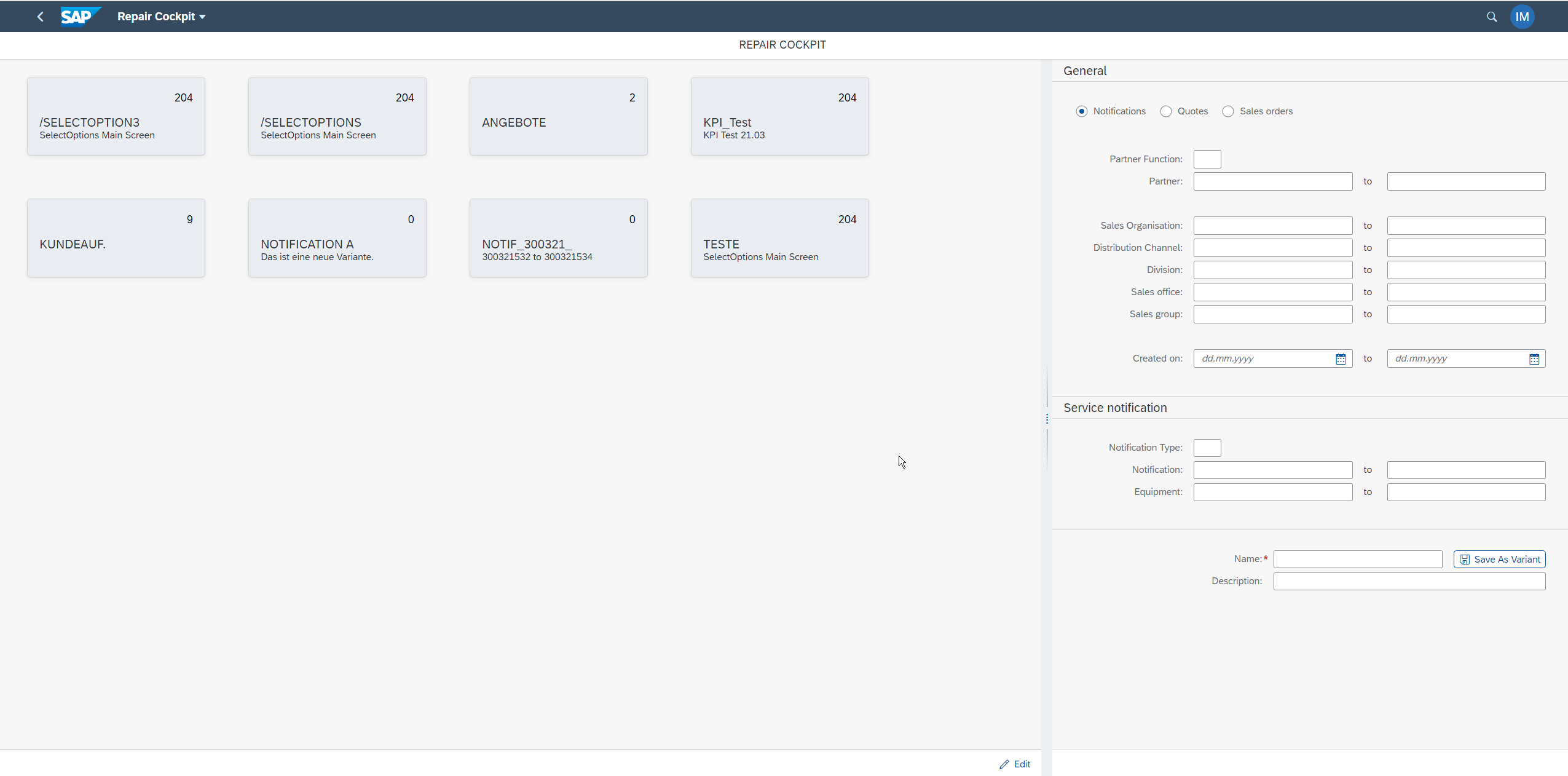
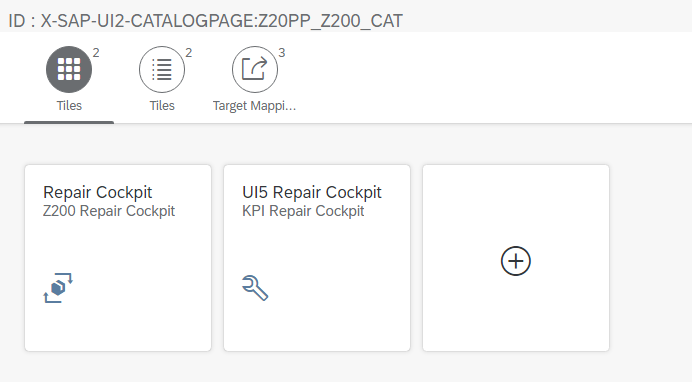
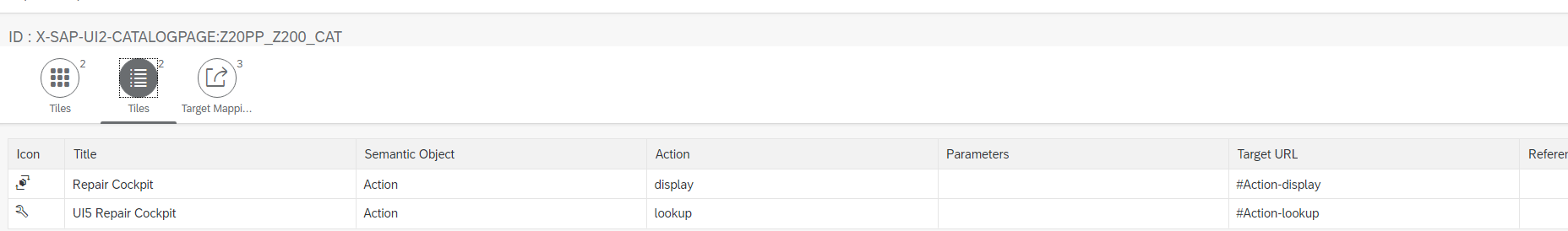
Call Transaction in SAP passing parameters and select options

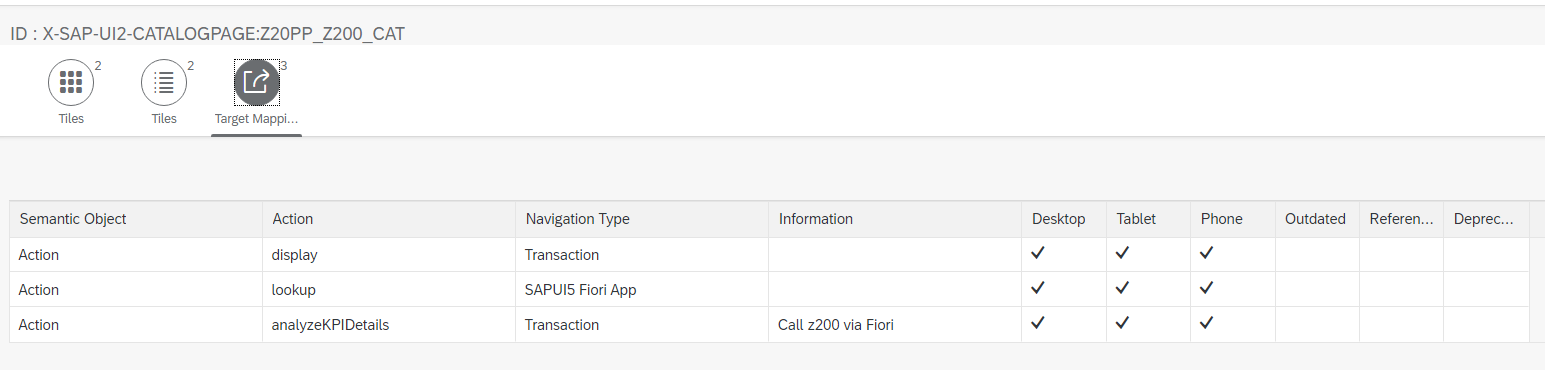


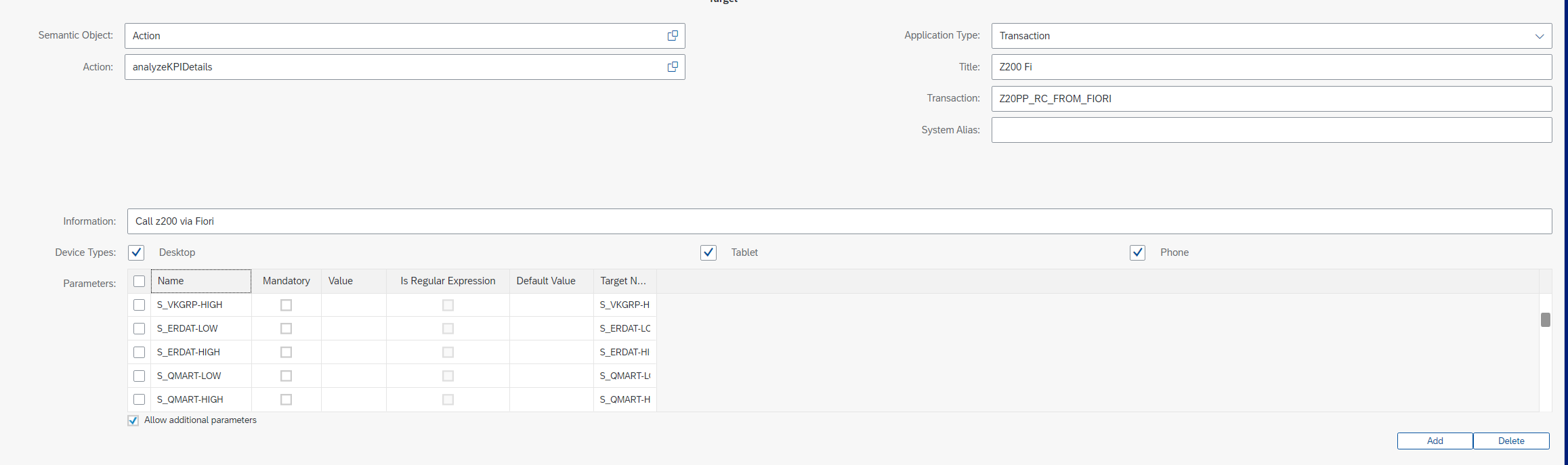
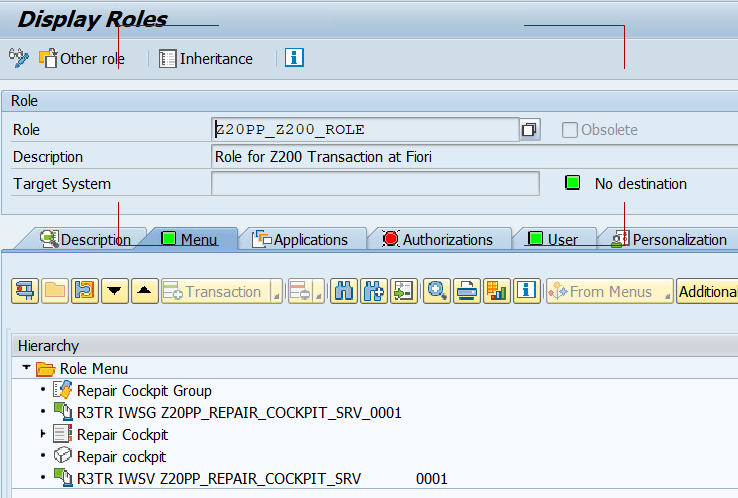


|  |
| --- |
| \*&---------------------------------------------------------------------\* \*& REPORT Z20PP\_RC\_FROM\_FIORI \*&---------------------------------------------------------------------\* \*& \*&---------------------------------------------------------------------\* REPORT z20pp\_rc\_from\_fiori.  TABLES: prps, vbak, vbap, vbep, qmel, equi, caufv.  DATA: gv\_parnr    TYPE i\_parnr.  SELECTION-SCREEN BEGIN OF BLOCK b100 WITH FRAME TITLE TEXT-b01.   PARAMETERS: p\_rbmld    RADIOBUTTON GROUP g1 USER-COMMAND rb             , p\_rbang    RADIOBUTTON GROUP g1             , p\_rbkda    RADIOBUTTON GROUP g1             .   PARAMETERS: p\_xpall TYPE z20pp\_z200\_pall AS CHECKBOX DEFAULT ' '.   SELECTION-SCREEN ULINE. \*  PARAMETERS: P\_NRART    TYPE NRART   PARAMETERS: p\_parvw    TYPE ihpa-parvw               .   SELECT-OPTIONS: s\_parnr  FOR gv\_parnr MATCHCODE OBJECT debi.   SELECTION-SCREEN ULINE.   SELECT-OPTIONS: s\_vkorg FOR vbak-vkorg                 , s\_vtweg FOR vbak-vtweg                 , s\_spart FOR vbak-spart                 , s\_vkbur FOR vbak-vkbur                 , s\_vkgrp FOR vbak-vkgrp                 .   SELECTION-SCREEN ULINE.   SELECT-OPTIONS: s\_erdat FOR vbak-erdat. SELECTION-SCREEN END OF BLOCK b100.  " MELDUNG SELECTION-SCREEN: BEGIN OF BLOCK b200 WITH FRAME TITLE TEXT-b02.   SELECT-OPTIONS: s\_qmart FOR qmel-qmart MODIF ID mld                                           NO INTERVALS                 , s\_qmnum FOR qmel-qmnum MODIF ID mld                 , s\_equnr FOR equi-equnr MODIF ID mld.   .   PARAMETERS: p\_xmmab    AS CHECKBOX MODIF ID mld.  SELECTION-SCREEN: END OF BLOCK b200.  " ANGEBOT SELECTION-SCREEN: BEGIN OF BLOCK b300 WITH FRAME TITLE TEXT-b03.   SELECT-OPTIONS: s\_auart    FOR vbak-auart MODIF ID ang   , s\_vbeln    FOR vbak-vbeln              MODIF ID ang   , s\_samat    FOR vbap-matnr              MODIF ID ang   , s\_sages    FOR vbak-gbstk NO INTERVALS MODIF ID ang   , s\_saabs    FOR vbak-abstk NO INTERVALS MODIF ID ang   , s\_sakre    FOR vbak-cmgst NO INTERVALS MODIF ID ang   , s\_safak    FOR vbak-fksak NO INTERVALS MODIF ID ang   , s\_saspe    FOR vbak-spstg NO INTERVALS MODIF ID ang   . SELECTION-SCREEN: END OF BLOCK b300.  " SD-AUFTRAG SELECTION-SCREEN: BEGIN OF BLOCK b400 WITH FRAME TITLE TEXT-b04.   SELECT-OPTIONS: s\_soart    FOR vbak-auart MODIF ID kda                 , s\_sovbn    FOR vbak-vbeln MODIF ID kda                 , s\_somat    FOR vbap-matnr MODIF ID kda                 , s\_soges    FOR vbak-gbstk NO INTERVALS MODIF ID kda                 , s\_soabs    FOR vbak-abstk NO INTERVALS MODIF ID kda                 , s\_solif    FOR vbak-lfstk NO INTERVALS MODIF ID kda                 , s\_sokre    FOR vbak-cmgst NO INTERVALS MODIF ID kda                 , s\_sofak    FOR vbak-fksak NO INTERVALS MODIF ID kda                 , s\_sospe    FOR vbak-spstg NO INTERVALS MODIF ID kda                 , s\_edatu    FOR vbep-edatu MODIF ID kda                 , s\_lprio    FOR vbap-lprio MODIF ID kda                 , s\_werks    FOR vbap-werks MODIF ID kda                 , s\_dispo    FOR caufv-dispo MODIF ID kda                 .   PARAMETERS: p\_xiha AS CHECKBOX  MODIF ID kda. SELECTION-SCREEN: END OF BLOCK b400.  " LAYOUTS SELECTION-SCREEN: BEGIN OF BLOCK b500 WITH FRAME TITLE TEXT-b05.    PARAMETERS: p\_vatre TYPE slis\_vari           VISIBLE LENGTH 20             , p\_vasal TYPE slis\_vari           VISIBLE LENGTH 20             , p\_vamat TYPE slis\_vari           VISIBLE LENGTH 20             , p\_vaact TYPE slis\_vari           VISIBLE LENGTH 20             , p\_xexp  TYPE xfeld             .  SELECTION-SCREEN: END OF BLOCK b500.   START-OF-SELECTION.    "THROUGH THE TARGET MAPPING WE RECEIVE THE VALUES FROM THE FIORI APP AND CALL THE TRANSACTION BY USING THE SUBMIT    DATA: lt\_pass\_tab TYPE TABLE OF rsparams,         ls\_pass\_tab TYPE rsparams.    "PARAMETERS   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_RBMLD' kind = 'P' sign = 'I' option = 'EQ' low  = p\_rbmld ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_RBANG' kind = 'P' sign = 'I' option = 'EQ' low  = p\_rbang ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_RBKDA' kind = 'P' sign = 'I' option = 'EQ' low  = p\_rbkda ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_XPALL' kind = 'P' sign = 'I' option = 'EQ' low  = p\_xpall ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_PARVW' kind = 'P' sign = 'I' option = 'EQ' low  = p\_parvw ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_XMMAB' kind = 'P' sign = 'I' option = 'EQ' low  = p\_xmmab ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_XIHA'  kind = 'P' sign = 'I' option = 'EQ' low  = p\_xiha ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_VATRE' kind = 'P' sign = 'I' option = 'EQ' low  = p\_vatre ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_VASAL' kind = 'P' sign = 'I' option = 'EQ' low  = p\_vasal ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_VAMAT' kind = 'P' sign = 'I' option = 'EQ' low  = p\_vamat ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_VAACT' kind = 'P' sign = 'I' option = 'EQ' low  = p\_vaact ) ).   lt\_pass\_tab = VALUE #( BASE lt\_pass\_tab ( selname = 'P\_XEXP'  kind = 'P' sign = 'I' option = 'EQ' low  = p\_xexp ) ).    "SELECT OPTIONS   lt\_pass\_tab = COND #( WHEN lines( s\_parnr[] ) GT 0 THEN                          VALUE #( BASE lt\_pass\_tab ( selname = 'S\_PARNR' kind = 'S'                          sign = s\_parnr[ 1 ]-sign option = s\_parnr[ 1 ]-option                          low  = s\_parnr[ 1 ]-low  high = s\_parnr[ 1 ]-high  ) ) ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_vkorg[] ) GT 0 THEN                        VALUE #( BASE lt\_pass\_tab ( selname = 'S\_VKORG' kind = 'S'                        sign = s\_vkorg[ 1 ]-sign option = s\_vkorg[ 1 ]-option                        low  = s\_vkorg[ 1 ]-low  high = s\_vkorg[ 1 ]-high  ) ) ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_vtweg[] ) GT 0 THEN                        VALUE #( BASE lt\_pass\_tab ( selname = 'S\_VTWEG' kind = 'S'                        sign = s\_vtweg[ 1 ]-sign option = s\_vtweg[ 1 ]-option                        low  = s\_vtweg[ 1 ]-low  high = s\_vtweg[ 1 ]-high  ) )  ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_spart[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SPART' kind = 'S'                      sign = s\_spart[ 1 ]-sign option = s\_spart[ 1 ]-option                      low  = s\_spart[ 1 ]-low  high = s\_spart[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_vkbur[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_VKBUR' kind = 'S'                      sign = s\_vkbur[ 1 ]-sign option = s\_vkbur[ 1 ]-option                      low  = s\_vkbur[ 1 ]-low  high = s\_vkbur[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_vkgrp[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_VKGRP' kind = 'S'                      sign = s\_vkgrp[ 1 ]-sign option = s\_vkgrp[ 1 ]-option                      low  = s\_vkgrp[ 1 ]-low  high = s\_vkgrp[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_qmart[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_QMART' kind = 'S'                      sign = s\_qmart[ 1 ]-sign option = s\_qmart[ 1 ]-option                      low  = s\_qmart[ 1 ]-low  high = s\_qmart[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_qmnum[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_QMNUM' kind = 'S'                      sign = s\_qmnum[ 1 ]-sign option = s\_qmnum[ 1 ]-option                      low  = s\_qmnum[ 1 ]-low  high = s\_qmnum[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_equnr[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_EQUNR' kind = 'S'                      sign = s\_equnr[ 1 ]-sign option = s\_equnr[ 1 ]-option                      low  = s\_equnr[ 1 ]-low  high = s\_equnr[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_auart[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_AUART' kind = 'S'                      sign = s\_auart[ 1 ]-sign option = s\_auart[ 1 ]-option                      low  = s\_auart[ 1 ]-low  high = s\_auart[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_vbeln[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_VBELN' kind = 'S'                      sign = s\_vbeln[ 1 ]-sign option = s\_vbeln[ 1 ]-option                      low  = s\_vbeln[ 1 ]-low  high = s\_vbeln[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_samat[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SAMAT' kind = 'S'                      sign = s\_samat[ 1 ]-sign option = s\_samat[ 1 ]-option                      low  = s\_samat[ 1 ]-low  high = s\_samat[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_sages[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SAGES' kind = 'S'                      sign = s\_sages[ 1 ]-sign option = s\_sages[ 1 ]-option                      low  = s\_sages[ 1 ]-low  high = s\_sages[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_saabs[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SAABS' kind = 'S'                      sign = s\_saabs[ 1 ]-sign option = s\_saabs[ 1 ]-option                      low  = s\_saabs[ 1 ]-low  high = s\_saabs[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_sakre[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_VKORG' kind = 'S'                      sign = s\_sakre[ 1 ]-sign option = s\_sakre[ 1 ]-option                      low  = s\_sakre[ 1 ]-low  high = s\_sakre[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_safak[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SAFAK' kind = 'S'                      sign = s\_safak[ 1 ]-sign option = s\_safak[ 1 ]-option                      low  = s\_safak[ 1 ]-low  high = s\_safak[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_saspe[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SASPE' kind = 'S'                      sign = s\_saspe[ 1 ]-sign option = s\_saspe[ 1 ]-option                      low  = s\_saspe[ 1 ]-low  high = s\_saspe[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_soart[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOART' kind = 'S'                      sign = s\_soart[ 1 ]-sign option = s\_soart[ 1 ]-option                      low  = s\_soart[ 1 ]-low  high = s\_soart[ 1 ]-high  ) )   ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_sovbn[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOVBN' kind = 'S'                      sign = s\_sovbn[ 1 ]-sign option = s\_sovbn[ 1 ]-option                      low  = s\_sovbn[ 1 ]-low  high = s\_sovbn[ 1 ]-high  ) )   ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_somat[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOMAT' kind = 'S'                      sign = s\_somat[ 1 ]-sign option = s\_somat[ 1 ]-option                      low  = s\_somat[ 1 ]-low  high = s\_somat[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_soges[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOGES' kind = 'S'                      sign = s\_soges[ 1 ]-sign option = s\_soges[ 1 ]-option                      low  = s\_soges[ 1 ]-low  high = s\_soges[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_soabs[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOABS' kind = 'S'                      sign = s\_soabs[ 1 ]-sign option = s\_soabs[ 1 ]-option                      low  = s\_soabs[ 1 ]-low  high = s\_soabs[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_solif[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOLIF' kind = 'S'                      sign = s\_solif[ 1 ]-sign option = s\_solif[ 1 ]-option                      low  = s\_solif[ 1 ]-low  high = s\_solif[ 1 ]-high  ) )  ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_sokre[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOKRE' kind = 'S'                      sign = s\_sokre[ 1 ]-sign option = s\_sokre[ 1 ]-option                      low  = s\_sokre[ 1 ]-low  high = s\_sokre[ 1 ]-high  ) )   ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_sofak[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOFAK' kind = 'S'                      sign = s\_sofak[ 1 ]-sign option = s\_sofak[ 1 ]-option                      low  = s\_sofak[ 1 ]-low  high = s\_sofak[ 1 ]-high  ) )  ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_sospe[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_SOSPE' kind = 'S'                      sign = s\_sospe[ 1 ]-sign option = s\_sospe[ 1 ]-option                      low  = s\_sospe[ 1 ]-low  high = s\_sospe[ 1 ]-high  ) ) ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_edatu[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_EDATU' kind = 'S'                      sign = s\_edatu[ 1 ]-sign option = s\_edatu[ 1 ]-option                      low  = s\_edatu[ 1 ]-low  high = s\_edatu[ 1 ]-high  ) ) ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_lprio[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_LPRIO' kind = 'S'                      sign = s\_lprio[ 1 ]-sign option = s\_lprio[ 1 ]-option                      low  = s\_lprio[ 1 ]-low  high = s\_lprio[ 1 ]-high  ) ) ELSE lt\_pass\_tab  ).    lt\_pass\_tab = COND #( WHEN lines( s\_werks[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_WERKS' kind = 'S'                      sign = s\_werks[ 1 ]-sign option = s\_werks[ 1 ]-option                      low  = s\_werks[ 1 ]-low  high = s\_werks[ 1 ]-high  ) )  ELSE lt\_pass\_tab ).    lt\_pass\_tab = COND #( WHEN lines( s\_dispo[] ) GT 0 THEN                      VALUE #( BASE lt\_pass\_tab ( selname = 'S\_DISPO' kind = 'S'                      sign = s\_dispo[ 1 ]-sign option = s\_dispo[ 1 ]-option                      low  = s\_dispo[ 1 ]-low  high = s\_dispo[ 1 ]-high  ) ) ELSE lt\_pass\_tab  ).    SUBMIT z20pp\_repair\_cockpit WITH SELECTION-TABLE lt\_pass\_tab. |







|  |
| --- |
| onPress: function(oEvt) {              if (!(sap.ushell && sap.ushell.Container)) {                  console.error("Navigation only available in Launchpad.");                  return;              }              const oSource = oEvt.getSource();              const oData = oSource.getBindingContext().getObject({expand:"ToKPI"})              const oParams = oData.ToKPI.reduce((aKPI, oKPI) => {                      if (oKPI.Kind === "P") {                          if (oKPI.Low) {                              aKPI[oKPI.Selname] = oKPI.Low;                          }                      } else if (oKPI.Kind === "S") {                          if (oKPI.Low) {                              aKPI[`${oKPI.Selname}-LOW`] = oKPI.Low;                          }                          if (oKPI.High) {                              aKPI[`${oKPI.Selname}-HIGH`] = oKPI.High;                          }                      }                      return aKPI;                  }, {});              sap.ushell.Container.getServiceAsync("CrossApplicationNavigation")                  .then(oService => {                      if (!oService) {                          return;                      }                      return oService.toExternal({                          target: {                              semanticObject: "Action",                              action: "analyzeKPIDetails"                          },                          params: oParams                      });                  });          }, |