# UIKit lib basic Integration

To support backward compatibility of different UI components, UIKitLIb uses appcompat lib. So the base activity always extends from AppCompatActivity.

Documentation: Open index.html file from JavaDoc.zip and search for MainActivity. MainActivity provides all the components Javadoc link.

## Steps to integrate:

Before using any feature, the lib must be included in the project.

The recommended way is to have artifactory dependency pointing to uikitlib.

compile 'com.philips.cdp:uikitLib:1.1.0'   (works in Philips network)

Otherwise add below dependency in your app build.gradle.

compile **'com.android.support:support-annotations:22.2.0'**compile **'com.android.support:appcompat-v7:23.1.1'**compile **'uk.co.chrisjenx:calligraphy:2.1.0'**compile **'com.android.support:design:23.1.1'**compile **'com.shamanland:fonticon:0.1.8'**compile **'com.android.support:cardview-v7:+'**compile **'com.android.support:recyclerview-v7:+'**

### Choose theme: Lib support two variants of themes (Solid and gradient version).

Various themes can be accessed with Theme.Philips.\* and Theme.Philips.\*.Gradient

Example: Blue theme Solid-> **Theme.Philips.DarkBlue**

Gradient-> **Theme.Philips.DarkBlue.Gradient**

### Extend from UIKitActivity-> UIKitActivity provides some common customizations to achieve some goals like having common fonts across all activities.

Example: **public class** MyActivity **extends** UiKitActivity {}

### Using custom fonts:

UIKit uses open source lib [Calligraphy](https://github.com/chrisjenx/Calligraphy) for font customization.

Define fontpath=”fonts/myfont.ttf” in xml for text view to use custom font.

### VectorDrawable Support: Custom <vector> tag is recognized only on or after Lollipop.

But still it can be used via programmatically.

Example: -> Drawable d =

com.philips.cdp.uikit.drawable.VectorDrawable.create(context, R.drawable.xyz);

imageView.setImageDrawable(d);