


# Referencia para el layout de las ventanas del “Iskit-vaadin-components”

(versión 1.0.1)

## Objective

The Iskit-vaadin-components starter for building responsive business web apps. It contains a collection of components, view templates, theming utilities and best practises that can help teams save time of development time at the start of a new project. It makes use of Vaadin’s free and open source framework and components.



[Home](#)  
[Operación](#) ▲  
[Operación Diaria](#)  
[Dummy page](#)

Ejemplo de una tabla con preferencias

ID	Instrumento	Tipo	Títulos	Promedio	\$ Mercado	Monto	Plusvalia	Ultima operación	Validado
						ALL	ALL	ALL	
0	CEMEX-L* 0	RV	2908	\$1,500.00	\$1,520.00	\$4,362,000....	\$58,160.00	27.09.2012	✕
1	CETE-2020...	RF	5281	\$300.00	\$320.00	\$1,584,300....	\$105,620.00	30.03.2006	✓
2	BONDE-20...	RF	3576	\$300.00	\$320.00	\$1,072,800....	\$71,520.00	29.11.2010	✓
3	PAGARE-20...	RF	7187	\$300.00	\$320.00	\$2,156,100....	\$143,740.00	09.01.2001	✓
4	PAGAFE-20...	RF	9492	\$300.00	\$320.00	\$2,847,600....	\$189,840.00	18.09.1994	✓
5	TELMEX-C*	RV	2706	\$1,500.00	\$1,520.00	\$4,059,000....	\$54,120.00	17.04.2013	✓
6	CETE-2020...	RF	3296	\$300.00	\$320.00	\$988,800.00	\$65,920.00	05.09.2011	✕
7	BONDE-20...	RF	2947	\$300.00	\$320.00	\$884,100.00	\$58,940.00	19.08.2012	✓
8	PAGARE-20...	RF	224	\$300.00	\$320.00	\$67,200.00	\$4,480.00	02.02.2020	✓
9	PAGAFE-20...	RF	6295	\$300.00	\$320.00	\$1,888,500....	\$125,900.00	20.06.2003	✓
10	TELEVISA...	RV	9363	\$1,500.00	\$1,520.00	\$14,044,50...	\$187,260.00	25.01.1995	✓
11	CETE-2020...	RF	1949	\$300.00	\$320.00	\$584,700.00	\$38,980.00	14.05.2015	✓
12	BONDE-20...	RF	9993	\$300.00	\$320.00	\$2,997,900....	\$199,860.00	05.05.1993	✕
13	PAGARE-20...	RF	4480	\$300.00	\$320.00	\$1,344,000....	\$89,600.00	08.06.2008	✓
14	PAGAFE-20...	RF	6657	\$300.00	\$320.00	\$1,997,100....	\$133,140.00	23.06.2002	✓
15	MAXCOM...	RV	9132	\$1,500.00	\$1,520.00	\$13,698,00...	\$182,640.00	13.09.1995	✓
16	CETE-2020...	RF	6053	\$300.00	\$320.00	\$1,815,900....	\$121,060.00	17.02.2004	✓
17	BONDE-20...	RF	6450	\$300.00	\$320.00	\$1,935,000....	\$129,000.00	16.01.2003	✓
18	PAGARE-20...	RF	2114	\$300.00	\$320.00	\$634,200.00	\$42,280.00	30.11.2014	✕
19	PAGAFE-20...	RF	1451	\$300.00	\$320.00	\$435,300.00	\$29,020.00	23.09.2016	✓
default									

---

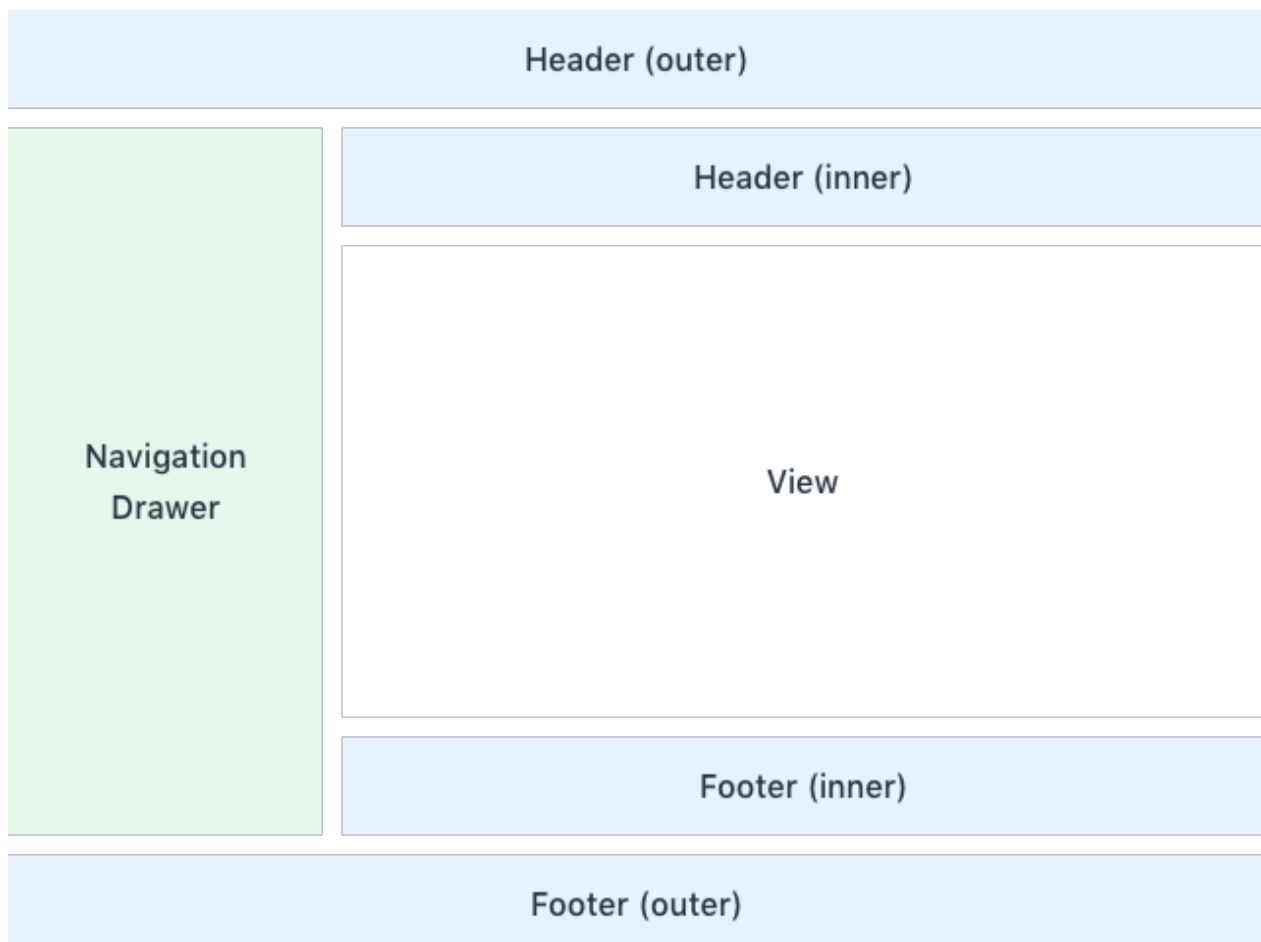
## Features

- ➡ Application header with avatar & tab support
- ➡ Customisable styles
- ➡ Installable on desktop & mobile
- ➡ Responsive navigation sidebar
- ➡ Templates for CRUD, dashboard & drilldown views
- ➡ 100% Java-based templates and Kotlin based Templates

---

## Structure

The Business App Starter's structure is comprised of several slots designed to help implement any layout



## Header & Footer Slots

These slots are intended for application-wide navigation, actions and branding. When using both the outer and inner elements, ensure the content hierarchy is clear and that they don't take up too much vertical space.

## Navigation Slot

The navigation slot is reserved for the `NaviDrawer`: a responsive and collapsible navigation sidebar that houses searchable and hierarchical navigation items. It consists of a header intended for branding & account switching, a list of navigation items, and a footer which by default houses the collapse button.

## Navigation Modes

The application supports two different navigation modes:

- **Traditional:** clicking a link changes the current view.
- **Tabbed:** a browser-esque experience where users can open and close tabs. Clicking a link changes the content of the current tab.

## View Slot

The intended slot for views. Place views in this slot by setting the `@Route` annotation's `layout` parameter to `MainLayout.class`.

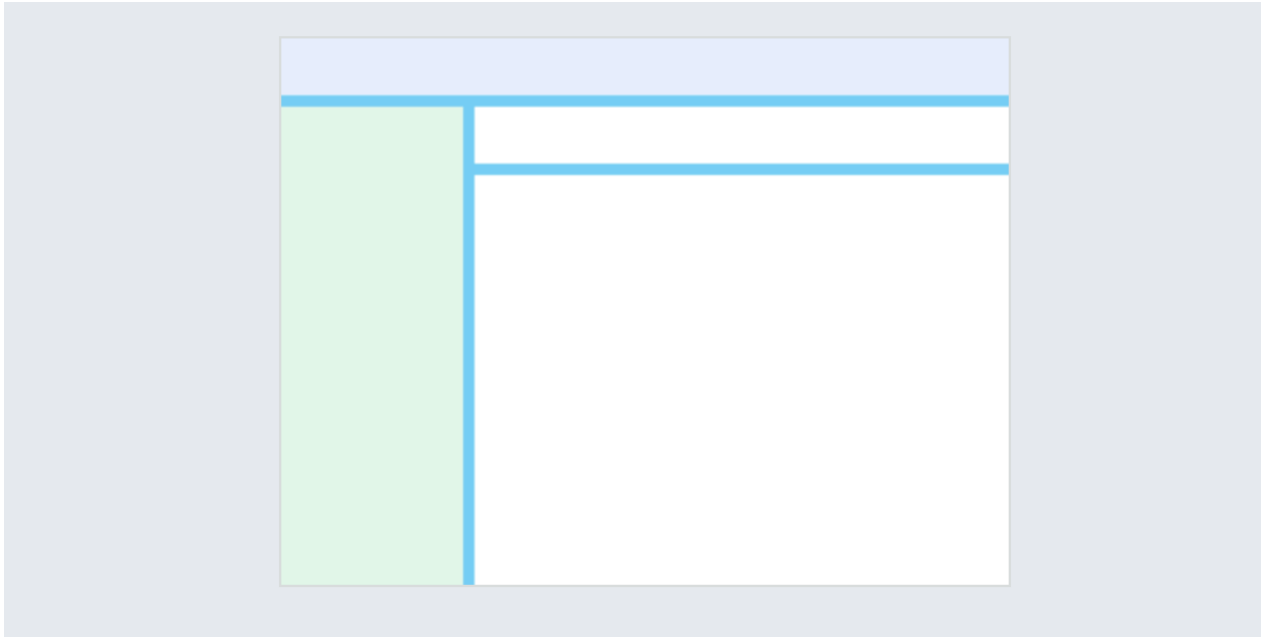
```
@Route(value = "my-view", layout = MainLayout.class)
@PageTitle("My View")
public class MyView extends ViewFrame {
```

## ViewFrame & SplitViewFrame

All of the Business App Starter's views extend either `ViewFrame` or `SplitViewFrame`. They simplify the process of creating views that contain a header, (scrollable) content and footer. `SplitViewFrame` houses an additional details slot for creating master-detail views.

## Layout Examples

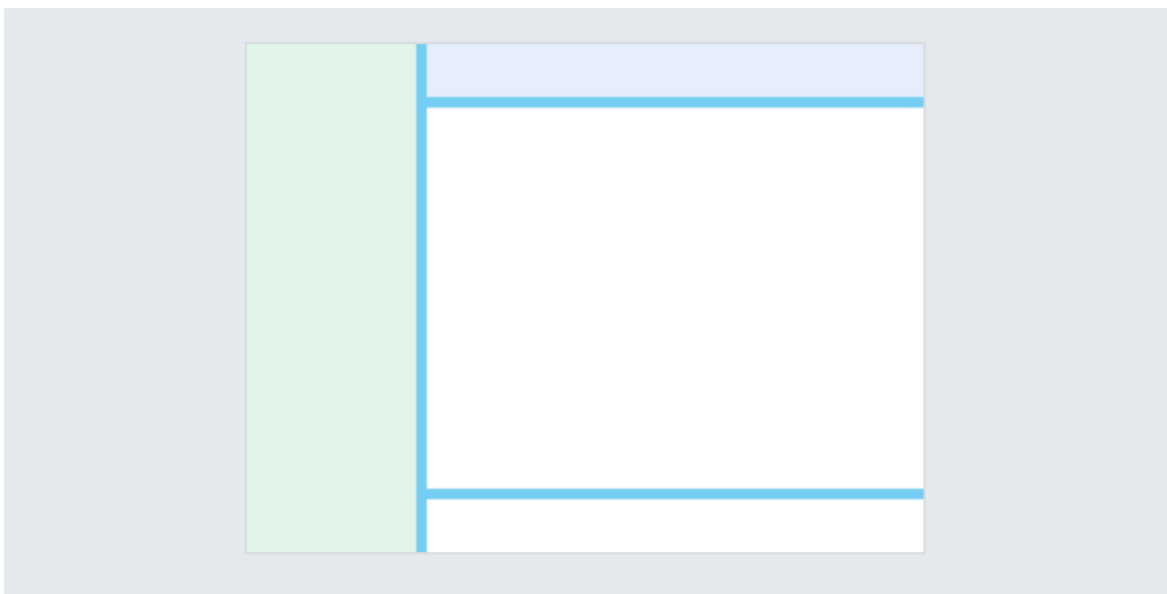
Here are a few examples of common layouts used in business applications.



An often-used composition where the outer header is the most prominent element. The view itself sports a header for showing the title.

Used slots:

- Header (outer)
- Navigation drawer
- View with header

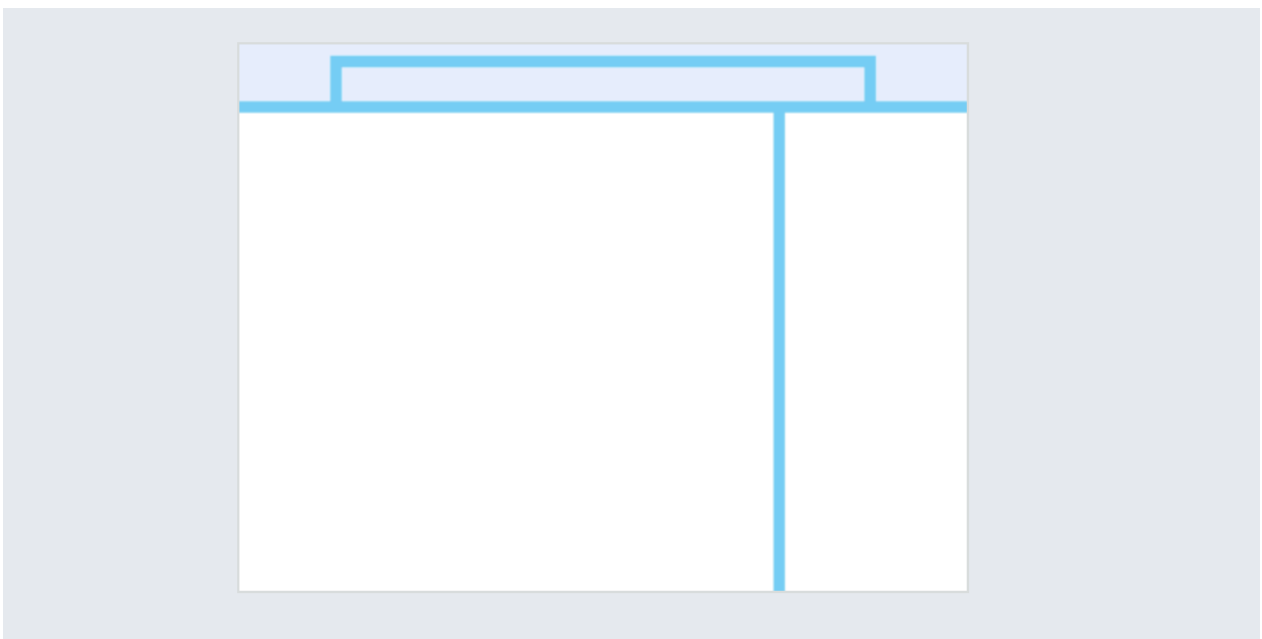


This layout puts more focus on the navigation drawer. Suitable for complex hierarchical navigation structures. The view footer houses view-specific actions.

Used slots:

- Navigation drawer
- Header (inner)
- View with footer

The navigation is placed in the outer header. The view area is maximised but the navigation has to be simple. The view's sidebar is used to display additional details.



Used slots:

- Header (outer)
- View with sidebar

---

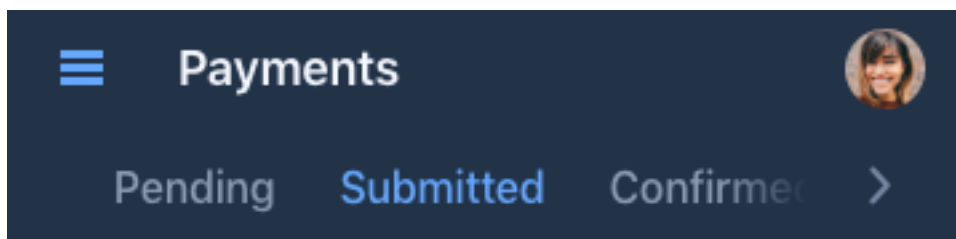
## Custom Components

The library comes with the following custom components:

- AppBar & TabBar (inside the application and not in Iskit-vaadin-components)
- Badge
- DetailsDrawer
- FlexBoxLayout
- ListItem
- NaviDrawer
- NaviTabs (closable)

### AppBar & TabBar

[AppBar](#) consists of a main menu icon, contextual navigation icon, title, tabs, action items and avatar. All building blocks are optional.



*AppBar with a menu icon, title, tabs and avatar*

[TabBar](#) is a simpler version of [AppBar](#) that discards the contextual navigation icon and title.

### Badge

Server-side component for [Material badges](#). TODO: check is implemented

### DetailsDrawer

Mainly intended for usage in master-detail views. Composed of a header, (scrollable) content and footer.

ID	Status	Customer / Order ID	Due Date	Amount (\$)
5253	Overdue	Brewer Holding Century Training 2446	Apr 08, 2019	320.35
6083	Overdue	Office Tele-Dynamics 1732	Apr 08, 2019	655.21
5784	Outstanding	Sharp Network Sciences 2836	Apr 09, 2019	706.41
9569	Overdue	Frank Tava 6751	Apr 10, 2019	1,039.25
473	Open	Office Tele-Dynamics 9063	Apr 10, 2019	954.59
6037	Open	Lafayette Printing 4498	Apr 11, 2019	200.11
9303	Open	Medical Telecommunications 2032	Apr 12, 2019	796.04
594	Outstanding	Sharpe Genetics 2833	Apr 12, 2019	654.54
7179	Overdue	Surous Plastics 2349	Apr 12, 2019	881.81

### Sharp Network Sciences

**Details** | Attachments | Activity

Invoice ID: 5784

Invoice Date: Apr 05, 2019

Order ID: 2836

Amount: \$706.41

Status: Outstanding | Open | Paid | Overdue

Due Date: 4/9/2019

**Save** **Cancel**

Master-detail view with a DetailDrawer shown on the far right

## FlexBoxLayout

**FlexBoxLayout** is a server-side implementation of [CSS Flexible Box Layout](#). It extends **FlexLayout** and contains convenience methods for setting **flex**, **flex-basis**, **flex-direction**, **flex-shrink** and **flex-wrap**. It also supports setting **background-color**, **margin**, **overflow**, **padding**, **shadow**, **spacing** and **theme**.

## ListItem

A list item comprised of a prefix, text (primary & secondary) and suffix. Only the primary text is mandatory. Any component can be placed in the prefix and suffix slots.



ListItem with a prefix icon, primary and secondary texts, as well as a suffix button

## NaviTabs

Tabs that when clicked navigate to a specific target (view) and that can be closed.



## Divider

Simple component to leave a height space between two components.

## Initials

This component list in a nicely manner some initial short name.

## SearchBar

Component (example of a Polymeter) in order to select som search criteria and can be attached to a Grid for example.

## SearchBarCombo

Same as the searchBar but instead of a check box an the criteria a comboBox is shown.

## Token

See demo, shows a title and a cross in front.

## OfflineBanner

Banner where it is shown when the app is off-line.

## TwinColSelect

TwinColselect is a component extracted from Vaadin Directory (an avoid to have too many libraries so the source code is included in the library with minor modifications, e.g.,material theme support) that to have two lists and select from<->to these list to select/unselect



different items. This is to assign or un-assign multiple options.

## ValidTextField

ValidTextField is a sub-class of TextField but it can handle java validation. Java validation normally is used using a DTO inside a FormLayout, but sometimes the programmers need validation for a single empty TextField without the need to declare a DTO object. This is when ValidTextField is needed.

Some example how to use it:

```
ValidTextField valid = new ValidTextField();

valid.addValidator(new StringLengthValidator("Wrong length", 1, 10));
valid.addValidator( s -> s.equals("LegoSoft"), "Not LegoSoft!");
```

---

## Other components

Other components are not described in this document because they belong to a complete internal use like:

- Preferences
  - PreferenceGridMenu
  - PreferenceInMainMenu
  - Preferences