**0) App → OnStart (load collections + computed fields)**

Adjust the data sources to your table names. If you already have flat KPITable, you can still use the collections; otherwise point them to your real tables/views.

// Periods (e.g., FY + Quarter)

ClearCollect(

colPeriod,

SortByColumns(

Distinct( KPITable, FiscalYear ), "Result", Descending

)

);

ClearCollect(

colPeriodDetail,

SortByColumns(

AddColumns(

KPITable,

PeriodLabel, Text(FiscalYear) & " " & Coalesce(FiscalQuarter, "")

),

"FiscalYear", Descending

)

);

// Pillars

ClearCollect(

colPillar,

SortByColumns(

Distinct(KPITable, Pillar),

"Result",

Ascending

)

);

// add display/order fields if you have a lookup table

// AddColumns(colPillar, PillarID, ..., PillarOrder, ...)

// Objectives (assuming KPITable has Objective fields)

ClearCollect(

colObjective,

SortByColumns(

Distinct(KPITable, Objective),

"Result",

Ascending

)

);

// Priorities (H/M/L or numeric)

ClearCollect(

colPriority,

SortByColumns(

Distinct(KPITable, Priority),

"Result",

Ascending

)

);

// KPIs

ClearCollect(

colKPI,

AddColumns(

KPITable,

PerfPct,

If(Value(KPI\_Targets)=0, Blank(), Value(KPI\_Actuals)/Value(KPI\_Targets)),

Status,

Switch(

true,

Value(KPI\_Actuals) >= Value(KPI\_Targets), "Green",

Value(KPI\_Actuals) >= Value(KPI\_Targets)\*0.8, "Yellow",

"Red"

)

)

);

Run **File → Run OnStart** once.

**1) Top-left: Period selector (gallery or dropdown)**

**If using a Dropdown (ddPeriod):**

* **Items**

SortByColumns(

Distinct(AddColumns(colPeriodDetail, PLabel, PeriodLabel), PLabel),

"PLabel",

Descending

)

* **DisplayFields**: ["PLabel"]

*(That’s what your line meant:)*  
SortByColumns(colPeriod, "FiscalYear", Descending) and display “FiscalYear FiscalQuarter”.

**2) Left gallery = Pillars (galPillars)**

* **Items**

SortByColumns(

AddColumns(colPillar, PillarName, Result), // Result is the Distinct() output

"PillarName",

Ascending

)

* Template label:  
  Text = ThisItem.PillarName

*(If you have a Pillar order field, use your original line:)*  
SortByColumns(colPillar, "PillarOrder", Ascending)

**3) Middle-left gallery = Objectives (galObjectives)**

* **Items**

SortByColumns(

Filter(

AddColumns(colObjective, ObjectiveTitle, Result),

// filter by selected pillar

!IsBlank(galPillars.Selected) &&

LookUp(KPITable, Objective = Result && Pillar = galPillars.Selected.PillarName, true)

),

"ObjectiveTitle",

Ascending

)

* Template label:

ThisItem.ObjectiveTitle

*(If you store IDs, use your original pattern with PillarID.)*

**4) Middle-right gallery = Priorities (galPriorities)**

* **Items**

SortByColumns(

Filter(

AddColumns(colPriority, PriorityTitle, Result),

!IsBlank(galObjectives.Selected) &&

LookUp(

KPITable,

Priority = Result && Objective = galObjectives.Selected.ObjectiveTitle &&

Pillar = galPillars.Selected.PillarName,

true

)

),

"PriorityTitle",

Ascending

)

* Template label:  
  ThisItem.PriorityTitle

**5) Right gallery = KPIs (galKPIs)**

* **Items**

Filter(

colKPI,

// by Period (optional)

(IsBlank(ddPeriod.Selected) || ddPeriod.Selected.PLabel in [Text(FiscalYear) & " " & Coalesce(FiscalQuarter,"")]),

// by Pillar

(IsBlank(galPillars.Selected) || Pillar = galPillars.Selected.PillarName),

// by Objective

(IsBlank(galObjectives.Selected) || Objective = galObjectives.Selected.ObjectiveTitle),

// by Priority

(IsBlank(galPriorities.Selected) || Priority = galPriorities.Selected.PriorityTitle)

)

**Inside the KPI template, add:**

* **KPI name** label  
  Text = ThisItem.KPI
* **Meta** label (small)  
  Text = ThisItem.Pillar & " • " & ThisItem.Objective & " • " & ThisItem.Priority
* **Target vs Actuals** label (right-aligned)

Text =

"Actual: " & Text(Value(ThisItem.KPI\_Actuals),"#,##0.##") &

" Target: " & Text(Value(ThisItem.KPI\_Targets),"#,##0.##")

* **Traffic-light** circle shpLight

Fill =

Switch(

ThisItem.Status,

"Green", Color.Green,

"Yellow", Color.Yellow,

"Red", Color.Red,

Color.Gray

)

* **Progress bar** (two rectangles: track + progress)
  + rectTrack.Fill = RGBA(230,230,230,1)
  + rectProgress.Width = rectTrack.Width \* Min(1, Max(0, ThisItem.PerfPct))
  + rectProgress.Fill = shpLight.Fill (or same Switch as above)
* **OnSelect** (set a global for detail panel):

Set(varSelectedKPI, ThisItem)

**6) Detail panel (bind to varSelectedKPI)**

* Title: varSelectedKPI.KPI
* Lines:

"Objective: " & varSelectedKPI.Objective & Char(10) &

"Pillar: " & varSelectedKPI.Pillar & Char(10) &

"Priority: " & varSelectedKPI.Priority & Char(10) &

"Period: " & Text(varSelectedKPI.FiscalYear) & " " & Coalesce(varSelectedKPI.FiscalQuarter,"") & Char(10) &

"Target vs Actual: " &

Text(Value(varSelectedKPI.KPI\_Targets),"#,##0.##") & " / " &

Text(Value(varSelectedKPI.KPI\_Actuals),"#,##0.##")

* Status chip fill (small label): same Switch on varSelectedKPI.Status.
* If you have multiple rows per KPI over time, a small **line chart**:
  + **Items**
* SortByColumns(
* Filter(colKPI, KPI = varSelectedKPI.KPI && Pillar = varSelectedKPI.Pillar),
* "FiscalYear",
* Ascending
* )
  + X: FiscalYear (or a combined date), Y: KPI\_Actuals.

**Notes / Tips**

* If your data is **flat in one table** (KPITable) this works as-is.  
  If you have **normalized lookups** (Pillar table, Objective table…), replace the Distinct() collections with those real tables and keep the same gallery wiring.
* To avoid empty right-hand galleries at app start, you can **preselect** first items:
* // Screen.OnVisible
* Select( First(galPillars.AllItems) );
* Select( First(galObjectives.AllItems) );
* Select( First(galPriorities.AllItems) );

If you share your exact column names for **FiscalYear/FiscalQuarter** (or if you use a single TimePeriod text), I’ll tweak those few lines to match perfectly.