

BAB III

PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Koordinasi

Pelaksanaan kegiatan magang dilakukan pada sebuah perusahaan dengan posisi sebagai ABAP Programmer yang mendukung area FICO. Program magang dilaksanakan selama lima bulan, mulai dari 4 Agustus 2025 hingga 4 Februari 2026. Selama masa magang, mahasiswa memperoleh arahan dari tim internal perusahaan yang bertanggung jawab terhadap proses pengembangan dan pemeliharaan sistem SAP, khususnya pada modul Financial Accounting (FI) dan Controlling (CO).

Dalam kegiatan sehari-hari, mahasiswa berperan dalam membantu proses pengembangan laporan (report), melakukan perbaikan logika pada program yang sudah ada, membuat mockup output, serta mendukung proses debugging bersama tim. Mahasiswa juga dilibatkan dalam proses analisis kebutuhan teknis berdasarkan requirement yang diberikan oleh tim functional FICO.

Apabila selama kegiatan magang terdapat kendala yang berkaitan dengan pemahaman proses bisnis FICO, mahasiswa dapat berdiskusi dan meminta arahan dari pihak functional yang menangani area FI dan CO. Sementara itu, jika kendala berhubungan dengan coding, struktur program, atau solusi teknis, mahasiswa dapat berkonsultasi dengan pihak technical atau senior ABAP yang membimbing dalam hal implementasi teknis.

3.2 Tugas yang Dilakukan

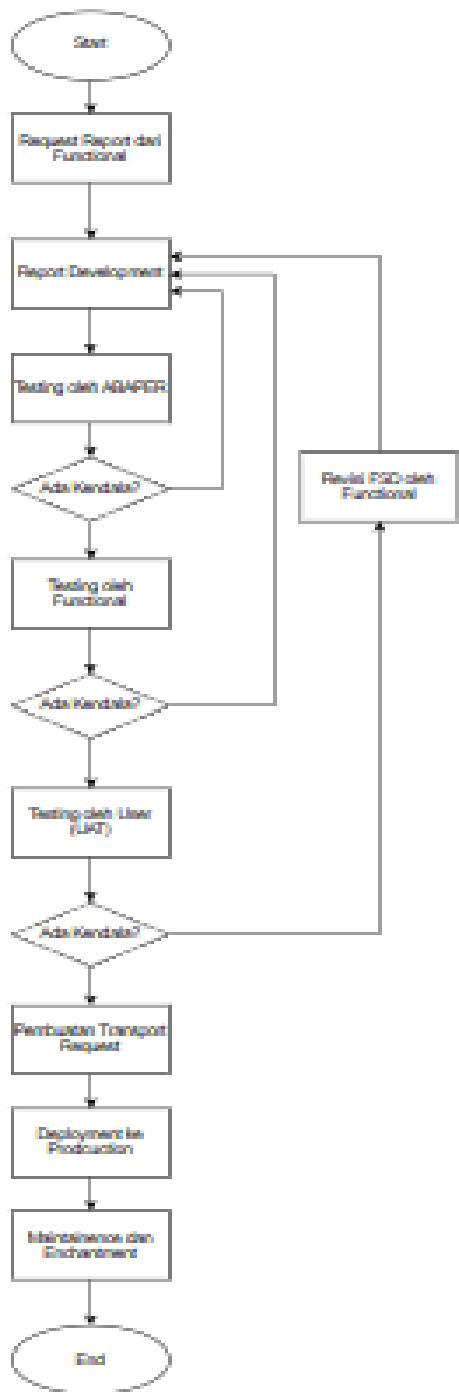
Berikut merupakan uraian tugas yang dilakukan selama program magang berlangsung sebagai ABAP Programer FICO:

1. Membuat Report ABAP untuk Mendukung Kebutuhan Fungsional

Salah satu tugas utama selama magang adalah membuat report ABAP untuk memenuhi kebutuhan tim Functional pada modul FICO. Proses ini meliputi pemahaman spesifikasi fungsional (FSD), perancangan logika program, serta

pemetaan tabel SAP dan field yang dibutuhkan. Mahasiswa mempelajari cara menulis SELECT statement, memanfaatkan join antar tabel, serta menerapkan logic kalkulasi dan filter sesuai kebutuhan laporan. Dengan membuat report ini, mahasiswa mendapatkan pengalaman praktis dalam mengubah kebutuhan fungsional menjadi program ABAP yang berjalan efektif dan menghasilkan data yang akurat

Pembuatan report mengikuti alur SDLC, di mana setelah request report diajukan oleh tim Functional, mahasiswa mengembangkan program ABAP sesuai spesifikasi. Selanjutnya dilakukan testing internal oleh ABAPER untuk memastikan query, join antar tabel, serta logika kalkulasi dan filter berjalan sesuai kebutuhan. Jika ditemukan kendala, proses kembali ke tahap development untuk perbaikan. Setelah lolos testing internal, report diuji oleh tim Functional, dan bila masih diperlukan penyesuaian, pengembangan dilakukan kembali hingga sesuai spesifikasi. Tahap akhir meliputi User Acceptance Testing (UAT), pembuatan transport, dan deployment ke production. Dengan mengikuti alur SDLC ini, mahasiswa mendapatkan pengalaman praktis dalam mengubah kebutuhan fungsional menjadi program ABAP yang berjalan efektif dan menghasilkan data yang akurat.



Gambar 3.1 SDLC Pengembangan Report SAP

2. Administrasi Teknis Sistem SAP

Tugas selanjutnya sebagai abap programer adalah melakukan administrasi teknis pada sistem SAP untuk memahami bagaimana sistem dikelola, dikonfigurasi, dan dipelihara agar dapat berjalan secara optimal. Dalam proses ini, mahasiswa diperkenalkan pada berbagai komponen dasar administrasi SAP seperti pengecekan status sistem, monitoring performa, pemantauan log, serta memahami alur transport request yang digunakan untuk memindahkan konfigurasi maupun program dari lingkungan pengembangan ke lingkungan produksi.

3. Membantu Penyelesaian Masalah Teknis SAP

Tugas lain yang dilakukan selama magang adalah membantu tim dalam menangani berbagai permasalahan teknis yang muncul pada sistem SAP. Ketika terjadi error atau ketidaksesuaian data pada modul FICO, mahasiswa memiliki kesempatan untuk mempelajari langkah-langkah troubleshooting awal, seperti membaca pesan error, mengecek data terkait, atau menelusuri log pada sistem.

Jika permasalahan terkait dengan proses bisnis, mahasiswa dapat berkonsultasi dengan tim functional untuk memahami kebutuhan bisnis yang mendasari terjadinya error. Sementara itu, jika masalah bersifat teknis seperti bug pada program, performa query, atau struktur logika kode, mahasiswa diarahkan untuk berdiskusi dengan tim technical atau senior ABAP untuk menemukan akar masalah dan menentukan solusi yang sesuai. Proses ini memberikan pemahaman praktis mengenai cara kerja kolaboratif antara tim functional dan technical dalam menyelesaikan isu SAP.

4. Memberikan Dukungan dan Pendampingan kepada Pengguna dan Tim Proyek

Selama program magang, mahasiswa juga turut berperan dalam memberikan bantuan teknis kepada pengguna maupun tim proyek yang

membutuhkan dukungan dalam penggunaan sistem SAP. Bantuan ini dapat berupa penjelasan sederhana mengenai cara kerja fitur tertentu, pengecekan awal terhadap masalah yang dialami pengguna, ataupun memberikan klarifikasi terkait fungsi program dan hasil output.

Selain itu, mahasiswa juga mempelajari cara melakukan dokumentasi atas masalah dan langkah penyelesaian yang telah dilakukan, agar tim lain dapat mengetahui riwayat troubleshooting yang telah dicoba. Keterlibatan dalam dukungan ini membantu mahasiswa mengembangkan kemampuan komunikasi profesional serta memahami pentingnya dokumentasi sebagai bagian dari operasional SAP.

3.3 Uraian Kerja Magang

Kegiatan magang dilaksanakan mulai tanggal 4 Agustus hingga 4 Februari 2026 dengan fokus pada pengembangan kemampuan teknis di lingkungan SAP. Selama periode ini, kegiatan dimulai dari masa pengenalan lingkungan kerja, pemahaman alur proses bisnis, serta adaptasi terhadap sistem dan tools yang digunakan. Tahap berikutnya difokuskan pada pembelajaran ABAP Basic dan pemahaman Functional Specification Document (FSD) sebagai dasar pengembangan report SAP. Berdasarkan pemahaman tersebut, dilakukan pembuatan dan pengembangan berbagai report, antara lain Report Monitoring Master Work Order, report internal, report external, dan Report WBS Summary, termasuk penambahan logic, revisi kolom, serta enhancement report AR Payment. Rangkaian kegiatan ini dilaksanakan secara terstruktur dan kronologis, yang rinciannya dapat dilihat pada tabel kegiatan berikut.

Table 3.1 Rangkuman Pekerjaan yang Dilakukan Selama Masa Magang

Periode (Tanggal)	Pekerjaan yang Dilakukan
4 Agustus – 8 Agustus 2025	Masa pengenalan lingkungan kerja dan alur proses bisnis
11 Agustus – 15 Agustus 2025	Adaptasi sistem dan tools kerja serta pengenalan SAP
11 Agustus – 22 Agustus 2025	Membuat mockup fungsional berdasarkan kebutuhan user
18 Agustus – 12 September 2025	Belajar ABAP Basic
1 September – 19 September 2025	Belajar membaca dan memahami Functional Specification Document (FSD)
1 September – 5 September 2025	Analisis kebutuhan report SAP berdasarkan FSD untuk Report Monitoring Master Work Order Internal
8 September – 12 September 2025	Pembuatan struktur program ABAP dan desain selection screen Report Monitoring Master Work Order
15 September – 19 September 2025	Pengambilan dan pengolahan data dari tabel SAP untuk Report Monitoring Master Work Order
22 September – 26 September 2025	Testing dan perbaikan (bug fixing) Report Monitoring Master Work Order
15 September – 19 September 2025	Pembuatan logika dasar dan struktur program untuk report internal
22 September – 26 September 2025	Pengembangan fitur dan tampilan output (ALV) report internal
29 September – 3 Oktober 2025	Testing, debugging, dan penyesuaian report internal sesuai kebutuhan user
29 September – 3 Oktober 2025	Analisis kebutuhan dan perancangan report external
6 Oktober – 10 Oktober 2025	Pengembangan logika pengolahan data dan tampilan report external
13 Oktober – 17 Oktober 2025	Testing dan penyempurnaan report external
6 Oktober – 10 Oktober 2025	Perancangan struktur data dan logika report WBS Summary
13 Oktober – 17 Oktober 2025	Pengembangan pengolahan data dan tampilan output report WBS Summary

20 Oktober – 24 Oktober 2025	Testing, optimasi, dan finalisasi report WBS Summary
13 Oktober – 31 Oktober 2025	Penambahan logic untuk data cancel pada report SAP
20 Oktober – 7 November 2025	Revisi dan penambahan kolom pada report SAP sesuai permintaan user
3 November – 21 November 2025	Enhancement Report AR Payment
10 November – 28 November 2025	Penambahan radio button dan kolom pada Report AR Payment
17 November – 5 Desember 2025	Revisi akhir dan penyempurnaan report SAP

Kegiatan magang diawali dengan tahap orientasi yang berlangsung pada minggu pertama. Pada tahap ini disampaikan pengenalan mengenai struktur kerja tim SAP, alur komunikasi antara tim functional dan technical, serta gambaran umum proses bisnis perusahaan yang berkaitan dengan modul FI/CO. Selain itu, diberikan pula pemahaman mengenai standar kerja yang digunakan perusahaan, termasuk tata cara membaca Functional Specification Document (FSD), mekanisme transport request, serta prosedur penyampaian kebutuhan sistem antar divisi.

Di minggu kedua, kegiatan berfokus pada pendalaman kemampuan teknis melalui pembelajaran dasar-dasar pemrograman ABAP. Materi yang dipelajari mencakup struktur program ABAP, penggunaan internal table, pembuatan selection screen, ALV report, hingga teknik debugging. Pada periode ini dilakukan juga analisis awal terhadap dokumen kebutuhan sistem serta penyusunan mockup desain report yang berfungsi sebagai acuan functional sebelum implementasi teknis dilakukan.

Setelah proses orientasi dan pembelajaran awal, kegiatan magang berlanjut ke tahap pengembangan dua project utama, yaitu Report Monitoring Master Work Order dan Enhancement Report Monitoring Progress Kredit Investasi / AR Payment. Kedua project ini dilaksanakan secara bertahap sejak fase analisis,

pembuatan desain, pengembangan logic program, hingga iterasi perbaikan berdasarkan evaluasi functional.

Project pertama, Report Monitoring Master Work Order, dikembangkan pada periode Agustus hingga Oktober. Ruang lingkup pekerjaan mencakup pembuatan report internal, report external, serta WBS Summary. Proses pengembangan mencakup pengambilan data dari tabel terkait, pembuatan output dalam format ALV, serta beberapa tahapan revisi yang meliputi penambahan kolom, penyesuaian logic, hingga implementasi logic untuk menangani data cancel.

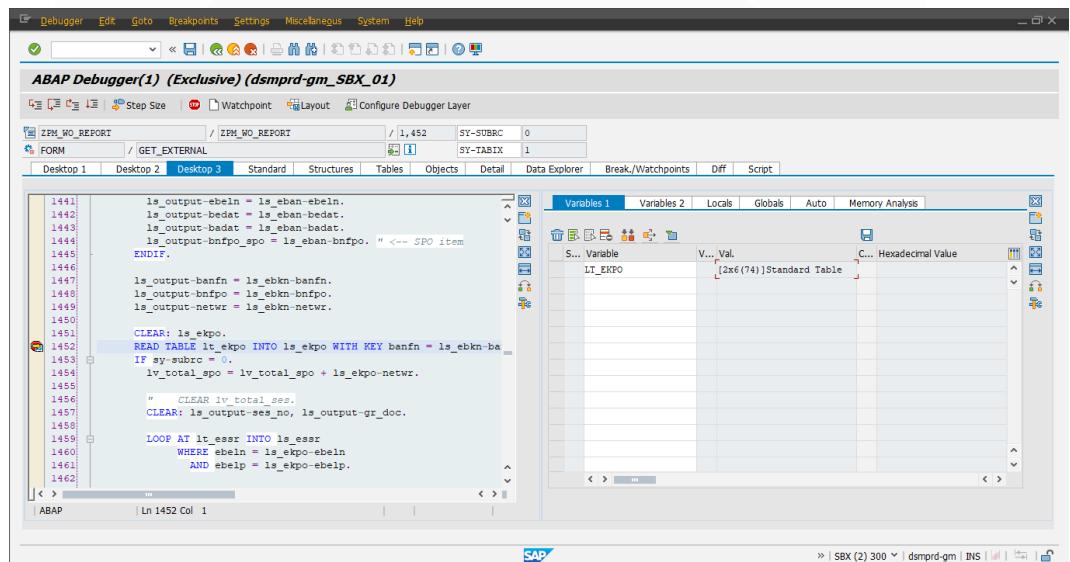
Pada periode berikutnya, November hingga Januari, kegiatan difokuskan pada project kedua, yaitu enhancement Report Monitoring Progress Kredit Investasi. Project ini memiliki cakupan pekerjaan berupa penambahan radio button pada selection screen, penambahan kolom laporan, serta penyesuaian logic perhitungan pada program existing. Pelaksanaan enhancement dilakukan melalui analisis struktur program sebelumnya, pengujian melalui debugging, serta revisi hingga hasil akhir sesuai dengan kebutuhan functional.

Menjelang akhir masa magang, seluruh aktivitas pengembangan dirangkum dalam bentuk dokumentasi yang berisi daftar perubahan, logic yang diterapkan, serta catatan teknis yang mendukung proses pemeliharaan sistem. Seluruh rangkaian kegiatan ini memberikan gambaran menyeluruh mengenai proses kerja pengembangan program SAP, dan penjelasan rinci mengenai tiap tahapan akan disajikan pada sub-bab berikutnya.

3.3.1 Tahap Orientasi dan Pemahaman Dasar Teknis

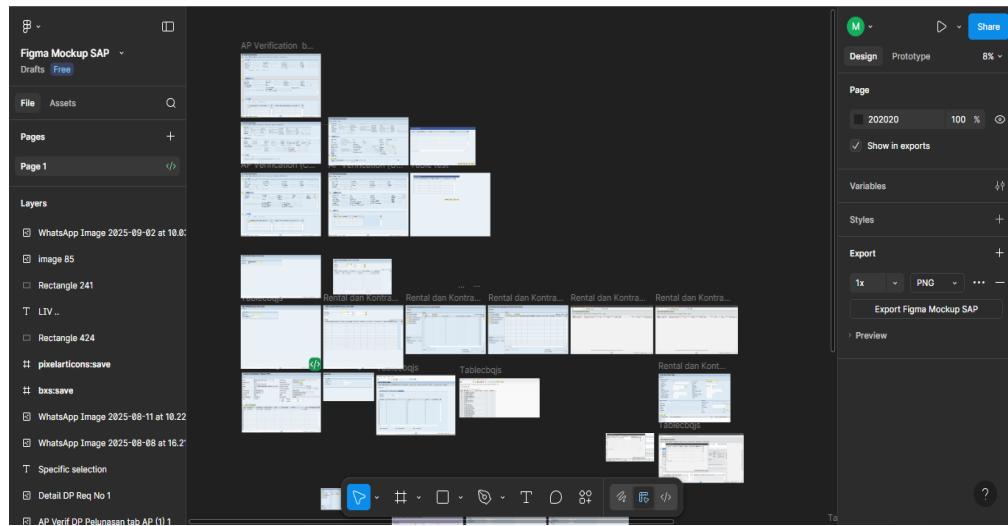
Tahap orientasi pada awal masa magang berfokus pada pemberian pemahaman menyeluruh terkait proses bisnis perusahaan serta alur kerja yang berkaitan dengan kebutuhan laporan pada sistem SAP. Pemahaman terhadap proses bisnis ini menjadi langkah fundamental karena menentukan struktur informasi yang harus ditampilkan dalam program report yang akan dikembangkan. Seiring dengan proses orientasi tersebut, dilakukan pula pembelajaran dasar-dasar pemrograman ABAP, mencakup

struktur program, penggunaan internal table, teknik seleksi data, hingga penelusuran logika melalui debugging.



Gambar 3.2 Debuging Abap SAP

Pada saat yang bersamaan, kegiatan pembelajaran ini diintegrasikan dengan tugas pembuatan mockup di Figma sebagai rancangan awal tampilan dan kebutuhan laporan yang akan digunakan oleh pihak functional dalam penyusunan dan validasi dokumen Functional Specification Document (FSD). Kegiatan membaca dan memahami FSD juga mulai dilakukan untuk memastikan setiap kebutuhan bisnis dapat diterjemahkan secara sistematis ke dalam desain dan logika program. Seluruh rangkaian aktivitas pada tahap ini memberikan dasar yang kuat sebelum memasuki fase pengembangan report secara lebih teknis dan terstruktur pada tahap berikutnya.



Gambar 3.3 Mockup yang Dikembangkan Melalui Figma

3.3.2 Report Monitoring Master Work Order

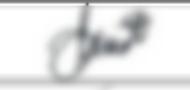
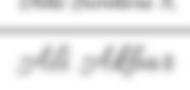
Report Work Order (WO) adalah laporan yang berfungsi untuk menampilkan informasi terkait aktivitas maintenance. Program penambahan report Work Order (WO) maintenance dikembangkan untuk meningkatkan visibilitas dan pengawasan terhadap aktivitas maintenance perusahaan. Report ini dirancang untuk menampilkan informasi status dan detail Work Order secara real-time maupun historis, sehingga memastikan bahwa seluruh kegiatan maintenance berjalan sesuai dengan jadwal, standar, dan prosedur yang telah ditetapkan. Dari sisi persyaratan bisnis, report ini mampu menampilkan informasi detail pada setiap inputan order maintenance, membedakan antara WO Internal (INT), WO External (EXT), serta menyajikan ringkasan WBS. Proses bisnis report monitoring Work Order mencakup seluruh aspek pekerjaan maintenance, mulai dari Work Order yang masih outstanding, dalam proses, hingga yang telah selesai

Spesifikasi Fungsional Enhancement

Title: REPORT MONITORING MASTER WORK ORDER

RICEF ID			
Transaction Code (jika ada)	ZPM_WO_REPORT		
Application Module	Plant Maintenance (PM)		
Nama Penulis			
Tanggal Pembuatan	11 September 2025	<input type="checkbox"/>	
Prioritas	<input type="checkbox"/> Tinggi	<input checked="" type="checkbox"/> Sedang	<input type="checkbox"/> Rendah
Penilaian Kompleksitas	<input type="checkbox"/> Tinggi	<input checked="" type="checkbox"/> Sedang	<input type="checkbox"/> Rendah
Jenis Proses	<input checked="" type="checkbox"/> Online	<input type="checkbox"/> Batch	<input type="checkbox"/> Lainnya:
Periode pemrosesan	<input checked="" type="checkbox"/> Harian	<input type="checkbox"/> Mingguan	<input type="checkbox"/> Bulanan
	<input type="checkbox"/> Mingguan	<input type="checkbox"/> Bulanan	<input type="checkbox"/> Tahunan
Perencanaan Tempo	<input type="checkbox"/> Lainnya:		
Catatan	Report Monitoring Master Work Order Internal, Work Order External dan Summary WBS untuk melihat lebih detail order dan secara real-time terhadap suatu cost.		

Persetujuan / Sign off Spesifikasi Enhancement

Peran	Nama	Tanggal	Tanda Tangan
Konsultan Functional	Mr. John Doe	10 September 2025	
Section Head	Mr. Study Nopita	10 September 2025	
Department Head	Mr. Business	10 September 2025	
Koordinator ABAPER	Mr. Manager	10 September 2025	

Catatan Perubahan

No	Tanggal	Deskripsi Singkat
1.	20 October 2025	Logic data cancel, dan add column pada WBS Summary
2.	03 November 2025	Revisi penambahan kolom di selection screen, dan logic pada display external & display summary WBS
3.		

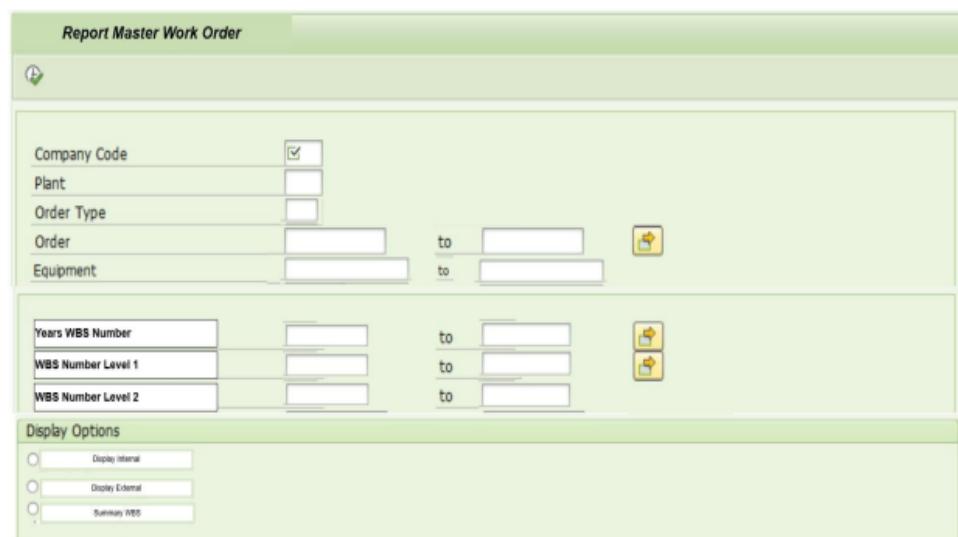
PMD - Functional Specification - Report Monitoring Work Order

Page | 1

Gambar 3.4 Sample FSD yang Diberikan Untuk Pengembangan Report Moniotoring Master Work Order

Gambar 3.1 merupakan sampel FSD yang diberikan pada awal pengembangan report. FSD tersebut menampilkan kebutuhan fungsional untuk area plant management, khususnya terkait pemantauan aktivitas operasional plant serta keterkaitannya dengan pencatatan biaya di modul FICO. Selain menjelaskan alur proses dan struktur informasi yang dibutuhkan, FSD ini juga memuat table logic fungsional yang mendetail, mencakup pemetaan tabel SAP, dependensi data, serta aturan kalkulasi yang harus diterapkan dalam report. Isi FSD tersebut membantu

memastikan bahwa data dari aktivitas plant—seperti pergerakan material, pemakaian biaya, dan output produksi dapat dikonversikan secara akurat menjadi informasi finansial dan controlling. Dengan prioritas penggerjaan yang dikategorikan sedang dan tingkat kompleksitas yang juga sedang, FSD ini memberikan kejelasan bagi tim ABAP untuk memahami kebutuhan pengguna sekaligus melakukan pengembangan report secara efisien dan terarah.



Gambar 3.5 Mockup Tampilan Awal Selection Screen Untuk Report Monitoring Master Work Order

Gambar 3.3 merupakan mockup tampilan selection screen yang ada di dalam FSD. Mockup display options tersebut menampilkan penggunaan radio Button untuk user mengisi salah satu filter untuk di lakukan tick dan memilih Program display mana yang ingin di tampilkan per masing – masing report, jika user melakukan tick pada display internal & external makan akan menampilkan sesuai pilihan .

Table 3.2 Spesifikasi Selection Screen Yang Diberikan Untuk Report Monitoring Master Work Order

Column	Table	Field Name	Functional Logic
Company Code	AUFK	BUKRS	GET BUKRS FROM AUFK
Plant	AUFK	WERKS	GET WERKS FROM AUFK
Order Type	AUFK	AUART	GET AUART FROM AUFK
Order	AUFK	AUFNR	GET AUFNR FROM AUFK
Equipment	EQUI	EQUNR	GET EQUNR FROM EQUI
WBS Number Level 1	PRPS	PSPNR	GET PSPNR FROM PRPS (Level 1)
WBS Number Level 2	PRPS	PSPNR	GET PSPNR FROM PRPS (Level 2)
Period Basic Start	AFKO	GSTRP	GET GSTRP FROM AFKO

Dalam implementasinya, selection screen tidak hanya berisi field standar, tetapi juga dilengkapi dengan radio button yang memungkinkan pengguna memilih jenis laporan yang ingin ditampilkan, yaitu Internal Work Order, External Work Order, dan WBS Summary. Ketiga opsi ini secara langsung mengendalikan field apa saja yang dapat diisi dan ditampilkan oleh sistem. Ketika pengguna memilih Internal WO atau External WO, maka selection screen hanya akan menampilkan dan mengaktifkan field yang memang relevan dengan work order pada modul PM/Plant Maintenance, yaitu Company Code, Plant, Order Type, Work Order , serta Equipment . Field-field ini seluruhnya diambil dari tabel order, dan equipment , sehingga berfungsi sebagai filter utama untuk mempersempit data order yang akan dibaca saat proses SELECT berjalan. Pada mode ini, field seperti WBS Number Level 1, WBS Number Level 2, dan Period Basic Start secara otomatis disembunyikan atau *di-disable*, karena kedua jenis report (internal maupun external) tidak menggunakan struktur WBS sebagai dasar pemilihannya.

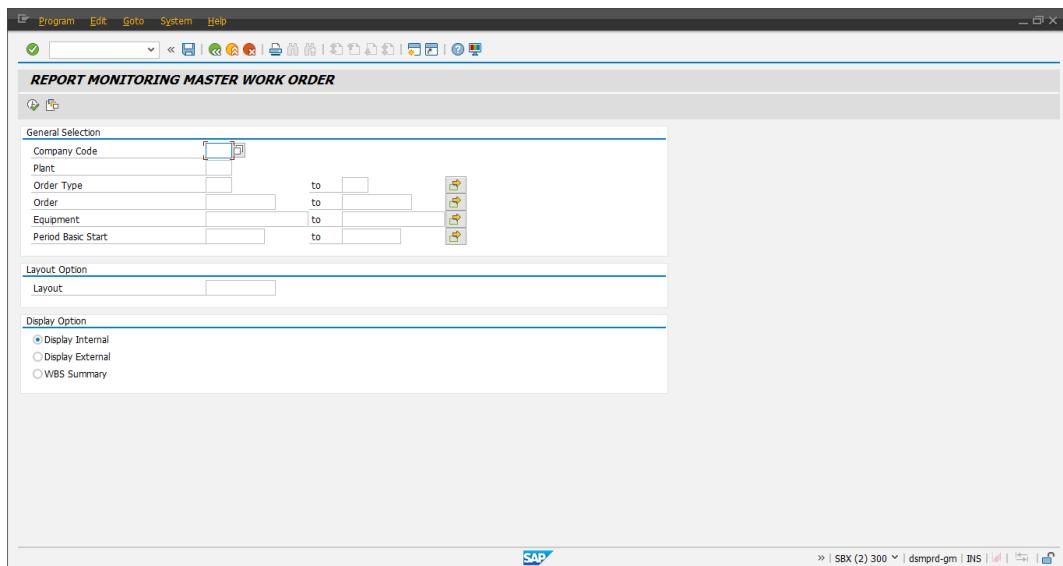
```

12  □ SELECTION-SCREEN BEGIN OF BLOCK b01 WITH FRAME TITLE text_b01.
13  |  SELECT-OPTIONS:
14  |  |  s_      FOP      NO INTERVALS NO-EXTENSION OBLIGATORY MODIF ID g1,
15  |  |  s_      FOP      NO INTERVALS NO-EXTENSION MODIF ID g1,
16  |  |  s_      FOP      MODIF ID il,
17  |  |  s_      FOP      MODIF ID g1,
18  |  |  s_      FOP      MODIF ID g1,
19  |  |  s_      FOP      MODIF ID g1,
20  |  |  s_      FOR      MODIF ID il.
21  |  SELECTION-SCREEN END OF BLOCK b01.
22
23  □ SELECTION-SCREEN BEGIN OF BLOCK b04 WITH FRAME TITLE text_b04.
24  |  PARAMETERS p_vari TYPE disvariant-variant.
25
26  |  SELECTION-SCREEN END OF BLOCK b04.
27
28  □ SELECTION-SCREEN BEGIN OF BLOCK b02 WITH FRAME TITLE text_b02.
29  |  SELECT-OPTIONS:
30  |  |  s_      FOR sy-datum+0(4) NO INTERVALS NO-EXTENSION MODIF ID w1,
31  |  |  s_      FOR      MODIF ID w1,
32  |  |  s_      FOR      MODIF ID w1.
33  |  SELECTION-SCREEN END OF BLOCK b02.
34
35  □ SELECTION-SCREEN BEGIN OF BLOCK b03 WITH FRAME TITLE text_b03.
36  |  PARAMETERS: disp_int RADIobutton GROUP grp1 USER-COMMAND rad,      " Internal
37  |  |  disp_ext RADIobutton GROUP grp1,                                " External
38  |  |  sum_wbs  RADIobutton GROUP grp1.                                " WBS
39  |  SELECTION-SCREEN END OF BLOCK b03.

```

Gambar 3.6 Code Untuk Tampilan Logic Awal Selection Screen Awal Untuk Report Monitoring Master Work Order

Sebaliknya, apabila pengguna memilih radio button WBS Summary, maka mekanisme selection berubah secara dinamis. Pada mode ini, field work order seperti Order Type, Work Order, dan Equipment tidak ditampilkan atau tidak dapat diakses, karena laporan WBS bertujuan menampilkan informasi berdasarkan struktur proyek dan bukan berdasarkan order individual. Sistem kemudian mengaktifkan field yang relevan dengan struktur proyek, yaitu WBS Number Level 1 dan WBS Number Level 2, yang diambil dari tabel PRPS sesuai dengan hierarchy Project System (PS). Pengguna juga dapat menentukan periode pekerjaan berdasarkan field Period Basic Start untuk mengontrol rentang waktu rencana mulai pekerjaan pada level WBS.



Gambar 3.7 Hasil Tampilan Awal Selection Option Awal Untuk Report Monitoring Master Work Order

1. Report Monitoring Work Order Internal

Report Work Order Internal dikembangkan untuk memberikan gambaran menyeluruh mengenai seluruh proses pengelolaan perintah kerja, mulai dari data master order, informasi teknis peralatan, detail aktivitas operasi, hingga realisasi biaya dan status penyelesaian pekerjaan. Secara prinsip, report ini dirancang dengan pendekatan yang terstruktur, yaitu memulai dari data tingkat header, kemudian bergerak menuju data detail aktivitas dan realisasi, dan akhirnya menghasilkan informasi nilai biaya serta status akhir order. Untuk mencapai tingkat akurasi tersebut, sistem memanfaatkan integrasi lintas berbagai tabel operasional dan finansial pada SAP, termasuk tabel-tabel proses produksi, pemeliharaan, log material, pencatatan akuntansi, serta tabel kustom yang digunakan untuk data karyawan.

Table 3.3 Spesifikasi yang Diberikan Oleh Functional Untuk Report Monitoring Master
Work Order Internal

No	Column Table	Table	Field Name	Functional Logic
1	Company Code	AUFK	BUKRS	GET BUKRS FROM AUFK
2	Plant	AUFK	WERKS	GET WERKS FROM AUFK
3	Equipment	EQUI	EQUNR	GET EQUNR FROM EQUI
4	Description Object Type	EQUI	EQKTX	GET EQKTX FROM EQUI
5	Notification Order	QMEL	QMNUM	GET QMNUM FROM QMEL
6	Created On	AUFK	ERDAT	GET ERDAT FROM AUFK
7	Entered By	AUFK	ERNAM	GET ERNAM FROM AUFK
8	Order Type	AUFK	AUART	GET AUART FROM AUFK
9	Work Order	AUFK	AUFNR	GET AUFNR FROM AUFK
10	Description Work Order	AUFK	KTEXT	GET KTEXT FROM AUFK
11	Activity No	AFVC	VORNR	GET AUFPL FROM AFKO JOIN AFVC ON AUFPL
12	Operation Short Text	AFVC	LTXA1	GET AUFPL FROM AFKO JOIN AFVC ON AUFPL
13	WBS Element Number	AUFK	PSPEL	GET PSPEL FROM AUFK
14	Control Key	AFVC	STEUS	GET AUFPL FROM AFKO JOIN AFVC
15	Plan Start Date	AFKO	GSTRP	GET GSTRP FROM AFKO
16	Plan End Date	AFKO	GLTRP	GET GLTRP FROM AFKO
17	Total Plan Mechanic	ZOPR_EMPLOYEE	ENAME1	GET AUFNR FROM AUFK JOIN ZOPR_EMPLOYEE ON AUFNR
18	Posting Date	AFRU	BUDAT	GET AUFNR FROM AUFK JOIN AFRU ON AUFNR, SORT DESCENDING (STOKZ, STZHL/RMZHL)
19	Actual Start Date	AFRU	ISDD	GET ISDD FROM AFRU, SORT DESCENDING
20	Actual End Date	AFRU	IEDD	GET IEDD FROM AFRU, SORT DESCENDING
21	Actual Start Time	AFRU	ISDZ	GET ISDZ FROM AFRU, SORT DESCENDING

22	Actual End Time	AFRU	IEDZ	GET IEDZ FROM AFRU, SORT DESCENDING
23	Total Actual Mechanic	AFRU	ISMNW	GET COUNT LINE FROM AFRU BY AUFNR
24	Actual Activity Work	AFRU	ISMNW	GET SUM ISMNW FROM AFRU
25	Unit for Work	AFRU	ISMNE	GET ISMNE FROM AFRU
26	Total GI Amount	MSEG	DMBTR	GET SUM DMBTR FROM MSEG (BWART 261 = +1, BWART 262 = -1)
27	Alloc WS per Order	ACDOCA	HSL	GET SUM HSL WHERE RLDNR = 'OL', RACCT = '80000124', AUFNR
28	Total Actual Cost	-	-	SUM (Total GI Amount + Alloc WS)
29	Actual Cost / H	-	-	Total Actual Cost / Actual Activity Work
30	Status Work Order (TECO / UNTECO)	JEST	STAT	GET OBJNR FROM AUFK JOIN JEST, STAT = I0045, INACT = X (UNTECO) / ACTIVE (TECO)
31	Teco Date	JCDS	UDATE	GET OBJNR FROM AUFK JOIN JCDS, STAT = I0045, INACT = ACTIVE
32	Last Change User (TECO / UNTECO)	AUFK	AENAM	GET AENAM FROM AUFK
33	Change Date	AUFK	AEDAT	GET AEDAT FROM AUFK

Table 3.2 merupakan sampel spesifikasi fungsional yang disusun oleh tim functional sebagai dasar pengembangan report Work Order Internal oleh tim abap. Tabel spesifikasi ini berfungsi sebagai panduan teknis utama dalam pengembangan report ini.

```

221 |     SELECT a~aufnr, a~auwart, a~bukrs, a~werks, a~erdat, a~ernam,
222 |         a~ktext, a~pspel, a~objnr, a~aenam, a~aedat, a~waers
223 |     FROM aufk AS a
224 |     INNER JOIN viaufks AS v
225 |         ON a~aufnr = v~aufnr
226 |     INNER JOIN afko AS f
227 |         ON a~aufnr = f~aufnr
228 |     WHERE a~bukrs IN @s_bukrs
229 |         AND a~werks IN @s_werks
230 |         AND a~auwart IN @s_auwart
231 |         AND a~aufnr IN @s_aufnr
232 |         AND v~equnr IN @s_equnr
233 |         AND f~gstrup IN @s_gstrup
234 |     INTO TABLE @lt aufk.

```

Gambar 3.8 Code Selection Awal Untuk Work Order Internal

Company Code dan Plant diambil dari tabel AUFK melalui field BUKRS dan WERKS untuk mendapatkan kode perusahaan dan lokasi plant dari Work Order. Equipment and Description Object Type diperoleh dari tabel EQUI menggunakan field EQUNR dan EQKTX, sehingga setiap Work Order dapat dihubungkan dengan peralatan dan tipe objeknya.

Notification Order diambil dari tabel QMEL melalui field QMNUM, sedangkan tanggal pembuatan (Created On) dan pembuat (Entered By) Work Order diambil dari AUFK melalui ERDAT dan ERNAM. Order Type dan nomor Work Order (Work Order) diambil dari AUFK melalui AUART dan AUFNR, sedangkan deskripsi Work Order (Description Work Order) diperoleh dari KTEXT di AUFK.

Detail aktivitas Work Order diambil dari tabel AFVC, seperti Activity No melalui join dengan AFKO menggunakan field AUFPL, dan Operation Short Text (LTXA1) juga diambil melalui join yang sama, sehingga struktur aktivitas dan deskripsi setiap operasi dapat ditampilkan. Control Key diambil dari AFVC yang di-join dengan AFKO untuk mendapatkan informasi jenis kontrol pada setiap aktivitas. WBS Element Number diambil dari AUFK melalui PSPEL untuk menghubungkan Work Order dengan struktur WBS proyek. Plan Start Date dan Plan End Date diambil dari AFKO melalui field GSTRP dan GLTRP.

```

258     IF lt_aufk[] IS NOT INITIAL.
259         SELECT aufnr, rueck, rmzhl, stzhl, budat, ismnw, isdd, iedd, isdz, iedz, ismne, stokz
260         FROM afru
261         FOR ALL ENTRIES IN @lt_aufk
262         WHERE aufnr = @lt_aufk-aufnr AND stokz = 'X'
263         INTO TABLE @DATA(lt_afru_x).
264
265         SELECT aufnr, rueck, rmzhl, stzhl, budat, ismnw, isdd, iedd, isdz, iedz, ismne, stokz
266         FROM afru
267         FOR ALL ENTRIES IN @lt_aufk
268         WHERE aufnr = @lt_aufk-aufnr
269         INTO TABLE @lt_afru_all.
270
271     ENDIF.
272     DATA: lt_result TYPE TABLE OF afru.
273
274     LOOP AT lt_afru_all INTO DATA(ls_afru_all).
275
276     DATA(lv_found) = abap_false.
277
278     LOOP AT lt_afru_x INTO DATA(ls_afru_x).
279     IF ls_afru_x-rmzhl = ls_afru_all-stzhl.
280         lv_found = abap_true.
281         EXIT.
282     ENDIF.
283     ENDLOOP.
284
285     IF lv_found = abap_false AND ls_afru_all-stokz IS INITIAL.
286         APPEND ls_afru_all TO lt_result.
287     ENDIF.
288     ENDLOOP.

```

Gambar 3.9 Code Untuk Select Order Confirmation Report Monitoring Work Order Internal

Di sisi biaya, Total GI Amount (DMBTR) dihitung dari tabel MSEG dengan filter pada movement type (BWART 261 = +1, BWART 262 = -1).

```

485     DATA: lv_amt_pos TYPE p DECIMALS 2,
486           lv_amt_neg TYPE p DECIMALS 2.
487
488     SELECT SUM( dmbtr ) INTO lv_amt_pos FROM mseg
489     WHERE aufnr = ls_aufk-aufnr AND bwart = '261'.
490
491     SELECT SUM( dmbtr ) INTO lv_amt_neg FROM mseg
492     WHERE aufnr = ls_aufk-aufnr AND bwart = '262'.
493
494     ty_internal_report-gi_amt = ( lv_amt_pos - lv_amt_neg ) .
495

```

Gambar 3.10 Code Untuk Select dan Logic Movement Type S Untuk Work Order Internal

Alloc WS per Order diambil dari ACDOCA dengan kondisi akun dan nomor Work Order tertentu, kemudian dijumlahkan untuk menghasilkan Total Actual Cost. Actual Cost per Hour dihitung dengan membagi Total Actual Cost dengan Actual Activity Work.

```
319:      SELECT aufnr, hsl
320:      FROM acdoca
321:      FOR ALL ENTRIES IN @lt_aufk
322:      WHERE aufnr = @lt_aufk-aufnr
323:          AND rldnr = '01'
324:          AND racct = '0080000124'
325:      INTO TABLE @lt_alloc_ws.
```

Gambar 3.11 Code Selection ACDOCA Pada Work Order Internal

```
498:      READ TABLE lt_sum INTO ls_sum WITH KEY aufnr = ls_aufk-aufnr.
499:      IF sy-subrc EQ 0.
500:          ty_internal_report-alloc_ws = ls_sum-total_hsl. "lv_alloc_ws. " ALLOC_WS
501:      ENDIF.
```

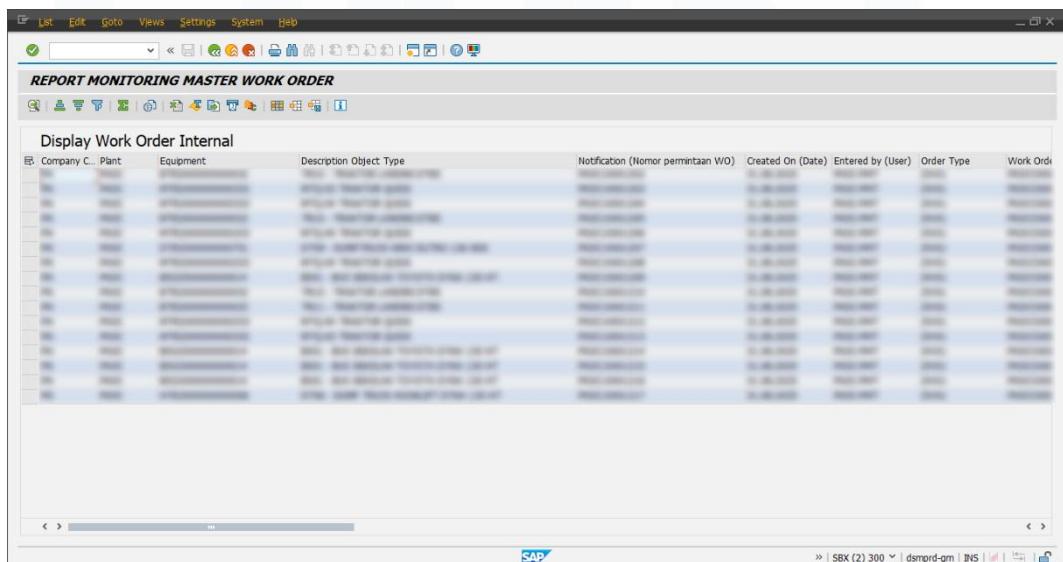
Gambar 3.12 Code Perhitungan Total Actual Cost Pada Work Order Internal

Status Work Order, termasuk TECO/UNTECO, diperoleh melalui join AUFK dengan JEST dan JCDS menggunakan OBJNR untuk mengambil status yang aktif atau tidak, sedangkan Teco Date diperoleh dari JCDS. Informasi terakhir seperti Last Change User dan Change Date diambil dari AUFK melalui AENAM dan AEDAT.

```
515     SELECT inact
516       FROM jest
517      INTO @DATA(lv_inact)
518     WHERE objnr = @lv_objnr
519       AND stat = 'I0045'.
520
521     lv_teco_found = abap_true.
522
523     IF lv_inact = ''." status aktif
524       lv_teco_active = abap_true.
525       EXIT.
526     ENDIF.
527   ENDSELECT.
528
529   IF lv_teco_found = abap_false.
530     ty_internal_report-status = ''." Belum pernah TECO
531   ELSEIF lv_teco_active = abap_true.
532     ty_internal_report-status = 'TECO'.
533   ELSE.
534     ty_internal_report-status = 'UNTECO'.
535   ENDIF.
```

Gambar 3.13 Code Logic Teco dan UnTeco Pada Work Order Internal

Hasil akhir dari report Work Order internal ditampilkan dalam format ALV (ABAP List Viewer), yang menyajikan informasi secara terstruktur mulai dari data header, aktivitas, realisasi tenaga kerja, hingga perhitungan biaya aktual per satuan kerja. Setiap field dan kolom dalam ALV mencerminkan logika yang telah diterapkan sesuai dengan spesifikasi fungsional



Gambar 3.14 Hasil Tampilan Report Work Order Internal dalam ALV

2. Report Monitoring Work Order External

Table 3.4 Spesifikasi yang Diberikan Oleh Functional Untuk Report Monitoring Master Work Order External

No	Column	Table	Field Name	Functional Logic
1	Company Code	AUFK	BUKRS	GET BUKRS FROM AUFK
2	Plant	AUFK	WERKS	GET WERKS FROM AUFK
3	Equipment	EQUI	EQUNR	GET EQUNR FROM EQUI
4	Description Object Type	EQUI	EQKTX	GET EQKTX FROM EQUI
5	Notification Order	QMEL	QMNUM	GET QMNUM FROM QMEL
6	Order Type	AUFK	AUART	GET AUART FROM AUFK
7	Created On	AUFK	ERDAT	GET ERDAT FROM AUFK
8	Work Order	AUFK	AUFNR	GET AUFNR FROM AUFK
9	Description Work Order	AUFK	KTEXT	GET KTEXT FROM AUFK
10	WBS Element Number	AUFK	PSPEL	GET PSPEL FROM AUFK
11	Control Key	AFKO–AFVC	STEUS	GET AUFPL FROM AFKO → JOIN AFVC → GET STEUS Condition: EXT & LCL
12	Activity No	AFKO–AFVC	VORNR	GET AUFPL FROM AFKO → JOIN AFVC → GET VORNR Condition: GET 2 LINE
13	Operation Short Text	AFKO–AFVC	LTXA1	GET AUFPL FROM AFKO → JOIN AFVC → GET LTXA1
14	Plan Start Date	AFKO	GSTRP	GET GSTRP FROM AFKO
15	Plan End Date	AFKO	GLTRP	GET GLTRP FROM AFKO
16	Actual Start Date	EBKN–EBAN–EKKO	BEDAT	GET BANFN FROM EBKN → JOIN EBAN → GET EBELN → JOIN EKKO → GET BEDAT

17	Actual End Date	EBKN-EBAN-ESSR	LBLDT	GET BANFN FROM EBKN → JOIN EBAN → GET EBELN → JOIN ESSR → GET LBLDT
18	SPR Date	EBKN-EBAN	BADAT	GET BANFN FROM EBKN → JOIN EBAN → GET BADAT
19	SPR Item	EBKN	BNFPO	GET BNFPO FROM EBKN
20	SPR Number	EBKN	BANFN	GET BANFN FROM EBKN
21	Total Planning Cost (PR)	EBKN	NETWR	GET NETWR FROM EBKN
22	SPO Item	EBKN-EBAN	BNFPO	GET BANFN FROM EBKN → JOIN EBAN → GET BNFPO
23	Tanggal SPO	EBKN-EBAN	BEDAT	GET BANFN FROM EBKN → JOIN EBAN → GET BEDAT
24	SPO Number	EBKN-EBAN	EBELN	GET BANFN FROM EBKN → JOIN EBAN → GET EBELN
25	Purchasing Org	EBKN-EBAN-EKKO	EKORG	GET BANFN → EBAN → EBELN → JOIN EKKO → GET EKORG
26	Purchasing Org Desc	EBKN-EBAN-EKKO-T024E	EKOTX	GET EKORG → JOIN T024E → GET EKOTX
27	Vendor Code	EBKN-EBAN-EKKO	LIFNR	GET BANFN → EBAN → EBELN → EKKO → GET LIFNR
28	Vendor Name	EBKN-EBAN-EKKO-LFA1	NAME1	GET LIFNR → JOIN LFA1 → GET NAME1

29	Total Amount SPO	EBKN-EBAN-EKPO	NETWR	GET BANFN → EBAN → EBELN → EKPO → GET NETWR
30	GR / SES Date	EBKN-EBAN-ESSR	LBLDT	GET BANFN → EBAN → EBELN → ESSR → GET LBLDT
31	SES Number	EBKN-EBAN-ESSR	LBLNI	GET BANFN → EBAN → EBELN → ESSR → GET LBLNI Condition: LOEKZ = NULL, GET 2 LINE
32	GR Document (SES Posted)	EBKN-EBAN-EKBE	BELNR	JOIN EKBE (BEWTP = E) → GET BELNR Condition: LFBNR COUNT = 1
33	Total Amount SES	EBKN-EBAN-ESSR	NETWR	GET BANFN → EBAN → EBELN → ESSR → GET NETWR
34	Variance SPO – SES	—	—	SPO Amount – SES Amount
35	% SPO VS SES	—	—	SES Amount / SPO Amount × 100%
36	Total GI Amount	MSEG	DMBTR	SUM DMBTR FROM MSEG Condition: BWART 261 = +1, BWART 262 = -1
37	Status Work Order	AUFK-JEST	STAT	JOIN AUFK → JEST → GET STAT (I0045)**INACT = X → UNTECO
38	Last Change TECO/UNTECO User	AUFK	AENAM	GET AENAM FROM AUFK
39	Tanggal LIV	EBKN-EBAN-RSEG-RBKP	BUDAT	GET BANFN → EBAN → EBELN → JOIN RSEG → RBKP → GET BUDAT
40	LIV Number	EBKN-EBAN-RSEG	BELNR	GET BELNR FROM RSEG
41	Clearing Date (Payment)	EBKN-EBAN-RSEG-BSEG-ZXX_XX_XXX_XX-ZXX_XX_XXXX	BUDAT	JOIN chain until PINO → GET BUDAT

42	Payment Amount	EBKN-EBAN- RSEG-BSEG- ZXX_XX_XXXX_XX	WRBTR	GET WRBTR FROM ZAC_FL_PINO_NT
43	Total Actual Cost (GR + SES + GI)	—	—	SUM SES + GI Condition: 1 line only
44	PDO Number	—	—	Pengembangan selanjutnya
45	PDO Amount	—	—	Pengembangan selanjutnya

Pada bagian pertama, pengambilan field dimulai dari tabel AUFK sebagai anchor utama karena AUFK menyimpan seluruh informasi header work order yang menjadi fondasi bagi join ke modul-modul lainnya. Field seperti Company Code (BUKRS), Plant (WERKS), Order Type (AUART), Work Order Number (AUFNR), Description (KTEXT), serta WBS Element (PSPEL) dapat diambil dengan SELECT tunggal berbasis AUFNR, sehingga pada tahap awal perlu menyiapkan internal table berisi daftar work order yang lolos selection screen untuk menjadi key pada seluruh proses data retrieval berikutnya.

```

1134 IF s_equnr[] IS NOT INITIAL.
1135   " Join AUFK with VIAUFKS and AFKO if equipment is selected
1136   SELECT a~aufnr, a~bukrs, a~werks, a~auart, a~erdat, a~ktext,
1137        a~pspel, a~aenam, a~objnr, f~gstrp
1138   FROM aufk AS a
1139   INNER JOIN viaufks AS v
1140     ON a~aufnr = v~aufnr
1141   INNER JOIN afko AS f
1142     ON a~aufnr = f~aufnr
1143   INTO TABLE @lt_aufk
1144   WHERE a~bukrs IN @s_bukrs
1145     AND a~werks IN @s_werks
1146     AND a~auart IN @s_auart
1147     AND a~aufnr IN @s_aufnr
1148     AND v~equnr IN @s_equnr
1149     AND f~gstrp IN @s_gstrp.
1150 ELSE.

```

Gambar 3.15 Code Selection Awal Untuk Work Order External

Tabel AUFK juga dikombinasikan dengan tabel EQUI untuk mendapatkan informasi teknis peralatan yaitu Equipment Number (EQUNR) dan Description Object Type (EQKTX). Karena beberapa order tidak selalu memiliki equipment,

maka diperlukan SELECT terpisah untuk mencegah missing record pada main dataset.

```
1167  IF lt_aufk[] IS NOT INITIAL.
1168    SELECT aufnr, equnr
1169    INTO TABLE @DATA(lt_viaufks)
1170    FROM viaufks
1171    FOR ALL ENTRIES IN @lt_aufk
1172    WHERE aufnr = @lt_aufk-aufnr.
1173
1174  IF lt_viaufks[] IS NOT INITIAL.
1175    SELECT equnr
1176    INTO TABLE @DATA(lt_equnr)
1177    FROM equi
1178    FOR ALL ENTRIES IN @lt_viaufks
1179    WHERE equnr = @lt_viaufks-equnr.
1180  ENDIF.
```

Gambar 3.16 Code Select Equipment Untuk Work Order External

Selanjutnya, bagian operasi kerja diambil melalui kombinasi tabel AFKO dan AFVC, yang membutuhkan join berdasarkan field AUFPL. Pada tahap ini, program harus memastikan bahwa AUFK–AFKO terhubung melalui AUFNR, lalu AFKO–AFVC terhubung melalui AUFPL, sehingga field seperti Control Key (STEUS), Activity Number (VORNR), dan Short Text (LTXA1) dapat dibaca sesuai kebutuhan report.

Khusus untuk Control Key yang memerlukan kondisi seperti EXT & LCL, filter tambahan perlu diterapkan saat SELECT atau setelah data staging untuk memastikan hanya operasi tertentu yang masuk ke dataset utama. Pada beberapa baris, disebutkan syarat GET 2 LINE, yang mengindikasikan bahwa sistem harus mengakomodasi kemungkinan dua jenis operation dalam satu order, sehingga diperlukan merancang internal table dengan kemampuan menampung multiple operations per work order tanpa kehilangan hubungan ke level header. Informasi Plan Start (GSTRP) dan Plan End (GLTRP) dari AFKO menjadi elemen penyusun timeline yang nantinya dibutuhkan saat user melakukan analisa perencanaan vs realisasi.

```

1205      IF lt_afko[] IS NOT INITIAL.
1206          SELECT aufpl, steus, vornr, ltxal
1207              INTO TABLE @DATA(lt_afvc)
1208              FROM afvc
1209              FOR ALL ENTRIES IN @lt_afko
1210              WHERE aufpl = @lt_afko-aufpl
1211                  AND steus IN ( 'EXT', 'LCL' ).
1212      ENDIF.

```

Gambar 3.17 Code Select Table Afko Untuk Work Order External

```

1393      LOOP AT lt_ebkn INTO ls_ebkn WHERE aufnr = ls_aufk-aufnr.
1394
1395          CLEAR: lv_total_spo, lv_total_ses.
1396          MOVE-CORRESPONDING ls_aufk TO ls_output.
1397
1398          READ TABLE lt_viaufks INTO DATA(ls_viaufks) WITH KEY aufnr = ls_aufk-aufnr.
1399          IF sy-subrc EQ 0.
1400              ls_output-equnr = ls_viaufks-equnr.
1401          ENDIF.
1402
1403          READ TABLE lt_afko INTO DATA(ls_afko) WITH KEY aufnr = ls_aufk-aufnr.
1404          IF sy-subrc EQ 0.
1405              ls_output-gstrp = ls_afko-gstrp.
1406              ls_output-gltrp = ls_afko-gltrp.
1407              READ TABLE lt_afvc INTO DATA(ls_afvc) WITH KEY aufpl = ls_afko-aufpl vornr+2(1) = ls_ebkn-bnfp0+4(1)
1408              IF sy-subrc EQ 0.
1409                  ls_output-steus = ls_afvc-steus.
1410                  ls_output-vornr = ls_afvc-vornr.
1411                  ls_output-ltxal = ls_afvc-ltxal.
1412              ENDIF.
1413          ENDIF.

```

Gambar 3.18 Code Pada Work Order External untuk mendapatkan Control Key, dan Activity Number

Setelah bagian operasi, logika berkembang ke area procurement, yang secara teknis merupakan bagian paling kompleks karena melibatkan banyak tabel MM serta hubungan multi-level dari PR → PO → SES → GR → LIV → Payment. Alur dimulai dari EBKN, yaitu tabel linkage antara work order dan purchase requisition. Field BANFN dan BNFPO dari EBKN menjadi entry point untuk masuk ke seluruh proses PR–PO–SES. Dengan BANFN sebagai key, dan melakukan join ke EBAN untuk mendapatkan informasi PR seperti SPR Date (BADAT), SPR Number, serta nilai Total Planning Cost (NETWR).

```

1221 IF lt_ebkn[] IS NOT INITIAL.
1222   " EBAN : PR header+item
1223   SELECT banfn, bnfpo, bedat, badat, ebeln
1224     INTO TABLE @lt_eban
1225     FROM eban
1226     FOR ALL ENTRIES IN @lt_ebkn
1227     WHERE banfn = @lt_ebkn-banfn
1228       AND bnfpo = @lt_ebkn-bnfpo.
1229
1230   " fill empty EBELN from BANFN yang sama
1231   DATA: lt_eban_map TYPE TABLE OF ty_eban_short,
1232         ls_eban_map TYPE ty_eban_short,
1233         ls_eban_fix TYPE ty_eban_short.
1234
1235   LOOP AT lt_eban INTO ls_eban_fix.
1236     IF ls_eban_fix-ebeln IS NOT INITIAL.
1237       READ TABLE lt_eban_map INTO ls_eban_map WITH KEY banfn = ls_eban_fix-banfn.
1238       IF sy-subrc <> 0.
1239         APPEND ls_eban_fix TO lt_eban_map.
1240       ENDIF.
1241     ENDIF.
1242   ENDLOOP.
1243
1244   LOOP AT lt_eban INTO ls_eban_fix.
1245     IF ls_eban_fix-ebeln IS INITIAL.
1246       READ TABLE lt_eban_map INTO ls_eban_map WITH KEY banfn = ls_eban_fix-banfn bnfpo = ls_eban_fix-bnfpo.
1247       IF sy-subrc = 0.
1248         ls_eban_fix-ebeln = ls_eban_map-ebeln.
1249         MODIFY lt_eban FROM ls_eban_fix TRANSPORTING ebeln WHERE banfn = ls_eban_fix-banfn AND bnfpo = ls_eban_fix-bnfpo.
1250       ENDIF.
1251     ENDIF.
1252   ENDLOOP.
1253
1254

```

Gambar 3.19 Code Select Table EBKN dan EBAN Untuk Work Order Internal

```

1439   READ TABLE lt_eban INTO ls_eban WITH KEY banfn = ls_ebkn-banfn bnfpo = ls_ebkn-bnfpo.
1440   IF sy-subrc = 0.
1441     ls_output-ebeln = ls_eban-ebeln.
1442     ls_output-bedat = ls_eban-bedat.
1443     ls_output-badat = ls_eban-badat.
1444     ls_output-bnfpo_spo = ls_eban-bnfpo. " <-- SPO item
1445   ENDIF.
1446
1447   ls_output-banfn = ls_ebkn-banfn.
1448   ls_output-bnfpo = ls_ebkn-bnfpo.
1449   ls_output-netwr = ls_ebkn-netwr.
1450
1451   CLEAR: ls_ekpo.
1452   READ TABLE lt_ekpo INTO ls_ekpo WITH KEY banfn = ls_ebkn-banfn bnfpo = ls_ebkn-bnfpo.
1453   IF sy-subrc = 0.
1454     lv_total_spo = lv_total_spo + ls_ekpo-netwr.

```

Gambar 3.20 Code Pengambilan Data Dari Table EBAN Untuk Work Order Internal

Dari tabel EBAN, program mengambil EBELN yang kemudian digunakan untuk masuk ke tabel EKKO, EKPO, dan tabel purchasing lainnya untuk mendapatkan Purchasing Org (EKORG), Purchasing Org Description (EKOTX via T024E), Vendor Code (LIFNR), Vendor Name (NAME1 via LFA1), serta nilai Total Amount SPO (NETWR di EKPO). Struktur join ini mengharuskan penggunaan key fields yang tepat seperti EBELN, BANFN, BNFPO untuk menghindari duplicate record, serta menerapkan grouping untuk konsolidasi nilai.

```

1213      SELECT banfn, bnfpo, aufnr, netwr
1214      INTO TABLE @lt_ebkn
1215      FROM ebkn
1216      FOR ALL ENTRIES IN @lt_aufk
1217      WHERE aufnr = @lt_aufk-aufnr.
1218      SORT lt_ebkn BY banfn ASCENDING bnfpo ASCENDING.
1219
1220 IF lt_ebkn[] IS NOT INITIAL.
1221   " EBAN : PR header+item
1222   SELECT banfn, bnfpo, bedat, badat, ebeln
1223   INTO TABLE @lt_eban
1224   FROM eban
1225   FOR ALL ENTRIES IN @lt_ebkn
1226   WHERE banfn = @lt_ebkn-banfn
1227   AND bnfpo = @lt_ebkn-bnfpo.

```

Gambar 3.21 Code Join Table EBKN, dan EBAN Pada Work Order External

```

1475 READ TABLE lt_ekpo INTO ls_ekpo WITH KEY banfn = ls_ebkn-banfn bnfpo = ls_ebkn-bnfpo.
1476 IF sy-subrc = 0.
1477   lv_total_spo = lv_total_spo + ls_ekpo-netwr.

```

Gambar 3.22 Code Untuk Mendapatkan Total SPO Pada Work Order External

Proses kemudian masuk ke tahap penerimaan jasa atau service entry melalui tabel ESSR, yang dihubungkan dari EBAN–EKKO melalui EBELN. Field seperti Actual End Date (LBLDT), SES Number (LBLNI), dan Total Amount SES (NETWR) diambil dari ESSR dengan ketentuan khusus seperti LOEKZ = NULL untuk memastikan hanya SES yang aktif yang masuk ke report.

Selain SES, tabel EKBE digunakan untuk membaca GR Document (BELNR) dengan kondisi BEWTP = 'E', yang berarti hanya GR yang berasal dari service entry diambil, dan pengecekan jumlah LFBNR harus memastikan bahwa GR valid hanya ketika jumlahnya satu.

```

1262      " ESSR : SES lines - include EBELP (item) and NETWR
1263      SELECT ebeln, ebelp, lblni, lblldt, netwr, loekz
1264      FROM essr
1265      FOR ALL ENTRIES IN @lt_eban
1266      WHERE ebeln = @lt_eban-ebeln
1267      AND ebelp = @lt_eban-bnfp
1268      AND loekz IS INITIAL
1269      INTO TABLE @lt_essr.
1270
1271      SELECT ebeln, ebelp, belnr, bwart, bewtp, shkzg, menge, lfbnr
1272      FROM ekbe
1273      FOR ALL ENTRIES IN @lt_eban
1274      WHERE ebeln = @lt_eban-ebeln
1275      AND bewtp = 'E'
1276      AND bwart IN ( '101', '102' )
1277      INTO TABLE @DATA(lt_ekbe_shortv2).

```

Gambar 3.23 Code Join ESSR dan EKBE Dengan EBAN Pada Work Order External

Pada kondisi tertentu, field memerlukan GET 2 LINE, sehingga harus memastikan bahwa query tidak collapse ke single row, melainkan dapat menangkap lebih dari satu SES per SPO jika memang terjadi di sistem.

```

1465      IF ls_output-ses_no IS INITIAL.
1466      ls_output-ses_no = ls_essr-lblni.
1467      ELSE.
1468      CONCATENATE ls_output-ses_no ls_essr-lblni INTO ls_output-ses_no SEPARATED BY ', '.
1469      ENDIF.
1470
1471      IF ls_output-gr_date IS INITIAL.
1472      ls_output-gr_date = ls_essr-lbldt.
1473      ls_output-lbldt  = ls_essr-lbldt.
1474      ENDIF.

```

Gambar 3.24 Code Logic Pengambilan Data SES dan SPO Untuk Work Order Internal

Masuk ke tahap financial posting, document flow berlanjut ke RSEG dan RBKP, yang digunakan untuk membaca LIV Date (BUDAT) dan LIV Number (BELNR). Join ini harus memperhatikan kombinasi key BELNR, GJAHR, dan BUZEI agar program tidak salah membaca line item FI. Setelah itu, proses dilanjutkan ke tabel BSEG, ZXX_XX_XXXX_XX, dan ZXX_XX_XXXX untuk menarik informasi Clearing Date (Payment) serta Payment Amount (WRBTR). Karena jalur FI-MM-Z table cukup panjang,

```

1581      IF ls_output-vornr = '0010'.
1582      IF ls_output-liv_no IS NOT INITIAL AND ls_output-bukrs IS NOT INITIAL.
1583
1584          DATA: lv_awkey TYPE awkey,
1585                lv_belnr TYPE belnr_d,
1586                lv_bukrs TYPE bukrs.
1587
1588          CONCATENATE ls_output-liv_no '2025' INTO lv_awkey.
1589
1590          SELECT SINGLE belnr, bukrs
1591              INTO (@lv_belnr, @lv_bukrs)
1592              FROM bseg
1593              WHERE awkey = @lv_awkey
1594                  AND bukrs = @ls_output-bukrs.
1595
1596          IF sy-subrc = 0.
1597              SELECT SINGLE nt-belnr,
1598                  nt-bukrs,
1599                  p-budat,
1600                  nt-wrbtr
1601                  INTO (@ls_payment-belnr,
1602                      @ls_payment-bukrs,
1603                      @ls_payment-clearing_date,
1604                      @ls_payment-payment_amt)
1605                  FROM sac_fi_pino_nt AS nt
1606                  INNER JOIN sac_fi_pino AS p
1607                      ON p-pino = nt-pino
1608                      AND p-bukrs = nt-bukrs
1609                      WHERE nt-belnr = @lv_belnr
1610                          AND nt-bukrs = @lv_bukrs.
1611
1612          IF sy-subrc = 0.
1613              ls_output-clearing_date = ls_payment-clearing_date.
1614              ls_output-payment_amt = ls_payment-payment_amt. " 100.
1615          ENDIF.
1616      ENDIF.
1617  ENDIF.

```

Gambar 3.25 Code Join Untuk Mendapatkan Payment Amount Pada Work Order

External

Pada bagian biaya, tabel MSEG digunakan untuk menghitung Total GI Amount, dengan aturan bahwa movement type 261 bernilai +1 dan 262 bernilai -1, sehingga perlu untuk mengimplementasikan logic penambahan atau pengurang dalam internal table. Nilai Total Actual Cost kemudian dihitung di program sebagai penjumlahan Total SES + Total GI, namun hanya untuk 1 line final output sehingga program harus menggunakan summarization logic LOOP + SUM, atau direct SQL aggregation.

```

1387      SELECT SUM( dmbtr )
1388      INTO @lv_amt_pos
1389      FROM mseg
1390      WHERE aufnr = @ls_aufk-aufnr
1391      AND bwart = '261'.
1392
1393      IF lv_amt_pos IS INITIAL.
1394      lv_amt_pos = 0.
1395      ENDIF.
1396
1397      SELECT SUM( dmbtr )
1398      INTO @lv_amt_neg
1399      FROM mseg
1400      WHERE aufnr = @ls_aufk-aufnr
1401      AND bwart = '262'.
1402
1403      IF lv_amt_neg IS INITIAL.
1404      lv_amt_neg = 0.
1405      ENDIF.
1406
1407      lv total_qi = lv amt pos - lv amt neg.

1445      IF sy-subrc = 0.
1446      lv_gi_vornr = ls_resb_temp-vornr.
1447      ENDIF.
1448
1449      " matching line gi berdasarkan vornr
1450      IF ls_output-vornr = lv_gi_vornr.
1451      ls_output-total_gi = lv_total_gi.
1452      ELSE.
1453      CLEAR ls_output-total_gi.
1454      ENDIF.

```

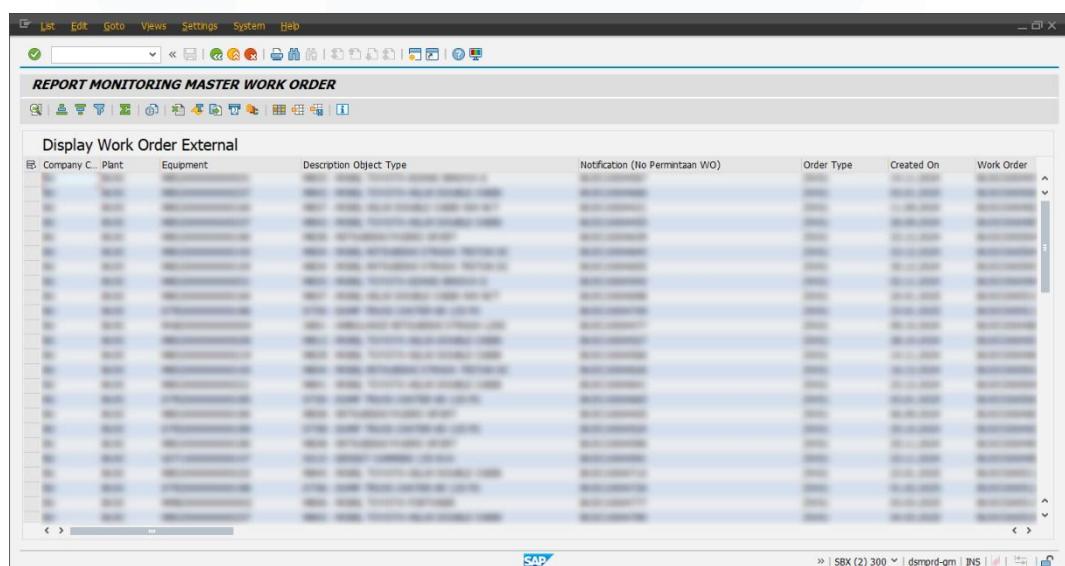
Gambar 3.26 Code Select DMBTR Untuk Work Order External

Akhirnya, status work order diperoleh melalui join AUFK ke JEST untuk membaca status I0045 dengan kondisi INACT = X untuk UNTECO dan INACT kosong untuk TECO, sehingga sistem harus memvalidasi status aktif vs inactive sebelum menentukan status final. Field audit seperti Last Change User (AENAM) diambil langsung dari AUFK.

```
1637     SELECT SINGLE inact
1638         INTO @DATA(lv_inact)
1639         FROM jest
1640         WHERE objnr = @ls_aufk-objnr AND stat = 'I0045'.
1641
1642     IF sy-subrc <> 0.
1643         ls_output-teco_status = ''.
1644     ELSEIF lv_inact = 'X'.
1645         ls_output-teco_status = 'UNTECO'.
1646     ELSE.
1647         ls_output-teco_status = 'TECO'.
1648     ENDIF.
```

Gambar 3.27 Code Untuk Mendapatkan Status TECO Pada Work Order External

Hasil akhir dari report Work Order External juga ditampilkan dalam format ALV.



Gambar 3.28 Hasil Tampilan ALV Pada Work Order External

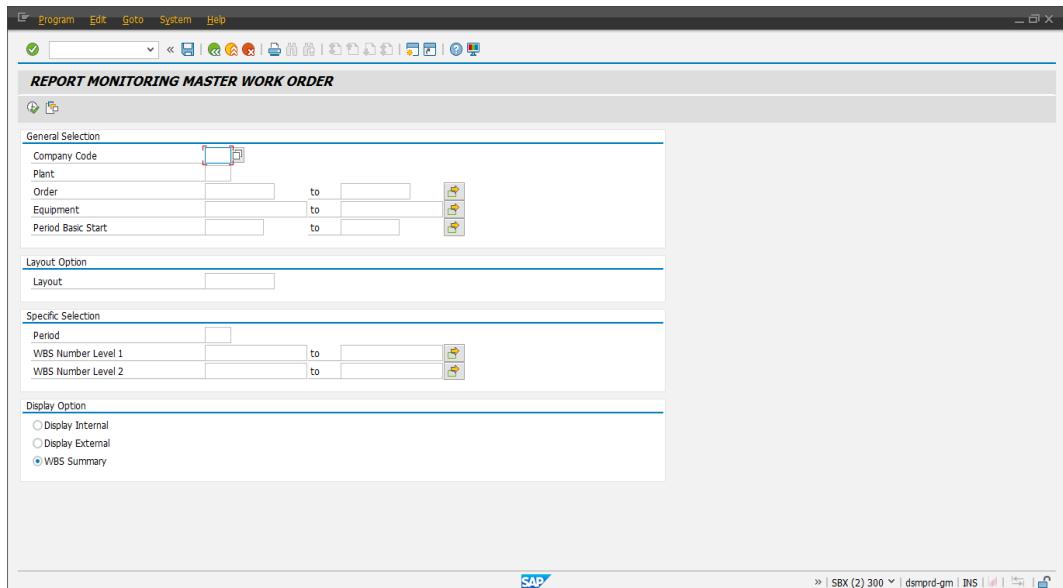
3. Report Monitoring Work Order WBS

Table 3.5 Spesifikasi yang Diberikan Oleh Functional Untuk Report Monitoring Master Work Order WBS

No	Column	Table	Field Name	Functional Logic
1	Plant	AUFK	WERKS	GET WERKS FROM AUFK
2	WBS Element	PRPS	PSPNR	GET PSPNR FROM PRPS (WERKS, POSID R) Condition: Level 1 & 2
3	Description WBS Element	PRPS	POST1	GET POST1 FROM PRPS
4	WBS Budget	PRPS-BPGE	WTGES	GET OBJNR FROM PRPS → JOIN BPGE (WRTTP = 41, LEDNR = 0002)
5	WBS Actual	PRPS-AUFK-ACDOCA	HSL	GET PSPEL (last digit = W,Y,X) → GET OBJNR → JOIN ACDOCA → SUM HSL (RACCT=80020100, RLDNR=0L)
6	Budget vs Actual	—	—	Budget – Actual
7	Budget vs Actual %	—	—	(Actual / Budget) × 100% Condition: without decimals

Tabel 3.3 merupakan spesifikasi struktur data yang digunakan pada laporan WBS Summary, yaitu laporan yang menampilkan informasi anggaran dan realisasi biaya berdasarkan struktur Work Breakdown Structure (WBS) dalam Project System (PS).

Berbeda dengan laporan internal maupun external work order, laporan WBS Summary berfokus pada level WBS, sehingga seluruh data yang disajikan bersumber dari tabel PS dan tabel costing yang berhubungan dengan WBS. Setiap kolom pada tabel ini dirancang untuk memberikan gambaran menyeluruh mengenai posisi anggaran, realisasi aktual, serta persentase pemakaian anggaran pada masing-masing WBS element.



Gambar 3.29 Tampilan Awal Report Ketika User Memilih Opsi WBS Summary

Kolom pertama adalah Plant, yang diambil dari tabel AUFK melalui field WERKS. Selanjutnya, sistem mengambil WBS Element (PSPNR) dari tabel PRPS. Pengambilan WBS dilakukan dengan mempertimbangkan hirarki WBS Level 1 dan Level 2, serta menggunakan POSID yang memiliki struktur karakter tertentu (misalnya diawali huruf R untuk WBS kategori proyek rekayasa). Deskripsi WBS kemudian diperoleh melalui field POST1 dari tabel PRPS.

Untuk menampilkan nilai anggaran WBS, sistem mengambil data WBS Budget melalui join antara PRPS dan BPGE. Mekanismenya dimulai dari identifikasi OBJNR milik WBS, kemudian dilanjutkan dengan mengambil nilai anggaran (WTGES) pada BPGE dengan kondisi WRTTP = 41 (Original Budget) dan LEDNR = 0002 (ledger budgeting).

Sementara itu, nilai WBS Actual dihitung dengan mengambil data realisasi biaya yang telah tercatat pada tabel ACDOCA. Prosesnya dimulai dengan membaca nilai PSPEL dari AUFK yang berakhiran W, Y, atau X — penanda bahwa order tersebut merupakan bagian dari struktur WBS tertentu. Setelah OBJNR dari order terkumpul, sistem melakukan join ke ACDOCA kemudian menghitung SUM(HSL) dengan filter RACCT = 80020100 dan RLDNR = '0L', sehingga nilai aktual yang

diambil hanya berasal dari akun realisasi biaya yang sesuai dengan struktur keuangan perusahaan.

```
2225  SELECT rldnr,rbukrs, gjahr, belnr, objnr, hsl
2226    INTO TABLE @lt_acdoca
2227    FROM acdoca
2228    FOR ALL ENTRIES IN @lt_aufk
2229    WHERE objnr = @lt_aufk-objnr
2230    AND rldnr = 'OL'
2231    AND racct = '0080020100'.
2232
2233    SORT lt_acdoca BY objnr ASCENDING.
2234
2235  IF lt_prps[] IS NOT INITIAL.
2236    SELECT objnr, wtges
2237      INTO TABLE @lt_bpge
2238      FROM bpge
2239      FOR ALL ENTRIES IN @lt_prps
2240      WHERE objnr = @lt_prps-objnr
2241      AND wrtpp = '41'
2242      AND lednr = '0002'
2243      AND vorga = 'KBUD'.
2244  ENDIF.
```

Gambar 3.30 Code Untuk Mendapatkan nilai WBS Budget dan Actual Pada Work Order WBS

Kolom berikutnya adalah Budget vs Actual, yaitu hasil perhitungan selisih antara nilai anggaran dan aktual. Kolom ini digunakan untuk melihat sisa anggaran yang masih tersedia pada WBS. Selain itu, laporan juga menampilkan Budget vs Actual % yang dihitung dengan rumus $(\text{Actual} / \text{Budget}) \times 100\%$.

```

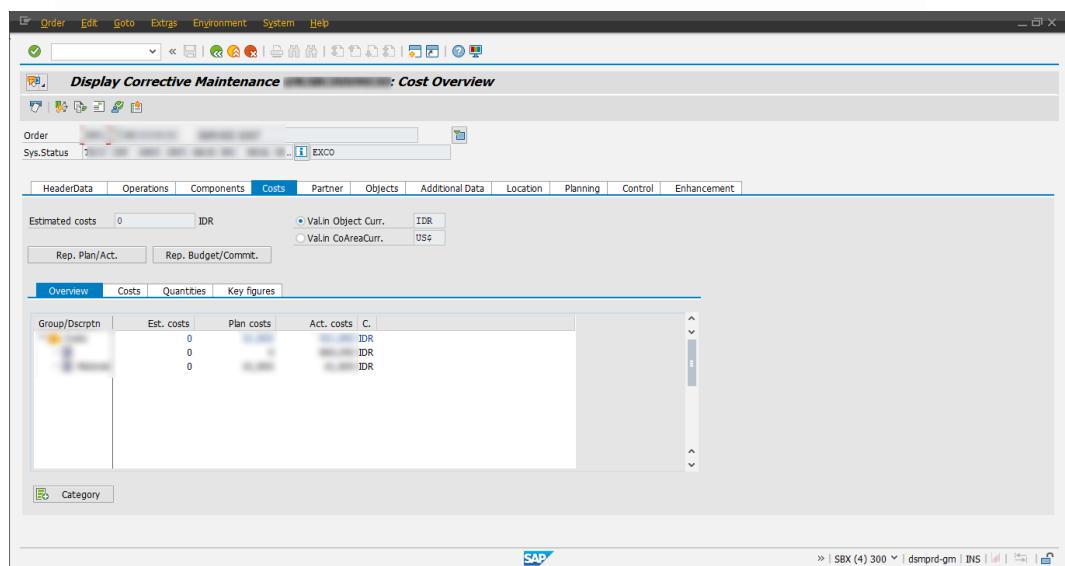
2322      "Actual
2323      ls_wbs_data-actual = abs( lv_hsl ).
2324      ls_wbs_data-diff_amt = ls_wbs_data-budget - ls_wbs_data-actual.
2325
2326      "Percentase
2327      IF ls_wbs_data-budget NE 0.
2328          x = ( ls_wbs_data-actual / ls_wbs_data-budget ) * 100.
2329          CALL FUNCTION 'ROUND'
2330              EXPORTING
2331                  decimals = 1
2332                  input    = x
2333              IMPORTING
2334                  output   = ls_wbs_data-diff_pct.
2335      ELSE.
2336          ls_wbs_data-diff_pct = 0.
2337      ENDIF.

```

Gambar 3.31 Code Untuk Mendapatkan Selisih Budget dan Actual Pada Work Order

External

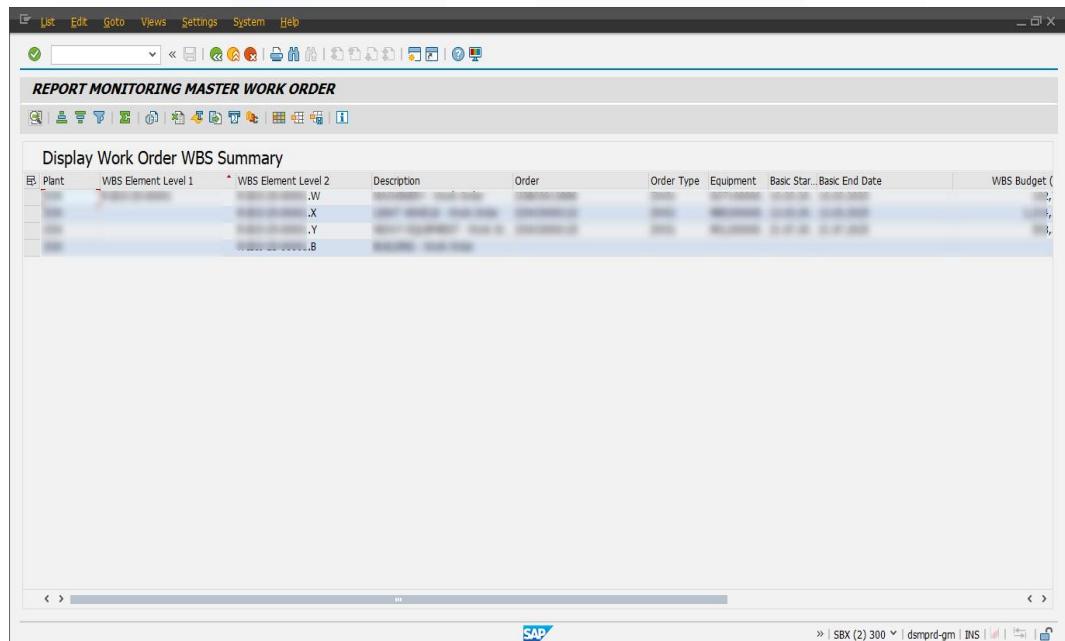
Data hasil actual dapat di check di dalam tcode IW33 untuk memastikan data yang di ambil sudah sesuai atau tidak



Gambar 3.32 Tampilan IW33 Untuk Pengecheckan Data Actual Pada Work Order

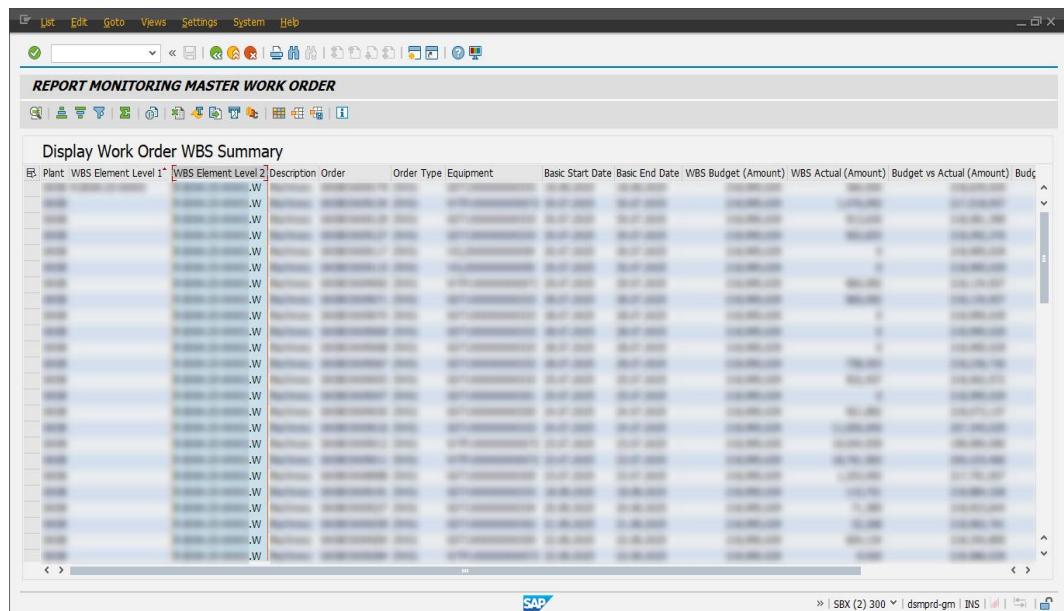
External

Dalam report ini, jika user memasukkan WBS Level 1 pada selection screen, sistem akan menampilkan semua WBS Level 2 yang berada di bawah Level 1 tersebut, memberikan ringkasan hierarki WBS secara menyeluruh.



Gambar 3.33 Hasil Tampilan ALV Pada Work Order WBS Jika User Mengisi Kolom WBS Level 1

Sebaliknya, apabila user memasukkan WBS Level 2, laporan akan menampilkan detail seluruh order yang terkait dengan WBS Level 2 tersebut, sehingga pengguna dapat menelusuri realisasi biaya hingga ke level order.



Gambar 3.34 Hasil Tampilan ALV Pada Work Order WBS Jika User Mengisi Kolom WBS Level 2

4. Scenario Testing

Scenario Testing dikembangkan sebagai panduan penting bagi programmer maupun konsultan fungsional untuk melakukan unit testing pada report Work Order. Tujuan utama dari skenario testing adalah memastikan bahwa program dapat menangani seluruh kemungkinan kondisi yang mungkin muncul selama eksekusi, sehingga output yang dihasilkan konsisten dan sesuai dengan kebutuhan bisnis. Skenario ini mencakup dua jenis pengujian utama.

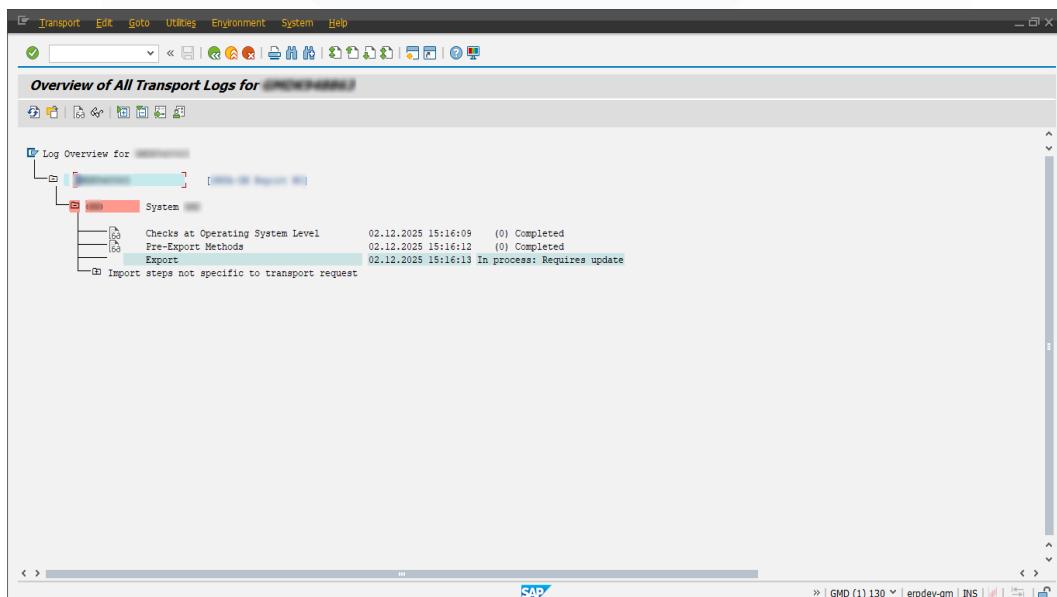
Pertama, Tes Positif, yaitu pengujian di mana program seharusnya memberikan hasil yang diharapkan ketika pengguna memasukkan nilai yang benar dan valid, sehingga validasi perhitungan biaya, aktivitas, dan status operasional dapat diverifikasi. Kedua,

Tes Negatif, yaitu pengujian di mana program seharusnya menampilkan pesan error atau warning kepada pengguna jika terjadi input yang salah atau data yang tidak konsisten. Tes ini diwajibkan untuk memastikan bahwa program cukup kuat untuk mencegah keluarnya hasil yang tidak diharapkan.

Selain itu, skenario testing juga dapat diperluas untuk mengakomodasi logika cancelation, yaitu kondisi di mana Work Order dibatalkan atau dikoreksi. Sehingga pengujian logika cancelation, program dapat menyesuaikan perhitungan biaya, realisasi aktivitas, dan status operasional secara otomatis.

5. Pembuatan Transport Request

Transport Request merupakan tahap terakhir dalam siklus pengembangan report Work Order sebelum program dapat digunakan di sistem produksi. Setelah pengembangan selesai, logika program telah diuji melalui scenario testing, dan hasilnya sesuai dengan spesifikasi fungsional, ABAPer membuat Transport Request (TR) untuk memindahkan objek program dari lingkungan pengembangan (development system) ke sistem kualitas (quality system) dan selanjutnya ke produksi (production system). Transport request ini berisi seluruh objek ABAP yang terkait, termasuk program report, ALV layout, dan tabel kustom yang digunakan.



Gambar 3.35 Transport Request SAP

Proses deployment dilakukan menggunakan Transaction Code (TCode) SE09 atau SE10, yang digunakan untuk membuat, memeriksa, dan me-release transport request.

3.3.3 Enhancement Report Monitoring Progress Kredit Investasi

Report Monitoring Progress Kredit Investasi adalah laporan yang berfungsi untuk memonitoring semua AR yang sudah diajukan untuk Kredit Investasi dimana setiap AR yang sudah diajukan atau mendapatkan nomor Kredit Investasi tersebut menampilkan nilai budget AR tersebut, nilai presentase dari Kredit Investasi atas AR tersebut, nilai plafon dari AR tersebut baik sebelum pajak maupun sudah terkena pajak, nilai commitment dari PO yang sudah dijalankan dari AR tersebut dan lain-lain.

Functional Design Specification

Title: Enhancement ZPS_KI_RPT

RICEF ID	ZPS_KI_RPT
Transaction Code (jika ada)	
Application Module	IMPS
Nama Penulis	
Tanggal Pembuatan	8 Oktober 2025
Prioritas	<input checked="" type="checkbox"/> Tinggi <input type="checkbox"/> Sedang <input type="checkbox"/> Rendah
Penilaian Kompleksitas	<input type="checkbox"/> Tinggi <input type="checkbox"/> Sedang <input type="checkbox"/> Rendah
Jenis Proses	<input type="checkbox"/> Online <input checked="" type="checkbox"/> Batch <input type="checkbox"/> Lainnya :
Periode pemasesan	<input type="checkbox"/> Harian <input type="checkbox"/> Mingguan <input type="checkbox"/> Bulanan <input type="checkbox"/> Tahunan <input type="checkbox"/> Lainnya:
Perencanaan Tanggal Jatuh Tempo	
Catatan	Report Monitoring Progress Kredit Investasi

Persetujuan/ Sign off Spesifikasi

Peran	Nama	Tanggal	Tanda Tangan
Functional			
Functional Coordinator			
Project Manager			
Business Process Owner (Upstream)			

Gambar 3.36 Sample FSD Enchantment Report Monitoring Progress Kredit Investasi

Dalam Enhancement yang diberikan, terdapat beberapa penambahan dan perbaikan report seperti penambahan radio button, field selection screen, dan penambahan kolom.

1. Penambahan Radio Button Untuk Filter KI, Non KI, dan All

Pada proses Enchantment Report Monitoring Progress Kredit Investasi ditambahkan fitur radio button “KI” sebagai filter untuk menampilkan data khusus Credit Investment (KI). Apabila pengguna

memilih opsi KI dan mengeksekusi report, sistem akan menampilkan seluruh nomor AR (Account Receivable) yang telah berstatus full release serta hanya menampilkan data AR dengan status KI.

```
171  "tambahan radio butoon ki dan non ki
172  SELECTION-SCREEN BEGIN OF BLOCK b2 WITH FRAME TITLE TEXT-002.
173  SELECTION-SCREEN BEGIN OF LINE.
174  PARAMETERS: rb_ki RADIobutton GROUP rgl DEFAULT 'X'.
175  SELECTION-SCREEN COMMENT 3(10) FOR FIELD rb_ki.
176  PARAMETERS: rb_nki RADIobutton GROUP rgl.
177  SELECTION-SCREEN COMMENT 18(10) FOR FIELD rb_nki.
178  PARAMETERS: rb_all RADIobutton GROUP rgl.
179  SELECTION-SCREEN COMMENT 33(10) FOR FIELD rb_all.
180  SELECTION-SCREEN END OF LINE.
181  SELECTION-SCREEN END OF BLOCK b2.

280  --- Filter rb button KI / NON KI / ALL
281  IF rb_ki = 'X'.
282  DELETE it_data WHERE invcrst <> 'KI'.
283  ELSEIF rb_nki = 'X'.
284  DELETE it_data WHERE invcrst = 'KI'.
285  ENDIF.
```

Gambar 3.37 Code Selection Screen Radio Button Report Monitoring Progress Kredit
Investasi

2. Penambahan Kolom AR Date dan Ar Status pada tampilan awal Report.

Dalam proses Enhancement Report Monitoring Progress Kredit Investasi, terdapat juga request penambahan kolom AR Date dan AR Status pada tampilan awal report. Kolom AR Date mengambil data dari tabel IMAK pada field ERDAT, Sementara itu, kolom AR Status diambil dari logika program ZXX_XXXXXX, yang menampilkan status terkini dari setiap AR.

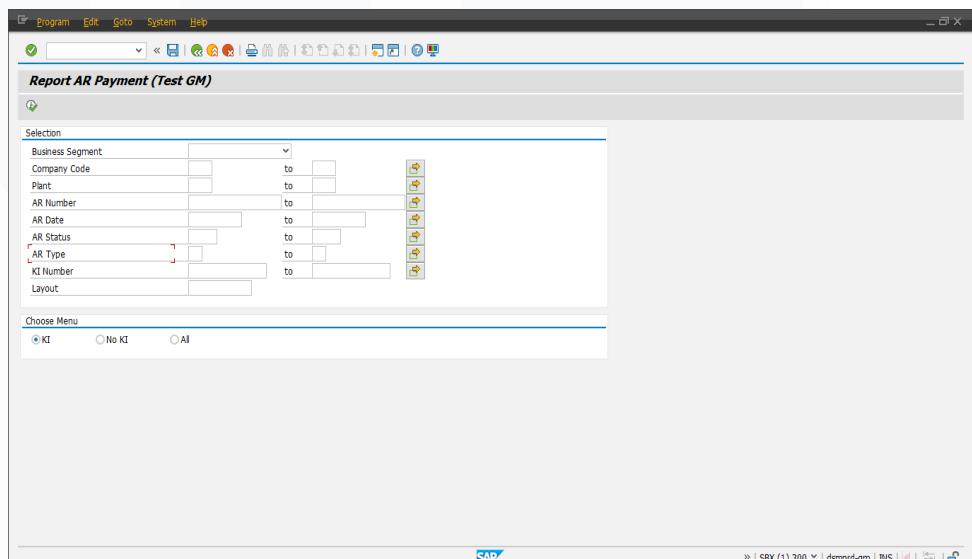
```

156  SELECTION-SCREEN BEGIN OF BLOCK bl WITH FRAME TITLE TEXT-001.
157  |  PARAMETERS : p_segmnt TYPE zbusstyp AS LISTBOX VISIBLE LENGTH 20 USER-COMMAND cmd10.
158  |  SELECT-OPTIONS : s_bukrs FOR so_bukrs.
159  |  SELECT-OPTIONS : s_werks FOR so_werks.
160  |  SELECT-OPTIONS : s_posid FOR so_posid.
161  |  SELECT-OPTIONS : s_erdat FOR imak-erdat. " AR DATE
162  |  SELECT-OPTIONS : s_estat FOR tj30-estat MATCHCODE OBJECT zh_ar_stat. "AR STATUS
163  |  SELECT-OPTIONS : s_ivart FOR taiflt-ivart.
164  |  SELECT-OPTIONS : s_kinum FOR so_kinum.
165  |  *PARAMETERS : p_ffp TYPE char1 RADIobutton GROUP g1 DEFAULT 'X'.
166  |  *PARAMETERS : p_ofp TYPE char1 RADIobutton GROUP g1.
167  |  PARAMETERS : p_layout TYPE slis_vari.
168  SELECTION-SCREEN END OF BLOCK bl.

```

Gambar 3.38 Code Selection Screen AR Date dan AR Status Dalam Report Monitoring

Progress Kredit Investasi



Gambar 3.39 Tampilan Selection Screen setelah Enchantment Pada Report Monitoring

Progress Kredit Investasi

3. Perubahan Tampilan Report dan Penambahan Kolom

Proses Enhancement Report Monitoring Progress Kredit Investasi, dilakukan beberapa penyesuaian pada tampilan report dan juga penambahan kolom.

Pertama, beberapa kolom mengalami perubahan judul agar lebih sesuai dengan informasi yang ditampilkan, seperti perubahan Comt Amt(Exc Tax) menjadi Comt Amount (DPP), Actual AR Amt(Exc Tax) menjadi Act AR Amount, Invoice Posted menjadi Invoice Posted (DPP), dan PINO Amt(IDR) Exc PP menjadi PINO Amt(IDR) ExPPH.

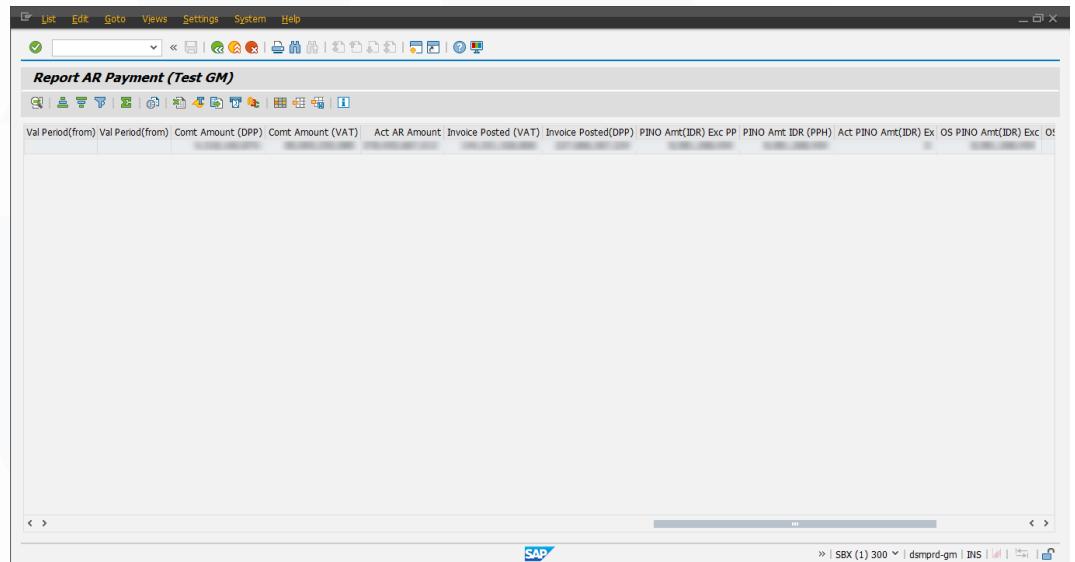
```
922 LOOP AT it_calc INTO wa_calc.
923   IF wa_calc-refdoc = 'PO'.
924     CLEAR wa_ekbe_tmp2.
925     CLEAR lv_netwr_gr.
926     CLEAR v_dec.
927
928   LOOP AT it_ekbe_tmp2 INTO wa_ekbe_tmp2 WHERE docnum = wa_calc-docnum AND
929         docitem = wa_calc-docitem.
930
931   lv_netwr_gr += wa_ekbe_tmp2-netwr2.
932
933 ENDLOOP.
934
935   "Comt Amt (VAT)
936   p_gs_alv_comt_amt_vat = p_gs_alv_comt_amt_vat + wa_calc-effwr.
937
938   p_gs_alv_comt_amt += wa_calc-netwr2.
939   p_gs_alv_comt_amt = p_gs_alv_comt_amt - lv_netwr_gr.
940
941   wa_calc-otsval = wa_calc-netwr2 - lv_netwr_gr.
942
943   MODIFY it_calc FROM wa_calc TRANSPORTING otsval WHERE docnum = wa_calc-docnum AND docitem = wa_calc-docitem.
944 ENDIF.
945
ENDLOOP.
```

Gambar 3.40 Code Logic Konversi Currency Untuk Field Comt Amount (DPP)

The screenshot shows a SAP Fiori application interface. The top navigation bar includes 'List', 'Edit', 'Goto', 'Views', 'Settings', 'System', and 'Help'. Below the navigation is a toolbar with various icons for search, refresh, and document management. The main title is 'Comt Amount (DPP)'. The application displays a table with the following columns: 'AR Code', 'Doc. Number', 'Doc.Item', 'Value', 'Outstanding Value', 'Curr', and 'Ref. Doc'. The 'Outstanding Value' column is highlighted with a red border. The 'Value' and 'Outstanding Value' columns contain a red 'x' character. The 'Curr' column has a red border and contains the value 'IDR'. The bottom status bar shows the SAP logo and the text '» | SBX (1) 300 | dsmod-prm | INS | 1/1 | 100% |'.

Gambar 3.41 Hasil Perbaikan Field Comt Amount (DPP) (Expanded)

Kedua, dilakukan penambahan kolom baru untuk menampilkan informasi tambahan, seperti Comt Amt (VAT), Invoice Posted (VAT), dan PINO Amt(IDR) (PPH).



Gambar 3.42 Hasil Perbaikan dan Penambahan Field Pada Report Radio Button Report Monitoring Progress Kredit Investasi

3.4 Kendala yang Ditemukan

Selama praktik kerja magang, beberapa kendala yang memengaruhi proses pengembangan sistem SAP ABAP antara lain:

1. Pengetahuan mengenai fundamental dan ABAP development di SAP masih minim karena di perkuliahan tidak membahas modul secara spesifik. Hal ini menyebabkan proses pengembangan memakan waktu lebih lama dari perkiraan.
2. Deadline yang diberikan kurang jelas karena tim menginginkan penggerjaan “secepatnya” tanpa adanya timeline yang spesifik. Meskipun di dalam FSD terdapat estimasi Perencanaan Tanggal Jatuh Tempo, tim functional terkadang tidak mengisinya sehingga Kondisi ini menyulitkan perencanaan dan pengaturan prioritas pekerjaan.

3. Penambahan atau perubahan fitur dari tim functional kadang menyebabkan struktur kode yang sudah dibuat menjadi tidak sesuai. Hal ini membutuhkan penggerjaan ulang atau penyesuaian tambahan pada kode yang sudah ada.
4. Keterbatasan pemahaman tentang detail proses bisnis menyebabkan beberapa permintaan dari tim fungsional kadang diterima secara mentah-mentah tanpa evaluasi lebih dulu. Hal ini berpotensi menghasilkan request yang sulit atau bahkan tidak memungkinkan secara teknis.
5. Meskipun sudah mencoba mencari solusi secara mandiri, proses pencarian sering memakan waktu lama, dan terkadang hasil yang ditemukan bukan best practice sehingga menyebabkan proses pengembangan menjadi kurang efisien dan berisiko menghasilkan implementasi yang tidak optimal.

3.5 Solusi atas Kendala yang Ditemukan

Berdasarkan skendala yang dihadapi, berikut merupakan solusi yang diterapkan selama masa magang untuk mengatasi kendala yang ada:

1. Meningkatkan pengetahuan ABAP dan fundamental SAP dengan memanfaatkan dokumentasi resmi SAP, modul online, dan bimbingan dari mentor atau supervisor.
2. Menetapkan deadline pribadi untuk menyelesaikan tugas, sehingga pekerjaan lebih terstruktur dan prioritas dapat diatur meskipun timeline dari tim tidak jelas.
3. Menjalin komunikasi yang intens dengan tim functional sebelum dan selama pengembangan untuk meminimalkan perubahan mendadak dan memastikan requirement lebih stabil.

4. Mengevaluasi permintaan dari tim functional secara teknis sebelum diterapkan, serta berkonsultasi dengan supervisor atau rekan kerja untuk memastikan implementasi sesuai standar SAP.
5. Mencoba mencari solusi secara mandiri dengan batas waktu tertentu. Jika solusi belum ditemukan saat waktu habis, langsung berkonsultasi dengan supervisor atau rekan kerja untuk mendapatkan arahan yang tepat dan efisien.

