatic mengetahui apa yang perlu di-*copy* dari *element* tersebut. Dalam hal ini, *Field Label* diganti dengan nama yang bebas sesuai dengan kemauan *user*. *Field Label* dalam hal ini merupakan nama kolom yang nantinya akan ditampilkan di file *.xlsx* atau *.csv*. Kemudian, hal lain yang harus diganti adalah *Elements Type*.



Gambar 3.14 Pilihan *Elements Type*

Gambar 3.14 merupakan pilihan dari *Elements Type*. *Elements Type* merupakan pilihan jenis data dari pilihan yang *user* seleksi. Berdasarkan gambar tersebut, dapat dilihat bahwa terdapat banyak pilihannya. Berdasarkan hal ini, *Text* adalah *element type* yang dipilih sehingga Hexomatic mengetahui bahwa hal yang nantinya akan di-*scrape* dari *element* tersebut adalah *text*. Hal ini akan membuat Hexomatic meng-*copy text* nama-nama perusahaan dalam *website* tersebut.

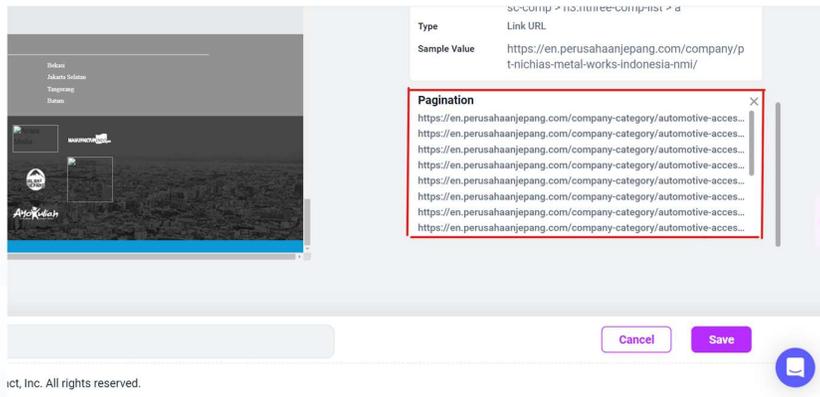
Selanjutnya, terdapat sebuah keunikan dari *website* yang diberikan *supervisor*, yaitu terdapat beberapa halaman yang dibuat menggunakan *pagination*.



Gambar 3.15 *Pagination Website*

Gambar 3.15 menunjukkan bahwa halaman *list* perusahaan tidak berhenti di satu halaman saja, tetapi masih banyak halaman lain. Berdasarkan gambar tersebut, setidaknya ada dua puluh halaman. Cara agar Hexomatic mengetahui bahwa terdapat *pagination* di halaman *website* ini adalah sebagai berikut.

Pada saat *user* mengklik salah satu *button* halaman, Hexomatic akan memunculkan *popup* yang sama seperti pada gambar 3.12. Pilih menu *Paginate (automatic)* supaya Hexomatic tahu bahwa halaman *website* ini memiliki beberapa *page*. Maka, hasilnya adalah sebagai berikut.



Gambar 3.16 *Pagination*

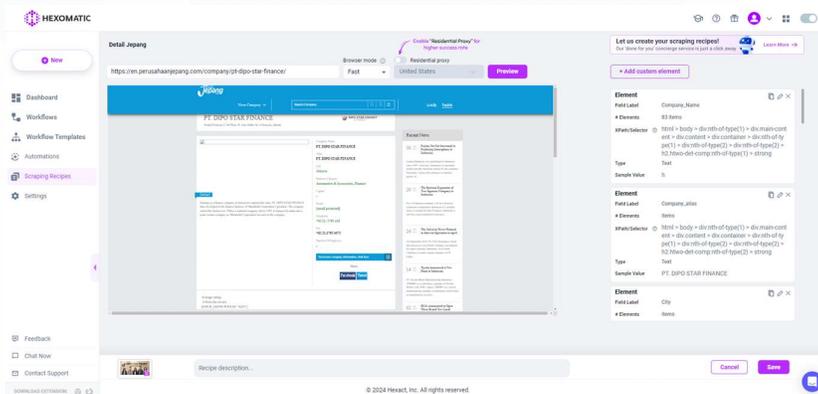
Gambar 3.16 merupakan hasil ketika *user* klik *Paginate (automatic)* di *popup* tersebut. Hexomatic akan secara otomatis meneteksi berapa banyak halaman yang akan dilakukan *scrapping* juga.

Selanjutnya, lakukanlah hal yang serupa untuk data-data lain yang akan di-*scrape*. Pastikan *Element Type* yang dimasukkan sesuai dengan kebutuhan. Namun, terdapat satu hal lagi yang tidak kalah penting sebelum Hexomatic dapat melakukan *scrapping data*. Jadi, bentuk *website* yang diberikan oleh *supervisor* bukan hanya *website list* perusahaan-perusahaan saja, melainkan jika salah satu nama perusahaan diklik, akan memunculkan halaman baru, yaitu halaman informasi yang lebih detail mengenai perusahaan tersebut. Data tersebut juga diperlukan oleh perusahaan. Oleh karena itu, diperlukan sebuah langkah tambahan untuk melakukan hal tersebut. Berikut ini adalah langkah-langkahnya.

Pertama-tama, kita akan klik salah satu *elements* di dalam *website* yang memiliki *link* yang menuju ke halaman informasi perusahaan tersebut. Dalam hal ini, *element* nama perusahaan merupakan *element* yang memiliki *link* untuk ke halaman informasi perusahaan. Selanjutnya, setelah di klik, pilih *Select all* dan di bagian

*Elements Type* yang telah dijelaskan di bagian sebelumnya, pilih *Link URL* Hal ini akan membuat Hexomatic meng-copy link URL dari setiap nama-nama perusahaan tersebut. Hal ini penting karena URL tersebut akan digunakan sebagai penghubung antara halaman *list* perusahaan dan informasi perusahaan nantinya. Kemudian, klik *Save*, *rename*, dan kembali ke halaman utama *Scrapping Reciepes*.

Setelah kembali ke halaman *Scrapping Reciepes*, klik *Create new* lagi untuk membuat *scrapping reciepe* yang baru. Berbeda dengan sebelumnya yang menampilkan halaman *list* perusahaan, kali ini *paste link* URL yang menampilkan halaman informasi perusahaan tersebut. Berikut ini merupakan hasilnya.

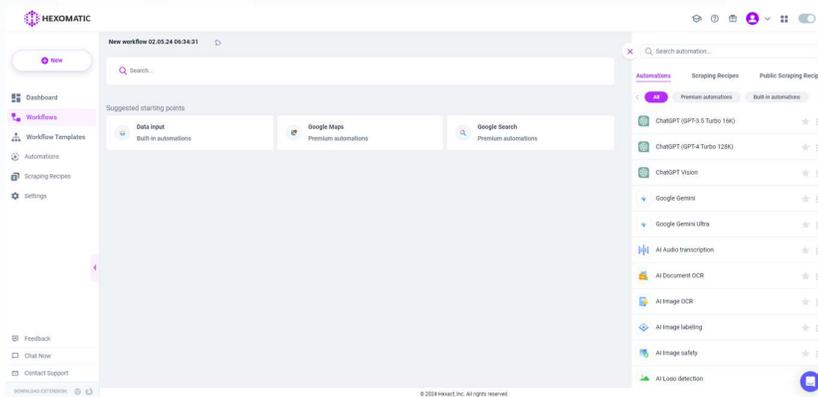


Gambar 3.17 *Scrapping Reciepe* Halaman Informasi Perusahaan

Gambar 3.17 merupakan halaman *Scrapping Reciepe* yang sama seperti sebelumnya. Hal yang membedakan adalah kali ini *Scrapping Reciepe* bukan menampilkan halaman *list* perusahaan, melainkan halaman informasi perusahaan. Sama seperti sebelumnya, klik di *element* yang diperlukan lalu pada menu *popup* seperti gambar 3.12, pilih *Select single*. Hal ini dikarenakan *element* dari halaman ini berbeda dengan halaman *list* perusahaan. Halaman ini memiliki *element* yang sama satu dan yang lainnya sehingga apabila diklik *Select all*, akan mengakibatkan dalam satu kolom dan baris semua

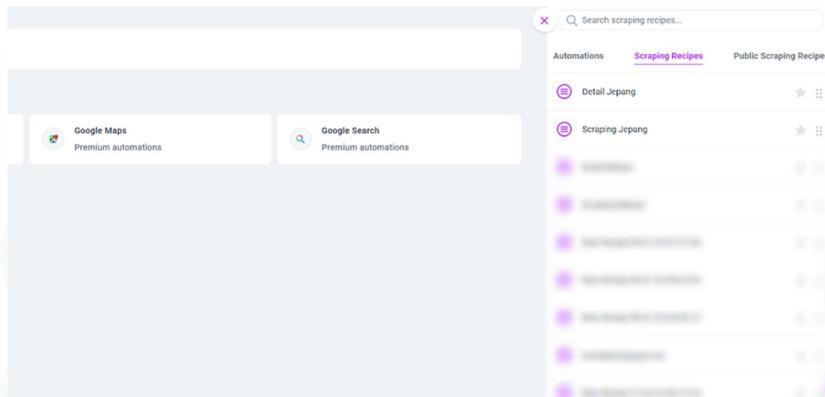
datanya akan tertumpuk. Oleh karena itu, pilih satu-satu *element* dari setiap *website* tersebut dengan memilih *Select single* supaya tidak bertumpuk datanya. Kemudian, jika sudah selesai, klik *Save* dan jangan lupa *rename*.

Selanjutnya, klik menu *Workflow* di *navigation bar* sebelah kiri. Lalu, klik *Create new*. Pada bagian ini, kita akan menggabungkan *scrapping* halaman *list* perusahaan dan halaman informasi perusahaan.



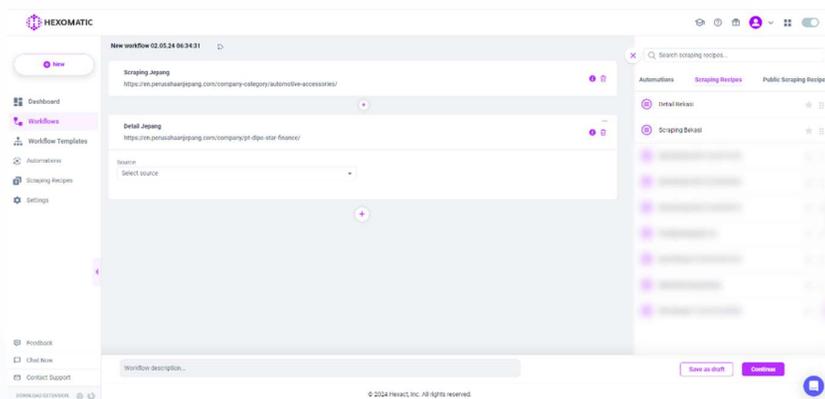
Gambar 3.18 Halaman *Workflow* Baru

Gambar 3.18 merupakan gambar halaman *Workflow* setelah *user* mengklik *Create new*. Halaman ini berfungsi untuk membuat sebuah *workflow* supaya Hexomatic dapat melakukan *scrapping* berdasarkan *workflow* tersebut. Hal pertama yang harus dilakukan adalah di bagian kanan pilih *tab Scrapping Reciepe*. Maka, akan muncul sebagai berikut.



Gambar 3.19 *Scraping Reciepe* dalam *Workflow*

Gambar 3.19 menunjukkan *Scraping Reciepe* yang ada di dalam *Workflow*. Fungsinya adalah untuk dimasukkan ke dalam *Workflow* nantinya. Berdasarkan hal tersebut, pilihlah kedua *Scraping Reciepe* yang baru saja dibuat, satu yang berisi *list* perusahaan dan yang satu yang berisi informasi perusahaan. Kurang lebih hasilnya adalah sebagai berikut.

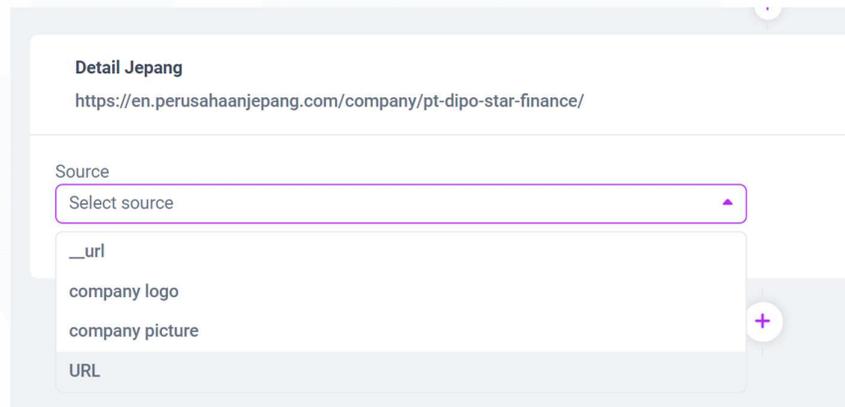


Gambar 3.20 Hasil Memasukkan *Scraping Reciepe*

Gambar 3.20 merupakan contoh hasil memasukkan kedua *scraping reciepe* yang baru saja dibuat. Pastikan halaman *list* perusahaan berada di bagian atas atau yang pertama kali dipilih dan

dilanjutkan oleh *scrapping recipe* yang berisi halaman informasi perusahaan.

Selanjutnya, di bagian *Select source*, pilih URL yang sebelumnya sudah dibuat. URL tersebut diibaratkan sebagai jembatan antara halaman *list* perusahaan dan halaman informasi perusahaan.



Gambar 3.21 Pilih URL

Gambar 3.21 merupakan gambar yang menunjukkan untuk memilih URL sesuai dengan *scrapping list* perusahaan yang sudah ditunjukkan sebelumnya. Alasan harus memilih URL setiap perusahaan adalah untuk hal ini karena URL diibaratkan sebagai penghubung antara halaman *list* perusahaan dan halaman informasi perusahaan. Kalau sudah klik *Continue*. Maka hasilnya nanti akan seperti ini.

id	company_name	company_address	company_type	company_phone	company_tag	company_picture URL	Company Name	Company Alias	City	Business Category	Capital	Email	Telephone	Fax	Number_of_Staff	Company Address	Company	
1	PT NICHIAAS MI	Kawasan Industri Automotiv...	6.41	+62 31-2511532	https://www...	https://www...	PT NICHIAAS MI	PT NICHIAAS MI	Bat	Automotive & B...			+62 31-2511532	+62 31-2511532				Kawasan Industri In...

Gambar 3.22 Hasil *Data Scrapping* Hexomatic

Gambar 3.22 merupakan hasil *data scrapping* dengan menggunakan Hexomatic. Hasilnya dapat dikatakan cukup baik karena akurat dan tidak ada data yang tertinggal.

Dengan demikian, Hexomatic merupakan sebuah *tools data scrapping* yang bagus karena bisa melakukan *data scrapping* dengan baik. Namun, terdapat beberapa kekurangan. Salah satu kekurangannya yang cukup fatal adalah sulit untuk mendeteksi *element-element* JavaScript yang menyebabkan banyak sekali data yang tidak terdeteksi. Kelemahan lain adalah ketika terdapat sebuah halaman informasi perusahaan, Hexomatic memerlukan sebuah *link* URL sebagai penyambung antara halaman informasi perusahaan dan halaman *list* perusahaan sehingga jika terdapat sebuah *website* yang menggunakan JavaScript untuk menampilkan data informasi perusahaan, Hexomatic tidak bisa melakukannya dikarenakan biasanya *website* yang menggunakan JavaScript untuk menampilkan informasi perusahaan tidak memiliki *link* URL yang menghubungkan antara halaman *list* perusahaan dan halaman informasi perusahaan seperti kasus sebelumnya. Oleh karena itu, diperlukan sebuah *tools* atau cara lain untuk melakukan *data scrapping* untuk *website* yang menggunakan JavaScript, yaitu dengan menggunakan pemrograman Python.

### 3.3.4.2 Python

Seperti yang sudah dijelaskan di bagian sebelumnya bahwa Hexomatic memiliki sejumlah kelemahan yang menyebabkan *website* lain yang diminta oleh *supervisor* untuk di-*scrapping* datanya tidak bisa dilaksanakan dengan menggunakan Hexomatic. Oleh karena itu, *supervisor* dan juga peserta magang menemukan solusi untuk menggunakan Python untuk melakukan *data scrapping*. Tentu, dengan menggunakan Python, diperlukan koding yang cukup rumit. Oleh karena itu, peserta magang mencari tahu bagaimana kodingannya dengan mencari *tutorial* di YouTube dan juga dengan bantuan ChatGPT.

Setelah mempelajari lebih banyak mengenai bahasa pemrograman Python untuk *data scrapping*, peserta magang menemukan sebuah *plugin* untuk *data scrapping* dengan menggunakan Python, yaitu BeautifulSoup. BeautifulSoup adalah sebuah *library* di Python yang memudahkan proses ekstraksi informasi dari halaman web. Hal ini sangat cocok bagi keperluan peserta magang untuk melakukan *data scrapping* dari berbagai *website*. Library ini bekerja dengan mengurai (parsing) dokumen HTML dan XML, memungkinkan pengguna untuk mengakses elemen-elemen halaman web melalui pemilihan *tag*, atribut, atau struktur tree DOM. Dengan BeautifulSoup, pengguna dapat dengan mudah menargetkan dan memodifikasi konten halaman *web*, seperti mengambil teks dari paragraf, mengumpulkan link, mengumpulkan data dari tabel, dan sebagainya.

Hal yang menjadi tantangan dengan menggunakan Python adalah selain kodingannya yang cukup rumit, setiap *website* memiliki *element* untuk data yang ingin di-*scrapping* yang berbeda-beda. Misalnya, terdapat *website* yang datanya terbentuk dari tabel sehingga harus mencari *element* `<tr>`, `<td>`, serta `<th>`, ada juga yang

menggunakan `<ul><li>`, `<div class>` biasa, dan masih banyak lagi. Pada akhirnya, setiap *element websitr* yang berbeda menyebabkan kodingannya harus berubah juga. Hal tersebut menjadi hambatan karena akan memperlambat proses *data scrapping*. Hambatan lain dengan menggunakan Python adalah prosesnya yang cukup lama. Hal ini dikarenakan Python memerlukan beberapa detik untuk menyalin data dalam *website* tersebut satu per satu sehingga memerlukan waktu yang tidak sebentar untuk melakukan *data scrapping* untuk satu *website* saja. Namun, menggunakan Python merupakan satu-satunya cara untuk melakukan *data scrapping* tanpa menggunakan Hexomatic sehingga tetap dilaksanakan meskipun terdapat beberapa kekurangan juga.

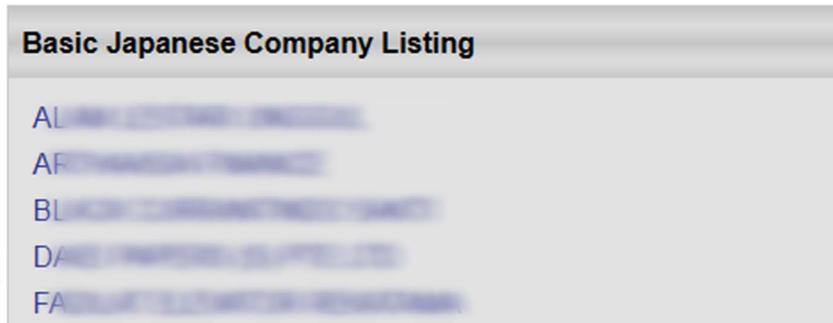
```

import requests
from bs4 import BeautifulSoup
import pandas as pd
# URL of the main page
main_url = "https://..."
# Function to scrape information from a company page
def scrape_company_info(company_url):
    response = requests.get(company_url)
    soup = BeautifulSoup(response.content, 'html.parser')
    # Find the table with class 'profile'
    profile_table = soup.find('table', class_='profile')
    # Check if the table is found
    if profile_table:
        # Extract information from the table
        company_info = {}
        for row in profile_table.find_all('tr'):
            columns = row.find_all(['td', 'th'])
            key = columns[0].get_text(strip=True)
            value = columns[2].get_text(strip=True)
            company_info[key] = value
        return company_info
    else:
        print(f"No profile table found on {company_url}")
        return None
# Fetch the main page content
main_page_response = requests.get(main_url)
main_page_soup = BeautifulSoup(main_page_response.content, 'html.parser')
# Find all links to company pages
company_links = main_page_soup.find_all('a', href=True)
# Create a list to store scraped data
data_list = []
# Loop through each company page and scrape information
for link in company_links:
    company_url = "https://..." + link['href'].replace("../", "")
    company_info = scrape_company_info(company_url)
    # Check if information is available
    if company_info:
        # Append the data to the list
        data_list.append(company_info)
# Create a DataFrame from the list
df = pd.DataFrame(data_list)
# Save the DataFrame to an Excel file
df.to_excel("scraped_jakartaBarat.xlsx", index=False)
print("Data saved to 'scraped_data.xlsx'")

```

Gambar 3.23 Kodongan *Data Scrapping* Python untuk Bentuk *Website* Tabel

Gambar 3.23 menunjukkan kodongan untuk melakukan *data scrapping* dengan menggunakan Python. *Website* yang diberikan oleh *supervisor* dalam hal ini adalah *website* yang memiliki data seperti tabel. Contohnya adalah sebagai berikut.



Gambar 3.24 Halaman *List* Perusahaan

Gambar 3.24 merupakan gambar mengenai halaman *list* perusahaan. Berdasarkan gambar tersebut, dapat dilihat bahwa *list* perusahaan berbeda dengan *website* sebelumnya. Kali ini, *list* perusahaan ditampilkan dengan menggunakan *table*. Apabila salah satu perusahaan tersebut diklik, akan memunculkan halaman berikut.



Gambar 3.25 Contoh Halaman Informasi Perusahaan

Gambar 3.25 merupakan gambar salah satu contoh halaman informasi perusahaan. Berdasarkan gambar tersebut, dapat dilihat bahwa struktur informasi perusahaan dalam *website* tersebut menggunakan *table* sehingga harus perlu penyesuaian kodingan berdasarkan *element* yang diperlukan untuk *data scrapping*.

Selanjutnya, kodingan pada gambar 3.22 memiliki beberapa kelemahan, salah satunya adalah tidak bisa *pagination*. Salah satu halaman *website* yang perlu dilakukan *scrapping* memiliki *pagination* sehingga harus ada modifikasi kodingan. Alhasil, kodingan tersebut ditambahkan kodingan berikut ini supaya bisa melakukan *pagination*.

```
# Loop through each page (assuming 9 pages)
for page_num in range(1, 10):
    # Fetch the main page content
    page_url = f"{main_url}&pageNum_company={page_num}"
    main_page_response = requests.get(page_url)
    main_page_soup = BeautifulSoup(main_page_response.content, 'html.parser')

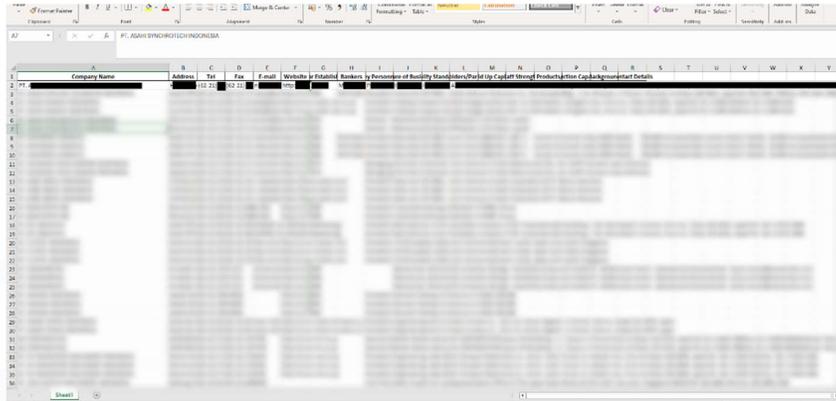
    # Find all links to company pages
    company_links = main_page_soup.find_all('a', href=True)

    # Loop through each company page and scrape information
    for link in company_links:
        company_url = "https://www.berita.com/" + link['href'].replace(".", "")
        company_info = scrape_company_info(company_url)

        # Check if information is available
        if company_info:
            # Append the data to the list
            data_list.append(company_info)
```

Gambar 3.26 Kodingan Tambahan *Pagination*

Gambar 3.26 merupakan kodingan tambahan untuk *pagination*. Berdasarkan gambar tersebut, kodingan tersebut menambahkan sebuah *array* yang telah ditentukan untuk halaman maksimal 10 halaman. Apabila halaman dalam *website* lebih dari 10, angka (1, 10) dapat diganti sesuai dengan jumlah halamannya. Hal yang dilakukan oleh kodingan ini adalah ia melakukan *looping* sebanyak jumlah halaman yang ditentukan untuk memeriksa apakah ada *element* yang ditentukan sebelumnya. Jika ditemukan, ia akan melakukan salin dan memasukannya ke *file* .xlsx nantinya. Berikut ini merupakan hasil salah satu *data scrapping* dengan menggunakan Python.



Gambar 3.27 Hasil *Data Scrapping* Menggunakan Python

Gambar 3.27 merupakan hasil *data scrapping* dengan menggunakan Python. Berdasarkan gambar tersebut, dapat dilihat bahwa hasil dari *data scrapping* tersebut dimasukkan ke dalam *file* dengan *format* .xlsx.

#### 3.3.4.3 Merapikan Data

Setelah *data scrapping* selesai dilaksanakan, terdapat satu hal lagi yang harus dilakukan, yaitu merapikan data. Hal ini dilakukan karena terkadang hasil dari *data scrapping* baik menggunakan Hexomatic maupun Python memiliki masalah yang sama, yaitu datanya tidak rapi. Terdapat sejumlah masalah yang menyebabkan data tersebut tidak bagus, yaitu data yang terduplikat banyak sekali, terdapat data *null*, sortir data tidak menggunakan abjad huruf awal (A-Z), dan penulisan data yang tidak. Hal tersebut mendorong peserta magang untuk merapikan data tersebut supaya pihak manapun di perusahaan yang nantinya akan menggunakan data tersebut dapat memanfaatkan data tersebut sesuai dengan tujuan dan kebutuhannya tanpa kebingungan melihat data yang berantakan. Cara untuk membersihkan datanya adalah dengan menggunakan pemrograman Python lagi untuk melakukan *data preparation* terhadap beberapa data yang nantinya akan digunakan oleh perusahaan.

### 3.3.4.3.1 Menghilangkan Data Duplikat

```
import pandas as pd

# Langkah 1: Membaca data dari file Excel
file_path = '...'xlsx'
df = pd.read_excel(file_path)

# Langkah 2: Menghapus duplikat berdasarkan kolom 'Company Name'
# keep='first' berarti hanya kemunculan pertama yang disimpan, sisanya dihapus
df_cleaned = df.drop_duplicates(subset=['Company Name'], keep='first')

# Langkah 3: Menyimpan DataFrame yang sudah dibersihkan ke file baru
df_cleaned.to_excel('...'xlsx', index=False)

print("Data duplikat telah dihapus dan hasilnya disimpan di '...'xlsx'.")
```

Gambar 3.28 Contoh Kodingan Menghilangkan Data Duplikat

Seperti yang sudah dijelaskan di bagian sebelumnya bahwa terdapat sejumlah hasil dari *data scrapping* yang menyebabkan data terduplikat. Gambar 3.28 menunjukkan sebuah kodingan untuk melakukan *remove* data yang terduplikat. Apa yang kodingan tersebut lakukan adalah menghilangkan satu baris data yang terdeteksi duplikat berdasarkan sebuah kolom data. Dalam hal ini kolom Company Name merupakan kolom yang diperiksa apakah ada data yang duplikat atau tidak. Jika ada, satu baris tersebut langsung dihapus.

### 3.3.4.3.2 Menghilangkan Data Null

```
import pandas as pd

null = pd.read_excel('...'xlsx')
null.head(5)
df = pd.DataFrame(null)
df['Name'].isnull().sum()
df_without_null = df.dropna(subset=['Name'])
df_without_null.isnull().sum()
output_file_path = '...'csv'
df_without_null.to_csv(output_file_path, index=False)
```

Gambar 3.29 Kodingan untuk *Remove Null*

Seperti yang telah dijelaskan di bagian sebelumnya bahwa hasil dari *data scrapping* banyak menyisakan data yang *null*. Alhasil, banyak baris data yang kosong. Hal tersebut menyebabkan data di dalam *file .xlsx* menjadi tidak enak untuk dipandang. Oleh karena itu, diperlukan sebuah cara cepat untuk menghilangkan data *null* tersebut. Cara cepat tersebut adalah dengan menggunakan Python. Gambar 3.29 merupakan gambar kodingan untuk me-*remove* data *null*. Apa yang kodingan tersebut lakukan adalah menghitung ada berapa jumlah data *null* di kolom Name dan kemudian menghilangkan satu baris kolom tersebut apabila terdapat data yang *null*.

#### 3.3.4.3.3 Sortir Data

```
import pandas as pd
# Read the Excel file
df = pd.read_excel(".....xlsx")
# Sort the DataFrame by the "Name" column
df_sorted = df.sort_values(by="Name")
# Write the sorted DataFrame to a new Excel file
df_sorted.to_excel(".....xlsx", index=False)
print("Data has been sorted alphabetically based on the Name column and saved to .....xlsx.")
```

Gambar 3.30 Kodingan untuk Sortir Data

Seperti yang telah dijelaskan di bagian sebelumnya bahwa terdapat hasil dari salah satu *data scrapping* yang tidak mengikuti urutan abjad *alphabet* secara *ascending* (A-Z). Oleh karena itu, diperlukan sebuah cara untuk menyortir data tersebut sehingga lebih enak untuk dibaca apabila disortir secara *ascending*. Gambar 3.30 merupakan sebuah kodingan Python yang berfungsi untuk melakukan penyortiran data. Data dalam kolom Name dalam hal ini disortir secara *ascending* dengan menggunakan abjad *alphabet* sehingga A-Z.

### 3.3.4.3.4 Memperbaiki Penulisan Data

```
import pandas as pd
import urllib.parse

# Load your Excel file into a pandas DataFrame
file_path = 'output_file6.xlsx' # Replace with your actual file path
df = pd.read_excel(file_path)

# Function to format the company link
def format_company_link(url):
    parts = url.split('/')

    # Lowercase the initial part of the URL
    formatted_url = f"{parts[0].lower()}://{parts[2].lower()}"

    # Replace spaces with %20 in the directory part
    directory_part = parts[4] if len(parts) > 4 else ""
    formatted_directory = '/'.join(urllib.parse.quote(part) for part in directory_part.split('/'))
    formatted_url += f"/directory/{formatted_directory}"

    return formatted_url

# Apply the function to the "Company Link" column
df['Company Link'] = df['Company Link'].apply(format_company_link)

# Save the modified DataFrame back to the Excel file
df.to_excel('output_file7.xlsx', index=False)
```

Gambar 3.31 Kodingan Pembetulan Penulisan Data

Seperti yang telah dijelaskan di bagian sebelumnya bahwa terdapat sejumlah penulisan data yang tidak tepat. Hal ini dikarenakan data yang diambil berbentuk URL sehingga akan banyak sekali URL yang tidak sesuai dengan URL aslinya. Gambar 3.31 menunjukkan kodingan mengenai bagaimana cara memperbaiki penulisan URL yang benar. Berdasarkan kasus tersebut, data URL tidak memiliki “/” sedangkan “/” diperlukan apabila URL tersebut memiliki kata yang dispasi. Oleh karena itu, kodingan tersebut menunjukkan bagaimana Python dapat menambahkan “/” di setiap spasi kata-kata di URL sehingga URL benar dan dapat digunakan.

Setelah semua data dirapikan, *supervisor* meminta data-data tersebut untuk di-*upload* ke Google Drive yang nantinya akan di-*share* ke karyawan-karyawan yang membutuhkan data tersebut termasuk divisi *web developer* yang diminta untuk memasukannya ke dalam *database* iiga.shop.

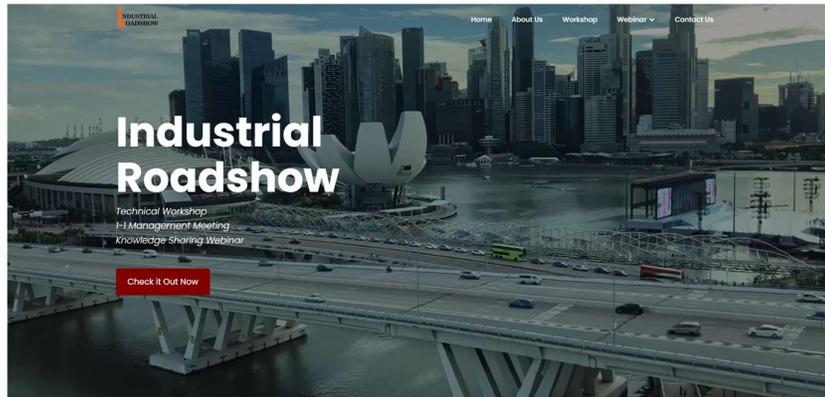
Dengan demikian, itulah tahapan-tahapan untuk melakukan *data scrapping* baik dengan menggunakan *tools* Hexomatic maupun pemrograman Python. Dalam hal ini, peserta magang dan *supervisor* belajar sangat banyak mengenai pemrograman dan *data scrapping* sehingga hal ini memperluas pengetahuan baik untuk peserta magang maupun untuk *supervisor*.

### **3.3.5 Membuat Industrialroadshow.com**

*Website* industrialroadshow.com merupakan sebuah *website* milik perusahaan yang berisi tentang *event roadshow* yang pernah diadakan oleh perusahaan sejak 2010-2023. Selain itu, *website* ini juga menunjukkan video mengenai webinar yang pernah diadakan oleh perusahaan. *Website* ini memiliki lima menu utama, yaitu *homepage*, *about us*, *workshop*, *webinar*, dan *contact us*. Dalam proses pembuatan *website* ini, peserta magang tidak sendirian, tetapi dibantu oleh salah satu peserta magang lain yang berasal dari divisi UI/UX yang kebutuhan memahami konsep pembuatan *website* sehingga proyek pengembangan *website* ini berjalan dengan cepat. Peserta magang juga dibantu oleh karyawan-karyawan IT perusahaan yang memang lebih paham tentang pembuatan *website*. Peserta magang juga diberikan semacam *template* kodingan supaya mempermudah peserta magang dan juga karyawan IT untuk memahami tampilan yang diinginkan. Berikut merupakan penjelasan halaman dan cara membuatnya.

#### **3.3.5.1 Homepage**

*Homepage* merupakan halaman utama dari *website* ini. *Homepage* menampung berbagai informasi mengenai *roadshow* ataupun *event* yang pernah digelar oleh perusahaan.

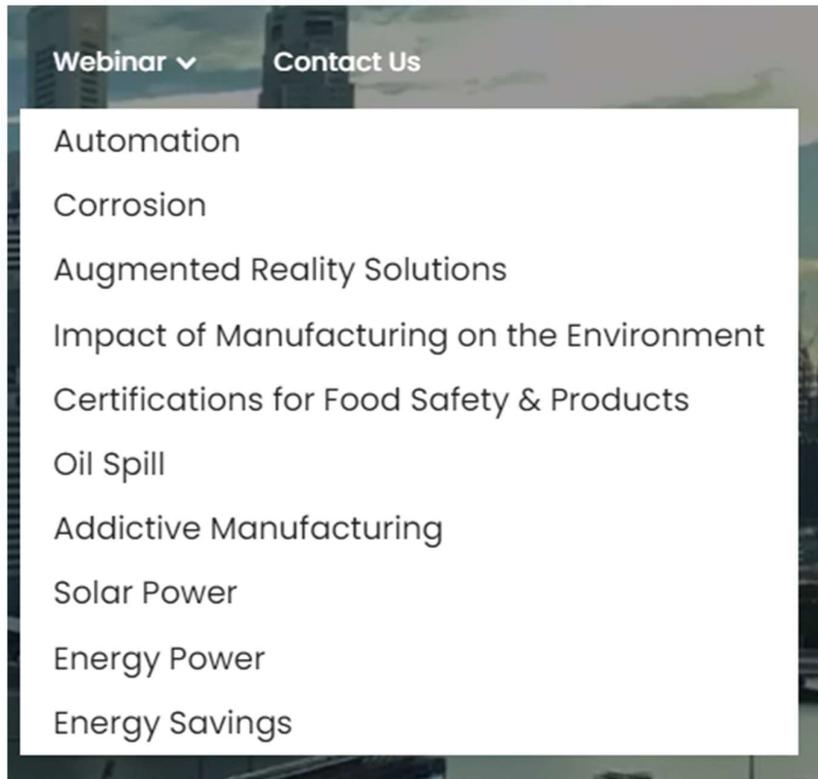


Gambar 3.32 *Homepage* bagian atas

Gambar 3.32 merupakan bagian *homepage* dari *website* [industrialroadshow.com](http://industrialroadshow.com). Bagian paling atas terdapat sebuah *navigation bar* yang berfungsi untuk me-*link* dari satu halaman ke halaman lainnya. Bagian pojok kiri *navigation bar* terdapat sebuah logo *event* perusahaan. Logo tersebut diletakkan di dalam sebuah *containter navigation bar* sehingga logo tersebut memiliki ukuran yang sesuai dengan *navigation bar* tersebut. Apabila logo tersebut diklik, akan me-*redirect* ke halaman utama atau *homepage* dari *website* ini.

Bagian berikutnya adalah bagian *navigation* menu yang terletak di sebelah kanan. *Navigation* menu tersebut berfungsi untuk me-*redirect* ke halaman-halaman lain yang ada di *website*. *Navigation* menu tersebut dibuat dengan menggunakan fungsi `<nav>` yang di dalamnya terdapat fungsi `<a>`. Fungsi `<nav>` dalam hal ini berfungsi untuk membuat tampilan seperti di gambar 3.32, yaitu menjadi seperti menu *link* yang berbaris, sedangkan `<a>` berfungsi untuk meletakkan *link* halaman *website* tersebut sesuai dengan peruntukannya. *Navigation Homepage* akan *direct* ke halaman *Homepage*, *About Us* akan *direct* ke halaman *About Us*, *Workshop* akan beralih ke *website* lain, yaitu [sgexpert.com](http://sgexpert.com), *Webinar* dalam hal ini memiliki *drop-down*

menu lagi yang akan *link* ke berbagai halaman, dan *Contact Us* akan membawa ke halaman *Contact Us*.



Gambar 3.33 *Drop-down Menu Webinar*

Gambar 3.33 menunjukkan gambar *drop-down menu* Webinar. *Navigation* menu Webinar berbeda dengan *Navigation* menu lainnya. Apabila tulisan Webinar diklik, akan memunculkan halaman Webinar, namun apabila logo *drop-down* yang diklik akan memunculkan berbagai pilihan halaman yang nantinya jika di klik akan *direct* ke halaman tersebut. Cara pembuatan menu Webinar ini sedikit berbeda dengan menu lainnya. Menu Webinar tidak dibuat dengan menggunakan `<a>` saja, tetapi dengan menggunakan `<div class>` yang berfungsi untuk meng-*grouping* tulisan Webinar dan tanda *drop-down* tersebut. Dalam `<div class>` tersebut terdapat sebuah `<div class` lagi yang berfungsi memunculkan tanda *drop-*

*down* dan terdapat fungsi `<a>` yang berfungsi untuk menaruh tulisan “Webinar” dan link ke halaman Webinar. Kemudian, terdapat sebuah fungsi `<nav>` lagi yang di dalamnya terdapat fungsi `<a>` yang berfungsi untuk menampilkan *list* Webinar seperti gambar 3.33 apabila tanda *drop-down* diklik. *List* tersebut akan *direct* ke halaman Webinar berdasarkan kategori dari *list* tersebut.

Hal yang menarik dari *website* ini adalah ketika *user* melakukan *scrolling*, bagian *navigation bar* akan berubah warnanya.



Gambar 3.34 *Navigation Bar* Berada di Paling Atas



Gambar 3.35 *Navigation Bar* Berada di Bagian Bawah Gambar



Gambar 3.36 *Navigation Bar* Ketika Sudah Melewati Gambar

Gambar 3.34, 3.35, dan 3.36 menunjukkan sebuah animasi dari *navigation bar*. Berdasarkan gambar tersebut, dapat dilihat bahwa ketika berada di bagian paling atas, *navigation bar* memiliki *background* yang transparan. Kemudian, pada saat *user* melakukan *scroll* tetapi belum melewati gambar, warna *background*-nya berwarna keabu-abuan. Ketika sudah melewati gambar, warna *background*-nya akan berubah menjadi putih. Animasi tersebut dibuat dengan menggunakan seperangkat koding JavaScript yang bisa melakukan *timing* untuk memainkan *opacity background navigation bar* sehingga apabila berada di paling atas *opacity*

*background*-nya adalah 0%, dan akan berjalan naik seiring melewati gambar hingga pada akhirnya setelah melewati gambar *opacity*-nya adalah 100%. Kemudian, seiring berjalannya *scrolling*, bagian bawah *navigation bar* terdapat sebuah garis berwarna merah *maroon*. Fungsi dari garis tersebut adalah untuk memberi tahu seberapa jauh *user* men-*scroll* halaman *website* tersebut. Garis tersebut akan memanjang jika *user* melakukan *scroll* ke bawah dan akan memendek ketika *user* melakukan *scroll* ke atas. Hal tersebut dibuat dengan menggunakan JavaScript dengan memainkan *width* dari garis tersebut.

Selanjutnya, kembali lagi ke gambar 3.32. Berdasarkan gambar tersebut, dapat dilihat bahwa terdapat sebuah gambar kota. Hal tersebut sebenarnya bukanlah sebuah gambar, melainkan sebuah *video* yang di-*looping*. Video tersebut diletakkan dan diberikan fungsi *z-index* yang berfungsi untuk memindahkan posisi video tersebut supaya berada di belakang *navigation bar* dan juga tulisan. Tulisan “Industrial” dan “Roadshow” dibuat dengan menggunakan fungsi `<h1>` yang dikostumisasi dengan menggunakan *.css* supaya berbentuk dan terletak seperti di gambar 3.32, sedangkan tulisan “Technical Workshop”, “1-1 Management Meeting”, dan “Knowledge Sharing Webinar” dibuat dengan menggunakan `<div class>` yang dikostumisasi juga melalui *.css* sehingga hasil akhirnya seperti di gambar 3.32. Terakhir, terdapat sebuah *button* yang bertuliskan “Check it Out Now” yang apabila diklik akan *scroll* down ke bagian *timeline* kegiatan perusahaan. *Button* dibuat dengan fungsi `<a>` dan diberikan warna *maroon*. *Button* juga diberikan seperangkat kodingan JavaScript sehingga ia dapat melakukan *scrolling* ke bagian lain dalam halaman tersebut.

Selanjutnya adalah bagian *timeline* kegiatan perusahaan yang menampilkan kegiatan-kegiatan perusahaan selama 2010-2023.



Gambar 3.37 Bagian *Timeline* Kegiatan Perusahaan

Gambar 3.37 merupakan sebuah gambar yang menunjukkan bagian *timeline* kegiatan *roadshow* perusahaan mulai sejak 2010. Kurang lebih tampilannya hampir sama untuk setiap tahunnya, yaitu terdapat tulisan judul acara, gambar atau video, garis berwarna merah, dan lingkaran yang di dalamnya bertuliskan tahun kegiatan. Pertama-tama, bagian ini dibuat dengan menggunakan sebuah `<div class>` yang berfungsi membungkus satu ke satuan *element* dalam satu tahun. Kemudian, terdapat sebuah `<div>` lagi untuk membungkus *element* sebelah kanan dan kiri. Dalam `<div>` tersebut, terdapat sebuah fungsi `<img>` untuk meletakkan gambar atau `<a>` untuk memberikan *link* video melalui YouTube. Semua `<div class>` diatur dengan menggunakan `.css` sedemikian rupa sampai bentuknya menyerupai gambar 3.37. Kemudian, di bagian tengah terdapat sebuah `<div>` lagi yang digunakan untuk meletakkan sebuah garis merah dan juga sebuah lingkaran berisikan tulisan tahun. Garis dalam hal ini dibuat hanya dengan menggunakan `.css` saja. Hal ini juga diberikan sebuah fungsi JavaScript yang dapat membuat animasi *scrolling* sehingga apabila *user* melakukan *scrolling*, garis yang sebelumnya berwarna abu-abu akan berubah menjadi merah kecoklatan seiring dengan berjalannya *scrolling* dan juga lingkaran yang berisikan tulisan tahun akan ikut bergerak turun ke bawah juga.

Setelah satu *element* jadi, sisanya hanya tinggal *copy* dan *paste* untuk mengganti tahun, judul kegiatan, gambar, dan videonya. Untuk saat ini, *timeline* hanya berhenti pada 2023 dikarenakan tahun 2024 belum ada *event* dari perusahaan dan juga tahun 2024 belum berakhir.

Bagian selanjutnya adalah bagian untuk mempromosikan *event-event* perusahaan.



Gambar 3.38 Bagian Promo *Event* Perusahaan

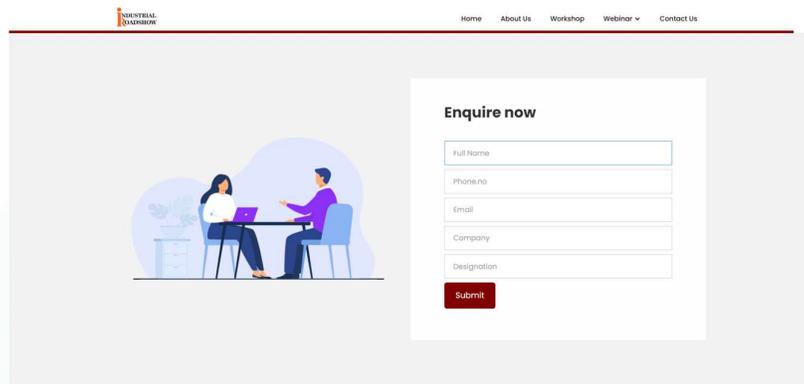
Gambar 3.38 merupakan sebuah gambar yang menunjukkan bagian selanjutnya dari *website* ini, yaitu bagian promosi *event* perusahaan. Dalam hal ini, terdapat tiga hal yang dipromosikan oleh perusahaan yaitu *event* Technical Workshop, Webinar, dan Management Meeting Robotics. Desain dari bagian promosi *event* dalam *website* ini sama, yaitu terdapat judul yang dibuat dengan fungsi `<h2>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Kemudian, di bawah judul terdapat sebuah garis merah yang dibuat dengan menggunakan `.css` dan juga JavaScript karena apabila *cursor mouse* didekatkan ke judul *event*, garis tersebut barulah muncul dan jika dijauhkan, garis tersebut akan menghilang. Kemudian, deskripsi acara tersebut dibuat dengan fungsi `<h3>` `.css` untuk mengatur ukuran dan posisi sampai

sedemikian rupa. Terdapat juga sebuah *button* “Check it Out Now” yang dibuat dengan fungsi `<a>` dan diberikan *background* merah *maroon*. *Button* tersebut seharusnya akan *direct* ke halaman atau *website* daftar *event* tersebut, tetapi karena *event*-nya sudah selesai, *link* di *button* tersebut dihilangkan sehingga sekarang *button* tersebut tidak berfungsi seperti yang seharusnya. Selanjutnya, diletakkan sebuah gambar atau *banner* mengenai *event* tersebut yang dibuat dengan fungsi `<img>` dan *.css* untuk mengatur ukuran dan posisi sampai sedemikian rupa.



Gambar 3.39 Bagian About Us

Gambar 3.39 merupakan bagian selanjutnya adalah bagian About Us. Dalam bagian ini, terdapat sebuah judul bertuliskan “About Us” yang dibuat dengan fungsi `<h2>` dan *.css* untuk mengatur ukuran dan posisi sampai sedemikian rupa. Kemudian, di bawah judul terdapat sebuah garis merah yang dibuat dengan menggunakan *.css* dan juga JavaScript karena apabila *cursor mouse* didekatkan ke judul *event*, garis tersebut barulah muncul dan jika dijauhkan, garis tersebut akan menghilang. Terdapat juga sebuah *button* “Check it Out Now” yang dibuat dengan fungsi `<a>` dan diberikan *background* merah *maroon*. *Button* tersebut akan *direct* ke halaman About Us. Selanjutnya, diletakkan sebuah gambar logo perusahaan yang dibuat dengan fungsi `<img>` dan *.css* untuk mengatur ukuran dan posisi sampai sedemikian rupa.



Gambar 3.40 Bagian *Form* Pertanyaan

Gambar 3.40 merupakan bagian *form* pertanyaan. Bagain ini berfungsi untuk *user* memasukkan data-datanya untuk menanyakan hal-hal tertentu nantinya. Jujur, bagian ini peserta magang tidak banyak tahu cara membuatnya dikarenakan bagian ini tersambung dengan *database* milik perusahaan. *Database* dalam hal ini diperlukan untuk mengetahui data-data *user* yang mengisi *form* tersebut untuk nantinya di-*follow up* oleh karyawan yang bertanggung jawab akan hal tersebut. *Database* yang digunakan dalam hal ini sangat *confidential* sehingga peserta magang tidak diberikan akses. Oleh karena itu, bagian ini dipegang oleh karyawan IT yang memang sudah pengalaman mengelola *database* tersebut. Terdapat sejumlah hal yang dikerjakan oleh peserta magang, yaitu meletakkan gambar di sebelah kiri dengan fungsi `<img>` dan `.css` untuk mengatur posisi dan ukuran sampai sedemikain rupa. Selain itu, bentuk formulir juga dikerjakan oleh peserta magang dengan judul tulisan “Enquire now” yang dibuat dengan fungsi `<h4>` dan `.css` untuk mengatur ukuran dan tata letaknya sehingga seperti di gambar 3.39. Kemudian, kita menggunakan fungsi `<form>` yang di dalamnya terdapat fungsi `<input type=“text”>`, `<input type=“tel”>`, dan `<input type=“email”>`. Fungsi-fungsi `<input>` tersebut diberikan kodingan tambahan, yaitu `maxlength=“256”` agar karakter yang

dimasukkan oleh *user* tidak lebih dari 256. Terakhir terdapat *button* “Submit” untuk men-*submit form* yang telah diisi oleh *user*. *Button* tersebut dibuat dengan menggunakan `<input type=”submit”>` yang diberikan warna merah *maroon*. *Button* tersebut terhubung ke seperangkat JavaScript dan juga ke *database* sehingga peserta magang bukanlah yang mengerjakan bagian tersebut. Semua form dibungkus dalam satu `<div class>` agar bisa diberikan *background* seperti gambar 3.40.



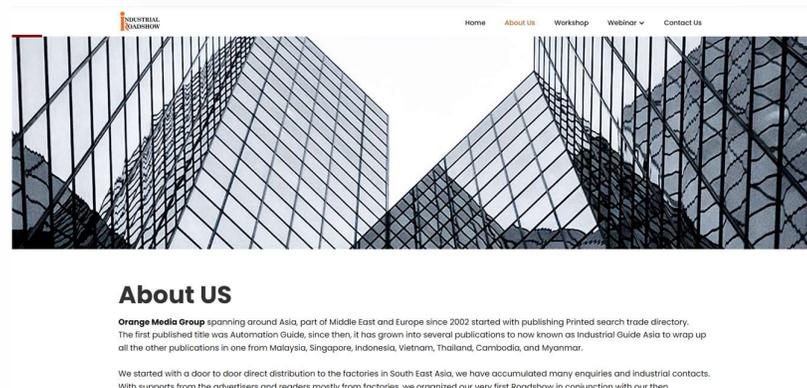
Gambar 3.41 *Footer*

Bagian terakhir dalam halaman *Homepage* adalah bagian *footer*. Gambar 3.41 merupakan bagian *footer* dari halaman *Homepage* dan juga *footer* untuk halaman-halaman lainnya nanti. Berdasarkan gambar tersebut, terdapat beberapa *element* dalam *footer*. Pertama, di bagian sebelah kiri terdapat logo perusahaan yang dibuat dengan fungsi `<img>`. Fungsi `<img>` tersebut terletak di dalam fungsi `<a>` yang berfungsi untuk meletakkan *link*. *Link* yang diletakkan dalam hal ini mengarah ke halaman *Homepage* lagi sehingga apabila *user* klik gambar tersebut, akan diarahkan kembali ke *Homepage*. Peserta magang juga menggunakan `.css` untuk mengatur ukuran dan posisi gambar tersebut. Kemudian, terdapat logo-logo media sosial di bawah gambar logo perusahaan yang berfungsi untuk me-*link* ke halaman media sosial perusahaan sesuai dengan logo media sosialnya. Sama seperti gambar logo perusahaan, dalam hal ini digunakan fungsi `<a>` untuk meletakkan *link* media sosial perusahaan, `<img>` untuk meletakkan logonya, dan `.css` untuk mengatur tata letak dan ukuran gambarnya. Kemudian, berdasarkan

gambar 3.41, terdapat tulisan di bawah gambar logo media sosial yang dibuat dengan fungsi `<h6>` dan `.css` untuk mengatur ukuran dan letaknya. Berikutnya, di bagian sebelah kanan, terdapat dua bagian, yaitu “Videos” dan “Quick Links”. Judul “Videos” dan “Quick Links” dibuat dengan menggunakan `<h6>`, sedangkan tulisan-tulisan lain di bawahnya dibuat dengan menggunakan fungsi `<a>` karena tulisan-tulisan tersebut akan *men-direct* ke halaman-halaman lain. Terakhir, di bagian paling bawah terdapat dua baris tulisan yang dibuat dengan fungsi `<a>` karena tulisan tersebut memiliki sebuah *link* ke halaman *website* yang saat ini sudah tidak aktif lagi.

### 3.3.5.2 About Us

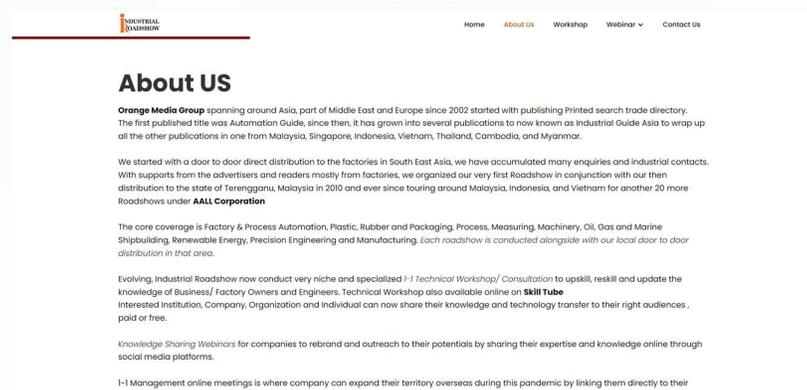
*About Us* merupakan sebuah halaman yang berfungsi untuk menampilkan deskripsi perusahaan. Berikut ini adalah uraian tampilan dan cara membuatnya.



Gambar 3.42 Tampilan Halaman *About Us* Bagian Atas

Gambar 3.42 merupakan halaman *About Us* bagian atas. Sama seperti di halaman *Homepage*, bagian paling atas adalah sebuah *navigation bar* yang memiliki desain yang sama dengan *Homepage*. Bahkan, masih terdapat garis merah yang memanjang apabila *user* melakukan *scroll* ke bawah dan memendek apabila *user* melakukan *scroll* ke atas. Hal yang membedakan adalah *background*

dari *navigation bar* tidak berubah *opacity*-nya seperti di halaman *Homepage* sehingga mulai dari atas sampai bawah gambar, *background* dari *navigation bar* tidaklah berubah. Hal lain yang berubah juga adalah sekarang, tulisan “About Us” di *navigation bar* berwarna *orange* yang menyatakan bahwa *user* berada di halaman “About Us”. Selanjutnya adalah terdapat sebuah gambar gedung yang diletakkan dengan menggunakan fungsi `<img>` dan `.css` yang berfungsi untuk mengatur tata letak dan ukuran gambar supaya seolah-olah berada di belakang *element* selanjutnya.



Gambar 3.43 Bagian Paragraf *About Us*

Gambar 3.43 merupakan sebuah gambar yang menunjukkan bagian selanjutnya dalam halaman *About Us*. Berdasarkan gambar tersebut, dapat dilihat bahwa terdapat judul tulisan “About Us” yang dibuat dengan fungsi `<h2>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Selanjutnya, terdapat sebuah paragraf yang bercerita sedikit tentang Orange Media Group yang dibuat dengan menggunakan fungsi `<p>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Kemudian, terdapat beberapa kata yang dicetak tebal (*bold*). Hal tersebut merupakan sebuah *link* yang akan *direct* ke halaman lain atau *website* lain. Kata-

kata tersebut dibuat dengan fungsi `<a>` yang kemudian diberikan *link* yang diletakkan di dalam fungsi `<p>`.



Gambar 3.44 Bagian *Our Past Events* dan *Footer*

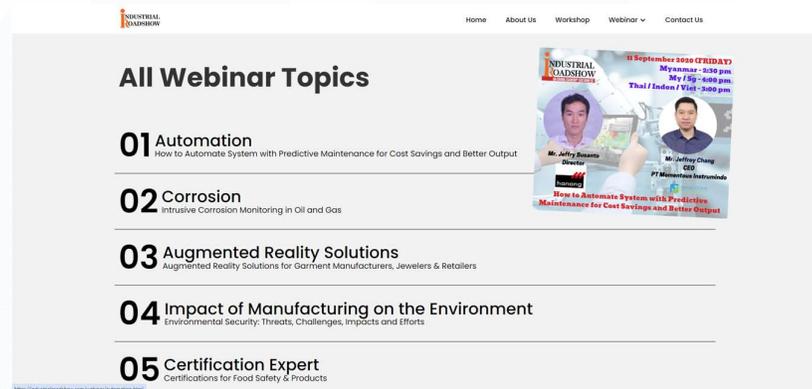
Gambar 3.44 merupakan sebuah gambar yang menunjukkan bagian bawah dari halaman *About Us* yang memperlihatkan bagian *Our Past Events* atau yang menunjukkan *event-event* yang sudah pernah dibuat oleh perusahaan. Judul yang bertuliskan “Our Past Events” dibuat dengan fungsi `<h2>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Kemudian untuk gambar-gambar logo *event* di bawahnya dibuat dengan menggunakan fungsi `<img>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Gambar-gambar tersebut sebenarnya merupakan sebuah animasi JavaScript yang bisa membuat gambar-gambar tersebut bergerak secara perlahan ke kiri untuk yang bagian atas dan ke kanan untuk yang bagian bawah untuk menampilkan lebih banyak lagi logo-logo *event* perusahaan. Bagian terakhir dalam halaman *About Us* merupakan sebuah *footer* yang sama dengan yang sudah dijelaskan di bagian halaman *Homepage*.

### 3.3.5.3 *Workshop*

Secara singkat saja, bagian *Workshop* merupakan sebuah bagian dalam *navigation bar* yang berfungsi untuk men-*direct* ke

halaman *website* *sgexpert.pro*. *Workshop* sendiri tidak memiliki halaman khusus dalam *website* ini.

### 3.3.5.4 Webinar



Gambar 3.45 Halaman Webinar

Gambar 3.45 merupakan gambar yang menunjukkan halaman Webinar. Berdasarkan gambar tersebut, dapat dilihat bahwa bagian paling atas halaman Webinar merupakan sebuah *navigation bar* yang sama dengan yang ada di halaman *About Us*. Terdapat sebuah judul yang bertuliskan “All Webinar Topics” yang dibuat dengan fungsi `<h2>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Kemudian, angka 01, 02, 03, dan seterusnya dibuat dengan fungsi `<h2>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Kemudian, untuk kata-kata di sebelah angka, seperti “Automation”, “Corrosion”, dan sebagainya menggunakan fungsi `<h3>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Untuk tulisan di bawahnya, menggunakan fungsi `<h5>` dan `.css` untuk mengatur ukuran dan posisi sampai sedemikian rupa. Kemudian, terdapat sebuah garis juga di bawahnya yang dibuat dengan `.css`. Selanjutnya, apabila *cursor mouse* didekatkan ke salah satu topik webinar, akan memunculkan sebuah gambar seperti di gambar 3.44 dan juga akan

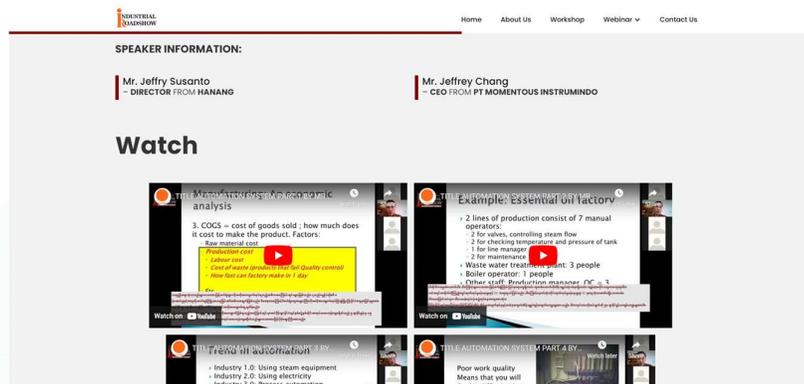
men-trigger sebuah animasi di bagian angka sehingga angkanya bisa naik dan turun. Hal tersebut dibuat dengan menggunakan JavaScript. Apabila diklik salah satu topik tersebut, akan membuka sebuah halaman *website* yang sesuai dengan topik tersebut. Berikut ini merupakan contohnya.



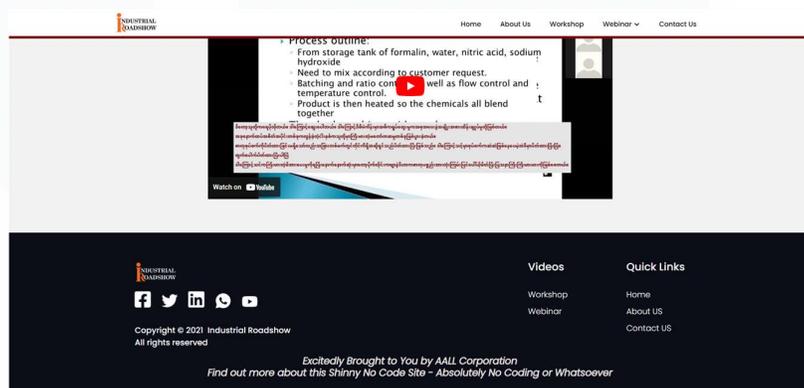
Gambar 3.46 Halaman Webinar *Automation* Bagian Atas



Gambar 3.47 Halaman Webinar *Automation* Bagian Tengah I



Gambar 3.48 Halaman Webinar *Automation* Bagian Tengah II

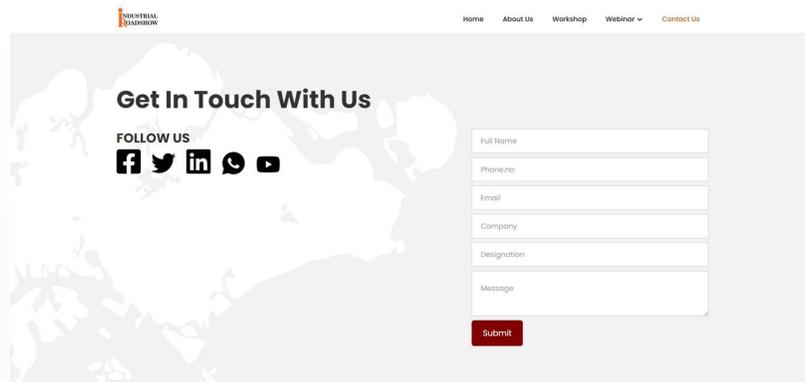


Gambar 3.49 Halaman Webinar *Automation* Bagian Bawah

Gambar 3.46 – 3.49 merupakan gambar-gambar yang menunjukkan salah satu halaman Webinar, yaitu *Automation*. Halaman ini dapat diakses melalui klik di salah satu topik di gambar 3.45 atau dengan klik di *drop-down* menu seperti di gambar 3.33. Setiap halaman Webinar tersebut memiliki desain yang relatif sama. Bagian atas terdapat sebuah *navigation bar* yang sama dengan halaman *About Us* di mana *background* dari *navigation bar* tersebut tidak berubah *opacity*-nya. Kemudian, terdapat sebuah gambar yang merepresentasikan setiap topik webinar di setiap halaman tersebut. Hal ini dibuat dengan menggunakan fungsi `<img>` dan `.css` untuk menyesuaikan agar gambar berada seolah-olah di belakang *element* berikutnya dan *widescreen*. Kemudian, terdapat sebuah judul topik

webinar yang sekaligus menjadi judul dari halaman tersebut yang dibuat dengan menggunakan fungsi `<h2>` dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Selanjutnya, terdapat gambar poster webinar yang dibuat dengan menggunakan fungsi `<img>` dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Berikutnya, terdapat sebuah tulisan “SPEAKER INFORMATION:” yang dibuat dengan fungsi `<h2>` dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Kemudian, dibentuklah sebuah `<div class>` untuk menaruh nama dan jabatan dari pembicara webinar tersebut. Keduanya dibuat dengan fungsi `<h6>` dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Terdapat juga sebuah garis merah di sebelah kiri nama pembicara yang dibuat dengan menggunakan `.css`. Selanjutnya, terdapat sebuah judul yang bertuliskan “Watch” yang dibuat dengan menggunakan fungsi `<h2>` dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Kemudian, terdapat beberapa video YouTube yang di-embed ke dalam *website* ini dengan menggunakan fungsi `iFrame` dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Terakhir, bagian paling bawah merupakan sebuah *footer* yang sama seperti di halaman-halaman sebelumnya.

#### **3.3.5.5 Contact Us**



Gambar 3. 50 Halaman *Contact Us*

Gambar 3.50 merupakan sebuah gambar yang menunjukkan halaman *Contact Us*. Halaman *Contact Us* merupakan sebuah halaman yang berfungsi untuk *user* bisa mengirimkan pesan ke perusahaan. Sama seperti bagian *form* pertanyaan di *homepage*, peserta magang juga tidak banyak bertugas di bagian ini dikarenakan bagian ini, terutama *form*, terhubung langsung ke *database* perusahaan yang peserta magang tidak diberikan aksesnya. Hal yang peserta magang buat di halaman ini adalah *front-end* dari halaman ini. Bagian atas terdapat sebuah *navigation bar* yang sama dengan yang di halaman *About Us*. Kemudian, terdapat sebuah *background* di belakang tulisan-tulisan dan juga *form*. Hal ini dibuat dengan cara meletakkan *file* gambar *background* tersebut di *file .css* dan kemudian memainkan fungsi *z-index* untuk mengatur tata letak *background* tersebut supaya berada di belakang semua *element* halaman *website* ini. Kemudian, terdapat juga tulisan “Get In Touch With Us” yang dibuat dengan fungsi `<h2>` dan *.css* untuk menyesuaikan ukuran dan posisi sedemikian rupa. Tulisan “Follow Us” yang dibuat dengan fungsi `<h1>` dan *.css* untuk menyesuaikan ukuran dan posisi sedemikian rupa. Kemudian terdapat logo-logo media sosial yang jika di klik akan mengarahkan langsung ke halaman media sosial perusahaan. Hal ini dibuat dengan

menggunakan fungsi `<a>` untuk menampung *link* media sosial tersebut dan kemudian di dalam fungsi `<a>` tersebut terdapat sebuah fungsi `<svg>` untuk meletakkan *icon* atau logo-logo media sosial tersebut. Selanjutnya, untuk membuat *form*, digunakanlah fungsi `<form>` dan di dalamnya terdapat fungsi `<input type="text">`, `<input type="tel">`, dan `<input type="email">`. Fungsi-fungsi `<input>` tersebut diberikan kodingan tambahan, yaitu `maxlength="256"` agar karakter yang dimasukkan oleh *user* tidak lebih dari 256. Terakhir terdapat *button* "Submit" untuk men-*submit form* yang telah diisi oleh *user*. *Button* tersebut dibuat dengan menggunakan `<input type="submit">` yang diberikan warna merah *maroon*.

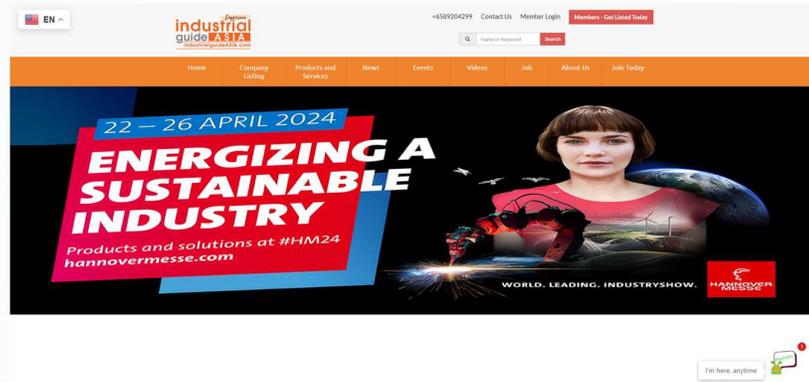
Dengan demikian, seperti itulah gambaran proses pembuatan *website* ini. Tentu *website* ini tidak dikerjakan oleh peserta magang sendirian, tetapi terdapat beberapa pihak yang turut serta membantu sehingga lebih cepat selesai. Setelah selesai, kodingan dan juga *assets* dari *website* ini di-*upload* ke penyedia *hosting*, yaitu iPower. iPower akan melakukan *hosting* sehingga *website* ini dapat dipakai dan diperlihatkan secara umum.

### 3.3.6 Merancang Ulang *Website iiga.shop*

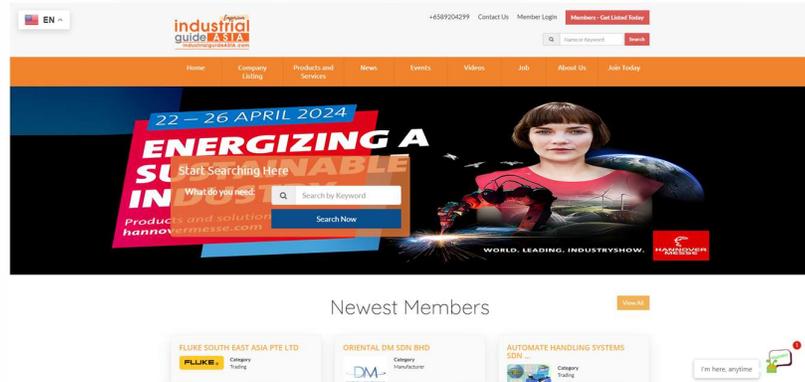
Seperti yang sudah dijelaskan di bagian sebelumnya bahwa *iiga.shop* merupakan *website e-commerce* milik perusahaan. Perancangan ulang *website iiga.shop* merupakan projek utama untuk peserta magang. Dalam hal ini, peserta magang tidak sendirian, tetapi dibantu oleh beberapa pihak, seperti *supervisor*, karyawan-karyawan IT, dan juga salah satu peserta magang lain yang berada di divisi UI/UX. Terdapat sejumlah halaman yang didesain ulang dalam projek ini, yaitu *Homepage*, *Login*, *Join Member*, *About Us*, *Contact Us*, *Privacy Policy*, *Terms and Conditions*, *Member Categories*, *Get Quotes*, dan *Upcoming Community Events*. Perlu diingat bahwa halaman-halaman yang baru ini belum sepenuhnya disetujui oleh *supervisor* sehingga halaman yang baru ini belum di-*deploy* oleh

perusahaan. Masih terdapat kemungkinan untuk direvisi atau sepenuhnya tidak digunakan. Berikut ini merupakan uraian dan cara membuatnya.

### 3.3.6.1 *Homepage*



Gambar 3.51 Halaman *Homepage* Baru Bagian Atas



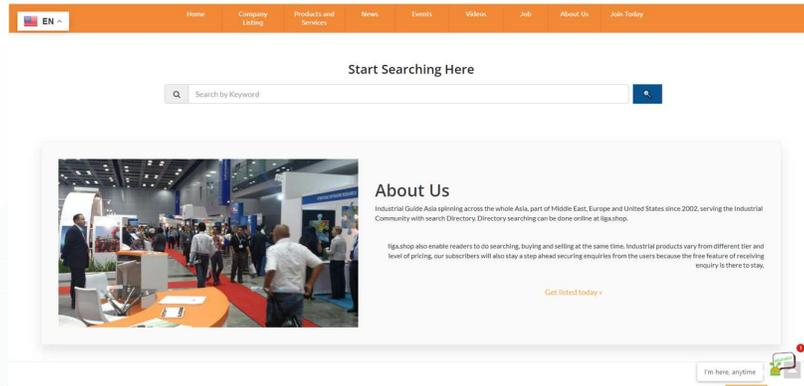
Gambar 3.52 Halaman *Homepage* Lama Bagian Atas

Gambar 3.51 merupakan gambar yang menunjukkan halaman *Homepage* bagian atas yang baru, sedangkan gambar 3.52 merupakan gambar yang menunjukkan halaman *Homepage* yang lama bagian atas. Tidak banyak hal yang berubah dalam hal ini karena bagian *header* masih sama persis. Hal yang membedakan adalah bagian bawahnya *header*, yaitu bagian *banner*. Bagian

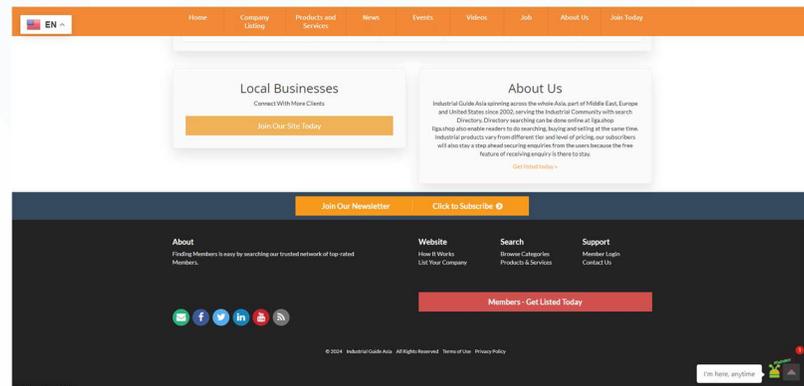
tersebut sebenarnya merupakan bagian *banner slider* di mana *banner* tersebut dapat berganti dengan menggeser ke sebelah kiri kurang lebih per lima detik. Hal tersebut dibuat dengan menggunakan JavaScript. Hal yang dibedakan adalah *width* dari halaman yang baru diperbesar supaya *banner-banner* yang baru bisa dimasukkan ke dalam *database* tanpa disesuaikan ulang resolusinya seperti yang lama. *Banner-banner* yang baru memiliki resolusi dengan *width* yang lebih besar dari pada *banner-banner* yang sudah lama. Maka, *width* tersebut perlu diperbesar. Jika tidak diperbesar *width*-nya, hal yang akan terjadi adalah gambar *banner* jadi *stretch*. Oleh karena halaman *website* yang baru belum di-*deploy* oleh perusahaan dan perusahaan masih menggunakan halaman *website* yang lama yang di mana masih memiliki masalah di *banner slider* tersebut, terpaksa *banner-banner* yang baru harus di-*stretch* supaya sesuai dengan *space width* dari *banner slider* yang lama. Berikutnya, hal yang hilang juga adalah bagian *search*. Bagian *search* dalam halaman *website* yang baru bukan dihilangkan, melainkan dipindahkan supaya tidak menutupi *banner website*. Sebenarnya, desain dari *banner-banner* tersebut menyesuaikan posisi *search* dari halaman *website* yang lama. Namun, seiring berjalannya waktu, *banner-banner* yang didesain sudah tidak lagi menyesuaikan posisi *search*. Oleh karena itu, posisi *search* dipindahkan.

Dalam *header*, sebelah kiri terdapat sebuah logo perusahaan. Hal tersebut dibuat dengan fungsi `<a>` untuk meletakkan *link* yang di dalamnya terdapat fungsi `<img>` untuk meletakkan gambar logonya dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Apabila di klik, ia akan *direct* ke halaman utama *website* lama untuk saat ini dikarenakan halaman utama *website* baru belum di-*deploy*. Kemudian, di bagian kanan, terdapat sebuah *list* menyamping yang berisi nomor telepon perusahaan, tulisan

“Contact Us”, tulisan “Member Login”, *button* “Member – Get Listed Today” dan sebuah *search bar*. Hal-hal tersebut dibuat dengan menggunakan fungsi <ul> dan <li>, kecuali *search bar*. Untuk tulisan “Contact Us”, “Member Login”, dan *button* “Member – Get Listed Today” dibuat dengan fungsi <a> karena hal tersebut merupakan *link* yang akan membawa ke halaman lain. Dalam hal ini, *button* diberikan *background* di .css. Nomor telepon perusahaan dalam hal ini tidak menggunakan fungsi <a>, tetapi menggunakan <span> dikarenakan nomor telepon perusahaan bukan sebuah *link*. Kemudian, segala sesuatunya, seperti ukuran dan posisi diatur di dalam .css. Selanjutnya, bagian *search bar* lagi-lagi bukan dibuat oleh peserta magang dikarenakan hal tersebut terhubung ke *database* perusahaan sehingga pembuatan *search bar* diambil alih oleh karyawan IT yang lebih ahli. Selanjutnya, terdapat bagian *navigation bar* juga yang dibuat dengan fungsi <ul> dan <li> juga yang di dalamnya terdapat sebuah fungsi <a> karena *navigation bar* tersebut merupakan sebuah *link* yang mengarahkan ke halaman-halaman sesuai dengan tulisan di *navigation bar* tersebut. Lagi-lagi, halaman yang dihubungkan adalah halaman yang lama karena halaman yang baru belum di-*deploy*. Kemudian, segala sesuatunya, seperti ukuran dan posisi diatur di dalam .css sampai sedemikian rupa. *Navigation bar* juga dalam hal ini tidak *stick* di bagian atas saja melainkan mengikuti *scroll* dari *user*. Terakhir, *banner slider* dalam hal ini juga bukan tanggung jawab dari peserta magang sehingga bagian *banner slider* dibuat oleh karyawan lain sehingga peserta magang tinggal mengubah *width* di .css untuk menyesuaikan ukuran *width banner*.



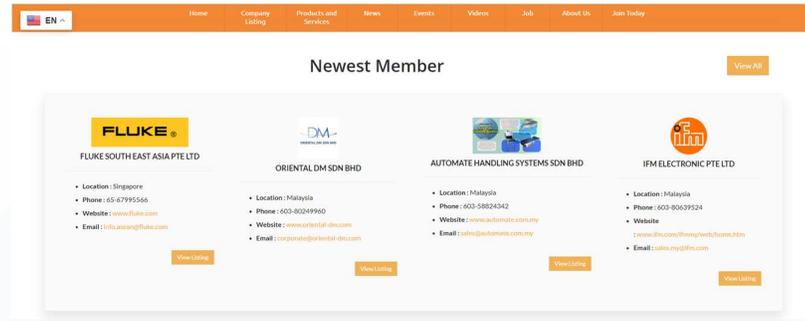
Gambar 3.53 Bagian *Search Bar* dan *About Us* Halaman *Website* Baru



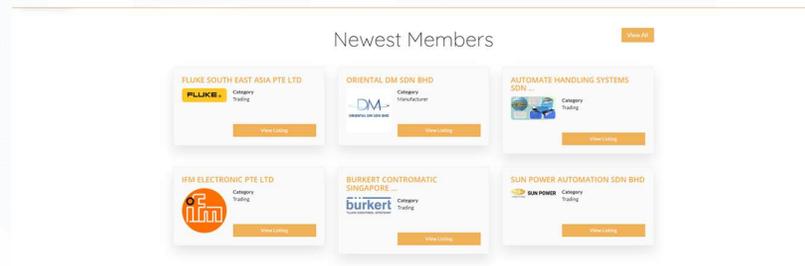
Gambar 3.54 Bagian *About Us* Halaman *Website* Lama

Gambar 3.53 merupakan gambar yang menunjukkan bagian *Homepage* baru yang berada di bawah *banner slider*. Berdasarkan gambar tersebut dapat dilihat bahwa setelah *banner slider*, bagian berikutnya adalah bagian *search bar* dan *About Us*. *Search bar* tersebut dipindahkan ke bagian bawah dan rata tengah. Tulisan “Start Searching Here” dibuat dengan menggunakan fungsi `<h2>` dan `.css` untuk menyesuaikan ukuran dan posisi sedemikian rupa. Sisanya dalam bagian *search bar* ini diambil alih oleh karyawan IT lain karena harus menyambungkan koneksi ke dalam *database* perusahaan.

Gambar 3.54 merupakan sebuah gambar yang menunjukkan halaman *Homepage* lama bagian *About Us*. Masih berhubungan dengan gambar 3.53, gambar 3.54 di atas merupakan gambar bagian *About Us* untuk halaman *website* yang lama. Berdasarkan kedua gambar tersebut, dapat dilihat bahwa posisi dari bagian *About Us* berpindah dari yang sebelumnya di bagian bawah halaman *Homepage* menjadi di bagian cenderung atas *Homepage*. Selain letaknya yang berubah, desain dari *About Us* juga berubah. Hal-hal yang berubah adalah bagian *Local Businesses* dihilangkan karena sudah tidak dibutuhkan lagi. Kemudian, diberikan sebuah gambar di sebelah kiri. Selanjutnya, judul yang bertuliskan “About Us” diubah menjadi rata kiri. Berikutnya, paragraf *About Us* dipisah menjadi rata kiri untuk paragraf pertama dan rata kanan untuk paragraf kedua. Terakhir, *link* yang bertuliskan “Get listed today >>” diubah posisinya saja. Cara membuatnya adalah dengan menggunakan fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan di dalamnya lagi terdapat fungsi `<td>`. Fungsi `<td>` dalam hal ini ada dua, satu untuk gambar sendiri dan satu untuk paragraf *About Us*. Kemudian, untuk gambar digunakan fungsi `<img>` dan untuk paragraf digunakan fungsi `<h2>` untuk judul “About Us”, `<p>` untuk paragraf di mana fungsi `<p>` tersebut dibedakan *class*-nya antara kanan, kiri, dan supaya bisa dibuat *alignment* seperti gambar tersebut, serta fungsi `<a>` untuk *link* yang ada di dalam tulisan “Get listed today >>”. Semuanya dibungkus dalam satu *container* yang di *.css*-nya nanti akan diberikan sebuah kodingan tambahan sehingga dapat membentuk sebuah *shadow* yang membedakan antara bagian *About Us* dan *background* seperti gambar 3.53.



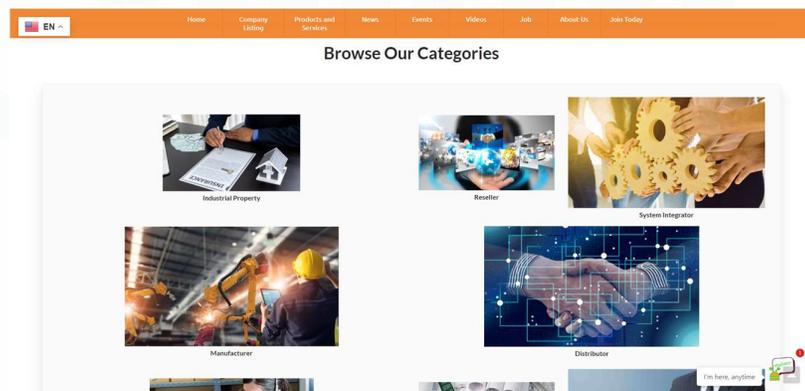
Gambar 3.55 Bagian *Newest Member* Halaman *Website* baru.



Gambar 3.56 Bagian *Newest Member* Halaman *Website* Lama

Gambar 3.55 merupakan bagian *Newest Member* yang baru, sedangkan 3.56 merupakan bagian *Newest Member* yang lama. Secara singkat, bagian *Newest Member* dalam hal ini diperlukan untuk menunjukkan vendor terbatu dari *website e-commerce* ini. Pertama-tama yang berubah adalah letaknya. Dalam halaman *website* yang lama, *Newest Member* berada di bagian bawah *banner slider*, tetapi di bagian baru, letaknya berpindah menjadi di bawah bagian *About Us*. Selain itu, hal lain yang berubah adalah tulisan judul menjadi cetak tebal (*bold*) dan juga desain dari bagian tersebut secara keseluruhan. Dalam bagian *Newest Member* di halaman yang lama, terdapat enam vendor, sedangkan yang baru maksimal hanya empat. Hal ini dikarenakan berbagai alasan, salah satunya adalah enam vendor terlalu banyak. Cara membuatnya adalah dengan menggunakan fungsi `<h1>` untuk judul yang bertuliskan “Newest Member” dan `.css` untuk mengatur tata letak dan ukuran *text*.

Kemudian, terdapat sebuah *button* bertuliskan “View All” yang dibuat dengan fungsi `<a>` yang diberikan *link* ke halaman lain dan juga diberikan *background* berwarna *orange*, serta menggunakan *.css* untuk mengatur tata letak, warna, dan ukuran *text*. Selanjutnya, sama seperti membuat bagian *About Us*, digunakanlah fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>`. Fungsi `<td>` dalam hal ini terdapat empat bagian guna menampilkan empat buah vendor baru. Dalam fungsi `<td>` tersebut terdapat fungsi `<img>` untuk menampilkan logo vendor, `<p>` untuk menampilkan judul nama vendor, `<hr>` untuk memberikan garis, `<ul>` serta `<li>` untuk menampilkan poin-poin informasi vendor, dan `<a>` untuk membuat *button* “View Listing” yang akan dimasukan *link* ke halaman vendor tersebut. Sama seperti bagian *About Us*, tabel ini dibungkus oleh sebuah *container* yang di *.css* diberikan kodingan khusus agar memiliki *shadow* yang membedakan antara tabel konten *Newest Member* dan *background*. Hal yang sedang diselidiki oleh divisi *web developer* dan juga divisi IT lainnya adalah bagaimana cara membuat hal ini otomatis sehingga apabila terdapat vendor baru, namanya akan langsung muncul di bagian tersebut beserta informasi-informasinya karena untuk sekarang kelemahannya adalah untuk mengubahnya harus manual.



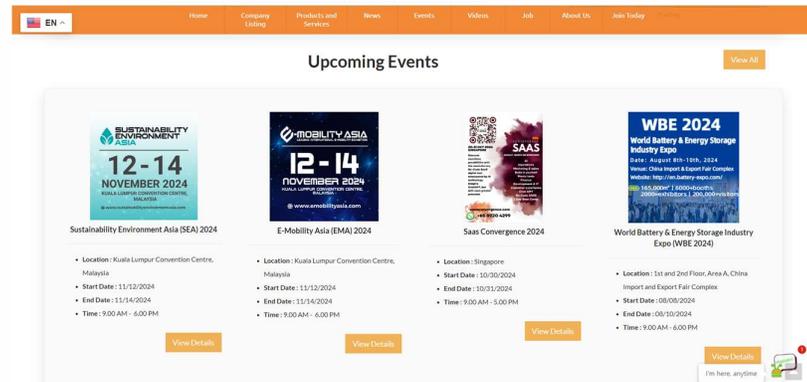
Gambar 3.57 Bagian *Categories* Halaman *Website* Baru



Gambar 3.58 Bagian *Categories* Halaman *Website* Lama

Gambar 3.57 merupakan sebuah gambar yang menunjukkan bagian yang menampilkan *categories* untuk setiap vendor di halaman *website* yang baru, sedangkan gambar 3.58 menunjukkan bagian *categories* untuk setiap vendor di halaman *website* yang lama. Hal pertama yang berubah adalah letaknya. Bagian *categories* seharusnya terletak di bagian cenderung agak bawah di halaman *website* yang lama, sedangkan di halaman *website* yang baru letaknya berada setelah *Newest Member*. Hal lain yang berubah adalah judulnya berubah dari “Search Populan Categories” menjadi “Browse Our Categories” dan dicetak tebal. Kemudian, desain secara keseluruhan juga berbeda. Desain yang lama hanya berbentuk kotak-kotak yang berjejer empat perbaris yang berisi gambar *categories*, sedangkan yang baru memiliki desain yang lebih unik, yakni gambar besar, kecil, dan tidak beraturan. Besar atau kecilnya sebuah gambar ditentukan oleh berapa banyak vendor yang ada di kategori tersebut. Semakin banyak jumlah vendor di kategori tersebut, semakin besar gambarnya dan demikian sebaliknya. Cara membuatnya adalah dengan menggunakan fungsi `<span>` untuk judul dan melakukan kostumisasi di bagian `.css` untuk ukuran dan letak. Kemudian menggunakan fungsi `<table>` yang di dalamnya terdapat `<tr>` dan `<td>`. Dalam fungsi `<td>`, terdapat fungsi `<a>` yang berfungsi menaruh *link* ke halaman *categories* tersebut dan di dalam `<a>` terdapat sebuah fungsi `<img>` untuk menampilkan gambar. Jadi, gambar apabila diklik akan menuju ke halaman *categories* yang

dituju. Kemudian di dalam `<td>` juga terdapat fungsi `<span>` yang di dalamnya terdapat fungsi `<strong>` untuk menampilkan tulisan nama kategori tersebut. Kemudian, dikostumisasi di bagian `.css`-nya untuk memperbesar dan memperkecil gambar tersebut. Kemudian, semuanya dibungkus ke dalam satu `<div class>` untuk nanti diberikan sebuah kodingan tambahan untuk memberikan *shadow* supaya beda antara *background* dan tabel *categories*. Kedepannya, karyawan IT dan *supervisor* peserta magang ingin mencari tahu bagaimana cara agar bagian tersebut terhubung ke *database* sehingga besar kecilnya gambar akan otomatis sesuai dengan banyaknya jumlah vendor di kategori tersebut. Selain itu, perusahaan ingin juga mencari tahu bagaimana menghubungkannya ke *database* sehingga apabila terdapat kategori baru, bisa langsung muncul di bagian tersebut. Hal ini dikarenakan apabila terdapat kategori baru, karyawan IT harus menambahkannya secara manual di *front-end*.



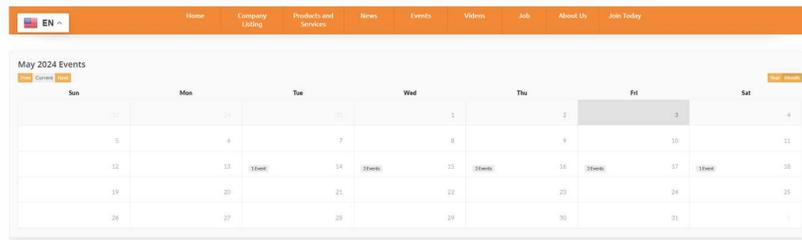
Gambar 3.59 Bagian *Upcoming Events* Halaman *Website* Baru



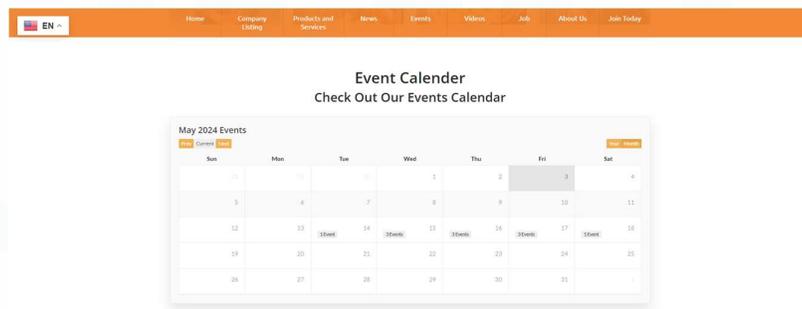
Gambar 3.60 Bagian *Upcoming Events* Halaman *Website* Lama

Gambar 3.59 merupakan gambar yang menunjukkan *Upcoming Events* untuk halaman *website* yang baru, sedangkan gambar 3.60 merupakan gambar yang menunjukkan *Upcoming Events* untuk halaman *website* yang lama. Pertama-tama yang berubah adalah letaknya. Dalam halaman *website* yang lama, *Upcoming Events* berada di bagian tengah-tengah *website* setelah bagian yang berjudul *Latest Article*, tetapi di bagian baru, letaknya berpindah menjadi setelah bagian *Browse Our Categories*. Selain itu, hal lain yang berubah adalah tulisan judul menjadi cetak tebal (*bold*) dan juga desain dari bagian tersebut secara keseluruhan. Dalam bagian *Upcoming Events* di halaman yang lama, terdapat sebuah *carousel* yang menampilkan *event-event* yang akan datang dan ditampilkan tiga per *slide*, sedangkan yang baru maksimal hanya empat *event* saja dan bukan *carousel*, melainkan sama seperti bagian *Newest Member* sebelumnya, yaitu menggunakan `<table>`. Cara membuatnya adalah dengan menggunakan fungsi `<h1>` untuk judul yang bertuliskan “Upcoming Events” dan `.css` untuk mengatur tata letak dan ukuran *text*. Kemudian, terdapat sebuah *button* bertuliskan “View All” yang dibuat dengan fungsi `<a>` yang diberikan *link* ke halaman lain dan juga diberikan *background* berwarna *orange*, serta menggunakan `.css` untuk mengatur tata letak, warna, dan ukuran *text*. Selanjutnya, sama seperti membuat bagian *Newest Member*, digunakanlah fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>`. Fungsi `<td>` dalam hal ini terdapat empat bagian guna menampilkan empat buah *event* baru. Dalam fungsi `<td>` tersebut terdapat fungsi `<img>` untuk menampilkan poster *event*, `<p>` untuk menampilkan judul *event*, `<hr>` untuk memberikan garis, `<ul>` serta `<li>` untuk menampilkan poin-poin informasi *event*, dan `<a>` untuk membuat *button* “View Details” yang akan dimasukan *link* ke halaman *event* tersebut. Sama seperti

bagian *Newest Member*, tabel ini dibungkus oleh sebuah *container* yang di *.css* diberikan kodingan khusus agar memiliki *shadow* yang membedakan antara tabel konten *Upcoming Events* dan *background*. Hal yang sedang diselidiki oleh divisi *web developer* dan juga divisi IT lainnya adalah bagaimana cara membuat hal ini otomatis sehingga apabila terdapat *event* baru yang ditambahkan di *back-end*, nama *event*, gambar poster, detail *event*, dan bahkan *button* “View Detail” juga berubah secara otomatis *link*-nya karena untuk sekarang kelemahannya adalah untuk mengubahnya harus manual. Para karyawan dan juga *supervisor* peserta magang juga ingin mencari cara juga untuk membuat *carousel* dengan *desain* yang seperti sekarang ini (halaman *website* baru).



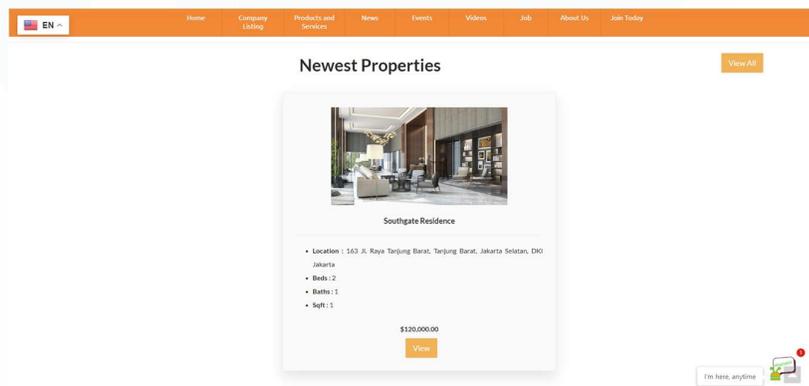
Gambar 3.61 Bagian *Event Calendar* Halaman *Website* Baru



Gambar 3.62 Bagian *Event Calendar* Halaman *Website* Lama

Gambar 3.61 merupakan gambar yang menunjukkan bagian *Event Calendar* di halaman *website* yang lama, sedangkan gambar 3.62 menunjukkan bagian *Event Calendar* di halaman *website* yang

lama. Hal yang membedakan dalam bagian ini adalah letaknya. Bagian *Event Calendar website* lama berada di bagian bawah tepat di atas bagian *About Us website* lama, sedangkan halaman yang baru bagian ini digabung dengan bagian *Upcoming Events* yang telah dijelaskan sebelumnya. Hal tersebut juga menjelaskan mengapa tidak ada judul di bagian yang baru. Hal lain tidak banyak berubah dan jujur bagian ini bukan merupakan tanggung jawab peserta magang. Hal ini dikarenakan bagian ini harus meng-embed sebuah code supaya bisa menampilkan event-event yang perusahaan sudah buat dalam Google Calendar ke dalam website ini. Hal inipun diambil alih oleh karyawan IT yang lebih professional.



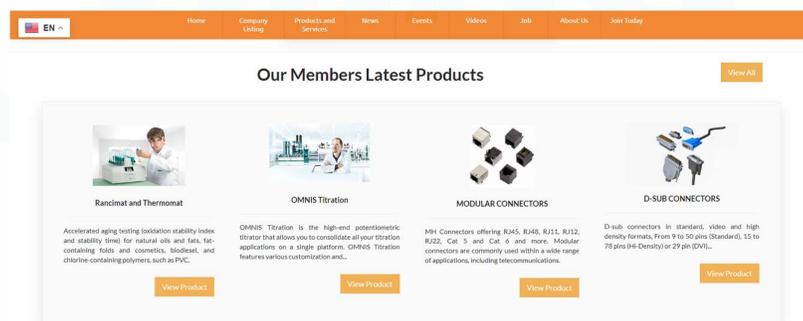
Gambar 3.63 Bagian *Newest Properties* Halaman *Website* Baru



Gambar 3.64 Bagian *Newest Properties* Halaman *Website* Lama

Gambar 3.63 merupakan sebuah gambar yang menunjukkan bagian *Newest Properties* di halaman website yang baru, sedangkan gambar 3.64 merupakan gambar yang menunjukkan bagian *Newest Properties* di halaman website yang lama. Hal yang membedakan

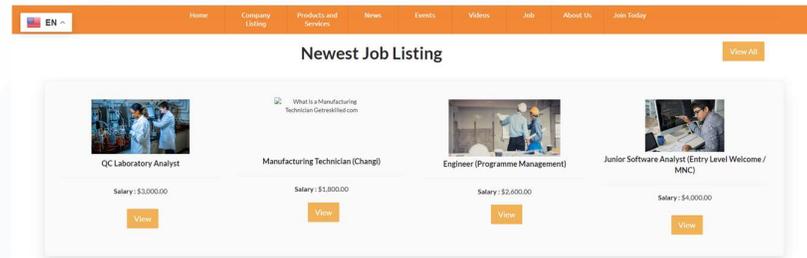
adalah desain secara keseluruhan. Mulai dari judul yang dicetak tebal hingga desain *list property*-nya yang berubah cukup drastis. Cara membuatnya adalah dengan menggunakan fungsi `<h1>` untuk judul yang bertuliskan “Newest Properties” dan `.css` untuk mengatur tata letak dan ukuran *text*. Kemudian, terdapat sebuah *button* bertuliskan “View All” yang dibuat dengan fungsi `<a>` yang diberikan *link* ke halaman lain dan juga diberikan *background* berwarna *orange*, serta menggunakan `.css` untuk mengatur tata letak, warna, dan ukuran *text*. Selanjutnya, digunakanlah fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>`. Dalam fungsi `<td>` tersebut terdapat fungsi `<img>` untuk menampilkan gambar *property*, `<p>` untuk menampilkan nama *property*, `<hr>` untuk memberikan garis, `<ul>` serta `<li>` untuk menampilkan poin-poin informasi *property*, `<span>` untuk menampilkan tulisan harga *property*, dan `<a>` untuk membuat *button* “View” yang akan dimasukan *link* ke halaman detail *property* tersebut. Tabel ini juga dibungkus oleh sebuah *container* yang di `.css` diberikan kodingan khusus agar memiliki *shadow* yang membedakan antara tabel konten *Newest Properties* dan *background*. Bagian ini ada kemungkinan besar akan dihapus nantinya mengingat perusahaan sudah tidak lagi aktif dalam bisnis *property*.



Gambar 3.65 Bagian *Our Members Latest Products* Halaman *Website Baru*

Gambar 3.65 merupakan sebuah gambar yang menunjukkan bagian *Our Members Latest Products*. Bagian ini tidak ada refrensi ke halaman *website* lama karena bagian ini merupakan bagian yang baru yang ditambahkan berdasarkan inisiatif peserta magang. Cara membuatnya adalah dengan menggunakan fungsi `<h1>` untuk judul yang bertuliskan “Our Members Latest Products” dan `.css` untuk mengatur tata letak dan ukuran *text*. Kemudian, terdapat sebuah *button* bertuliskan “View All” yang dibuat dengan fungsi `<a>` yang diberikan *link* ke halaman lain dan juga diberikan *background* berwarna *orange*, serta menggunakan `.css` untuk mengatur tata letak, warna, dan ukuran *text*. Selanjutnya, digunakanlah fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>`. Fungsi `<td>` dalam hal ini terdapat empat bagian guna menampilkan empat buah produk baru dari setiap vendor. Dalam fungsi `<td>` tersebut terdapat fungsi `<img>` untuk menampilkan gambar ataupun ilustrasi gambar dari produk tersebut, `<p>` untuk menampilkan nama produk, `<hr>` untuk memberikan garis, `<span>` untuk menampilkan paragraf singkat deskripsi produk, dan `<a>` untuk membuat *button* “View Products” yang akan dimasukan *link* ke halaman produk tersebut. Tabel ini dibungkus oleh sebuah *container* yang di `.css` diberikan kodingan khusus agar memiliki *shadow* yang membedakan antara tabel konten *Our Members Latest Products* dan *background*. Hal yang sedang diselidiki oleh divisi *web developer* dan juga divisi IT lainnya adalah bagaimana cara membuat hal ini otomatis sehingga apabila terdapat produk baru yang ditambahkan di *back-end*, nama produk, gambar produk, paragraf detail produk, dan bahkan *button* “View Product” juga berubah secara otomatis *link*-nya karena untuk sekarang kelemahannya adalah untuk mengubahnya harus manual. Para karyawan dan juga *supervisor* peserta magang juga ingin mencari

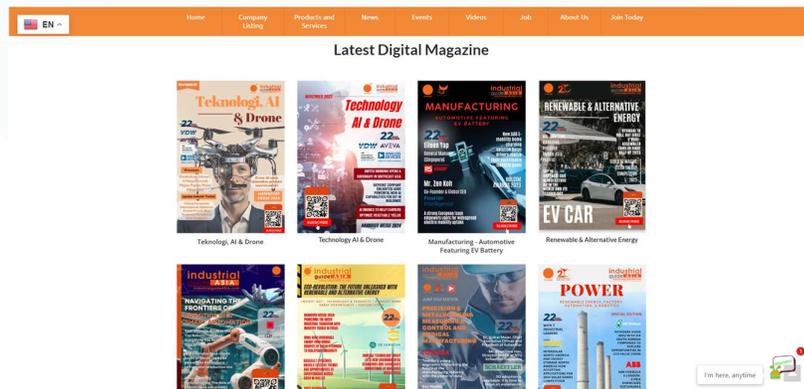
cara juga untuk membuat *carousel* dengan *desain* yang seperti sekarang ini (halaman *website* baru).



Gambar 3.66 Bagian *Newest Job Listing* Halaman *Website* Baru

Gambar 3.66 merupakan sebuah gambar yang menunjukkan bagian *Newest Job Listing*. Bagian ini tidak ada refrensi ke halaman *website* lama karena bagian ini merupakan bagian yang baru yang ditambahkan berdasarkan inisiatif peserta magang. Cara membuatnya adalah dengan menggunakan fungsi `<h1>` untuk judul yang bertuliskan “Newest Job Listing” dan `.css` untuk mengatur tata letak dan ukuran *text*. Kemudian, terdapat sebuah *button* bertuliskan “View All” yang dibuat dengan fungsi `<a>` yang diberikan *link* ke halaman lain dan juga diberikan *background* berwarna *orange*, serta menggunakan `.css` untuk mengatur tata letak, warna, dan ukuran *text*. Selanjutnya, digunakanlah fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>`. Fungsi `<td>` dalam hal ini terdapat empat bagian guna menampilkan empat buah pekerjaan baru. Dalam fungsi `<td>` tersebut terdapat fungsi `<img>` untuk menampilkan gambar ataupun ilustrasi gambar dari produk tersebut, `<p>` untuk menampilkan nama pekerjaan, `<hr>` untuk memberikan garis, `<span>` untuk menampilkan gaji pekerjaan, dan `<a>` untuk membuat *button* “View” yang akan dimasukan *link* ke halaman detail pekerjaan tersebut. Tabel ini dibungkus oleh sebuah *container* yang di `.css` diberikan kodingan khusus agar memiliki *shadow* yang membedakan antara tabel konten *Newest Job Listing* dan

*background*. Hal yang sedang diselidiki oleh divisi *web developer* dan juga divisi IT lainnya adalah bagaimana cara membuat hal ini otomatis sehingga apabila terdapat pekerjaan baru yang ditambahkan di *back-end*, nama pekerjaan, gambar pekerjaan, detail gaji, dan bahkan *button* “View” juga berubah secara otomatis *link*-nya karena untuk sekarang kelemahannya adalah untuk mengubahnya harus manual. Para karyawan dan juga *supervisor* peserta magang juga ingin mencari cara juga untuk membuat *caraousel* dengan *desain* yang seperti sekarang ini (halaman *website* baru).

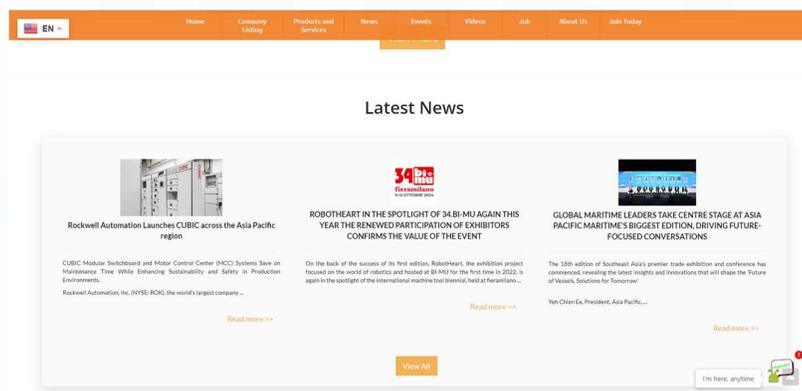


Gambar 3.67 Bagian *Latest Digital Magazine* Halaman *Website* Baru

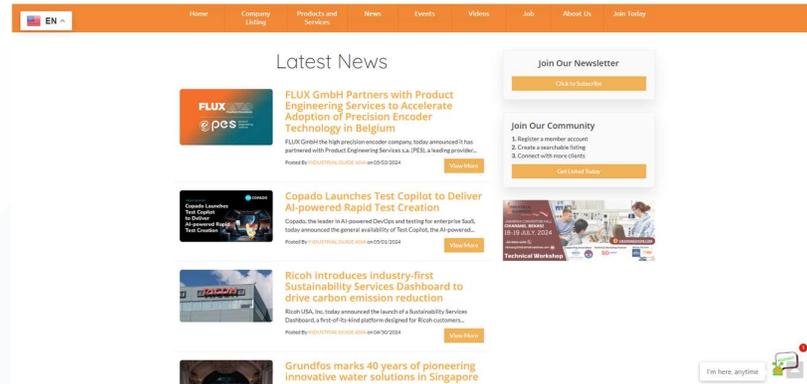


Gambar 3.68 Bagian *Latest Digital Magazine* Halaman *Website* Lama

Gambar 3.67 merupakan sebuah gambar yang menunjukkan bagian *Latest Digital Magazine* di halaman *website* yang baru, sedangkan gambar 3.68 merupakan gambar yang menunjukkan bagian *Latest Digital Magazine* di halaman *website* yang lama. Dalam hal ini, tidak ada perbedaan yang signifikan untuk bagian ini hanya letaknya saja. Bagian *Latest Digital Magazine* di halaman *website* yang lama berada di bawah bagian *Newest Properties*, sedangkan di halaman *website* yang baru berada di bawah *Newest Job Listing*.



Gambar 3.69 Bagian *Latest News* Halaman *Website* Baru

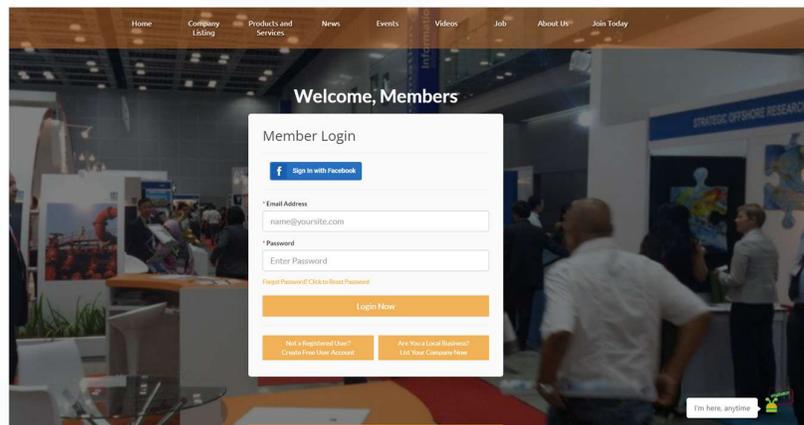


Gambar 3.70 Bagian *Latest News* Halaman *Website Lama*

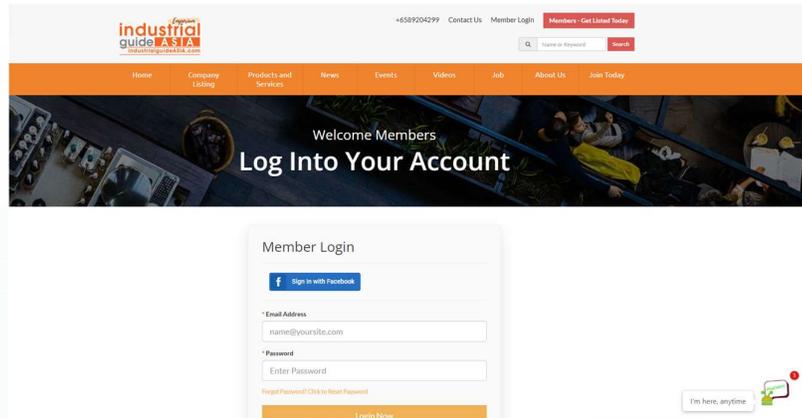
Gambar 3.69 merupakan sebuah gambar yang menunjukkan bagian *Latest News* di halaman *website* yang baru, sedangkan gambar 3.70 merupakan gambar yang menunjukkan bagian *Latest News* di halaman *website* yang lama. Bagian ini memiliki perbedaan yang cukup signifikan. Mulai dari letaknya yang seharusnya berada di bawah *Newest Member* berubah menjadi di bagian paling bawah halaman *website* hingga desain secara keseluruhan. Berdasarkan kedua gambar tersebut hal yang membedakan adalah *list* berita yang bukan lagi ke bawah, melainkan ke samping, warna judul berita yang berubah dari *orange* menjadi hitam, judul yang bertuliskan “Latest News” dicetak tebal, dan juga bagian iklan di samping kanan dihilangkan secara total. Cara membuatnya adalah dengan menggunakan fungsi `<h1>` untuk judul yang bertuliskan “Latest News” dan `.css` untuk mengatur tata letak dan ukuran *text*. Selanjutnya, digunakanlah fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>`. Fungsi `<td>` dalam hal ini terdapat tiga bagian guna menampilkan tiga buah berita baru. Dalam fungsi `<td>` tersebut terdapat fungsi `<img>` untuk menampilkan *thumbnail* berita, `<p>` untuk menampilkan judul berita serta paragraf kecil untuk mendeskripsikan berita, `<hr>` untuk memberikan garis, dan `<a>` untuk membuat tulisan *link* “Read more >>” yang akan dimasukkan

*link* ke halaman detail berita tersebut. Tabel ini dibungkus oleh sebuah *container* yang di .css diberikan kodingan khusus agar memiliki *shadow* yang membedakan antara tabel konten *Latest News* dan *background*. Kemudian, terdapat sebuah *button* bertuliskan “View All” yang dibuat dengan fungsi <a> yang diberikan *link* ke halaman lain dan juga diberikan *background* berwarna *orange*, serta menggunakan .css untuk mengatur tata letak, warna, dan ukuran *text*. Hal yang sedang diselidiki oleh divisi *web developer* dan juga divisi IT lainnya adalah bagaimana cara membuat hal ini otomatis sehingga apabila terdapat berita baru yang ditambahkan di *back-end*, judul berita, *thumbnail* berita, paragraf singkat detail berita, dan bahkan tulisan *link* “Read more >>” juga berubah secara otomatis *link*-nya karena untuk sekarang kelemahannya adalah untuk mengubahnya harus manual.

### 3.3.6.2 Login Page

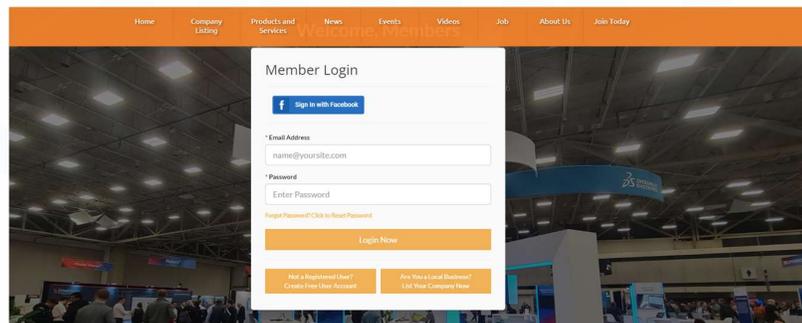


Gambar 3.71 Halaman *Login* Baru



Gambar 3.72 Halaman *Login Lama*

Gambar 3.71 merupakan gambar halaman *login page* yang baru, sedangkan gambar 3.72 merupakan gambar halaman *login page* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Selanjutnya, gambar *widescreen* di bawah *header* (lihat gambar 3.72) juga dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Welcome, Members” yang dibuat dengan fungsi `<span>`. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Berikut merupakan contohnya.



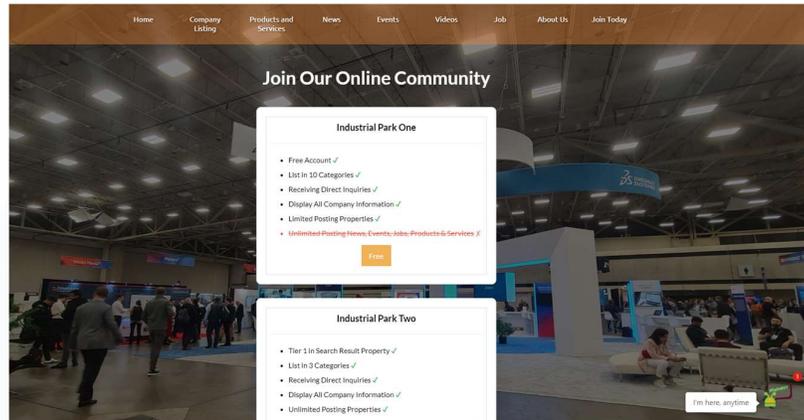
Gambar 3.73 *Navigation Bar* Berubah *Opacity*

Gambar 3.73 merupakan sebuah gambar yang menunjukkan *navigation bar* yang berubah *opacity*-nya jika *user* melakukan *scrolling*. Berdasarkan gambar tersebut, dapat dilihat perbedaan *opacity navigation bar* berdasarkan gambar 3.71 dan 3.73. Hal ini dibuat dengan menggunakan seperangkat kodingan JavaScript untuk membuat animasi tersebut.

Hal selanjutnya yang berbeda adalah *background* dari halaman itu sendiri. Jika kita lihat di gambar 3.72, *background* dari halaman *website* ini berwarna putih polos, sedangkan jika dilihat di gambar 3.71, *background* dari *website* ini merupakan sebuah gambar. Gambar tersebut merupakan sebuah *image slideshow* yang dibuat dengan seperangkat kodingan JavaScript sehingga gambar tersebut akan berubah-ubah per lima detik. Buktinya adalah gambar 3.71 dan gambar 3.73 memiliki *image background* yang berbeda. Berbicara mengenai *background* juga, kecerahan dari gambar-gambar yang ditampilkan di *background* juga sedikit diredupkan. Caranya adalah dengan membuat sebuah *elements* berwarna hitam yang kemudian diletakkan di paling belakang semua *elements* dengan fungsi *z-index* dan kemudian diatur *opacity* gambar dan *element* hitam tersebut.

Hal-hal seperti form *login* tersebut tidaklah diubah dari halaman yang lama sehingga bentuknya masih sama. Hal ini dikarenakan terdapat sebuah koneksi antara *element* tersebut dan *database* sehingga tidak diubah karena takut bermasalah.

### 3.3.6.3 Join Community Page



Gambar 3.74 Halaman *Join Community* Baru

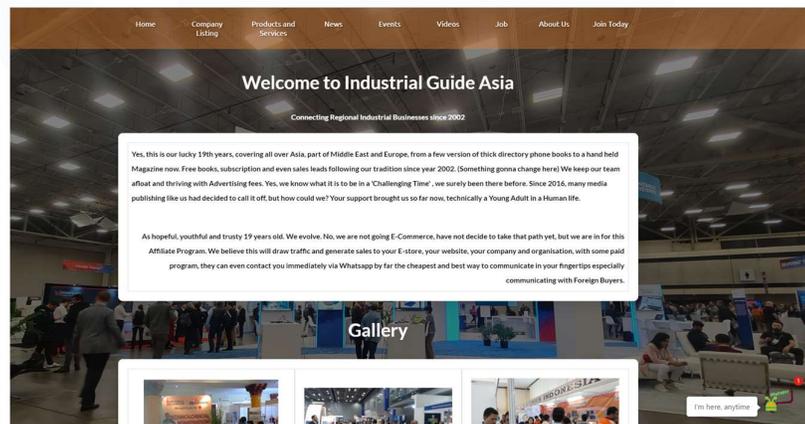


Gambar 3.75 Halaman *Join Community* Lama

Gambar 3.74 merupakan gambar halaman *Join Community* yang baru, sedangkan gambar 3.75 merupakan gambar halaman *Join Community* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Selanjutnya, gambar *widescreen* di bawah *header* (lihat gambar 3.75) juga dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Join Our Online Community” yang dibuat dengan fungsi `<span>`. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman *login*

yang telah dijelaskan di bagian sebelumnya. Hal yang menjadi perbedaan utamanya adalah terdapat sebuah *table* yang berisikan informasi mengenai masing-masing *membership*. Hal ini dibuat dengan menggunakan fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>` yang di dalamnya terdapat fungsi `<span>` untuk judul, `<ul>` serta `<li>` untuk poin-poin *benefit membership*, dan fungsi `<a>` untuk *button* di setiap *table*. Jika terdapat tulisan yang dicoret, dapat menggunakan fungsi `<s>`. Kemudian, fungsi `<table>` dalam halaman ini diberikan *background* berwarna putih dan garis *border* warna abu-abu.

#### 3.3.6.4 About Us



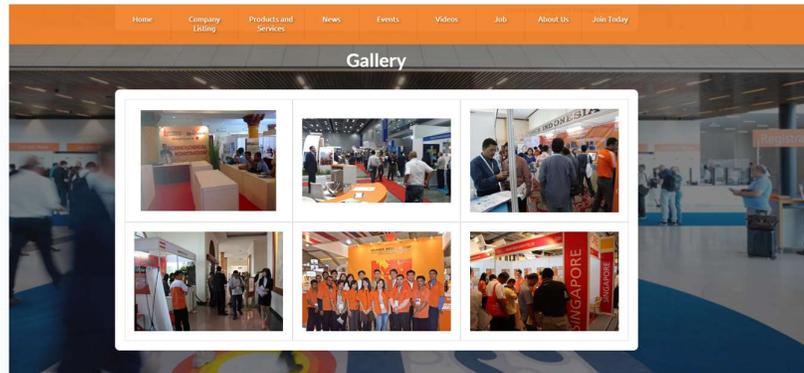
Gambar 3.76 Halaman *About Us* Baru



Gambar 3.77 Halaman *About Us* Lama

Gambar 3.76 merupakan gambar halaman *About Us* yang baru, sedangkan gambar 3.77 merupakan gambar halaman *About Us* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Selanjutnya, gambar *widescreen* di bawah *header* (lihat gambar 3.77) juga dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Welcome to Industrial Guide Asia” yang dibuat dengan fungsi `<span>` dan *text* “Connecting Regional Industrial Businesses Since 2022” yang juga dibuat dengan fungsi `<span>` hanya saja ukurannya lebih kecil. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman-halaman sebelumnya. Perbedaan utamanya adalah terdapat sebuah *element* yang dibuat dengan menggunakan fungsi `<table>` yang berisikan paragraf mengenai Industrial Guide Asia. Paragraf tersebut juga dibuat menjadi dua paragraf terpisah. Paragraf pertama menggunakan *allignment* kiri, sedangkan paragraf kedua menggunakan *allignment* kanan. Hal tersebut dibuat dengan cara membedakan `<span class>` sehingga di bagian *.css* dapat diatur sedemikian rupa. Kemudian, foto di sebelah kanan halaman lama,

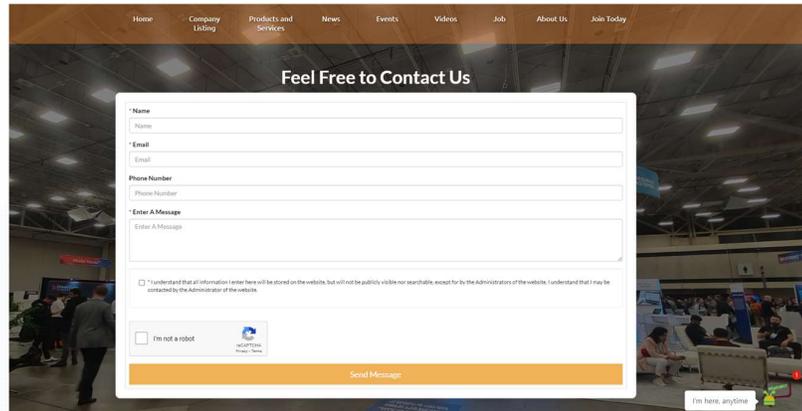
dipindahkan ke bagian *Gallery* yang terletak di bagian bawah setelah paragraf *About Us*.



Gambar 3.78 *Gallery* di Halaman *About Us* Baru

Gambar 3.78 merupakan gambar yang menunjukkan *element* baru di halaman *About Us*, yaitu *Gallery*. *Gallery* berfungsi untuk menunjukkan foto-foto *event* perusahaan. Hal ini dibuat dengan menggunakan fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>` yang di dalam `<td>` terdapat fungsi `<img>` untuk meletakkan gambar tersebut. Kemudian, fungsi `<table>` dalam halaman ini diberikan *background* berwarna putih dan garis *border* warna abu-abu.

### 3.3.6.5 *Contact Us*



Gambar 3.79 Halaman *Contact Us* Baru

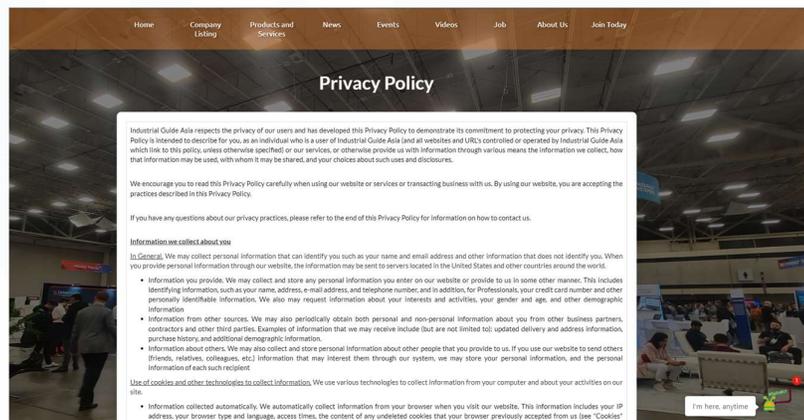


Gambar 3.80 Halaman *Contact Us* Lama

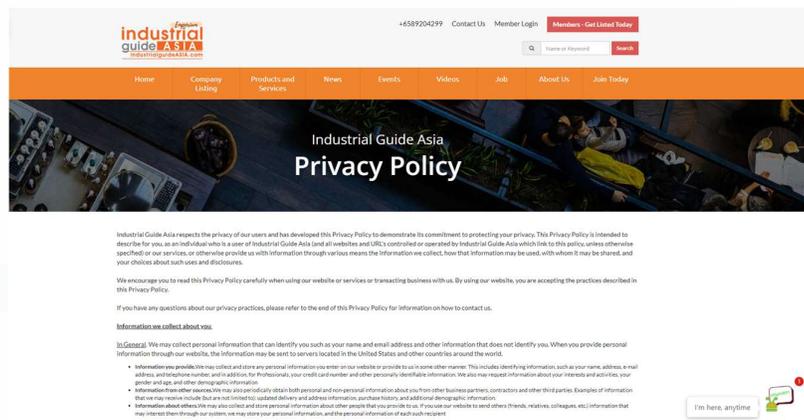
Gambar 3.79 merupakan gambar halaman *Contact Us* yang baru, sedangkan gambar 3.80 merupakan gambar halaman *Contact Us* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Selanjutnya, gambar *widescreen* di bawah *header* (lihat gambar 3.80) juga dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Feel Free to Contact Us” yang dibuat dengan fungsi `<span>`. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman-halaman

sebelumnya. Perbedaan utamanya adalah *form* untuk *contact us* diperbesar dan diberikan *background* warna putih. Hal ini lagi-lagi memerlukan sebuah koneksi *database* sehingga bagian ini diambil alih oleh karyawan IT yang lebih profesional dan memiliki akses ke *database* perusahaan. Perbedaan lain adalah bagian “How Can We Help You” sepenuhnya dihilangkan.

### 3.3.6.6 Privacy Policy



Gambar 3.81 Halaman *Privacy Policy* Baru

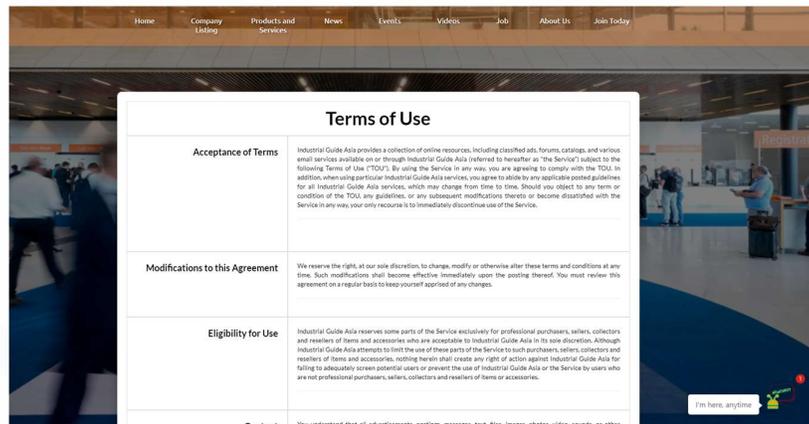


Gambar 3.82 Halaman *Privacy Policy* Lama

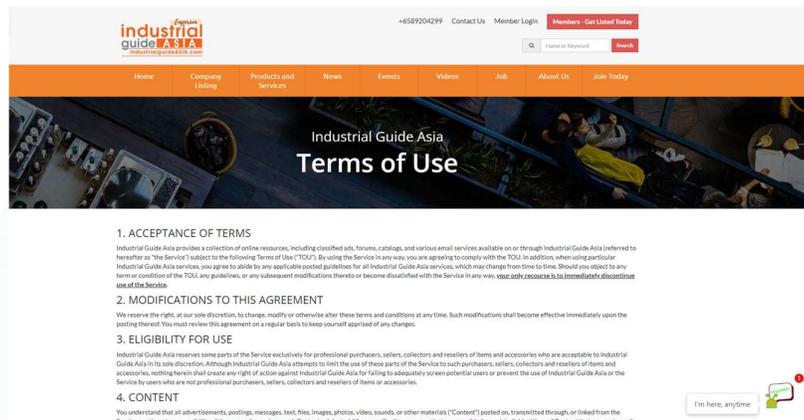
Gambar 3.81 merupakan gambar halaman *Privacy Policy* yang baru, sedangkan gambar 3.82 merupakan gambar halaman

*Privacy Policy* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Selanjutnya, gambar *widescreen* di bawah *header* (lihat gambar 3.82) juga dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Privacy Policy” yang dibuat dengan fungsi `<span>`. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman-halaman sebelumnya. Perbedaan utamanya adalah paragraf *privacy policy* dibuat di dalam sebuah tabel dengan menggunakan fungsi `<table>` yang di dalamnya terdapat fungsi `<tr>` dan `<td>` yang di dalam `<td>` terdapat fungsi `<p>` untuk paragraf, `<br>` untuk melakukan *enter*, `<ul>` serta `<li>` untuk bagian poin-poin. Kemudian, fungsi `<table>` dalam halaman ini diberikan *background* berwarna putih dan garis *border* warna abu-abu.

### 3.3.6.7 Terms of Use



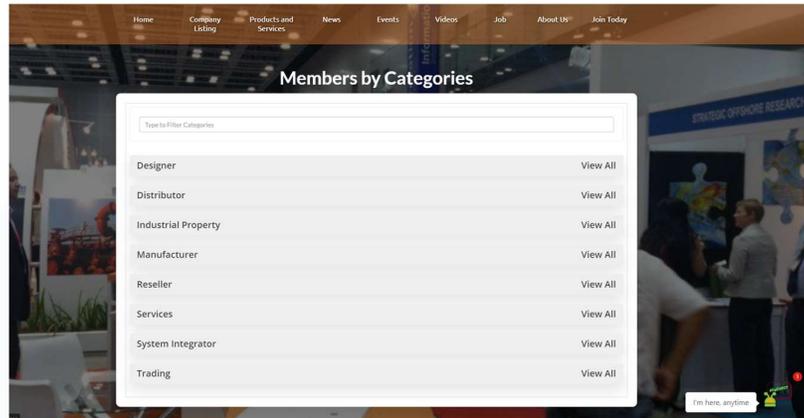
Gambar 3.83 Halaman *Terms of Use* Baru



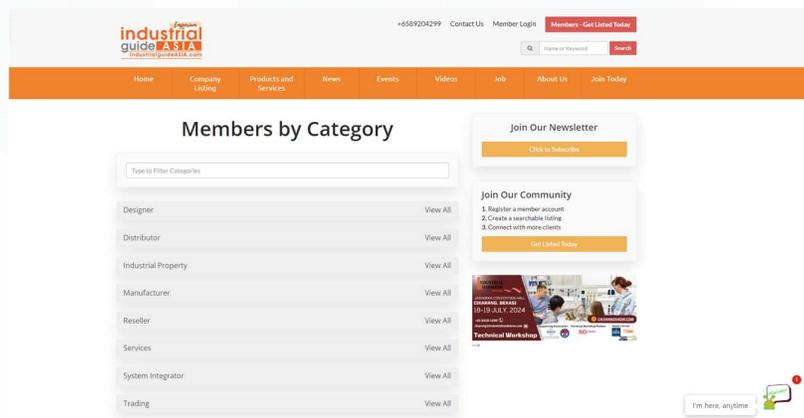
Gambar 3.84 Halaman *Terms of Use* Lama

Gambar 3.83 merupakan gambar halaman *Terms of Use* yang baru, sedangkan gambar 3.84 merupakan gambar halaman *Terms of Use* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Selanjutnya, gambar *widescreen* di bawah *header* (lihat gambar 3.84) juga dihilangkan. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman-halaman sebelumnya. Perbedaan utamanya adalah sekarang paragraf *terms of use* dibuat dengan menggunakan tabel agar terlihat lebih rapi. Cara membuatnya lagi-lagi menggunakan fungsi `<table>` yang di dalamnya terdapat fungsi `<td>` dan `<tr>`. Kemudian, diatur juga *alignment* dari setiap *text*. Untuk *text* judul *terms*, akan menggunakan *alignment* rata kanan, sedangkan untuk penjelasannya menggunakan *alignment justify*. Kemudian, fungsi `<table>` dalam halaman ini diberikan *background* berwarna putih dan garis *border* warna abu-abu.

### 3.3.6.8 Member Categories



Gambar 3.85 Halaman *Members by Categories* Baru

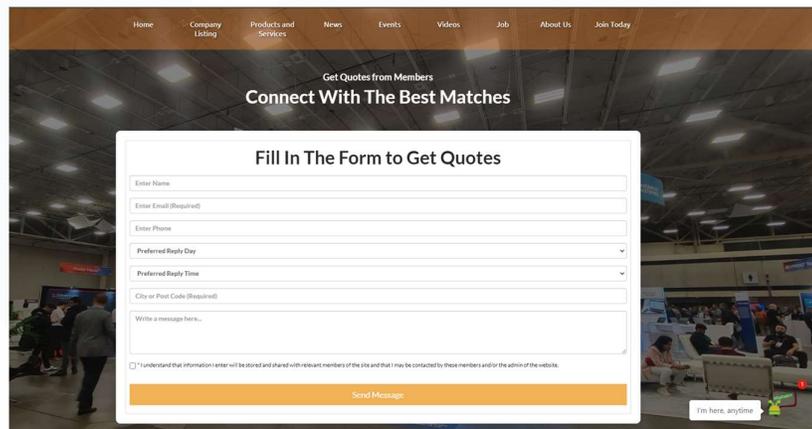


Gambar 3.86 Halaman *Members by Categories* Lama

Gambar 3.85 merupakan gambar halaman *Members by Categories* yang baru, sedangkan gambar 3.86 merupakan gambar halaman *Member by Categories* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Members by Categories” yang dibuat dengan fungsi `<span>`. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman-halaman sebelumnya. Perbedaan utamanya adalah pilihan *categories* tersebut

dibuat dalam sebuah tabel. Tugas dari peserta magang dalam hal ini hanyalah memberikan *background* berwarna putih dengan menggunakan *.css*. Hal-hal lain dalam hal ini dilakukan oleh karyawan IT yang lebih professional karena terdapat hubungan dengan *database*.

### 3.3.6.9 Get Quotes



Gambar 3.87 Halaman *Get Quotes* Baru

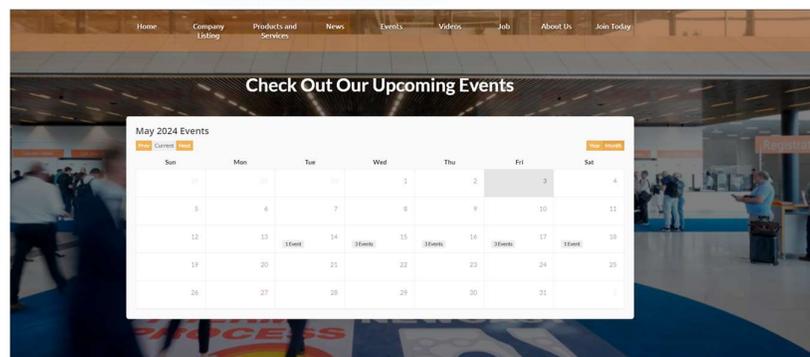


Gambar 3.88 Halaman *Get Quotes* Lama

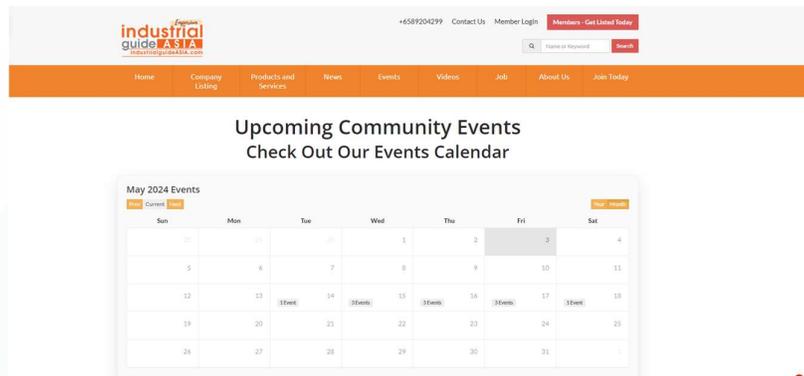
Gambar 3.87 merupakan gambar halaman *Get Quotes* yang baru, sedangkan gambar 3.88 merupakan gambar halaman *Get Quotes* yang lama. Terdapat beberapa perbedaan. Pertama, bagian

*header* dihilangkan. Selanjutnya, gambar *widescreen* di bawah *header* (lihat gambar 3.88) juga dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Get Quotes from Members” dan “Connect With The Best Matches” yang dibuat dengan fungsi `<span>` dengan ukuran yang berbeda. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman-halaman sebelumnya. Perbedaan utamanya adalah bagian *form* tersebut diperbesar dan diberikan *background* warna putih. Kemudian, bagian “The Best Way to Get Quotes” di halaman lama juga dihilangkan secara total. Halaman ini tidak ditugaskan kepada peserta magang karena memerlukan akses ke *database* sehingga diserahkan kepada karyawan IT yang lebih profesional.

#### 3.3.6.10 *Upcoming Community Events*



Gambar 3.89 Halaman *Upcoming Community Events* Baru



Gambar 3.90 Halaman *Upcoming Community Events* Lama

Gambar 3.89 merupakan gambar halaman *Upcoming Community Events* yang baru, sedangkan gambar 3.90 merupakan gambar halaman *Upcoming Community Events* yang lama. Terdapat beberapa perbedaan. Pertama, bagian *header* dihilangkan. Lalu, terdapat judul berwarna putih bertuliskan “Check Out Our Upcoming Events” yang dibuat dengan fungsi `<span>`. Kemudian, bagian *navigation bar* dibuat berkurang *opacity*-nya jika di paling atas dan akan bertambah *opacity*-nya sampai 100% jika *user* melakukan *scrolling*. Kurang lebih perbedaannya hampir sama dengan halaman-halaman sebelumnya. Tidak ada perbedaan yang signifikan dari segi kalender tersebut karena kalender tersebut merupakan sebuah *embed code* yang dilakukan oleh karyawan IT untuk menampilkan kalender *timeline event* perusahaan yang berasal dari Google Calendar.

Dengan demikian, itulah halaman-halaman yang direvitalisasi. Proyek ini belum sepenuhnya selesai. Masih banyak hal-hal yang harus direvisi, dibahas kembali, diotomastisasi, dan bahkan dihilangkan atau diubah sepenuhnya.

### 3.3.7 Email Marketing Automation

*Email marketing automation* adalah sebuah tindakan untuk melakukan otomatisasi *email* guna untuk memberikan pengumuman, promosi, notifikasi, dan sebagainya. Dalam kasus perusahaan Orange Media Group sendiri, *email marketing automation* sendiri sering digunakan untuk promosi produk dan notifikasi berhasil menjadi *subscriber website* mereka. Salah satu *tools email marketing automation* yang digunakan oleh perusahaan adalah VBout. Alasan kenapa VBout digunakan adalah VBout dapat memenuhi kebutuhan perusahaan yang berurusan dengan *email marketing*. Selain itu, *supervisor* peserta magang yang sekaligus seorang *marketing director* memiliki hubungan yang erat dengan CEO dari VBout. Hal tersebut menyebabkan adanya interaksi yang saling menguntungkan baik untuk Orange Media Group maupun VBout sendiri. Selama menjalani program magang, peserta magang diberikan dua tugas yang berhubungan dengan *email marketing automation* ini, yaitu *blasting email* dan juga menambahkan fitur *email notification* untuk salah satu *website* perusahaan, yaitu *iiga.news*.

#### 3.3.7.1 Blasting Email

*Blasting email* adalah sebuah tindakan untuk mengirimkan *email* ke banyak *email*. Tentu hal tersebut apabila dilakukan satu per satu akan sangat memakan waktu. Oleh karena itu, diperlukan sebuah *tools* yang dapat melakukan *blasting email*. Dalam hal ini, VBout merupakan *tools* yang cocok untuk melakukan ini.

Langkah pertama jika ingin melakukan *blasting email* adalah memasukkan *email-email* tujuan ke dalam VBout. Dalam hal ini, apabila dimasukkan satu per satu sebenarnya bisa saja, tetapi hal tersebut sama saja dengan mengirimkan *email* satu per satu secara manual dan tentu saja akan memakan waktu yang banyak. Oleh

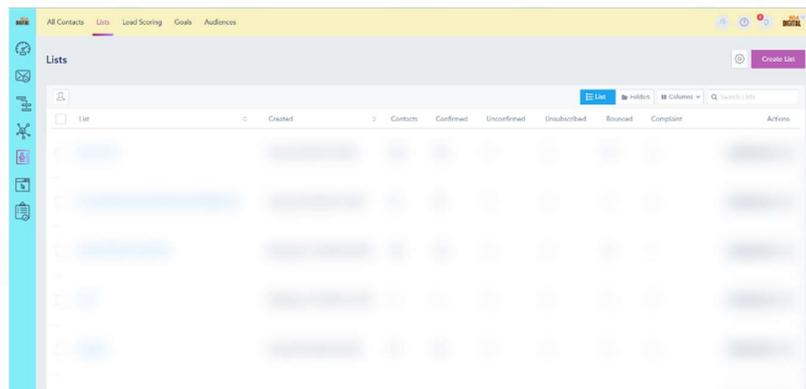
karena itu, lebih disarankan untuk memasukkan *email-email* tujuan ke dalam *file* excel terlebih dahulu. Berikut merupakan contohnya.



FIRST NAME	LAST NAME	EMAIL ID
[Blurred]	[Blurred]	[Blurred]

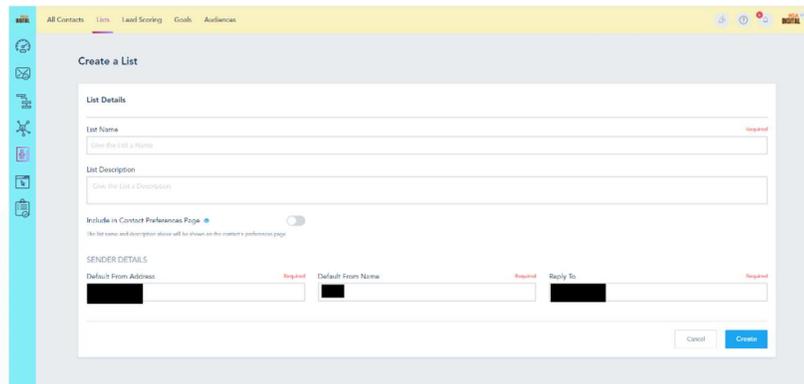
Gambar 3.91 Contoh *List Email* untuk VBout

Gambar 3.91 merupakan sebuah contoh *list email* yang bisa dimasukkan ke dalam VBout. Hal yang paling penting untuk dimasukkan ke VBout adalah alamat *email* karena tujuannya adalah untuk melakukan *blasting email*. Jika tidak ada *email*, VBout akan otomatis *skip* untuk mengirim *email* ke orang tersebut. Kemudian, simpan *file* tersebut ke dalam format *.csv*. Selanjutnya, pergi ke VBout dan kemudian pilih menu *Contacts* dan ke bagian *List*. Tampilannya kurang lebih seperti ini.



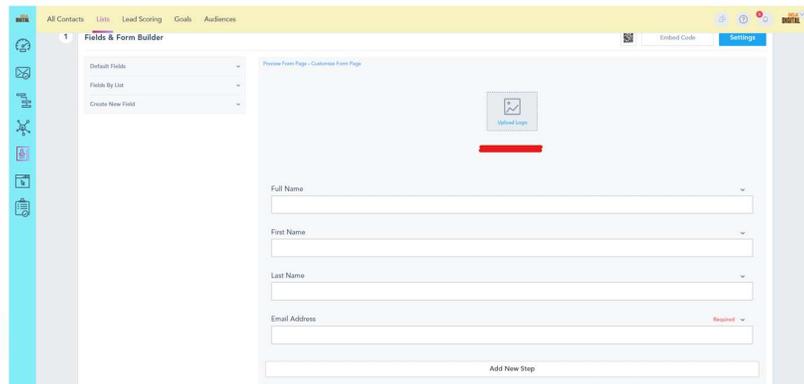
Gambar 3.92 Tampilan Bagian *List Contact*

Gambar 3.92 merupakan tampilan bagian *list contact* di dalam VBout. Klik *button* Create List di kanan atas.



Gambar 3.93 Tampilan *Create List*

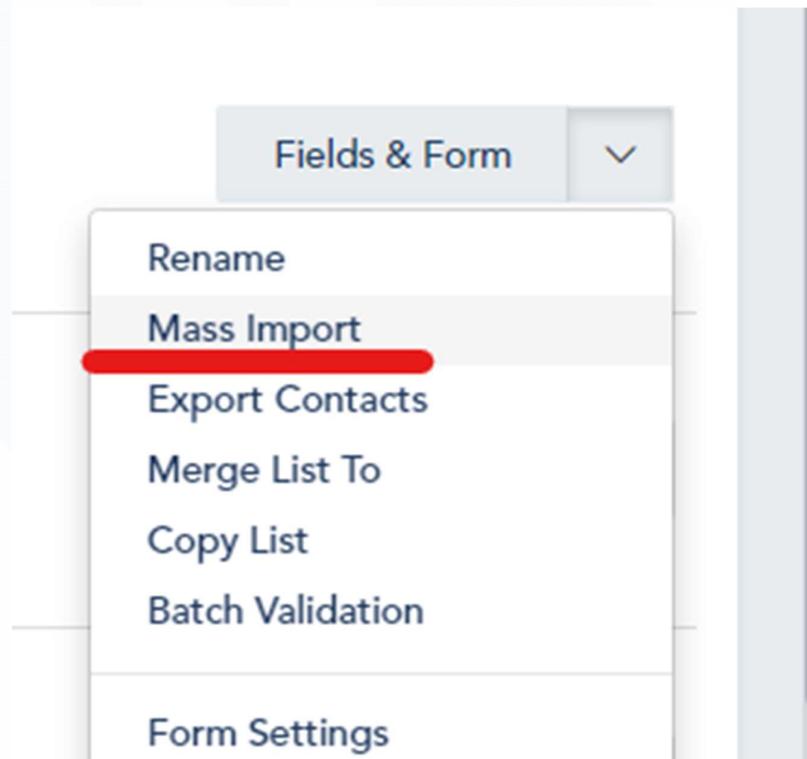
Gambar 3.93 merupakan tampilan *Create List*. Dalam halaman ini, *user* dapat memasukkan nama *list* (*required*) dan *list description* (*optional*). Jika sudah diisi, klik *button* *Create* di kanan bawah. Sesudah itu, *user* akan kembali ke halaman *Contact List* seperti di gambar 3.92. Hal yang berbeda adalah nama *list* yang sudah ada di *list* halaman tersebut. Klik *Fields & Form* di bagian paling ujung kanan nama *list*.



Gambar 3.94 Tampilan *Fields & Form*

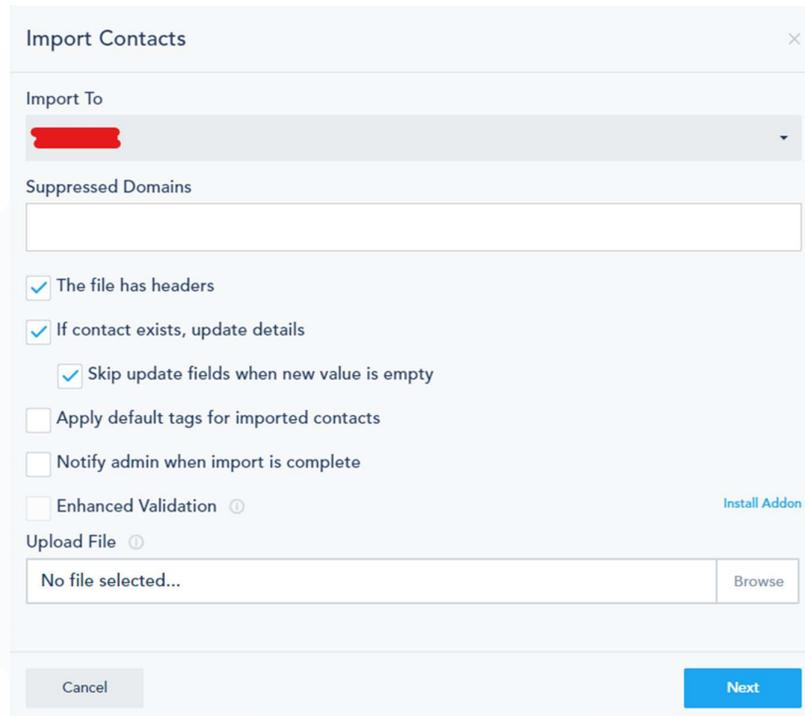
Gambar 3.94 menunjukkan gambar tampilan ketika *user* klik *Fields & Form*. *Fields & Form* merupakan sebuah bagian untuk menyesuaikan *file .csv* yang akan di-*upload* nantinya supaya tidak terjadi *error* di *VBout*. Pastikan *column* yang *user* ingin masukkan

sesuai dengan *form input* di VBout. Jika sudah sesuai, klik Save Form di kanan bawah. Lalu, *user* akan kembali ke halaman *list contact*. Kemudian, klik *icon drop-down* di sebelah tulisan Fields & Form.



Gambar 3.95 *Drop-down* Menu Sebelah Fields & Form

Gambar 3.95 merupakan gambar yang menunjukkan menu *drop-down* di sebelah Fields & Form. Untuk melakukan *bulk import* data dari *file .csv*, pilih menu Mass Import. Jika diklik, akan muncul *popup* sebagai berikut.



Gambar 3.96 *Popup* Mass Import

Gambar 3.96 merupakan gambar yang menunjukkan *popup* jika *user* klik Mass Import. Dalam *popup* ini pilihlah *list* yang masu di-*import* datanya di bagian Import To. Lalu, lakukan centang untuk hal-hal yang diperlukan dan terakhir, pilih *file* .csv yang akan di-*upload*. Jika sudah, klik Next di bagian kanan bawah.

Import Contacts from Excel to "[redacted]"

Import "Full Name" to  
 Custom Field Full Name

Import "First Name" to  
 Default Field First Name

Import "Last Name" to  
 Default Field Last Name

Import "Work Email #1" to  
 Email Field Email Address

Import "Work Email #2" to  
 None

Import "Direct Email #1" to  
 None

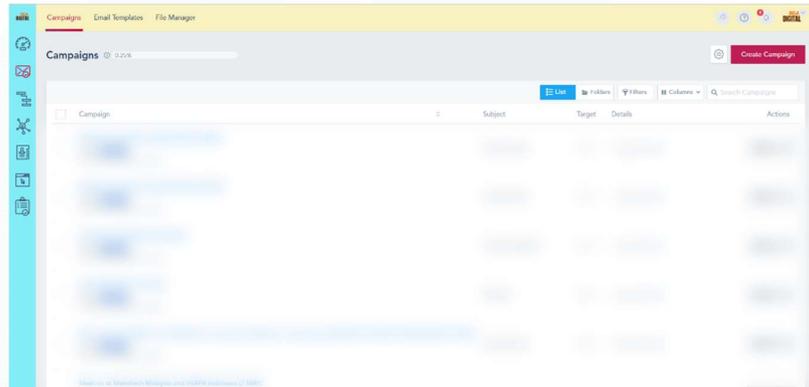
Import "Direct Email #2" to  
 None

Cancel Import

Gambar 3.97 *Popup* untuk Mencocokkan Nama *Column* dan *Form*

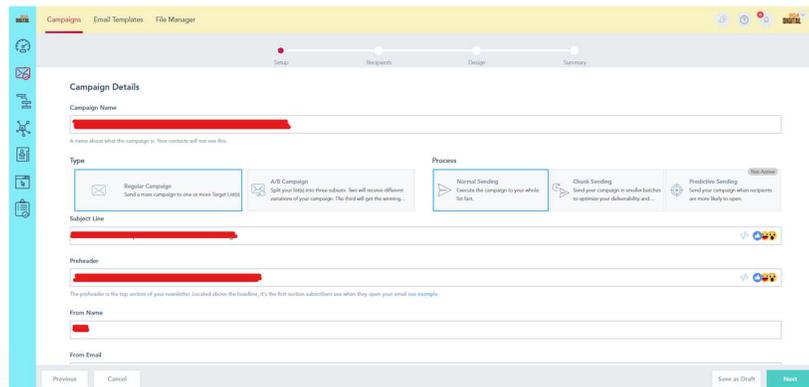
Gambar 3.97 merupakan sebuah *popup* untuk mencocokkan nama *column* di *file .csv* sama dengan nama *form*. Hal ini dilakukan supaya data yang dimasukkan sesuai dengan yang seharusnya di dalam VBout. Jika sudah sesuai, klik Import di bagian kanan bawah. Setelah itu, VBout akan memasukkan data-data dari *.csv* tersebut. Lama tidaknya VBout memasukkan data tergantung banyak data dalam *.csv* tersebut. Setelah selesai, periksa kembali nama-nama *contacts* apakah sudah masuk semua atau belum. Dengan demikian, itulah langkah pertama, yaitu memasukkan *contact list* ke dalam VBout.

Langkah berikutnya adalah membuat *email marketing* itu sendiri. Caranya adalah pertama-tama, klik menu *email* dan pergi ke bagian *campaign*. Berikut merupakan tampilannya.



Gambar 3.98 Halaman *Email Campaign*

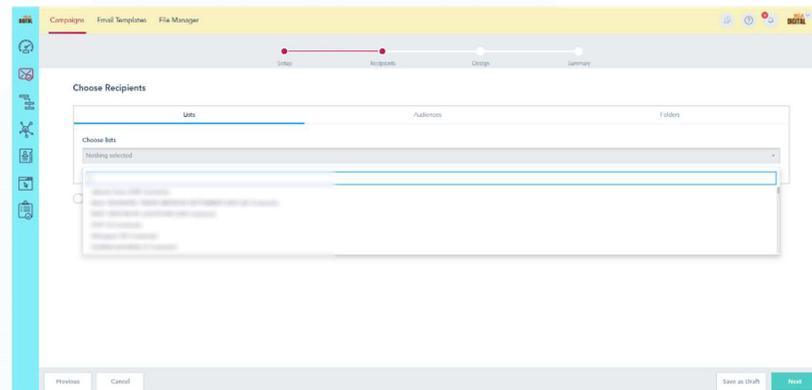
Gambar 3.98 merupakan sebuah gambar yang menunjukkan *list* dari *email campaign*. Untuk membuat *email campaign*, klik *button* Create Campaign di kanan atas.



Gambar 3.99 Halaman Create Campaign

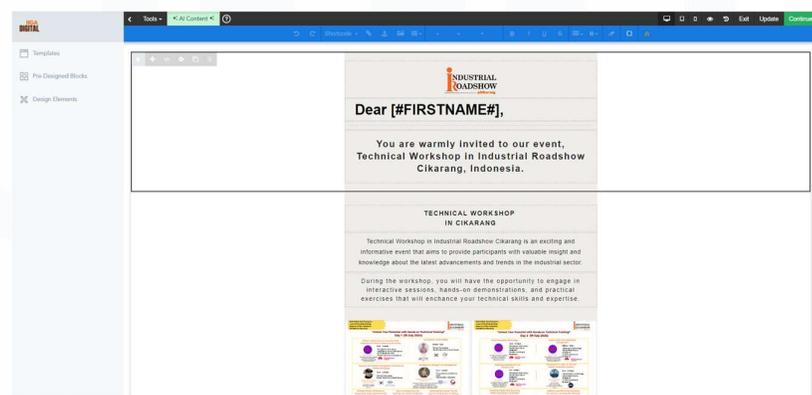
Gambar 3.99 merupakan gambar yang menunjukkan halaman Create Campaign. Dalam hal ini, *user* diminta untuk memasukkan beberapa hal, yaitu Campaign Name, Subject Line, dan Preheader. Tiga hal tersebut merupakan hal yang paling penting.

Sisanya, *autogenerated* sesuai dengan *domain email sender* yang dimasukkan.



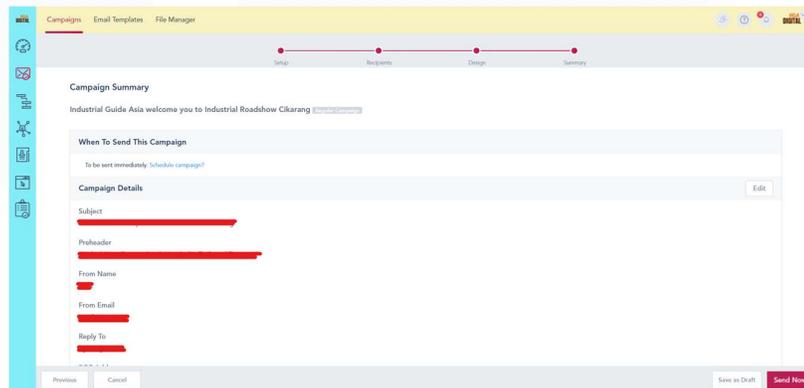
Gambar 3.100 Halaman *Choose Recipients*

Gambar 3.100 merupakan gambar yang menunjukkan halaman *Choose Recipients*. Dalam hal ini *user* diminta untuk memilih *list* mana saja dari *contacts* yang sebelumnya telah dijelaskan untuk dikirim *email campaign* tersebut. *User* hanya tinggal melakukan klik di *list-list* yang tersedia di menu *drop-down* tersebut. Hal yang menarik juga adalah bisa memilih lebih dari satu *list*.



Gambar 3.101 Bagian Desain *Email Campaign*

Gambar 3.101 merupakan gambar yang menunjukkan bagian untuk mendesain poster *email campaign*. Hal yang menarik adalah *user* bisa menggunakan AI untuk menggunakan *content*, melakukan *import* poster, atau melakukan desain sendiri *from scratch*. Kemudian, *user* juga bisa menambahkan sebuah *code* untuk memanggil data. Sebagai contoh di dalam gambar 3.136 adalah memanggil data *firstname* sehingga nanti pada saat *email* sudah dikirim, *code* *firstname* tersebut akan berubah menjadi nama orang yang memiliki *email* tersebut berdasarkan data.



Gambar 3.102 Bagian *Summary*

Gambar 3.102 merupakan gambar yang menunjukkan bagian *summary* yang merupakan tahap akhir dalam *email automation* dengan menggunakan VBout. Dalam bagian ini *user* dapat memeriksa terlebih dahulu apakah semuanya sudah benar atau belum. Lalu, *user* dapat men-*schedule* kapan *email* tersebut akan dikirimkan atau bisa juga dikirimkan saat itu juga dengan mengklik tombol Send Now di kanan bawah. Bahkan, *user* juga bisa mengirimkan *email* testing ke beberapa alamat *email* untuk memeriksa bagaimana hasil dari *email campaign* tersebut. Dengan demikian, itulah bagaimana cara peserta magang melakukan *email blasting* yang biasanya digunakan oleh perusahaan untuk

mempromosikan apabila ada produk baru, berita baru, majalah baru, *event* baru, dan sebagainya.

### 3.3.7.2 *Email Notification Iiga.news*



Gambar 3.103 Halaman Utama Bagian Atas Iiga.news

Gambar 3.103 merupakan sebuah gambar salah satu *website* perusahaan, yaitu *iiga.news*. Secara singkat, *iiga.news* merupakan sebuah *website* perusahaan yang menampilkan berita-berita milik perusahaan dan juga majalah hasil karya perusahaan. *Iiga.news* dibuat dengan menggunakan CMS yang ternama, yaitu Webflow. Alasan perusahaan menggunakan Webflow adalah terdapat *CMS Collection* yang siap digunakan sebagai *database* berita-berita milik perusahaan yang dapat ditampilkan di *front-end website* tersebut. Dalam hal ini, peserta magang tidak ditugaskan untuk membuat *website* *iiga.news*, tetapi ditugaskan untuk menambahkan *email marketing* di dalam *website* tersebut. Kasusnya adalah sebagai berikut.



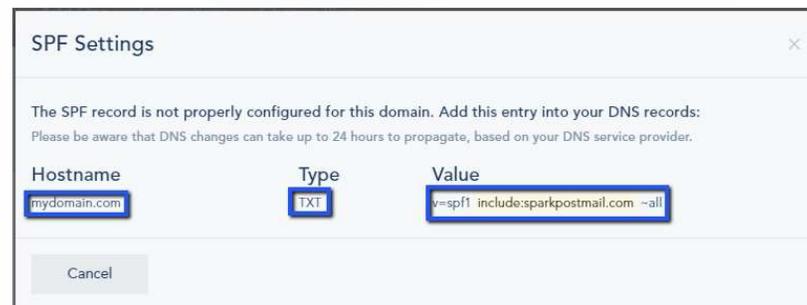
Gambar 3.104 Bagian *Footer* iiga.news

Gambar 3.104 merupakan sebuah gambar yang menunjukkan bagian *footer* dari iiga.news. Dalam *footer* tersebut, terdapat sebuah *form* yang dapat diisi oleh *user*. *Form* tersebut digunakan untuk mengumpulkan data-data *user* yang nantinya akan dimasukkan ke dalam *automation tools* untuk di *blast email* promosi perusahaan nantinya. Namun, *form* tersebut rusak. Seharusnya, ketika *user* klik tombol *Submit*, *user* akan mendapatkan sebuah *email* notifikasi mengenai tanda terima kasih perusahaan kepada *user* tersebut sudah melakukan *subscribe* ke iiga.news. Kemudian, sebagai admin, *supervisor* peserta magang juga harus mendapatkan *email* notifikasi yang menyatakan bahwa seseorang telah *subscribe* ke iiga.news lengkap dengan nama dan *email*. Dalam hal ini, peserta magang diberikan sebuah tugas untuk memperbaiki kerusakan *form* tersebut dengan menggunakan VBout.

Langkah pertama yang harus dilakukan adalah melakukan *embed code* yang akan diletakkan di Google Tag Manager. Kemudian, lakukan *inspect elements* untuk *form* tersebut guna mencari id dari masing-masing *input type*. Hal ini penting untuk dilakukan karena id dari masing-masing *input type* ini akan dimasukkan ke dalam *embed code* untuk Google Tag Manager juga. Sesudah itu, masukkan *embed code* ke dalam Google Tag Manager lalu *publish*. Setelah itu, terdapat sebuah *embed code* lagi yang

berasal dari Google Tag Manager. *Copy embed-code* tersebut dan akan di *paste* ke dalam Webflow dengan menggunakan fungsi Embed di dalam Webflow. Selanjutnya, *publish website* tersebut.

Selanjutnya, kita harus mengatur *domain sender* supaya *email* yang diterima oleh *user* berasal dari *domain* iiga.news dan bukan dari *domain* VBout. Langkah pertama adalah kita harus memeriksa apakah SPF, DKIM, dan DMARC dari *domain website* yang kita miliki sudah terverifikasi atau belum. Hal yang beruntung untuk peserta magang adalah DKIM dan DMARC dari *website* iiga.news sudah terverifikasi. Hanya SPF saja yang belum terverifikasi sehingga peserta magang harus melakukan verifikasi tersebut. Untuk melakukan verifikasi, langkah pertama yang harus dilakukan adalah mendapatkan *Hostname*, *Type*, dan juga *Value* yang nantinya akan dimasukkan ke dalam penyedia layanan *hosting website*. Berikut ini merupakan contohnya.

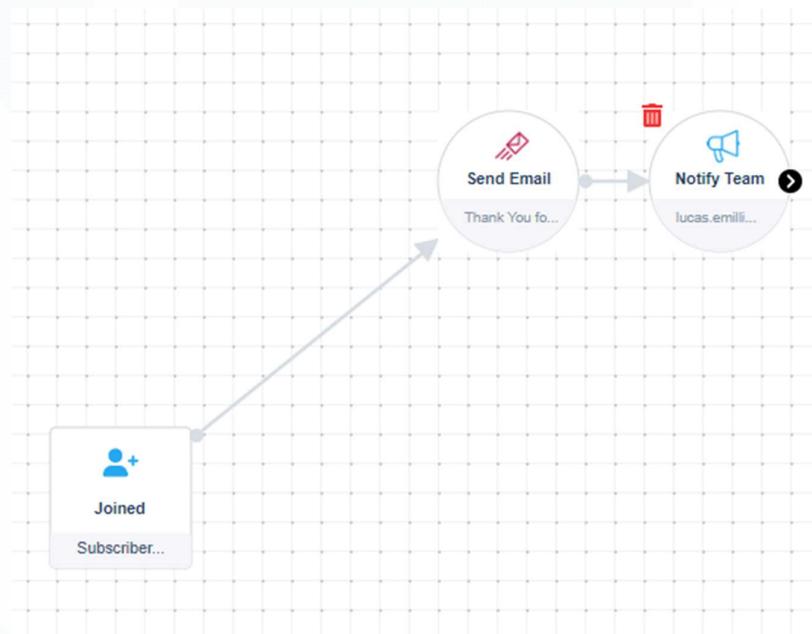


Gambar 3.105 Contoh SPF *Verification* [5]

Gambar 3.105 merupakan sebuah gambar yang menunjukkan SPF *verification* yang nantinya akan di masukkan ke dalam *hosting website*. Perlu diingat bahwa, gambar tersebut hanyalah contoh dan bukan SPF milik iiga.news. Berikutnya, untuk melakukan verifikasi, perlu untuk memasukkan *Hostname*, *Type*, dan juga *Value* ke dalam *hosting website*. Oleh karena itu, peserta magang meminta izin *supervisor* untuk masuk ke dalam *website*

penyedia layanan *hosting* untuk *iiga.news*, yaitu *namecheap*. Setelah mendapatkan *email* dan *password* untuk masuk ke *namecheap*, peserta magang memasukkan *Hostname*, *Type*, dan juga *Value* ke tempat yang diperlukan. Kemudian, setelah semuanya di-*save*, harus menunggu kurang lebih tiga puluh menit untuk melihat perubahan yang terjadi. Setelah tiga puluh menit, akhirnya *domain* *iiga.news* terverifikasi secara SPF. Hal tersebut menyatakan bahwa *domain* *iiga.news* siap untuk melakukan pengiriman *email*.

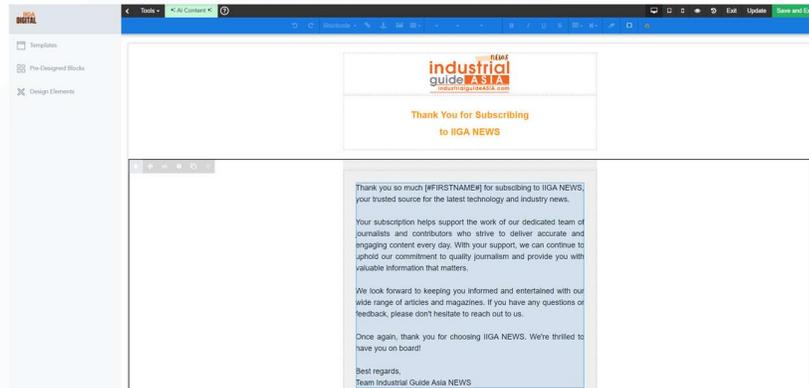
Langkah berikutnya adalah membuat *logic* seperti gambar berikut.



Gambar 3.106 *Logic Email Automation* ke *User* dan *Admin*

Gambar 3.106 merupakan sebuah gambar yang menunjukkan *logic* sederhana pengiriman *email* tanda terima kasih kepada *user* yang sudah klik *button* Submit di *form footer* *iiga.news*. Setelah *emain* terkirim ke *user*, terdapat *email* juga yang terkirim ke admin yang berbunyi bahwa seseorang telah *subscribe*.

Langkah selanjutnya adalah mendesain poster tanda terima kasih. Hasilnya adalah sebagai berikut.



Gambar 3.107 Desain untuk Poster Terima Kasih Telah *Subscribe*

Gambar 3.107 merupakan gambar desain yang diberikan kepada *user* sebagai tanda terima kasih sudah *subscribe* ke *iiga.news*. Setelah selesai, kembali lagi ke halaman *logic* untuk melakukan perubahan di bagian Send Email. Hal yang diubah adalah menambahkan *email campaign* supaya yang terkirim ke *user* adalah poster terima kasih yang telah dibuat.



Gambar 3.108 Hasil untuk *User*



Gambar 3.109 Hasil untuk Admin

Gambar 3.108 merupakan gambar hasil yang diperoleh dari sisi *user*. Berdasarkan gambar tersebut, *user* akan melihat *poster* tanda terima kasih lengkap dengan namanya sesuai dengan data yang diperoleh. Gambar 3.109 merupakan gambar hasil untuk admin. Berdasarkan gambar tersebut, dapat dilihat bahwa dari sisi Admin mendapatkan sebuah *email* yang mentakan bahwa “Someone just Subscribe to IIGA News” dan kemudian *email* tersebut berisi data-data dari *form* yang berasal dari *iiga.news*

### 3.3.8 *Quality Assurance Iiga.news*

*Quality assurance* merupakan sebuah tindakan untuk memeriksa kualitas dalam sebuah *website*, aplikasi, program, dan lain-lain. Dalam hal ini, hal yang akan diperiksa kualitasnya adalah *website* *iiga.news*. Dalam hal ini, digunakan sebuah *tools* yang bernama Labrica untuk memeriksa kondisi dari *iiga.news*. Perlu diingat bahwa bagian *quality assurance* ini tidak akan terlalu panjang untuk dibahas karena alasan utama *supervisor* meminta peserta magang untuk melakukan *quality assurance* adalah untuk melihat kualitas *website* dan juga melihat apakah selama ini para karyawan *sales / marketing* mengunggah konten berita di *website* *iiga.news* dengan benar atau tidak. Berikut ini merupakan hasil auditnya.

Technical Audit		
<div style="border: 1px solid orange; padding: 5px;"> <span style="color: orange;">⚠</span> Many technical errors were found that may be damaging your rankings         </div>		
Site summary report		
<input checked="" type="checkbox"/> Show critical technical errors	11130	<a href="#">Description</a>
<input checked="" type="checkbox"/> Show other technical errors	6	<a href="#">Description</a>
<input checked="" type="checkbox"/> Show warnings	343962	<a href="#">Description</a>
<input checked="" type="checkbox"/> Show notices	0	<a href="#">Description</a>

Gambar 3.110 Hasil Audit Labrica

Gambar 3.110 merupakan gambar hasil dari audit Labrica. Berdasarkan gambar tersebut terdapat sebelas ribu lebih *error* yang cukup fatal. Dalam hal ini, Labrica tidak hanya memeriksa *error* di dalam kodingan *website*, tetapi juga memeriksa hal-hal apa saja yang menyebabkan *website* iiga.news jelek di mata Google. Hal-hal tersebut yang dibutuhkan untuk *supervisor* bisa mengevaluasi pekerjaan karyawan *sales / marketing* karena merekalah yang paling sering melakukan *upload content* di iiga.news.

Berdasarkan hasil rangkuman Labrica, terdapat beberapa masalah yang menyebabkan *website* iiga.news jelek di mata Google.

1. Terdapat beberapa berita yang terdeteksi mengandung kata-kata kasar dan konten dewasa. Hal ini cukup unik dikarenakan kata-kata kasar dan konten dewasa dalam berita-berita iiga.news adalah tidak lain nama orang yang memiliki seolah-olah memiliki kata-kata kasar. Kemudian, terdapat kata-kata juga yang sebenarnya tidak kasar bagi kita, tetapi dianggap kasar oleh Google, seperti *strap-on*, *dong*, *cing*, dan lain sebagainya.
2. Terdapat beberapa halaman yang terduplikasi. Berdasarkan apa yang peserta magang temukan, halaman yang terduplikasi merupakan halaman yang untuk coba-coba tetapi ter-*publish*. Sebagai contoh, seseorang mencoba untuk mengunggah sebuah berita dan kemudian

tidak jadi menggunakan halaman tersebut. Kemudian, ia membuat halaman baru dengan judul yang sama dengan halaman percobaannya sebelumnya. Hal tersebut tanpa sengaja bisa saja ter-*publish* dua-duanya dan menyebabkan terdeteksi duplikasi halaman *web*.

3. Konten berita memiliki kata-kata di bawah 150 kata. Hal tersebut menyebabkan sebuah masalah yang membuat *rank website* iiga.news menurun karena isi konten yang sedikit.
4. *Meta-description* tidak boleh terlalu panjang ataupun terlalu pendek. Setidaknya 15-150 kata.
5. *SEO-description* tidak dicantumkan, sehingga menyebabkan *website* iiga.news tidak dikenal oleh *user* Google. Hal ini cukup fatal karena SEO berperan penting untuk memperkenalkan konten *website* di Google. Tanpa SEO, *user* Google tidak akan bisa menemukan iiga.news di Google sehingga *rank website* pun akan turun. Oleh karena itu, sangat krusial untuk mencantumkan *SEO-description*.
6. Beberapa *link* mengarahkan ke halaman yang tidak ada lagi atau rusak. Hal ini akan memperburuk kualitas *website* di mata Google.
7. Terdapat juga *link* yang mengarah ke *website* yang tidak terpercaya. Hal ini juga akan memperburuk kualitas *website* di mata Google.

Dengan demikian, itulah masalah-masalah yang menyebabkan iiga.news jelek di mata Google. Oleh karena itu, diperlukan solusi yang tepat dan baik untuk mengatasi masalah-masalah di atas, seperti penyensoran kata-kata kasar, menambah isi konten, mengisi *SEO-description*, menghilangkan *link* rusak, dan sebagainya.

### 3.3.9 Rancang *Website* Cikarangshow.com

Tugas berikutnya yang diberikan *supervisor* kepada peserta magang adalah ikut serta membantu perancangan sebuah *website* bernama cikarangshow.com. Secara singkat, perusahaan Orange Media Group ingin mengadakan sebuah *event technical workshop* yang diadakan di Cikarang pada 18 Juli 2024 – 19 Juli 2024. *Technical workshop* tersebut berisi kelas-

kelas, seperti kelas robotik, manufaktur, industri, dan lain sebagainya. *Event* juga akan ada pameran perusahaan-perusahaan industri sehingga mereka dapat memamerkan barang dan jasanya di pameran tersebut. *Event* ini akan dihadiri oleh banyak pengajar dari luar negeri untuk mengajar *workshop* tersebut dan juga dihadiri oleh perusahaan-perusahaan yang berasal dari dalam dan luar negeri juga. Oleh karena itu, *event* ini merupakan *event* yang besar yang diberikan oleh perusahaan untuk Indonesia. Berdasarkan hal tersebut, perusahaan memerlukan sebuah *website* yang dapat melakukan promosi *event* perusahaan dan juga dapat melakukan registrasi. Maka, dibuatlah *website* bernama cikarangshow.com.

Dalam menengerjakan *website* ini, peserta magang dibantu oleh beberapa pihak, seperti *supervisor*, divisi UI/UX, divisi IT, dan divisi *sales / marketing*. *Supervisor* bertugas untuk memberikan arahan konsep tampilan *website*. Divisi UI/UX bertugas untuk membuat desain UI dari *website* ini dan juga merancang gambar, poster, dan *banner*. Divisi IT, yang di dalamnya peserta magang juga turut serta, bertugas untuk merancang *website* ini sampai *deploy*. Divisi *sales / marketing* bertugas untuk memasukkan konten dan juga mempromosikan *website* ini nantinya apabila sudah jadi. Karena banyaknya pihak dan karyawan dari divisi IT yang bertindak untuk pengembangan *website* ini, peserta magang tidak mengerjakan semua bagian dari *website* ini. Oleh karena itu, peserta magang akan lebih berfokus untuk mencertiakan bagian yang dibuat oleh peserta magang saja.



Gambar 3.111 Halaman Utama Cikarangshow.com

Gambar 3.111 merupakan sebuah gambar yang menunjukkan halaman utama dari cikarangshow.com. Berdasarkan *navigation bar* yang berada di atas, terdapat beberapa menu yang akan menuju ke halaman-halaman tersebut. Dalam hal ini, bagian yang dibuat oleh peserta magang adalah bagian *Technical Workshop*.

Terdapat tiga buah halaman yang dikerjakan oleh peserta magang dan dibantu oleh karyawan IT lainnya, yaitu halaman utama *Technical Workshop*, *Day 1 Workshop*, dan *Day 2 Workshop*.

### 3.3.9.1 Halaman Utama *Technical Workshop*



Gambar 3.112 Halaman *Technical Workshop* Bagian Atas

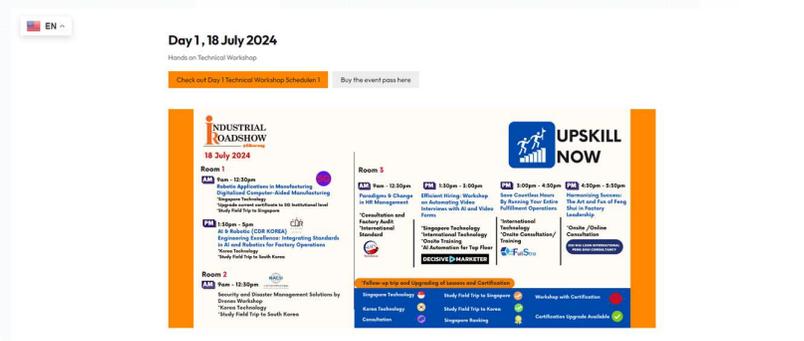
Jika *user* klik menu *Technical Workshop*, akan memunculkan halaman seperti gambar 3.112. Terdapat sebuah judul bertuliskan “Technical Workshop” yang dibuat dengan fungsi `<h1>`. Judul tersebut dibungkus dengan fungsi `<section>` sehingga dapat dikustomisasi di `.css` sampai menjadi sedemikian rupa. Bagian bawahnya terdapat tulisan “18 - 19 July 2024, Jababeka Convention Center Cikarang, Indonesia” yang dibuat dengan menggunakan fungsi `<p>` dan dibungkus dengan fungsi `<section>` juga untuk dikustomisasi di `.css` agar menjadi sedemikian rupa. Kemudian, bagian bawahnya terdapat sebuah poster besar yang diletakkan dengan fungsi `<img>` yang dibungkus dengan `<div>` supaya bisa dikustomisasi di `.css` sampai sedemikian rupa.



Gambar 3.113 *Gallery Technical Workshop*

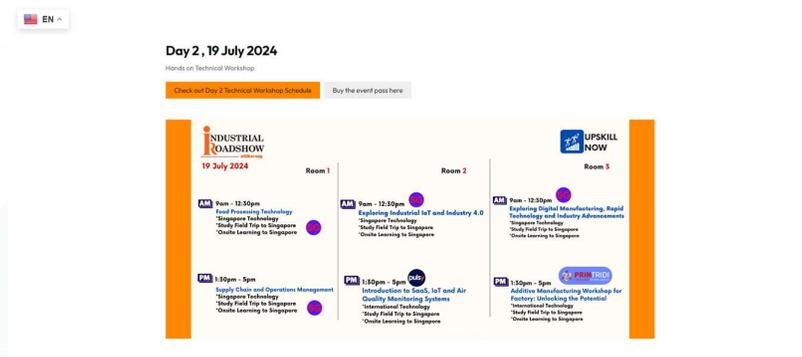
Bagian selanjutnya dalam halaman utama *Technical Workshop* adalah *gallery* (gambar 3.113). *Gallery* ini menampilkan foto-foto *workshop* yang pernah dilaksanakan oleh perusahaan tahun-tahun sebelumnya. Dalam hal ini, terdapat sebuah judul yang dibuat dengan fungsi `<h2>` dan `.css` untuk diatur letak, ukuran, dan posisinya sampai sedemikian rupa. Kemudian, untuk gambar-gambar tersebut digunakanlah fungsi sebuah JavaScript karena terdapat sebuah animasi di mana ketika *cursor mouse* didekatkan ke salah satu gambar, ia akan meng-*highlight* gambar tersebut dan

menurunkan *opacity* untuk gambar lainnya. Lalu, digunakanlah fungsi `<img>` untuk meletakkan gambar. Kemudian, *element* `<img>` dan JavaScript dibungkus dengan sebuah `<div class>` yang kemudian semuanya dibungkus lagi dalam sebuah `<div class>` yang akan dikostumisasi di `.css` menjadi seperti *grid* yang terlihat di gambar 3.113.



Gambar 3.114 Bagian *Day 1*

Gambar 3.114 merupakan bagian yang memperlihatkan poster *day 1* untuk acara ini. Terdapat sebuah judul bertuliskan “Day 1, 18 July 2024” yang dibuat dengan fungsi `<h2>`. Judul tersebut dibungkus dengan fungsi `<section>` sehingga dapat dikostumisasi di `.css` sampai menjadi sedemikian rupa. Bagian bawahnya terdapat tulisan “Hands on Technical Workshop” yang dibuat dengan menggunakan fungsi `<p>` dan dibungkus dengan fungsi `<section>` juga untuk dikostumisasi di `.css` agar menjadi sedemikian rupa. Kemudian, di bawah judul tersebut terdapat sebuah *button* yang dibuat dengan fungsi `<button>` yang akan *link* ke halaman lain. Hal ini juga dikostumisasi di `.css` hingga menjadi sedemikian rupa. Kemudian, di bawahnya terdapat sebuah *banner* yang menunjukkan kegiatan hari pertama. Untuk meletakkan gambar ini, digunakanlah fungsi `<img>` dan `.css` untuk dikostumisasi sampai menjadi sedemikian rupa.

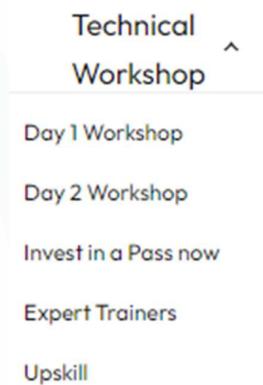


Gambar 3.115 Bagian *Day 2*

Gambar 3.115 merupakan sebuah gambar yang menunjukkan bagian poster *day 2* untuk acara ini. Kurang lebih cara pembuatannya sama dengan bagian *day 1* sebelumnya sehingga bagian ini tidak dijelaskan ulang cara pembuatannya.

### 3.3.9.2 Halaman *Technical Workshop Day 1*

Jika kita kembali ke bagian *navigation bar*, terdapat sebuah menu *drop-down* di bagian menu *Technical Workshop*.



Gambar 3.116 *Drop-down Menu Technical Workshop*

Gambar 3.116 merupakan sebuah gambar yang menunjukkan *drop-down menu* untuk menu *Technical Workshop*. Fungsi dari *drop-down menu* tersebut adalah untuk men-*direct* ke

halaman lain di *website* ini. Berdasarkan kelima halaman tersebut, dua halaman yang dibuat oleh peserta magang dan dibantu oleh karyawan IT lainnya adalah halaman *Day 1 Workshop* dan *Day 2 Workshop*.



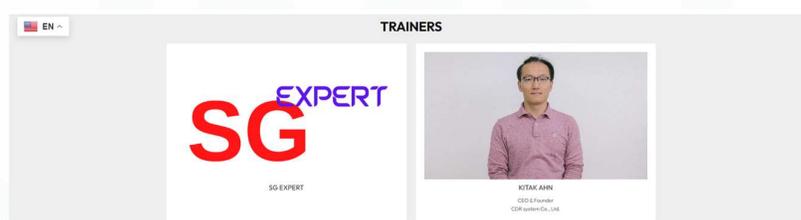
Gambar 3.117 Halaman *Day 1* Bagian Atas

Gambar 3.117 merupakan gambar yang menunjukkan bagian atas dari halaman *day 1*. Berdasarkan gambar tersebut, dapat dilihat bahwa terdapat sebuah *banner* yang diletakkan dengan menggunakan fungsi `<img>` dan `.css` untuk mengatur letak dan ukuran sampai sedemikian rupa. Kemudian, di bawahnya terdapat sebuah *button* yang bertuliskan “Buy the pass here” yang dibuat dengan fungsi `<button>` dan dikostumi dengan menggunakan `.css` supaya terlihat seperti gambar 3.117. *Button* tersebut akan menuju ke halaman pembelian tiket. Kemudian, baik *banner* maupun *button* dibungkus dengan sebuah `<section class>` yang nantinya akan dikostumisasi dengan menggunakan `.css` untuk diberikan *background* berwarna *orange*.



Gambar 3.118 Bagian Poster *Room 1*

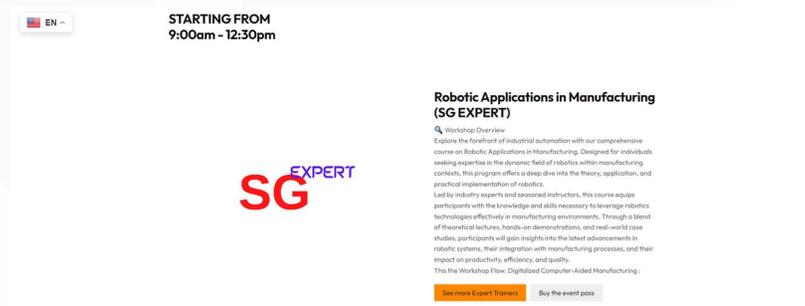
Gambar 3.118 merupakan gambar yang menunjukkan bagian poster kegiatan *Room 1*. Terdapat sebuah judul bertuliskan “Room 1” yang dibuat dengan fungsi `<h2>`. Judul tersebut dibungkus dengan fungsi `<section>` sehingga dapat dikustomisasi di `.css` sampai menjadi sedemikian rupa. Bagian bawahnya terdapat tulisan “Smart Manufacturing & I4.0” yang dibuat dengan menggunakan fungsi `<p>` dan dibungkus dengan fungsi `<section>` juga untuk dikustomisasi di `.css` agar menjadi sedemikian rupa. Kemudian, bagian bawahnya terdapat sebuah poster besar yang diletakkan dengan fungsi `<img>` yang dibungkus dengan `<div>` supaya bisa dikustomisasi di `.css` sampai sedemikian rupa.



Gambar 3.119 Bagian *Gallery Trainers*

Gambar 3.119 merupakan gambar yang menunjukkan bagian foto-foto *trainers* atau pengajar di *workshop* ini. Terdapat sebuah judul yang bertuliskan “TRAINERS” yang dibuat dengan fungsi `<h2>` dan `.css` untuk dikustomisasi. Kemudian, bagian bawah

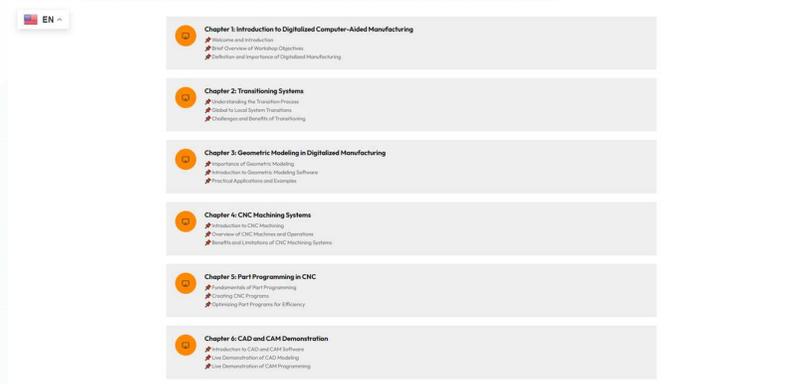
terdapat sebuah gambar foto *trainer* yang dibuat dengan seperangkat JavaScript karena terdapat animasi apabila *cursor mouse* didekatkan ke gambar, ia akan meng-*highlight* gambar tersebut dan menurunkan *opacity* gambar yang lain. Gambar tersebut diletakkan dengan menggunakan fungsi `<img>`. Kemudian, untuk deskripsinya digunakan fungsi `<h3>` dan `<p>` yang dikstumisasi menggunakan `.css` sampai menjadi sedemikian rupa. Terdapat juga fungsi `<a>` karena gambar akan memiliki *link* ke halaman *website trainer* atau LinkedIn *trainer*.



Gambar 3.120 Kelas Pertama untuk *Room 1 Day 1*

Gambar 3.120 merupakan gambar yang menunjukkan deskripsi kelas pertama dari *Room 1 Day 1*. Berdasarkan gambar tersebut, dapat dilihat bahwa terdapat judul, gambar *trainer*, deskripsi topik, dan juga dua *button* yang akan *direct* ke halaman *trainer* dan juga halaman beli tiket. Judul di kanan atas dibuat dengan fungsi `<h2>` dan `.css` untuk dikostumisasi. Kemudian, di bagian bawahnya, terdapat sebuah gambar *trainer* di sebelah kiri yang dibuat dengan menggunakan fungsi `<img>` dan di sebelah kanan terdapat sebuah deskripsi topik yang dibuat dengan menggunakan `<h2>` untuk judul, `<p>` untuk paragraf deskripsi, dan `<button>` untuk dua *button* di bawah. Semuanya dibungkus ke dalam sebuah `<div class>` yang dikostumisasi dengan menggunakan `.css`

supaya berbentuk sedemikian rupa. Untuk memisahkan kiri dan kanan, digunakan fungsi *display: flex*.



Gambar 3.121 Bagian Topik Pembahasan

Gambar 3.121 merupakan gambar yang menunjukkan bagian selanjutnya setelah deskripsi *workshop* kelas pertama, yaitu *list* mengenai pembahasan apa saja di kelas tersebut. Pertama-tama, terdapat sebuah *icon* sebelah kiri yang diletakkan dengan menggunakan fungsi `<i>`, lalu terdapat judul *chapter* yang dibuat dengan fungsi `<h3>`, lalu terdapat fungsi `<p>`, `<ul>`, dan `<li>` untuk menunjukkan poin-poin kegiatan. Semuanya dibungkus ke dalam satu `<div class>` yang nantinya di `.css` akan diberikan sebuah *background* berwarna abu-abu. Lalu, `<div class>` tersebut di *copy* dan *paste* berkali-kali sesuai dengan banyaknya *chapter*.

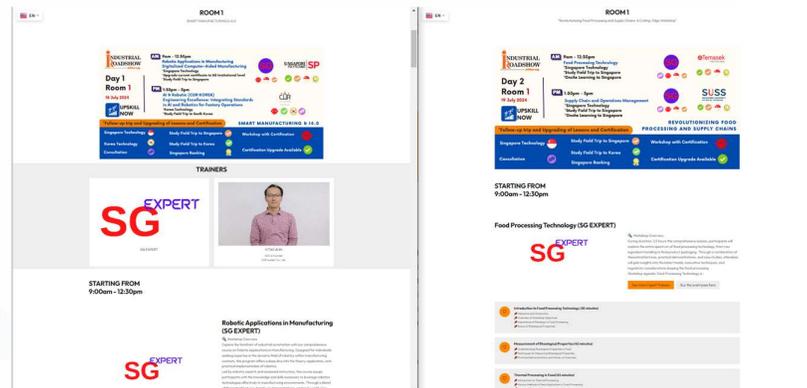
Bagian-bagian selanjutnya dalam halaman ini memiliki desain yang sama dengan gambar 3.119 – 3.121. Hal yang membedakan hanyalah isi kontennya saja, seperti judulnya yang berbeda, *banner*-nya, foto *trainer*-nya, deskripsi kelasnya, *chapter*-nya, dan sebagainya sehingga peserta magang merasa bagian selanjutnya tidak perlu dijelaskan lagi karena banyak hal yang sama dengan bagian sebelumnya dan juga cara membuatnya juga sama.

### 3.3.9.3 Halaman *Technical Workshop Day 2*



Gambar 3.122 Halaman *Day 2* Bagian Atas

Gambar 3.122 merupakan gambar yang menunjukkan halaman *day 2* bagian atas. Bagian ini tidak ada perbedaan sama sekali dengan *day 1* (lihat gambar 3.117). Bahkan, keseluruhan halaman *day 2* juga memiliki kesamaan desain dengan *day 1* hanya saja berbeda konten sehingga cara membuatnya tidak perlu dijelaskan lagi. Namun, terdapat sebuah perbedaan yang menarik untuk dibahas.



Gambar 3.123 Perbedaan antara *Day 1* dan *Day 2*

Gambar 3.123 merupakan sebuah gambar yang menunjukkan perbedaan antara halaman *day 1* dan *day 2*. Gambar sebelah kiri merupakan halaman *day 1*, sedangkan sebelah kanan merupakan halaman *day 2*. Berdasarkan gambar tersebut, dapat

dilihat bahwa perbedaannya di bagian *Room 1* adalah tidak ada *gallery trainers* untuk halaman *day 2*. Hal ini dikarenakan untuk *day 2 room 1*, *trainer*-nya hanya satu sehingga *supervisor* menyuruh peserta magang untuk menghapus *gallery trainer* untuk *day 2* karena dinilai jelek jika hanya ada satu *trainer* saja di *gallery trainer*. Sebaliknya, halaman ini memiliki desain yang sama dengan *day 1* dan yang membedakan hanyalah konten-kontennya.

### 3.3.10 RSS Feeds dan iFrame

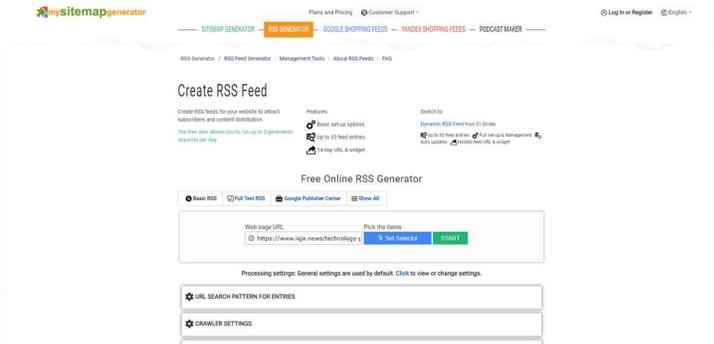
Tugas berikutnya yang diberikan oleh *supervisor* kepada peserta magang adalah mencari cara untuk melakukan RSS Feeds dan juga iFrame. Berikut merupakan uraiannya.

#### 3.3.10.1 RSS Feeds

Secara singkat, RSS Feeds merupakan sebuah tindakan untuk menerima *updates* dari sebuah konten secara reguler atau berkala [6]. Dalam konteks perusahaan, RSS Feeds dibutuhkan oleh perusahaan supaya konten berita yang ada di dalam *website* perusahaan, iiga.news, dapat di-embed ke *website* lain sehingga konten berita tersebut dapat muncul di halaman *website* lain dan terus-menerus *update* secara otomatis apabila ada berita baru. Terdapat dua *tools* yang digunakan, yaitu MySiteMapGenerator dan juga Fuita.

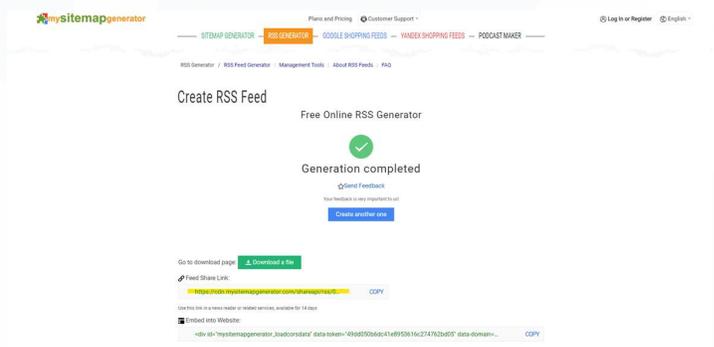
##### 3.3.10.1.1 MySiteMapGenerator

MySiteMapGenerator.com merupakan sebuah *website* (mysitemapgenerator.com) yang dapat melakukan konversi *link* URL *website* menjadi *link* rss.xml.



Gambar 3.124 Tampilan MySiteMapGenerator

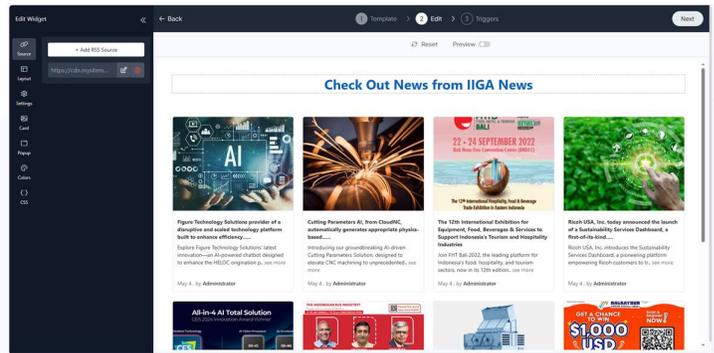
Gambar 3.124 merupakan tampilan dari MySiteMapGenerator. Hal yang perlu dilakukan hanyalah meletakkan *link* URL *website* di bagian *Web Page URL* lalu klik *Start*.



Gambar 3.125 Copy Link RSS

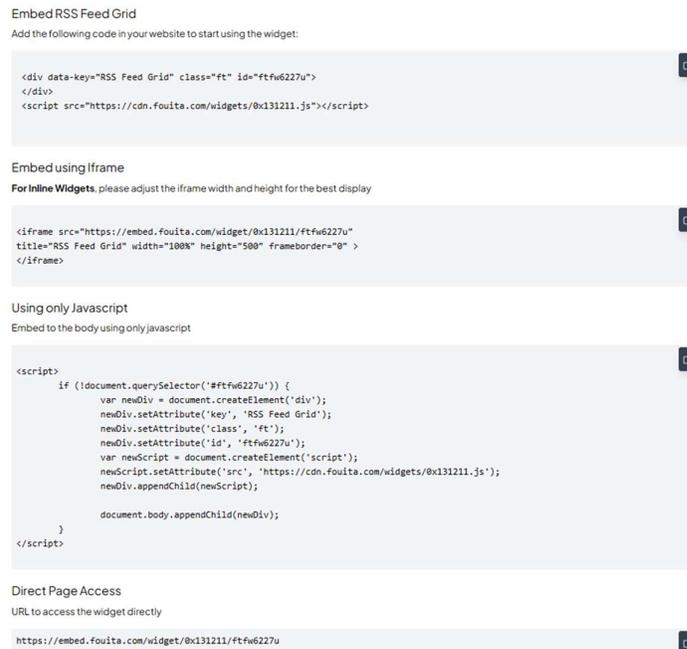
Gambar 3.125 merupakan gambar yang menunjukkan hasil dari proses konversi *link* URL menjadi *link* RSS. Hal yang perlu dilakukan hanyalah klik *Copy* di bagian yang di *highlight* warna kuning lalu *paste link* tersebut ke Fuita nanti.

### 3.3.10.1.2 Fuita



Gambar 3.126 *Preview* RSS Feeds dengan Fouita

Gambar 3.126 merupakan sebuah gambar yang menunjukkan hasil memasukkan *link* RSS dari MySiteMapGenerator ke dalam Fouita. Secara otomatis, Fouita akan membuatkan sebuah *preview* seperti gambar 3.126 sebagai gambaran bagaimana bentuk RSS Feeds jika nanti dipasang ke halaman *website* lain. Fouita memiliki *settings* yang cukup fleksibel sehingga jika *user* tidak mau tampilannya seperti gambar 3.126, *user* dapat mengatur ulang tampilan melalui *settings*.



Gambar 3.127 *Embed Code Fuita*

Gambar 3.127 merupakan gambar yang menunjukkan sebuah kodingan yang berguna untuk nantinya di-embed RSS Feeds tersebut ke halaman *website* lain. Berdasarkan gambar tersebut, terdapat empat cara, yaitu *RSS Feed Grid*, *iFrame*, *JavaScript*, dan *Direct Page Access*. Hal yang paling sering digunakan oleh peserta magang untuk melakukan *embed* RSS Feeds adalah dengan menggunakan metode JavaScript. Caranya adalah tinggal *copy* kodingan JavaScript tersebut dan *paste* sebagai *embed code* di halaman *website*.



Gambar 3.128 Hasil Akhir *Embed* RSS Feed

Gambar 3.128 merupakan hasil akhir dari *embed* dengan menggunakan RSS Feeds. Hasil tersebut dibuat di halaman kosong sebuah *website* dengan meng-*embed* kodingan JavaScript sebelumnya.

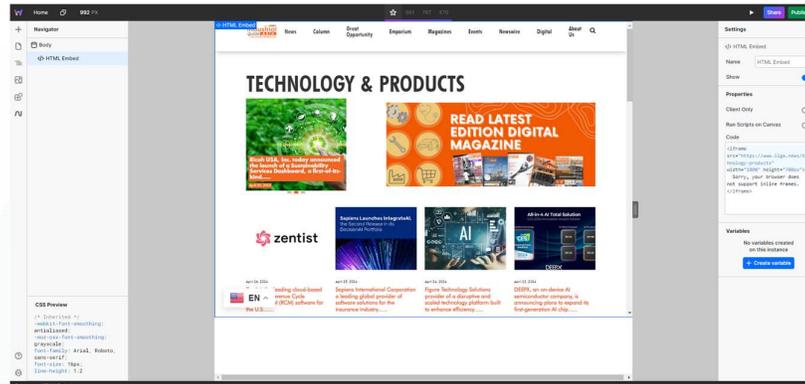
### 3.3.10.2 iFrame

Metode iFrame adalah sebuah metode untuk memasukkan satu halaman *website* ke dalam halaman *website* lainnya. Dalam hal ini, alasan *supervisor* ingin melakukan hal tersebut adalah *supervisor* ingin memasukkan *website* perusahaan, iiga.news, ke dalam *website* lain sehingga di dalam *website* tersebut terdapat satu buah halaman yang menunjukkan *website* iiga.news.

```
<iframe src="https://www.iiga.news/technology-products" width="100%" height="700px">
  Sorry, your browser does not support inline frames.
</iframe>
```

Gambar 3.129 Kodingan iFrame

Gambar 3.129 merupakan sebuah gambar yang menunjukkan sebuah kodingan iFrame yang nantinya akan di-*embed* ke halaman *website* lain. Hasilnya adalah sebagai berikut.



Gambar 3.130 Hasil *Embed* iFrame

Gambar 3.130 merupakan hasil dari *embed code* iFrame. Berdasarkan gambar tersebut, digunakanlah sebuah Saas penyedia layanan *website* kodong yang kemudian di-*testing embed code* iFrame-nya di situ.

### 3.3.11 Proyek Mempersatukan *Partner* Bisnis Perusahaan

Pada akhir April 2024, *supervisor* mengundang beberapa karyawannya yang dari Indonesia khususnya divisi *sales / marketing* dan juga IT untuk membahas sebuah proyek baru perusahaan secara *offline* di Jakarta. Pertama-tama, *supervisor* menceritakan bahwa perusahaan ini dan juga perusahaan media internasional di seluruh dunia banyak yang mengalami kekalahan saing dengan perusahaan *public relations*. Hal ini dikarenakan perusahaan *public relations* dapat memunculkan artikel-artikel yang dimilikinya ke berbagai *website* (dengan kata lain, mereka dapat *embed* artikel mereka ke *website* lain), sedangkan perusahaan media hanya bisa memunculkan artikelnya di *website*-nya sendiri saja. Dengan adanya hal tersebut, perusahaan *public relations* lebih diminati karena apabila sebuah perusahaan ingin melakukan promosi, perusahaan *public relations* memiliki jangkauan yang lebih luas dari pada perusahaan media. Hal yang lebih parahnyanya lagi adalah beberapa perusahaan *public relations* melakukan *embed code* di halaman *website* perusahaan media secara gratis. Hal tersebut

sama saja seperti menumpang iklan di *website* orang lain secara gratis. Sebagai contoh, salah satu perusahaan *public relations* yang cukup dekat dengan Orange Media Group adalah PR Newswire. Perusahaan tersebut merupakan perusahaan *public relations* yang bergerak dalam bidang artikel berita. Mereka melakukan *embed* artikel-artikelnya di *website* iiga.news secara gratis. Tidak hanya di iiga.news, PR Newswire juga meng-*embed* artikel-artikelnya di *website* perusahaan media lainnya. Tanpa sadar, hal tersebut telah merugikan perusahaan-perusahaan tersebut. Hal ini dikarenakan sebuah perusahaan apabila ingin melakukan promosi, mereka tentunya akan lebih memilih perusahaan *public relations* yang jelas bisa menjangkau lebih luas karena perusahaan *public relations* bisa meng-*embed* artikel-artikelnya di *website* lain sehingga jika seandainya PR Newswire sudah meng-*embed* artikel mereka ke lima *website*, hal tersebut berarti secara tidak langsung perusahaan tersebut sudah melakukan promosi ke setidaknya lima media dengan hanya membayar ke PR Newswire saja dan tidak membayar ke perusahaan media tersebut. Sebuah solusi yang muncul dari masalah ini adalah dengan menghilangkan PR Newswire dari iiga.news dan melakukan *embed-code* artikel iiga.news menggunakan RSS Feeds ke *website* lain juga. Nyatanya, hal tersebut tidak semudah yang dibayangkan. Pertama, apabila perusahaan menghilangkan sepenuhnya PR Newswire dari iiga.news, perusahaan kemungkinan juga akan mengalami kerugian entah bagaimana caranya. Kedua, jika kita ingin melakukan *embed code* dengan RSS Feeds untuk artikel berita iiga.news, ke *website* manakah harus kita *embed* mengingat tidak mungkin perusahaan *embed* ke *website* media dan *partner website* iiga.news jika dibandingkan dengan PR Newswire tentunya akan lebih banyak PR Newswire. Berdasarkan hal tersebut, solusi yang muncul adalah dengan menggabungkan perusahaan-perusahaan media internasional untuk bersatu membuat sebuah program yang seperti perusahaan-perusahaan *public relations* melalui sebuah *website* yang berisi gabungan artikel-artikel *partner* perusahaan. *Website* tersebut nantinya akan

di-embed ke *website-website* lain sama seperti apa yang PR Newswire lakukan. Dengan demikian, proyek baru berdasarkan masalah ini adalah untuk membuat sebuah *website* yang bisa menggabungkan artikel-artikel berita dari berbagai perusahaan media di berbagai belahan dunia. Tugas utama dari divisi IT adalah untuk merancang bangun *website* ini yang diberi nama Trade PR, sedangkan untuk divisi *sales / marketing* bertugas untuk mengumpulkan perusahaan media di seluruh dunia. Dengan ini, diharapkan perusahaan-perusahaan media tidak hanya Orange Media Group, tetapi seluruh perusahaan media di seluruh dunia dapat bersatu untuk bersaing melawan perusahaan *public relations* supaya perusahaan-perusahaan media tetap bertahan hidup. Proyek ini belum sepenuhnya berjalan karena masih banyak yang harus didiskusikan. Dalam divisi IT sendiri masih banyak perdebatan untuk melakukan proyek ini, seperti apa *database* yang terbaik untuk hal ini, akankah kita menggunakan Saas CMS saja, dan masih banyak lagi. Maka, tugas yang diberikan kepada peserta magang adalah melakukan *review* terhadap beberapa *website* Saas dan CMS yang kira-kira cocok untuk melakukan hal ini. Tiga hal yang diperlukan oleh perusahaan adalah sebuah *website* dengan *back-end* yang mudah dipahami, bisa menunggah artikel berita, dan bisa melakukan pembayaran untuk menjadi *member*. Berdasarkan keperluan tersebut, masih menjadi perdebatan apakah akan digunakan *website* dengan Saas yang menyediakan CMS atau koding *website* dari awal beserta *back-end*-nya.

### **3.3 Kendala yang Ditemukan**

Berdasarkan tugas dan uraian kerja magang yang sudah dijelaskan di bagian sebelumnya, terdapat beberapa kendala yang ditemukan sebagai berikut.

#### **3.3.1 Kurangnya Perhatian dari *Supervisor***

Seperti yang sudah dijelaskan di bagian-bagian sebelumnya bahwa *supervisor* peserta magang, Miss Nicole, selain *supervisor* peserta magang, beliau juga merupakan *marketing director*. Beliau memiliki peran yang sangat penting di perusahaan sehingga waktu untuk mengawasi pekerjaan

dari peserta magang nyaris tidak ada. Hal tersebut merupakan sebuah hambatan karena terkadang peserta magang merasa bingung dengan tugas yang beliau berikan.

### **3.3.2 Kurangnya Pemahaman Bahasa Pemrograman *Website***

Bahasa pemrograman *website*, seperti HTML, CSS, PHP, dan JavaScript memang sudah pernah dipelajari selama perkuliahan, tetapi terdapat beberapa hal baru, khususnya PHP dan JavaScript yang tidak dipelajari selama perkuliahan. Hal tersebut menjadi hambatan karena peserta magang harus mempelajari lebih lanjut mengenai hal-hal baru tersebut.

### **3.3.3 Kurangnya Pemahaman akan *Tools* yang diberikan**

*Tools-tools* yang diberikan, seperti Labrica, VBout, Hexomatic, dan sebagainya merupakan *tools* yang belum pernah peserta magang sentuh sama sekali sehingga peserta magang harus mempelajari terlebih dahulu mengenai *tools* tersebut.

## **3.4 Solusi atas Kendala yang Ditemukan**

Berdasarkan kendala yang ditemukan selama kerja magang, terdapat beberapa solusi yang ditemukan sebagai berikut.

### **3.3.4.1 Menghubungi *Supervisor***

Salah satu kendala yang dihadapi adalah kurangnya perhatian dari *supervisor*. Hal tersebut dapat diatasi dengan sering-sering menghubungi *supervisor* melalui *chat* di Ring Central. Meskipun balasnya terkadang lama sekali karena *supervisor* sibuk, tetapi paling tidak peserta magang mengetahui apakah pekerjaannya benar atau harus direvisi.

### **3.3.4.2 Mencari *Tutorial***

Kendala berikutnya yang dialami adalah kurangnya pemahaman akan bahasa pemrograman *website* dan juga *tools* yang diberikan. Hal tersebut dapat diatasi salah satu dengan cara mencari *tutorial* di internet.

#### **3.3.4.3 Bertanya ke *Customer Service***

Selain mencari *tutorial* di internet, peserta magang disarankan oleh *supervisor* untuk menanyakan cara menggunakan *tools* yang diberikan kepada *customer service tools* tersebut. Hal ini dikarenakan beberapa *tools* yang diberikan sangat minim informasinya di internet dan juga *customer service* pastinya lebih paham akan produknya sendiri sehingga mereka bisa membantu peserta magang untuk menggunakan *tools* tersebut sesuai dengan kebutuhan.

#### **3.3.4.4 Menggunakan ChatGPT**

Selain mencari *tutorial* di internet, menggunakan ChatGPT juga merupakan salah satu solusi untuk mengatasi masalah kurangnya pemahaman peserta magang terhadap bahasa pemrograman *website*. Apabila terjadi *error*, peserta bisa langsung *copy* dan *paste* error tersebut ke ChatGPT dan ChatGPT akan mencoba untuk memperbaikinya. Selain itu, apabila peserta magang tidak tahu kodingannya dan tidak ada sumber di internet, ChatGPT dapat digunakan untuk membuat kodingannya.

