SAP Best Practices

SAP Fiori Design (U55)

Building Block Configuration Guide

Copyright

Icons

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Typographic Conventions

Type Style	Description
Example text	Words or characters that appear on the screen. These include field names, screen titles, pushbuttons as well as menu names, paths and options. Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, titles of graphics and tables.
EXAMPLE TEXT	Names of elements in the system. These include report names, program names, transaction codes, table names, and individual key words of a programming language, when surrounded by body text, for example, SELECT and INCLUDE.
Example text	Screen output. This includes file and directory names and their paths, messages, source code, names of variables and parameters as well as names of installation, upgrade and database tools.
EXAMPLE TEXT	Keys on the keyboard, for example, function keys (such as F2) or the ENTER key.
Example text	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Pointed brackets indicate that you replace these words and characters with appropriate entries.

Content

Figures

SAP Fiori Design: Configuration Guide

1 Purpose

The purpose of this document is to describe the best practices in the user experience (UX) according to SAP Fiori design. You can use these best practices to design and develop your own user interfaces incorporating the look-and-feel of SAP Fiori. This document outlines the

- basic principles of the SAP Fiori user experience;
- framework for the SAP Fiori layout;
- layout of selected app types based upon the SAP Fiori principles;
- controls and patterns that are used in these app types.

This document is aimed equally at UX designers as well as developers of user interfaces.

> For a complete checklist of items relevant for SAP Fiori refer to the configuration guide *SAP Fiori Checklist* (building block U58). This checklist can be used as a reference source during UX design and development and as a verification tool during the sign-off process.

> This document concentrates on use cases for transactional app types that can be run using any database. This may be on SAP NetWeaver 7.31 or SAP NetWeaver 7.4 (with an SAP HANA data base or another data base). This document does not provide any guidance on use cases that specifically require an SAP HANA database.

SAP Fiori Guidelines provides comprehensive information for designing and developing SAP Fiori–designed apps.

2 Preparation

2.1 Prerequisites

Before working with this document, you should be familiar with topics covered in preceding building blocks. The preceding building blocks are listed in the section *Solution Overview* of the *Configuration Guide to Implementing SAP Fiori Design Rapid-Deployment Solution V2*.

This guide provides best practices for designing and developing user interfaces according to SAP Fiori principles. As such this document does not have specific pre-requisites. However, SAP Fiori user interfaces are built using the SAP UI Development Toolkit for HTML5 (also referred to as SAPUI5). Therefore developers should certainly be familiar with the SAPUI5 toolkit. This is covered in the configuration guide *SAPUI5 General Configuration* (building block U50). This document also includes links to further SAPUI5 documentation.

> Technical documentation for SAPUI5 is available in the *UI development toolkit for HTML5 – Demo Kit*.

3 SAP Fiori Design

The following sections provide in-depth information on the following topics:

- SAP Fiori Principles

This describes the principles that have been identified as central to the UX for SAP Fiori and that are included in every user interface that is developed according to SAP Fiori.

- App Framework

    This describes the set of user interface patterns that have been specified for SAP Fiori. These patterns embody the principles for the SAP Fiori UX and can be used as a template for developing further user interfaces.

- Application Types

    This describes selected types of apps for SAP Fiori design, according to specific use cases.

- Controls

    This section describes selected controls that are within the most commonly used app types.

- Patterns

    This section describes selected patterns that are within the most commonly used app types.

Additionally the Appendix provides detailed information on specifications for topics such as typography, colors, and icons.

3.1 SAP Fiori Principles

3.1.1   People Centric Design

SAP Fiori-designed apps are not structured along the transactions that users carry out. Instead, the SAP Fiori paradigm focuses on people and their goals when using the apps. This requires a very different approach to design, which is fully described in *From Transactions to Apps*.

3.1.2   Five Core Principles

SAP Fiori has five core principles. Each user interface must include all of the following principles in order to be regarded as designed according to SAP Fiori:

- Role-based
- Responsive
- Simple
- Coherent
- Delightful

SAP Fiori UX Design Principles
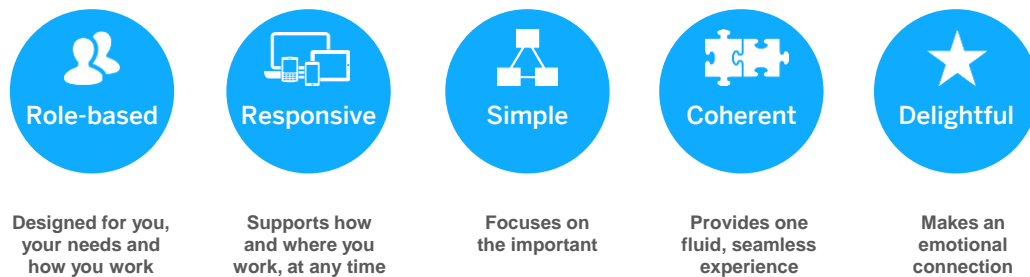
Figure 1: SAP Fiori UX Design Principles

Fiori Design Principles illustrates the main aspects of these principles, which are explained in more depth in the following sections.

3.1.2.1   Role-based

The user interfaces cover the most frequently used activities that a user carries out. The user interfaces are designed to match how people work, rather than forcing people to change the way they work to fit the user interface.

3.1.2.2   Responsive

The user interface is designed and developed for a consistent experience on many devices.

A responsive user interface caters for

- Multiple form factors, for smartphone, tablet and desktop;
- Multiple interaction styles, for touch or keyboard;
- Multiple technologies, for iOS, Android, Windows, Mac, or PC;

Moreover this is achieved with a single code-line.

3.1.2.3   Simple

The user interface is focused on user experience. This is more important than including multiple features and functions.

A simple user interface covers one use case for one user with a maximum of three screens. This makes the important aspects of each use case really easy to learn and use.

Coverage of all or most of the tasks that a user has to carry out is achieved by many user interfaces for individual use cases, not one complicated user interface that attempts to cover all functions.

3.1.2.4   Coherent

User interfaces provide the same user experience across all use cases. This is achieved by identifying best practices and incorporating these in all user interfaces.

3.1.2.5   Delightful

User interfaces offer low barriers to adoption. User interfaces which are delightful are

- Easy to adopt: the user interfaces can be run on the releases with largest installed base with no upgrade;
- Easy to adapt: the user interfaces can be themed, personalized, and extended;
- Easy to deploy: the user interface has no footprint and requires little administration;
- Easy to introduce: the user interface requires no training;
- Easy to grow: the portfolio of user interfaces can be added to by SAP and partners.

3.2 App Framework

The design principles for SAP Fiori have been taken by SAP to determine best practices for user interface layout for different types of use cases using SAP Fiori. SAP Fiori has a simple user interface hierarchy. The launchpad is the entry point for the user, providing entry points to the user to their processes.

The user interfaces for one business process are referred to as an app. All the user's apps are presented in the form of tiles on the Homepage, which forms the heart of the Launchpad. Most apps' designs are based on one of the two basic templates:

- Master/Detail;
- Full Screen.

> In the context of this document the term app refers to the user interface and the related software to conduct a specific business process within an SAP system. These apps are multi-channel, therefore although they can be run on a smartphone or tablet, this should not be confused with an app dedicated to be downloaded onto and used on a mobile device.

Within each app there may be one or more screens (according to the simple principle, a maximum of three screens). For each pattern

- the layout of each screen

and

- the screen order

is defined according to the best practices for SAP Fiori.

3.2.1   Full Screen

The full screen template affords maximum flexibility while still providing the look and feel of SAP Fiori.

The template for the full screen does not present a pre-defined hierarchy of content and thus provides a higher degree of flexibility. UI elements inside the template can be arranged individually based on the app's use cases.
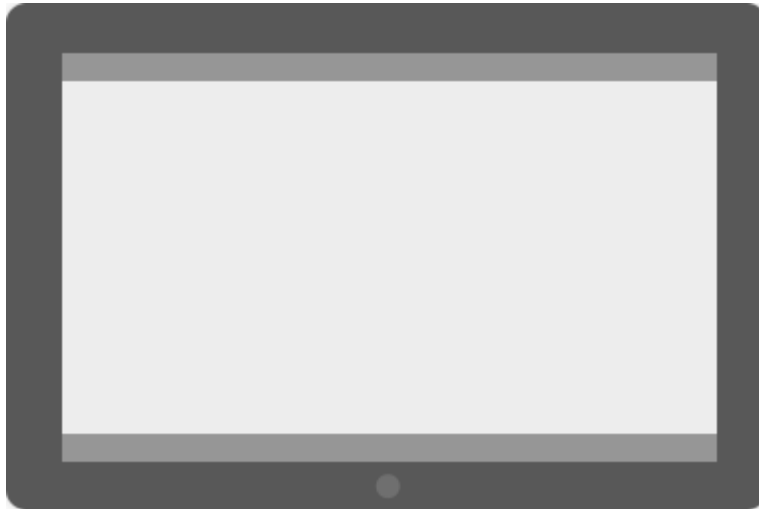
Figure 2: Full Screen

Subsequent pages within the full screen template can also make use of the master/detail template.

Figure 3: Full Screen Template Sequence

The content area of the full screen pattern is the main part of the full screen pattern, situated between below the page header and above the footer toolbar. There are no rules on what to display or what not to display on the content area. Nevertheless, rules might be provided by the app type. For example, for a tracking app, a tab bar is required to show the status.

> In many cases, you will be providing a phone version of the app as well. This is important to bear in mind during the design and development process, particularly if you are displaying a table in the content area.

> For further information on the full screen pattern, refer to *Full Screen* in the *SAP Fiori Guidelines*.

3.2.2 Master/Detail

The master/detail template consists of a list that shows a number of items of which the details are shown on selection. This template is used in several apps. It offers a smart layout and is responsive to different form factors and orientation.

The template for master/detail consists of a master list on the left-hand side and a details area on the right-hand side of the screen.

Figure 4: Master/Detail

On phones, the master list and the detail page feature on two separate screens. The detail page can include items such as list items. By clicking (or tapping) on an item on the detail page, a line item detail page can be displayed, which shows the details of the item in the details area whilst retaining the master list display. The master/detail template must contain the detail page/list; the line item detail page is optional.

As a result the master/detail template can consist of a sequence of up to four pages:

- Launchpad;
- Master List;
- Detail Page/List;
- Line Item Detail page (optional).

Figure 5: Master/Detail Template Sequence

> For further information on the master/detailpattern, refer to *Full Screen* in the *SAP Fiori Guidelines*.

3.2.3   When to Use Which Template?

When deciding which template to use for your app, you will need to consider what the user wants to achieve when working in the app. For this it is helpful to answer questions on the nature of the tasks. The outcome will provide guidance on what type of template to use.

Question	Outcome
Is viewing, inspecting or editing details one or several elements from a list of elements an important use-case?	If so, use the master/detail pattern, for example a tracking app.

Question	Outcome
Is inspecting the status of one or more objects important?	If so you should use a full screen app with an icon tab bar.
Are the objects you are inspecting so complex that you require charts to illustrate a point fast?	If so you should use a full screen app with charts.

3.2.4   Page Header and Footer Toolbar

Both the master/detail pattern and the full screen pattern are anchored in a basic framework using the page header and the footer toolbar. The page header indicates which page is being displayed and focuses on navigation services. The footer toolbar primarily contains all currently available actions that can be triggered. These two patterns are part of every SAP Fiori-designed app that is based on either the master/detail or full screen pattern.

Currently there are only two apps which do not rely on those templates:

- Factsheets (which has the SAP HANA database as a pre-requisite and is therefore not in scope for this document);
- News App.

When you create a new SAP Fiori-designed app, use the master/detail or full screen pattern, and include the page header and footer toolbar as a basic framework.

Full information on the page header and footer toolbar is available in *Basic Framework*.

3.2.5   Launchpad and Home Page

A launchpad provides entry points to apps. A launchpad contains several tiles that are arranged centered on the screen. The number of tiles that are visible on the screen is limited by the screen resolution. If more tiles are included on the launchpad than fit on one screen, an overflow mechanism is automatically provided.

Tiles are included in the launchpad screen using the Tile control.

For information on including the Tile control according to SAP Fiori principles refer to Tile.

3.2.5.1   What is the Difference between a Launchpad and a Homepage?

A Launchpad is the technical entry point for one or more SAP Fiori apps. A launchpad contains one or more tiles for the apps.

From a technical point-of-view, an app cannot be launched directly with a URL address. A URL address navigates to a launchpad, from which an app can be launched by selecting the respective tile.

A Homepage is the first page that a user sees after they log into a system. It acts as their entry point to their work tasks using SAP Fiori applications. In a Home Page the tiles are usually arranged thematically, for example: *My Home* or *Working Space*.
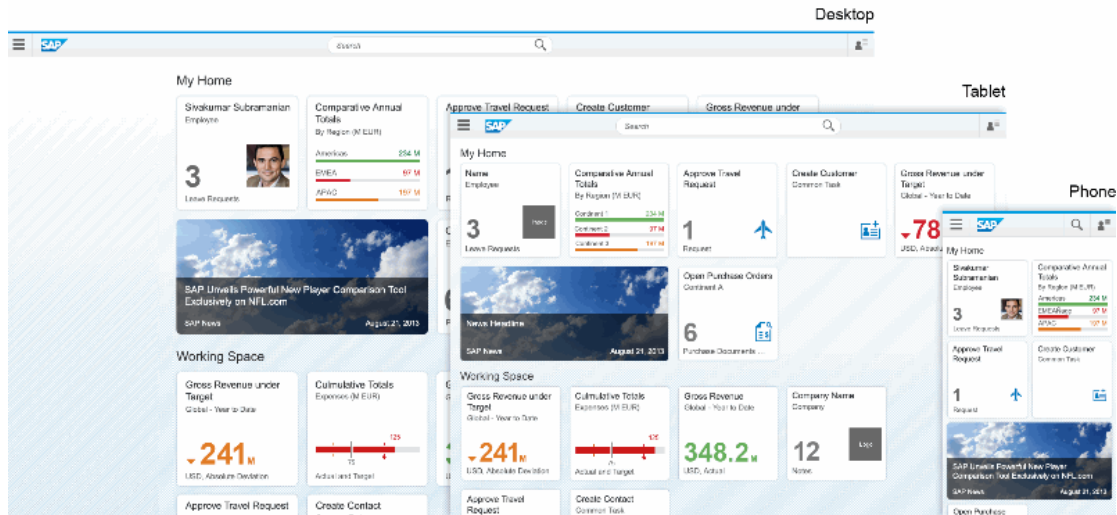
Figure 6: Homepage

In this regard, a Homepage is a specialization of a Launchpad. Within best practices for user experience, you should provide your users with home pages from which they can execute their regular work processes and services.

3.2.5.2   Tile display on Homepage

The Homepage consists of two areas: a tiles area displaying a selection of tiles, and a collapsible group panel, enabling navigation between different tile groupings. The panels can be collapsed via the respective 'panel' icon.

Tiles on the Homepage are arranged in groups. The user's initial grouping is called *My Home*. Users can create additional groups and manage them via the group panel. However, on phones, users cannot move or delete tiles or groups.

Users are able to group and rearrange, as well as to add or remove tiles from the Homepage. They are also able to create new groups, to delete, or to reset them.

Action	Comment
Add a tile	Clicking or tapping the + icon in any empty group, or clicking on *Open Catalog* in the group panel redirects users to the available apps. Users are able to select a tile category from the dropdown box and/or make use of the search field. Users can either click the tile that they want to add to a group, or drag this tile to a group of their choice in the group panel.
Move a tile	A tile can be rearranged by dragging it to a new location within the same group or to a different group.
Delete a tile	In order to delete a tile, users have to drag it to the lower-right corner where the icon of a bin appears.
Open a tile	Clicking or tapping a tile opens the underlying application or content item.
Add a group	Clicking or tapping the + icon in the group panel adds a new group. Users are able to enter a group name. If no group name is entered, a default group name gets selected automatically. The new group appears at the end of the panel.
Move a group	A group can be rearranged by dragging it to a new location within the group panel.

Action	Comment
Rename a group	Users are able to rename groups by double-clicking or double-tapping a group name in the group panel. A textbox opens in place and, after having typed the new name, it gets saved automatically. To discard the change while the text box is in focus, users must choose *Esc*.
Delete a group	Users are able to delete groups (including all containing tiles) that they have created. Pre-defined groups, for example, *My Home*, cannot be deleted. To delete a group, users have to drag a group to the lower left corner where the icon of a bin appears. Before deletion, a confirmation message is displayed asking for confirmation.
Reset a group	Users are able to reset a personalized pre-defined group to its initial state. In order to do so, they can drag a group that they want to reset to the lower-left corner where a bin appears. A confirmation message box is displayed. If users confirm, the group is reset to its initial state.

3.3 Application Types

The full screen and master/detail patterns can be used as templates to create a consistent layout for various different types of apps. Each template has a particular use case in focus.

For further information on app types, refer to *Application Types* in the *SAP Fiori Guidelines*.

The sections below provide information on selected app types.

These example app types are included in the configuration guide *SAP Fiori UI* (building block U56) and the corresponding business process documentation *SAP Fiori UI Business Process* (scope item U57).

3.3.1 Tracking App

A tracking app provides the user with a read-only overview of the status of a specific object, for example a purchase order, and the objects' contents within a process. If the list is required to display a large amount of data it is appropriate to use a full-screen pattern.

According to the application types outlined in the *SAP Fiori Guidelines*, a tracking app is a specalized example of a process app.

Figure 7: Tracking App Example

An example tracking app is provided with the configuration guide *SAP Fiori UI* (building block U56) and the corresponding business process documentation *SAP Fiori UI Business Process* (scope item U57).

Another, similar use case with a full-screen pattern would be for an app to guide a user through a process for an object. Further details on the process application type is available in the *SAP Fiori Guidelines* in the section *Process App*.

The screen layout of the main page consists of the following elements:

- Page Header
- Content
  The content has two parts:
  - o Tab bar
    The tab bar shows the individual process steps with the number of items with this status and the status of the step (using a semantic color).
  - o Content of the selected tab
    This is a list of the objects which have the selected status with details of the objects.
- Footer Toolbar

3.3.2   Monitoring App

Users may often require an overview of a complex topic at a glance. In this case, an app can provide a read-only overview over a topic using charts and diagrams. This app is often referred to as a monitoring app. This is a specialized example of a full screen pattern

An example monitoring app is provided with the configuration guide *SAP Fiori UI* (building block U56) and the corresponding business process documentation *SAP Fiori UI Business Process* (scope item U57).

During the design and development phase, you must bear performance considerations in mind. An app that provides several graphics for at-a-glance analysis, but which requires a very long time to process the data, will not be acceptable to users.

3.3.2.1.1   Home Page

The tile for the monitoring app consists of an icon and the app title.

3.3.2.1.2   Main Screen

The layout of the main page of the monitoring app consists of the following elements:

- Page Header
- Object Header

Displays key information of the data represented in the chart

- View Switches
    o Allows switching between one or several chart types and a tabular view on the data
    o Switching between two or more aspects of the data if applicable
- Chart Area

One chart or table showing currently selected view on the data
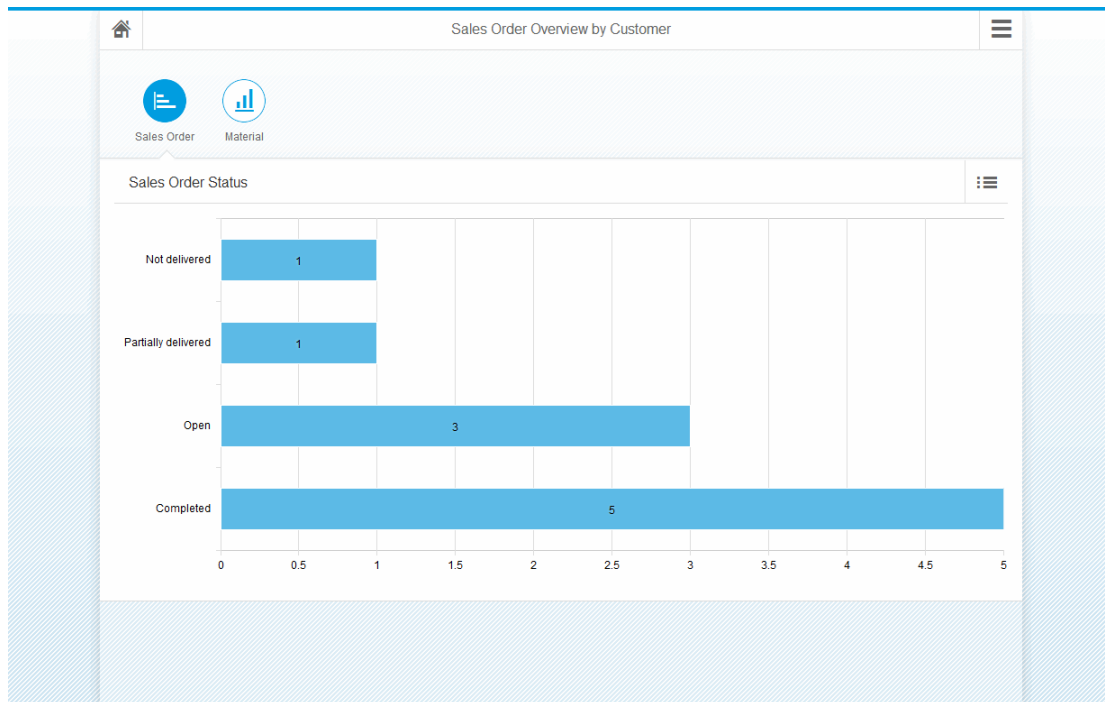
- Footer Toolbar

Figure 8: Monitoring App Example

A monitoring app covers only read-only scenarios; therefore the actions that a user can carry out on this screen are limited. The user can carry out the following actions:

- Change the filter on the date on which the chart is based upon. This is carried out in the Page Header
- Change the view on the data using the view switches
- Show more details on specific aspects of the data. This is achieved by navigating to the detail page

3.3.2.1.3   Detail

The detail page has the same general layout as the main screen. Therefore the detail page contains the same elements as the main page:

- Page Header
- Object Header
- View Switches
- Chart Area

The elements on the detail page are in the same order and with the same features as on the main page.

The detail page provides a detailed view of one aspect of the topic, rather than a general overview of the topic that is provided by the charts and tables that are included in the main screen. If necessary the chart can include a tree-like view to display and navigate through a hierarchical aggregation of the data.

Also on the detail page, provide a tabular text-only view and a view switch.

The user can carry out the following actions:

- Change the filter on the date on which the chart is based upon. This is carried out in the Page Header
- Change the view on the data using the view switches
- Navigate through the data hierarchy to show more detailed breakdown of the data.

3.3.3 Approval App

Approval is a specific workflow task type. An approval app allows the person who is responsible for approving the task to complete the task. The user can choose between the following options: approve, reject or forward.

The approval app uses the master/detail pattern: the master list displays all the items that are pending approval by the user. The details area holds the details of an item and the respective actions to complete approval. On the desktop and tablet landscape both the master and detail are visible; on the tablet portrait and phone they are automatically on one single page.

> An example approval app is provided with the configuration guide *SAP Fiori UI* (building block U56) and the corresponding scope item *SAP Fiori UI Business Process* (scope item U57).
>
> Further details on the approval application type is available in the *SAP Fiori Guidelines* in the section *Approval App*.

3.3.4 Self-Service App

Users execute self-service apps to perform routine tasks, for example as an employee entering leave requests or ordering office supplies.

Figure 9: Self-Service App Example

The user can carry out one of three different activities

- Create items, for example a leave request, a shopping cart, and so on;
- Check the status of specific items;
- General overview, for example of leave balance.

Depending on the use case, a self-service app may be more complex than other SAP Fiori app types. However, in a simple use case, the layout may be similar to a change app type. The self-service can provide a specialized example of a master/detail template as it can include a form, typically to request a new self-service object. The form is executed by editing

an existing object or by creating a new object, for example editing an existing travel request or creating a new travel request. The form is opened in the detail area of the master/detail page.

Depending on the use case, a self-service app may be more complex than other SAP Fiori app types. However, in a simple use case, the layout may be similar to a process app type.

An example self-service app is provided with the configuration guide *SAP Fiori UI* (building block U56) and the corresponding business process documentation *SAP Fiori UI Business Process* (scope item U57).

Further details on the form control is avaialble in the section *Form*.

3.3.5   News App

A news app is one app that does not adhere to the app framework paradigm. The news app uses neither the full screen nor the master/detail template. A news app provides a link to a feed from a defined data source.

An example news app is provided with the configuration guide *SAP Fiori UI* (building block U56) and the corresponding scope item *SAP Fiori UI Business Process* (scope item U57).

Further details on the approval application type is available in the *SAP Fiori Guidelines* in the section *News App*.

3.4 Controls

This section describes selected controls that are within the most commonly used app types. The descriptions include technical details on using the respective controls within screens according to SAP Fiori.

Full documentation on all controls that are available in the UI5 development toolkit for HTML5 is provided in the Controls *Gallery*. The gallery documentation is general SAPUI5 documentation and does not refer specifically to SAP Fiori concepts and layouts.

	The app-specific settings can either lead
	• directly to a settings dialog that contains a simple set of options;
	• or to a structured settings dialog, where each setting option is followed by a subset of further options.

3.4.1   Button

Buttons are represented using icons and are primarily, but not exclusively, located in the footer toolbar.

For a visual guidance on the usage of the Button control in SAP Fiori, refer to Button.

3.4.1.1   Icon Usage

3.4.1.1.1   Page Header Icon Buttons

The following icons can be included in the page header.

Icon Name	Icon	Description	Position
nav-back	←	Back	Master, detail, and full screen
multi-select sys-cancel		Toggle Icon: Select / Cancel	only in master header
up down	▲ ▼	Item Paging: Page Up / Page Down	only in line item detail page
filter	▼	Facet Filter	only in full screen

3.4.1.1.2   Generic Footer Toolbar Icon Buttons

No text buttons are included in the footer toolbar, with the following exceptions:

Icon Name	Icon	Description	Position
sort	↑↓	Sort	Master and full screen
filter	▼	Filter	Master and full screen
group-2	▤	Group	Master and full screen
add	+	Add/Create	Master only - always on the far right, never disappears in overflow menu (in full screen or detail as text only button)
action-settings	⚙	Settings menu contains: Settings (app-specific), About Fiori, Logout (easily enabled via the API-s of the footer bar and unified shell) and Login Details See corresponding chapter About-Setting-Logout.	Footer toolbar left aligned
action	↗	Actions/share menu: • Everything that you can do with an object outside of the app should be in this menu. See also Footer Toolbar. • Even if there is only	Detail and full screen - always on the very right, never disappears in overflow menu

		one action contained display the share/actions menu. • The following actions for the actions/share menu are optional. Only the *Save as tile* action is mandatory on all screens, except on Line Item Detail Page! It is easily enabled via the APIs of the footer bar and unified shell.	

As the control `sap.m.Bar` does not have an automatic overflow mechanism, the maximum length for button texts in the footer toolbar is 14 characters.

⚠

If this maximum length is exceeded, the button may overly following buttons and these may not be visible to the user.

This maximum length applies to the text in the original language and in translation of the original text.

3.4.1.2   Semantic Buttons

Use default buttons and positive/negative buttons to support the user in the tasks that he/she is most likely to perform.

Do not combine a default button and positive/negative buttons in one toolbar.

💡

This guideline has one exception: in edit mode on master/detail screen. For more information refer to the section footer toolbar.

3.4.1.2.1   Default Action

The default action should be defined for the action that a user will carry out in 80% of the use cases, for example *Edit, Create* or *Save*.

Only one default or recommended action is allowed per screen.

Actions within (modal) dialogs have no highlighting.

The only case when a default button can be combined with buttons in semantic (green and red) colors is in edit mode when the master list has approve/reject feature.

3.4.1.2.2   Positive/Negative Actions

Actions with a clear semantic meaning can be colored in red/green.

Red and green buttons cannot be combined with a default button.

💡

For the exception to this rule see Default Action.

Approve and *Reject* shall always be colored in green and red.

Apart from some exceptions all buttons should be placed in the footer toolbar of the app or the footer area of popups/dialogs.

💡

Exceptions to this rule are:

- upload attachment
- some chart controls
- facet filter

3.4.1.3　Button Visibility

Do not display actions or buttons to users who are not allowed to use these,

> You can use the concept of role-based actions, to ensure that users only see the screens, actions, and buttons that they are authorized to.

For actions or buttons which are temporarily inactive, use the enabled/disabled status.

3.4.2　Object Header

The SAP Fiori applications use a specific structure to display the key attributes of an object, when the details of the object are displayed, for example in the detail page of the master/detail pattern.

> For guidelines on the object header control, refer to *Object Header*.

3.4.3　Progress Indicator

A progress indicator is used to visualize the current progress towards completion of an operation.

> For guidelines on the progress indicator control, refer to *Progress Indicator*.

> ⚠️ The progress indicator is not used to visualize an ongoing process to load data. For this use case, you must use the patterns *Busy Dialog and Busy Indicator*.

3.4.4　Tile

Tiles on the launchpad are the entry points of apps. A launchpad screen has several tiles that are arranged centered on the screen.
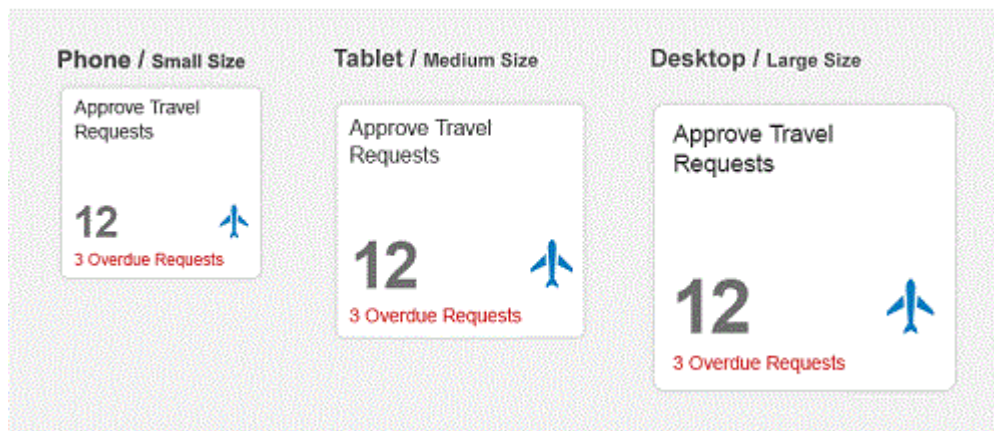

Figure 10: Tile

All tile types are available in all tile sizes. (In order to avoid repetition and save space, the following image shows only one tile example for the small and large size.) The following table shows the technical specification for tile sizes depending on the use case, as well as the source to be referenced for the tile control.

Control	`tile`
Library	`sap.m`
Size Desktop	224 x 212 px
Size Tablet	184 x 174 px
Size phone	148 x 142 px

The following list provides guidance on how tiles are included on a launchpad screen.

Design	
General Design	The number of visible tiles on one page is limited by the screen resolution and is therefore device dependent. If there are more tiles available than fit on one screen, an overflow mechanism is automatically provided. The tiles page offers an option allowing adding, rearranging, or deleting tiles from the Tiles page.
	A tile consists of an icon, a title, a key attribute (type dependent) and optional status summarizing app-specific information. The status may be displayed in a semantic color, for example negative, critical, positive or neutral.
General Recommendations	Use status only for information that can be influenced by the user in the app.
	Only display actions and problems and do not display positive messages.
	Only use gray icon for action tiles
	If a message with high priority has to be displayed in the status information area and the tile has already a status, then substitute the existing with the high priority message.
Tile Types	News Tiles
	KPI Tiles
	Fact Sheet Tile
	Plus Tile (top) • The plus tile on top is visible when no tiles are added to the ungrouped area.
	Plus Tile (ungrouped) • When tiles are added to the ungrouped area, the plus tile appears behind them.

3.5 Patterns

Patterns are re-usable user interface objects that are more complex than simple controls. They encapsulate recommended best practices for common use cases. If you use these

throughout the design of your apps, you can achieve a consistent user experience across all SAP Fiori-designed apps.

3.5.1   About/Settings/Logout

The *About/Settings/Logout* menu is accessed by clicking the *Settings* icon. The menu contains options

- to change the behavior or the layout of the app (for example the mode to re-arrange the tiles);
- for the trigger for the *About Fiori* box;
- for the *Log Out* option.

The *Settings* icon is in the top right corner of the shell bar. This opens an action sheet, which holds app specific settings, app and user information, as well as the log out function.
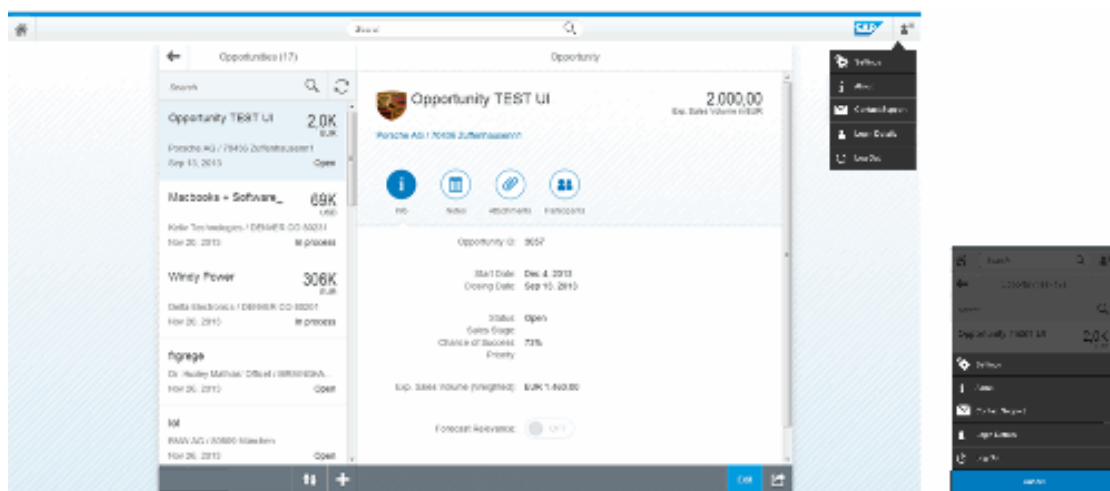

Figure 11: About/Settings/Logout

The menu contains following options, in the following order from top to bottom:

- *Settings*: application settings (not shown when the action sheet is called from the Homepage)
- *Contact Support:* opens a dialog
- *About*: provides details about the app (not shown when the action sheet is called from the Homepage)
- *Login Details*: provides details about the user that is logged in
- *Log Out*: opens confirmation dialog.
- *Cancel*: closes the menu

   The *Cancel* option is only shown on a phone.

The following guidelines apply to the following aspects of the menu:

Login Details Dialog	
	The login details dialog contains: - user name

	• server URL including port • language

About Dialog	
	The *About* dialog contains: • app icon • official app name (same as on the Tile page) • full technical app name • app version • build ID • SAPUI5 version • user agent details

Settings	
	The *Settings* dialog can be either: • a simple set of options; or • a structured settings dialog, for more complex use cases. In a structured settings dialog, the individual settings options can have different for each set of settings options. One set of options may be a set of checkboxes; another set of options may be a set of radio buttons; another set may be a slider option.

3.5.2   Charts

Business Charts help you to visualize analytical information to the user.

> For full guidelines on the chart pattern, refer to *Charts*.
>
> The `sap.ca` library contains the visual design defined for charts for SAP Fiori. Therefore application developers can comply with the visual signature in charts with little effort.

3.5.3   Dialog

The dialog box is a forced decision or a confirmation that needs to be signed-off by the user.

The dialog can hold two actions, a positive one (left) and a negative one (right).

The dialog looks identical on a tablet and on a phone. Buttons are not stacked on the smartphone.

> For further guidelines on the dialog pattern, refer to *Dialog*.

3.5.4   Form

A form is used to present data to the user and/or to allow users to enter data in a structured way. It contains other UI elements, like labels, input fields, checkboxes, sliders, and so on. Forms come in two flavors:

- read-only where the data is only presented as label-value pairs;
- editable where users can enter data.

> For further guidelines on the form pattern, refer to *Form*.

3.5.5   Loading

When data is loaded into a screen you must consider a number of issues to keep the user informed on the process. Two important aspects are:

- Indicating that the process is ongoing, in other words that it is busy
- Informing the user when no data is available for display, in other words that the page is empty.

3.5.5.1   Busy Dialog and Busy Indicator

Busy dialogs and busy indicators show that an operation is ongoing.

A busy dialog blocks the user interface until the currently running operation has been finished. No other activities can be started in parallel.

A busy indicator shows that some operation is ongoing. This does not block the user interface and other operations can be triggered in parallel.

> In contrast, a *Progress Indicator* is used to visualize the current progress towards completion of an operation.

When to Use What?	
If the user should not start any other activity, use a busy dialog. In any other case, use a busy indicator.	- Use the busy dialog with caution and only rarely. Blocking the user interface also provides the impression of using a slow application. - Do not use a busy dialog when loading data from the backend. Instead, use the concepts for an *Empty Page*.
If the ongoing operation covers a part of the screen with multiple controls, use a busy indicator.	- If you need to display additional information, use the busy indicator, as it does not cover the corresponding controls. In this case, leave empty space to allow for displaying the busy indicator. - If the user needs to be able to cancel the ongoing operation, use the busy indicator. This can be displayed with additional controls, for example a *Cancel* button or a *Cancel* icon.
If you do use a busy indicator, provide the indication at the position where the result is expected after the operation is complete.	

Additional Guidance	
Do not show Busy Indicators or Busy Dialogs for operations which last less than 2 seconds	
For other operations, show the Busy Indicator or Busy Dialog after 1.5 seconds.	
If displaying a Busy Dialog on direct initiation of the end user, use a short and precise text describing the operation. Start the text with a verb and end it with an ellipsis ("...").	• The text can be as short as one verb. *Loading...* *Refreshing...* *Sending...*
If the end user did not directly initiate the Busy Dialog, use at least one sentence to describe the operation. Start either the first or the last sentence with *Please wait*.	*The system is busy searching data. Please wait until data are loaded.*
Do not change the mouse cursor to indicate the ongoing operation.	
Do not show several busy indicators or busy states at the same time.	• Use a busy dialog only if the loading process corresponds to the whole screen.

Exceptional Cases	
Master/Detail	• When switching items usually take long time, show a busy dialog including the text *Loading...* centered horizontally and vertically on the details area. • Only cover the details area but not the master list - the end user should still be able to select another list item. • Show the busy dialog after 1.5 seconds.
Master List	• Whilst loading, a server roundtrip, for example for a filter, sort or search, usually takes a longtime. In this case display the text *Loading...*. For more information, refer to *Empty Page*.
Tables	• Whilst loading, a server roundtrip, for example for a filter, sort or search, usually takes a longtime. In this case display the text *Loading...* inside the table.

3.5.5.2   Empty Page

There are a several different use cases when an empty page might be shown. The empty page layout is the same for each use case as shown in the illustration below, but the text must be changed to reflect the exact use case.
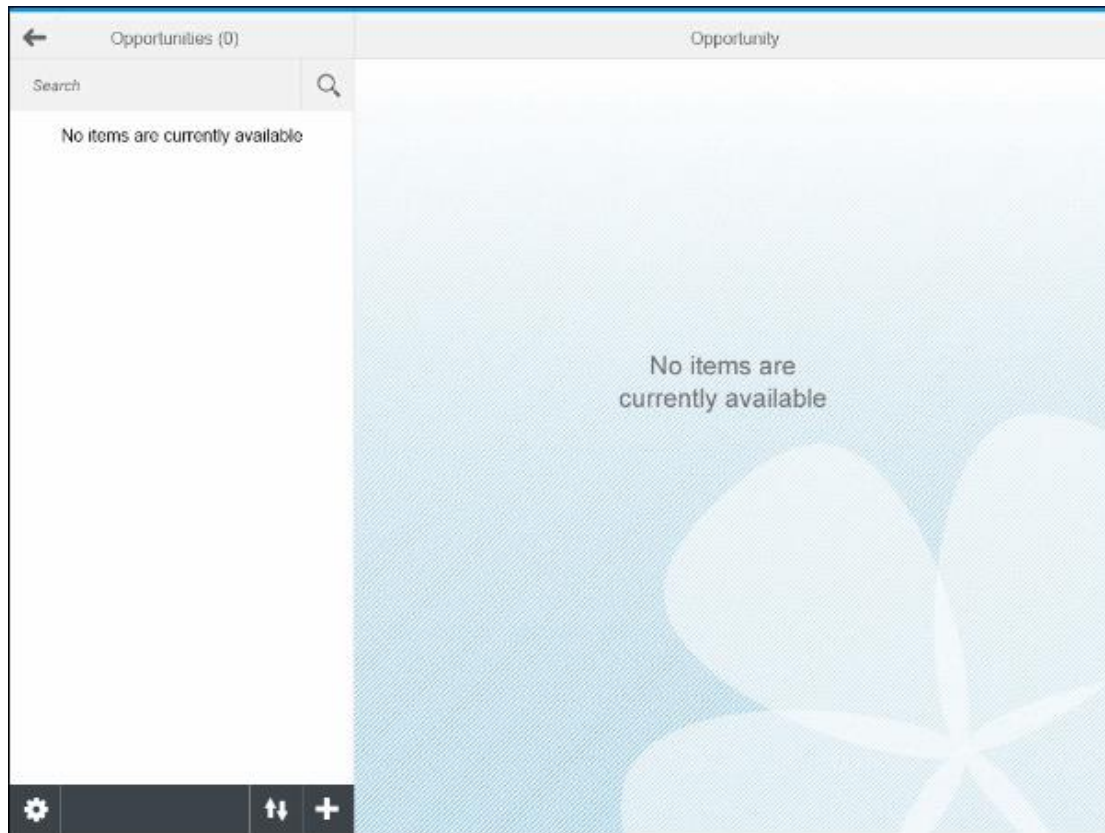

Figure 12: Empty Page

3.5.6   Messaging

Messages can be displayed by using one of two controls:

- Message Box

  A message box is used to display warnings or notify errors and require the user's sign-off, for example press *OK*.

- Message Toast

  A message toast is a small, non-disruptive dialog box for success messages that disappears automatically after a few seconds.

  For further guidelines on messaging patterns, refer to *Messaging*.

3.5.7   Search

Search functionality is provided where the amount of data is too large to find something just by scanning through, for example in the master list (see Master-Detail or when selecting a value from a large pre-defined set.

  For guidelines on the search pattern, refer to *Search*.

3.5.8   Tab Content

Tabs provide access to additional object facets to the user. In this case, as soon as there is more than one item inside, the sum of all items is provided on the tab. Tabs can also be used to filter lists and allow the user to view either the whole list or just the items with a specific attribute.

For guidelines on the tab content pattern, refer to *Tab Content*.

3.5.9   Table

The table holds a set of line items. Depending on the scenario the user may be able to view additional details.

For guidelines on the table pattern, refer to *Table*.

Further Design Best Practices	
Content Formatting	The table provides flexibility (incl. multi-line cells) by providing the possibility to put controls into cells. Specific controls are available to support the two most common content formats:
	For the key identifier of a list item, use `sap.m.ObjectIdentifier`. Show the key identifier in the first cell.
	For numbers with units, show the correct formatting by using `sap.m.ObjectNumber`.
	For strings with IDs, show the ID in brackets after the corresponding string
	Avoid truncation. Use controls, which wrap the text.
Line Items	Per Line Item, set `sap.m.ListType` in the following way: • To allow navigation to a line item details page, set `sap.m.ListType` to *Navigation*. This will result in an according indicator at the end of the line >. • If no navigation is possible, set `sap.m.ListType` to *Inactive*.
Selection Types	None: items cannot be selected. ⚠ Line items can nevertheless use the `sap.m.ListType` *Navigation,* which allows click handling on specific line items. This should only be used when the click triggers navigation to a corresponding line item details page.
	Single Select: one item of the table can be selected.
	MultiSelect: allows selection of one or more items. For this, the table provides checkboxes on the left side of each line item. *Select All* works via a checkbox on the left of the column header. ⚠ In multi-selection tables, avoid having check boxes

	in your first column.
Personalization	The following features are supported: • add/remove columns • re-arrange columns
	Both features are available in a personalization dialog.
	How to open the dialog? The trigger for the personalization dialog is placed on the table title row.
Merging duplicates within a column	To simulate the behavior of row spanning, you can merge cells of consecutive rows inside a column automatically, if they have the same value.

4 Appendix

4.1 SAP Fiori Typography

SAP Fiori design uses a defined set of fonts with set font sizes for various headline and general text styles. This achieves a consistent format for all SAP Fiori design apps.

> Further information on type fonts is available in the *SAP Fiori Guidleines* in the section *Typography*.

4.2 SAP Fiori Colors

SAP Fiori design has a designated palette of colors that provide a clear and consistent look-and-feel for all apps. The specifications for the colors that are to be used for user interface elements are determined according to particular use cases.

> When referring to colors you should always refer to the variable quoted in the column **Variable (Less)**. This value will be constant, whereas the color, or the tone of color, and hence **Value Hex** and **Value (RGB)**, may be changed over time.
>
> Further information on using colors is available in the *SAP Fiori Guidleines* in the section *Colors*.

4.3 SAP Fiori Icons

4.3.1   General Fiori Icon Sizes

Use –case	Example	Size
Within some controls	Checkbox Notification icons in Table	16px
In App icons	Buttons	22px
Tiles on phone		24px
Tiles on Tablet/Desktop		32px

Use –case	Example	Size
Thumb icons Pictograms	Attachment list	40px

4.3.2   SAPUI5 Icons

SAPUI5 comes with a built-in set of icons that you can use in your applications. For documentation on these icons refer to the Icon Explorer in the *UI5 development toolkit for HTML5.*