



DEVELOPING OUTSYSTEMS MOBILE APPS

# Charts



---

# Introduction

Over the course of this set of exercise labs, you will create a mobile application. The application will focus on creating and managing To Dos. The To Dos will be persisted in a database so they can be accessed from and shared across multiple devices. To Dos will have attributes such as category, priority (low, medium or high), due date and they can be marked as important (starred) by the user.

Users of the To Do application will be able to access all of this information regardless of whether the device is online or offline. When offline, users will still be able to keep interacting with the application and changes will be saved locally in the device local storage. When the device returns to online mode, changes made while offline will automatically be synced to the server.

You constantly will be expanding your application, publishing it to the server and testing it in your mobile device. Throughout the process you will be learning and applying new OutSystems concepts.

At the end of this set of exercise labs, you will have a small, but well-formed application, spanning multiple screens and concepts that you can easily access from your mobile device.

In this specific exercise lab, you will:

- Add a **Donut Chart** to view the number of To Dos by Priority
- Add a **Bar Chart** to view the number of To Dos by Category

---

# Table of Contents

Introduction.....	2
Table of Contents.....	3
Part 1: Add Charts to the Dashboard .....	4
End of Lab .....	9
List of Figures.....	10

## Part 1: Add Charts to the Dashboard

In this part of the exercise, you will add a **Donut Chart** and a **Bar Chart** to the **Dashboard** Screen.

1. Add a **Donut Chart** to the **Dashboard** Screen to display the number of To Dos for each priority.

a) Right-click the **Dashboard** Screen and choose 'Fetch Data from Local Storage' to add a new Aggregate.

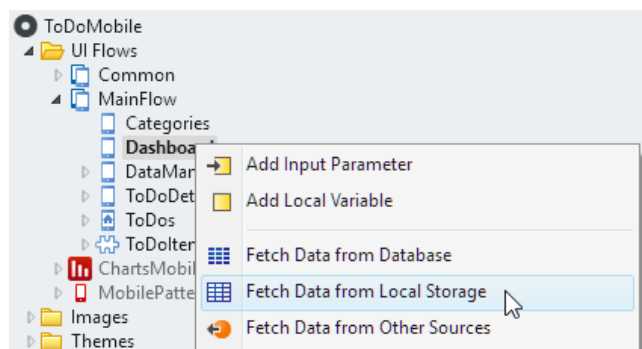


Figure 1. Add a new Local Storage Aggregate

b) From the **Data** tab drag the **LocalToDo** Entity and drop it inside the Aggregate editor, then repeat the same for the **LocalPriority** Entity.

c) Change the Name of the Aggregate to 'GetToDosWithPriorities'.

d) Place the mouse over the **LocalPriority Label** attribute then click the arrow to expand the context menu.



Figure 2. Open attribute context menu

e) Choose 'Group by Label' in the context menu.

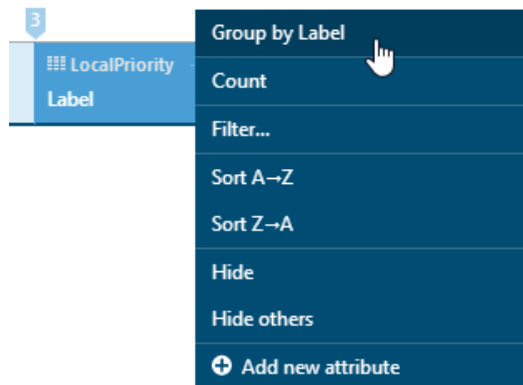


Figure 3. Group by Label

- f) Open the context menu of the **LocalToDo Title** attribute, then choose 'Count'.
- g) Open the **Dashboard** Screen.
- h) Drag a **Container** Widget to the main area of the Screen, inside the **Content** placeholder.
- i) Set the **Align** property to 'Center'.
- j) Inside the **Container** write 'To Dos by Priority', then apply the "heading3" **Style Class** to the text.
- k) Drag a **Donut Chart** and drop it below the text but still inside the surrounding **Container**.

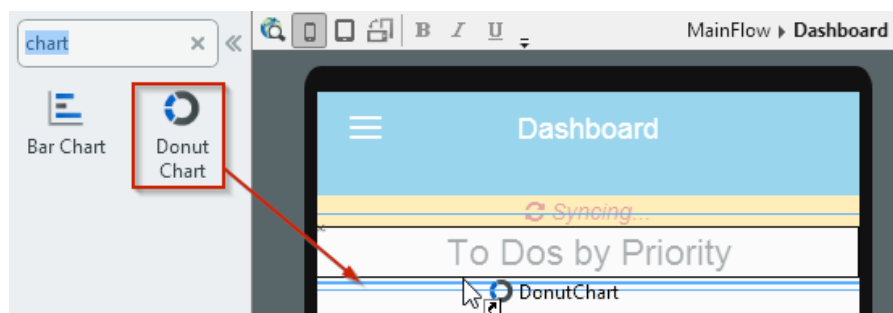


Figure 4. Drag and drop a Donut Chart

- l) In the properties, set the **SourceDataPointList** parameter to 'GetToDosWithPriorities.List'.
- m) In the Mapping to DataPoint, set the **Label** to 'Label', and the **Value** to 'Count'.

Charts\DonutChart	
Name	
Source Block	Charts\DonutChart
SourceDataPointList	GetToDosWithPriorities.List
Mapping to DataPoint	
Label	Label
Value	Count
DataSeriesName	
Tooltip	
Color	
Height	
LegendPositionId	
ChartFormat	
Clickable	

Figure 5. DataPoint mapping

2. Add a **Bar Chart** to the **Dashboard** Screen to display the number of To Dos for each category.

- a) Right-click the **Dashboard** Screen and choose 'Fetch Data from Local Storage' to create a new local Aggregate.
- b) Drag the **LocalToDo** and **LocalCategory** Entities into the Aggregate editor, then create a **Group By** of the **LocalCategory** Name attribute, and a **Count** by the **LocalToDo** Title attribute.
- c) Change the Name of the Aggregate to 'GetToDosWithCategories'.
- d) Your Aggregate should look like this

GetToDosWithCategories		Sources ②	Filters	Sorting
1				
Group of Name	Count of T...	LocalToDo	LocalToDo	LocalToDo
Name	Count	Title	IsStarred	Notes

Figure 6. Aggregate with Group by LocalCategories

- e) Return to the **Dashboard** Screen.
- f) Drag a **Separator** Widget and drop it below the Container that has the **Donut Chart**.

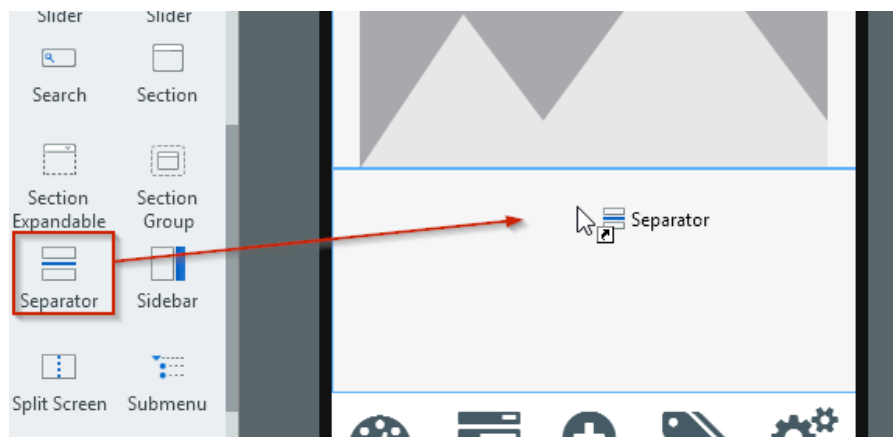


Figure 7. Drag and drop a Separator Widget

- g) Drag a new **Container** and drop it below the **Separator**.
- h) Set the **Align** property of the Container to 'Center'.
- i) Inside the Container, type 'To Dos by Category' and then apply the "heading3" **Style Class** to the Text.
- j) Drag a **Bar Chart** Widget and drop it below the Text written above, but still inside the Container.
- k) Set the **SourceDataPointList** to use the list from the Aggregate created above, then define the mapping as below

Charts\BarChart	
Name	
Source Block	Charts\BarChart
SourceDataPointList	GetToDosWithCategories.List
Mapping to DataPoint	
Label	Name
Value	Count
DataSetName	
Tooltip	
Color	
StackingType	

Figure 8. Bar Chart properties

3. Publish and test the application.
  - a) Click the **1-Click Publish** Button to publish the module to the server.
  - b) Click the **Open in Browser** Button to open the application, or test the application in your device.

- c) Navigate to the **Dashboard** Screen and verify that both Charts appear correctly.

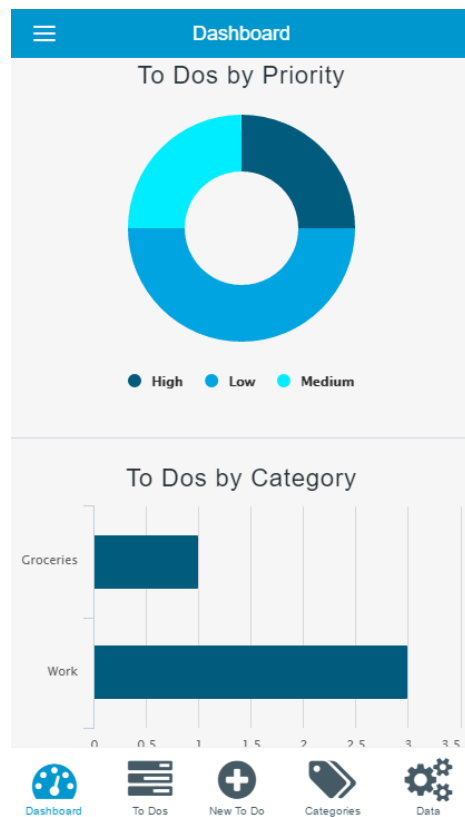


Figure 9. Dashboard Screen



---

## End of Lab

In this exercise, you added two **Charts** to the **Dashboard** Screen: a **Donut Chart** and a **Bar Chart**. These **Charts** enable users to have an overview of To Dos by Priority and by Category.

---

## List of Figures

Here is the list of screenshots and pictures used in this exercise.

Figure 1. Add a new Local Storage Aggregate .....	4
Figure 2. Open attribute context menu.....	4
Figure 3. Group by Label .....	5
Figure 4. Drag and drop a Donut Chart .....	5
Figure 5. DataPoint mapping.....	6
Figure 6. Aggregate with Group by LocalCategories.....	6
Figure 7. Drag and drop a Separator Widget.....	7
Figure 8. Bar Chart properties.....	7
Figure 9. Dashboard Screen .....	8