

Introduction to Mobile App Development

MODULE 4: Unit Converter App

THOMPSON RIVERS UNIVERSITY | COMPUTING SCIENCE

Module 4

1. Design Concepts
2. Corona Widget Library - Switches
3. An Implementation Example of the Unit Converter App

Design Concepts

3

Design Concepts

Examples of User Interface Options



4

Corona Widget Library

5

Corona Widget Library

On-Off Switch

```
local widget = require( "widget" )

-- Handle press events for the On-Off Switch
local function onSwitchPress( event )
    local switch = event.target
    print( "Switch with ID "..switch.id.." is on: "..tostring(switch.isOn) )
end

-- Create the widget
local onOffSwitch = widget.newSwitch(
{
    x = display.contentCenterX,
    y = display.contentCenterY,
    style = "onOff",
    id = "onOffSwitch",
    onPress = onSwitchPress
}
)
```

<https://docs.coronalabs.com/api/library/widget/newSwitch.html>

6

Corona Widget Library

Radio Buttons

```
local widget = require( "widget" )

-- Handle press events for the buttons
local function onSwitchPress( event )
    local switch = event.target
    print( "Switch with ID '"..switch.id.." is on: "..tostring(switch.isOn) )
end

-- Create a group for the radio button set
local radioGroup = display.newGroup()

-- Create two associated radio buttons (inserted into the same display group)
local radioButton1 = widget.newSwitch(
{
    x = 100,
    y = 100,
    style = "radio",
    id = "RadioButton1",
    initialSwitchState = true,
    onPress = onSwitchPress
}
)
radioGroup:insert( radioButton1 )

local radioButton2 = widget.newSwitch(
{
    x = 100,
    y = 200,
    style = "radio",
    id = "RadioButton2",
    onPress = onSwitchPress
}
)
radioGroup:insert( radioButton2 )
```

<https://docs.coronalabs.com/api/library/widget/newSwitch.html> 7

Corona Widget Library

Check boxes

```
local widget = require( "widget" )

-- Handle press events for the checkbox
local function onSwitchPress( event )
    local switch = event.target
    print( "Switch with ID '"..switch.id.." is on: "..tostring(switch.isOn) )
end

-- Create the widget
local checkboxButton = widget.newSwitch(
{
    x = display.contentCenterX,
    y = display.contentCenterY,
    style = "checkbox",
    id = "Checkbox",
    onPress = onSwitchPress
}
)
```

<https://docs.coronalabs.com/api/library/widget/newSwitch.html> 8

An Implementation Example of the Unit Converter App

9

Unit Converter App

Design Steps

- 1- Design a User Interface
- 2- Prepare your Art Work / Add a Background
- 3- Add Buttons/Switches/Textfields/etc.
- 4- Test your code
- 5- Discuss the limitations of Corona SDK

10

Unit Converter App

Design Steps: User Interface



11

Unit Converter App

Design Steps: Background

```
local widget = require( "widget" )

local image = display.newImageRect( "background.jpg",
    display.contentWidth, display.contentHeight)
image.x = display.contentCenterX
image.y = display.contentCenterY
```

12

Unit Converter App

Design Steps: Add Buttons/Switches/Textfields/etc.

```
local widget = require( "widget" )

local image = display.newImageRect( "background.jpg",
    display.contentWidth, display.contentHeight)
image.x = display.contentCenterX
image.y = display.contentCenterY

local counter = 0
local mylabel = display.newText( "Unit Converter", display.contentCenterX, display.contentCenterY-180,
    native.systemFont, 30 )

local myText = display.newText( "Original Text!", display.contentCenterX, display.contentCenterY+160,
    native.systemFont, 30 )

local inputText = native.newTextField( display.contentCenterX, display.contentCenterY-100, 300, 30 )
inputText.inputType = "number"

myText:setFillColor( 1, 1, 0.5)
```

13

Unit Converter App

Design Steps: Add Buttons/Switches/Textfields/etc.

```
-- Function to handle button events
local function handleButtonEvent1( event )

    if ( "ended" == event.phase ) then
        if (inputText.text == "") then
            myText.text = "Please Enter a Value"
            myText:setFillColor( 1, 0, 0.5)
        else
            num = inputText.text
            myText.text = num*39.3701 .. " inch"
            myText:setFillColor( 1, 1, 0.5)
        end
    end
end

-- Function to handle button events
local function handleButtonEvent2( event )

    if ( "ended" == event.phase ) then
        if (inputText.text == "") then
            myText.text = "Please Enter a Value"
            myText:setFillColor( 1, 0, 0.5)
        else
            num = inputText.text
            myText.text = num/39.3701 .. " meter"
            myText:setFillColor( 1, 1, 0.5)
        end
    end
end
```

14

Unit Converter App

Design Steps: Add Buttons/Switches/Textfields/etc.

```
clickbutton_1 = widget.newButton{
    id = "clickbutton1",
    --label = "Click",
    labelColor = { default={ 1, 1, 1 }, over={ 0, 0, 0, 0.5 }},
    emboss=true,
    width = 300,
    height = 60,
    fontSize = 30,
    defaultFile = "but1.png",
    overFile = "but1_over.png",
    onEvent = handleButtonEvent1
}
-- Center the button
clickbutton_1.x = display.contentCenterX
clickbutton_1.y = display.contentCenterY-20

clickbutton_2 = widget.newButton{
    id = "clickbutton2",
    --label = "Click",
    labelColor = { default={ 1, 1, 1 }, over={ 0, 0, 0, 0.5 }},
    emboss=true,
    width = 300,
    height = 60,
    fontSize = 30,
    defaultFile = "but2.png",
    overFile = "but2_over.png",
    onEvent = handleButtonEvent2
}
-- Center the button
clickbutton_2.x = display.contentCenterX
clickbutton_2.y = display.contentCenterY+60
```

15

Unit Converter App

Corona Limitations



16



End of Module 4