

## **BAB 3**

### **PELAKSANAAN KERJA MAGANG**

Pelaksanaan kerja magang dilakukan selama 6 bulan atau minimal 800 jam kerja dimulai dari tanggal 14 Juni 2021 sampai 13 Desember 2021. Disebabkan pandemi COVID-19, maka absensi karyawan dilakukan secara *hybrid*, WFH dan WFO, dengan ketentuan WFO 2 atau 3 hari setiap minggunya tergantung dari *shift* yang sudah ditentukan oleh perusahaan.

#### **3.1 Kedudukan dan Organisasi**

Kedudukan adalah sebagai anggota tim divisi Finance Project Analyst yang berperan dalam mendukung dan menyediakan berbagai kebutuhan divisi *finance* di Kawan Lama Group. Baik dalam koordinasi antar divisi *finance* dengan divisi IT hingga terjun dalam pembuatan sistem itu sendiri.

Didukung oleh *supervisor* Ibu Martha Liliana selaku Head of Finance Project Analyst Team serta rekan-rekan satu divisi, *meeting* secara rutin dilaksanakan baik dengan divisi IT, pengguna maupun sesama anggota tim sendiri untuk mendapatkan bimbingan dan diberi arahan dalam membangun sistem yang sesuai dengan permintaan pengguna.

#### **3.2 Tugas yang Dilakukan**

Tugas yang dilakukan selama bekerja di Kawan Lama Group melingkupi berbagai aspek mulai dari Front End Developer hingga Google Developer. Tugas pertama yang dikerjakan adalah membuat tampilan UI/UX yang baru untuk *website* IVP (Integrated Vendor Portal). *Prototype* UI/UX dibangun dari nol menggunakan *figma*, kemudian *prototype* tersebut dibangun ke dalam bentuk *website* dengan menggunakan bahasa *HTML*, *CSS* dan *Javascript*. Sedangkan tugas kedua sampai kelima adalah membangun berbagai sistem otomatis menggunakan Google Workspace dengan bahasa *Javascript*. Tugas-tugas tersebut adalah pembangunan *web app* CSV to Spreadsheet Converter, Automatic Send Email from Spreadsheet, Sheet to CSV Converter dan Sheet to Sheet.

Diluar dari tugas yang diberikan, beberapa tanggung jawab yang dimiliki juga meliputi hal-hal berikut :

- Menemukan kekurangan dan *bug* dari *website* yang sudah dibangun.
- Belajar secara mandiri menggunakan Figma dan Google Workspace.
- Berkolaborasi dengan divisi IT maupun sesama anggota tim untuk berdiskusi tentang masalah yang ditemukan.
- Mempresentasikan kemajuan proyek setidaknya sekali dalam seminggu.

### 3.3 Uraian Pelaksanaan Magang

Selama melaksanakan kerja magang ada 5 proyek berbeda yang sudah diselesaikan, yaitu :

1. UI/UX *Website* IVP.
2. CSV to Spreadsheet Web Converter.
3. Automatic Send Email from Spreadsheet.
4. Sheet to CSV Web Converter.
5. Sheet to Sheet Web App.

Berikut adalah timeline pengerjaan proyek :

Tabel 3.1. Minggu 1 Pelaksanaan Magang

Minggu Ke -	Pekerjaan yang dilakukan
1	<ul style="list-style-type: none"> <li>• Pengenalan perusahaan dan struktur organisasi.</li> <li>• Penjelasan detail proyek yang akan dikerjakan.</li> <li>• <i>Setup software</i> yang diperlukan.</li> <li>• Diskusi dengan pengguna mengenai <i>requirement</i>.</li> <li>• Finalisasi <i>sketch</i> UI untuk <i>prototype</i>.</li> <li>• Pembuatan <i>prototype</i> menggunakan Figma.</li> </ul>

Tabel 3.2. Minggu 2 - 5 Pelaksanaan Magang

Minggu Ke -	Pekerjaan yang dilakukan
2	<ul style="list-style-type: none"> <li>• Membangun <i>prototype</i> UI/UX menggunakan figma untuk halaman <i>Login, Information Center, Company List</i> dan <i>E-Invoice</i>.</li> </ul>
3	<ul style="list-style-type: none"> <li>• Finalisasi <i>prototype</i> UI/UX di figma.</li> <li>• Desain tambahan untuk <i>pop up</i> dan <i>sidebar</i>.</li> <li>• Revisi beberapa bagian tampilan UI.</li> <li>• <i>Review</i> dengan pengguna untuk finalisasi desain tampilan.</li> </ul>
4	<ul style="list-style-type: none"> <li>• Koding tampilan baru <i>website</i> IVP berdasarkan <i>prototype</i> yang sudah dibuat untuk halaman <i>Login, Information Center, Company List</i> dan <i>E-Invoice</i></li> </ul>
5	<ul style="list-style-type: none"> <li>• Melanjutkan pembangunan tampilan baru <i>web</i> IVP.</li> <li>• Penambahan fitur baru di halaman <i>E-Invoice</i>.</li> <li>• <i>fixing bug</i>.</li> <li>• Merapikan code.</li> </ul>

Tabel 3.3. Minggu 6 - 8 Pelaksanaan Magang

Minggu Ke -	Pekerjaan yang dilakukan
6	<ul style="list-style-type: none"> <li>• Melanjutkan pembangunan tampilan baru <i>web IVP</i> untuk bagian <i>pop up</i> dan <i>table</i>, beberapa penyesuaian tampilan seperti warna dan penambahan efek untuk memperindah tampilan.</li> <li>• <i>Review</i> halaman <i>E-Invoice</i> secara keseluruhan.</li> <li>• <i>fixing bug</i>.</li> <li>• <i>Review update</i> dengan pengguna.</li> </ul>
7	<ul style="list-style-type: none"> <li>• Melanjutkan pembangunan tampilan baru <i>web IVP</i> pada bagian fitur <i>preview PDF</i> dan <i>pop up</i>.</li> <li>• <i>fixing bug</i> dan <i>UX</i> halaman <i>E-Invoice</i>.</li> <li>• Perubahan desain pada beberapa bagian di halaman <i>E-Invoice</i>.</li> </ul>
8	<ul style="list-style-type: none"> <li>• <i>fixing bug</i> pada halaman <i>E-Invoice</i> untuk melempar data ke <i>table</i> melalui <i>input</i> pada <i>pop up</i>.</li> <li>• Merapikan <i>Figma</i>.</li> <li>• Memperbaiki pada halaman <i>E-Invoice</i>.</li> <li>• <i>Review web page E-Invoice</i> secara keseluruhan.</li> </ul>

Tabel 3.4. Minggu 9 - 13 Pelaksanaan Magang

Minggu Ke -	Pekerjaan yang dilakukan
9	<ul style="list-style-type: none"> <li>• Finalisasi halaman <i>web E-Invoice</i>.</li> <li>• <i>Review web page E-Invoice</i> secara keseluruhan.</li> <li>• <i>Online course</i>, belajar mandiri tentang penggunaan Google Workspace dan Apps Script.</li> </ul>
10	<ul style="list-style-type: none"> <li>• <i>Review web IVP</i> yang sudah <i>dideploy</i> oleh tim <i>Front End</i>.</li> <li>• <i>Online course</i>, belajar mandiri tentang penggunaan Google Workspace dan Apps Script.</li> </ul>
11	<ul style="list-style-type: none"> <li>• <i>Online course</i>, belajar mandiri tentang penggunaan Google Workspace dan Apps Script.</li> <li>• Pengarahan dan persiapan proyek selanjutnya (CSV to Spreadsheet Converter).</li> <li>• Pembangunan CSV to Spreadsheet Converter <i>web app</i> menggunakan Google Workspace.</li> </ul>
12	<ul style="list-style-type: none"> <li>• Pembangunan CSV to Spreadsheet Converter <i>web app</i> menggunakan Google Workspace.</li> <li>• Perbaikan <i>layout website IVP</i>.</li> </ul>
13	<ul style="list-style-type: none"> <li>• <i>Testing</i> dan penambahan fitur CSV to Spreadsheet Converter <i>web app</i>.</li> </ul>

Tabel 3.5. Minggu 14 - 16 Pelaksanaan Magang

Minggu Ke -	Pekerjaan yang dilakukan
14	<ul style="list-style-type: none"> <li>• Pembangunan <i>responsive web</i> IVP untuk halaman <i>Login</i> dan <i>Company List</i>.</li> <li>• <i>Fixing bug</i> pada <i>web</i> IVP.</li> <li>• Melanjutkan pembangunan <i>CSV to Spreadsheet Converter web app</i> menggunakan <i>Google Workspace</i>.</li> </ul>
15	<ul style="list-style-type: none"> <li>• Perubahan alur sistem dan pembuatan database <i>CSV to Spreadsheet Converter web app</i> .</li> <li>• Menambahkan fitur <i>last update</i> dan <i>set trigger</i> pada <i>CSV to Spreadsheet Converter web app</i> .</li> <li>• <i>Testing CSV to Spreadsheet Converter web app</i> dengan <i>dummy data</i>.</li> </ul>
16	<ul style="list-style-type: none"> <li>• <i>Testing CSV to Spreadsheet Converter web app</i> dengan data asli.</li> <li>• <i>fixing bug</i> pada <i>CSV to Spreadsheet Converter web app</i> .</li> <li>• Menambahkan <i>function deleteTrigger()</i> pada <i>program CSV to Spreadsheet Converter web app</i>.</li> <li>• Merapikan code.</li> <li>• <i>Review</i> performa dan hasil <i>output website</i>.</li> </ul>

Tabel 3.6. Minggu 17 - 19 Pelaksanaan Magang

Minggu Ke -	Pekerjaan yang dilakukan
17	<ul style="list-style-type: none"> <li>• Menambahkan fitur <i>input delimiter</i> pada CSV to Spreadsheet Converter <i>web app</i> .</li> <li>• <i>fixing bug</i> pada CSV to Spreadsheet Converter <i>web app</i> .</li> <li>• <i>Testing</i> CSV to Spreadsheet Converter <i>web app</i> dengan data asli.</li> <li>• <i>Review proyek</i> CSV to Spreadsheet Converter <i>web app</i>.</li> </ul>
18	<ul style="list-style-type: none"> <li>• Membuat UI, mengubah semua <i>field input</i> menjadi <i>required</i>, memberikan <i>alert</i> jika <i>field input</i> masih kosong dan memberikan informasi "<i>Success</i>" atau "<i>Failed</i>" saat pengguna menekan tombol <i>convert</i> pada CSV to Spreadsheet Converter <i>web app</i> .</li> <li>• <i>Embed</i> CSV to Spreadsheet Converter <i>web app</i> ke Google Sites.</li> <li>• <i>Testing</i> CSV to Spreadsheet Converter <i>web app</i>.</li> <li>• Pengarahan proyek selanjutnya (Send automatic Email from Spreadsheet).</li> </ul>
19	<ul style="list-style-type: none"> <li>• Koding program Send Automatic Email from Spreadsheet pada Spreadsheet Site Master Maintenance, Cost Center dan GL Account menggunakan Google Workspace.</li> </ul>

Tabel 3.7. Minggu 20 - 21 Pelaksanaan Magang

Minggu Ke -	Pekerjaan yang dilakukan
20	<ul style="list-style-type: none"> <li>• Koding Sheet to CSV Converter <i>web app</i>.</li> <li>• Koding program untuk membuat <i>sumarry</i> dari <i>selected range</i> di spreadsheet menggunakan Google Workspace dan <i>query SQL</i>.</li> </ul>
21	<ul style="list-style-type: none"> <li>• Koding program Sheet to Sheet <i>web app</i> menggunakan Google Workspace.</li> <li>• <i>Testing</i> Sheet to Sheet <i>web app</i>.</li> <li>• Automatic Send Email from Spreadsheet.</li> </ul>

Pada minggu pertama diperkenalkan lingkungan perusahaan dan dengan rekan-rekan satu tim, serta instalasi *software* yang diperlukan seperti Figma, Visual Studio Code dan XAMPP. Proyek pertama yang akan dikerjakan adalah membuat UI/UX yang baru untuk website IVP (Integrated Vendor Portal) yang meliputi 4 halaman, yaitu *Login*, *Company List*, *Information Center* dan *E-Invoice*. *Meeting* dengan pengguna dilakukan untuk mendapatkan detail *requirement proyek*, kemudian berdasarkan hasil *meeting* tersebut dibuatlah *sketch* UI untuk halaman *Login*, *Company List* dan *Information Center website IVP*. *Meeting* kemudian dilakukan kembali untuk finalisasi dan persetujuan *sketch*, dan dilanjut dengan membuat *prototype* keempat halaman menggunakan figma di minggu kedua hingga minggu ketiga.

Minggu keempat mengawali development UI/UX dalam bentuk koding, *prototype* dari figma yang sudah disetujui diimplementasikan ke dalam bentuk *website* dari sisi *client side*. Pengerjaan ini memakan waktu enam minggu dari minggu keempat hingga minggu kesepuluh sampai UI/UX keempat halaman diselesaikan, termasuk menggabungkan UI dengan *back end*, *testing*, dan *fixing bug*.

Minggu kesepuluh hingga sebelas adalah waktu untuk belajar mandiri tentang *Javascript* dan Google Workspace untuk persiapan proyek kedua,



dilanjutkan dengan pengarahan proyek dan pembangunan CSV to Spreadsheet Converter *web app* pada hari terakhir minggu kesebelas. Fungsi dari *web* ini adalah untuk mengkonversi data dari CSV ke Spreadsheet secara otomatis pada waktu yang sudah ditentukan. Proyek kedua ini memakan waktu selama 7 minggu dari minggu kesebelas hingga minggu kedelapan belas. Namun pada minggu keempat belas pengerjaan sempat tertunda kurang lebih selama 3 atau 4 hari karena ada penambahan fitur pada proyek sebelumnya, yaitu membuat *responsive web* untuk *website IVP* pada halaman *Login* dan *Company List*.

Diakhir minggu kedelapan belas, proyek kedua selesai dan diberikan pengarahan untuk proyek ketiga, yaitu Automatic Send Email from Spreadsheet. Proyek ini berfungsi untuk menambahkan waktu *approval* pada *sheet* dan mengirim *email* secara otomatis kepada divisi selanjutnya untuk memberitahu bahwa divisi sebelumnya sudah melakukan *approval* pada *sheet*, serta untuk meminta divisi selanjutnya melakukan *approval* dibagian miliknya. Proyek ini memakan waktu selama satu minggu dari akhir minggu kedelapan belas hingga ke hari terakhir diminggu kesembilan belas.

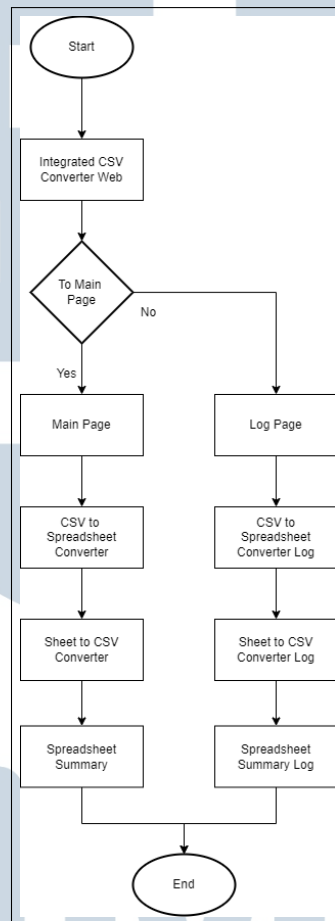
Pada minggu kedua puluh diberikan pengarahan singkat untuk proyek keempat yaitu membangun Sheet to CSV Converter *web app*. Alur programnya tidak jauh beda dengan proyek kedua sehingga hanya memakan waktu pengerjaan selama empat hari. hari terakhir di minggu kedua puluh digunakan untuk pengarahan proyek ke lima yaitu membuat Sheet to Sheet *web app*. *Web* tersebut berfungsi untuk membuat spreadsheet baru berdasarkan data-data yang dipilih pengguna dari *source sheet* aslinya. Sehingga *output* dari program ini adalah data yang hanya ingin dilihat oleh pengguna pada spreadsheet yang baru. Pengerjaan proyek kelima ini memakan waktu satu minggu mulai dari akhir minggu kedua puluh hingga akhir minggu kedua puluh satu.

UI/UX *website IVP* adalah proyek pertama yang dikerjakan dan dikembangkan bersama dengan rekan tim *front end* lainnya. Automatic Send Email from Spreadsheet adalah proyek ketiga yang programnya ada di dalam *sheet* yang berisikan data-data milik kantor, proyek ini dikerjakan secara individu. Sedangkan proyek kedua, keempat dan kelima digabungkan menjadi 1 *web* yang terintegrasi. Ketiga *web app* tersebut digabung ke dalam satu alamat *web* utama dengan menggunakan Google Sites. Sehingga ketika *web* dibuka, pengguna dapat memilih fitur yang ingin dipakai sesuai dengan kebutuhan. Pembangunan dan penggabungan ketiga proyek dikerjakan secara individu. Berikut adalah rincian pengerjaan, hasil akhir tampilan *web* dan fitur yang dimiliki oleh *web* CSV Converter yang sudah

diintegrasikan.

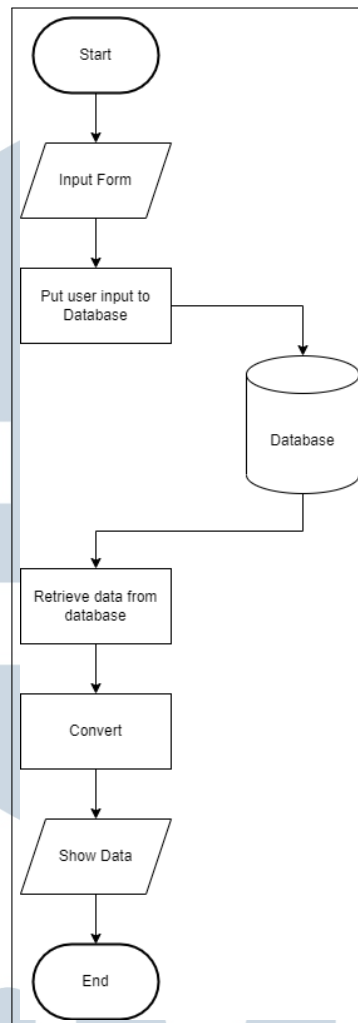
### 3.3.1 Perancangan

Berikut adalah *flowchart* setiap fitur beserta penjelasannya :



Gambar 3.1. Struktur Integrated Web CSV Converter

Gambar 3.1 merupakan struktur dari *web* yang sudah diintegrasikan. *Web* ini mengintegrasikan 3 *web app* yang sebelumnya terpisah yaitu CSV to Spreadsheet Converter, Sheet to CSV Converter dan Sheet to Sheet. *Integrated web* ini memiliki 2 halaman yaitu *Home Page* untuk menampilkan fitur-fitur yang tersedia dan *Log page* untuk cek *last update file* yang *diconvert*.



Gambar 3.2. Sistem CSV to Spreadsheet Converter

Gambar 3.2 merupakan *flowchart* dari fitur CSV to Spreadsheet Converter. *Web* akan menampilkan 6 kolom *input* yang harus diisi oleh pengguna, berikut adalah hal yang harus *diinput* disetiap kolom :

1. Link CSV  
Link CSV dari Google Drive yang ingin *diconvert*.
2. Delimiter  
Batas yang digunakan untuk memisahkan data di dalam CSV.
3. Link Spreadsheet  
Link Google Spreadsheet untuk menampilkan hasil *output* konverter.

#### 4. Nama Sheet

*Sheet* di dalam Google Spreadsheet untuk menampilkan hasil dari CSV yang sudah *diconvert*.

#### 5. Sheet *diupdate* setiap

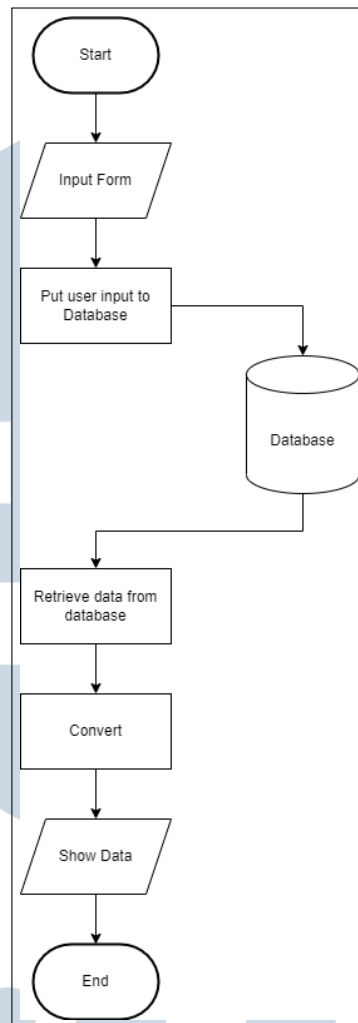
Memilih jangka waktu untuk otomatis update CSV ke spreadsheet secara berkala.

#### 6. Deskripsi File

Keterangan tambahan. Deskripsi hanya akan disimpan di *database* sebagai dokumentasi dan tidak ditampilkan pada *spreadsheet output*.

Setelah pengguna mengisi semua kolom *input*, data kemudian akan disimpan di dalam *database*. *Database* yang digunakan adalah spreadsheet untuk menampung *input user* dalam bentuk *string*. Sistem kemudian akan menarik kembali data yang baru didaftarkan tersebut untuk menampilkan hasil akhir dari CSV yang sudah *diconvert*. Hasilnya dapat dilihat dalam bentuk spreadsheet di dalam *URL spreadsheet* yang diberikan pengguna.





Gambar 3.3. Sistem Sheet to CSV Converter

Gambar 3.3 merupakan *flowchart* dari fitur Sheet to CSV Converter. *Web* akan menampilkan 6 kolom *input* yang harus diisi oleh pengguna, berikut adalah hal yang harus *diinput* disetiap kolom :

1. Link Google Spreadsheet  
Link Google Spreadsheet dari Google Drive yang ingin *diconvert*.
2. Nama Sheet  
Sheet di dalam Google Spreadsheet yang ingin *diconvert* ke CSV.
3. Delimiter  
Batas yang digunakan untuk memisahkan data di dalam CSV.

#### 4. Link Folder *Output*

Link Folder Google Drive untuk menyimpan hasil *output* yang sudah *diconvert*.

#### 5. CSV diupdate setiap

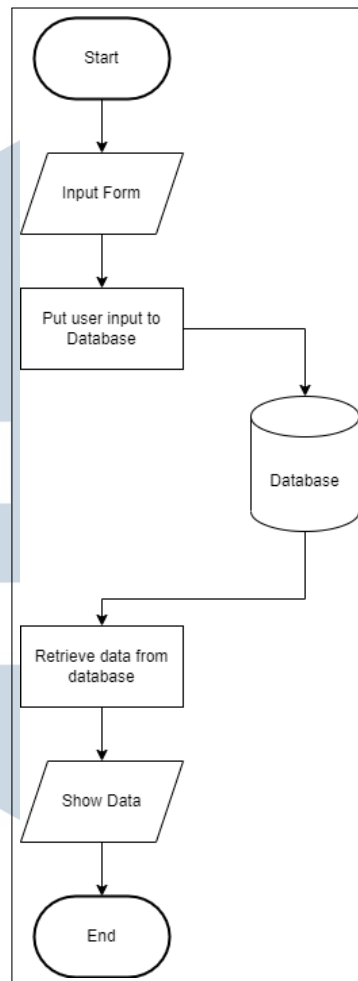
Memilih jangka waktu untuk otomatis update CSV secara berkala.

#### 6. Deskripsi File

Keterangan tambahan. Deskripsi hanya akan disimpan di *database* sebagai dokumentasi dan tidak ditampilkan pada *spreadsheet output*.

Sama seperti sistem CSV to Spreadsheet Converter, setelah semua kolom input diisi, data kemudian akan disimpan di dalam *database*. Sistem kemudian akan menarik kembali data baru tersebut dan mengkonversinya ke dalam file baru dengan format CSV. Hasil *outputnya* dapat dilihat di dalam direktori folder Google Drive yang sudah pengguna berikan linknya pada saat input.





Gambar 3.4. Sistem Sheet to Sheet

Gambar 3.4 merupakan *flowchart* dari fitur Sheet to Sheet. *Web* akan menampilkan 8 kolom *input* yang harus diisi oleh pengguna, berikut adalah hal yang harus *diinput* disetiap kolom :

1. Link Google Spreadsheet Data

Link Google Spreadsheet berisi data yang ingin diambil beberapa datanya.

2. Nama *Sheet* Data

Nama *Sheet* dari Google Spreadsheet berisi data yang ingin diambil beberapa datanya.

3. *Range* Data

*Range* di dalam *Sheet* yang ingin diambil datanya. Contoh: A1:C3.

#### 4. Link Google Spreadsheet Tujuan

Link Google Spreadsheet untuk menampilkan data yang sudah dipilih dari Google Spreadsheet data.

#### 5. Nama *Sheet* Data

*Sheet* di dalam Google Spreadsheet untuk menampilkan data yang sudah dipilih dari Google Spreadsheet data.

#### 6. *Cell* tujuan

*Starting point* atau *cell* yang diinginkan untuk menampilkan data. Contoh : A1.

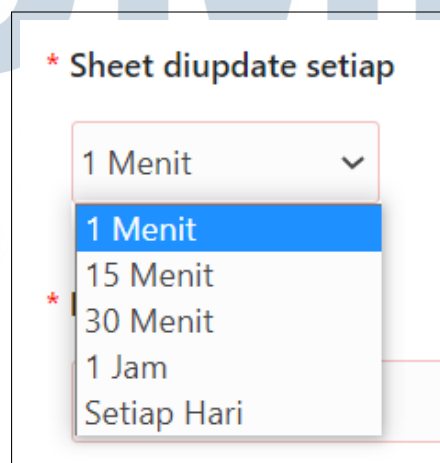
#### 7. *Sheet* diupdate setiap

Memilih jangka waktu untuk otomatis *update sheet* secara berkala.

#### 8. Deskripsi File

Keterangan tambahan. Deskripsi hanya akan disimpan di *database* sebagai dokumentasi dan tidak ditampilkan pada spreadsheet output.

Sama seperti kedua sistem sebelumnya, setelah semua kolom input diisi, data kemudian akan disimpan di dalam *database*. Sistem kemudian akan menarik kembali data baru tersebut dan menampilkan data yang telah dipilih dari spreadsheet data ke dalam spreadsheet tujuan. Hasil *output*nya dapat dilihat di dalam *sheet* dari spreadsheet yang sudah ditentukan pengguna pada saat *input*.



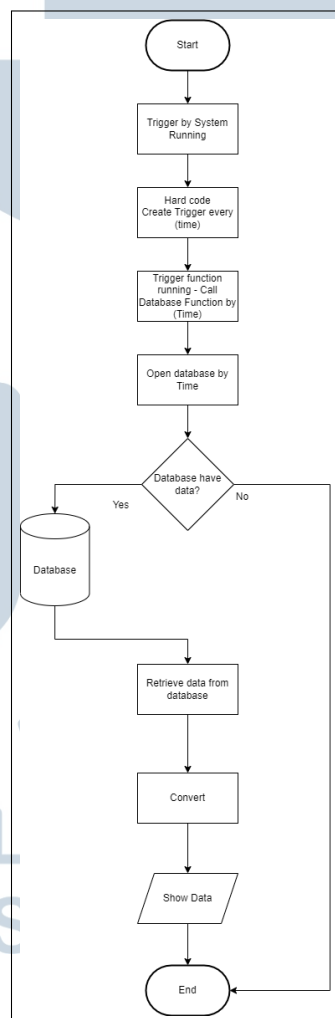
Gambar 3.5. Pilihan trigger



Gambar 3.5 merupakan pilihan *trigger* yang dapat dipilih oleh pengguna. Ada 5 pilihan waktu yang berbeda, yaitu :

- Setiap 1 Menit
- Setiap 15 Menit
- Setiap 30 Menit
- Setiap 1 Jam
- Setiap 1 Hari

Jika pengguna memilih *trigger* setiap 1 hari, maka data tersebut akan *diupdate* tepat tengah malam setiap harinya.



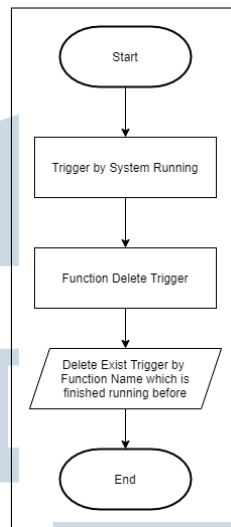
Gambar 3.6. Sistem update file secara berkala menggunakan time trigger

Apps Script menyediakan fitur *trigger* pada platform mereka. Dengan *trigger*, program dapat menjalankan sebuah *function* secara berkala berdasarkan waktu yang sudah ditentukan. Ketiga fitur pada *integrated web* juga memiliki *trigger* di dalam programnya, fungsinya adalah untuk memanggil *function* untuk menarik data dari *database* dan mengupdate isi dari *sheet* ataupun CSV secara otomatis berdasarkan waktu yang sudah ditentukan pengguna.

Gambar 3.6 menunjukkan alur *trigger* yang diimplementasikan pada ketiga fitur. Pertama *trigger by system* atau *trigger* yang dibuat menggunakan bawaan dari Apps script akan memanggil *function createTrigger()* di dalam program (*hard code trigger*). *Function createTrigger()* yang dipanggil tersebut akan memanggil *function* untuk menarik data dari *database* berdasarkan waktu yang sudah ditentukan pengguna. Misalnya, pengguna menentukan waktu update setiap 15 menit sekali, maka *function* yang akan dipanggil adalah *function* database 15 menit, yang berisi data yang semua datanya memiliki waktu *trigger* yang sama, yaitu 15 menit.

Kondisi pada alur sistem mengindikasikan bahwa *function* memeriksa isi di dalam *database*, jika *database* mempunyai data, maka sistem akan berjalan untuk mengkonversi atau mengupdate *sheet* sesuai dengan fungsinya, namun jika tidak ada data, *trigger* akan tetap berjalan, dan tidak akan ada *error message* dalam programnya. *Output* setiap hasil dapat dilihat di direktori yang sama yang sudah ditentukan pengguna pada saat melakukan *input* data.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 3.7. Function Delete Exist Trigger

Gambar 3.7 adalah alur *function deleteTrigger()* bekerja. Pertama-tama *trigger* bawaan dari Apps Script akan memanggil *function deleteTrigger()* pada program (*hard code deleteTrigger()*), kemudian *function* yang dipanggil tersebut akan mencari dan menghapus *trigger* berdasarkan nama *function* yang sudah dijalankan. Misalnya, *function hard code DeleteTrigger1Menit()* dijalankan oleh *trigger* dari Apps Script, maka *function* akan mencari dan menghapus *trigger* yang menjalankan *function* bernama Database1Menit().

Ketiga fitur memiliki *function deleteTrigger()* di dalam programnya. Fungsinya adalah untuk menghapus *trigger* yang sudah dieksekusi dan tidak lagi terpakai. Setiap *scripts* mempunyai batas kapasitas untuk menampung jumlah *trigger* yang dimiliki, jika sudah penuh, maka *script* tidak akan bisa membuat *trigger* yang baru. Maka dari itu dibuatlah *function deleteTrigger()* untuk mencegah kepenuhan kapasitas tersebut.

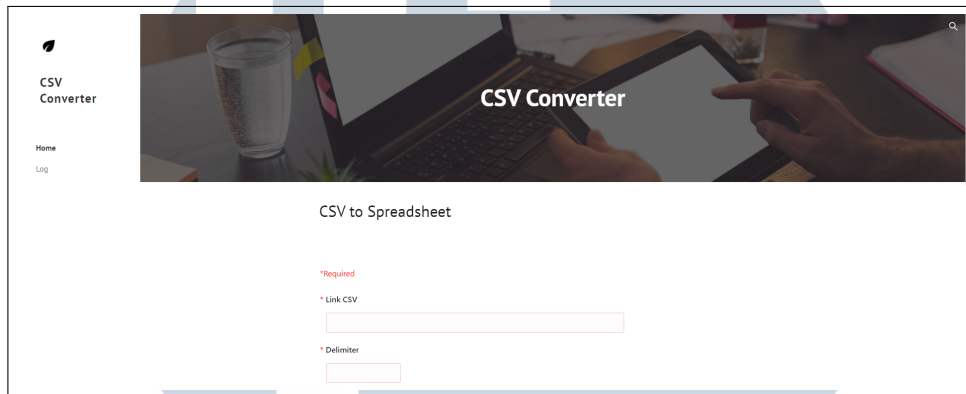
Berikut adalah waktu *function deleteTrigger()* aktif :

- Delete *trigger* dengan nama "Database1Menit" setiap 5 menit sekali.
- Delete *trigger* dengan nama "Database15Menit" setiap 1 jam sekali.
- Delete *trigger* dengan nama "Database30Menit" setiap 1 jam sekali.
- Delete *trigger* dengan nama "Database1Jam" setiap 4 jam sekali.
- Delete *trigger* dengan nama "DatabaseEveryday" setiap seminggu sekali.

DatabaseEverday akan dihapus setiap seminggu sekali pada hari senin tepat tengah malam.

### 3.3.2 Implementasi

Berikut adalah tampilan *website* dari Integrated CSV Converter :



Gambar 3.8. Halaman Utama Integrated CSV Converter Website

Gambar 3.8. menampilkan halaman utama website konverter dengan 2 *side menu* disebelah kiri, yaitu Home dan Log.

UMMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## CSV to Spreadsheet

---

**\*Required**

**\* Link CSV**

**\* Delimiter**

**\* Link Spreadsheet**

**\* Name Sheet**  
Harap sesuaikan dengan nama sheet yang akan dituju

\*Jika tidak diisi hasil otomatis akan berada di Sheet1

**\* Sheet diupdate setiap**

**\* Deskripsi File**

Gambar 3.9. Fitur CSV to Spreadsheet Converter

Gambar 3.9 menampilkan fitur pertama yang terdapat di website yaitu CSV to Spreadsheet Converter. Pada fitur ini pengguna dapat mengkonversi data transaksi di dalam file CSV menjadi data yang di tampilkan di dalam sheet. Ada 6 *field* yang harus di *input*, yaitu :

1. Link CSV

Link CSV dari Google Drive yang ingin *diconvert*.

2. Delimiter

Batas yang digunakan untuk memisahkan data di dalam CSV.

3. Link Spreadsheet

Link Google Spreadsheet untuk menampilkan hasil *output* konverter.

#### 4. Nama Sheet

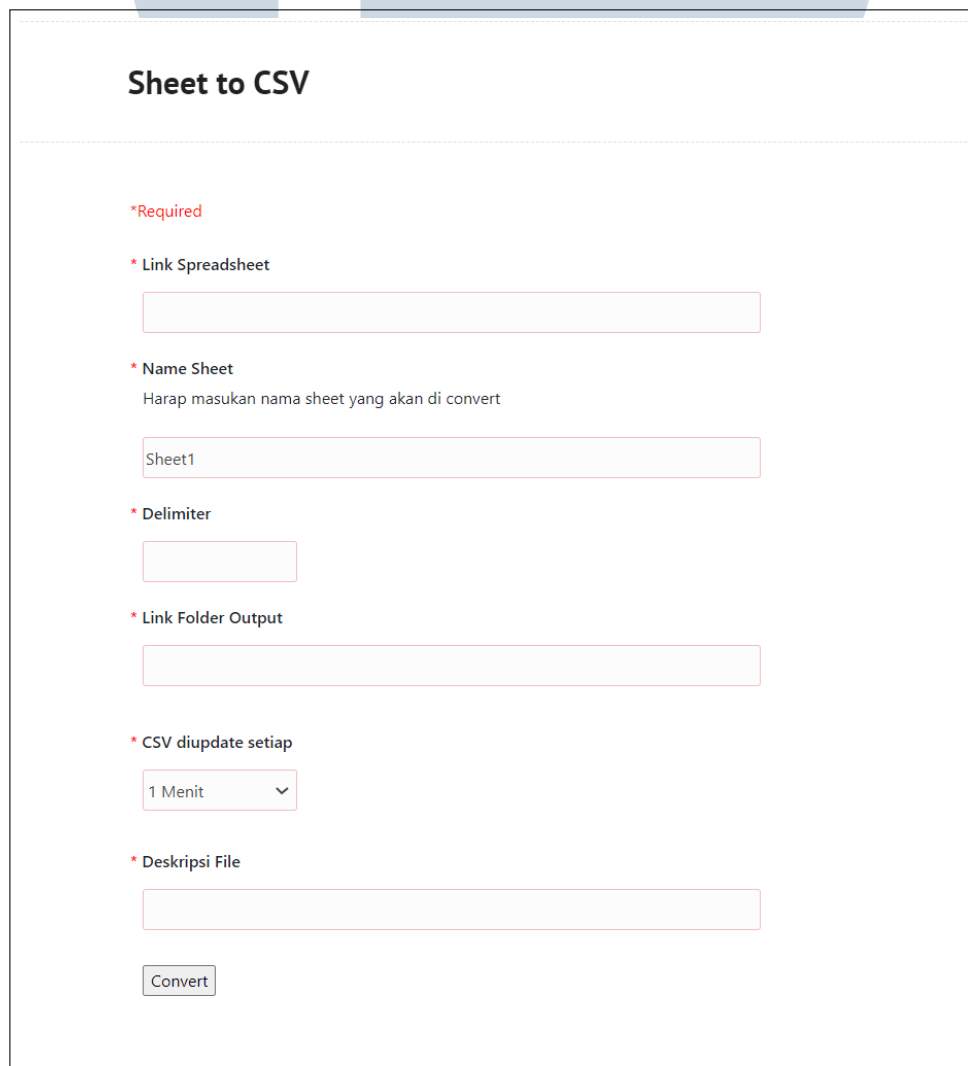
*Sheet* di dalam Google Spreadsheet untuk menampilkan hasil dari CSV yang sudah *diconvert*.

#### 5. Sheet diupdate setiap

Memilih jangka waktu untuk otomatis update CSV ke spreadsheet secara berkala.

#### 6. Deskripsi File

Keterangan tambahan. Deskripsi hanya akan disimpan di *database* sebagai dokumentasi dan tidak ditampilkan pada *spreadsheet output*.



The image shows a web interface titled "Sheet to CSV". It contains several input fields, each with a red asterisk indicating it is required. The fields are: "Link Spreadsheet" (text input), "Name Sheet" (text input with the value "Sheet1" and a subtext "Harap masukan nama sheet yang akan di convert"), "Delimiter" (text input), "Link Folder Output" (text input), "CSV diupdate setiap" (dropdown menu with "1 Menit" selected), and "Deskripsi File" (text input). At the bottom, there is a "Convert" button.

Gambar 3.10. Fitur Sheet to CSV Converter

Gambar 3.10 menampilkan fitur kedua yang terdapat di website yaitu Sheet to CSV Converter. Pada fitur ini pengguna dapat mengkonversi data transaksi di dalam sheet menjadi file CSV. Ada 6 *field* yang harus di *input*, yaitu :

1. Link Google Spreadsheet  
Link Google Spreadsheet dari Google Drive yang ingin *diconvert*.
2. Nama Sheet  
Sheet di dalam Google Spreadsheet yang ingin *diconvert* ke CSV.
3. Delimiter  
Batas yang digunakan untuk memisahkan data di dalam CSV.
4. Link Folder *Output*  
Link Folder Google Drive untuk menyimpan hasil *output* yang sudah *diconvert*.
5. CSV diupdate setiap  
Memilih jangka waktu untuk otomatis update CSV secara berkala.
6. Deskripsi File  
Keterangan tambahan. Deskripsi hanya akan disimpan *didatabase* sebagai dokumentasi dan tidak ditampilkan pada *spreadsheet output*.

U M N  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

## Sheet to Sheet

**\*Required**

**\* Link Spreadsheet Data**

**\* Name Sheet Data**  
Harap masukan nama sheet yang akan di convert

**\* Range Data**  
Harap masukan Range data yang ingin diambil

**\*Spreadsheet Tujuan**

**\* Name Sheet Tujuan**  
Harap masukan nama sheet yang akan di convert

**\*Cell Tujuan**  
Harap masukan Cell tempat data ingin ditampilkan

**\* Sheet diupdate setiap**

**\* Deskripsi File**

Gambar 3.11. Fitur Sheet to Sheet

Gambar 3.11 menampilkan fitur ketiga yang terdapat di website yaitu Sheet to Sheet. Pada fitur ini pengguna dapat menyalin data yang diinginkan ke Sheet ataupun Spreadsheet lainnya. Ada 8 *field* yang harus di *input*, yaitu :



1. Link Google Spreadsheet Data

Link Google Spreadsheet berisi data yang ingin diambil beberapa datanya.

2. Nama *Sheet* Data

Nama *Sheet* dari Google Spreadsheet berisi data yang ingin diambil beberapa datanya.

3. *Range* Data

*Range* di dalam *Sheet* yang ingin diambil datanya. Contoh: A1:C3.

4. Link Google Spreadsheet Tujuan

Link Google Spreadsheet untuk menampilkan data yang sudah dipilih dari Google Spreadsheet data.

5. Nama *Sheet* Data

*Sheet* di dalam Google Spreadsheet untuk menampilkan data yang sudah dipilih dari Google Spreadsheet data.

6. *Cell* tujuan

*Starting point* atau *cell* yang diinginkan untuk menampilkan data. Contoh : A1.

7. *Sheet* diupdate setiap

Memilih jangka waktu untuk otomatis *update sheet* secara berkala.

8. Deskripsi File

Keterangan tambahan. Deskripsi hanya akan disimpan *didatabase* sebagai dokumentasi dan tidak ditampilkan pada spreadsheet output.

U M M N  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA

**CSV to Spreadsheet Convert Log**

CSV to Spreadsheet Database						
CSV Link	CSV ID	Spreadsheet Output	Sheet Name	Delimiter	Description	Update
<a href="https://drive.google.com/file/d/10G106XXKw460rR/">https://drive.google.com/file/d/10G106XXKw460rR/</a>		<a href="https://docs.google.com/spreadsheets/">https://docs.google.com/spreadsheets/</a>	Spreadsheet		Test CSV to Spreadsheet	Thu Jan 09 2022 20:48:14 GMT+0900

1 Menu 25 Menu 30 Menu 1 Jan Export

Gambar 3.12. Halaman Log CSV to Spreadsheet Converter

Gambar 3.12 menampilkan section log dari CSV to Spreadsheet di halaman Log yang dapat diakses dari *side bar*. Pengguna dapat melihat input yang di masukan sebelumnya dan melihat waktu terakhir data transaksi di update.

**Sheet to CSV Convert Log**

Sheet to CSV Database					
CSV Link	Sheet Name	Delimiter	Folder Link	Description	Update
<a href="https://docs.google.com/spreadsheets/">https://docs.google.com/spreadsheets/</a>	toCSV		<a href="https://drive.google.com/drive/folders/1t-CeVtRd/">https://drive.google.com/drive/folders/1t-CeVtRd/</a>	Testing Sheet to	Thu Jan 09 2022 20:48:20 GMT+0900 (Eastern Standard

1 Menu 25 Menu 30 Menu 1 Jan Export

Gambar 3.13. Halaman Log Sheet to CSV Converter

Gambar 3.13 menampilkan section log dari Sheet to CSV di halaman Log yang dapat diakses dari *side bar*. Pengguna dapat melihat input yang di masukan sebelumnya dan melihat waktu terakhir CSV di update.

Spreadsheet Source	Sheet Source	Range	Spreadsheet Tujuan	Sheet Tujuan	Cell Tujuan	Description	Update
<a href="https://docs.google.com/spreads...">https://docs.google.com/spreads...</a>	Spreadsheet	A2:C7	<a href="https://docs.google.com/spreads...">https://docs.google.com/spreads...</a>	Summary	D4	Test	Thu Jan 06 2022 20:46

Gambar 3.14. Halaman Log Sheet to Sheet

Gambar 3.12 menampilkan section log dari Sheet to Sheet di halaman Log yang dapat diakses dari *side bar*. Pengguna dapat melihat input yang di masukan sebelumnya dan melihat waktu terakhir *Range* yang telah di pilih di update.

Halaman *log* berfungsi agar pengguna dapat memeriksa apakah *input* sudah berhasil masuk ke dalam *database* atau belum, serta untuk melihat data-data apa saja yang sudah didaftarkan sebelumnya dan sedang berjalan saat ini. Keterangan *last update* pada *database* juga ditambahkan agar pengguna mengetahui kapan waktu terakhir konversi atau *sheet* diupdate oleh program, sehingga lebih mudah juga untuk pengguna mengetahui kapan waktu *update* selanjutnya. Fitur *last update* ini hanya ditampilkan di dalam *database* dan bukan untuk ditampilkan pada hasil *output* data yang telah dikonversi.

### 3.4 Kendala dan Solusi yang Ditemukan

#### 3.4.1 Kendala yang Ditemukan

Selama proses kerja magang ada beberapa kendala yang ditemukan, diantaranya adalah :

1. Perubahan atau penambahan *requirement* yang mendadak.
2. Adanya *miss communication* antara tim dengan pengguna.

3. Sumber dokumentasi dan referensi yang sedikit, serta tidak banyak rekan yang menguasai Google Workspace.

#### **3.4.2 Solusi Dari Kendala yang Di Temukan**

Berikut adalah solusi dari setiap permasalahan yang dilakukan selama melaksanakan kerja magang :

1. Melakukan *meeting* untuk mendapatkan kesepakatan akhir terhadap requirement yang dibutuhkan.
2. Memberikan laporan secara rutin tentang *progress* serta berdiskusi mengenai kendala yang ditemukan selama *deployment*.
3. Mencari jawaban permasalahan lewat *stack overflow* maupun blog yang berhubungan dengan permasalahan yang dicari.

