

BAB 3

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Organisasi

Dalam perusahaan Kalbe Consumer Health, terdapat beberapa tingkatan dalam divisi IT dan *Digital Transformation*. Divisi IT dan *Digital Transformation* terdiri dari *Head of Digital Transformation, System Architect, Business Analyst, Application Developer, Oracle Functional, Oracle Technical, Network and Infra, IT Support Staff, Application Developer Intern, and Technical Oracle Intern*.

Kedudukan penulis selama masa magang adalah sebagai *Technical Oracle Support Intern* dengan mentor dari *Oracle Technical* bernama T. Ryan Adi Pangestu. Dalam masa program magang, *mentor* akan memberikan tugas baru ketika terdapat masalah dalam report yang user bicarakan kepada *mentor*.

3.2 Tugas yang Dilakukan

Tugas yang dikerjakan selama melakukan program magang adalah sebagai berikut:

1. Mempelajari *Module Pembelajaran Oracle*.
2. Mempelajari membuat *report* dan menggunakan *oracle apps*.
3. Membuat template dan *program reports* dalam *oracle apps*.
4. Memperbaiki template dan *program reports* dalam *oracle apps*.
5. Menganalisis reports yang dihasilkan dalam *oracle apps*.
6. Modifikasi template dan *program reports* dalam *oracle apps*.
7. Membuat program notifikasi *email* menggunakan *oracle apps*.
8. Membuat *program* untuk *upload file CSV* kedalam database menggunakan *oracle apps*.
9. Mengupload *program* dari *database* awal ke *database* yang lain.

3.3 Uraian Pelaksanaan Magang

Berikut adalah *detail* terkait kegiatan-kegiatan yang dilakukan selama praktik kerja magang.

Tabel 3.1. Uraian pelaksanaan praktik kerja magang

| Minggu ke | Pekerjaan yang dilakukan |
|--------------------------------|---|
| 1 | <ul style="list-style-type: none"> • Melakukan instalasi <i>program</i> dan <i>setup</i> jaringan <i>WiFi</i> untuk mengakses <i>database</i> • <i>Meeting</i> dengan <i>mentor</i> untuk membahas mengenai tugas yang akan diberikan dalam <i>program</i> magang • Pembelajaran mandiri menggunakan <i>module</i> 1 - 10 untuk <i>database</i> dan <i>oracle</i> |
| 2 | <ul style="list-style-type: none"> • Membuat PPT mengenai pengertian P2P dan O2C • Mempelajari mengenai <i>PL SQL Cursor</i> dalam <i>query</i> • Mempelajari <i>statement</i> dasar dari <i>query</i> secara mandiri |
| 3 | <ul style="list-style-type: none"> • Mempelajari <i>statement</i> dan <i>function query</i> secara mandiri • Mempelajari cara membuat <i>template report</i> dengan menggunakan <i>Word</i> • Mempelajari membuat <i>RTF file</i> untuk menjalankan <i>report</i> dengan menggunakan <i>report builder</i> • Mempelajari cara membuat dan menjalankan <i>reports</i> menggunakan <i>oracle apps</i> |
| 4 | <ul style="list-style-type: none"> • Mempelajari cara membuat <i>template report</i> dengan menggunakan <i>Word</i> • Memperbaiki <i>template reports</i> dari <i>reports Budget Capex dan Receiving Vs Assets</i> • Mempelajari mengenai <i>API</i> dalam <i>query</i> |
| Lanjut pada halaman berikutnya | |

Tabel 3.1 Uraian pelaksanaan praktik kerja magang (lanjutan)

| Minggu ke | Pekerjaan yang dilakukan |
|--------------------------------|---|
| 5 | <ul style="list-style-type: none"> • Menerapkan <i>API</i> dalam <i>database</i> dengan menggunakan <i>procedure</i> • Mempelajari mengenai <i>personalized</i> dalam <i>oracle apps</i> • Membuat <i>personalized</i> dalam <i>oracle apps</i> |
| 6 | <ul style="list-style-type: none"> • Menerapkan <i>API</i> dalam <i>database</i> dengan menggunakan <i>procedure</i> • Membuat <i>template report Ledger Activity 2</i> • Membuat <i>RTF file</i> untuk <i>report Ledger Activity 2</i> • Membuat <i>program</i> untuk menjalankan <i>report Ledger Activity 2</i> dalam <i>oracle apps</i> |
| 7 | <ul style="list-style-type: none"> • Membuat <i>procedure</i> untuk memvalidasi <i>invoice</i> • Menganalisa dan mengupdate <i>report Ledger Activity 2</i> menjadi <i>Ledger Activity 3</i> • Mengupdate <i>database 8001</i> dengan cara menambahkan <i>table</i> dan <i>procedure</i> dari <i>database 8000</i> |
| 8 | <ul style="list-style-type: none"> • Melakukan <i>meeting</i> mengenai aplikasi pembayaran dengan <i>API</i> untuk menghubungkan <i>database</i> dari aplikasi dengan <i>database</i> dari <i>oracle</i> • Menganalisa dan mengupdate <i>report Ledger Activity 3</i> • Mengupdate <i>database 8000</i> dengan cara menambahkan <i>view</i> dari <i>database 8001</i> • Mengupdate <i>template report Ledger Activity 3</i> |
| 9 | <ul style="list-style-type: none"> • Mempelajari mengenai <i>Rest Service Oracle</i> • Menganalisa dan mengupdate <i>report Cash Flow</i> • Menganalisa dan mengupdate <i>report COGM</i> • Menganalisa dan mengupdate <i>report Data Budget Capex</i> |
| Lanjut pada halaman berikutnya | |

Tabel 3.1 Uraian pelaksanaan praktik kerja magang (lanjutan)

| Minggu ke | Pekerjaan yang dilakukan |
|--------------------------------|---|
| 10 | <ul style="list-style-type: none"> • Menganalisa dan mengupdate <i>report ap tax bupot</i> • Mendokumentasikan <i>privilage</i> dari <i>user</i> baru dalam <i>database oracle</i> • Menganalisa dan mengupdate <i>report costing journal subledger process</i> |
| 11 | <ul style="list-style-type: none"> • Menganalisa dan mengupdate <i>report costing journal subledger process</i> • Membuat <i>program</i> untuk mengirimkan <i>notifikasi email</i> jika <i>storage</i> dalam <i>database</i> menyentuh persentase tertentu |
| 12 | <ul style="list-style-type: none"> • Menganalisa dan mengupdate <i>report data unbudget capex</i> • Menganalisa dan mengupdate <i>report costing shipment vs invoice report</i> • Menganalisa dan mengupdate <i>report costing journal subledger</i> • Menganalisa dan mengupdate <i>report cash flow detail</i> |
| 13 | <ul style="list-style-type: none"> • Menganalisa dan mengupdate <i>report cash flow detail</i> • Menganalisa dan mengupdate <i>report budget capex</i> menjadi <i>report budget capex 2</i> • Membuat <i>program</i> untuk mengupload <i>file CSV</i> dari <i>oracle apps</i> menuju ke <i>database oracle</i> dan menunjukkan hasil <i>report</i> dalam bentuk <i>html</i> |
| 14 | <ul style="list-style-type: none"> • Membuat <i>program</i> untuk mengupload <i>file CSV</i> dari <i>oracle apps</i> menuju ke <i>database oracle</i> dan menunjukkan hasil <i>report</i> dalam bentuk <i>html</i> • Mengupdate <i>program notifikasi email</i> dari <i>oracle apps</i> • Menganalisa dan mengupdate <i>report generate</i> faktur pajak • Menganalisa dan mengupdate <i>report ledger activity 3</i> |
| Lanjut pada halaman berikutnya | |

Tabel 3.1 Uraian pelaksanaan praktik kerja magang (lanjutan)

| Minggu ke | Pekerjaan yang dilakukan |
|-----------|---|
| 15 | <ul style="list-style-type: none"> • Menganalisa dan mengupdate <i>report ledger activity 3</i> • Menganalisa dan mengupdate <i>report ledger activity 2</i> • Menganalisa dan mengupdate <i>report unbudgeted capex</i> • Menganalisa dan mengupdate <i>procedure</i> untuk <i>upload file</i> dari <i>website</i> ke <i>database</i> menggunakan <i>API</i> • Membuat <i>personalized</i> dalam <i>invoice batches</i> untuk <i>database 8001 dan 8000</i> |

3.3.1 Mempelajari Mengenai Program

Dalam pelaksanaan tugas pembuatan *program upload CSV* dalam *oracle apps*, langkah pertama yang dilakukan adalah melakukan pembelajaran mengenai *program* yang akan dibuat. Pembelajaran mengenai pembuatan *program* dilakukan dengan bantuan *mentor*. Pembelajaran mengenai alur *program* dipelajari dengan menggunakan *procedure Upload CSV Resource Cost* sebagai referensi untuk membuat *program upload CSV*.

Procedure Upload CSV Resource Cost merupakan sebuah *procedure* dalam *database* yang berfungsi untuk *mengupload CSV Resource Cost* dari *file CSV* kedalam *database oracle*. Cara kerja *procedure Upload CSV Resource Cost* adalah

- Melakukan pengecekan apakah *procedure* sedang diakses atau tidak
- Mengambil *file CSV* dari *FND LOBS*
- Melakukan *insert* ke dalam *table history file*
- Memproses *file CSV* ke dalam *table database*
- Mengambil *parameter data* untuk menjalankan proses *upload*
- Melakukan *insert data* ke dalam *table staging file*
- Membuat *ouput log* untuk melihat *data* yang telah di *upload* dalam *report* dengan *html*

3.3.2 Proses Pembuatan Program

Proses pembuatan *program* dimulai dengan membuat *pseudocode* untuk *program* yang akan dibuat. *Pseudocode* dibuat dengan tujuan untuk menentukan poin-poin yang perlu dibuat dalam *program*. Berikut *pseudocode* dari *program*:

```
1 [ declare
2 cursor xys is
3 select ....
4 where status = 'U';
5
6 L_CCID NUMBER;
7
8 BEGIN
9
10 FOR MY IN XYS
11 LOOP
12
13 IF MY.ATTRIBUTE1 OR MY.ATTRIBUTE2 OR MY.ATTRIBUTE3 OR MY.
    ATTRIBUTE4 IS NULL THEN
14
15     UPDATE .... SET MESSAGE = Cannot update Null Attribute , STATUS
    = F
16     WHERE COA = MY.COA;
17
18 ELSE
19
20     ---GET CCID
21     BEGIN
22     SELECT CODE_COMBINATION_ID INTO L_CCID
23     WHERE CONCATENATED_SEGMENTS = MY.COA
24     EXCEPTION WHEN OTHERS THEN
25         L_CCID := 0;
26     END;
27
28     IF L_CCID IS NULL THEN
29         UPDATE .... SET MESSAGE = Cant find CCID, STATUS = F
30         WHERE COA = MY.COA;
31     else
32
33         update XSFL_UPDATE_CF_ATTRIBUTE set ccid = L_CCID, , STATUS
    = S
34         whehere coa = my.coa;
```

```

35
36     commit;
37
38     update gl_code_combinations set att1 ..... att5
39     where code_combination_id = l_ccid;
40
41
42 end if;
43 END LOOP;
44
45
46 END;
47 ]

```

Kode 3.1: Pseudocode program upload CSV

Program dibuat dengan menggunakan Oracle SQL. Program upload CSV dibuat dengan cara membuat procedure kedalam database dan memanggil procedure kedalam oracle apps sehingga oracle apps akan memproses file CSV dan menghasilkan database.

Program upload CSV dimulai dengan melakukan cek terhadap program jika program sedang diakses atau tidak. Hal ini dilakukan untuk mencegah agar program tidak berjalan lebih dari 1 kali dalam proses upload file.

```

1 [
2 BEGIN
3 V_USER_ID := P_USER_ID;
4
5     --Cek apakah conccurent sedang di akses atau tidak
6     SELECT COUNT(1)
7         INTO V_CEK
8     FROM APPS.XSFL_PROSES_UPLOAD_TEMP_2;
9
10    IF V_CEK = 0 THEN
11
12        BEGIN
13            INSERT INTO APPS.XSFL_PROSES_UPLOAD_TEMP_2
14                (
15                    PROSES
16                    , USER_ID
17                )
18            VALUES
19                (
20                    1

```

```

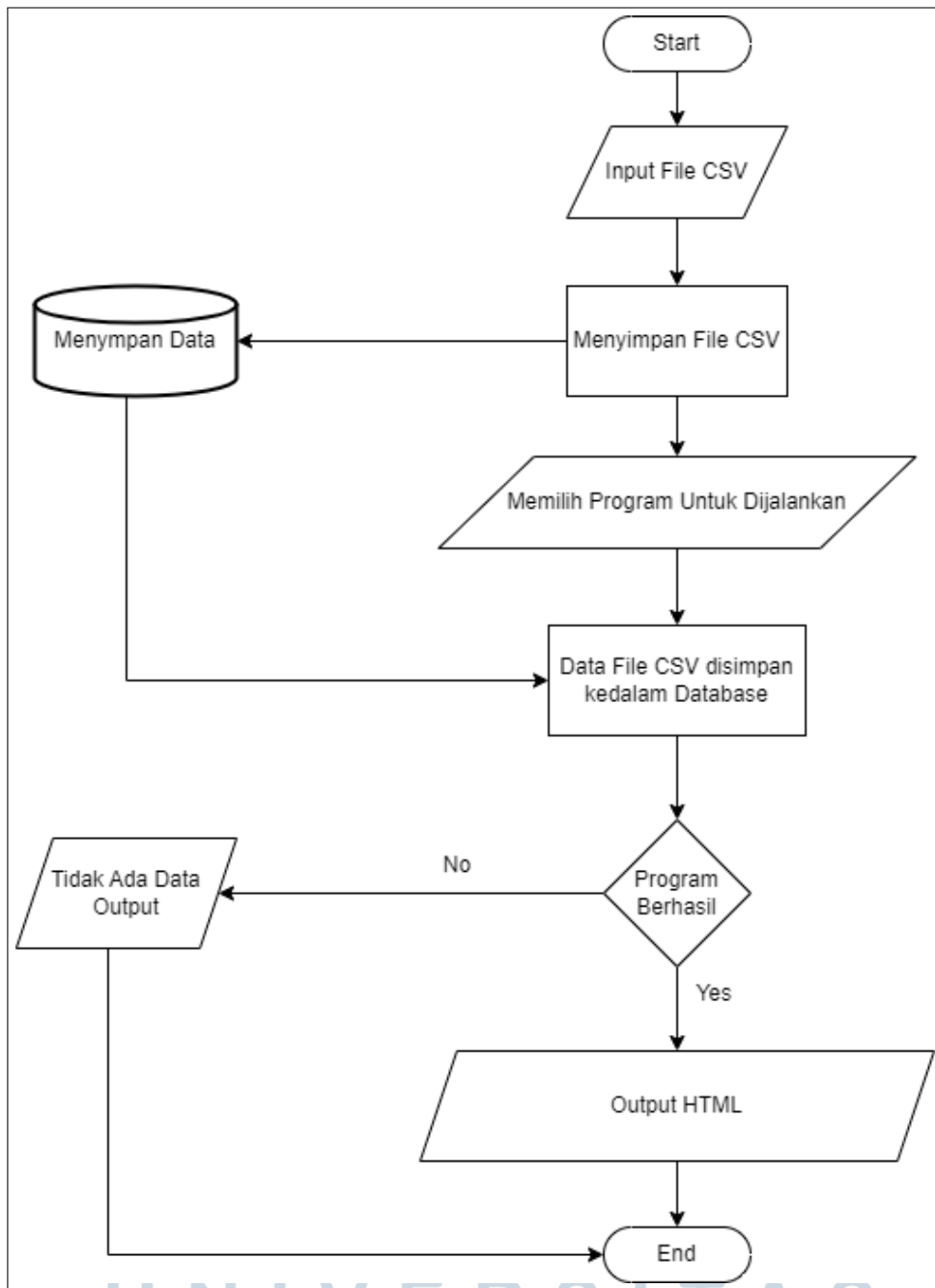
21         , V_USER_ID
22     );
23
24     COMMIT;
25
26     EXCEPTION
27     WHEN OTHERS THEN
28         ROLLBACK;
29         FND_FILE.PUT_LINE (FND_FILE.OUTPUT, 'Gagal insert
30 ke table temp user session');
31         DELETE FROM APPS.XSFL_PROSES_UPLOAD_TEMP_2;
32         COMMIT;
33         RAISE;
34         P_RETCODE := 2;
35     END;

```

Kode 3.2: Code Program Upload CSV

Berikut merupakan *flowchart* dari *program upload file CSV*:

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.1. Flowchart Program

Lalu melakukan *insert* kedalam *table* penampung yang bernama *XSFL UPDATE CSV* untuk mengambil *data* dari *table* CSV kedalam *table* penampung.

1 [
 2
 3]

Kode 3.3: Code Program Upload CSV

Lalu *program* akan melakukan *loop* untuk memproses *data* dari *table* penampung dan *looping* untuk memproses *parameter* yang digunakan dalam *procedure* untuk mengupload *data CSV* ke dalam *database*.

```

1 [
2 IF (V_STATUS = 'S' AND V_CEK_FILE > 0) THEN
3     FND_FILE.PUT_LINE (FND_FILE.OUTPUT, 'File ' || V_FILE_NAME || '
4         sudah di upload, silahkan upload file yang baru' );
5     P.RETCODE := 2;
6     RAISE_APPLICATION_ERROR(-20000, '');
7
8
9 ELSIF (V_STATUS = 'N' AND V_CEK_FILE > 0) THEN
10
11 RETURN;
12
13 ELSE
14 BEGIN
15 INSERT INTO XSFL_UPDATE_CSV
16     (
17         FILE_ID
18         , FILE_NAME
19         , UPLOAD_DATE
20         , CREATED_BY
21         , CREATION_DATE
22         , STATUS
23     )
24 VALUES
25     (
26         P_FILE_ID
27         , V_FILE_NAME
28         , V_UPLOAD_DATE
29         , V_USER_ID
30         , SYSDATE
31         , 'N'
32     );
33 COMMIT;
34 EXCEPTION WHEN OTHERS THEN
35 ROLLBACK;
36 DBMS_OUTPUT.PUT_LINE('Error insert table APPS.XSFL_UPDATE_CSV ' ||
37     SQLERRM);
38 FND_FILE.PUT_LINE (FND_FILE.OUTPUT, 'Error insert table APPS.
39     XSFL_UPDATE_CSV ' || SQLERRM);

```

```

37 DELETE FROM APPS.XSFL_PROSES_UPLOAD_TEMP_2;
38 COMMIT;
39 RAISE;
40
41 END;
42 ]

```

Kode 3.4: Code Program Upload CSV

Selanjutnya, *program* akan melakukan *update* kedalam *table* dalam *database* dengan *requirement* yang telah diatur.

```

1 [
2 VS_VAR_BUF := PARSE_BROW( VL_LOB, VN_STR, VN_STR2 );
3
4 IF LENGTH( TRIM( VS_VAR_BUF ) ) > 0 THEN
5     VS_VAR_BUF := PARSE_BROW( VL_LOB, VN_STR, VN_STR2 );
6 END IF;
7
8 VN_CJML := VN_CJML + 1;
9
10 WHILE VN_STR2 > 1
11
12 LOOP
13     IF VN_CJML = 0 THEN
14         VN_STR := VN_STR + VN_LENROW;
15     ELSE
16         VN_LENROW := ( VN_STR2 - VN_STR );
17         VN_STR := VN_STR2 + 1;
18         VN_STR2 := VN_STR + VN_LENROW - 1;
19     END IF;
20
21     VS_VAR_BUF := PARSE_BROW( VL_LOB, VN_STR, VN_STR2 );
22
23     IF LENGTH( TRIM( VS_VAR_BUF ) ) > 0 THEN
24
25         FOR REC IN (
26
27             SELECT
28                 MAX( WORD.1 ) COA
29                 , MAX( WORD.2 ) ATTRIBUTE1
30                 , MAX( WORD.3 ) ATTRIBUTE2
31                 , MAX( WORD.4 ) ATTRIBUTE3
32                 , MAX( WORD.5 ) ATTRIBUTE4
33                 , MAX( WORD.6 ) ATTRIBUTE5
34             FROM (

```

```

34                                     SELECT
35                                     DECODE(
36 ROWNUM, 1, WORD, NULL ) WORD_1
37                                     , DECODE(
38 ROWNUM, 2, WORD, NULL ) WORD_2
39                                     , DECODE(
40 ROWNUM, 3, WORD, NULL ) WORD_3
41                                     , DECODE(
42 ROWNUM, 4, WORD, NULL ) WORD_4
43                                     , DECODE(
44 ROWNUM, 5, WORD, NULL ) WORD_5
45                                     , DECODE(
46 ROWNUM, 6, WORD, NULL ) WORD_6
47                                     , DECODE( ROWNUM, 11, WORD, NULL ) WORD_11
48                                     FROM ( SELECT ROWNUM
49                                     , SUBSTR(
50 WORD
51                                     ,
52 INSTR( WORD
53                                     , ' ' --P_SEPARATOR--
54                                     , 1
55                                     , LEVEL )
56                                     +
57                                     1
58                                     ,
59 INSTR( WORD
60                                     , ' ' --P_SEPARATOR-- ' ; '
61                                     , 1
62                                     , LEVEL + 1 )
63                                     -
64 INSTR( WORD
65                                     , ' ' --P_SEPARATOR-- ' ; '
66                                     , 1
67                                     , LEVEL )

```

```

57 1 )
58                                WORD
59                                FROM (SELECT '
60                                , '--P_SEPARATOR-- ' ; '
61                                VS_VAR_BUF || ' , '--P_SEPARATOR-- ' ; '
62                                WORD FROM DUAL)
63                                CONNECT BY INSTR (
64                                WORD
65                                , ' ,
66                                '--P_SEPARATOR-- ' ; '
67                                , 2
68                                ,
69                                LEVEL ) <> 0)
                                )
                                LOOP
]

```

Kode 3.5: Code Program Upload CSV

Dan pada akhirnya, *program* akan mengeluarkan *log* berupa *html* untuk melihat hasil dari proses *upload table* dan menghapus *data table* penampung.

```

1 [
2 fnd_file.put_line(fnd_file.output, '<!DOCTYPE html>');
3 fnd_file.put_line(fnd_file.output, '<html>');
4 fnd_file.put_line
5 (
6     fnd_file.output,
7     '<style>
8         table {
9             border-collapse: collapse;
10        }
11
12        table, th, tr, td {
13            border: 1px solid #ddd;
14        }
15
16        th {
17            align-text: center;
18        }
19

```

```

20         th, td {
21             padding-left: 15px;
22             padding-right: 15px;
23             padding-top: 5px;
24             padding-bottom: 5px;
25         }
26     </style>'
27 );
28     fnd_file.put_line(fnd_file.output, '<body>');
29
30     fnd_file.put_line(fnd_file.output, '<table>');
31     fnd_file.put_line(fnd_file.output, '<tr>');
32     fnd_file.put_line(fnd_file.output, '<th colspan
33 =12>Upload Adjustment Cost Report </th>');
34     fnd_file.put_line(fnd_file.output, '</tr>');
35     fnd_file.put_line(fnd_file.output, '<tr>');
36     fnd_file.put_line(fnd_file.output, '<th>File Name
37 </th>');
38     fnd_file.put_line(fnd_file.output, '<th>CODE
39 COMBINATION ID</th>');
40     fnd_file.put_line(fnd_file.output, '<th>COA</th>');
41 ;
42     fnd_file.put_line(fnd_file.output, '<th>ATTRIBUTE
43 1</th>');
44     fnd_file.put_line(fnd_file.output, '<th>ATTRIBUTE
45 2</th>');
46     fnd_file.put_line(fnd_file.output, '<th>ATTRIBUTE
47 3</th>');
48     fnd_file.put_line(fnd_file.output, '<th>ATTRIBUTE
49 4</th>');
50     fnd_file.put_line(fnd_file.output, '<th>ATTRIBUTE
51 5</th>');
52     fnd_file.put_line(fnd_file.output, '<th>STATUS</th>');
53     fnd_file.put_line(fnd_file.output, '<th>MESSAGE</
54 th>');
55     fnd_file.put_line(fnd_file.output, '<th>Creation
56 Date</th>');
57     fnd_file.put_line(fnd_file.output, '</tr>');
58     for r_output in
59 (
60         SELECT fl.file_name, xaft.* FROM
61         XSFL_UPDATE_CF_ATTRIBUTE_2 xaft

```

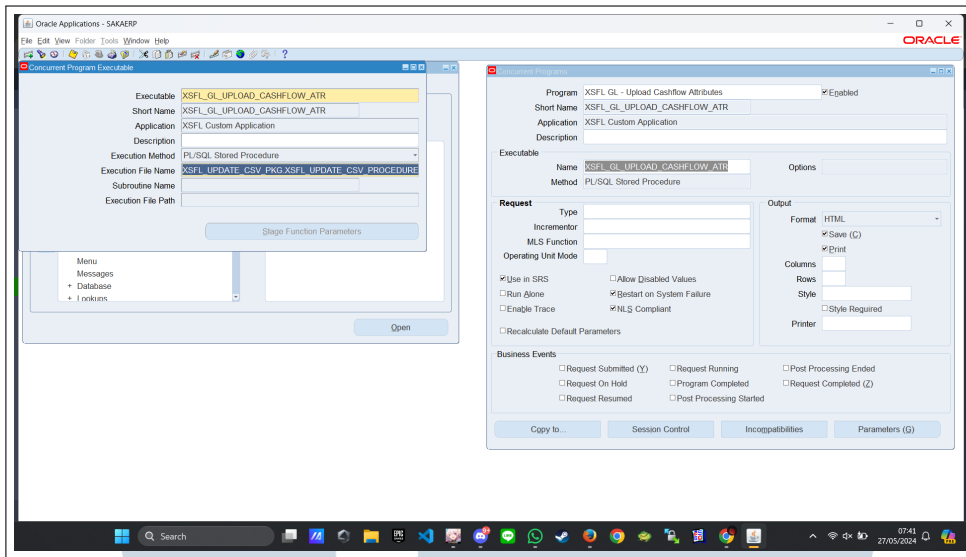
```

50         ,fnd_lobs fl
51         WHERE xaft.FILE_ID = P_FILE_ID--688827
52         and fl.file_id = xaft.file_id
53     )
54     loop
55         fnd_file.put_line(fnd_file.output, '<tr>');
56         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.file_name || '</td>');
57         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.CODE.COMBINATION_ID || '</td>');
58         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.COA || '</td>');
59         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.ATTRIBUTE1 || '</td>');
60         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.ATTRIBUTE2 || '</td>');
61         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.ATTRIBUTE3 || '</td>');
62         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.ATTRIBUTE4 || '</td>');
63         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.ATTRIBUTE5 || '</td>');
64         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.STATUS || '</td>');
65         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.MESSAGE || '</td>');
66         fnd_file.put_line(fnd_file.output, '<td>' ||
r_output.CREATION_DATE || '</td>');
67         fnd_file.put_line(fnd_file.output, '</tr>');
68     end loop;
69     fnd_file.put_line(fnd_file.output, '</table>');
70
71     fnd_file.put_line(fnd_file.output, '</body>');
72     fnd_file.put_line(fnd_file.output, '</html>');
73 ]

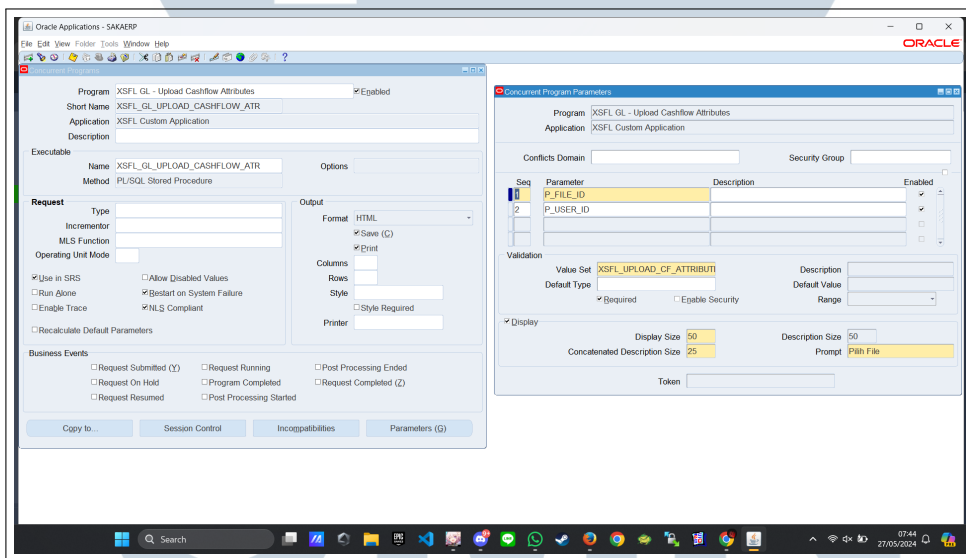
```

Kode 3.6: Code Program Upload CSV

Setelah membuat *program* di dalam *database*, langkah selanjutnya adalah untuk mengatur *oracle apps* untuk menampilkan *program* yang telah dibuat. Langkah pertama adalah untuk membuat *concurrent program* dan *concurrent executable* untuk menjalankan *program*.



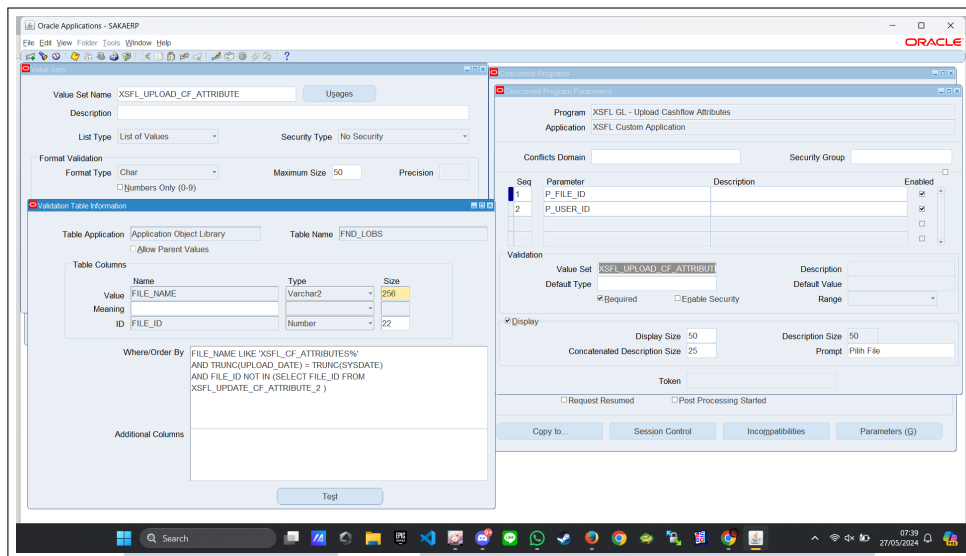
Gambar 3.2. Pembuatan *Executable* dalam *Oracle Apps*



Gambar 3.3. Pembuatan *Program* dalam *Oracle Apps*

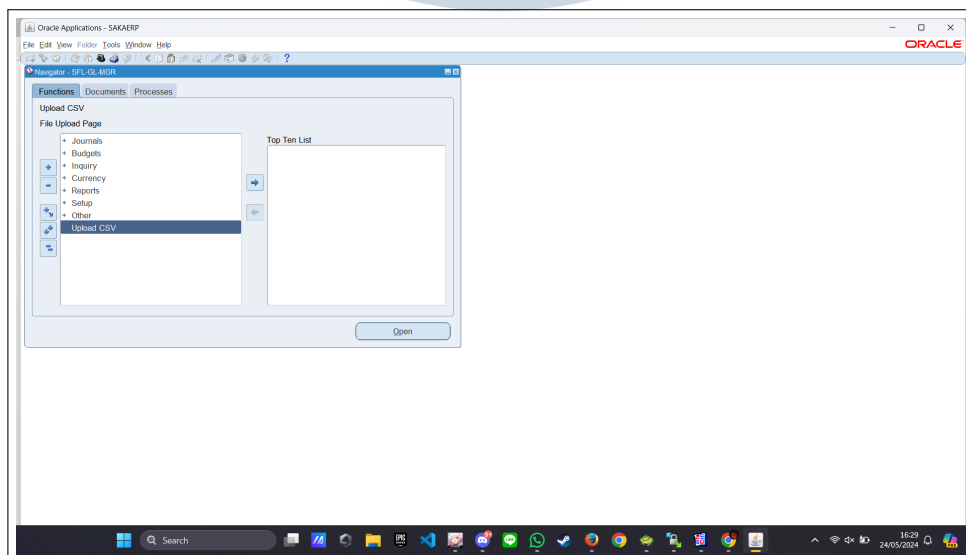
Langkah selanjutnya adalah untuk membuat *parameter* khusus untuk *concurrent program* agar *program* dapat memakai *paramater file CSV* untuk dipilih oleh *user*.

UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.4. Pembuatan parameter dalam Oracle Apps

Langkah terakhir adalah untuk membuat menu dalam responsibility *SFL-GL-MGR* untuk membuat user dapat mengupload file CSV. File CSV ini akan diupload sehingga oracle apps dapat memilih file apa yang akan diupload kedalam database.

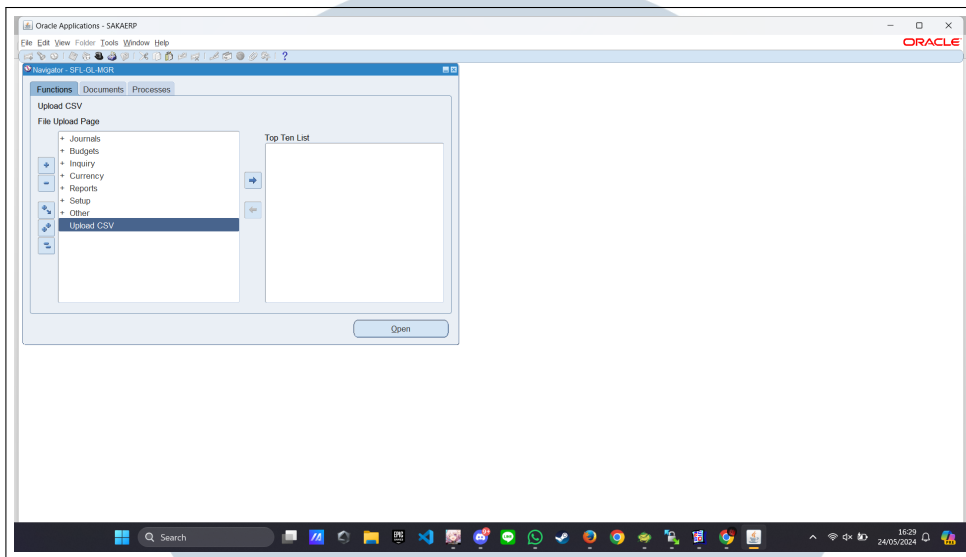


Gambar 3.5. Gambar Menu dalam Responsibility SFL-GL-MGR

3.3.3 Menjalankan Program

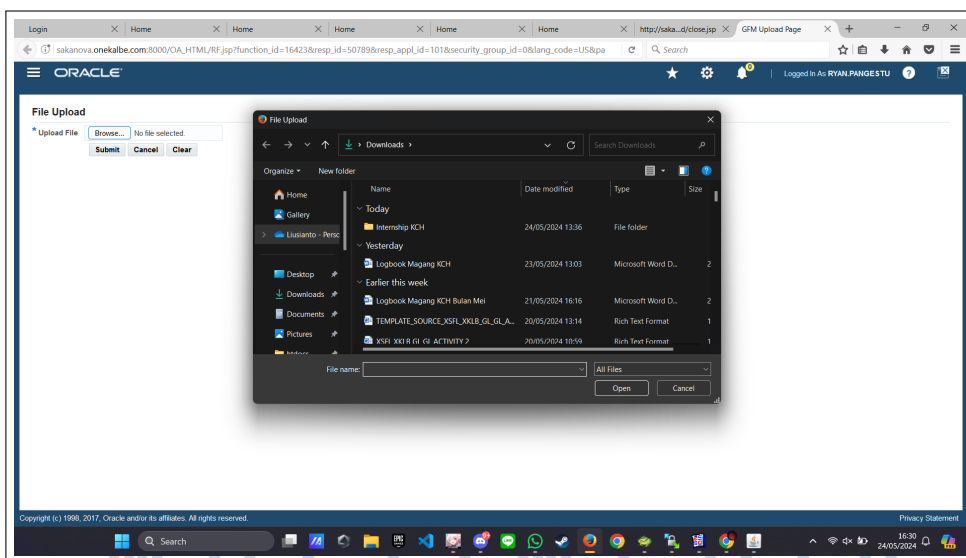
Dalam menjalankan program, pertama user perlu untuk membuka oracle apps yang merupakan aplikasi oracle untuk user. Didalam oracle apps terdapat

menu *SFL-GL-MGR* dimana berfungsi untuk mengupload dan mendownload *report* yang telah dibuat.



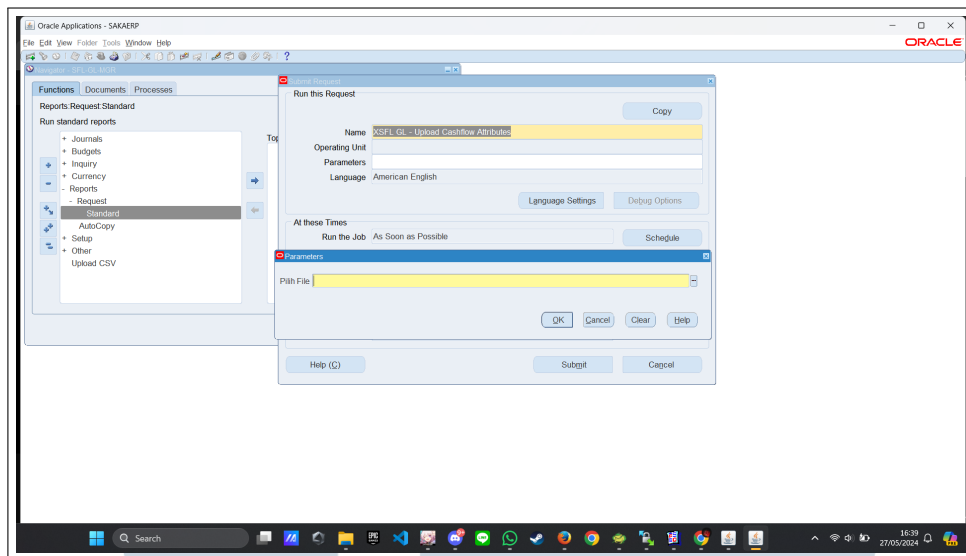
Gambar 3.6. Gambar menu *SFL GR MGR*

Lalu *user* akan mengupload *file CSV* dengan memilih menu *Upload CSV* dimana akan mengarahkan *user* kedalam tempat untuk memilih *file* untuk diupload.



Gambar 3.7. Gambar menu *Upload CSV*

Lalu *user* akan menjalankan *program* dengan cara memilih *menu report* dan memilih *program upload CSV* dan *program* akan menjalankan *procedure* untuk menghasilkan *report* berupa *html* dan mengupload *data* ke dalam *table database*.



Gambar 3.8. Gambar report Upload CSV dalam oracle apps

| File Name | CODE COMBINATION ID | COA | ATTRIBUTE 1 | ATTRIBUTE 2 | ATTRIBUTE 3 | ATTRIBUTE 4 | ATTRIBUTE 5 | STATUS | MESSAGE | Creation Date |
|--|---------------------|------------------------------------|-------------------|-----------------|---------------|-------------|-------------|--------|---------|---------------|
| XSFL_CF_ATTRIBUTES_test2024051Ver2.csv | 225005 | 05-61130-851-90001-2EMX3-000-00000 | AKTIVITAS OPERASI | PENGELUARAN KAS | BIAYA OPERASI | LAIN-LAIN | OTC | S | | 15-MAY-24 |
| XSFL_CF_ATTRIBUTES_test2024051Ver2.csv | 219004 | 05-61110-512-90002-2BDG5-000-00000 | AKTIVITAS OPERASI | PENGELUARAN KAS | BIAYA OPERASI | LAIN-LAIN | OTC | S | | 15-MAY-24 |
| XSFL_CF_ATTRIBUTES_test2024051Ver2.csv | 219005 | 05-61110-511-90002-2BDG5-000-00000 | AKTIVITAS OPERASI | PENGELUARAN KAS | BIAYA OPERASI | LAIN-LAIN | OTC | S | | 15-MAY-24 |
| XSFL_CF_ATTRIBUTES_test2024051Ver2.csv | 212006 | 05-61130-813-90002-2EWEN-000-00000 | AKTIVITAS OPERASI | PENGELUARAN KAS | BIAYA OPERASI | LAIN-LAIN | OTC | S | | 15-MAY-24 |

Gambar 3.9. Gambar report html dari proses upload CSV

3.4 Kendala dan Solusi yang Ditemukan

Kendala yang ditemukan selama proses pembuatan *program Upload CSV* ini adalah masih belum banyak memahami mengenai *oracle apps* dimana dalam bagian membuat *menu* dan membuat *parameter* masih membutuhkan bimbingan dari *mentor* dan pembelajaran yang lebih dalam lagi sehingga membuat pengerjaan pembuatan *program* menjadi lebih lama dari yang diinginkan.

Solusi yang ditemukan dari masalah adalah dengan melakukan penelitian dari bagian yang dirasa masih belum dipahami dan bimbingan dari *mentor* membuat

pengertian mengenai pembuatan *program upload CSV* dapat diselesaikan tanpa menghadapi masalah.



UMMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA