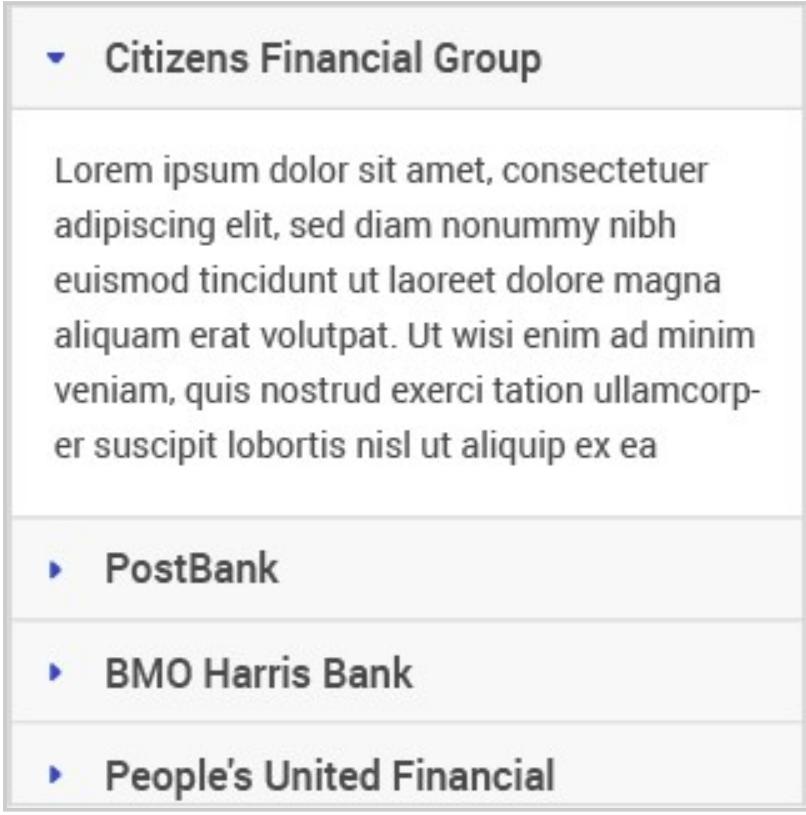




**Telerik UI for Xamarin** offers high quality Xamarin Forms UI components and Visual Studio item templates to enable every developer, regardless of their experience, to build professional-looking modern mobile applications for iOS, Android and UWP.

# Accordion

RadAccordion for Xamarin is a collapsible content control that helps you save screen space and at the same time present the content in an easily accessible way.



▼ **Citizens Financial Group**

Lore ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea

▶ **PostBank**

▶ **BMO Harris Bank**

▶ **People's United Financial**

Telerik Accordion for Xamarin is a vertically collapsible content panel that displays only one of its items at a time within the available space. RadAccordion helps you save screen space and at the same time present the content to the end user in an easily accessible way.

[Online documentation](#)

[Navigate to examples](#)

# Accordion

## First Look

- ▼ **Citizens Financial Group**

Lore ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea
- ▶ **PostBank**
- ▶ **BMO Harris Bank**
- ▶ **People's United Financial**

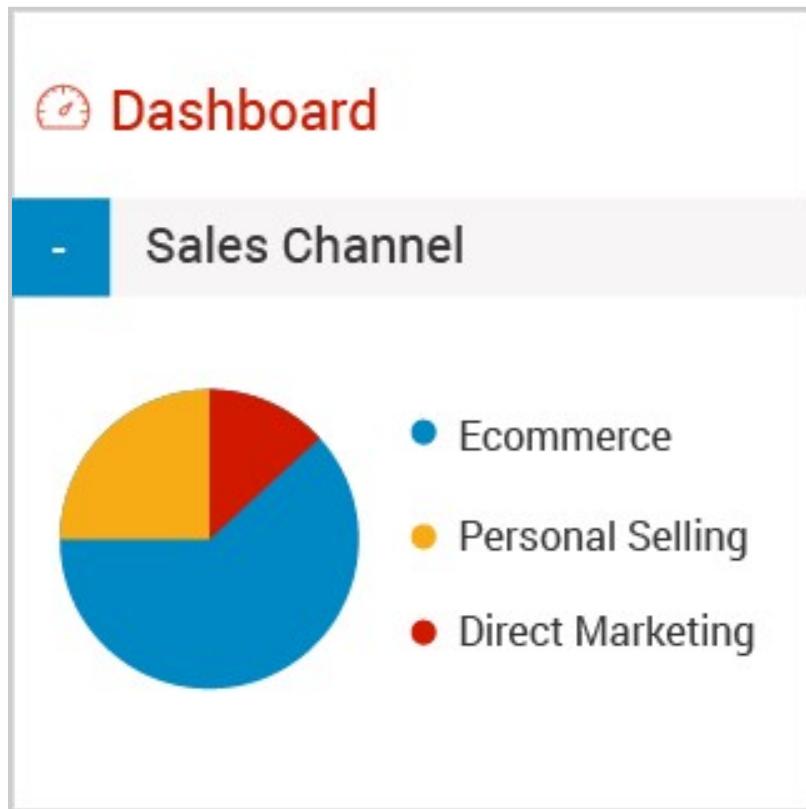
This example demonstrates basic setup of RadAccordion for presenting content in collapsible sections. The control is designed in such a way that opening one section automatically closes the previously displayed content.

[See the code](#)

[Navigate to example](#)

# Accordion

## Customization



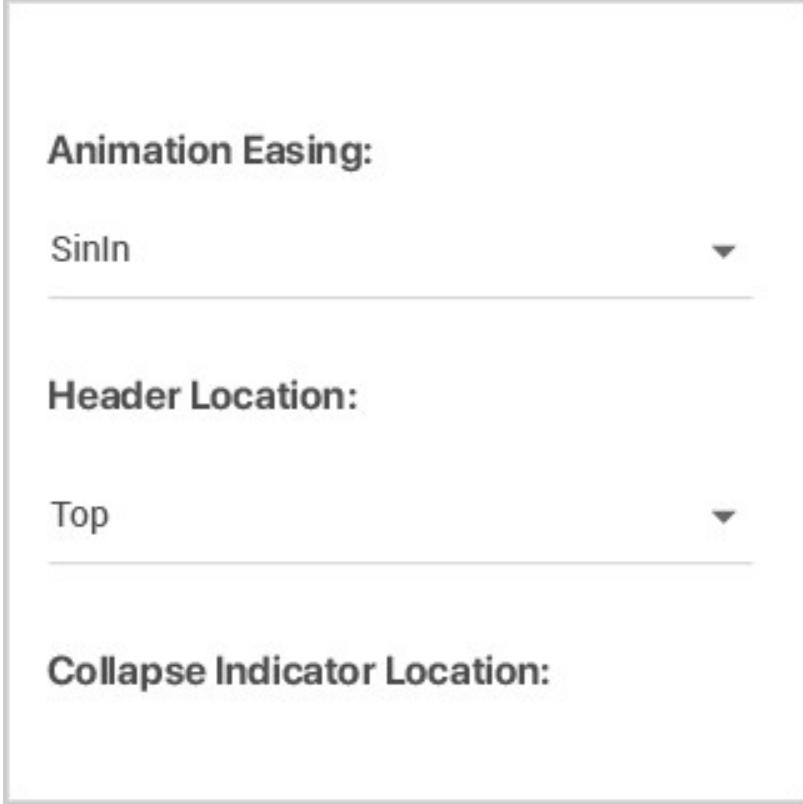
This example demonstrates how you could customize the visual appearance of RadAccordion by predefining the Accordion item header template.

[See the code](#)

[Navigate to example](#)

# Accordion

## Configuration



The screenshot shows a configuration panel for the Accordion control. It contains three dropdown menus:

- Animation Easing:** Set to "SinIn".
- Header Location:** Set to "Top".
- Collapse Indicator Location:** This menu is partially visible at the bottom.

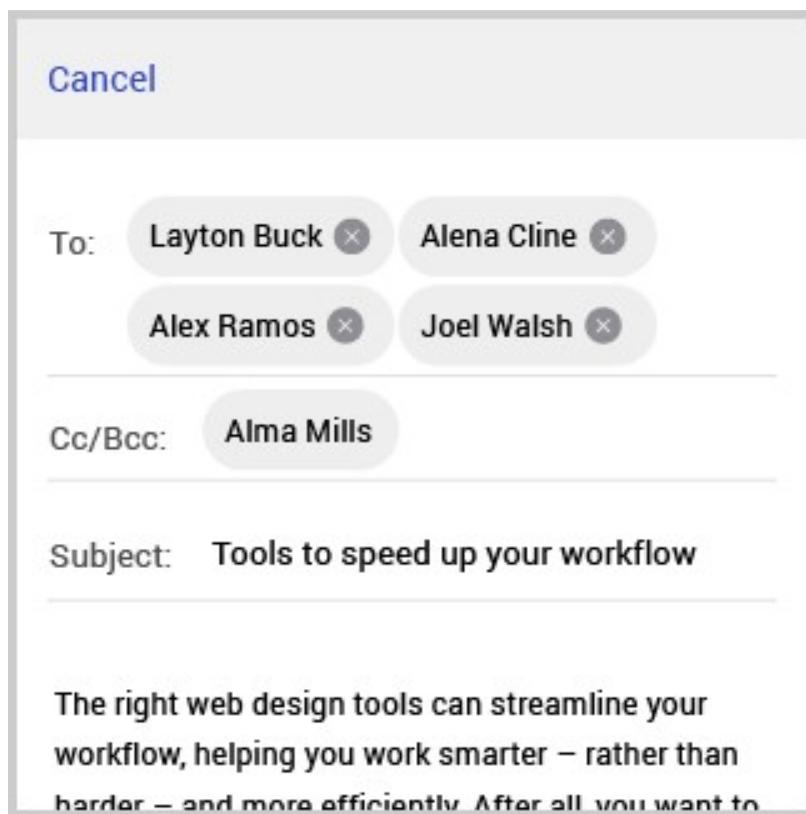
This example presents various configuration options you could use to make the Accordion control look and animate per your choice.

[See the code](#)

[Navigate to example](#)

# AutoCompleteView

Rad.AutoCompleteView control can automatically complete user input string by comparing the text being entered to all strings in the associated data source.



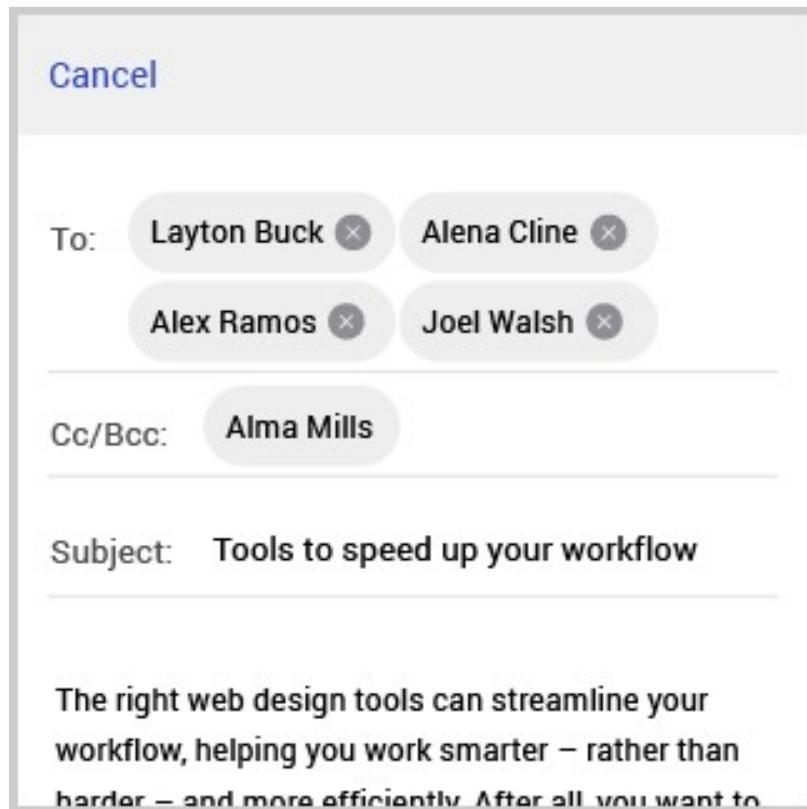
Telerik UI for Xamarin AutoCompleteView can automatically complete user input string by comparing the text being entered to all strings in the associated data source. The control has a number of advanced features such as different filtering options, tokens support and remote search, as well as full customization capabilities.

[Online documentation](#)

[Navigate to examples](#)

# AutoCompleteView

## First Look



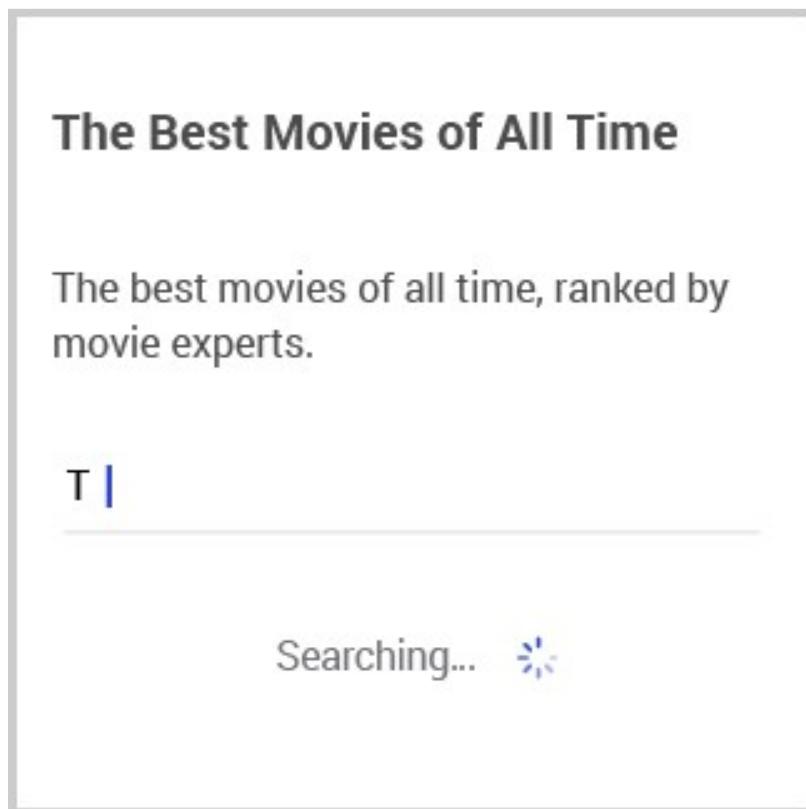
An example demonstrating how the AutoCompleteView control works with Tokens.

[See the code](#)

[Navigate to example](#)

# AutoCompleteView

## Remote Search



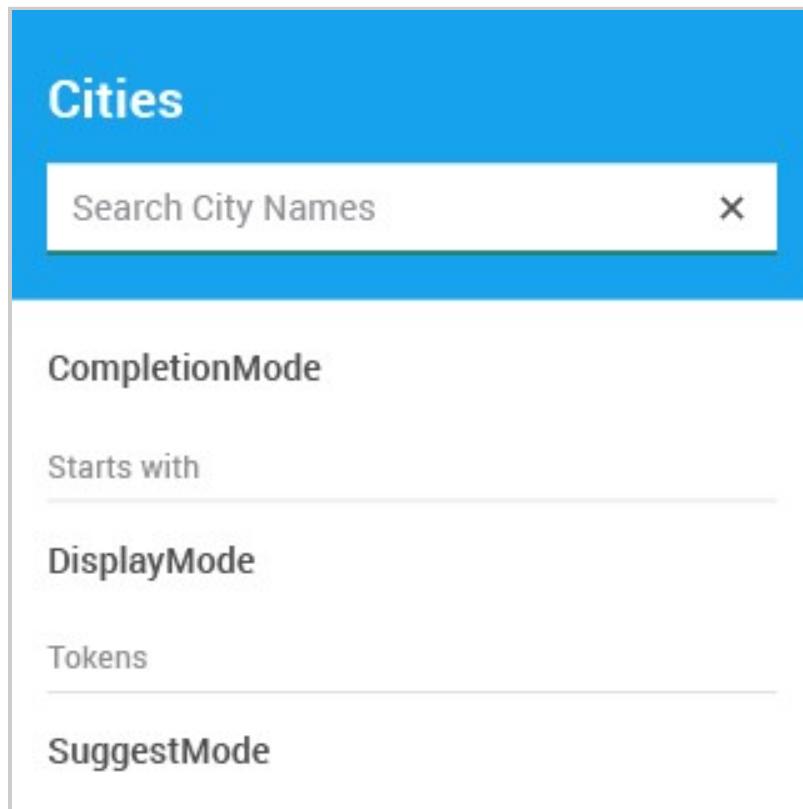
Telerik UI for Xamarin Rad.AutoCompleteView can set its ItemsSource at a later stage after the end user starts typing in the search field and text is changed.

[See the code](#)

[Navigate to example](#)

# AutoCompleteView

## Configuration



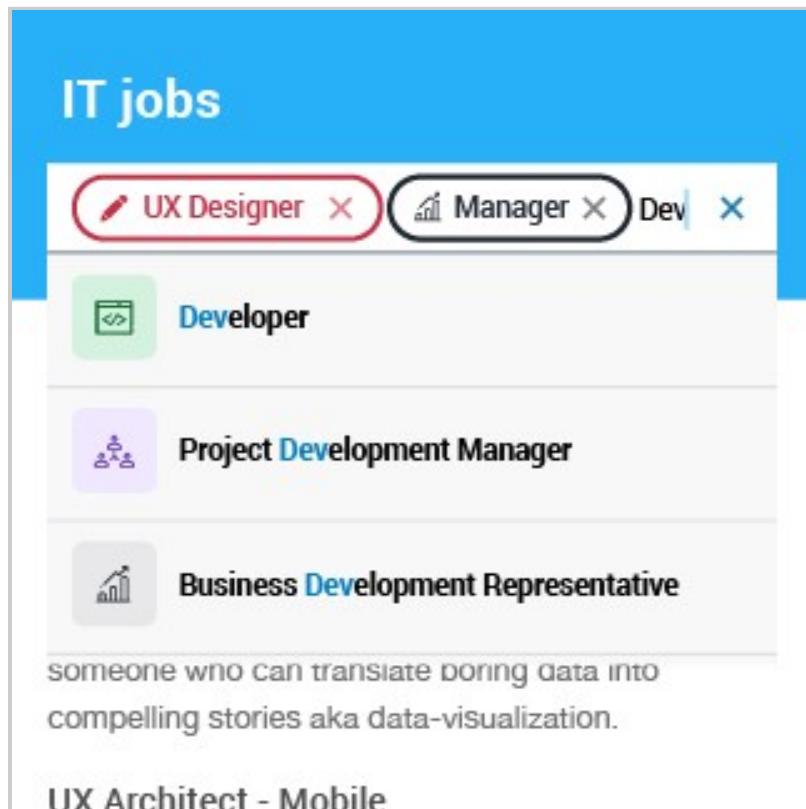
This example presents various configuration options that the AutoCompleteView control provides.

[See the code](#)

[Navigate to example](#)

# AutoCompleteView

## Customization



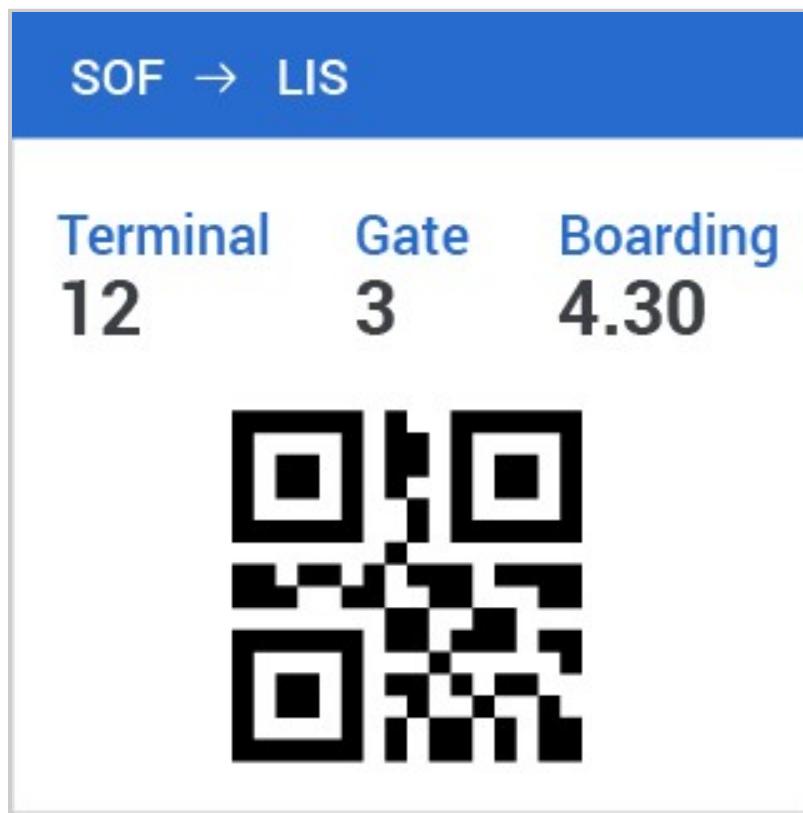
This example presents how the AutoCompleteView control templates could be customized.

[See the code](#)

[Navigate to example](#)

## Barcode

RadBarcode is a UI component used for creating and showing 1D and 2D barcodes.



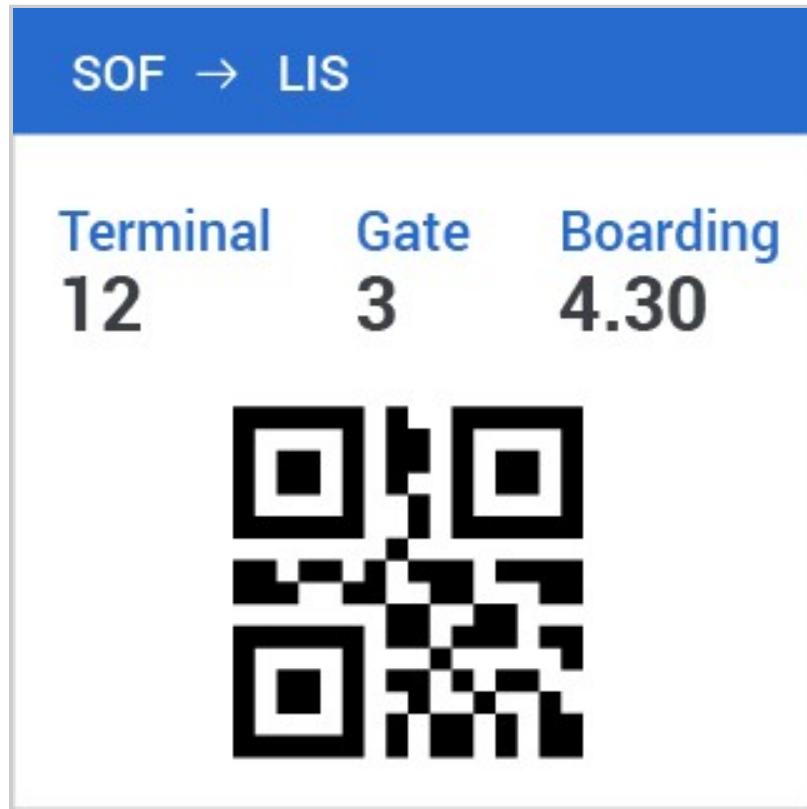
Telerik RadBarcode for Xamarin is a control used for creating and showing barcodes. You can generate and visualize barcodes in a machine-readable format via the RadBarcode control by providing numeric or character data.

[Online documentation](#)

[Navigate to examples](#)

# Barcode

## Boarding Pass



Telerik UI for Xamarin Barcode can generate a variety of 1D and 2D barcodes. This example demonstrates how RadBarcode can be used to display a 2D QR code inside a sample boarding pass page of an application. The QR code system has become popular due to its fast readability and great storage capacity. QR codes are often used in

[See the code](#)

[Navigate to example](#)

# Barcode

## Special Offer

**SPECIAL OFFER**  
**20% OFF**

for any salad until May 21st



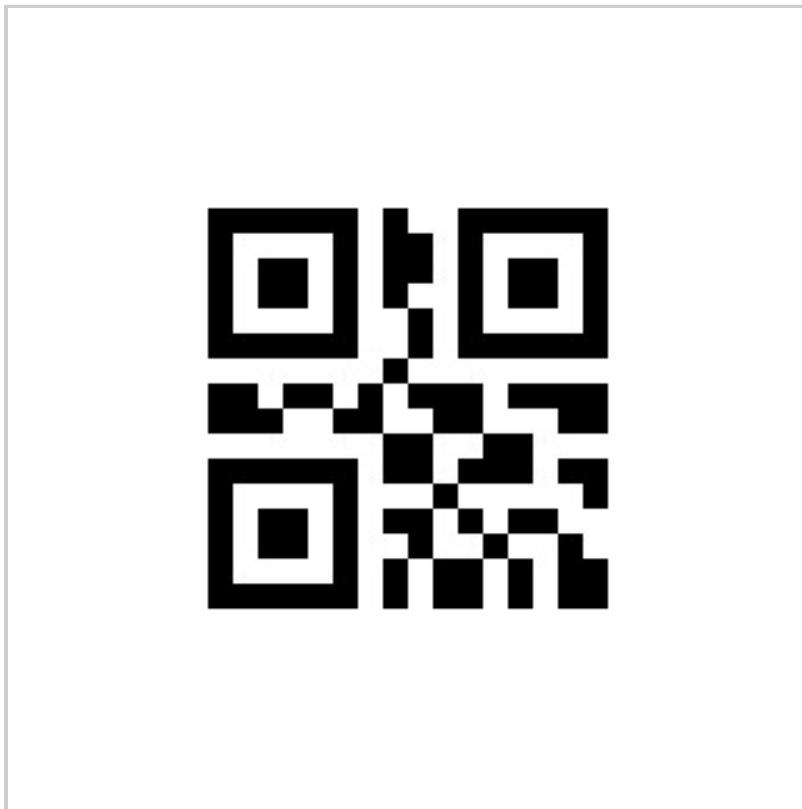
In this example RadBarcode for Xamarin is used to generate a linear (1D) barcode in order to identify a discount coupon. With RadBarcode you can choose between several types of 1D and 2D barcode symbologies and have also control over the visual appearance and sizing of your barcodes.

[See the code](#)

[Navigate to example](#)

# Barcode

**QR Code**



With RadBarcode you could generate the popular QRCode. This example demonstrates the available settings of the QRCode symbology.

[See the code](#)

[Navigate to example](#)

# Barcode

PDF417



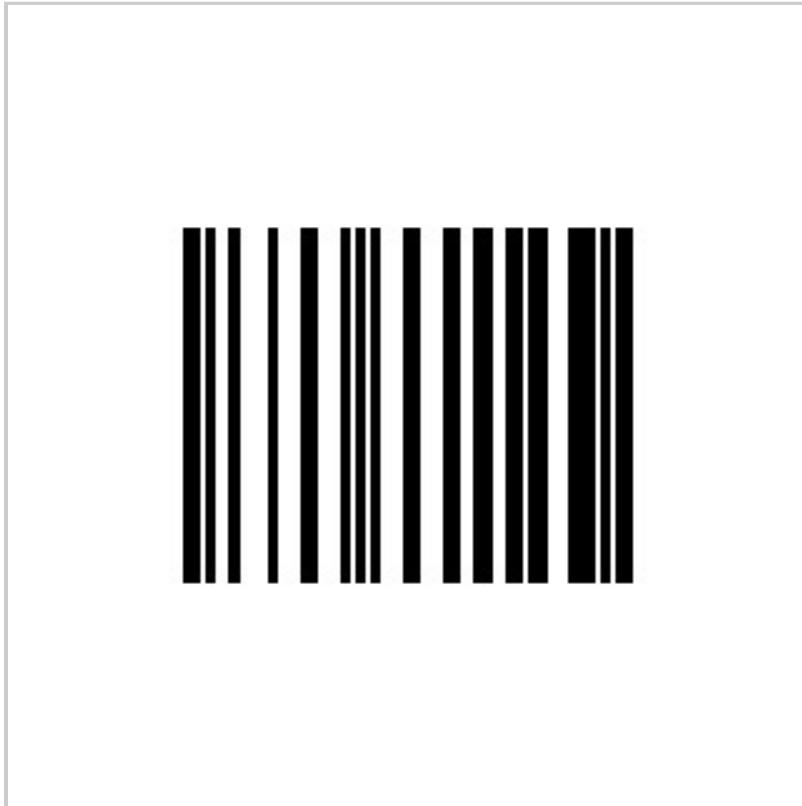
An example demonstrating the available settings of the PDF417 code symbology.

[See the code](#)

[Navigate to example](#)

# Barcode

## 1D Barcode Gallery



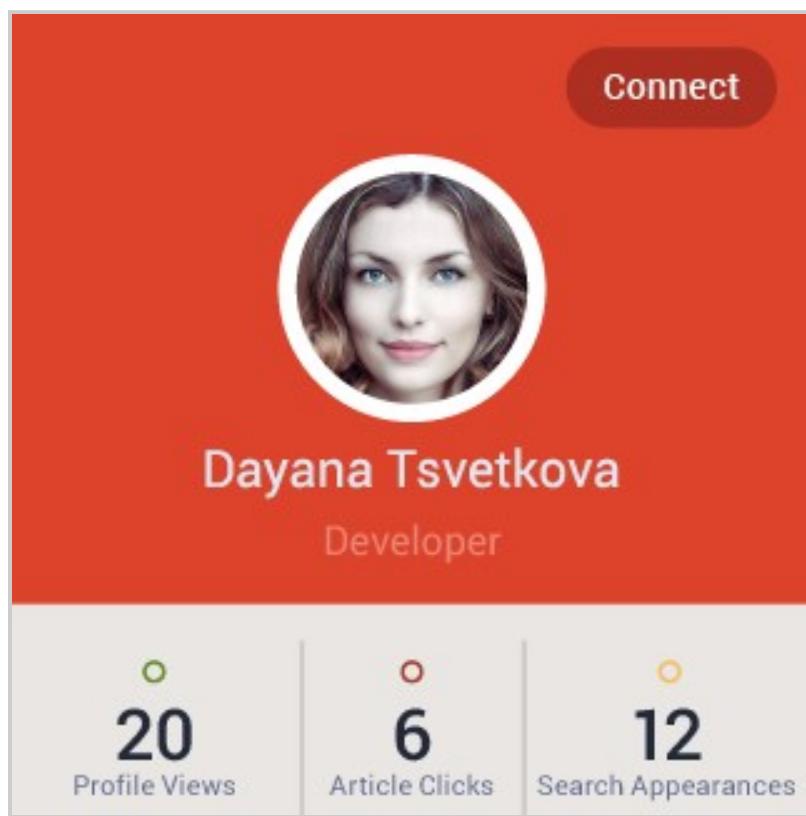
This example demonstrates all the linear (1D) barcodes the RadBarcode could generate including the most popular ones such as EAN13, EAN8, UPC-A, UPC-E and Code39.

[See the code](#)

[Navigate to example](#)

## Border

With Telerik RadBorder for Xamarin you will have full control over the look & feel of the border that wraps around your Xamarin views.



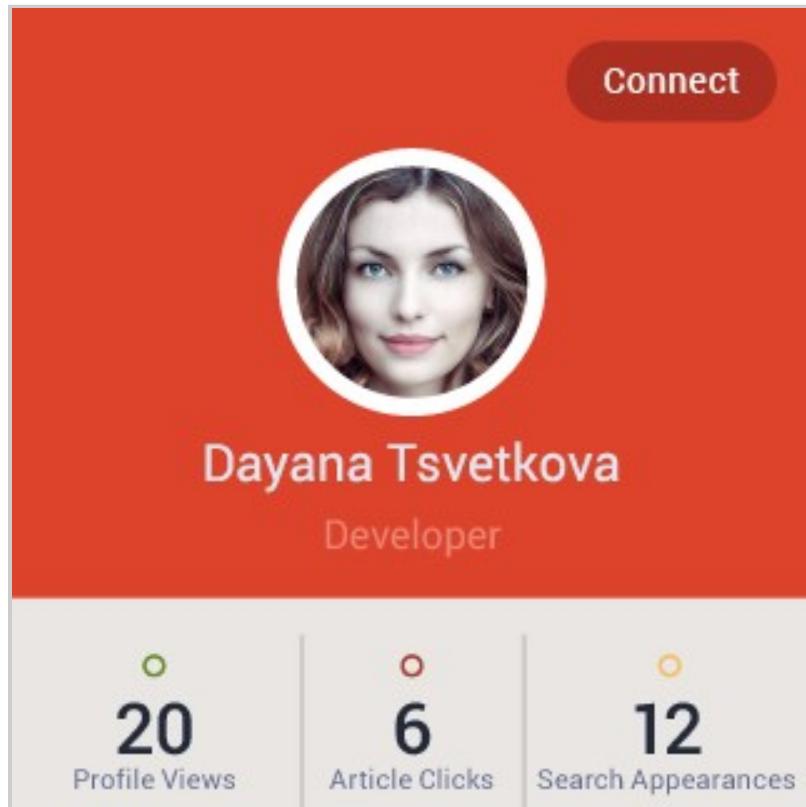
With Telerik RadBorder for Xamarin you will have full control over the look & feel of the border that wraps around your Xamarin views. You could surround any view such as Label, Image, so on and specify different border thickness and corner radius on each side.

[Online documentation](#)

[Navigate to examples](#)

# Border

## First Look



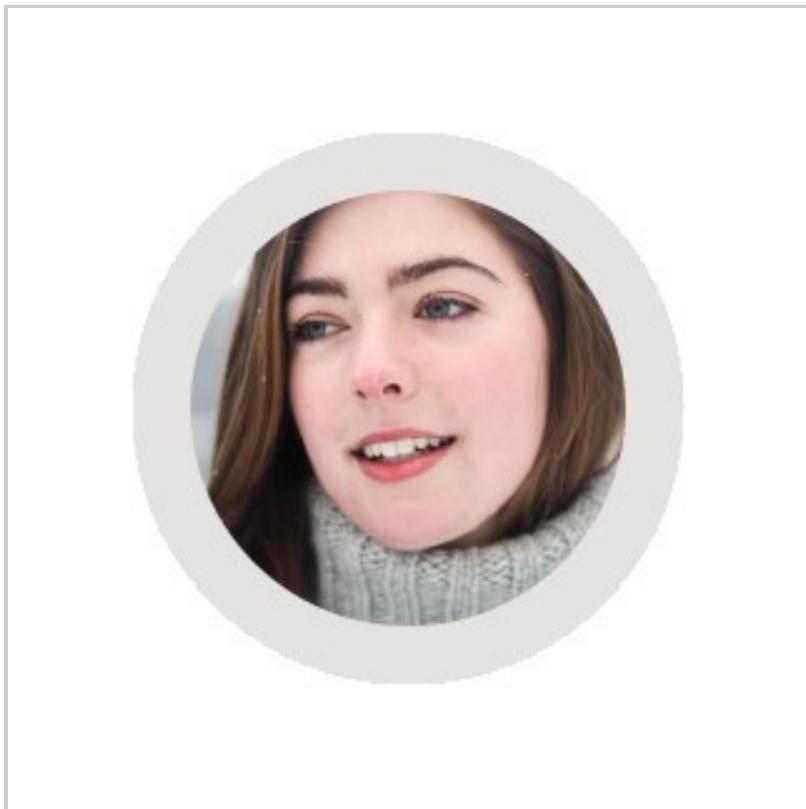
This example shows how you could wrap RadBorder around different Views, such as Label and Image. In the case of the Image you could benefit from RadBorder's clipping functionality and have circular images and in the case of the Label you have different corner radius on the four angles.

[See the code](#)

[Navigate to example](#)

# Border

## Configuration



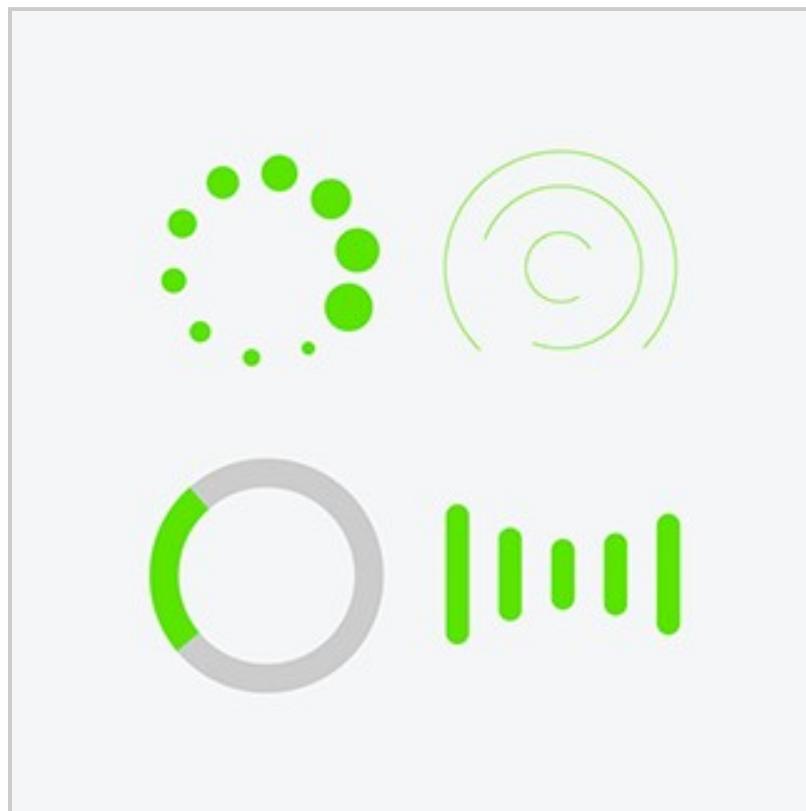
RadBorder provides a few useful properties for configuring its color, thickness and corner radius. This example lets you play with these properties and see the result right away.

[See the code](#)

[Navigate to example](#)

# BusyIndicator

RadBusyIndicator allows you to display a notification whenever a longer-running process is being handled by the application, thus making the user experience smoother.



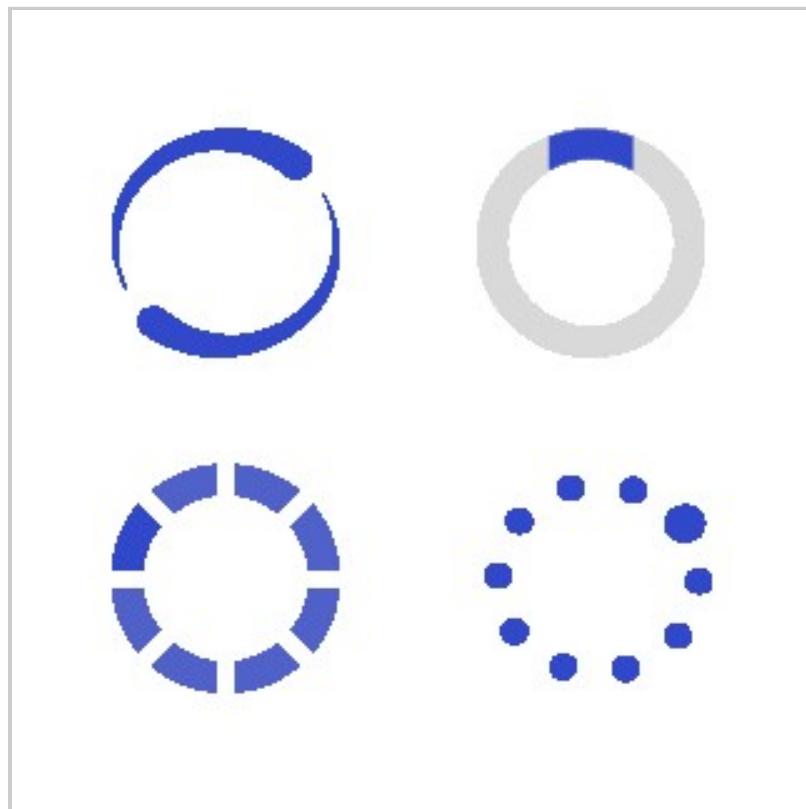
Telerik UI for Xamarin BusyIndicator enables you to display a notification whenever a longer-running process is being handled by the application. This makes the UI more informative and the user experience smoother.

[Online documentation](#)

[Navigate to examples](#)

# BusyIndicator

## Theming



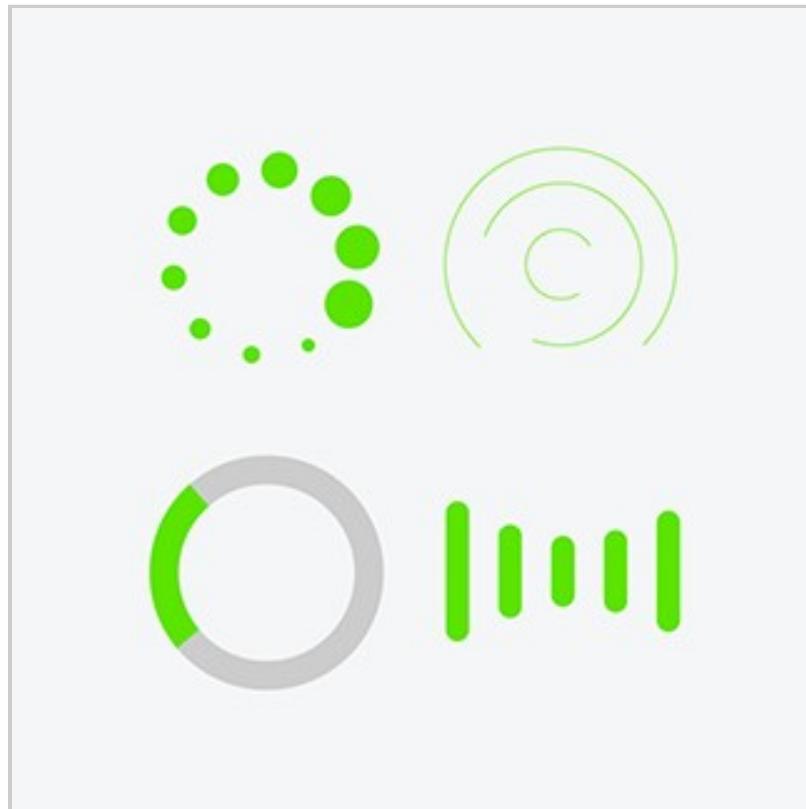
An example demonstrating the Blue Theme of the BusyIndicator.

[See the code](#)

[Navigate to example](#)

# BusyIndicator

## Gallery



An example demonstrating the BusyIndicator.

[See the code](#)

[Navigate to example](#)

# BusyIndicator

## Custom Indicator



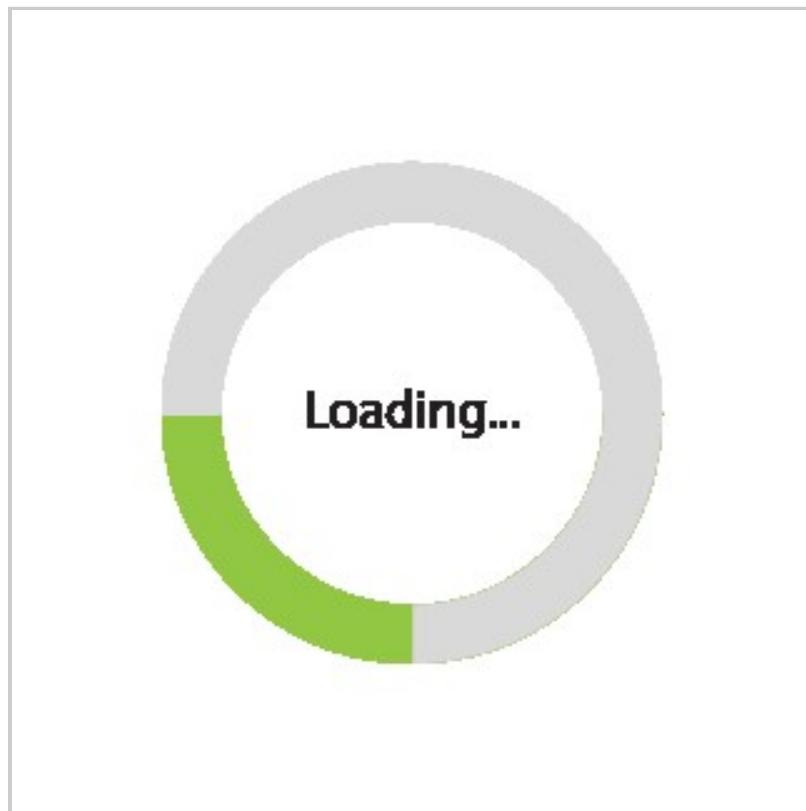
An example demonstrating custom BusyIndicator.

[See the code](#)

[Navigate to example](#)

# BusyIndicator

## Custom Busy Content



An example demonstrating a BusyIndicator with custom busy content and a custom busy content template.

[See the code](#)

[Navigate to example](#)

# Button

RadButton is a button control which enhances the standard Xamarin.Forms Button, by providing you full control over its visual appearance.

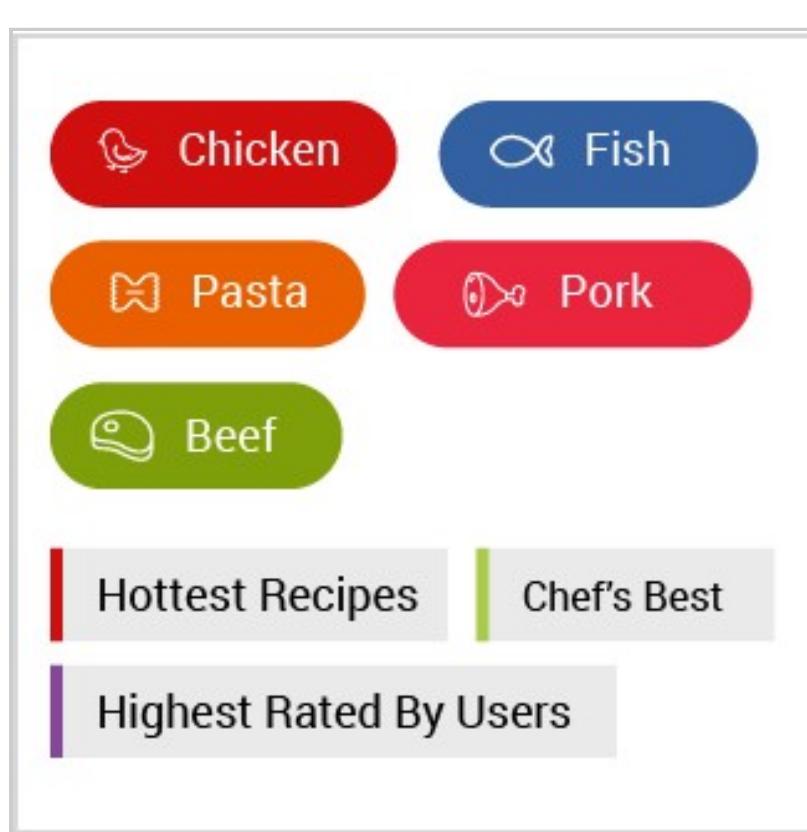


Telerik UI for Xamarin Button is a button control which enhances the standard Xamarin Forms Button, by providing you full control over its visual appearance and adding support for our Theming mechanism. UI customization is possible through the provided themes, or you can add rotation, shapes, transparency, text, and backgrounds and images.

[Online documentation](#)

[Navigate to examples](#)

# Button



An example demonstrating the Button.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

RadCalendar extends native UI functionality and supports various view modes, selection, appointments, styling customizations.



Telerik UI for Xamarin Calendar allows you to easily implement various calendar scenarios in your apps built with Xamarin or Xamarin.Forms. The control capitalizes on all the innate benefits of the native UI but exposes all objects and properties in C#, allowing for no-compromise customization and flexibility. It supports various view modes (year,

[Online documentation](#)

[Navigate to examples](#)

# Calendar & Scheduling

## Month View



The example demonstrates how the Calendar control displays dates of a month, showing how events information can be integrated in the Calendar cell's content. You can control the selection mode from the SideDrawer to show different events in the Agenda View

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Appointment Template

**Security Training**

Details for the event come here - for example place, participants and add. information ...



**One to One Meeting**

Details for the event come here - for example place, participants and add. information ...

**Marketing Research Results**

Details for the event come here - for example place, participants and add. information ...



**User Interview with VR Team's Client**

Details for the event come here - for example place, participants and add. information ...



**Cake Time with Xamarin Team**

Location: 601



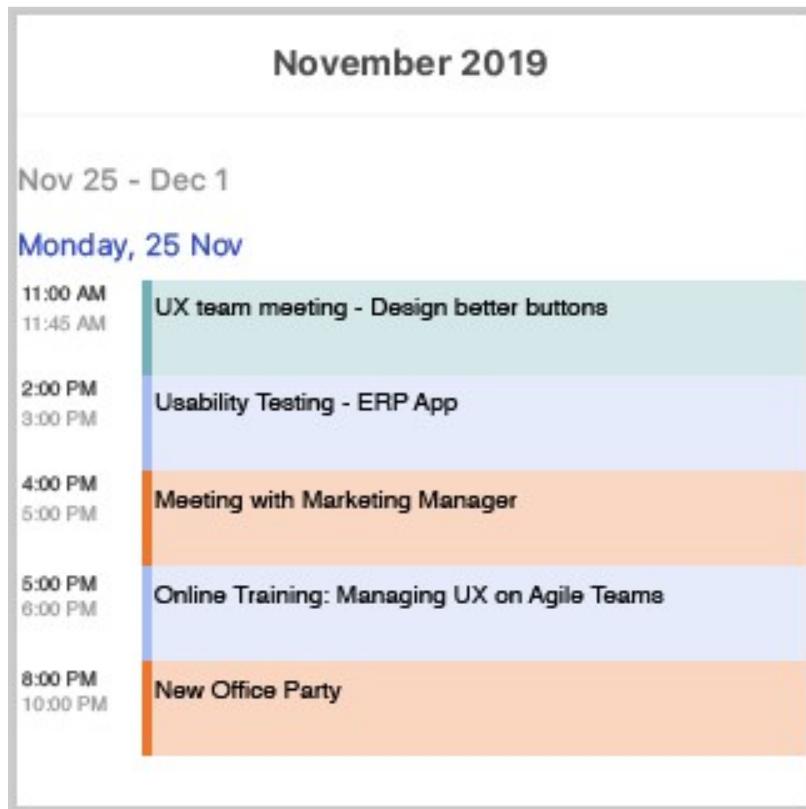
This example demonstrates how to create fully customized appointments and display them in Day & MultiDay views.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Agenda View Configuration



This example presents various configuration options such as different formats for the items you could utilize to make the AgendaView look the way you'd need.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Agenda View Customization



Demonstrates how you could customize the visual appearance of the Agenda view through the `AgendaItemStyleSelector`.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Special Slots



This example demonstrates how specific slots of the Calendar could be visualized as special using the `SpecialSlotsSource` functionality. The `SpecialSlotStyleSelector` allows you to apply different look for the slots.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Day View



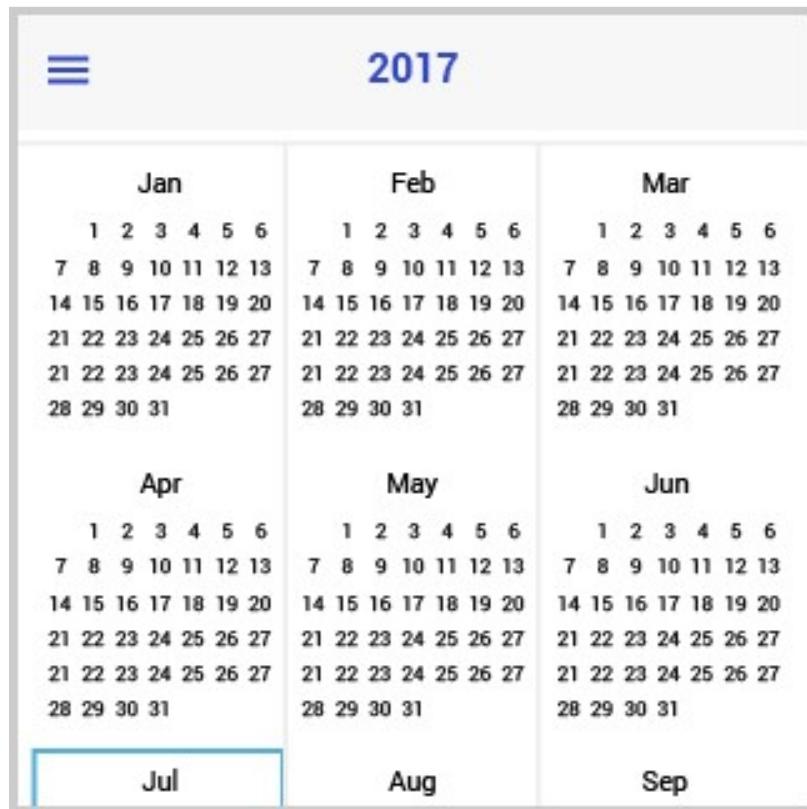
Day view mode of RadCalendar with all day area, configurable time ruler and API allowing you to schedule appointments and achieve your scheduling functionality.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Scheduling



This example demonstrates the scheduling capabilities and some of the built-in view modes of the control. The end-user could easily switch the modes using the toolbar or using the UI provided in the SideDrawer control.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Theming



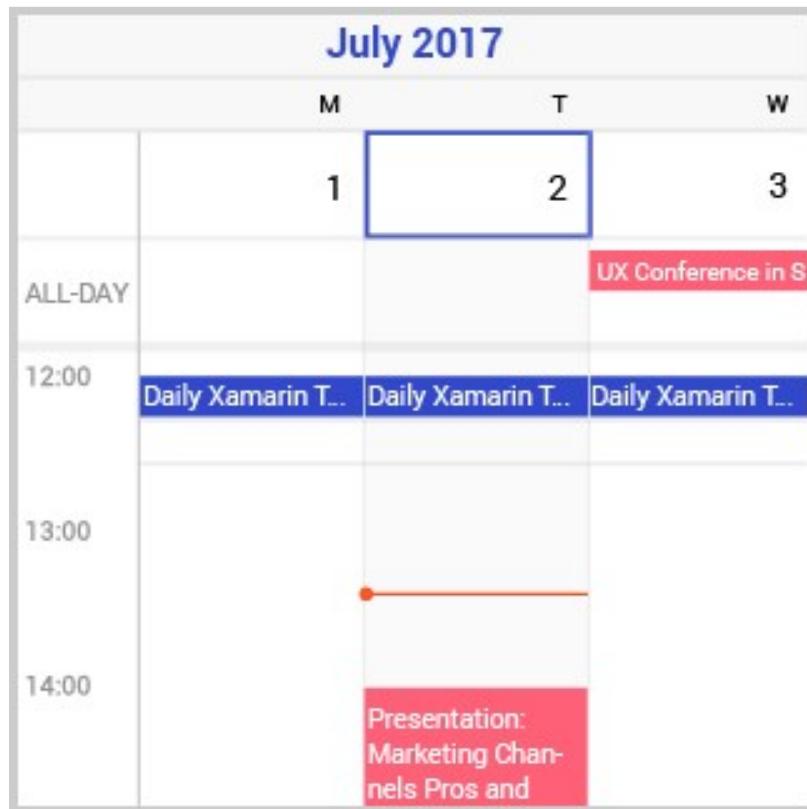
This example demonstrates the Blue theme, which is available for styling the control.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## MultiDay View



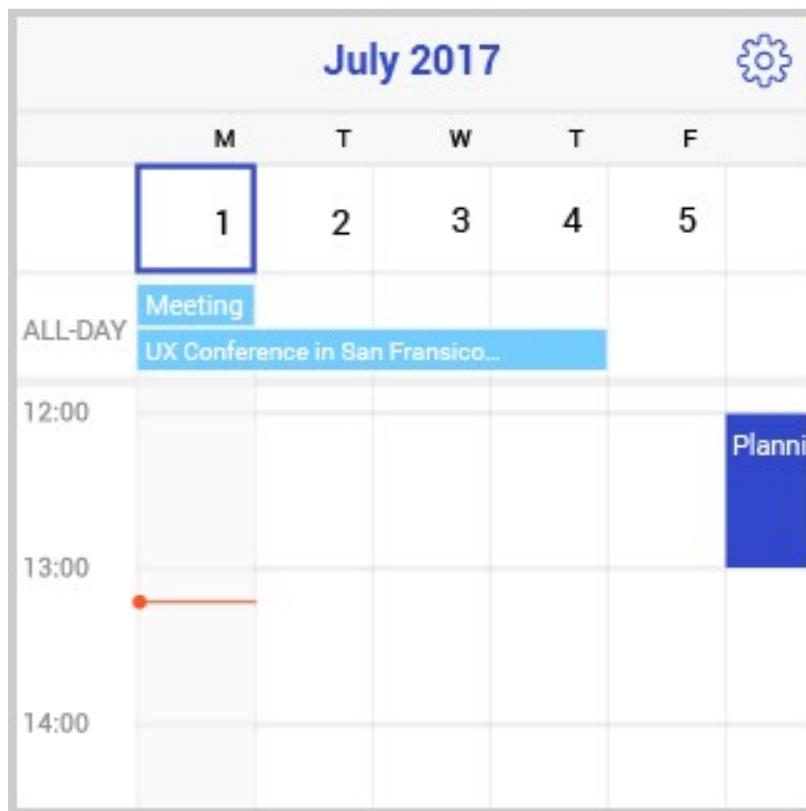
Demonstrates displaying the appointments of several people in multi-day view mode. The appointments are styled in such a way to help users quickly distinguish the person they refer to.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## MultiDay View Configuration



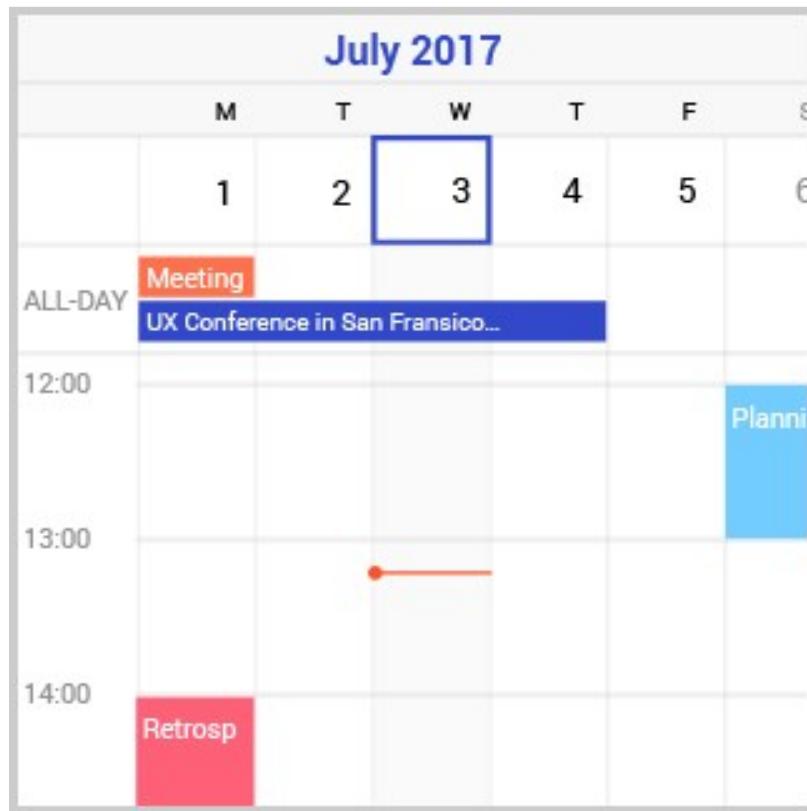
This example presents various configuration options such as day start and end times, timeline settings and weekend display, you could utilize to make the MultiDay view look the way you'd need.

[See the code](#)

[Navigate to example](#)

# Calendar & Scheduling

## Add/Edit Appointments



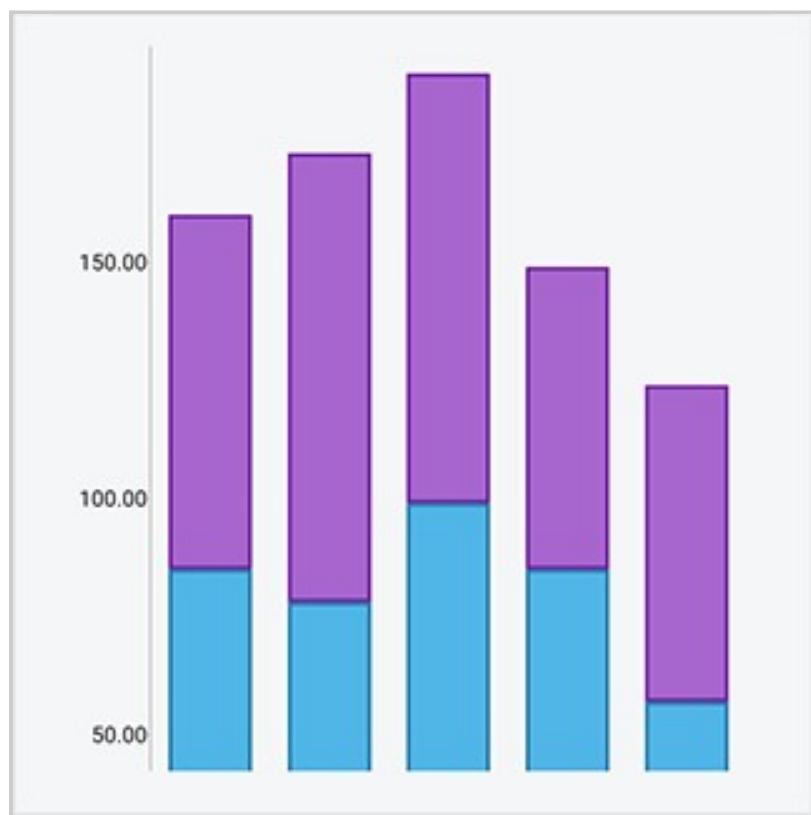
This example demonstrates display and management of appointments in MultiDay view. The end-user could easily add and edit appointments by tapping on the timeslot/appointment respectively. Delete option is also available from the appointment edit page.

[See the code](#)

[Navigate to example](#)

# Chart

RadChart is a feature-rich, intuitive and easy to use data visualization control which employs the Xamarin technology and allows you to build native iOS, Android and UWP apps in C#.



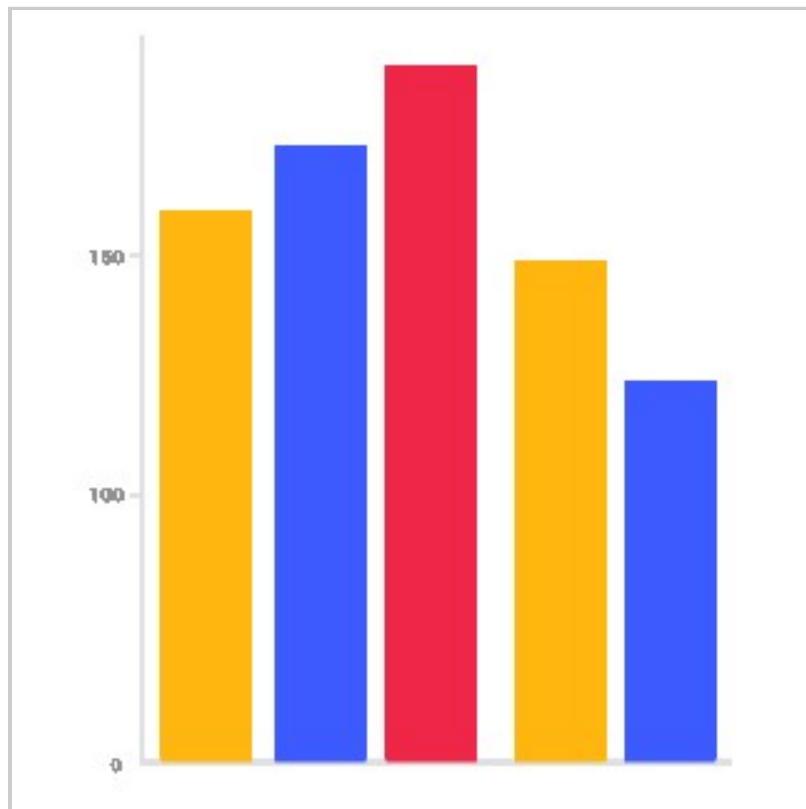
Telerik UI for Xamarin Chart is a feature-rich, intuitive and easy to use data visualization control which employs the Xamarin.Forms technology and allows you to build native apps in C#. It supports various chart types (bar, line, area, candlestick, scatter and more), various axes types (categorical, numeric, date-time and more), selection, annotations,

[Online documentation](#)

[Navigate to examples](#)

# Chart

## Theming



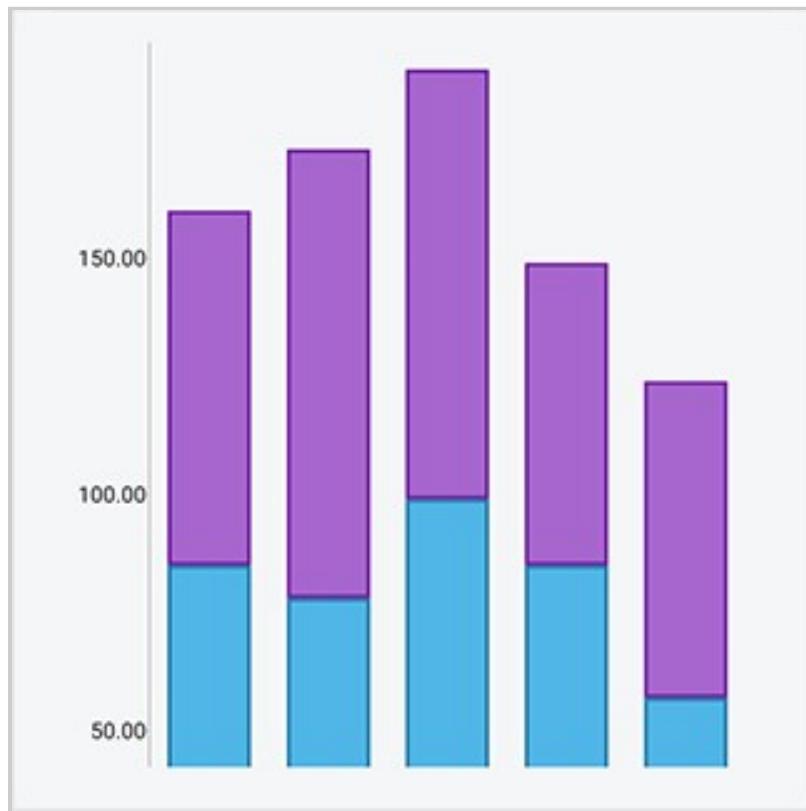
Demonstrates the theming capabilities of the chart.

[See the code](#)

[Navigate to example](#)

# Chart

## Bar Series



Bar chart is one of the most common type of graphs used to display data. Sometimes known as 'column charts' depending on the orientation, bar charts are used to visualize data points as bar blocks where the width/height of each bar denotes the magnitude of its value.

[See the code](#)

[Navigate to example](#)

# Chart

## Line Series



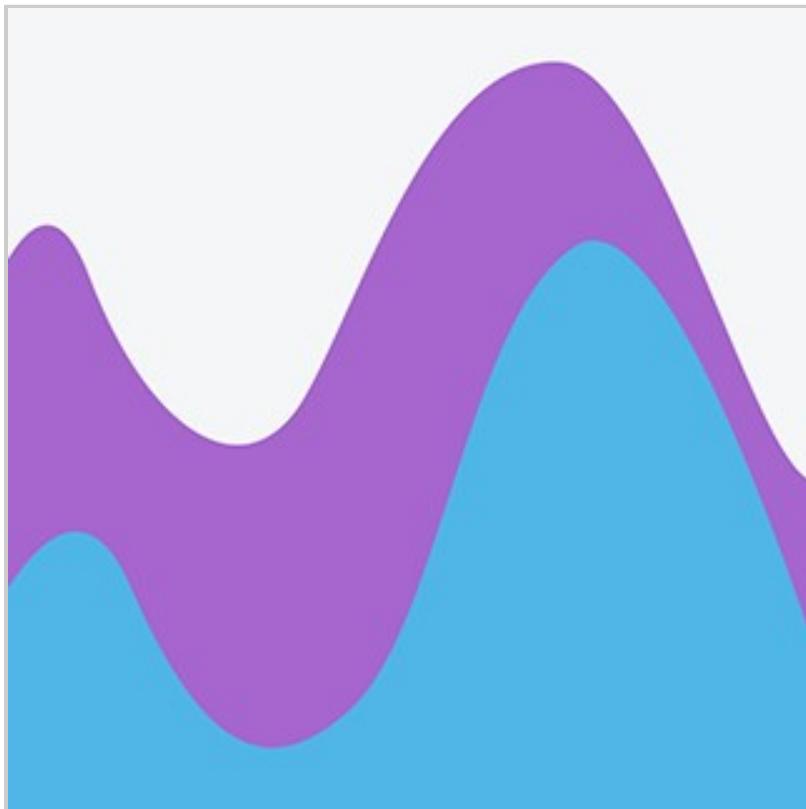
Line charts are often used to plot changes in data over time, such as monthly temperature changes or daily changes in stock market prices. If you are plotting changes in data over time, time is plotted along the horizontal axis and your other data is plotted as individual points along the vertical axis.

[See the code](#)

[Navigate to example](#)

# Chart

## Area Series



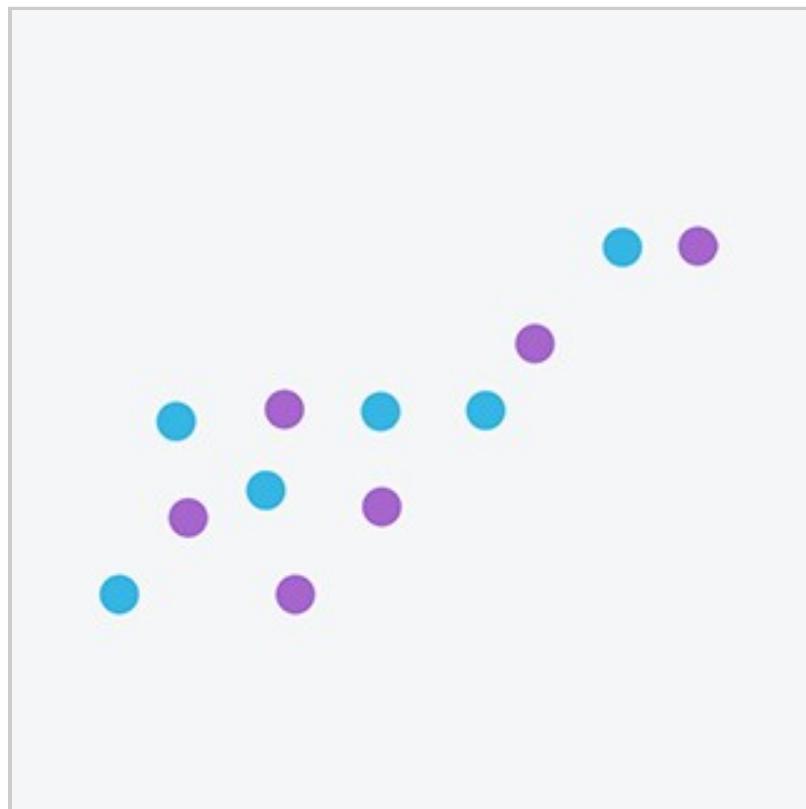
Area charts are much like line charts, but they display different colors in the areas below the lines. This colorful and visual display distinguishes the data more clearly. The Area charts support Stack, Stack100 and standard plotting modes.

[See the code](#)

[Navigate to example](#)

# Chart

## Scatter Series



A scatter chart is a two-dimensional graphical representation of a set of data where each x/y variable is represented on the graph as a dot. This type of chart can be used in to visually describe relationships between two numerical parameters or to represent distributions.

[See the code](#)

[Navigate to example](#)

# Chart

## Pie & Donut Series



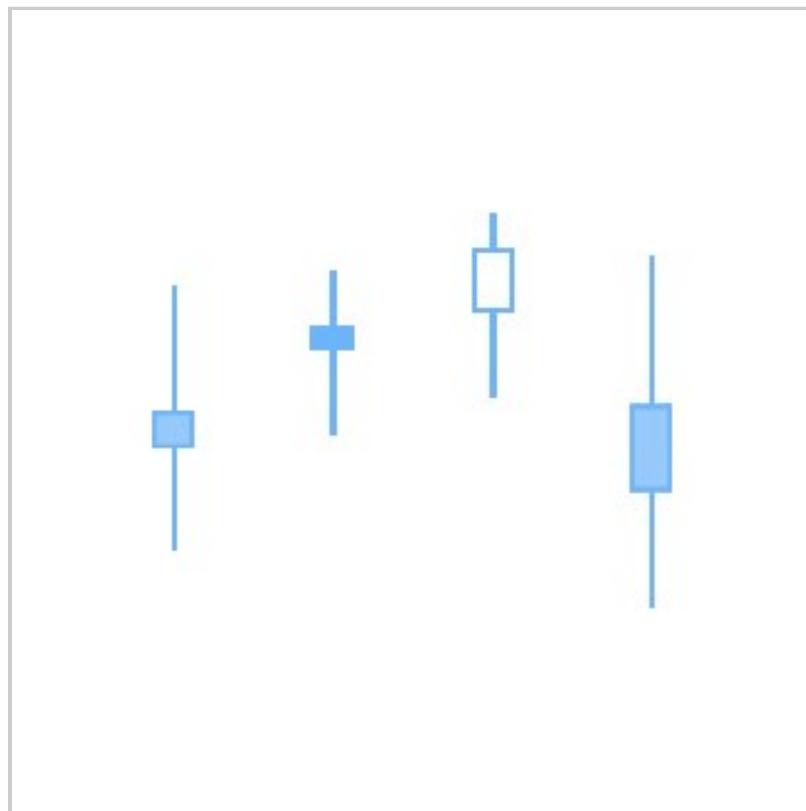
The pie chart visualizes its data in the shape of a pie. Each data item is visually represented by a pie/donut slice. The ratio between the space consumed by each slice and the space consumed by the whole chart is the same as the ratio between the value of the data point that it represents and the total value of all data points in the series.

[See the code](#)

[Navigate to example](#)

# Chart

## Financial Series



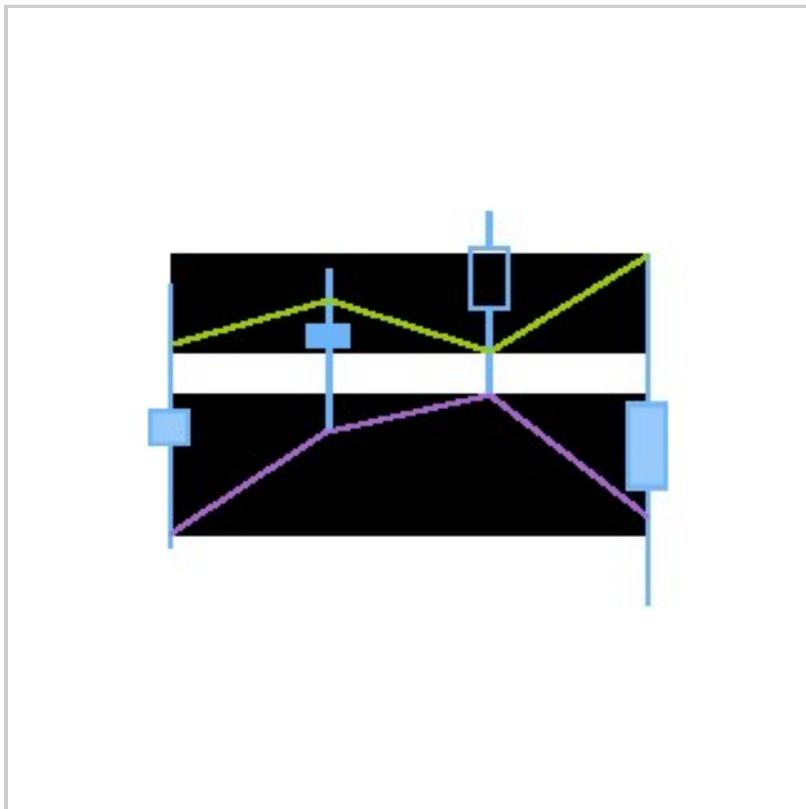
The financial series are represented by the `OhlcSeries` and `CandlestickSeries` classes. They only differ in the way they visualize the data point. OHLC stands for Open, High, Low and Close.

[See the code](#)

[Navigate to example](#)

# Chart

## Financial Indicators



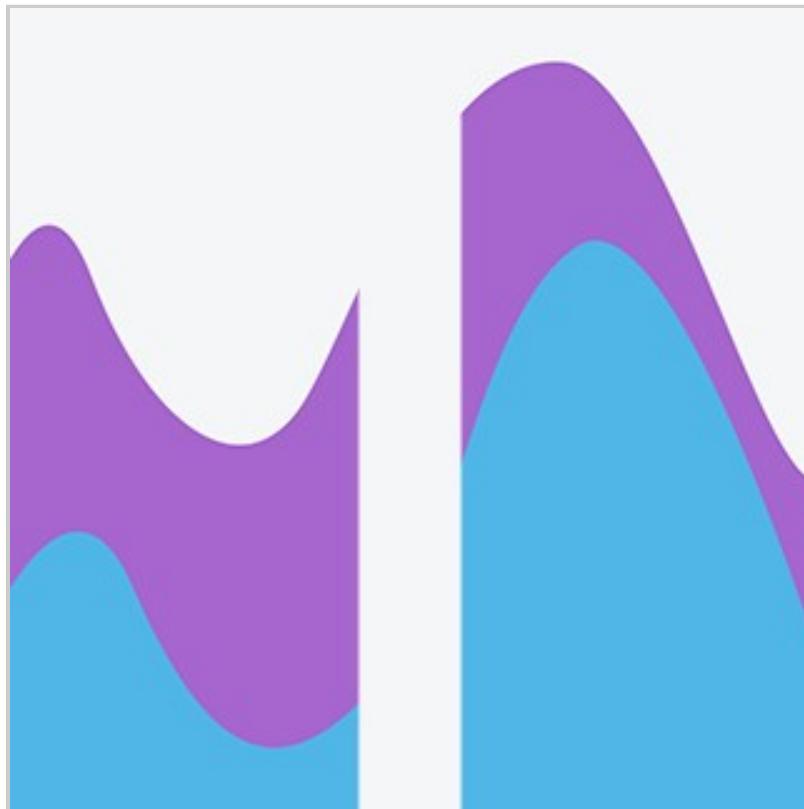
The financial indicators are a collection of DataPoints (calculated with different algorithms using the series data) visualized using Line Shape.

[See the code](#)

[Navigate to example](#)

# Chart

## Null Values



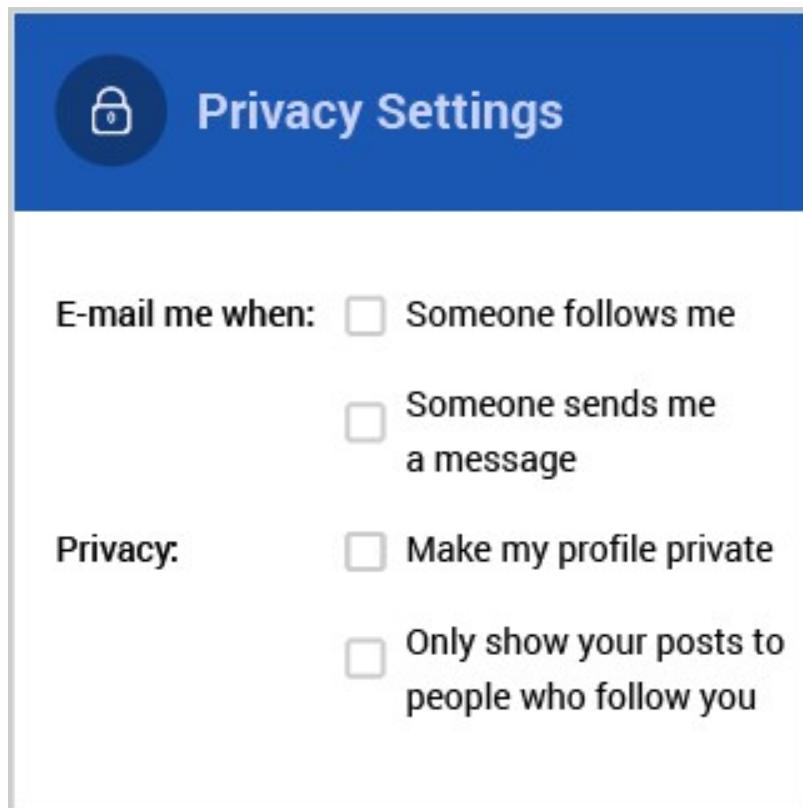
Line charts and area charts can visualize null values as gaps in the chart.

[See the code](#)

[Navigate to example](#)

## CheckBox

RadCheckBox for Xamarin is a checkbox control which enables users to make a choice between two mutually exclusive options.



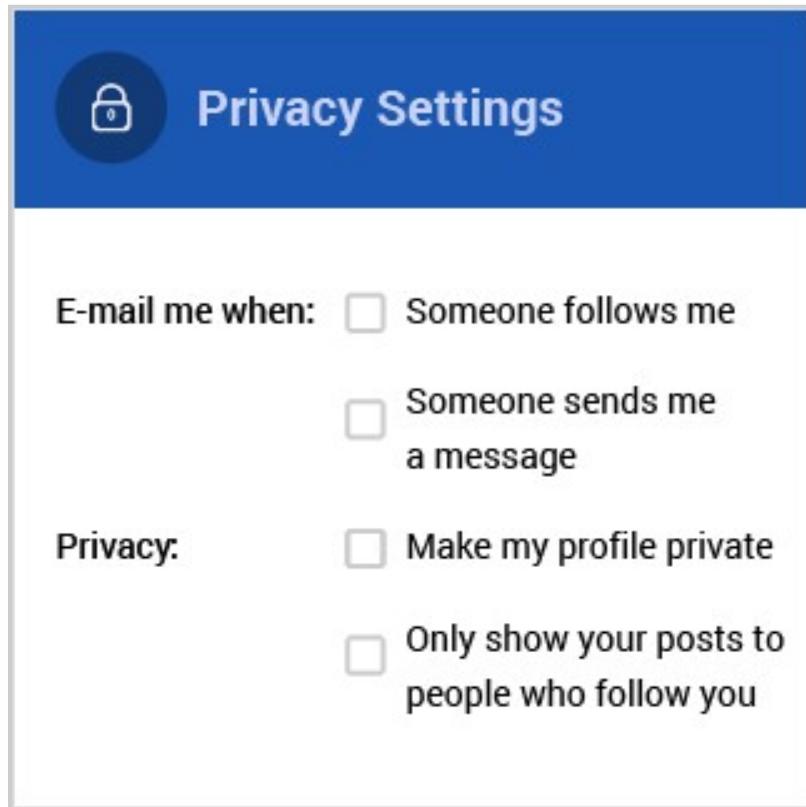
Telerik RadCheckBox for Xamarin is a checkbox control which enables users to make a choice between two mutually exclusive options. In addition, RadCheckBox provides indeterminate state which indicates the control is neither checked nor unchecked. RadCheckBox exposes various Color properties you could use to customize its visual

[Online documentation](#)

[Navigate to examples](#)

# CheckBox

## First Look



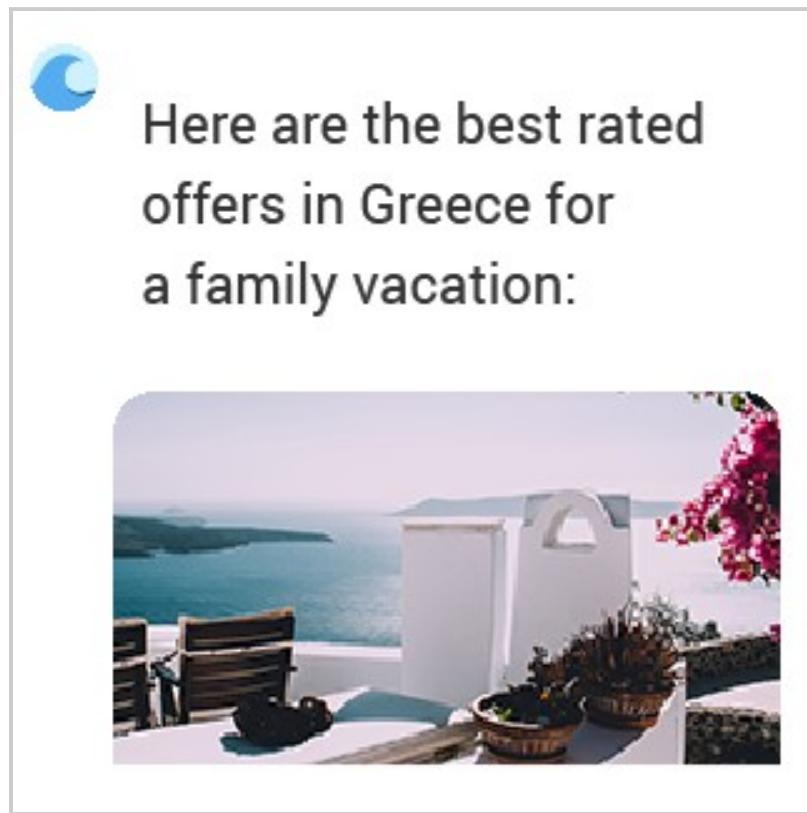
This example demonstrates how you could use a series of RadCheckBox controls, so that the end user may select several of the provided choices.

[See the code](#)

[Navigate to example](#)

## Conversational UI

RadChat for Xamarin is a UI component that provides flexible API for achieving a variety of chat and chat-bot scenarios.



 Here are the best rated offers in Greece for a family vacation:



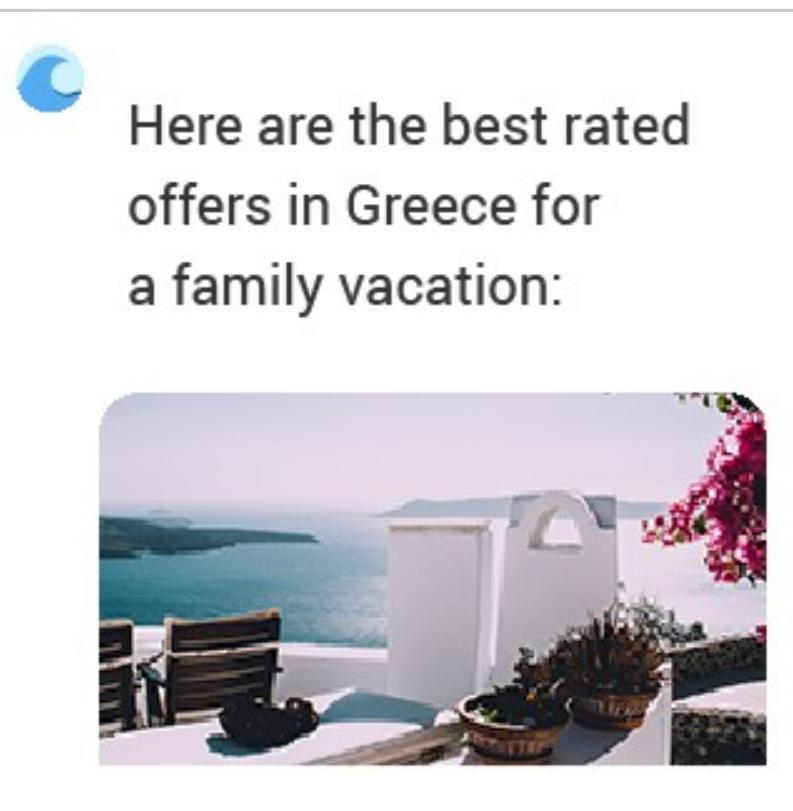
RadChat enables easy implementation of conversational UI in Xamarin applications, whether by utilizing certain chatbot framework, by following a predefined logical tree, or just for integrating P2P chat capabilities.

[Online documentation](#)

[Navigate to examples](#)

# Conversational UI

## Travel Assistance



A bot message card with a blue circular icon containing a white crescent moon. The text reads: "Here are the best rated offers in Greece for a family vacation:" followed by a photograph of a white balcony overlooking the sea and mountains.

This example demonstrates a possible implementation of the RadChat control for a travel agency by integrating it with the Microsoft Bot Framework via Azure Bot Service.

[See the code](#)

[Navigate to example](#)

# Conversational UI

## Insurance Assistance



Let me introduce you to our various insurance options



## Full Coverage

This example demonstrates a possible implementation of the RadChat control for a car insurance company by integrating it with Google's DialogFlow framework.

[See the code](#)

[Navigate to example](#)

# Conversational UI

## Healthcare Assistance



Please, confirm your details for a check-up:



Dr. Nicole Graham

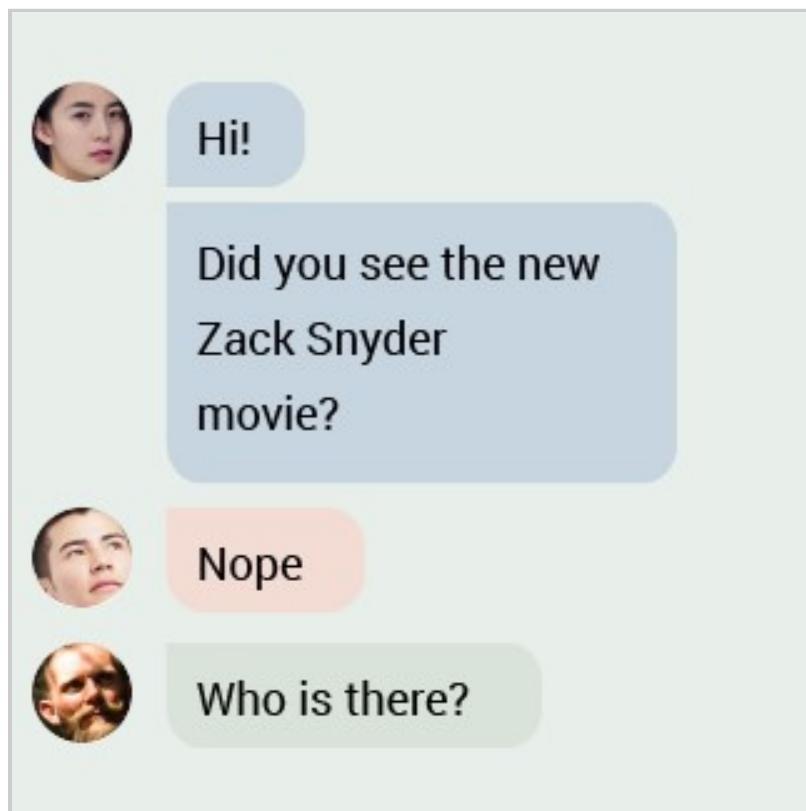
This example demonstrates a possible implementation of the RadChat control for a health insurance company by integrating it with the Microsoft Bot Framework via Azure Bot Service.

[See the code](#)

[Navigate to example](#)

# Conversational UI

## Chat Room



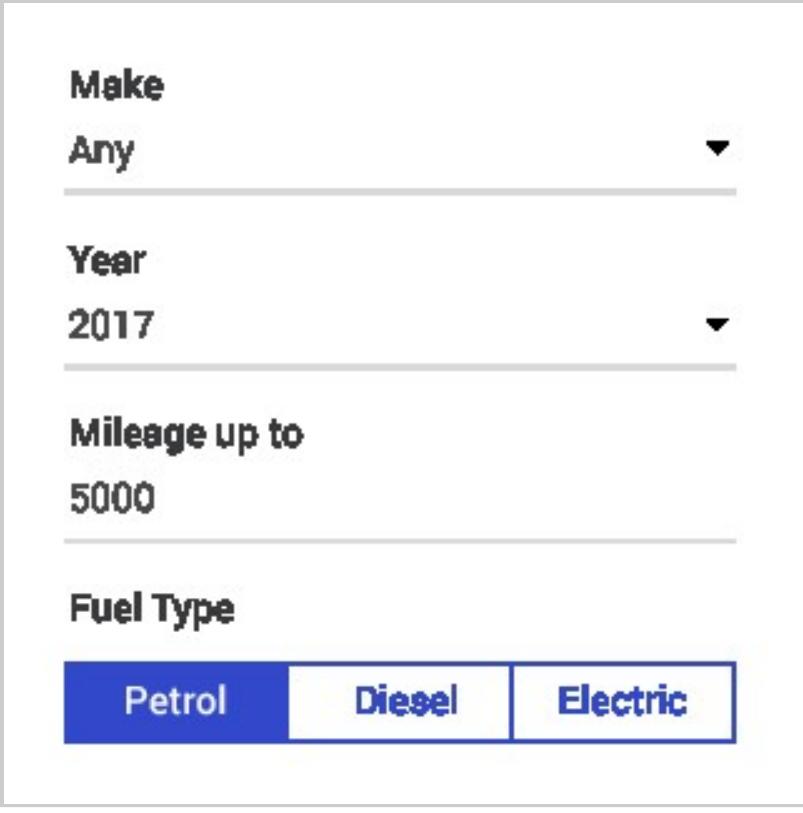
This example demonstrates one way to go about implementing a chat room with multiple participants. Some of the features that are demonstrated are: typing indicator, timebreaks, color customization, MVVM support.

[See the code](#)

[Navigate to example](#)

# DataForm

RadDataForm is a customizable component that allows you to easily create a form for collecting data and provides different validation and commit modes.



The screenshot displays a RadDataForm control with the following configuration:

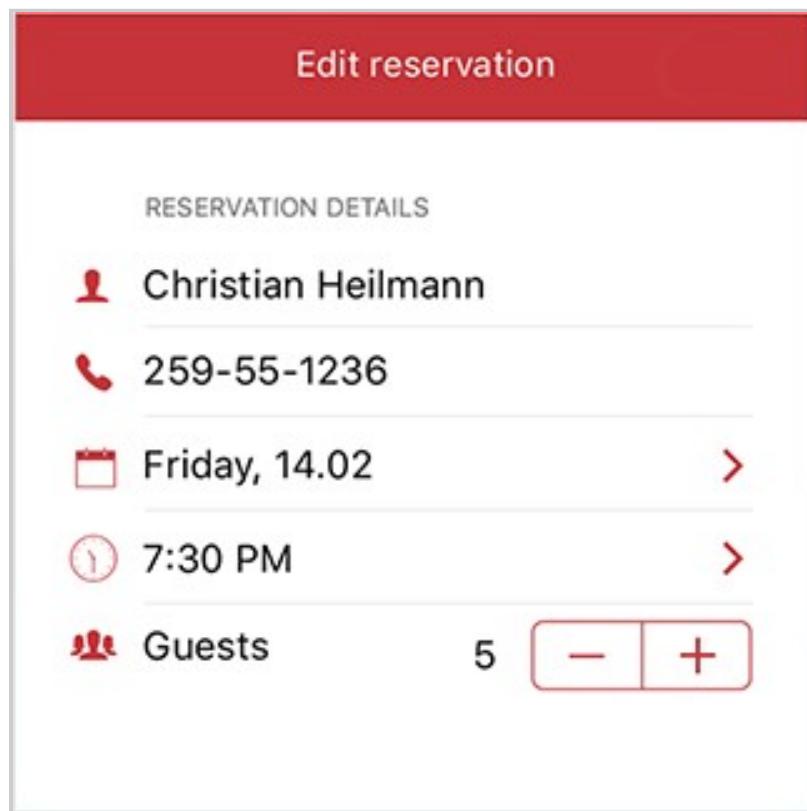
- Make:** A dropdown menu currently showing "Any".
- Year:** A dropdown menu currently showing "2017".
- Mileage up to:** A text input field containing the value "5000".
- Fuel Type:** A radio button group with three options: "Petrol", "Diesel", and "Electric". The "Petrol" option is currently selected.

Telerik UI for Xamarin DataForm allows for easy building of a data fill-in form from a single shared codebase. The control detects the types of the data object's properties and automatically displays the appropriate editors. It provides a set of built-in editors for the available primitive types such as numeric, string, Boolean and enumerations. You can also provide your own custom editor and implement it easily thanks to the rich API of the control.

[Online documentation](#)

[Navigate to examples](#)

# DataForm



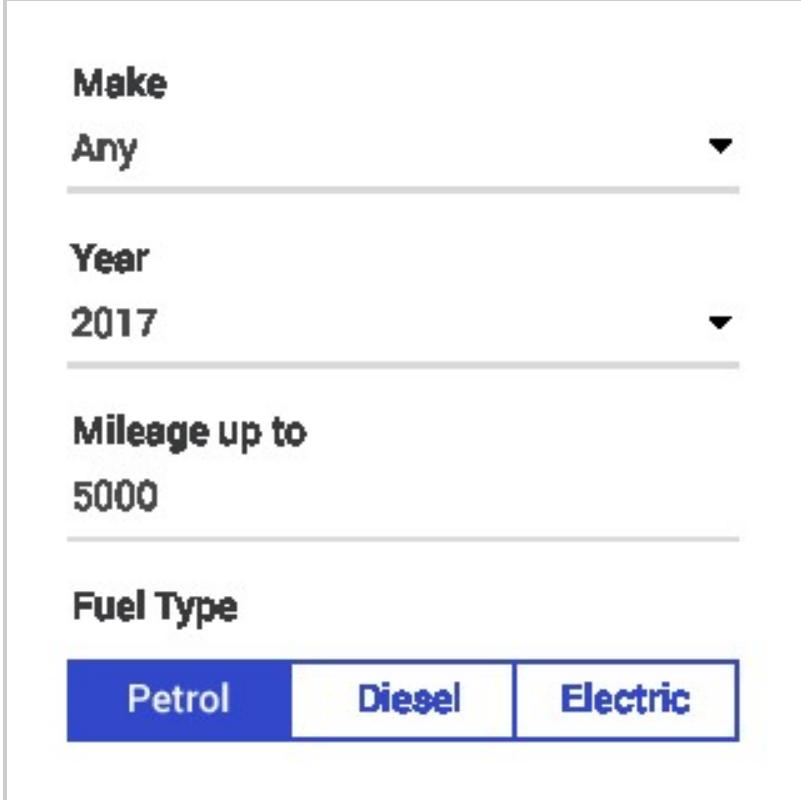
Demonstrates the most important set of features in DataForm.

[See the code](#)

[Navigate to example](#)

# DataForm

## Theming



The screenshot shows a DataForm interface with the following fields:

- Make:** A dropdown menu set to "Any".
- Year:** A dropdown menu set to "2017".
- Mileage up to:** A text input field containing "5000".
- Fuel Type:** A segmented button group with three options: Petrol (selected), Diesel, and Electric.

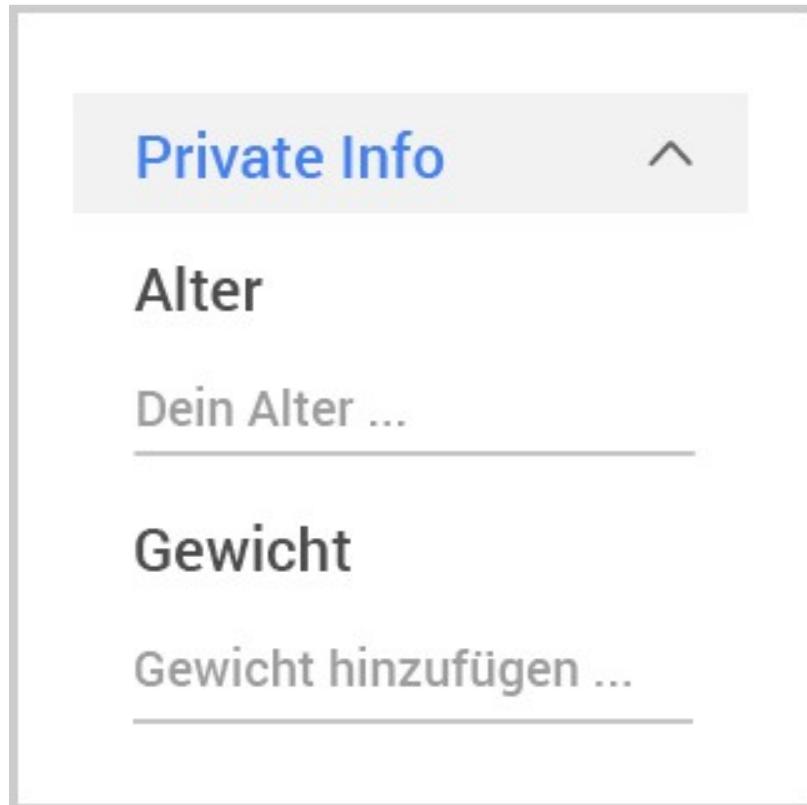
Demonstrates the theming of the DataForm.

[See the code](#)

[Navigate to example](#)

# DataForm

## Localization & Globalization



Demonstrates how to set up the globalization of the DataForm.

[See the code](#)

[Navigate to example](#)

## DataGrid

RadDataGrid is a powerful and feature rich control that allows you to easily visualize and edit tabular represented data in your apps.

Ship owner	Order ID	Ship name
▼ Alfreds Futterkiste		
	10643	Alfreds Futterkiste
	10692	Alfreds Futterkiste
	10702	Alfreds Futterkiste
	10835	Alfreds Futterkiste
	10952	Alfreds Futterkiste

Telerik UI for Xamarin DataGrid is a powerful and feature rich control that allows you to easily visualize and edit tabular represented data. The control provides the ability to easily perform operations like sorting, filtering, grouping and editing over the underlying data.

[Online documentation](#)

[Navigate to examples](#)

# DataGrid

## First Look

Ship owner	Order ID	* Ship name
▼ Alfreds Futterkiste		
	10643	Alfreds Futterkiste
	10692	Alfreds Futterkiste
	10702	Alfreds Futterkiste
	10835	Alfreds Futterkiste
	10952	Alfreds Futterkiste

Demonstrates how you can easily configure the DataGrid to display data and perform selection and editing on it.

[See the code](#)

[Navigate to example](#)

# DataGrid

## Theming

Order ID	Ship Name
10246	Vins et alcools Chevalier
10247	Toms Spezialitäten
10248	Hanari Carnes
10249	Victuailles en stock
10250	Suprèmes délices
10251	Hanari Carnes
10252	Chop-suey Chinese

Demonstrates how the DataGrid could easily be styled using the built-in Blue theme.

[See the code](#)

[Navigate to example](#)

# DataGrid

## Customization

Sales Person	:	Sales (\$)	:	City	:
US					USA
CE					Central
	River Johnston	\$ 115 118		Chicago	
	Carmen Ball	\$ 23 896		Chicago	

Demonstrates how you could customize the visual appearance of the DataGrid control through various Style and StyleSelector properties such as RowBackgroundStyle, AlternateRowBackgroundStyle, SelectionStyle, GroupHeaderStyleSelector, etc.

[See the code](#)

[Navigate to example](#)

# DataGrid

## Load On Demand

Sales Person	Sales (\$)	City
 Jesse Avila	\$ 123 456	Atlanta
 Blake Wells	\$ 89 990	Chicago
 River Johnston	\$ 115 118	Chicago

 Load More

Loading a large data set on a mobile device has its challenges. One of the most popular approaches is using incremental data loading in the moment the items need to be visualized. This example demonstrates how you could utilize the LoadOnDemand feature of RadDataGrid to load the data items when the "Load More" button is activated. Additionally, RadDataGrid offers automatic LoadOnDemand mode in which the items are loaded once you scroll near to the last available record.

[See the code](#)

[Navigate to example](#)

# DataGrid

## CRUD Operations

CRUD Operations		
Order ID	Order Date	Ship Name
10248	12.08.2018	Victuailles en

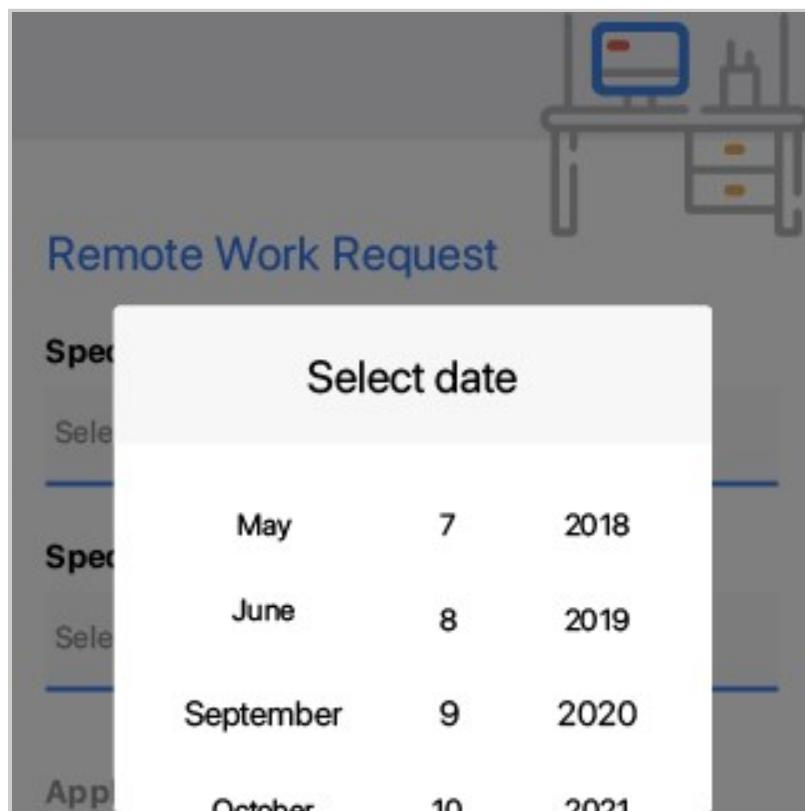
An example that shows the DataGrid's CRUD capabilities such as addition, deletion and editing.

[See the code](#)

[Navigate to example](#)

# Date Picker

RadDatePicker provides an easy way to pick a date.



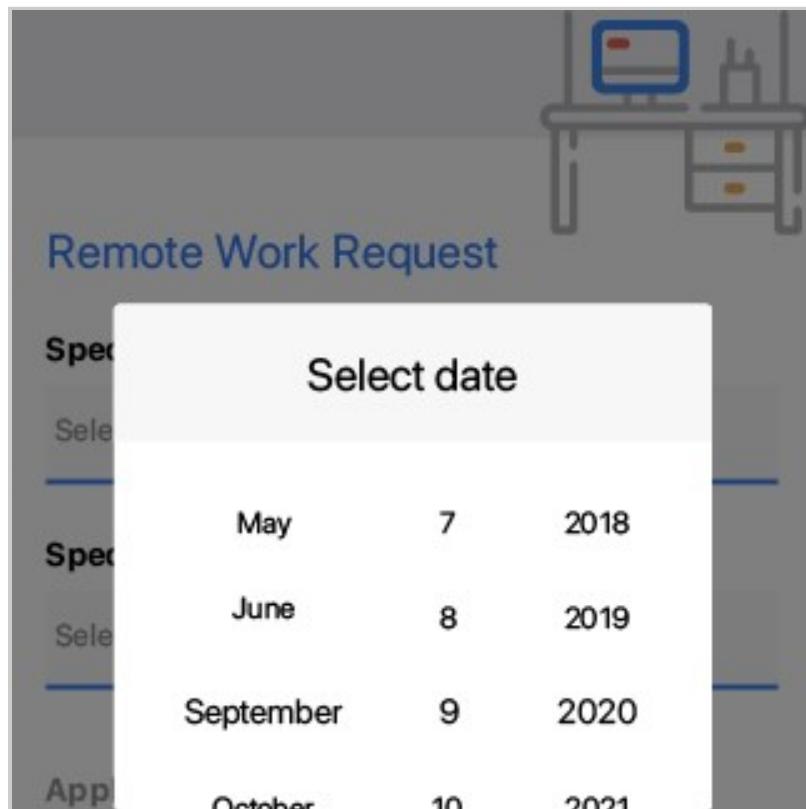
Telerik Date Picker for Xamarin allows the users to select a date. Its items are visualized inside a popup. Date Picker control has a number of features which allows you to set a date range, date format and fully customize the dialog appearance such as its header and footer.

[Online documentation](#)

[Navigate to examples](#)

# Date Picker

## First Look



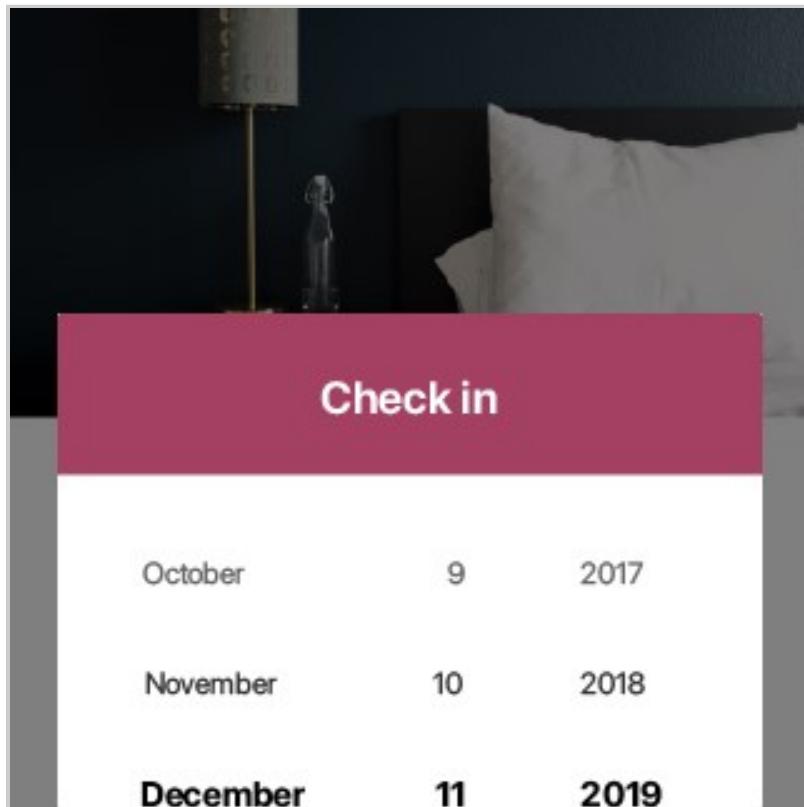
RadDatePicker provides the users an easy way to pick a date. The control allows you to define minimum and maximum date. This example demonstrates how you can setup the DatePicker control, so the app users can pick a date and send a request to work remotely.

[See the code](#)

[Navigate to example](#)

# Date Picker

## Customization



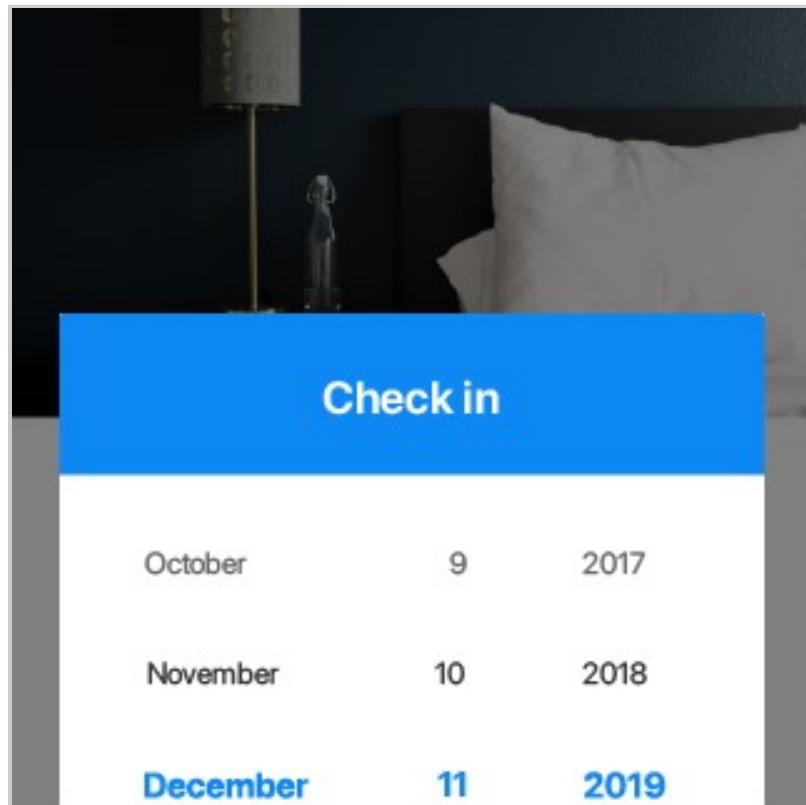
This example demonstrates how to customize the visual appearance of RadDatePicker using the control's flexible styling API.

[See the code](#)

[Navigate to example](#)

# Date Picker

## Configuration



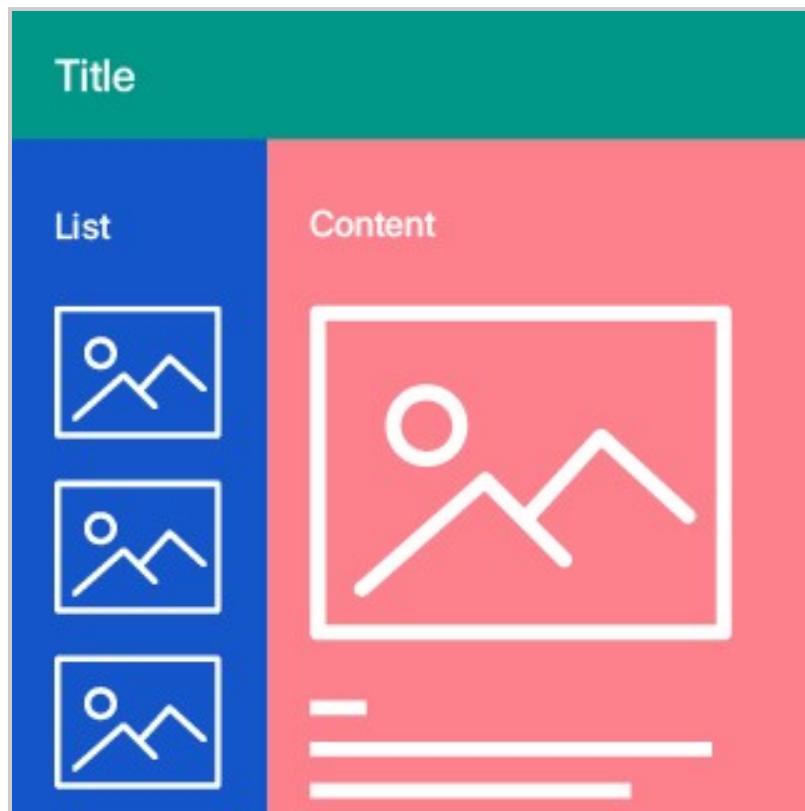
This example presents various configuration options that the RadDatePicker provides. You can change the Popup BackgroundColor, Accept and Cancel button text and color, etc.

[See the code](#)

[Navigate to example](#)

# DockLayout

RadDockLayout is a layout that provides a mechanism for child elements to be docked to the left, right, top or bottom edge or to occupy the center area of the layout.



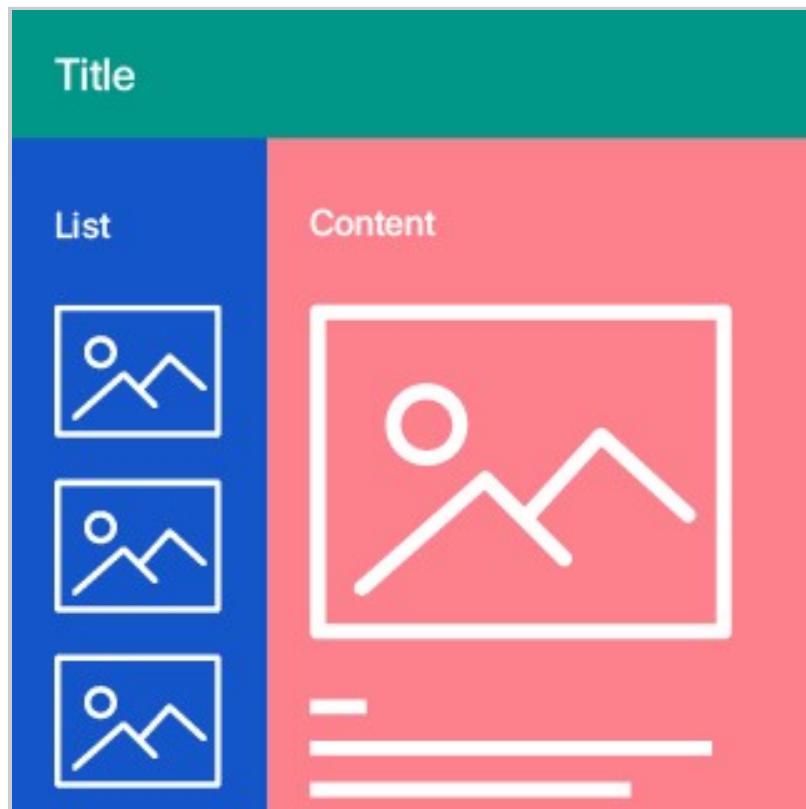
Telerik UI for Xamarin DockLayout is a layout that provides a mechanism for child elements to be docked to the left, right, top or bottom edge of the layout. You could easily define the docking side of each element through a `RadDockLayout.Dock` attached property. In addition, you can determine whether the last child element inside the layout

[Online documentation](#)

[Navigate to examples](#)

# DockLayout

## First Look



Demonstrates how you can easily configure the DockLayout.

[See the code](#)

[Navigate to example](#)

# Entry

RadEntry is a text input control, which accepts string input from the users and provides means for customizing its look & feel.



Telerik UI for Xamarin Entry is a text input control, which accepts string input from the users and provides consistent look and feel with the rest of the Telerik UI for Xamarin components through innovative theming mechanism, including customization of the borders of the Entry.

[Online documentation](#)

[Navigate to examples](#)

# Entry

## First Look



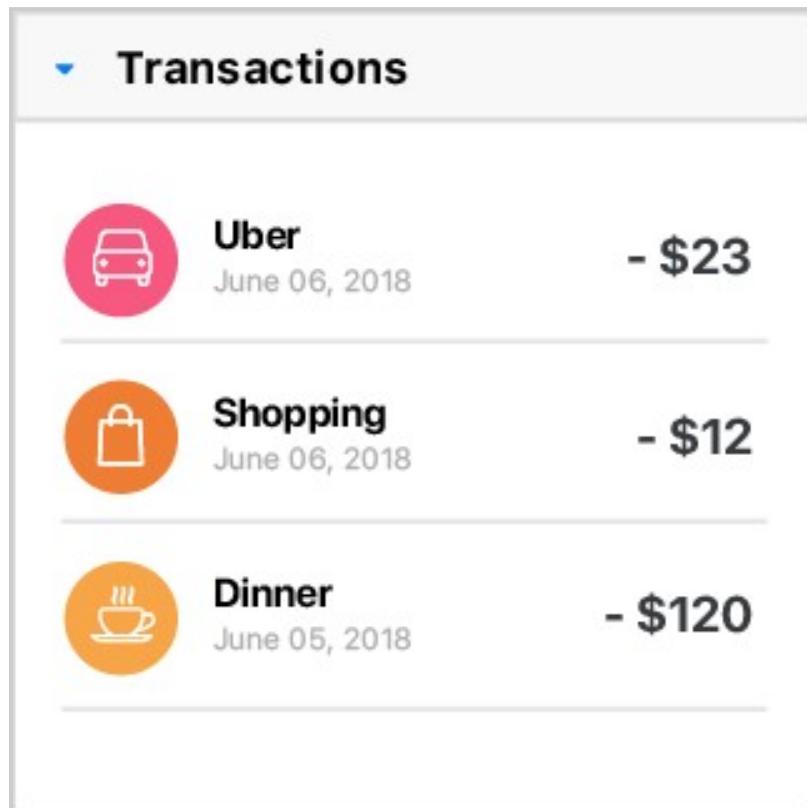
Demonstrates how you could utilize the key features of RadEntry such as watermark support and border styling to achieve slick design across all platforms.

[See the code](#)

[Navigate to example](#)

# Expander

RadExpander for Xamarin helps you save screen space by presenting content in an expandable container that can be easily expanded/collapsed.



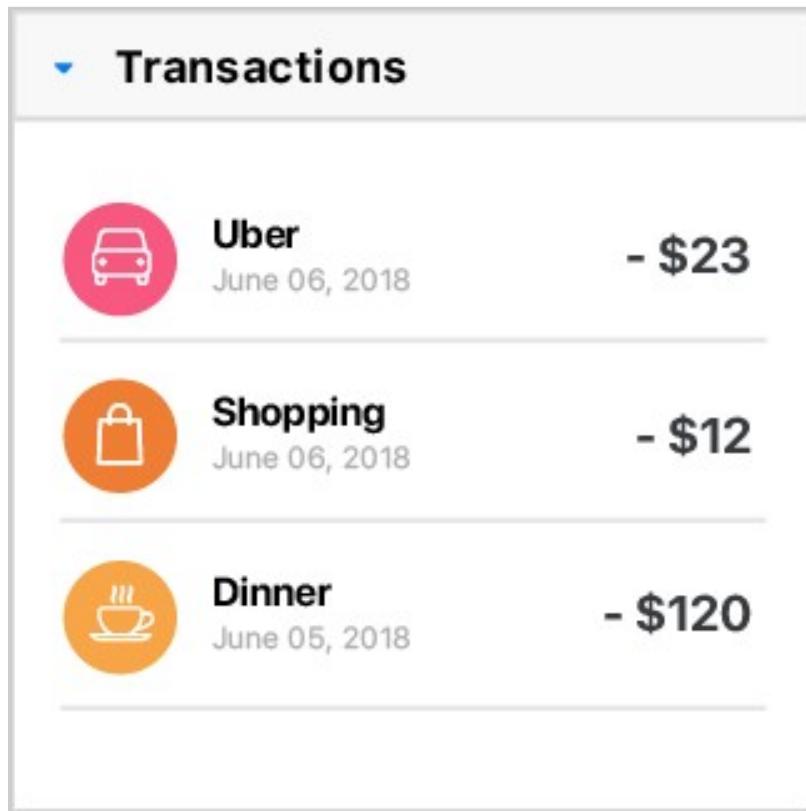
Telerik Expander for Xamarin helps you save screen space by presenting content in an expandable container that can be easily expanded/collapsed by tapping on the header of the control.

[Online documentation](#)

[Navigate to examples](#)

# Expander

## First Look



Demonstrates basic setup of RadExpander for presenting content in an expandable container.

[See the code](#)

[Navigate to example](#)

# Expander

## Customization



### Mallorca Island Vacation Deals

This beautiful island is best known for its beaches, which often overshadow its equally stunning monasteries, medieval towns and traditional markets.

[Read More](#)



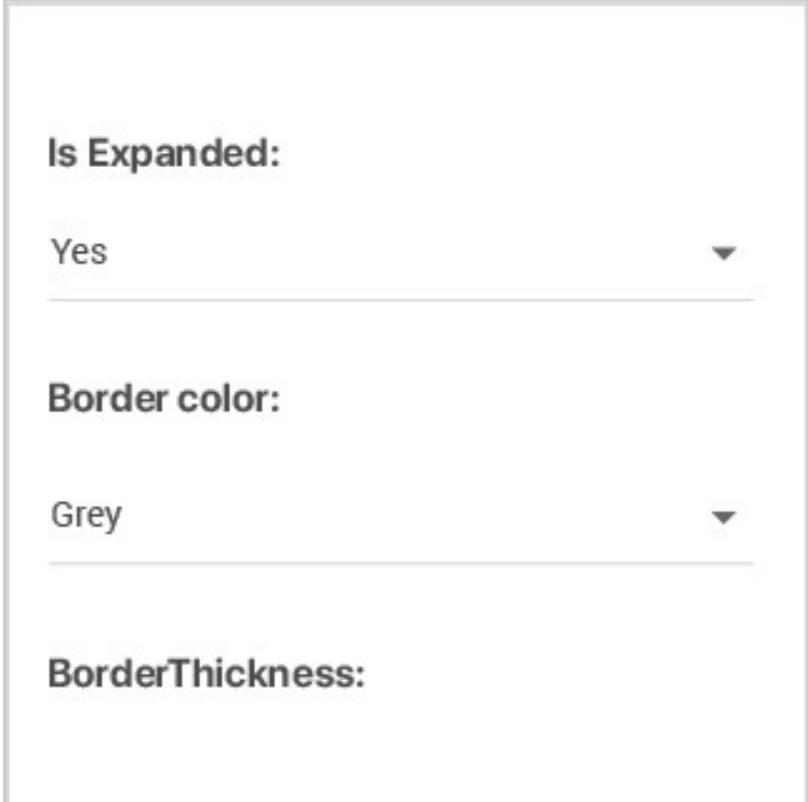
Demonstrates how you could customize the RadExpander to achieve slick design across all platforms.

[See the code](#)

[Navigate to example](#)

# Expander

## Configuration



The screenshot shows a configuration interface for an Expander control. It consists of three vertically stacked dropdown menus. The first menu is labeled "Is Expanded:" and has the value "Yes" selected. The second menu is labeled "Border color:" and has the value "Grey" selected. The third menu is labeled "BorderThickness:" and is currently empty. Each menu has a small downward arrow icon to its right.

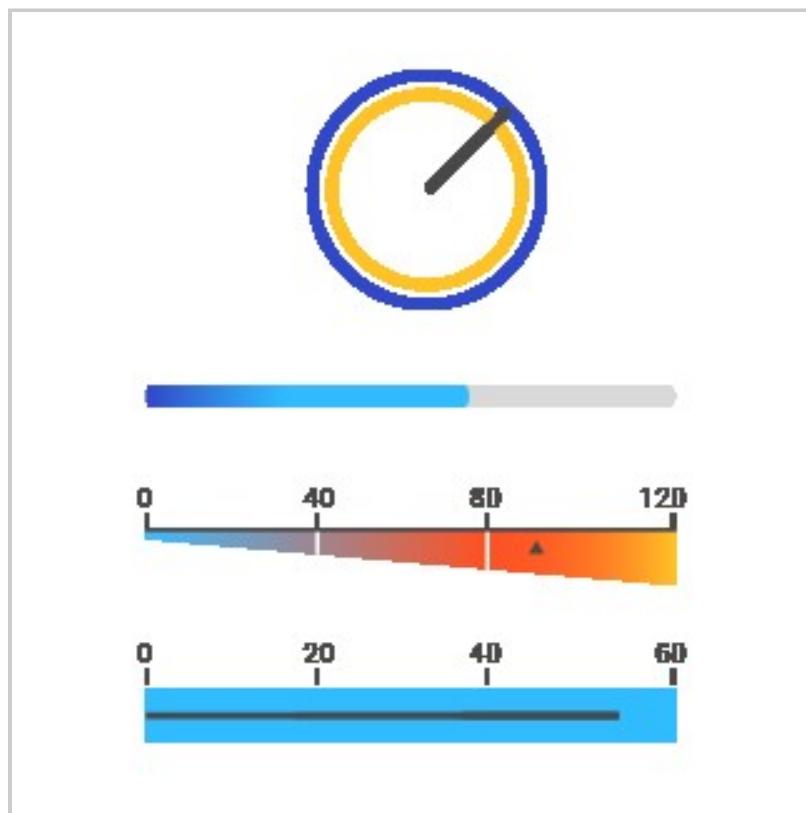
This example presents various configuration options you could use to make the Expander control look and animate per your choice.

[See the code](#)

[Navigate to example](#)

# Gauge

RadGauge serves as an instrument that indicates and gives a visual display of amount, level, or content of something. These values are presented via indicators (needles, arrows, and others).



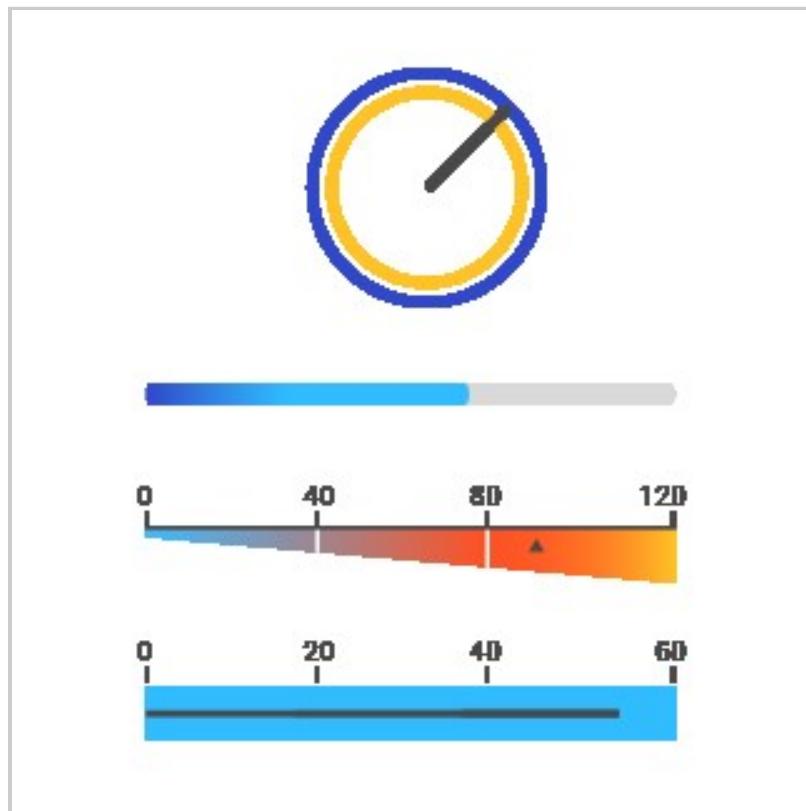
Telerik UI for Xamarin Gauge is an essential data visualization component that uses indicators to display numeric values, and uses ranges to give a context to the indicated values.

[Online documentation](#)

[Navigate to examples](#)

# Gauge

## Customization



Demonstrates the customization of the Gauge.

[See the code](#)

[Navigate to example](#)

# Gauge

## Activity Dashboard



Demonstrates the Gauge and some of the built-in indicators.

[See the code](#)

[Navigate to example](#)

# Gauge

## Radial Gauge Gallery



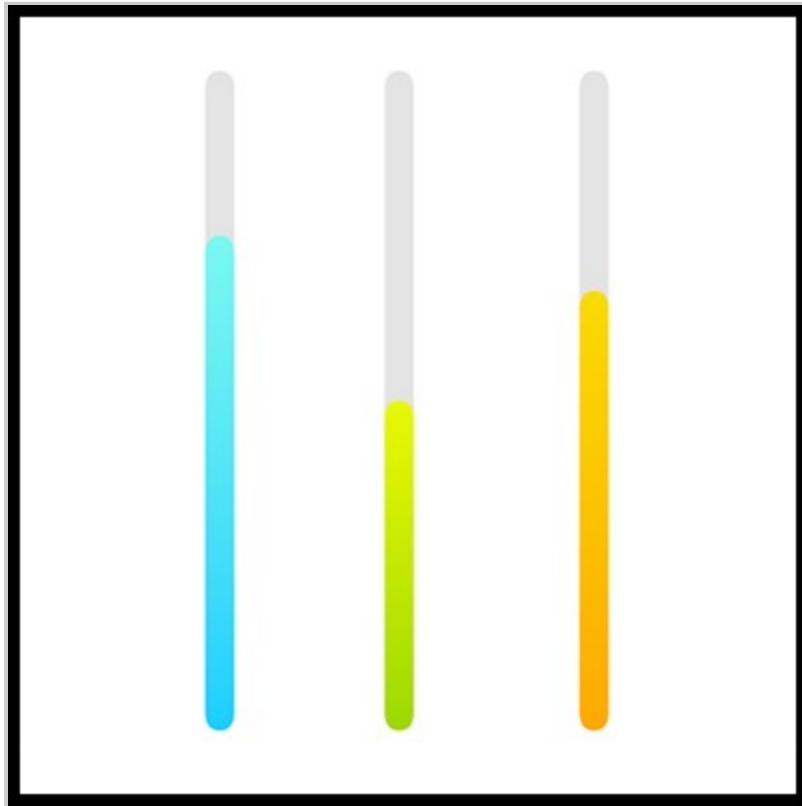
Demonstrates some of the built-in features of the gauge.

[See the code](#)

[Navigate to example](#)

# Gauge

## Vertical Gauge Gallery



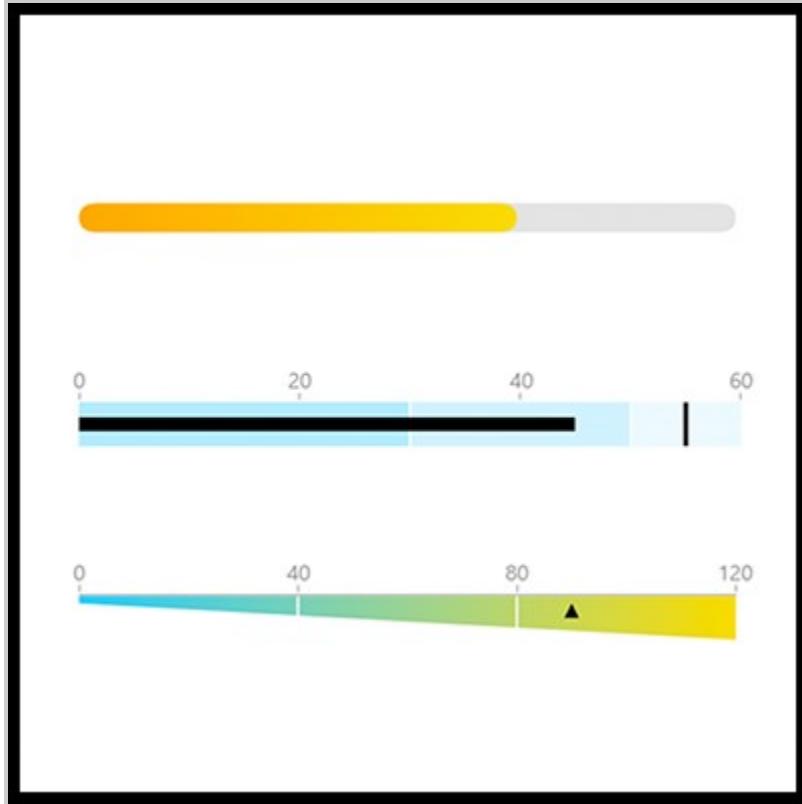
Demonstrates some of the built-in features of the gauge.

[See the code](#)

[Navigate to example](#)

# Gauge

## Horizontal Gauge Gallery



Demonstrates some of the built-in features of the gauge.

[See the code](#)

[Navigate to example](#)

# ImageEditor

ImageEditor is a component that enables you to easily visualize and edit images in different file formats in your mobile application.



Telerik UI for Xamarin ImageEditor is a component that enables you to easily visualize and edit images in different file formats in your mobile application. As an addition to the RadImageEditor control you can use an UI - RadImageEditorToolbar. This UI includes all editing capabilities of the control and it could be easily customized.

[Online documentation](#)

[Navigate to examples](#)

# ImageEditor

## First Look



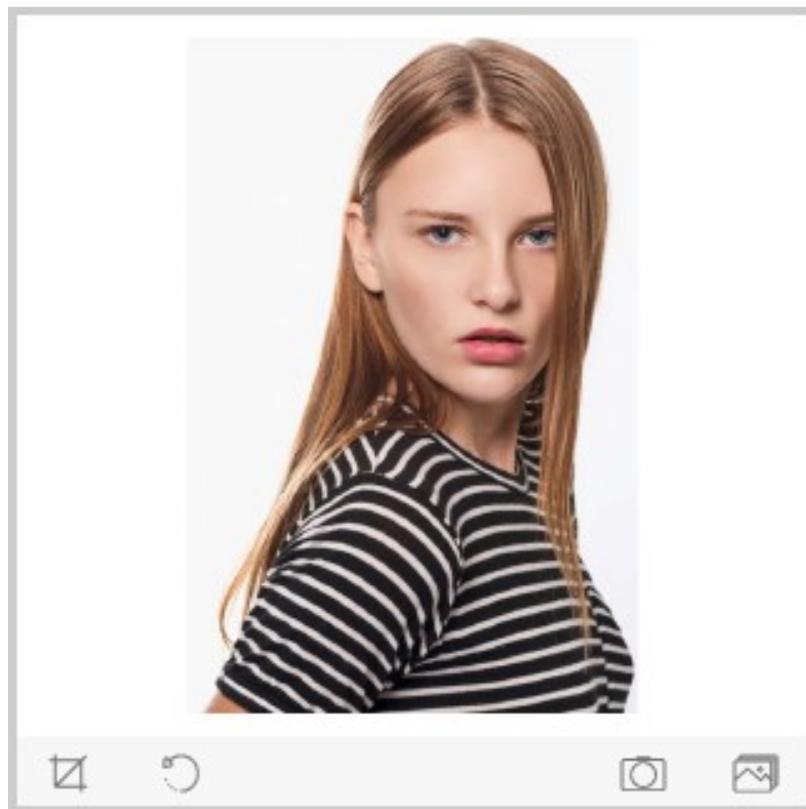
Demonstrates the built-in image editing capabilities of the ImageEditor component.

[See the code](#)

[Navigate to example](#)

# ImageEditor

## Custom Toolbar



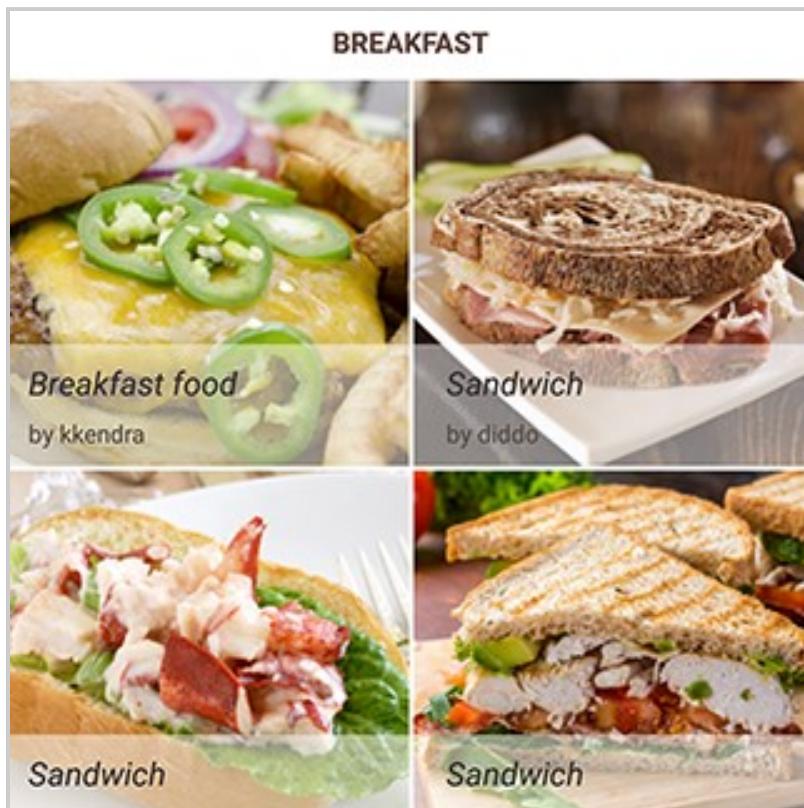
Demonstrates the customization capabilities of the built-in toolbar for the ImageEditor component.

[See the code](#)

[Navigate to example](#)

# ListView

RadListView is a virtualizing list component that provides the most popular features associated with scenarios where a list of items is used.



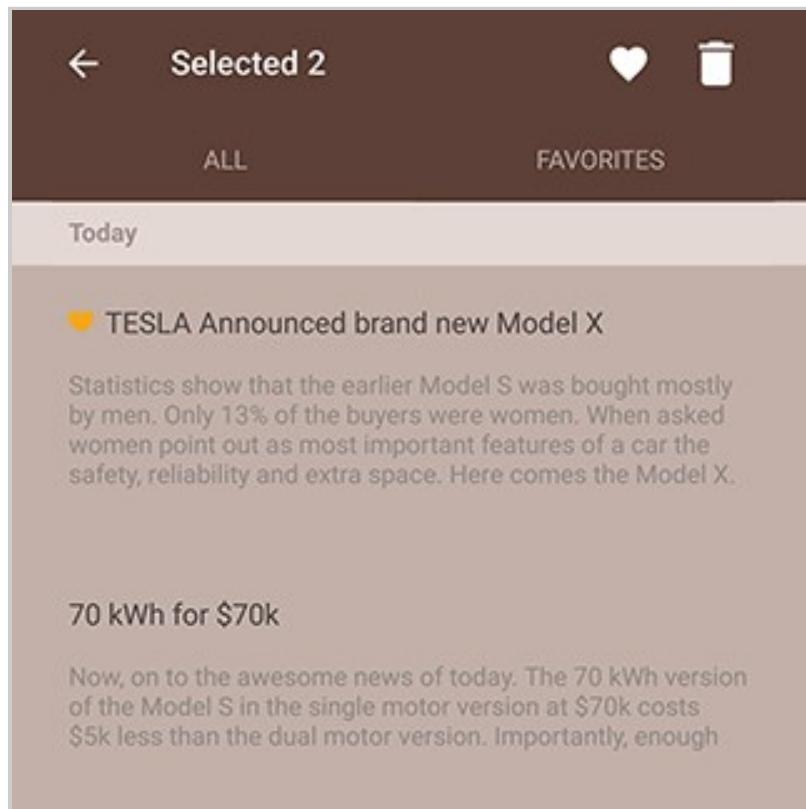
Telerik UI for Xamarin ListView is a virtualized high-performance list component that provides the most popular features associated with scenarios where a list of items is used. All these features are embedded in one control with the idea to save developer time and provide better experience. The control's features include different layouts, pull-to-refresh, grouping, displaying actionable buttons on item swipe, changing the items order by a drag and drop operation.

[Online documentation](#)

[Navigate to examples](#)

# ListView

## Selection



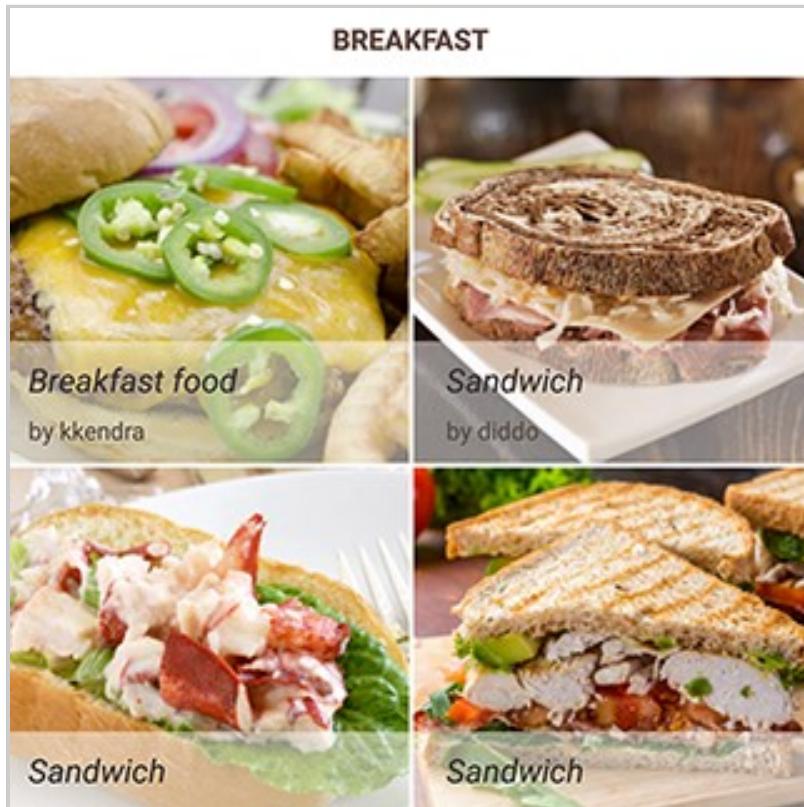
The ListView selection feature allows the end-user to select one or multiple items. The items can also be swiped left or right which reveals actionable buttons which the end-user can tap to execute some tasks.

[See the code](#)

[Navigate to example](#)

# ListView

## Layouts



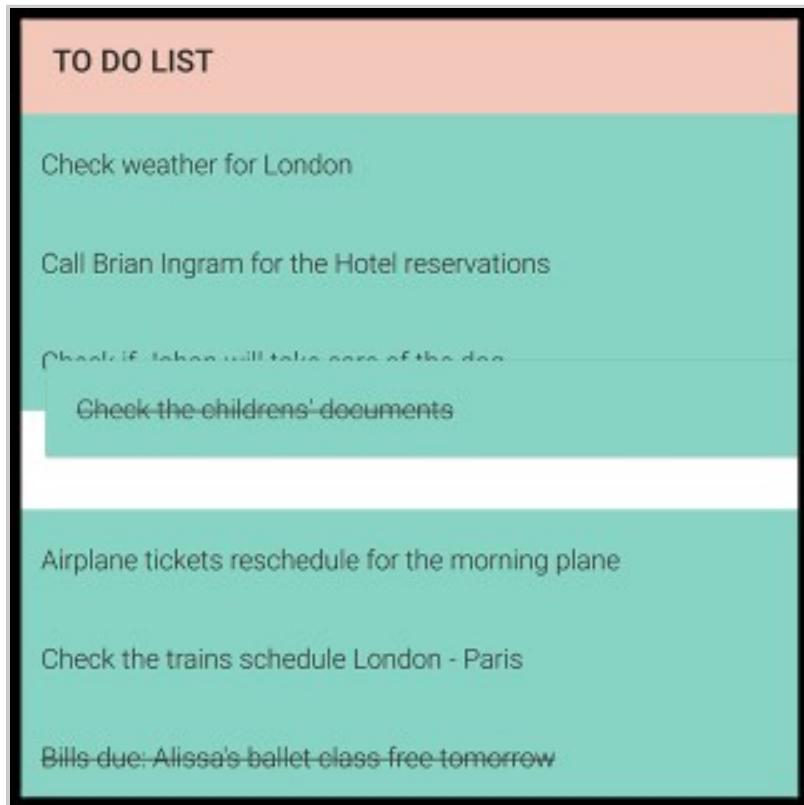
Demonstrates the different layouts that the ListView supports - linear layout where there is one item per row and grid layout where you define how many items there should be per row.

[See the code](#)

[Navigate to example](#)

# ListView

## Reorder



Demonstrates how the end-user can reorder items by a drag and drop operation.

[See the code](#)

[Navigate to example](#)

# ListView

## Theming



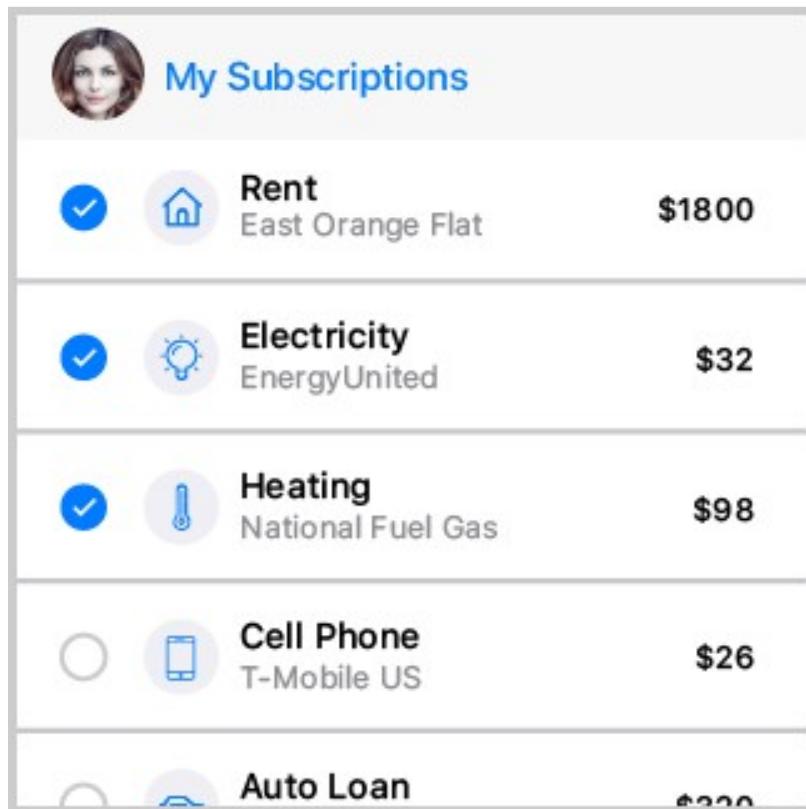
Demonstrates the Blue theme, which is available for styling the control.

[See the code](#)

[Navigate to example](#)

# ListView

## Header & Footer



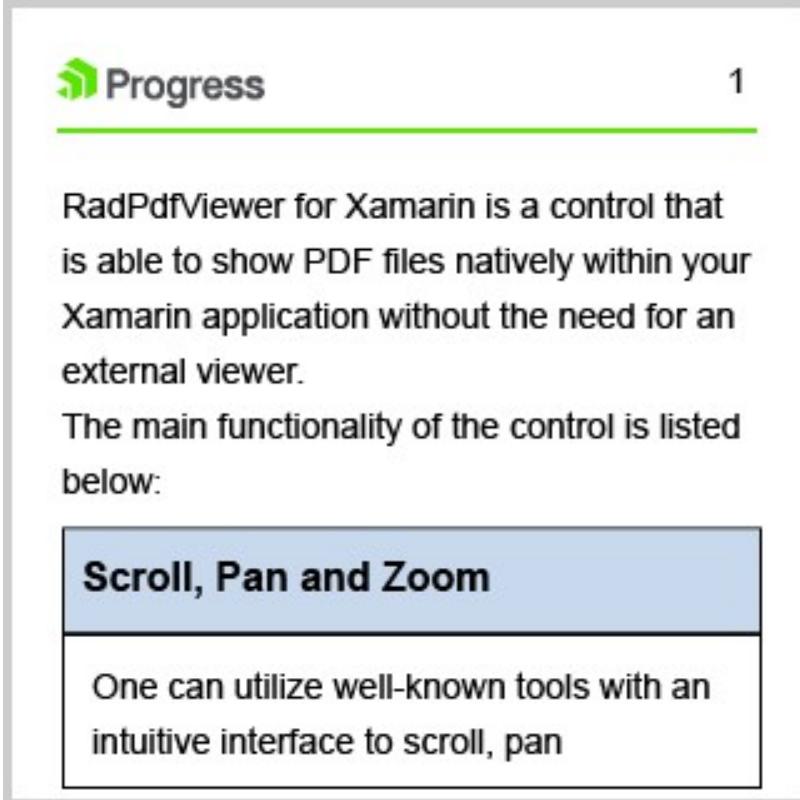
Demonstrates a ListView with header and footer templates. Scroll down to reveal the footer.

[See the code](#)

[Navigate to example](#)

# PdfViewer

RadPdfViewer is a control that can visualize PDF documents straight into your application.



RadPdfViewer for Xamarin is a control that is able to show PDF files natively within your Xamarin application without the need for an external viewer. The main functionality of the control is listed below:

**Scroll, Pan and Zoom**

One can utilize well-known tools with an intuitive interface to scroll, pan

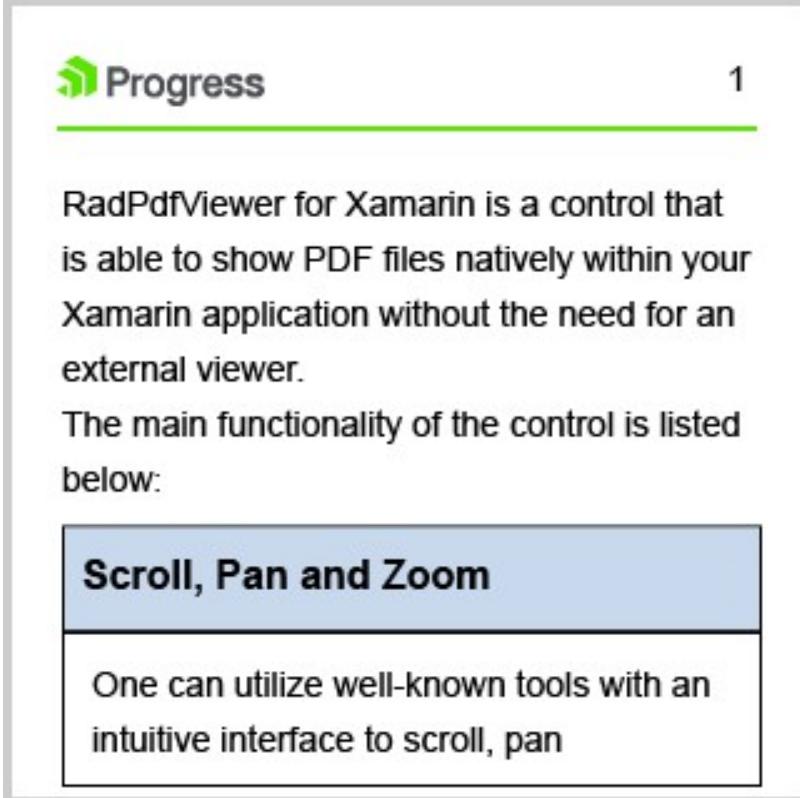
Telerik UI for Xamarin PdfViewer is a control that can visualize PDF documents straight into your application. It has been made more than simple with the exposed commands that can be easily bound to and the full integration with the RadPdfViewerToolbar.

[Online documentation](#)

[Navigate to examples](#)

# PdfViewer

## First Look



RadPdfViewer for Xamarin is a control that is able to show PDF files natively within your Xamarin application without the need for an external viewer.

The main functionality of the control is listed below:

**Scroll, Pan and Zoom**

One can utilize well-known tools with an intuitive interface to scroll, pan

An example that demonstrates how the PdfViewer can easily visualize an embedded into your application PDF document and the flawless integration with the PdfViewerToolbar.

[See the code](#)

[Navigate to example](#)

# PdfViewer

## Customization

Progress Telerik UI for Xamarin

## PdfViewer

---

### Overview

The PdfViewer control in Telerik UI for Xamarin enables the end-users of your application to open and view PDF documents straight through their mobile app without having to install any third-party software or switching back and forth between the two.

### Support for Various PDF Document Elements

The PdfViewer can display PDF documents, which contain images, shapes, colors, ordered and bullet lists among other element.

### Single Page and Continuous Scrolling Support

End users can scroll PDF documents in the viewer with a fluent experience utilizing the two scrolling modes – Continuous scroll for scrolling all pages or SinglePage scrolling for scrolling one page at a time.

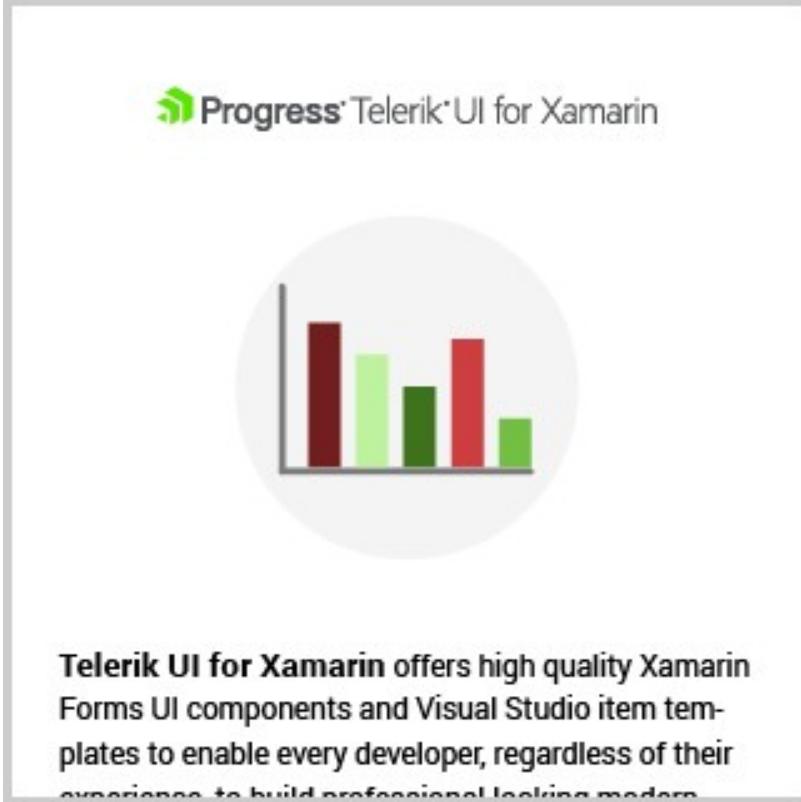
This example demonstrates how the look & feel of the PdfViewer selection and toolbar can be easily customized by changing their color, alignment and items position.

[See the code](#)

[Navigate to example](#)

# PdfViewer

## Large file



The screenshot shows a large PDF document page. At the top left is the Progress Telerik UI for Xamarin logo. In the center is a circular placeholder icon containing a bar chart with five bars of varying heights and colors (dark red, light green, dark green, red, and light green). Below the placeholder is a text box containing the following text:

**Telerik UI for Xamarin offers high quality Xamarin Forms UI components and Visual Studio item templates to enable every developer, regardless of their experience, to build professional looking modern**

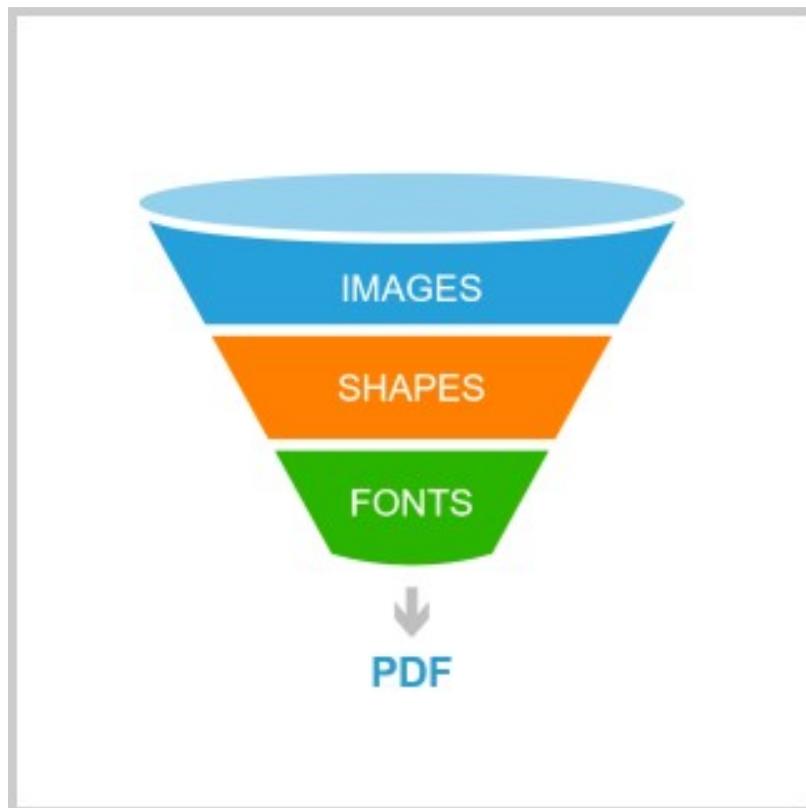
A demonstration of the read-on-demand feature that allows to quickly open large files. This is done by not loading the whole document initially, but a page is loaded on demand, only when it enters the viewable area.

[See the code](#)

[Navigate to example](#)

# PdfProcessing

RadPdfProcessing is a processing library that allows creation, import and export of PDF documents.



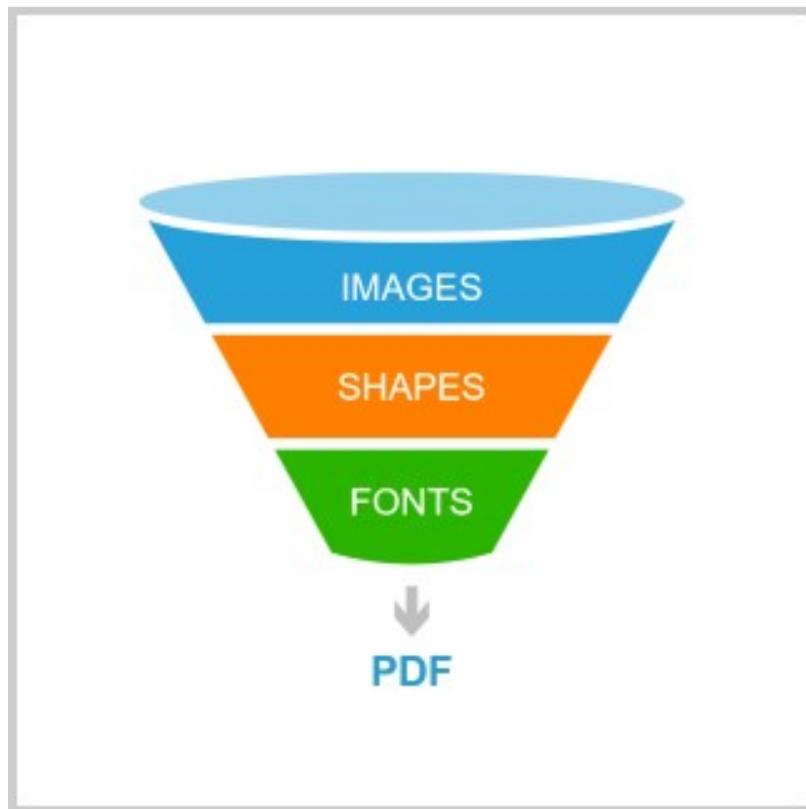
The API of RadPdfProcessing contains two different editors, enabling you to choose between editing in a flow-like manner, or using the much more powerful and flexible fixed document structure and drawing on the page.

[Online documentation](#)

[Navigate to examples](#)

# PdfProcessing

## First Look



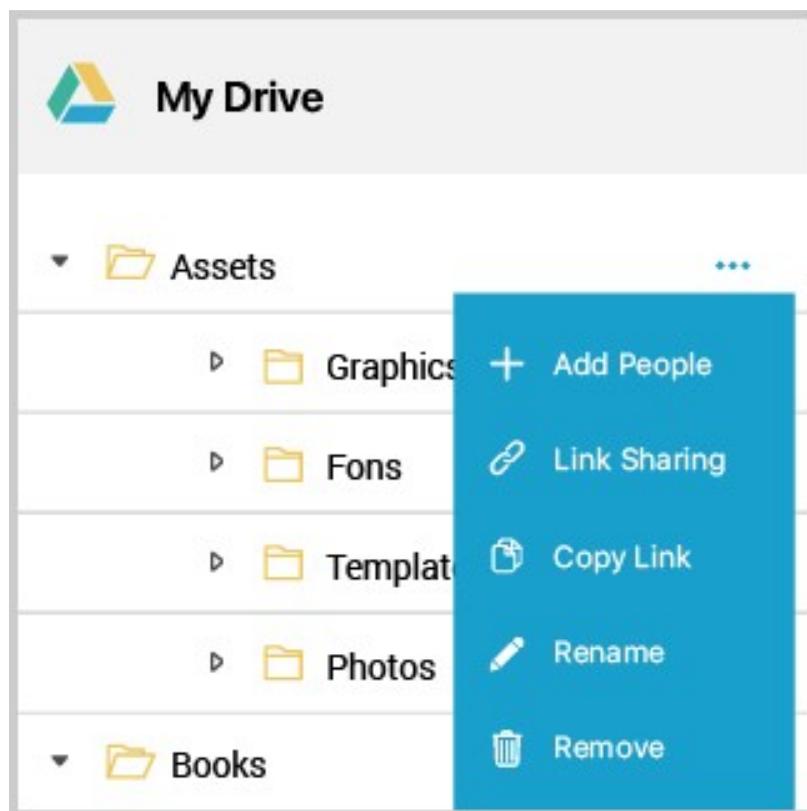
Telerik RadPdfProcessing provides easy to use API which allows code-only generation of PDF documents. This example demonstrates how to create a document containing images, text and shapes and export it to PDF.

[See the code](#)

[Navigate to example](#)

# Popup

With RadPopup you could easily add modal popups to your application in order to draw attention to important information or receive user input.



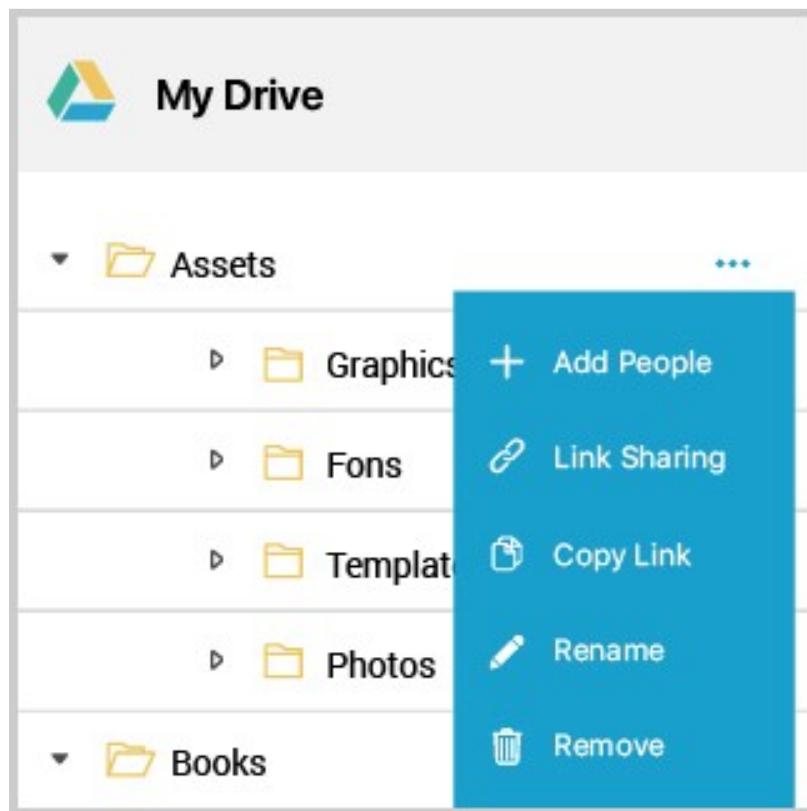
Telerik UI for Xamarin Popup lets you display content of your choice on top of existing view. You could define whether the Popup will be modal or not as well as control its position through various placement settings.

[Online documentation](#)

[Navigate to examples](#)

# Popup

## Context Menu



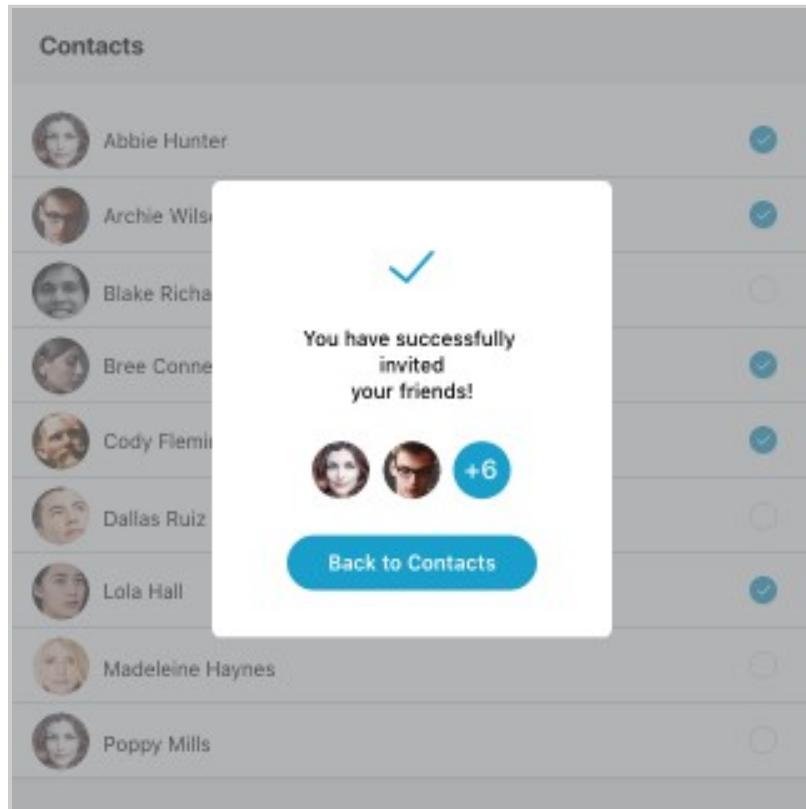
This example demonstrates how you could use RadPopup control as a context menu - you can host any content inside the Popup and easily position it according to the target element.

[See the code](#)

[Navigate to example](#)

# Popup

## Modal



This example demonstrates how you could use RadPopup as a modal, so that the UI behind it gets inactive and cannot be used until the Popup is closed.

[See the code](#)

[Navigate to example](#)

# Popup

## Configuration



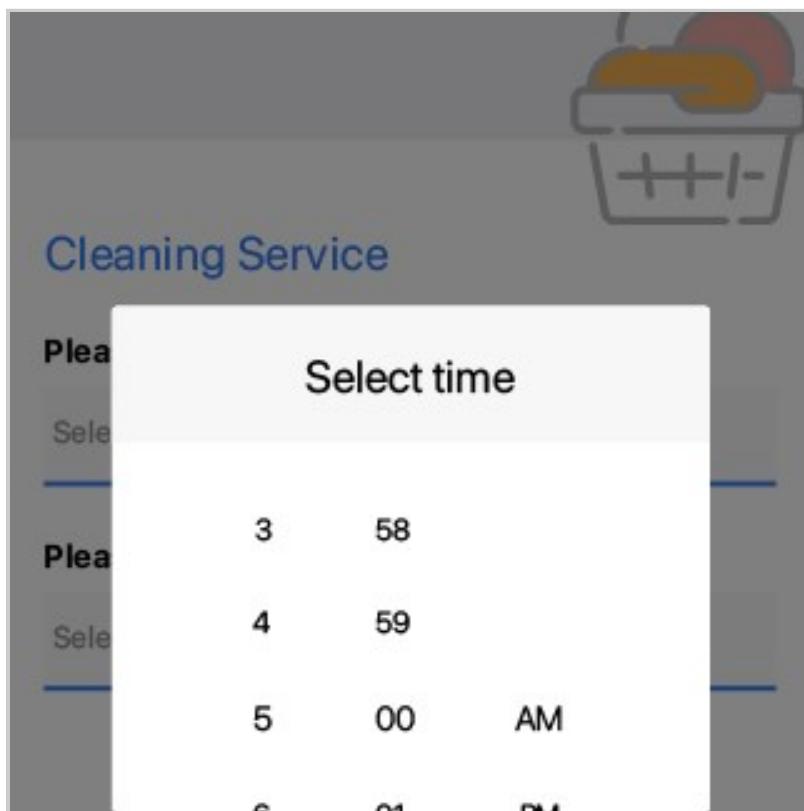
This example presents various configuration options that the Popup control provides.

[See the code](#)

[Navigate to example](#)

# Time Picker

RadTimePicker provides a convenient way for the app users to select a time.



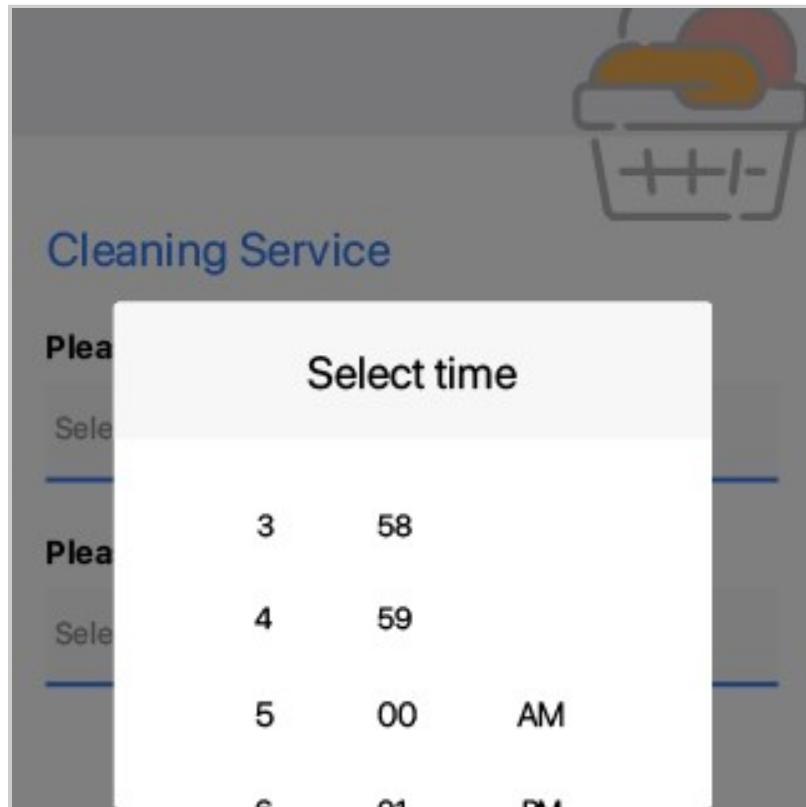
Telerik Time Picker for Xamarin provides a convenient way for the app users to select a time. Its items are visualized inside a popup. Time Picker control has a number of features which allows you to set a time range, time format and fully customize the dialog appearance such as its header and footer.

[Online documentation](#)

[Navigate to examples](#)

# Time Picker

## First Look



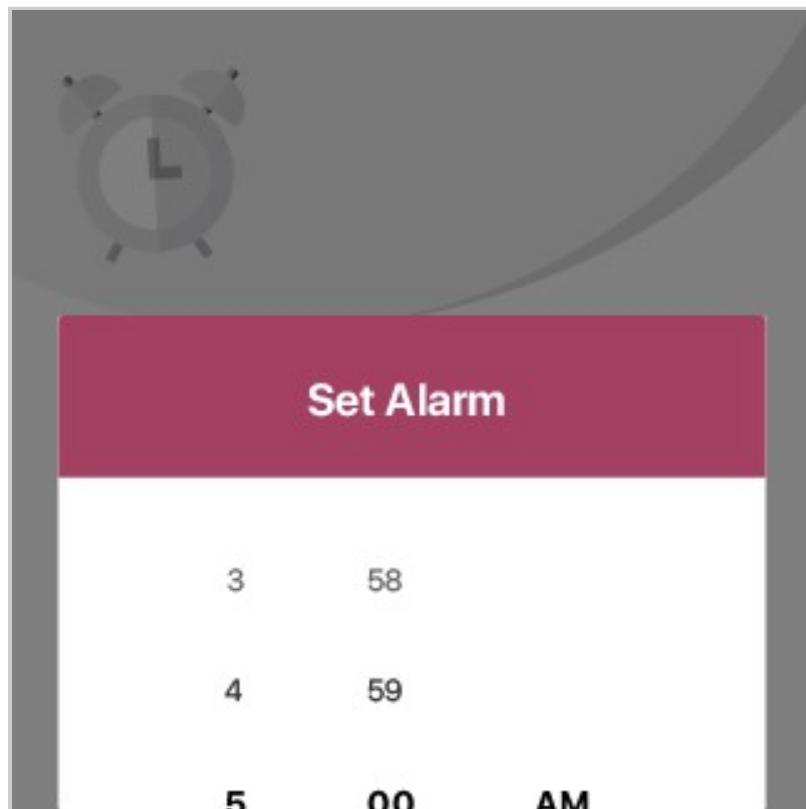
RadTimePicker provides the users an easy way to pick a time. The control allows you to define minimum and maximum time. This example demonstrates how you can setup the TimePicker control, so the app users can book a time for cleaning service.

[See the code](#)

[Navigate to example](#)

# Time Picker

## Customization



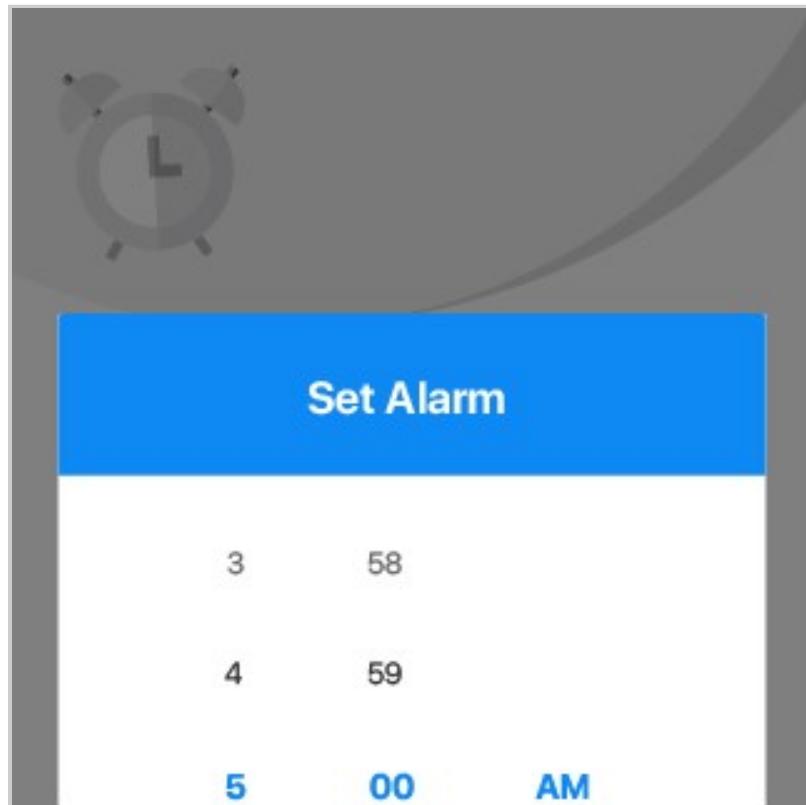
This example demonstrates how to customize the visual appearance of RadTimePicker using the control's flexible styling API.

[See the code](#)

[Navigate to example](#)

# Time Picker

## Configuration



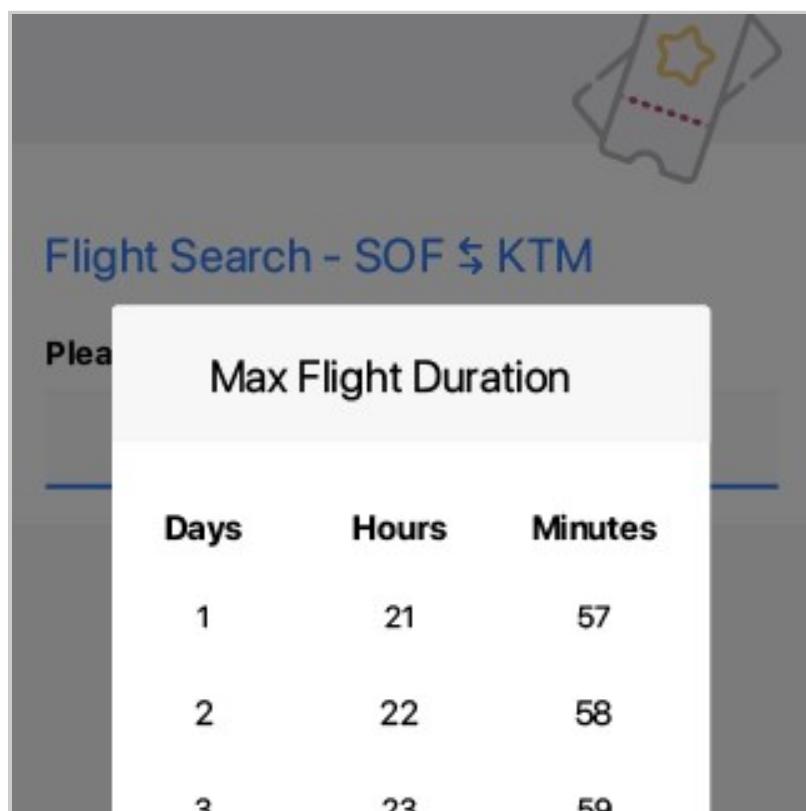
This example presents various configuration options that the RadTimePicker provides.

[See the code](#)

[Navigate to example](#)

# TimeSpan Picker

Rad TimeSpan Picker provides an easy way to select a time duration.



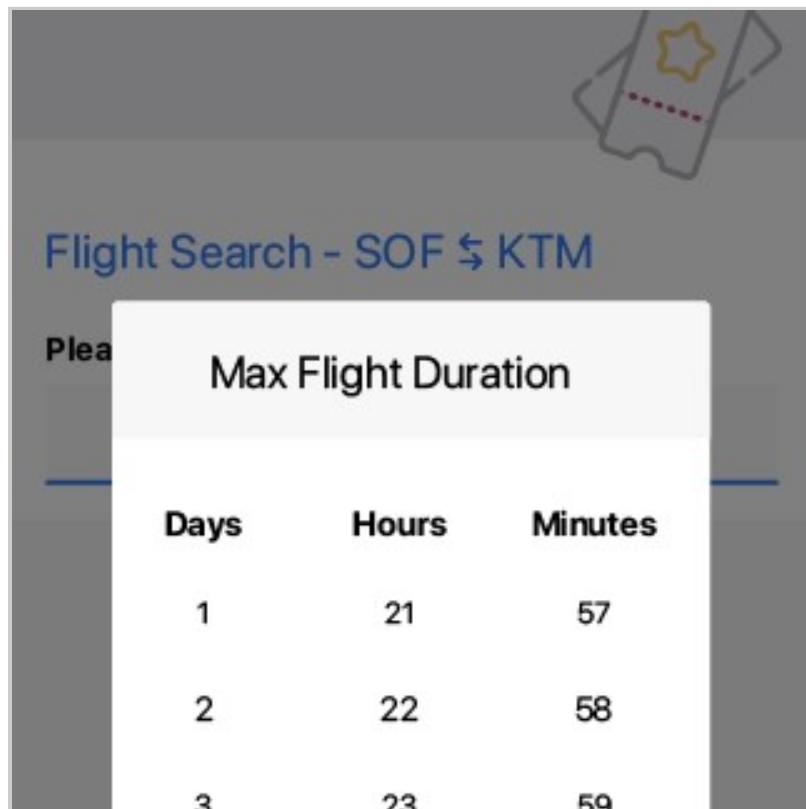
Telerik TimeSpan Picker for Xamarin provides an easy way to select a time duration. TimeSpan control represents a time interval and allows you to set a time range between two times measured in numbers of days, hours, minutes and seconds.

[Online documentation](#)

[Navigate to examples](#)

# TimeSpan Picker

## First Look



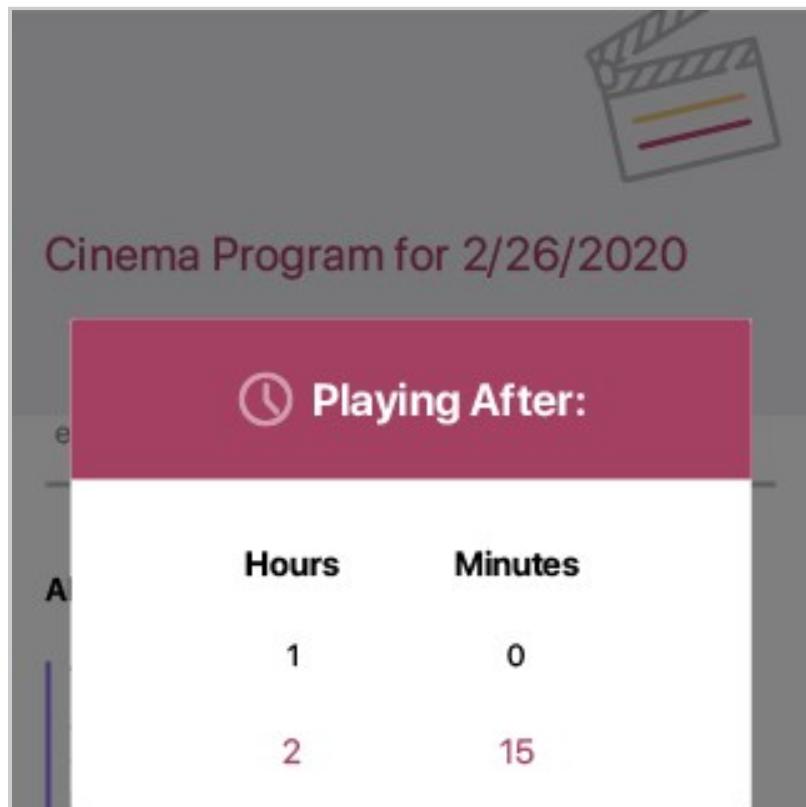
RadTimeSpanPicker provides the users an easy way to select a time interval. The control allows you to define minimum and maximum time. This example demonstrates how you can setup the TimeSpanPicker control, so the app users can select a maximum flight duration.

[See the code](#)

[Navigate to example](#)

# TimeSpan Picker

## Customization



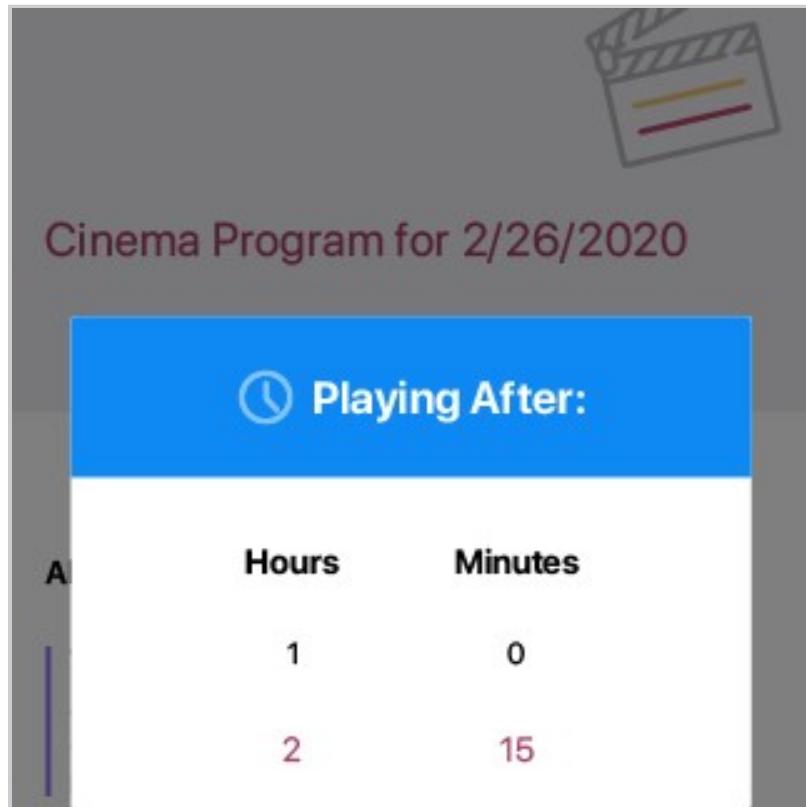
This example demonstrates how the TimeSpanPicker visual appearance could be customized using the flexible styling API and control's templates.

[See the code](#)

[Navigate to example](#)

# TimeSpan Picker

## Configuration



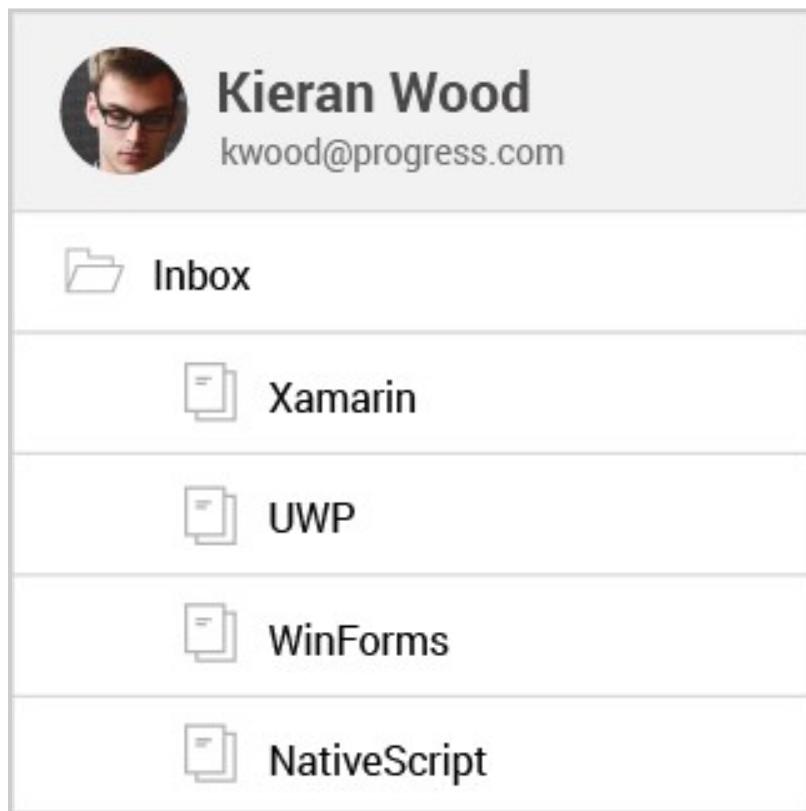
This example presents various configuration options that the RadTimeSpanPicker provides.

[See the code](#)

[Navigate to example](#)

## TreeView

RadTreeView facilitates display, management, and navigation of hierarchical data structures. The product offers many advanced features like commands mechanism, data binding, checkbox support.



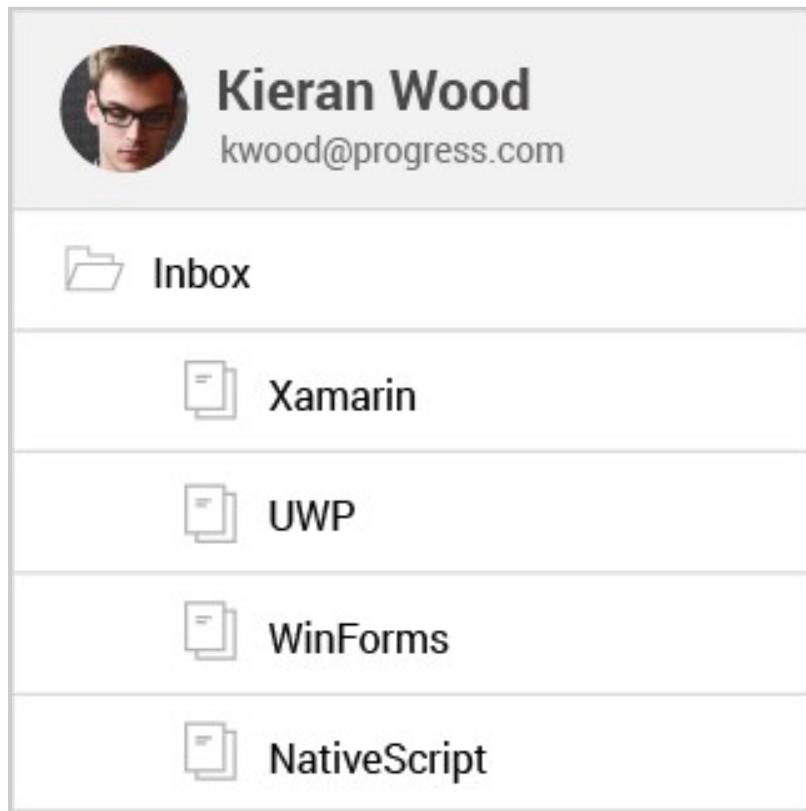
RadTreeView facilitates display, management, and navigation of hierarchical data structures. The product offers many advanced features like commands mechanism, data binding, checkbox support. RadTreeView allows you to achieve complex behavior and nicely blends into the interface of your application.

[Online documentation](#)

[Navigate to examples](#)

# TreeView

## First Look



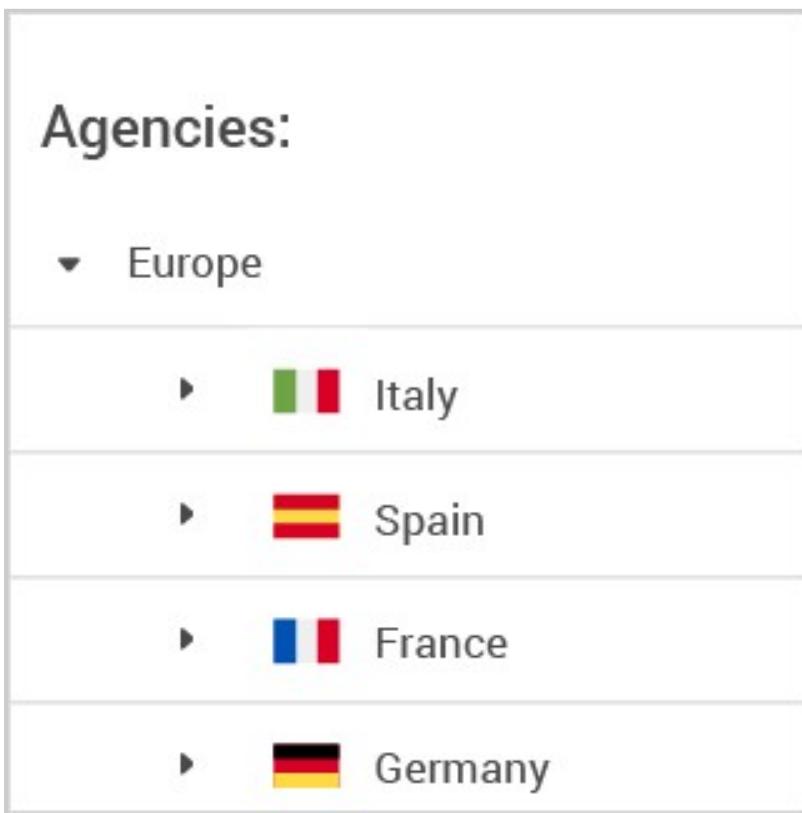
Demonstrates how the RadTreeView component can be used within your application.

[See the code](#)

[Navigate to example](#)

# TreeView

## Customization



Agencies:

- ▼ Europe
  - ▶  Italy
  - ▶  Spain
  - ▶  France
  - ▶  Germany

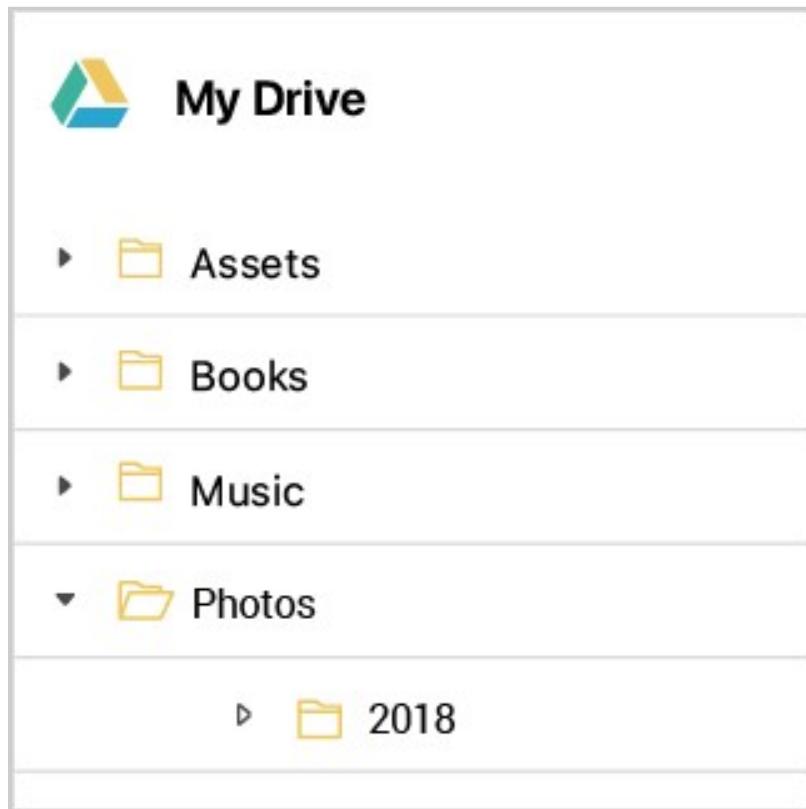
This example demonstrates how to modify the look & feel of RadTreeView items by defining separate ItemTemplates for different item types.

[See the code](#)

[Navigate to example](#)

# TreeView

## Load On Demand



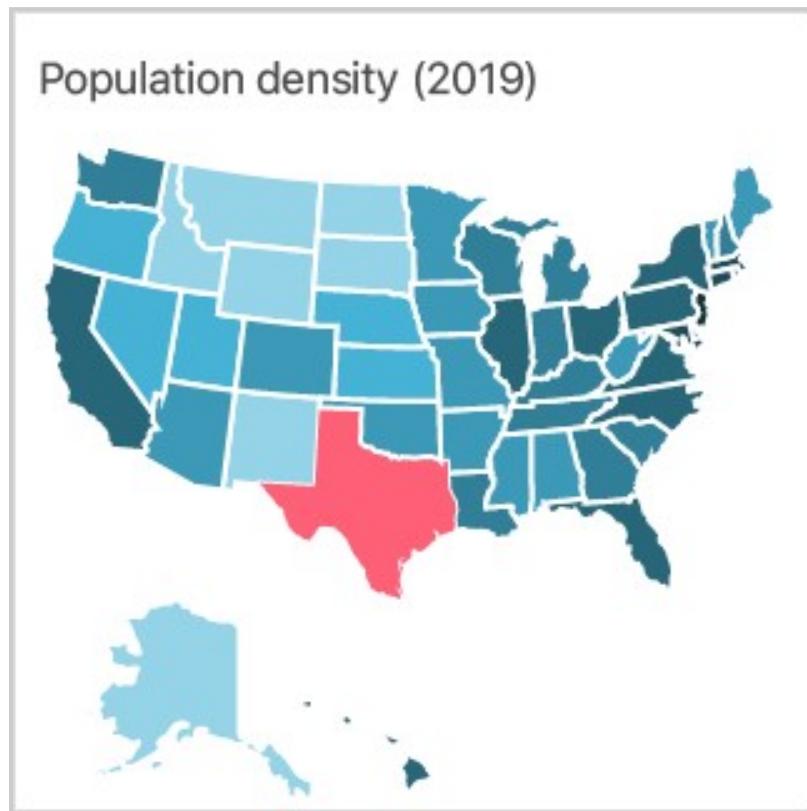
Demonstrates how you could utilize Load On Demand feature in order to delay the population of RadTreeView and load subitems only when they're requested, thus improving the performance of your app.

[See the code](#)

[Navigate to example](#)

# Map

RadMap is a control that can visualize stunning map overlays through ESRI Shapefiles straight into your application.



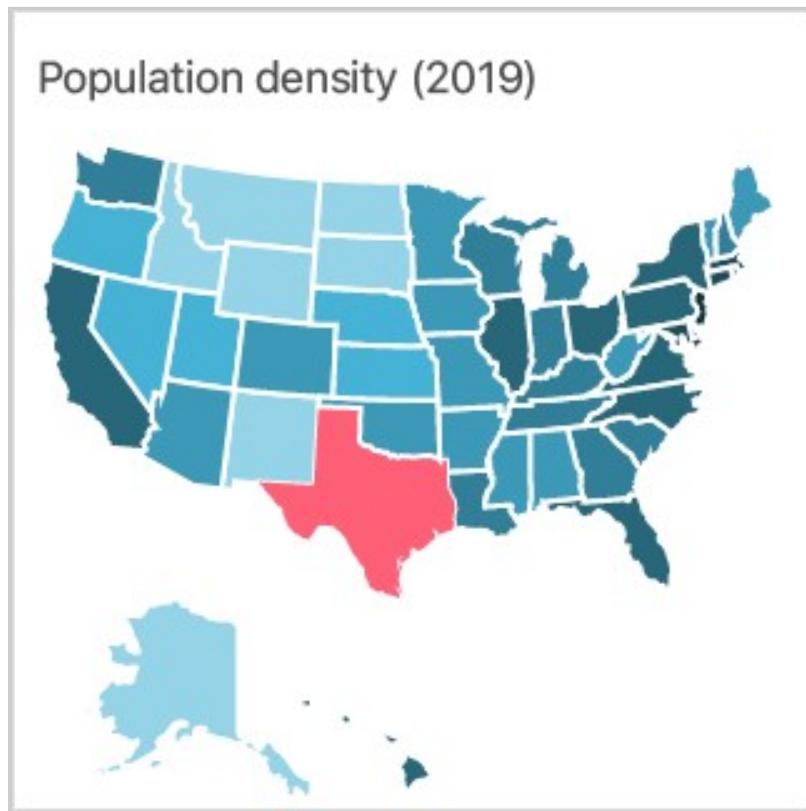
Telerik UI for Xamarin Map lets you display stunning map overlays using ESRI Shapefiles. The control provides support for labels, multiple, single selection and powerful customization APIs.

[Online documentation](#)

[Navigate to examples](#)

# Map

## First Look



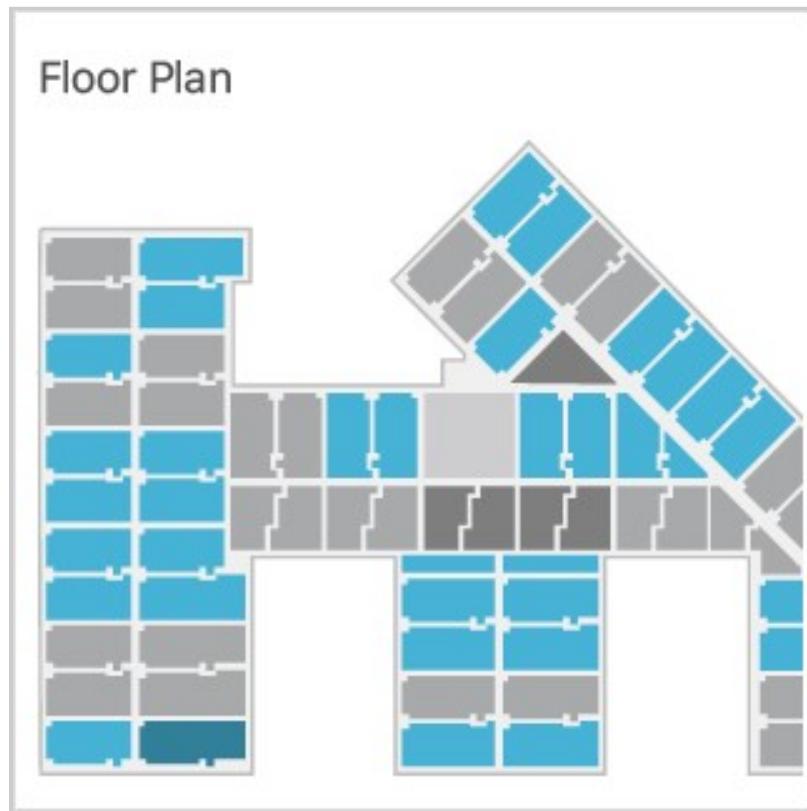
An example that demonstrates how the Map can easily visualize an embedded into your application ESRI Shapefile.

[See the code](#)

[Navigate to example](#)

# Map

**Hotel Floor Plan**



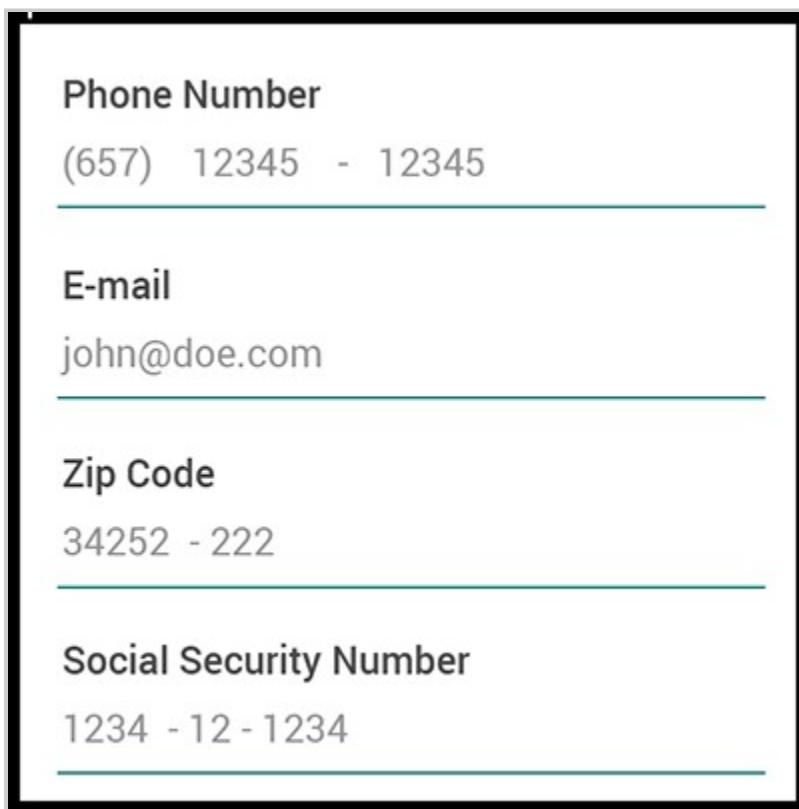
This example demonstrates the usage of custom shapefile to represent hotel floor plan and allows you to interact with it.

[See the code](#)

[Navigate to example](#)

# MaskedInput

RadMaskedInput allows to control the data input within your Xamarin.Forms app. It enhances the functionality of the Entry control by adding input validation and custom masks.



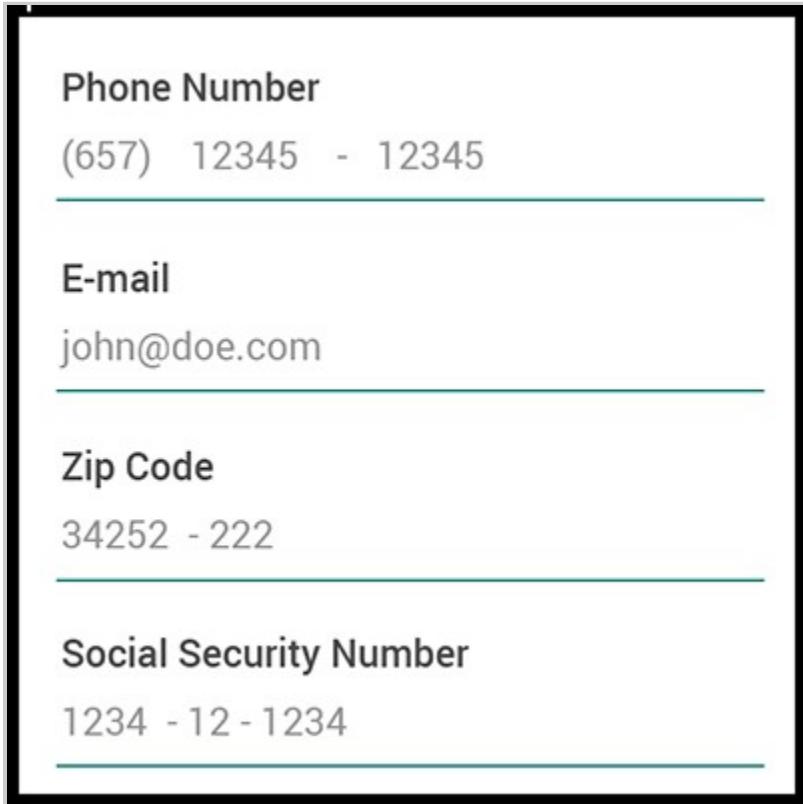
Telerik UI for Xamarin MaskedInput allows you to control data input in your applications. They enhance the functionality of an Entry by adding input validation and masks, such as E-mail, IP Address, SSN, phone number, digits, letters and symbols.

[Online documentation](#)

[Navigate to examples](#)

# MaskedInput

## First Look



The screenshot displays a user interface with four input fields, each enclosed in a black-bordered box:

- Phone Number**: The value is (657) 12345 - 12345.
- E-mail**: The value is john@doe.com.
- Zip Code**: The value is 34252 - 222.
- Social Security Number**: The value is 1234 - 12 - 1234.

Demonstrates the RadMaskedInput.

[See the code](#)

[Navigate to example](#)

# MaskedInput

## Theming

**Phone Number**

(657) 12345 - 12345

---

**E-mail**

john@doe.com

---

**Zip Code**

34252 - 222

---

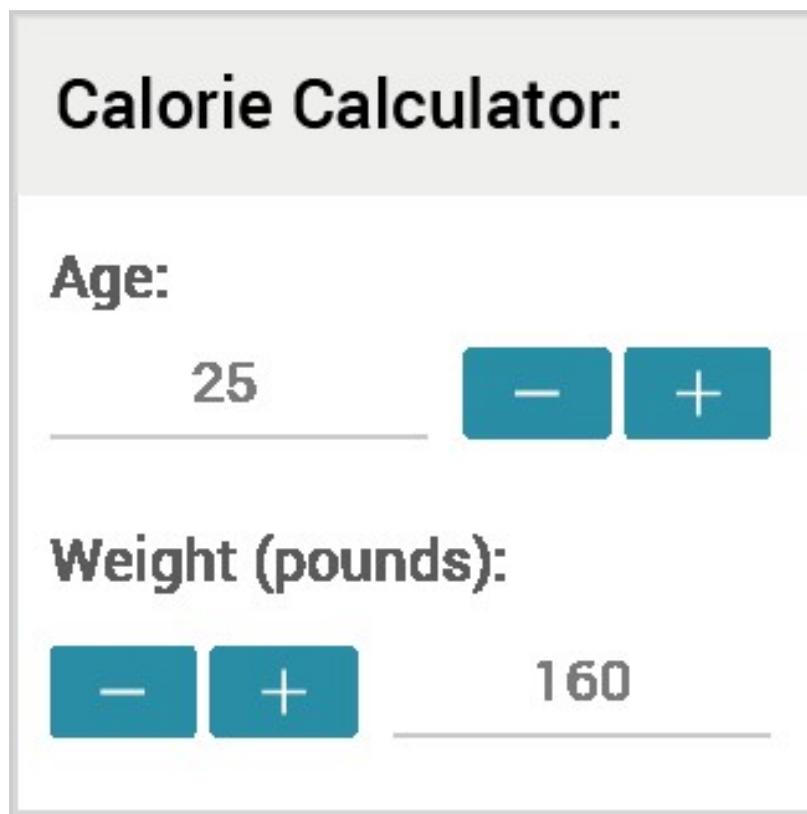
Demonstrates the theming of the RadMaskedInput.

[See the code](#)

[Navigate to example](#)

## NumericUpDown

RadNumericInput is a customizable input control for numeric data. It allows the user to edit a number using the decrease and increase buttons or directly enter it in the input.



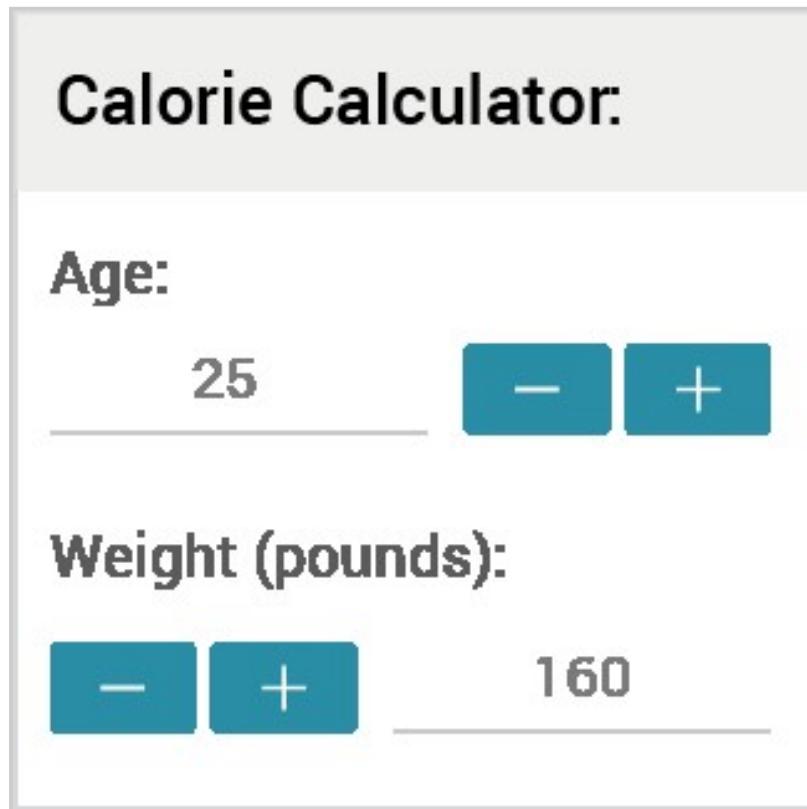
Telerik UI for Xamarin NumericUpDown is a highly customizable input control for numeric data. It allows the user to set/edit a number using the decrease and increase buttons or directly enter it in the input field. RadNumericInput also provides consistent look with the rest of the controls from the Telerik UI for Xamarin suite.

[Online documentation](#)

[Navigate to examples](#)

# NumericUpDown

## First Look



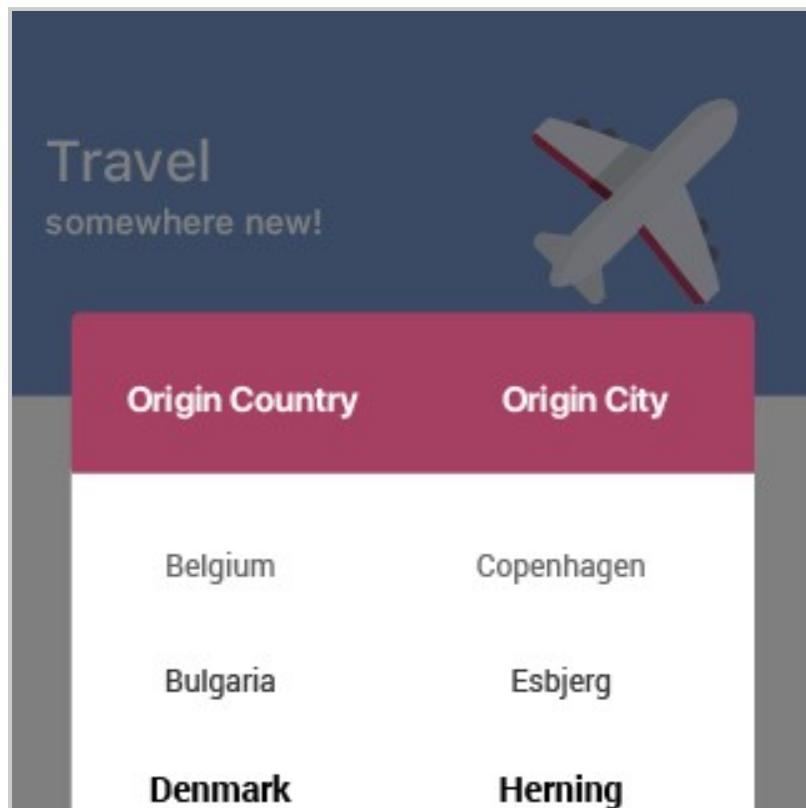
Demonstrates how you could use RadNumericInput to enable the end users easily enter numeric data and to achieve slick design across all platforms.

[See the code](#)

[Navigate to example](#)

# Templated Picker

RadTemplatedPicker is a fully customizable picker control that covers a wide range of item selection scenarios for mobile applications.



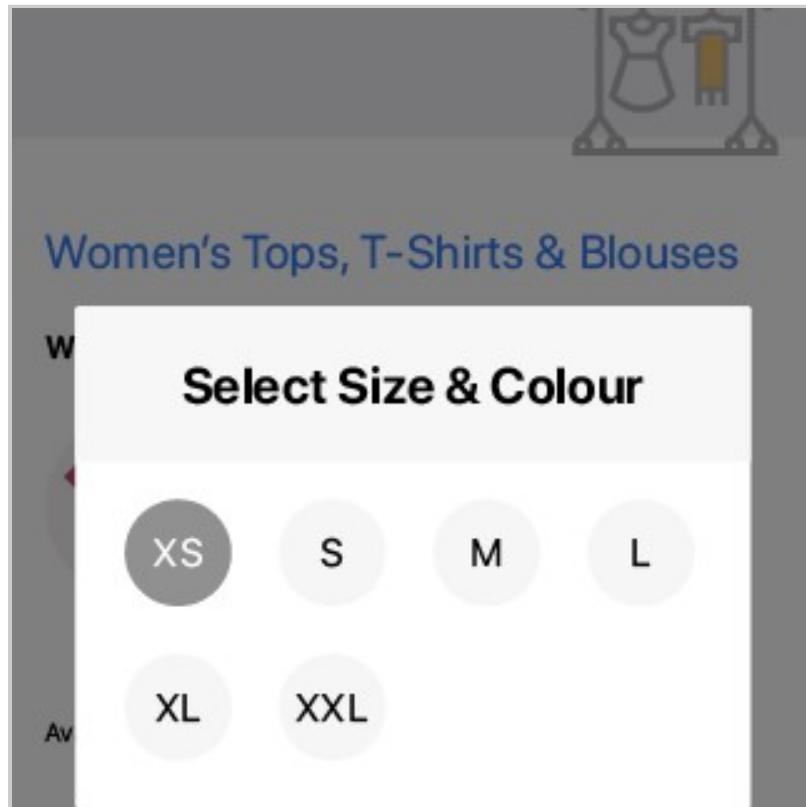
Telerik UI for Xamarin Templatized Picker is a control designed to support the need for a tailored item selection based on various potential end-user scenarios (e.g. pick a country and city for a travel app). The Templatized Picker enables you to define your custom template for an item selection and customize it to ensure user experience consistency.

[Online documentation](#)

[Navigate to examples](#)

# Templated Picker

## First Look



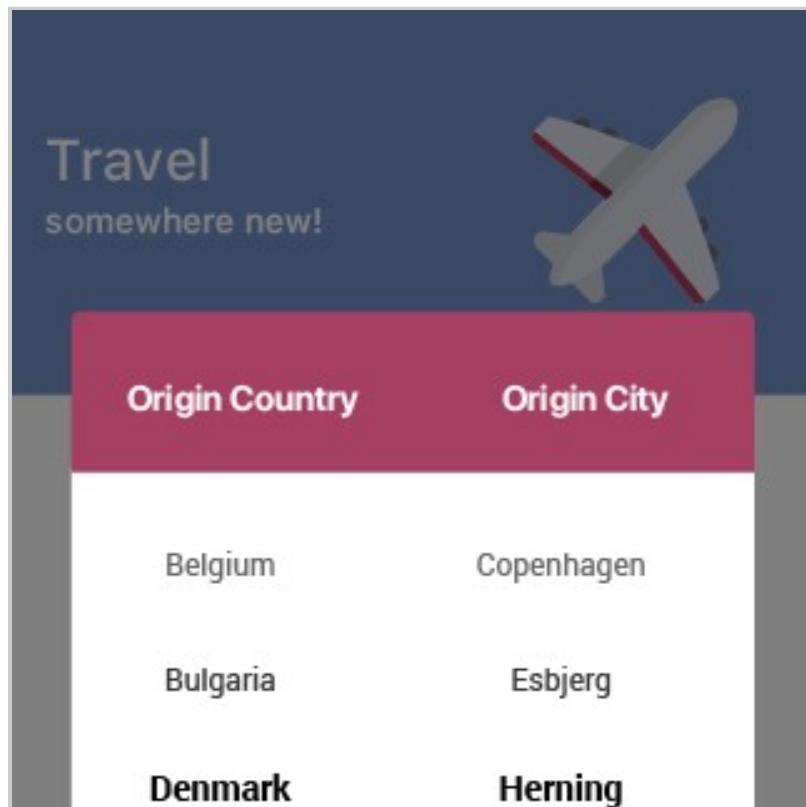
Demonstrates how you can utilize RadTemplatedPicker in a sample scenario for selecting t-shirt color and size. Through the custom selector template you can place any content inside the picker popup based on the scenario at hand.

[See the code](#)

[Navigate to example](#)

# Templated Picker

## Customization



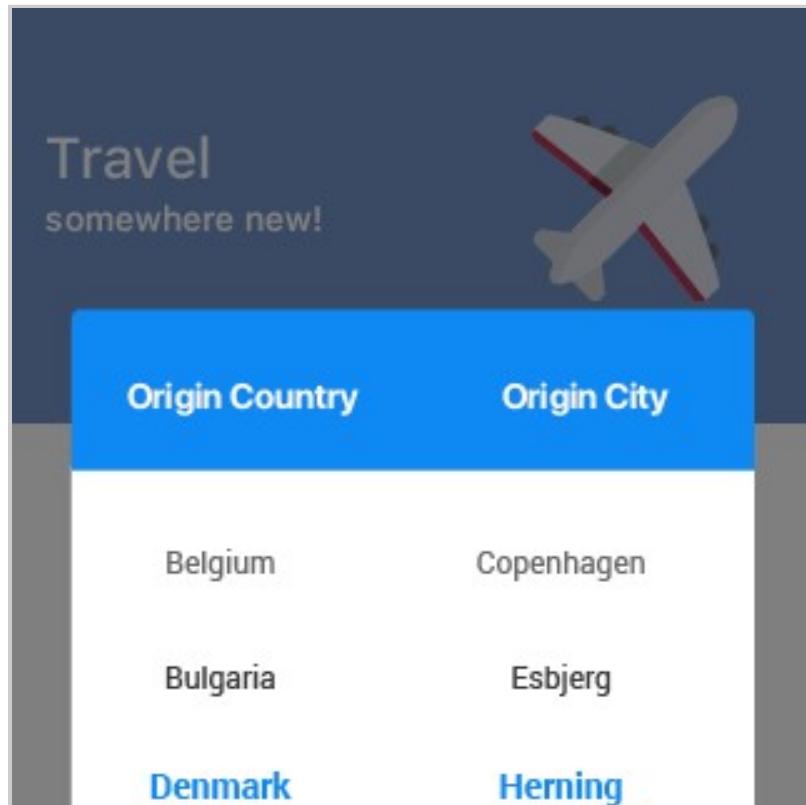
Demonstrates how you can utilize RadTemplatedPicker in a sample scenario for searching flights. Through the custom selector template you can place any content inside the picker popup based on the scenario at hand.

[See the code](#)

[Navigate to example](#)

# Templated Picker

## Configuration



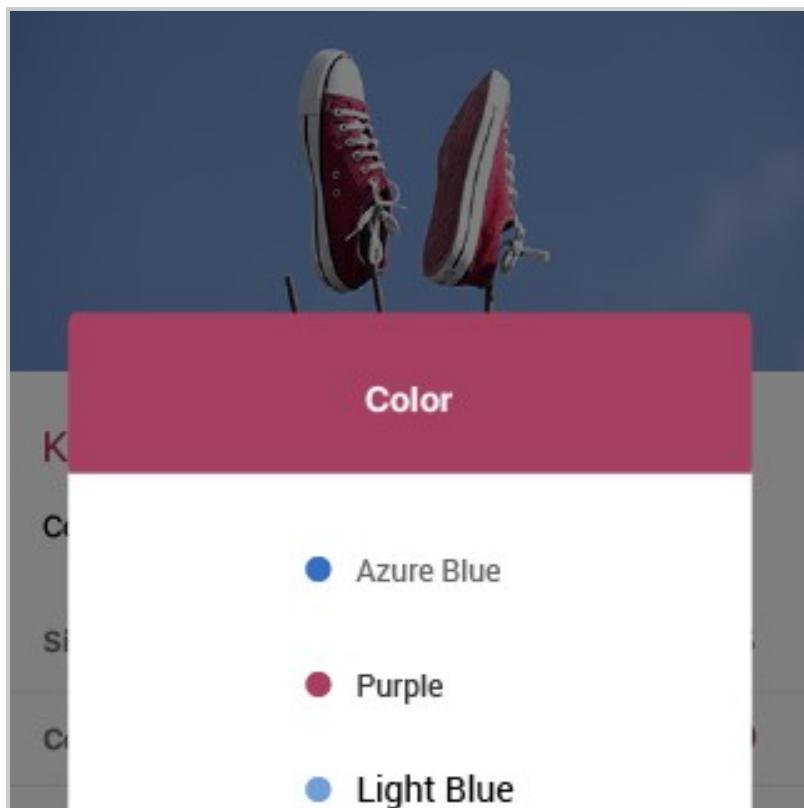
This example presents various configuration options you can apply to the RadTemplatedPicker control.

[See the code](#)

[Navigate to example](#)

## List Picker

RadListPicker allows you to select an item from a list, visualized inside a popup. The control can be setup to loop its items infinitely while scrolling.



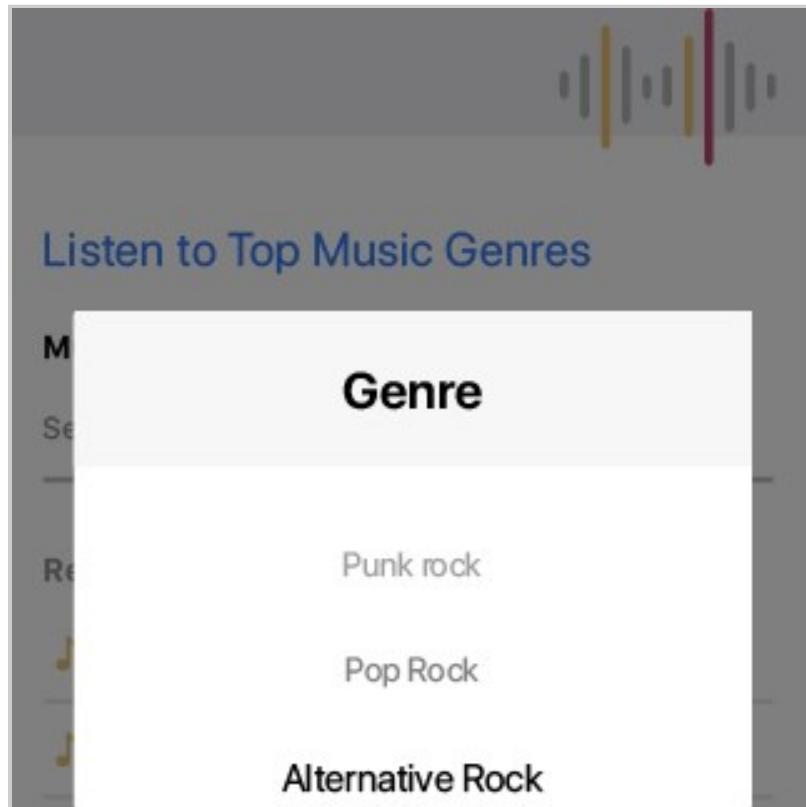
Telerik UI for Xamarin List Picker provides an elegant UI experience for end users who need to pick an item from a pre-defined list. The List Picker component loops its listed items infinitely while scrolling and is flexible to various customizations to ensure consistent user experience across the mobile applications. Developers can easily customize the dialog appearance, style the listed and selected items and define a placeholder text when an item is not selected.

[Online documentation](#)

[Navigate to examples](#)

# List Picker

## First Look



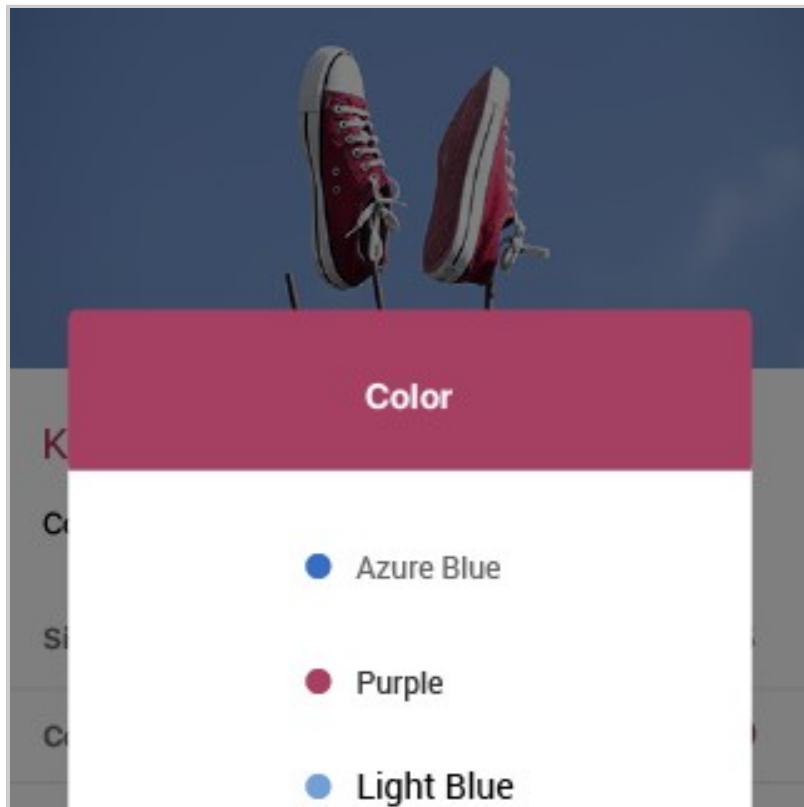
RadListPicker can be useful in various scenarios when you'd like to give the app users a convenient way to choose a value from a list of items. This example presents a sample scenario for selecting a music genre.

[See the code](#)

[Navigate to example](#)

# List Picker

## Customization



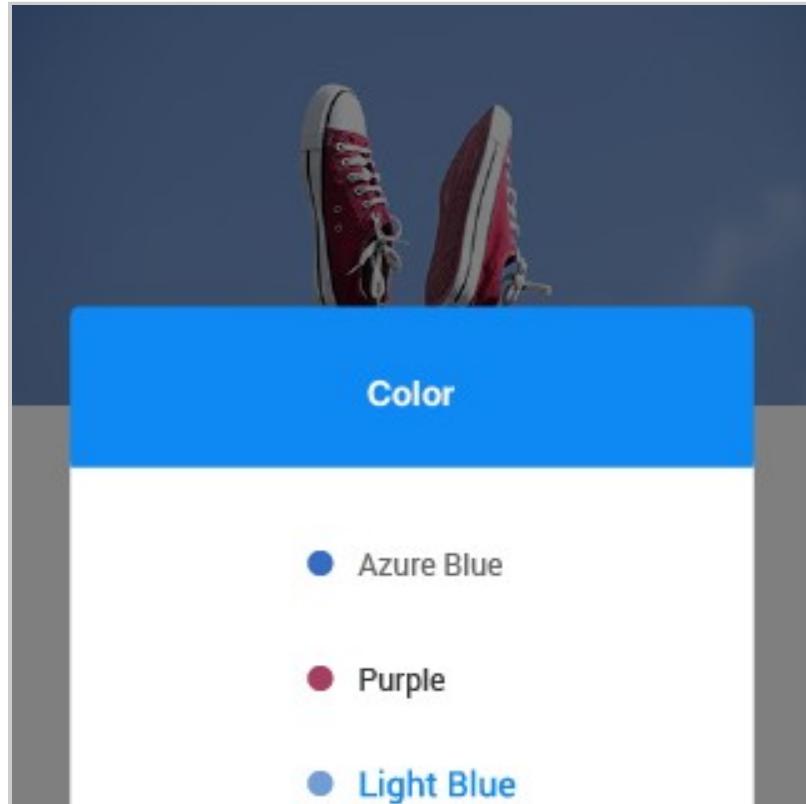
RadListPicker allows you to customize its look in a way you want using the flexible styling API. This example presents a sample scenario for selecting a size and a color.

[See the code](#)

[Navigate to example](#)

# List Picker

## Configuration



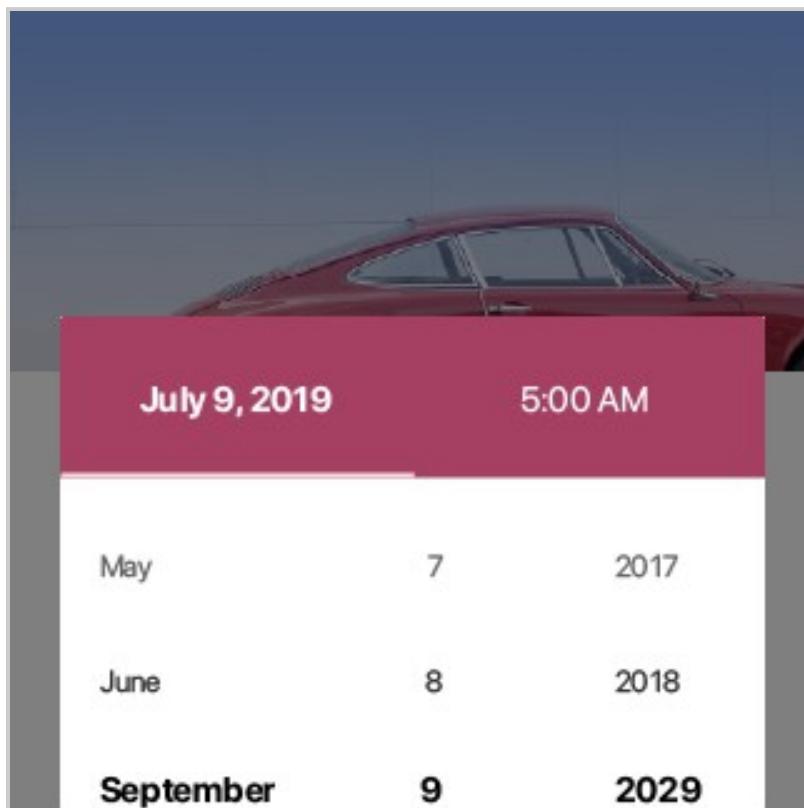
This example presents various configuration options you can apply to the RadListPicker control.

[See the code](#)

[Navigate to example](#)

# DateTime Picker

RadDateTimePicker provides an easy way to pick a date, time or date and time depending on the used format string.



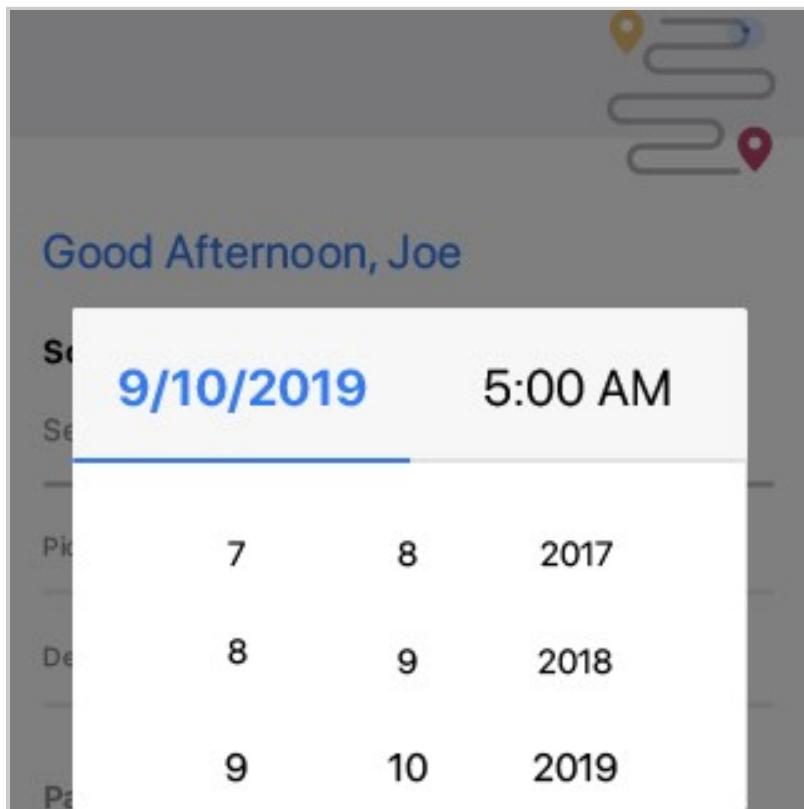
Telerik UI for Xamarin DateTimePicker enables users to easily pick a date, time or both from a pre-defined set of values. The DateTimePicker control has several features which allow you to set a date range, date and time format and fully customize the dialog appearance.

[Online documentation](#)

[Navigate to examples](#)

# DateTime Picker

## First Look



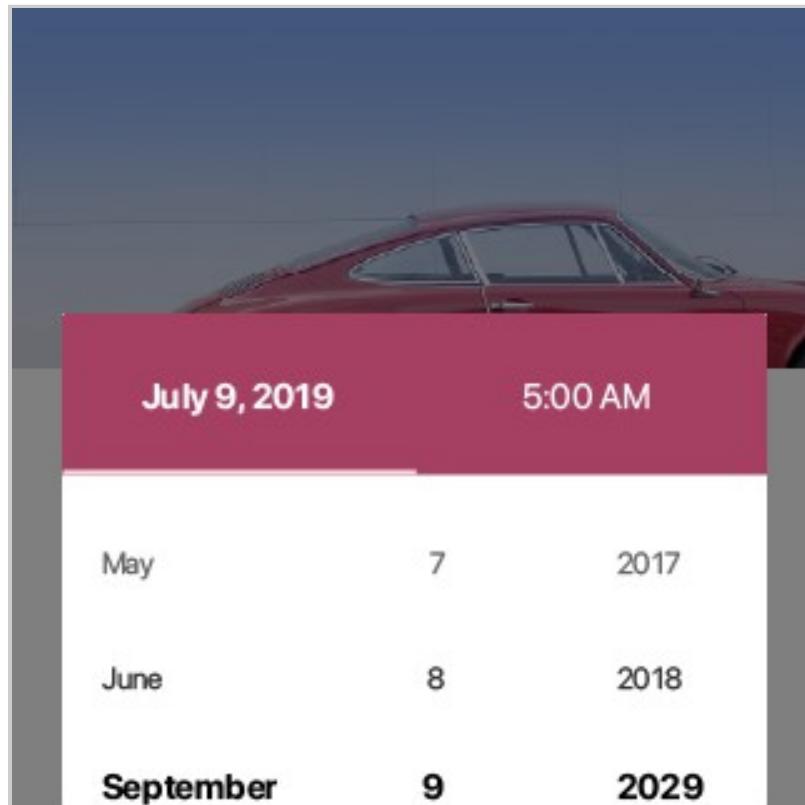
This example demonstrates how you can setup the DateTime Picker control and choose date and time in the same UI.

## See the code

[Navigate to example](#)

# DateTime Picker

## Customization



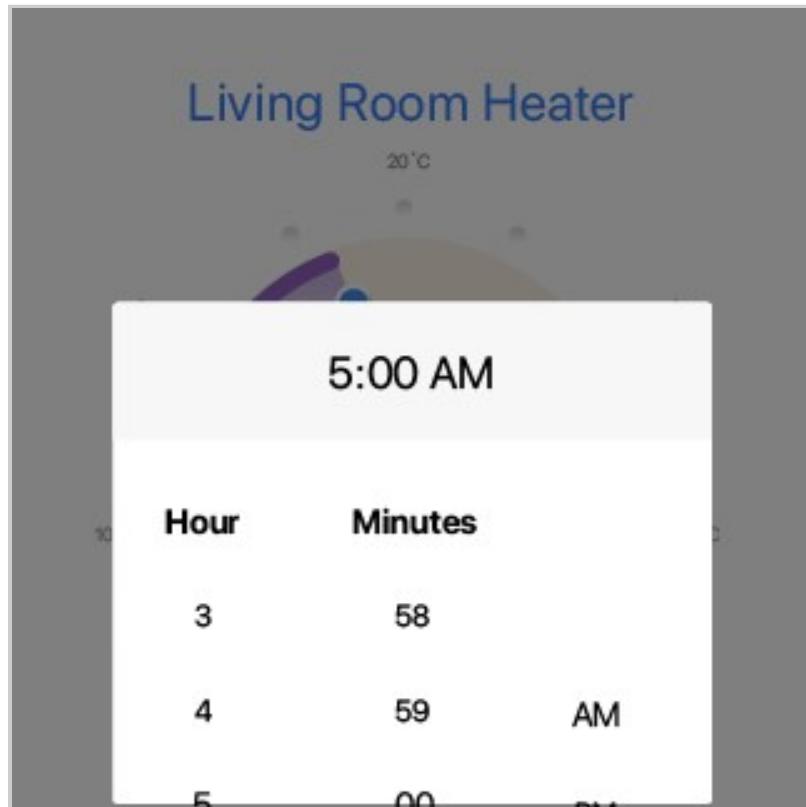
This example demonstrates how the DateTime Picker visual appearance could be customized using the control's flexible styling API.

[See the code](#)

[Navigate to example](#)

# DateTime Picker

## TimeMode



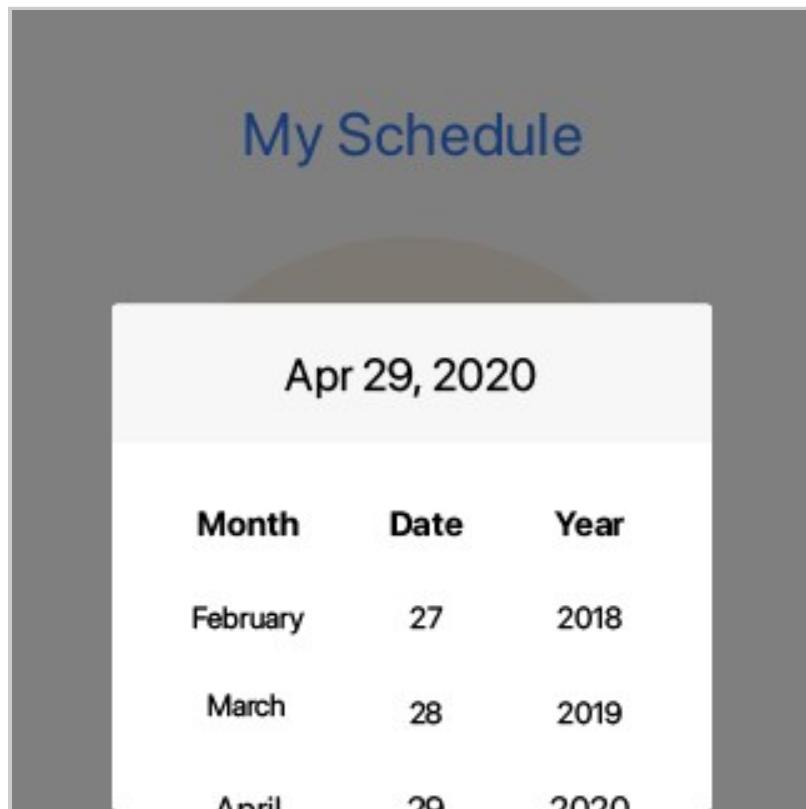
This example demonstrates how you can setup the DateTime Picker control, so that the app users can choose a time.

[See the code](#)

[Navigate to example](#)

# DateTime Picker

## DateMode



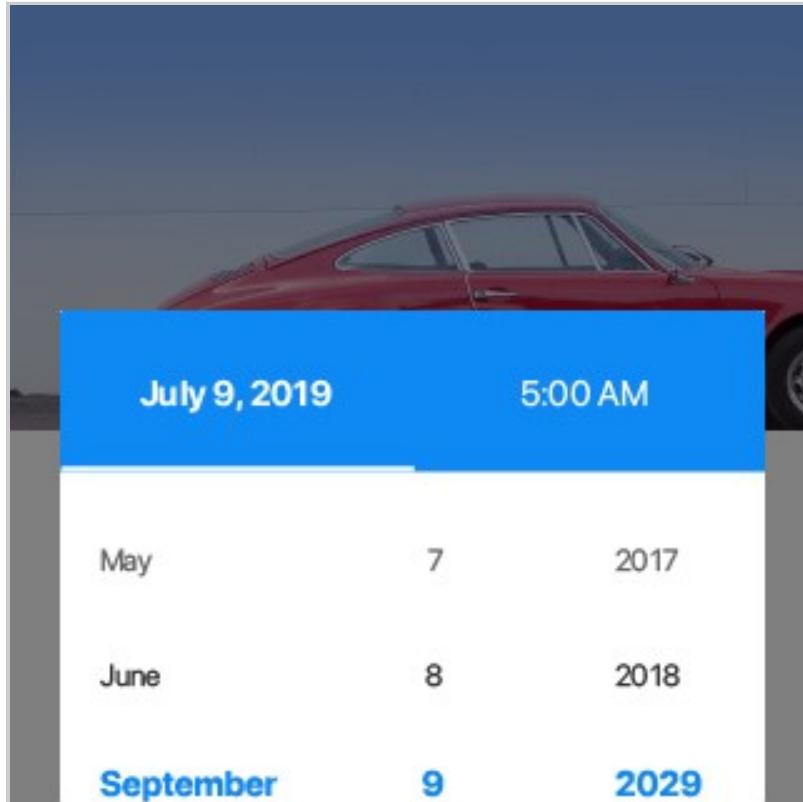
This example demonstrates how you can setup the DateTime Picker control, so that the app users can choose a date.

[See the code](#)

[Navigate to example](#)

# DateTime Picker

## Configuration



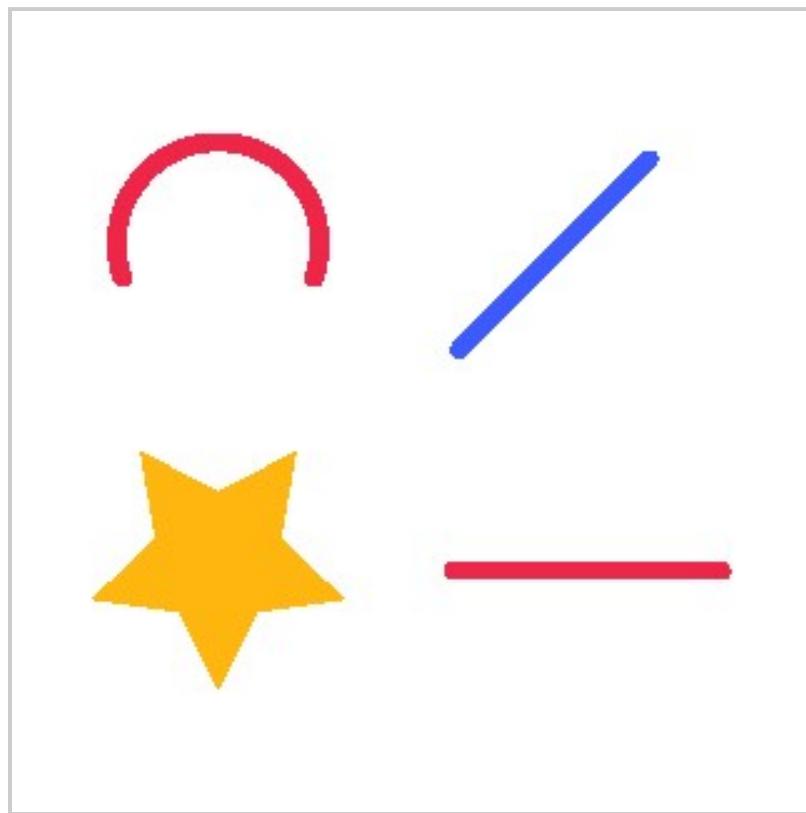
This example presents various configuration options that the RadDateTimePicker provides.

[See the code](#)

[Navigate to example](#)

# Path

RadPath is a control which can be used to draw complex shapes that are composed of different geometries like arcs, ellipses, lines and rectangles.



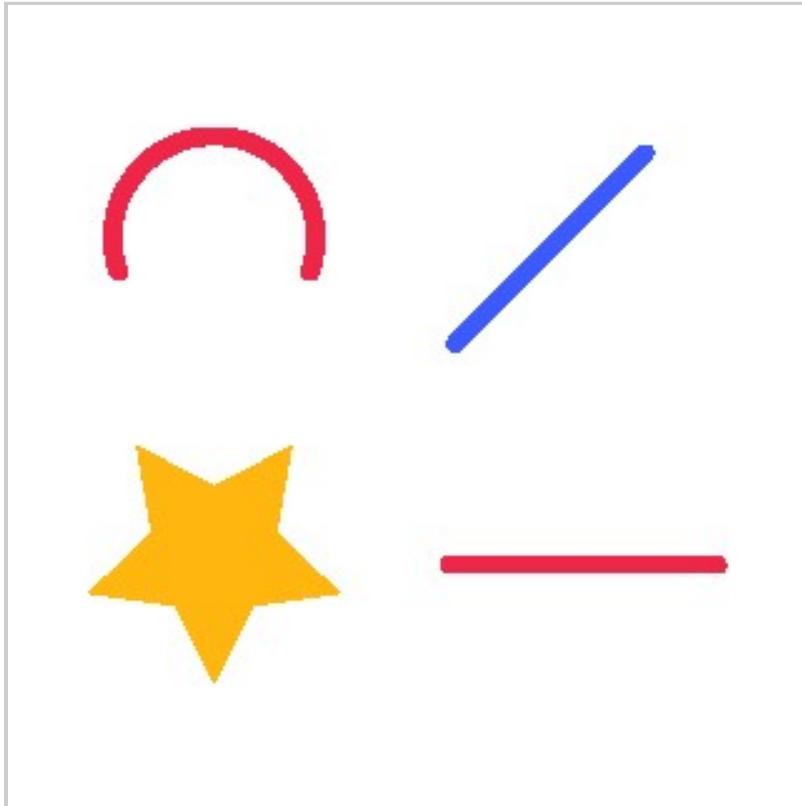
Telerik UI for Xamarin Path is a control which can be used to draw complex shapes that can be composed of different geometries like arcs, ellipses, lines and rectangles.

[Online documentation](#)

[Navigate to examples](#)

# Path

## Figures



Demonstrates RadPath elements with different figures.

[See the code](#)

[Navigate to example](#)

# Rating

RadRating is UI component that allows users to intuitively rate by selecting number of items [stars] from a predefined number of items.



Telerik UI for Xamarin Rating is a flexible UI component that allows users to intuitively rate by making a selection.

[Online documentation](#)

[Navigate to examples](#)

# Rating

## Restaurant Review



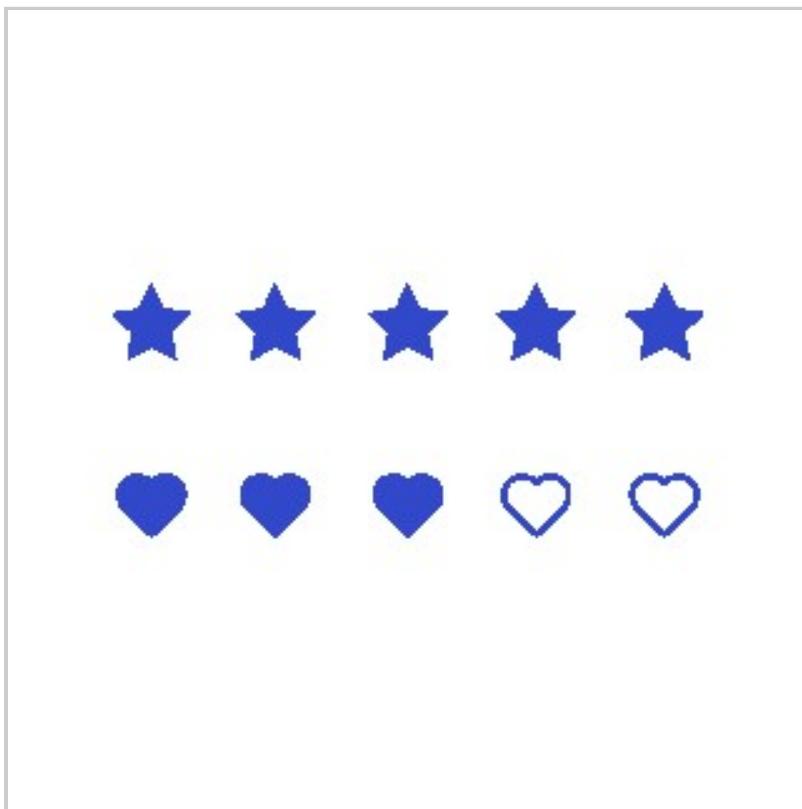
Demonstrates integration between RadRating and RadGauge components.

[See the code](#)

[Navigate to example](#)

# Rating

## Theming



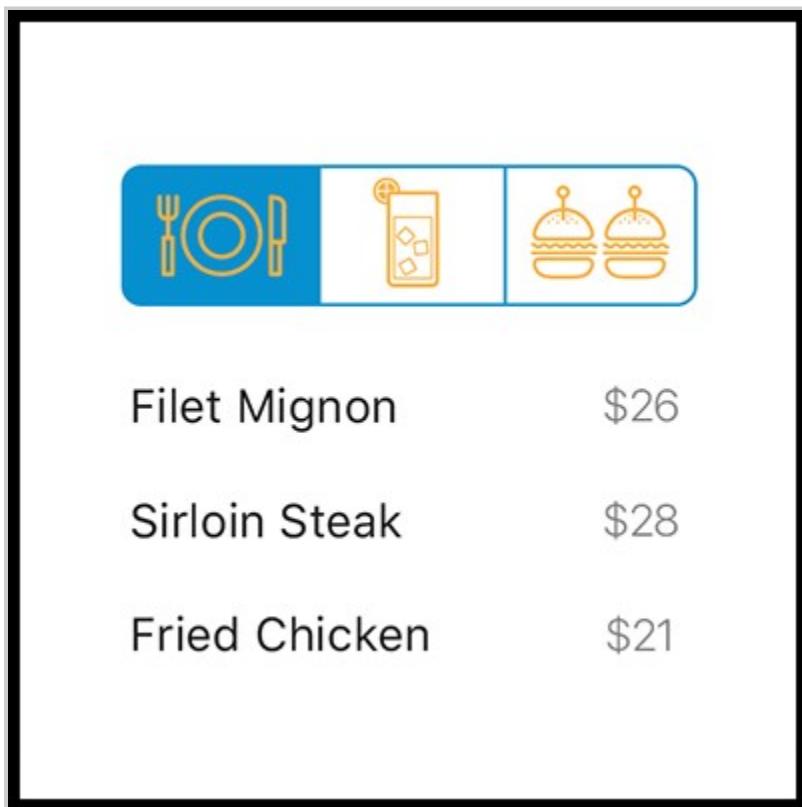
Demonstrates Blue theme applied on RadRating.

[See the code](#)

[Navigate to example](#)

# SegmentedControl

RadSegmentedControl allows you to display a list of horizontally aligned, mutually exclusive options, which can be selected by the user. Each option can be presented with a text or an image.



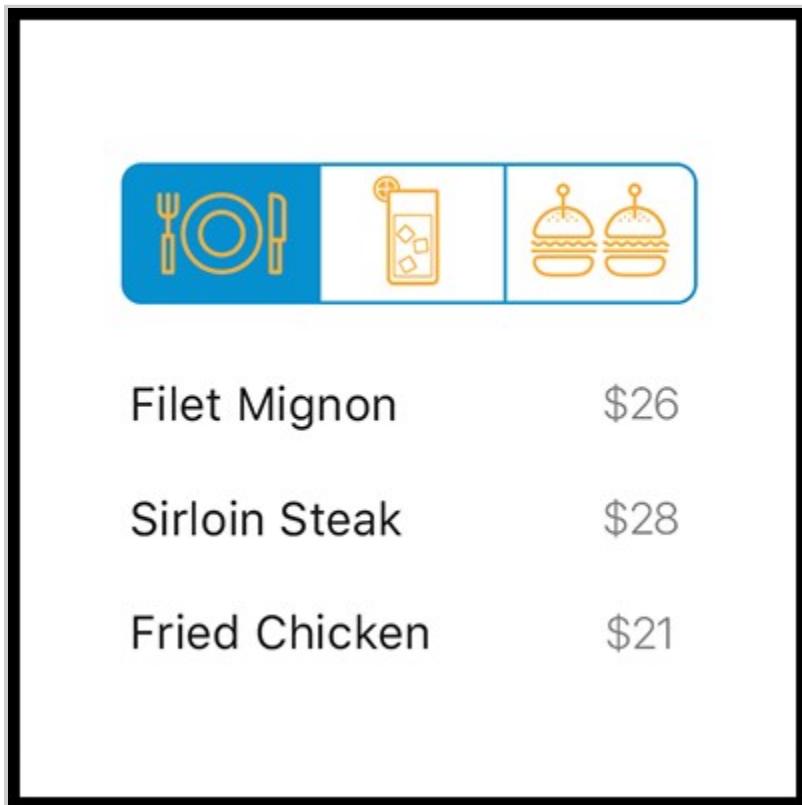
Telerik UI for Xamarin Segmented control enables you to display several options in a convenient picker and select value with a single tap.

[Online documentation](#)

[Navigate to examples](#)

# SegmentedControl

## First Look



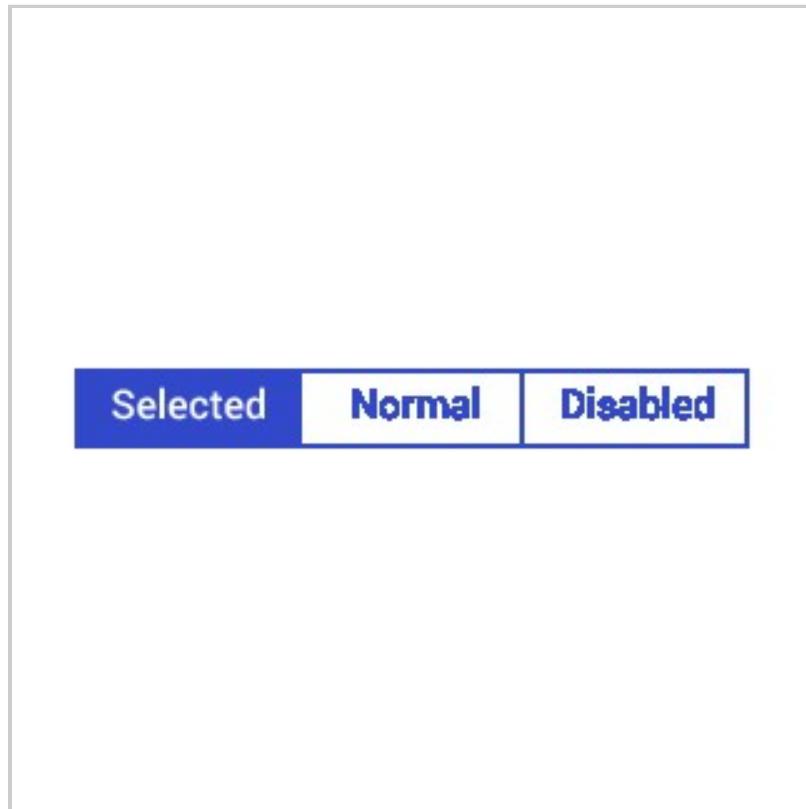
Demonstrates the most important set of features of SegmentedControl.

[See the code](#)

[Navigate to example](#)

# SegmentedControl

## Theming



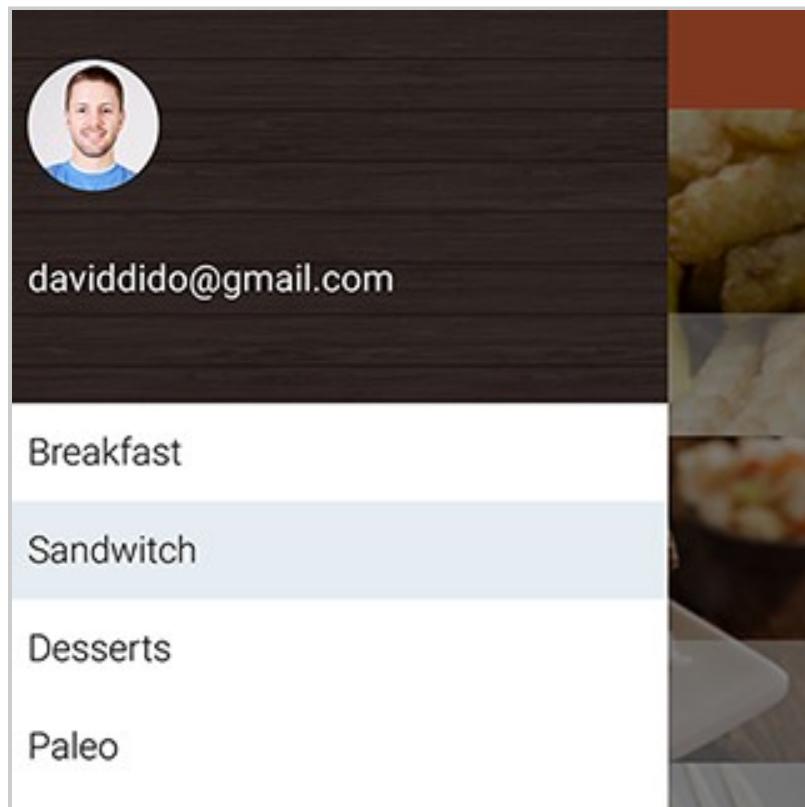
Demonstrates Blue theme applied on SegmentedControl.

[See the code](#)

[Navigate to example](#)

# SideDrawer

RadSideDrawer component is designed to enable users to visualize a hidden view in their applications. That view can host navigation UI, common setting or any other UI typical for a View.



Telerik UI for Xamarin SideDrawer extends the popular slide-out design pattern which is mainly associated with navigational purposes. The control allows developers to embed any content inside the sliding panel from text and icons to sliders and filters. The control is highly customizable and slides in from all four sides of the screen. Various effects and transition modes come out of the box, including Push, Reveal, Reverse Slide Out, Slide Along, Slide In On Top, Scale Up, Fade In.

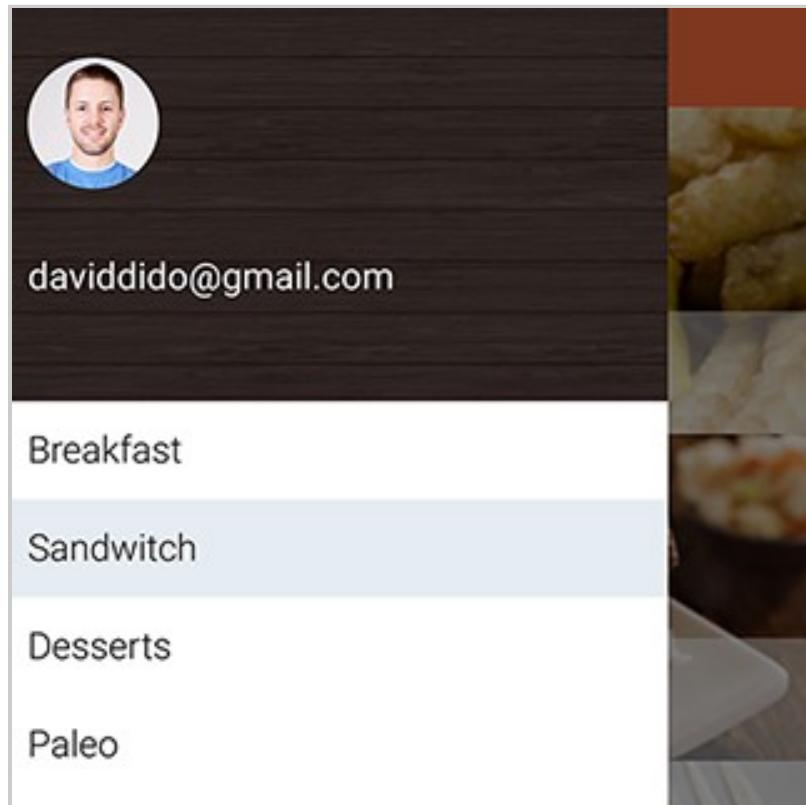
[Online documentation](#)

[Navigate to examples](#)

[telerik.com](http://telerik.com)

# SideDrawer

## Recipes



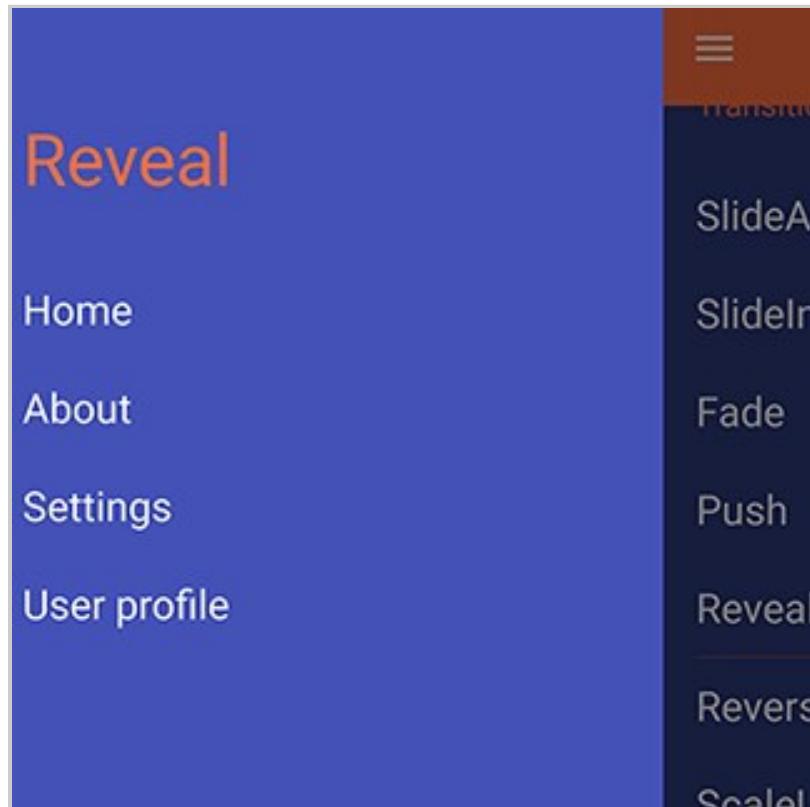
Demonstrates a real-world scenario where the SideDrawer control can come in handy. Thanks to the flexible API you can customize the SideDrawer to match the look and feel of your application.

[See the code](#)

[Navigate to example](#)

# SideDrawer

## Settings



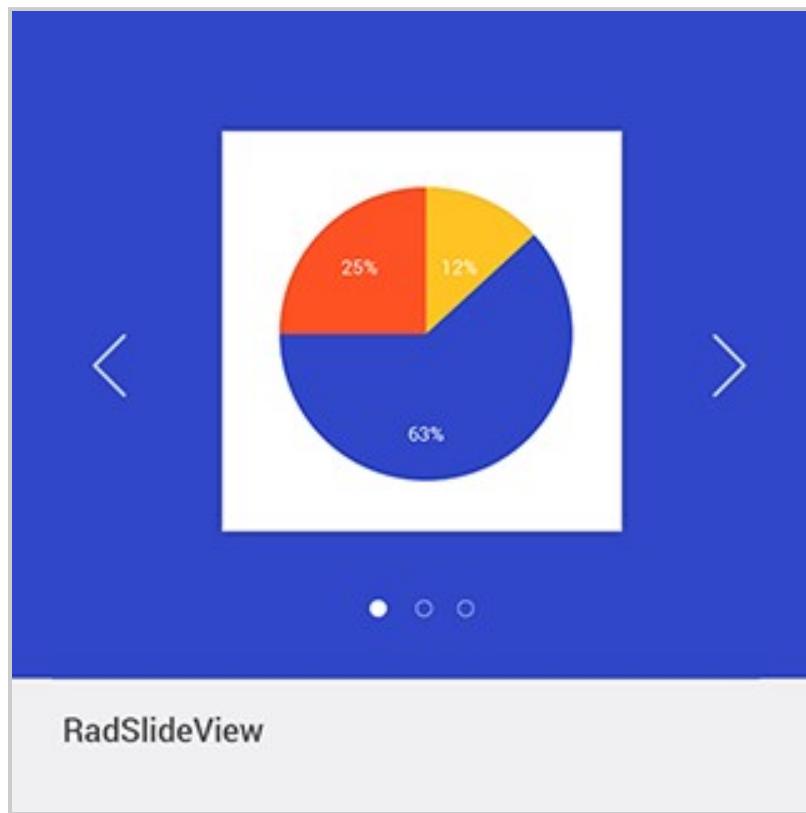
The Telerik SideDrawer can display any type of content and can be opened from any edge of the screen. Further customization can be achieved through the predefined open/close animations.

[See the code](#)

[Navigate to example](#)

## SlideView

RadSlideView is a flexible navigation control that allows you to slide between different views, thus providing an interactive navigation.



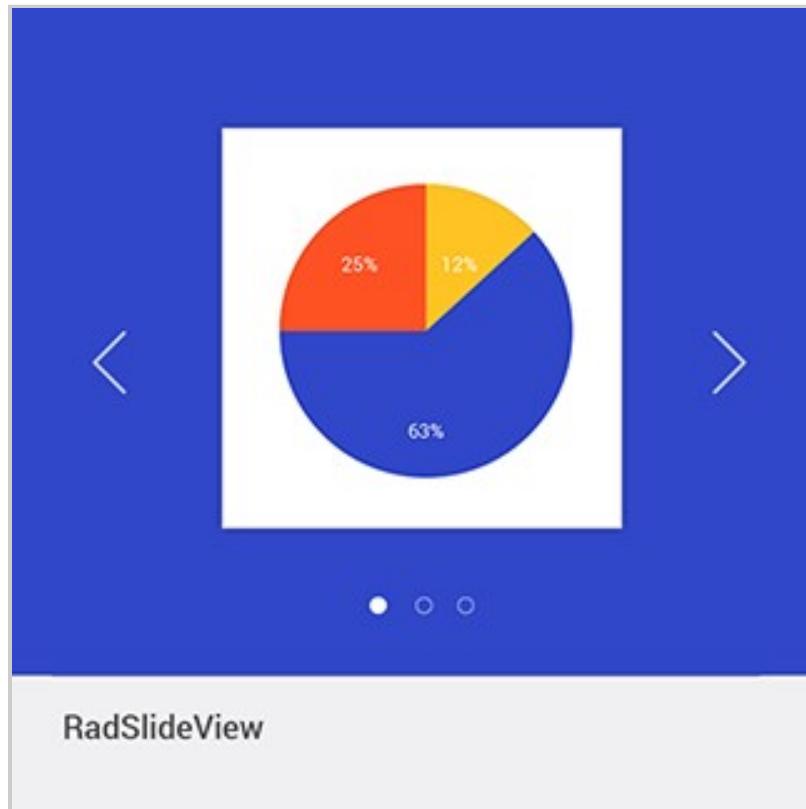
From stock tickers and news scrollers, to product spotlights, the Telerik UI for Xamarin SlideView control delivers high interactivity and user involvement. It allows extensive customizations in terms of functionality and appearance.

[Online documentation](#)

[Navigate to examples](#)

# SlideView

## Sliding Images



From stock tickers and news scrollers, to product spotlights, the SlideView control delivers high interactivity and user involvement. It allows extensive customizations in terms of functionality and appearance.

[See the code](#)

[Navigate to example](#)

# SlideView

## Theming

### SlideView Key Features

ScrollDirection -  
specifies  
possible directions  
for scrolling



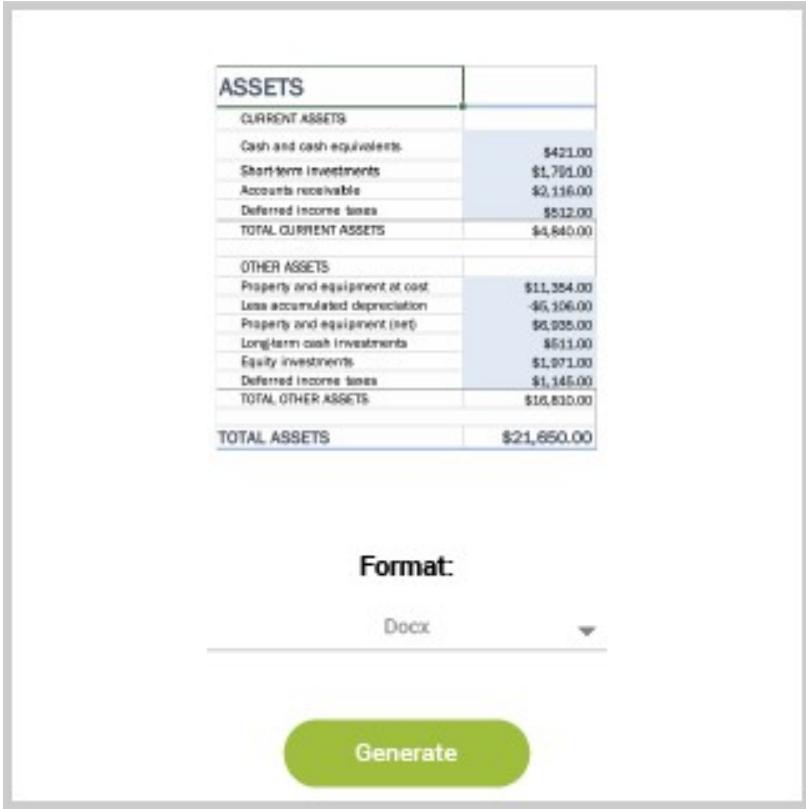
Demonstrates the theming of the SlideView.

[See the code](#)

[Navigate to example](#)

# SpreadProcessing

This library enables you to work with spreadsheet documents – create and modify documents or convert between the most common spreadsheet formats.



The screenshot shows a software interface for generating spreadsheets. At the top, there is a preview area displaying a table titled "ASSETS". The table details various asset categories and their values, such as Current Assets (\$4,840.00) and Other Assets (\$16,830.00), totaling \$21,650.00. Below the preview, there is a "Format:" dropdown menu set to "Docx". A large green "Generate" button is centered at the bottom of the interface.

ASSETS	
CURRENT ASSETS	
Cash and cash equivalents	\$421.00
Short-term investments	\$1,791.00
Accounts receivable	\$2,116.00
Deferred income taxes	\$512.00
TOTAL CURRENT ASSETS	\$4,840.00
OTHER ASSETS	
Property and equipment at cost	\$11,354.00
Less accumulated depreciation	-6,196.00
Property and equipment (net)	\$6,958.00
Long-term cash investments	\$611.00
Equity investments	\$1,971.00
Deferred income taxes	\$1,145.00
TOTAL OTHER ASSETS	\$16,830.00
TOTAL ASSETS	\$21,650.00

This library enables you to work with spreadsheet documents – create ones from scratch, modify existing documents or convert between the most common spreadsheet formats.

[Online documentation](#)

[Navigate to examples](#)

# SpreadProcessing

## First Look



The screenshot shows a spreadsheet application interface with a light gray background. At the top, there is a title bar with the word "Format:" and a dropdown menu set to "Docx". Below the title bar is a green button labeled "Generate". The main content area displays a table representing an asset balance sheet. The table has a header row "ASSETS" and two main sections: "CURRENT ASSETS" and "OTHER ASSETS". The "CURRENT ASSETS" section includes rows for Cash and cash equivalents (\$421.00), Short-term investments (\$1,791.00), Accounts receivable (\$2,136.00), and Deferred income taxes (\$532.00). The "TOTAL CURRENT ASSETS" is \$4,840.00. The "OTHER ASSETS" section includes rows for Property and equipment at cost (\$11,354.00), Less accumulated depreciation (-\$6,106.00), Property and equipment (net) (\$5,248.00), Long-term cash investments (\$611.00), Equity investments (\$1,971.00), Deferred income taxes (\$1,145.00), and the "TOTAL OTHER ASSETS" is \$16,830.00. The "TOTAL ASSETS" is \$21,650.00.

ASSETS	
CURRENT ASSETS	
Cash and cash equivalents	\$421.00
Short-term investments	\$1,791.00
Accounts receivable	\$2,136.00
Deferred income taxes	\$532.00
TOTAL CURRENT ASSETS	\$4,840.00
OTHER ASSETS	
Property and equipment at cost	\$11,354.00
Less accumulated depreciation	-\$6,106.00
Property and equipment (net)	\$5,248.00
Long-term cash investments	\$611.00
Equity investments	\$1,971.00
Deferred income taxes	\$1,145.00
TOTAL OTHER ASSETS	\$16,830.00
TOTAL ASSETS	\$21,650.00

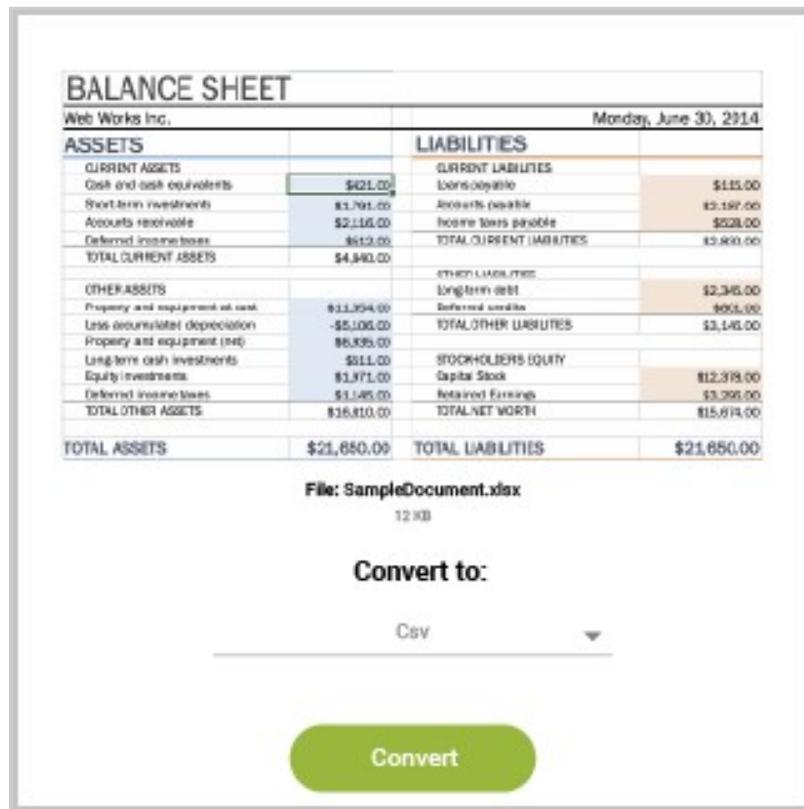
Telerik SpreadProcessing library provides easy to use API to create, modify, and convert spreadsheet documents. This example demonstrates how to create a document.

[See the code](#)

[Navigate to example](#)

# SpreadProcessing

## Convert



The screenshot shows a "BALANCE SHEET" for "Web Works Inc." dated "Monday, June 30, 2014". The table is divided into "ASSETS" and "LIABILITIES" sections. The "ASSETS" section includes "CURRENT ASSETS" (Cash and cash equivalents, Short-term investments, Accounts receivable, Deferred income taxes, TOTAL CURRENT ASSETS) and "OTHER ASSETS" (Property and equipment at cost, Less accumulated depreciation, Property and equipment net, Long-term cash investments, Equity investments, Deferred income taxes, TOTAL OTHER ASSETS). The "LIABILITIES" section includes "CURRENT LIABILITIES" (Accounts payable, Accrued liabilities, Income taxes payable, TOTAL CURRENT LIABILITIES) and "OTHER LIABILITIES" (Long-term debt, Deferred revenue, TOTAL OTHER LIABILITIES). The bottom row shows "TOTAL ASSETS" and "TOTAL LIABILITIES" both equaling "\$21,650.00". Below the table, it says "File: SampleDocument.xlsx" and "12 KB". A "Convert to:" dropdown menu is set to "Csv". A large green "Convert" button is at the bottom.

BALANCE SHEET			
Web Works Inc.		Monday, June 30, 2014	
ASSETS		LIABILITIES	
CURRENT ASSETS		CURRENT LIABILITIES	
Cash and cash equivalents	\$421.00	Accounts payable	\$115.00
Short-term investments	\$1,791.00	Accrued liabilities	\$3,187.00
Accounts receivable	\$2,116.00	Income taxes payable	\$228.00
Deferred income taxes	\$512.00	TOTAL CURRENT LIABILITIES	\$3,460.00
TOTAL CURRENT ASSETS	\$4,840.00		
OTHER ASSETS		OTHER LIABILITIES	
Property and equipment at cost	\$13,394.00	Long-term debt	\$2,346.00
Less accumulated depreciation	-\$5,106.00	Deferred revenue	\$65.00
Property and equipment net	\$8,288.00	TOTAL OTHER LIABILITIES	\$2,411.00
Long-term cash investments	\$511.00		
Equity investments	\$1,971.00		
Deferred income taxes	\$1,485.00		
TOTAL OTHER ASSETS	\$16,810.00		
TOTAL ASSETS	\$21,650.00	TOTAL LIABILITIES	\$21,650.00

File: SampleDocument.xlsx  
12 KB

Convert to:

Csv

Convert

An example that demonstrates how to import documents and convert to different formats.

[See the code](#)

[Navigate to example](#)

# SpreadProcessing

## Find & Replace

File: SampleDocument.xlsx

[View Document](#)

**Find Text:**

Enter Text to be found ex. equity

**Replace with:**

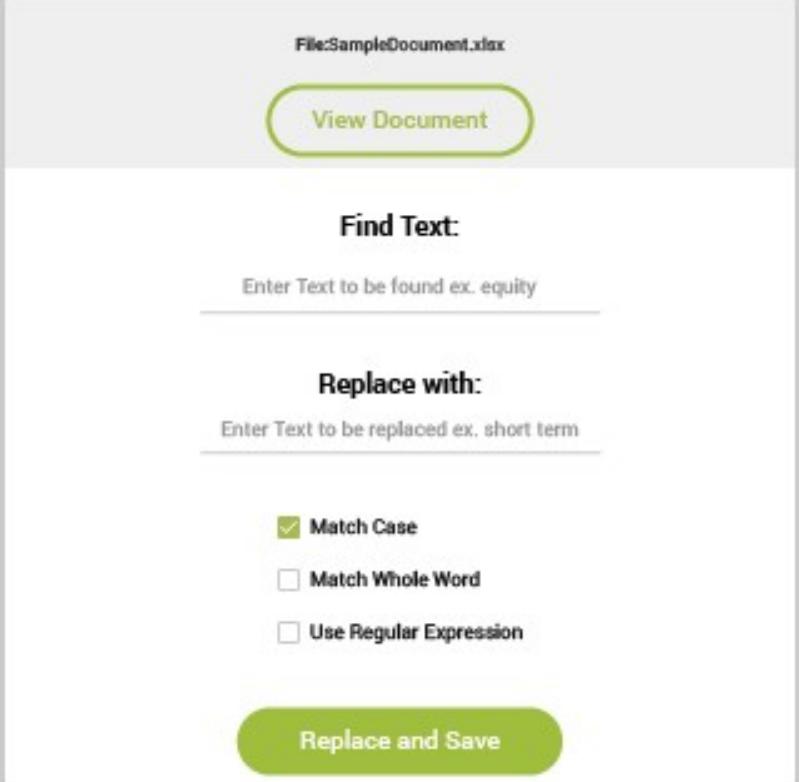
Enter Text to be replaced ex. short term

Match Case

Match Whole Word

Use Regular Expression

[Replace and Save](#)



A demonstration of the built-in Find and Replace functionality in the SpreadProcessing library.

[See the code](#)

[Navigate to example](#)

# SpreadStreamProcessing

RadSpreadStreamProcessing is a document processing paradigm that allows you to create big spreadsheet documents with great performance and minimal memory footprint.

MY COURSES	
TITLE	UNIVERSITY
Data Science	John Hopkins University
Machine Learning	University of Washington
Big Data	University of California
Product Management	University of California
Business Foundations	University of Pennsylvania
Python for Everybody	University of Michigan
Data Science	John Hopkins University
Machine Learning	University of Washington

Telerik UI for Xamarin SpreadStreamProcessing is a document processing library which allows you to create big spreadsheet documents with great performance and minimal memory footprint.

[Online documentation](#)

[Navigate to examples](#)

# SpreadStreamProcessing

## Generate Spreadsheet

MY COURSES	
TITLE	UNIVERSITY
Data Science	John Hopkins University
Machine Learning	University of Washington
Big Data	University of California
Product Management	University of California
Business Foundations	University of Pennsylvania
Python for Everybody	University of Michigan
Data Science	John Hopkins University
Machine Learning	University of Washington

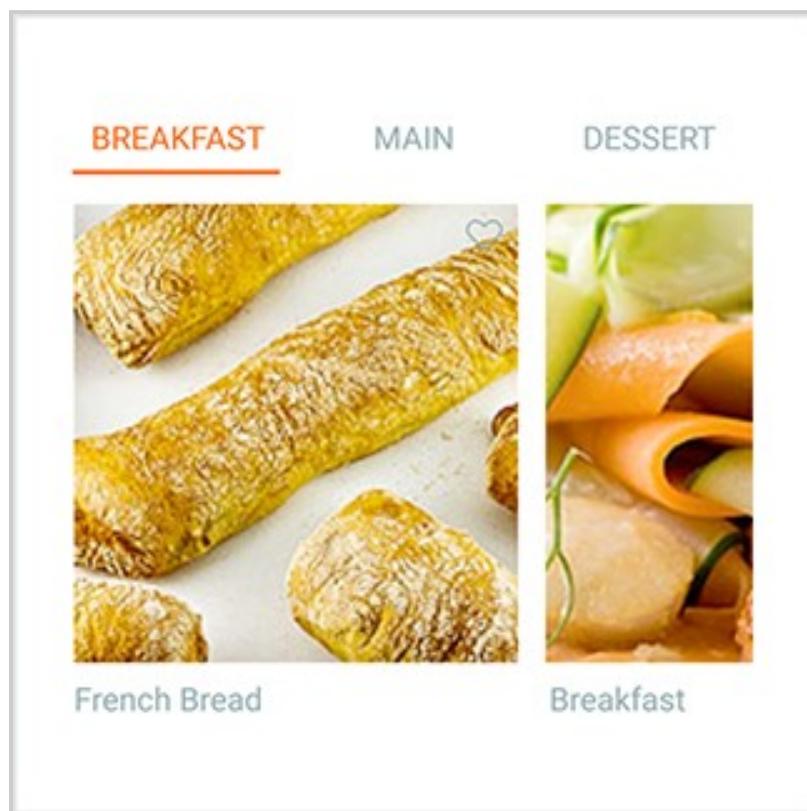
An example demonstrating generating spreadsheet document using SpreadStreamProcessing library.

[See the code](#)

[Navigate to example](#)

# TabView

RadTabView is a flexible navigation control that allows you to build tabbed interfaces. Each tabview item has an associated content displayed on selection.



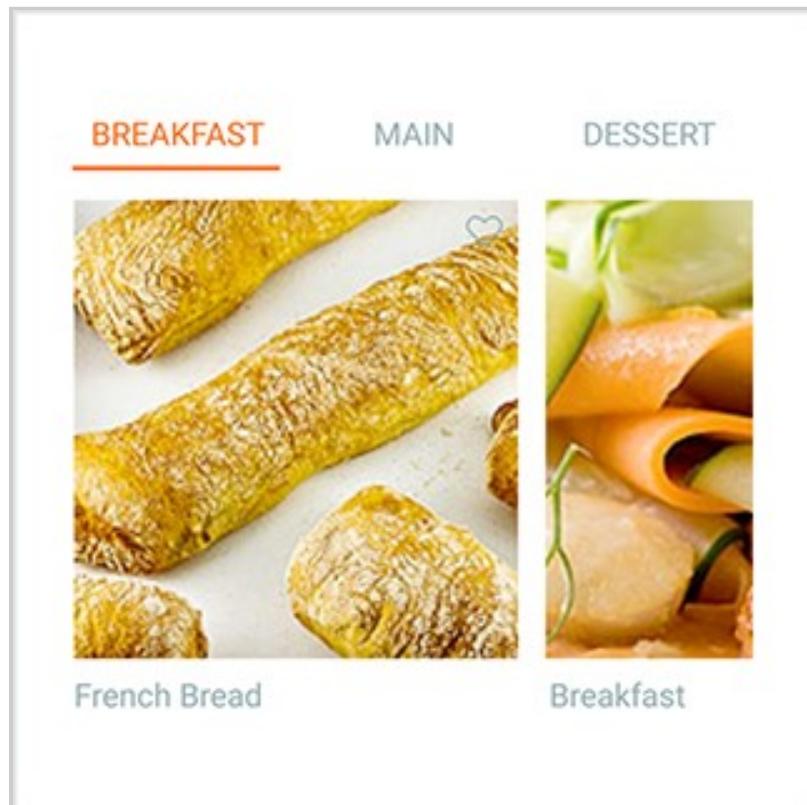
Telerik UI for Xamarin TabView is a flexible navigation control that allows you to build tabbed interfaces. Each tabview item has an associated content displayed on selection.

[Online documentation](#)

[Navigate to examples](#)

# TabView

## Restaurant Menu



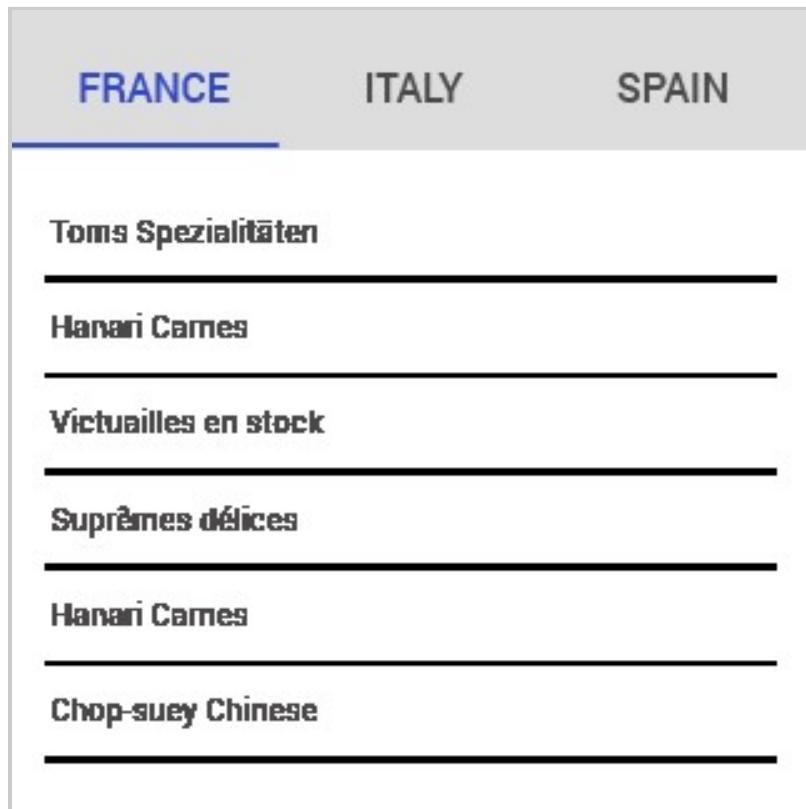
Demonstrates the main functionality of the TabView.

[See the code](#)

[Navigate to example](#)

# TabView

## Theming



Demonstrates the theming of the TabView.

[See the code](#)

[Navigate to example](#)

# WordsProcessing

RadWordsProcessing is a processing library that allows creation, import and export of DOCX documents.



Telerik RadWordsProcessing is a processing library that allows code-only generation of documents. Through the API you can create a document from scratch and export it to one of the supported formats – Docx, Rtf, Html, PDF and Txt.

[Online documentation](#)

[Navigate to examples](#)

# WordsProcessing

## First Look



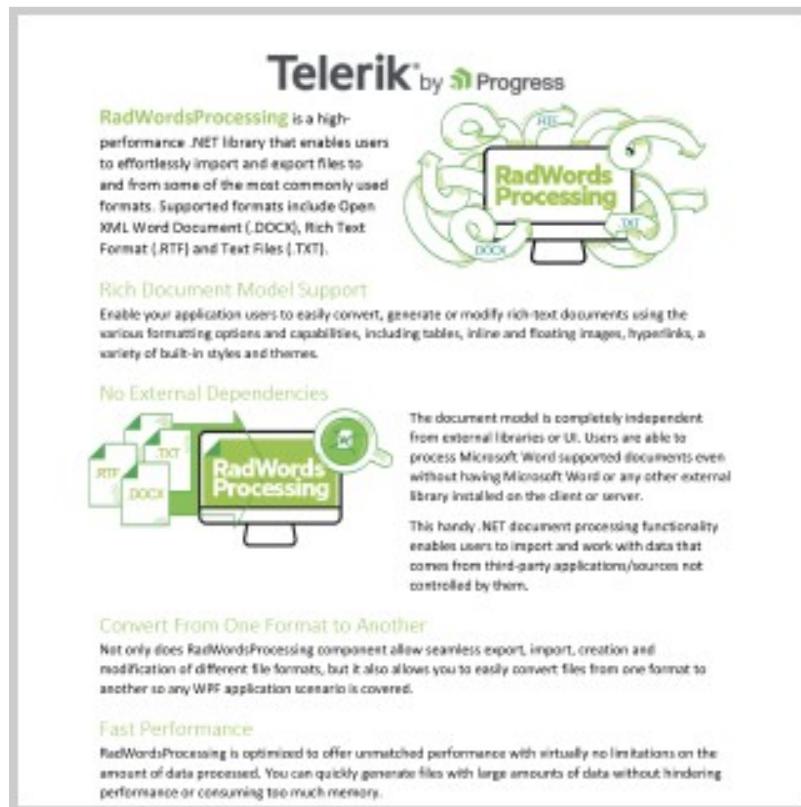
Telerik RadWordProcessing provides easy to use API which allows code-only generation of DOCX documents. This example demonstrates how to create a document containing images, text and shapes and export it to DOCX, RTF, HTML, TXT.

[See the code](#)

[Navigate to example](#)

# WordsProcessing

## Export



**Telerik® by Progress**

RadWordsProcessing is a high-performance .NET library that enables users to effortlessly import and export files to and from some of the most commonly used formats. Supported formats include Open XML Word Document (.DOCX), Rich Text Format (.RTF) and Text Files (.TXT).

**Rich Document Model Support**

Enable your application users to easily convert, generate or modify rich-text documents using the various formatting options and capabilities, including tables, inline and floating images, hyperlinks, a variety of built-in styles and themes.

**No External Dependencies**

The document model is completely independent from external libraries or UI. Users are able to process Microsoft Word supported documents even without having Microsoft Word or any other external library installed on the client or server.

This handy .NET document processing functionality enables users to import and work with data that comes from third-party applications/sources not controlled by them.

**Convert From One Format to Another**

Not only does RadWordsProcessing component allow seamless export, import, creation and modification of different file formats, but it also allows you to easily convert files from one format to another so any WPF application scenario is covered.

**Fast Performance**

RadWordsProcessing is optimized to offer unmatched performance with virtually no limitations on the amount of data processed. You can quickly generate files with large amounts of data without hindering performance or consuming too much memory.

This example demonstrates export of flow documents to PDF, RTF, HTML and TXT using RadWordsProcessing. You can use the imported Word(docx) sample and convert it to PDF, RTF, HTML, TXT.

[See the code](#)

[Navigate to example](#)

# WordsProcessing

## Find & Replace

File: SampleDocument.docx

[View Document](#)

**Find Text:**

Enter Text to be found ex. fast

---

**Replace with:**

Enter Text to be replaced ex. slow

---

Match Case

Match Whole Word

Use Regular Expression

[Replace and Save](#)

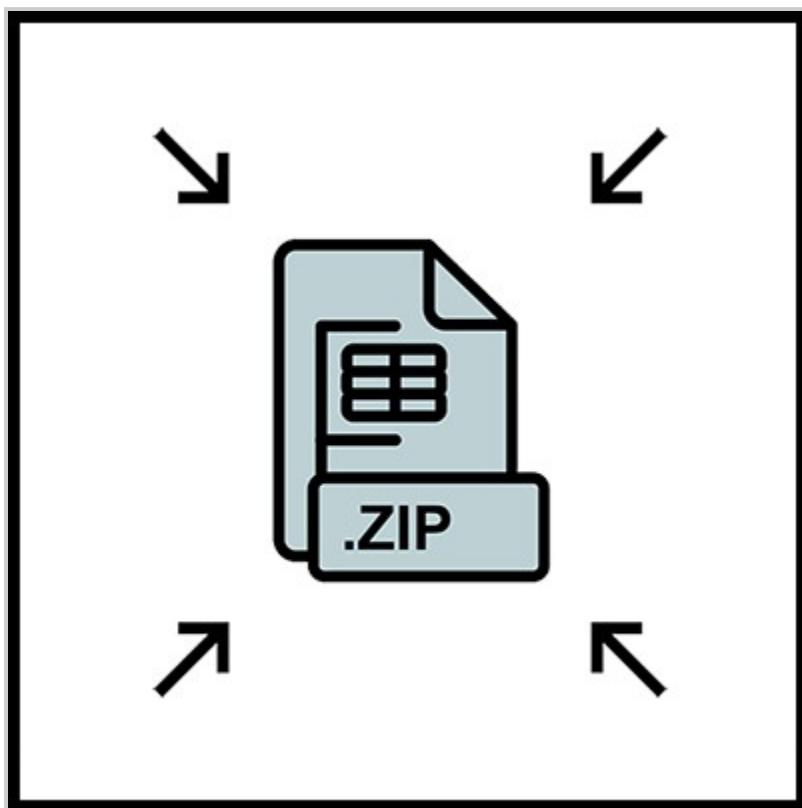
This example illustrates the Replace functionality in RadWordsProcessing. You can replace or highlight text in the sample document, using regular expressions or strings and then export the document to view the result.

[See the code](#)

[Navigate to example](#)

## ZipLibrary

RadZipLibrary allows you to compress data like images, docx or pdf files and send them over the wire, thus achieving fast and secure transactions.



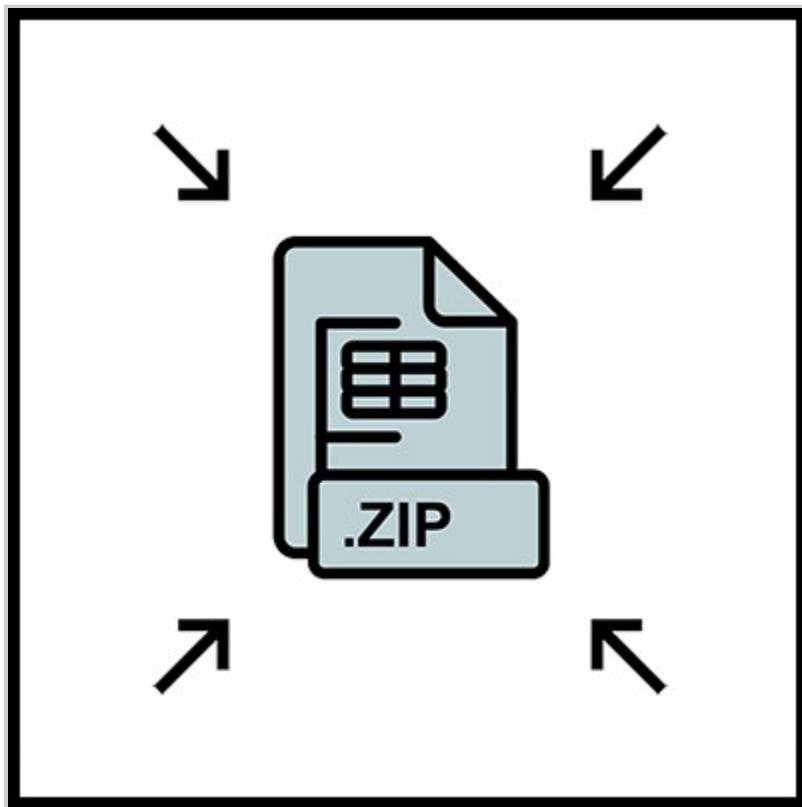
Telerik UI for Xamarin ZipLibrary provides easy-to-use API to both create and load zip files as well as browse the contents of a zip file previously created. The API supports password protection and different compression levels. You can also create ZIP files in memory or add data to ZIP file from stream.

[Online documentation](#)

[Navigate to examples](#)

# ZipLibrary

## Create Archive



An example demonstrating creating zip archive using ZipLibrary.

[See the code](#)

[Navigate to example](#)