

Working with Navigation Containers and Embedding Fonts and Images

Import the Project

1. In Flex Builder, select **File > Import > Flex Project**.
2. In the dialog window, select **Archive File** and browse to where `Ex14_Starter.zip` is located in your local file system.
3. Click **Finish**.

Create and use a TabNavigator

4. In Flex Builder open **ReservationSystem.mxml** from the **src** folder of the project.
5. After the beginning `Application` tag, add a `Canvas` tag set.
6. Between the `Canvas` tags, add a `TabNavigator` tag set.
7. Add the following properties and values to the beginning `TabNavigator` tag:
 - `id : tn`
 - `x : 0`
 - `y : 0`
 - `width : 864`
 - `height : 567`

```
<mx:Canvas>
  <mx:TabNavigator id="tn"
    x="0" y="0"
    width="864" height="567">
  </mx:TabNavigator>
</mx:Canvas>
```

Note: Generally you would not need the parent `Canvas` container for the display since setting the `layout` property to `absolute` in the `Application` tag will have the same effect. However, this `Canvas` container will be used in the next exercise to enhance the application layout.

8. In the beginning `Application` tag, add a new namespace named `comp` that points to the `components` folder.

```
xmlns:comp="components.*"
```

9. Within the `Canvas` tag block, below the `TabNavigator`, add the `Footer` component in the `comp` namespace.
10. Add an `x` property with a value of 0 (zero) and a `y` property with a value of 567.

```
<comp:Footer x="0" y="567"/>
```

11. Between the `TabNavigator` tags add an instance of the `RoomReservations` component from the `comp` namespace. Add a `label` property with a value of `ROOM RESERVATION`.

```
<comp:RoomReservations label="ROOM RESERVATION"/>
```

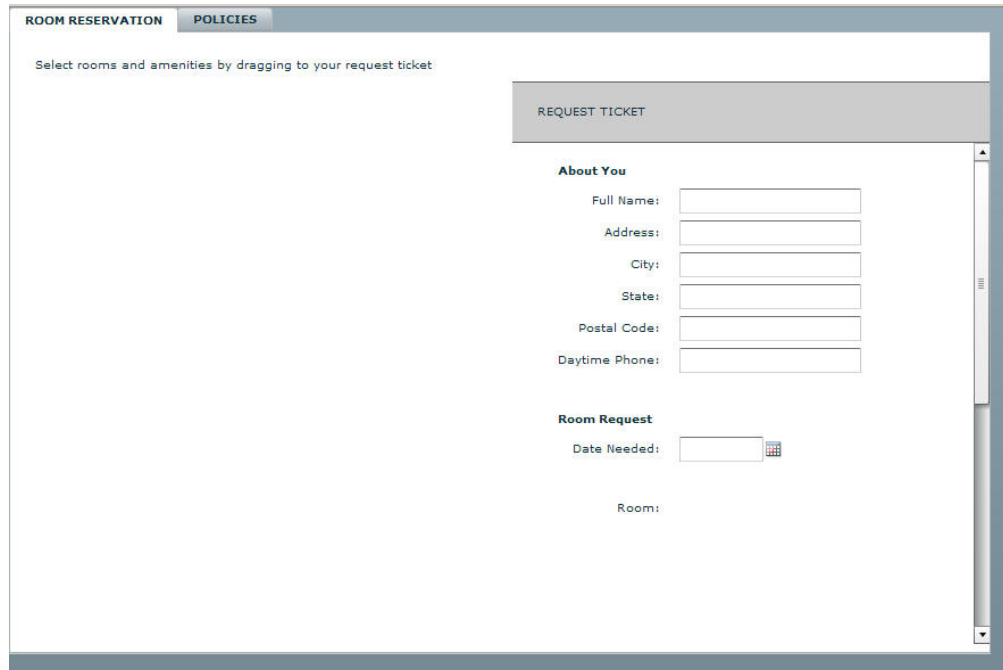
12. After the `RoomReservations` component instance, add an instance of the `Policies` component from the `comp` namespace.
13. Add a `label` property with a value of `POLICIES`.

```
<comp:Policies label="POLICIES"/>
```

Your code should appear as follows:

```
<mx:Canvas>
  <mx:TabNavigator id="tn"
    x="0" y="0"
    width="864" height="567">
    <comp:RoomReservations
      label="ROOM RESERVATIONS"/>
    <comp:Policies label="POLICIES"/>
  </mx:TabNavigator>
  <comp:Footer x="0" y="567"/>
</mx:Canvas>
```

14. Save the file and run the application.
You should see a tabbed application as shown in the following figure.



15. Click the **Policies** tab.
You should see a listing of policies.

Create and use an Accordion

16. From the **Flex Navigator** view, open the **RoomReservations.mxml** file from the `components` folder.
17. Using the **Outline** view, locate the `Canvas` tag following the `Text` tag.
18. Between the `Canvas` tags and before the `RequestTicket` component, add an `Accordion` tag set.
19. To the beginning `Accordion` tag add a `width` property with a value of 442, a `height` property with a value of 100% and a `y` property with a value of 0 (zero).
20. Between the `Accordion` tags add an instance of the `Amenities` component from the `comp` namespace and add a `label` property with a value of `AMENITIES`.

```
<comp:Amenities label="AMENITIES" />
```

21. After the `Amenities` component, add an instance of the `Rooms` component in the `comp` namespace.
22. Add a `label` property having a value of `ROOMS`, and a `roomList` property bound to the `roomList` object.

```
<comp:Rooms label="ROOMS"
    roomList="{roomList}"/>
```

Your code should appear as follows:

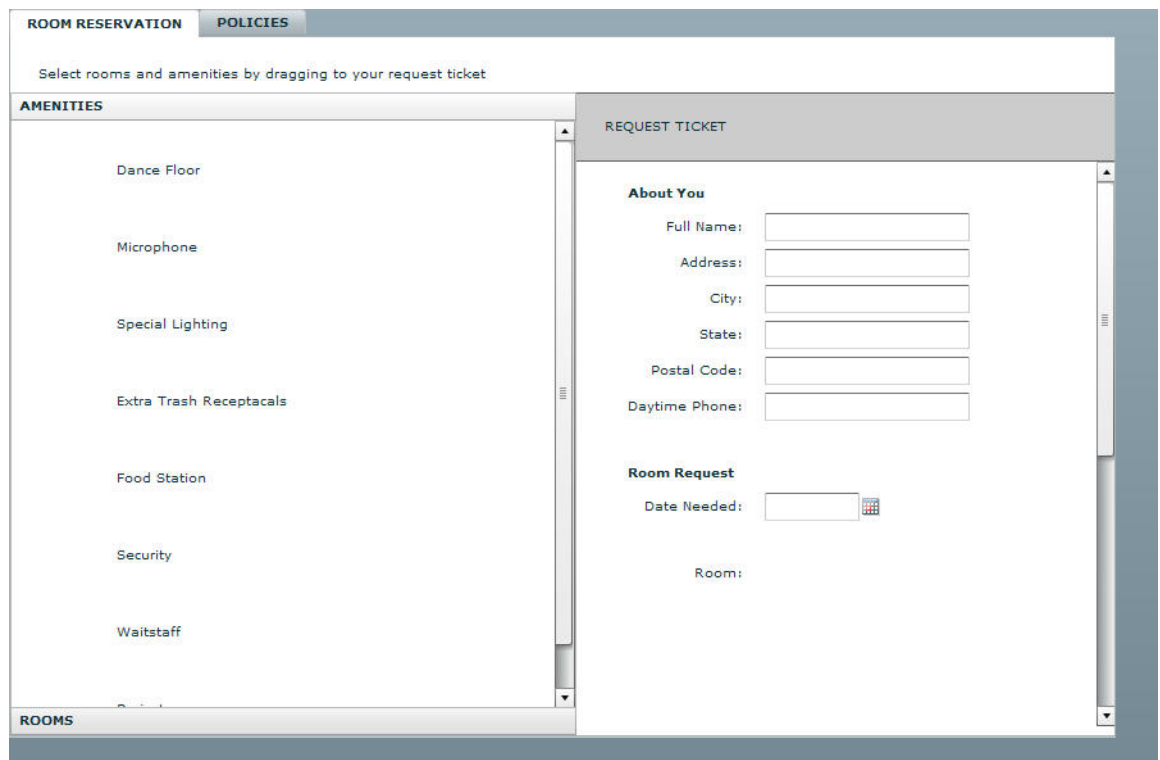
```
<mx:Canvas y="33" height="100%">
  <mx:Accordion width="442" height="100%"
    y="0">
    <comp:Amenities label="AMENITIES" />
    <comp:Rooms label="ROOMS"
      roomList="{roomList}" />
  </mx:Accordion>
  <comp:RequestTicket id="rt" width="422"
    y="0" x="442"
    dragDrop="doDragDrop(event)"
    dragEnter="doDragEnter(event)" />
</mx:Canvas>
```

23. Save the file and run the application.

You should see the left side of the **Room Reservations** tab now has an accordion. The **Amenities** pleat is open.

24. Click the **Rooms** pleat.

You will see an empty space that will be populated in a later exercise.



Embed a Font

25. Download the **Justus** font from <http://www.dafont.com/justus.font>.
26. If you have a Windows machine, install the fonts into your **Fonts** folder usually located here: C:/WINDOWS/Fonts/. If you have a MAC OS, install the fonts into the **/Library/Fonts** folder.
27. After you install the fonts, close Flex Builder and open it again so that the fonts become available.
28. Return to **ReservationSystem.mxml**.
29. After the beginning `Application` tag, create a `Style` tag set.
30. Create a font description using the `@font-face` property.

```
<mx:Style>
    @font-face {
    }
</mx:Style>
```

31. Add the following attributes and values to the font description:

- `src: local("Justus")`
- `fontFamily: "fontJustusRoman"`
- `unicodeRange: U+0061-U+007A, U+0043`

```
@font-face {
    src: local("Justus");
    fontFamily: "fontJustusRoman";
    unicodeRange: U+0061-U+007A, U+0043;
}
```

32. Within the `Style` block after the last font description, create another font description using `@font-face` with the following attributes and values:

- `src: local("Justus")`
- `fontFamily: "fontJustusBold"`
- `font-weight: bold`
- `unicodeRange: U+0058-U+005A`

```
@font-face {
    src: local("Justus");
    fontFamily: "fontJustusBold";
    font-weight: bold;
    unicodeRange: U+0058-U+005A;
}
```

Your code should appear as follows:

```
<mx:Style>
    @font-face{
        src: local("Justus");
        fontFamily: "fontJustusRoman";
        unicodeRange: U+0061-U+007A, U+0043;
    }

    @font-face{
        src: local("Justus");
        fontFamily: "fontJustusBold";
        font-weight: bold;
        unicodeRange: U+0058-U+005A;
    }
</mx:Style>
```

33. Save the file.

34. From the **assets** folder, open **styles.css**.

Notice that there are three selectors already created for you. You will use those selectors later.

35. Create a `XYZLogoFont` class selector with the following attributes and values:

- `fontFamily: "fontJustusRoman";`
- `fontSize: 22pt;`
- `color: #363636;`

36. Create a `XYZLogoFontBold` class selector with the following attributes and values:

- `fontFamily: "fontJustusBold";`
- `fontSize: 22pt;`
- `fontWeight: bold;`
- `color: #363636;`

Your code should appear as follows:

```
.XYZLogoFont {
```

```

        fontFamily: "fontJustusRoman";
        fontSize: 22pt;
        color: #363636;
    }
    .XYZLogoFontBold {
        fontFamily: "fontJustusBold";
        fontSize: 22pt;
        fontWeight: bold;
        color: #363636;
    }

```

37. Save the file.

38. Return to **ReservationSystem.mxml**.

After the `Style` block, add another `Style` tag with a `source` property set to a value of `assets/styles.css`.

```
<mx:Style source="assets/styles.css"/>
```

39. Save the file.

40. From the **components** folder, open the **Footer.mxml** file.

41. After the `HRule` component, create a `Label` component with a `text` property set to a value of `XYZ`, a `styleName` property set to a value of `XYZLogoFontBold`, and an `x` property set to a value of `20`.

```
<mx:Label text="XYZ"
          styleName="XYZLogoFontBold"
          x="20" />
```

42. After the `Label` component add another `Label` component with a `text` property set to a value of `Company`, a `styleName` property set to a value of `XYZLogoFont`, and an `x` property set to a value of `75`.

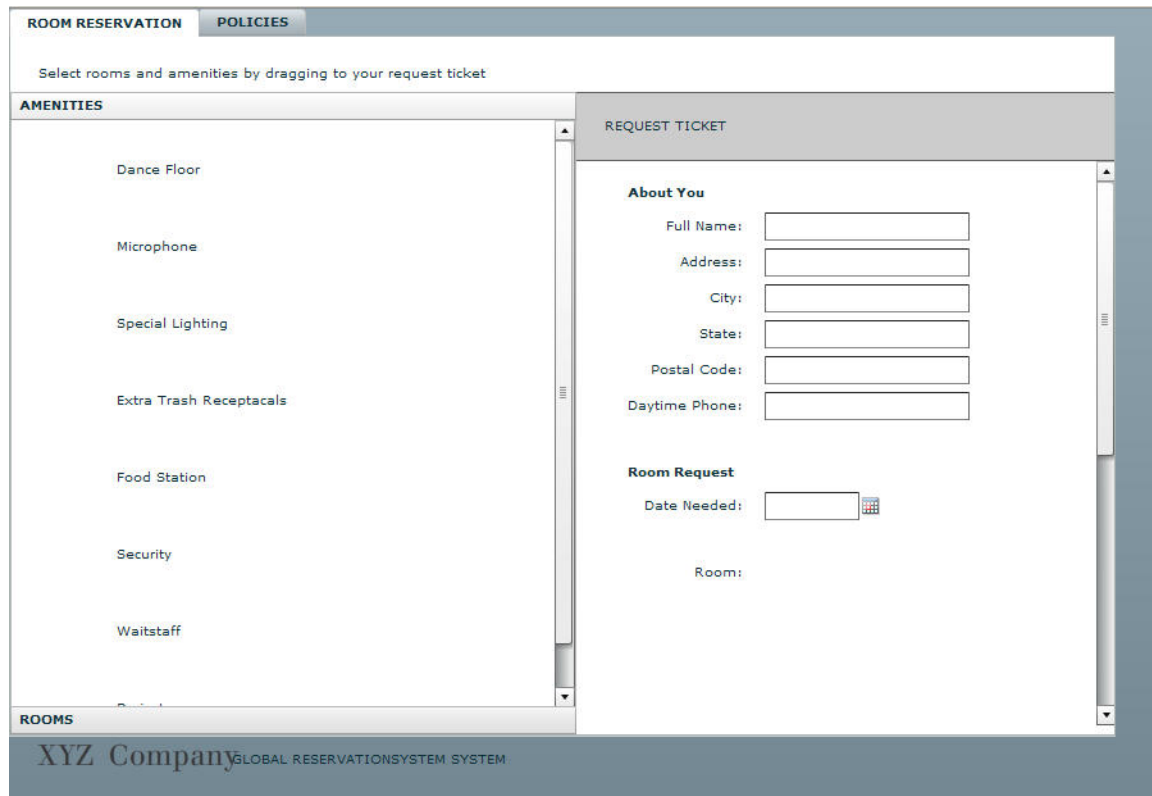
```
<mx:Label text="Company"
          styleName="XYZLogoFont"
          x="75" />
```

43. Add another `Label` component with a `text` property of `GLOBAL RESERVATION SYSTEM`, an `x` property of `172` and a `y` property of `10`.

```
<mx:Label text="GLOBAL RESERVATION SYSTEM"
          x="171" y="10"/>
```

44. Save the file and run the application.

You should see the formatted XYZ Company text displayed below the **Room Reservation**. Note that this text runs into the GLOBAL RESERVATION SYSTEM text. This area will be formatted further in the next exercise.



Embed an Image

45. From the **components** folder open **Amenities.mxml**.

46. After the beginning Canvas tag, create an Image tag.

47. Add an id property set to a value of image1 and a source property set to a value of

```
@Embed(source='assets/ReservationSystem_Icon.swf',  
symbol='Dancing').
```

48. Add a x property set to a value of 20 and a y property set to a value of 20.

Your code should appear as follows:

```
<mx:Image id="image1"  
          source="@Embed(source='assets/Reservations_Icons.s  
wf', symbol='Dancing')"
```

```
x="20" y="20" />
```

49. Save the file and run the application.

You should see a dance icon to the left of the Dance Floor amenity.

50. Return to **Amenities.mxml** and uncomment the group of images at the bottom of the file.

51. Save the file and run the application.

You should see all the images in the **Amenities** accordion.

ROOM RESERVATION POLICIES

Select rooms and amenities by dragging to your request ticket

AMENITIES

- Dance Floor
- Microphone
- Special Lighting
- Extra Trash Receptacals
- Food Station
- Security
- Waitstaff

REQUEST TICKET

About You

Full Name:

Address:

City:

State:

Postal Code:

Daytime Phone:

Room Request

Date Needed:

Room:

XYZ Company GLOBAL RESERVATION SYSTEM SYSTEM