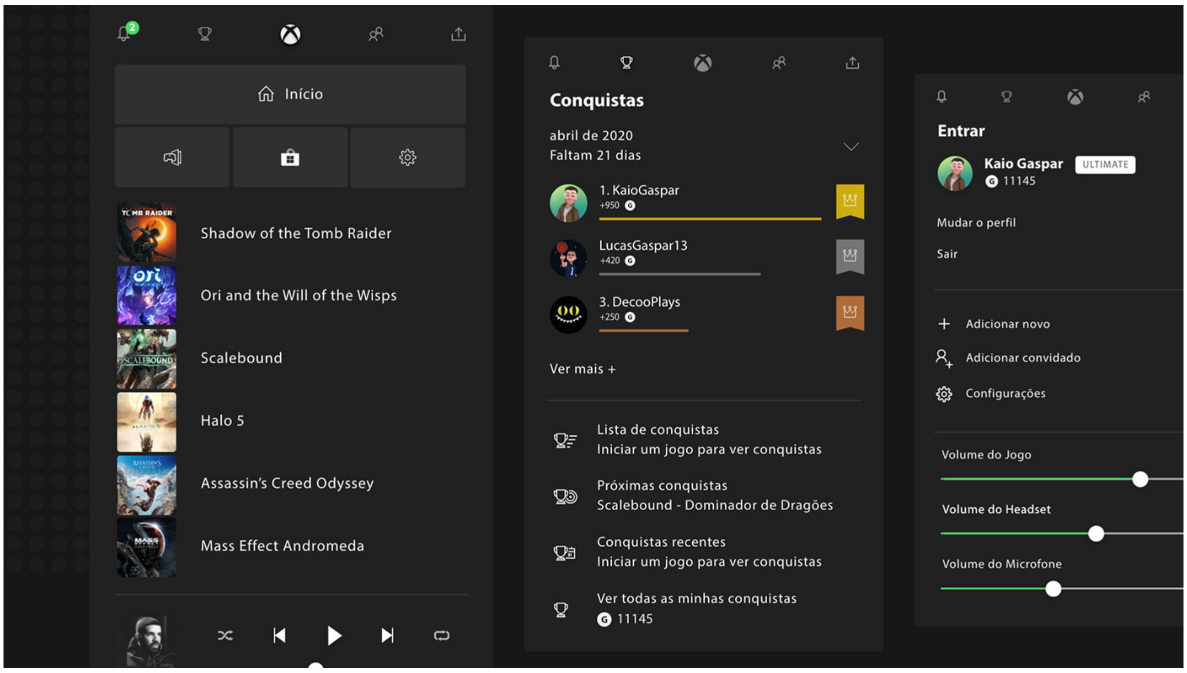
Layouts



Layouts is one of the most important concepts in any Graphical User Interface (GUI) designing.

An application's User Interface may consist of several screens or pages. The Lay out of each screen is what is commonly referred to as the "layout", so layout is self explanatory. But what you may not know is, that certain parts of the screen may change but not those directly inside a layout. View allows positions of an item inside of it to change, but layouts don't, that is to say that there is another UI concept and component called views which will be discussed later.

So, Layouts are static whereas Views are dynamic. Views reside inside layouts, and also views may lay out themselves using layouts. So knowledge of layouts is indispensable.

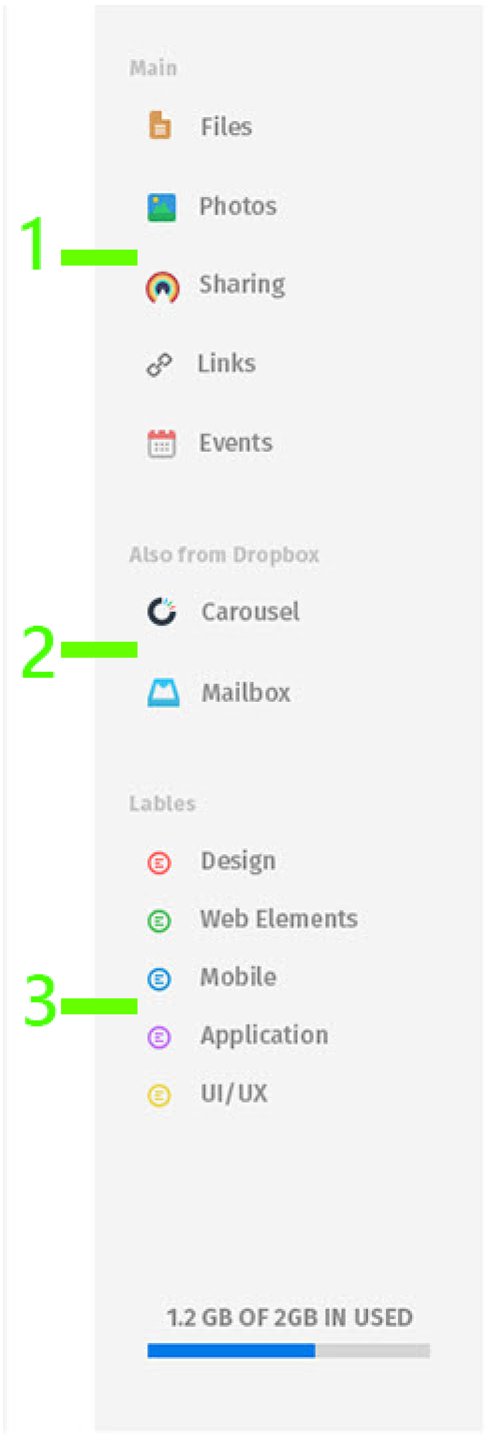
Qml has four(4) layout types, you can combine then to create whatever layout you want. The types are RowLayout (row layout), ColumnLayout (column layout), GridLayout (grid layout), and StackLayout (stack layout).

ColumnLayout

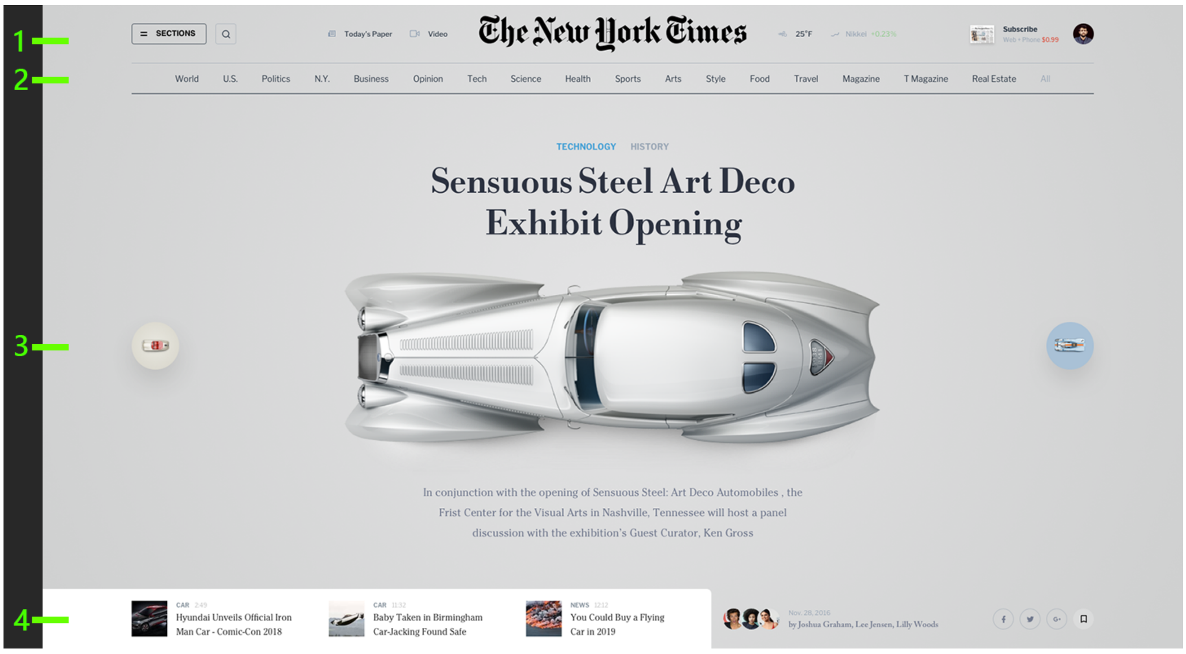
ColumnLayout - arranges its contents so that we get a column, but also each of the items in there is a row inside the column. So we have something like shown in the images below.

The numbers shows the direct contents of the Column layout. These are rows inside a column. Together they form a column.

App sample 1


Main
Files
Photos
1
Sharing
Links
Events
Also from Dropbox
Carousel
2
Mailbox
Lables
Design
Web Elements
Mobile
3
Application
UI/UX
1.2 GB OF 2GB IN USED

App sample 2



Sensuous Steel Art Deco Exhibit Opening

In conjunction with the opening of Sensuous Steel: Art Deco Automobiles , the Frist Center for the Visual Arts in Nashville, Tennessee will host a panel discussion with the exhibition's Guest Curator, Ken Gross

CAR 2:43 Hyundai Unveils Official Iron Man Car - Comic-Con 2018

CAR 11:32 Baby Taken in Birmingham Car-Jacking Found Safe

NEWS 12:12 You Could Buy a Flying Car in 2019

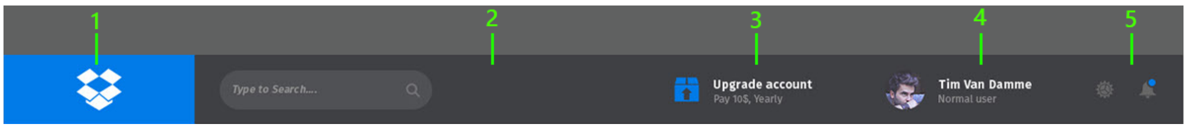
Nov. 28, 2016 by Joshua Graham, Lee Jensen, Lilly Woods

RowLayout

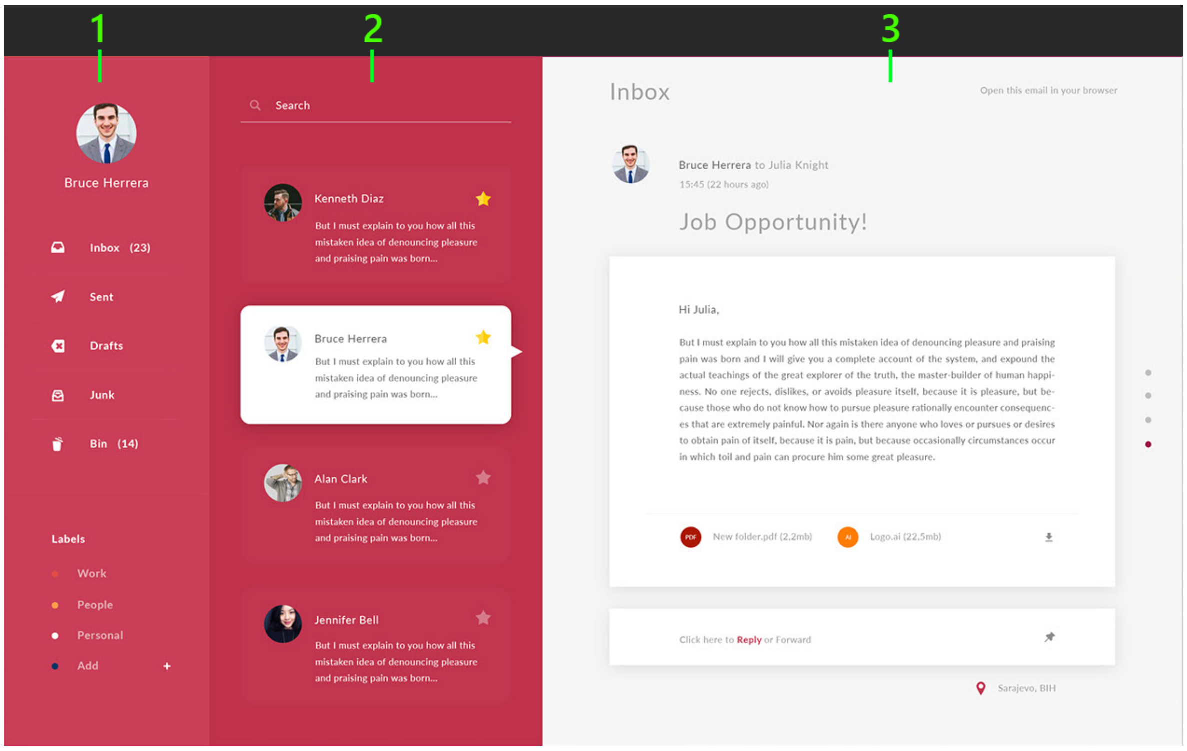
RowLayout - arranges its contents so that we get a row, but each of the items in there is a column inside the row. So we have something like shown in the images below.

The numbers shows the direct contents of the Row layout. These are columns inside a row. Together they form a row.

App sample 1



App sample 2



Kenneth Diaz
But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born...

Bruce Herrera
But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born...

Alan Clark
But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born...

Jennifer Bell
But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born...

Bruce Herrera to Julia Knight
15:45 (22 hours ago)

Job Opportunity!

Hi Julia,

But I must explain to you how all this mistaken idea of denouncing pleasure and praising pain was born and I will give you a complete account of the system, and expound the actual teachings of the great explorer of the truth, the master-builder of human happiness. No one rejects, dislikes, or avoids pleasure itself, because it is pleasure, but because those who do not know how to pursue pleasure rationally encounter consequences that are extremely painful. Nor again is there anyone who loves or pursues or desires to obtain pain of itself, because it is pain, but because occasionally circumstances occur in which toil and pain can procure him some great pleasure.

New folder.pdf (2,2mb) Logo.ai (22,5mb)

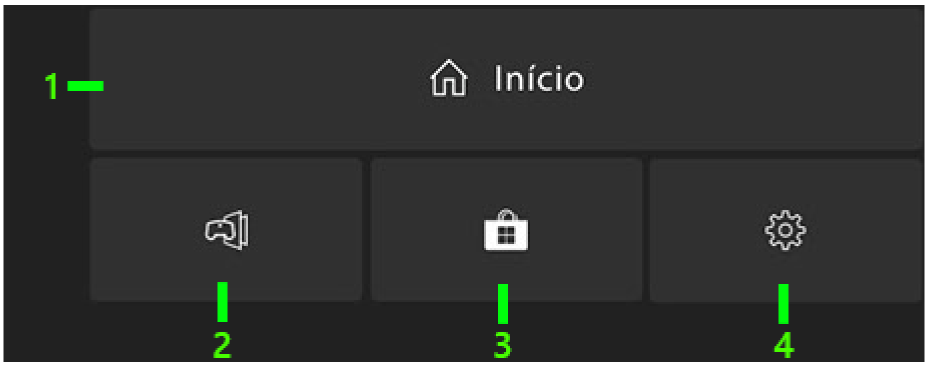
Click here to Reply or Forward

Sarajevo, BIH

GridLayout

GridLayout - arranges its contents so that we get a grid, but each of the items in there is a cell inside the grid. When a grid layout is implemented. The number of rows and/or columns are indicated. Which means that the minimum width and/or height of each cell is indicated, so if a cell wants to be larger than the rest it can just say so and take up space belonging to other cells. So we have something like shown in the images below.

The numbers shows the direct contents of the Grid layout. These are cells inside the grid. Together they form a grid.



StackLayout

StackLayout - arranges its contents so that only one is visible at any particular time, the rest are hidden, behind it, so to speak. Each of the items is called a stack item.

Import layouts

To start using layouts in any qml file, you will need it imported in that qml file.

```
import QtQuick.Layouts 1.3
```

The digit after the dot corresponds with you QtQuick import. eg:

If you import *QtQuick 2.12*, you should import *QtQuick.Layouts 1.12,* for *QtQuick 2.13*, you import *QtQuick.Layouts 1.13*, and so on. Going back to *QtQuick 2.11*, *QtQuick.Layouts 1.4* is used and not (*QtQuick.Layouts 1.11*), from *QtQuick 2.10*, *QtQuick.Layouts 1.3* is used and so on.

So, the top most of your qml file looks like this at the least, if you want to work with layouts.

```
import QtQuick 2.12
import QtQuick.Layouts 1.12
```

From now on you can start declaring qml layout type, like we have explained about above.

Layout

Layout type exists only to provide properties to use when defining your layouts. Actually all the other layout types inherent from this basic type. You will not use it directly. When studying the other types you will see it being declared as

```
Layout.fillWidth: true
```

NB: Qml object types always start with an uppercase letter.

ColumnLayout

ColumnLayout description

RowLayout

RowLayout description

GridLayout

GridLayout description

StackLayout

StackLayout description