Alloy Getting Started

Overview

This guide provides basic instructions on creating a quick Alloy project. Alloy is an MVC framework to help you quickly build Titanium applications.

Command-line interface installation

To create and manage Alloy applications, you need the Alloy command-line interface. If you already installed Appcelerator Studio or the Appcelerator CLI, the Alloy CLI will be automatically installed for you. If you need to install Appcelerator Studio, see the Titanium SDK Getting Started or if you need to manually install the Appcelerator CLI:

1. Download and install Node.js from http://nodejs.org/#download, which includes the npm package manager needed to install the Appcelerator CLI.
2. From a terminal, run the following command to install the CLI:

```
[sudo] npm install appcelerator -g
```

3. After installation, run the setup command to install required components:

```
appc setup
```

After the CLI downloads and installs the required components, you will be prompted to login and authorize your computer. The CLI will ask to send an authorization token to your e-mail account or as a text to your mobile phone. Enter the authorization token once you receive it to complete the registration process.

Create a project

Using Studio

To create a new Alloy project, start Appcelerator Studio, then

1. From the menu, select **File > New > Mobile App Project**. The **New Mobile App Project** wizard appears.
2. Select **Alloy** in the **Available Templates** box, choose a template, then click the **Next** button.
3. Complete all of the fields, then click the **Finish** button.

A new skeleton Alloy project will be generated. If you are familiar with classic Titanium development, the `Resources` folder is hidden from the App and Project Explorer. All work for an Alloy project is done in the `app` directory.

Using the CLI

To create a new Alloy project, use the Appcelerator CLI to create a new Alloy project. Run the following command in a terminal:

```
appc new -t titanium
```

You will be prompted to enter an application name and application ID.

A new skeleton Alloy project will be generated in the directory named after the application name.

Simple example

The following example converts the `image_view_file.js` file from the Titanium KitchenSink sample application to an Alloy project.

To see the example in the KitchenSink application, click on the `Base UI` tab, then navigate to **Views > Images Views > Image File**.

View

The view file declares the structure of the GUI. For example, the file below defines a window with an image and label.

Replace the contents of `app/views/index.xml` with the following:

```
<Alloy>
 <Window>
  <ImageView id="imageView" onClick="clickImage"/>
  <Label id="l">Click Image of Apple Logo</Label>
 </Window>
</Alloy>
```

Style

The style file formats and styles the components in the view file. For example, the style below defines the background color of the window; position, dimensions and color of the label; and position, dimensions and location of the image.

Replace the contents of `app/styles/index.tss` with the following:

```
"Window": {
 backgroundColor:"white"
},
"#l":{
 bottom:20,
 width: Ti.UI.SIZE,
 height: Ti.UI.SIZE,
 color:'#999'
},
"#imageView":{
 image:"/images/apple_logo.jpg",
 width:24,
 height:24,
 top:100
}
```

Controller

The controller file contains the presentation logic, which responds to input from the user. For example, the controller code below creates an alert dialog when the user clicks on the image and opens the window when the application starts.

Replace the contents of `app/controllers/index.js` with the following:

```
function clickImage(e) {
 Titanium.UI.createAlertDialog({title:'Image View', message:'You clicked
me!'}).show();
}

$.index.open();
```

Asset

Create a folder called `app/assets/images` and copy the `apple_logo.jpg` file from the KitchenSink project. This file will be copied to `Resources/images/apple_logo.jpg` by Alloy during the build process.

Compile and run

The Alloy CLI converts the view, style and controller in to a Titanium project, which can be built and ran by Studio or the Appcelerator CLI.

Using Studio

In the **Project Explorer** view, select your project, then from the global tool bar, select **Run** from the **Launch Mode** drop-down and select the device to launch the application on in the **Target** drop-down. Alloy will generate the Titanium files from the Alloy project files, which will then be compiled by Studio and launched on the device simulator.

Using the CLI

From a console window, go to the root directory of the project, then

```
appc run [-p platform]
```

The Appcelerator CLI contains hooks to the Alloy CLI, so you do not need to run any Alloy commands.

Refer to the Appcelerator Command-Line Interface Reference for more information about using the `appc run` command.

More examples

For more examples of Alloy, see Alloy Samples.

Next steps

Review Alloy Concepts to learn more about Alloy and how to structure your project. From there, visit the links on Alloy Views, Alloy Controllers, and Alloy Models to learn how to write views, controllers and models, respectively.