A Guide to App Studio 9.5.1

AP

Quark
App Publishing
Studio

Contents

Note for users of App Studio 9.2 and 9.3

Beginning with version 9.5, App Studio switched from the proprietary AVE format to the industry-standard HTML5 format.

If you want to continue creating AVE issues and AVE apps, you can do so. Note, however, that the following changes have been made:

- The **App Studio** palette has been renamed to the **Quark AVE Interactivity** palette.
- The AVE App Studio Publishing Portal has been renamed and moved to *http://avepublishing.quark.com*.
- App Studio Issue Previewer has been renamed to Quark AVE Previewer.
- App Studio Factory has been renamed to Quark AVE AppFactory.

If you keep these changes in mind, you can continue to use *A Guide to App Studio 9.3* to create AVE apps and issues.

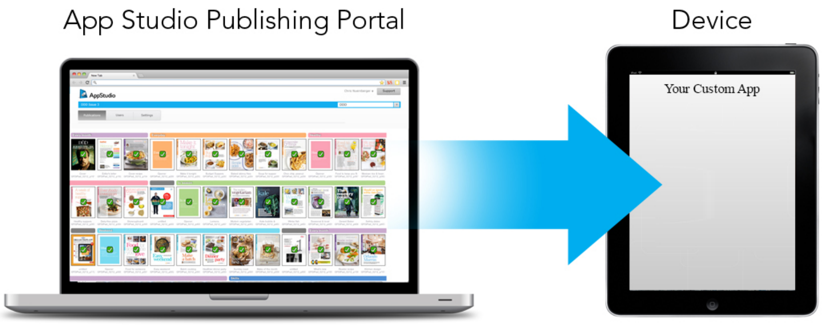
➡ AVE apps can only display AVE issues, and App Studio 9.5 and 9.5.1 apps can only display HTML5 App Studio issues.

Understanding App Studio

The App Studio feature lets you create a customized app for the iPad®, iPhone®, or Android device; distribute that app through the Apple® App Store or an Android app store; and then create and publish content that your customers can purchase and download from within the app.

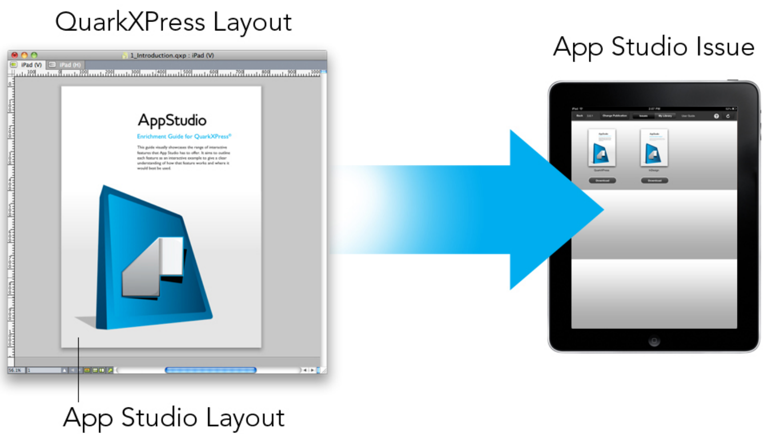
There are two parts to any App Studio solution:

- An *App Studio app* is an app that you build with the App Studio Publishing Portal. After you build and test your App Studio app, you can submit it to the Apple App Store or an Android app store. If it is approved, it will be made available to your customers.



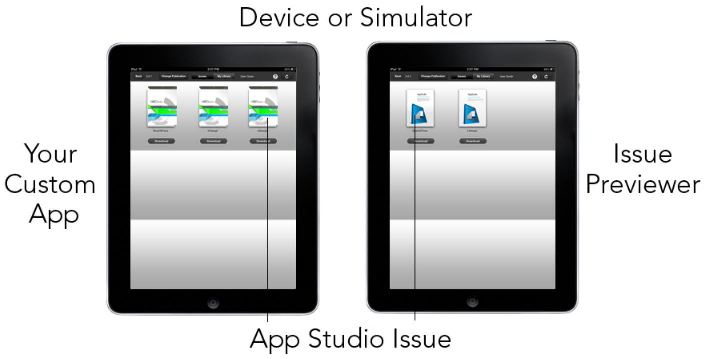
You can use the App Studio Publishing Portal to create your own custom apps.

- An *App Studio issue* is the digital equivalent of a book or a magazine. You can create and export App Studio issues in QuarkXPress.



You can use QuarkXPress to create App Studio issues.

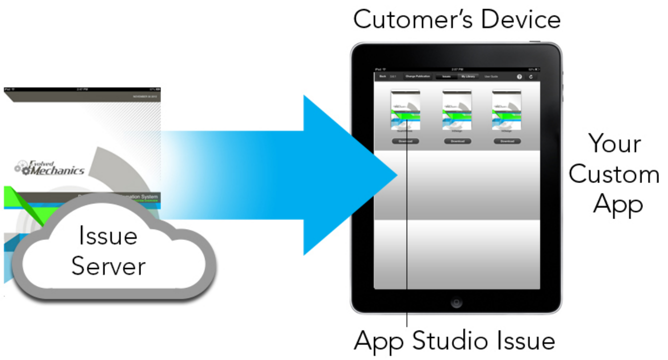
Your customers will use your App Studio app to view your App Studio issues. You and your customers can also view and test App Studio issues in a free Previewer iPad app that is available from the Apple App Store.



You can view App Studio issues in your own custom app or in a Previewer app.

Once you've created your app and your issues, you can make the issues available on a Web server, set up in-app purchases for each of your issues on the Apple developer Web site or on an Android app store, and then submit your app to the app store.

Once your app and its content is approved, your customers will be able to download your App Studio app from the app store and use it to browse, purchase, and read your App Studio issues. And you don't even have to set up your own eCommerce solution — the money from issue purchases goes automatically into your account through the in-app purchase feature.



Your customers can buy your issues from within your custom app. When they do, the issues download to their device from a Web server.

Understanding the App Studio format

The App Studio format lets you create issues using HTML5 and JavaScript. Because HTML5 is a platform-independent format, you can read App Studio issues in any Web browser that has a compatible HTML5 renderer.

You can use two types of apps to view App Studio issues:

- App Studio Issue Previewer. This free app is designed to let you view and test App Studio issues on a physical device.
- A custom App Studio app. For example, if you are publishing a magazine, you might have an app that has been built specifically to deliver that magazine.

Each App Studio issue is made up of one or more *articles*. Each article corresponds to a single QuarkXPress project file and may consist of one App Studio layout (if the article supports only one orientation) or two App Studio layouts (if the article supports dual orientations). An article can be one or more pages long, and does not have to represent a discrete unit of content.

➡ App Studio articles are not the same thing as QuarkCopyDesk articles.

Exported App Studio issues are hosted on the App Studio Publishing Portal. Because they are in HTML5 content, you can view them in a Web browser by logging in to the App Studio Publishing Portal. There, you can navigate to your issues, display their component articles, and view individual pages in any orientation. For more information, see "*Understanding the App Studio Publishing Portal*."

Understanding the App Studio Publishing Portal

On the App Studio Publishing Portal (*http://my.appstudio.net*), content is organized into organizations, publications, issues, and articles.

- *Organization:* Represents the organization that is responsible for one or more related publications. When you create an account on the App Studio Publishing Portal, one of the first things you do is create an organization.
- *Publication:* Represents a set of related issues. For example, a publication might represent a particular magazine title.
- *Issue:* Represents a discrete unit of content to be downloaded as a whole. For example, an issue might represent a single issue of a magazine title.
- *Article:* Issues are divided into one or more articles, each of which is created from a single QuarkXPress project. Articles also allow issue creators to break issues down into multiple QuarkXPress projects. Each article displays as an entry in the automatically generated table of contents.

The content of your issues is stored on the App Studio Portal, which serves it to your App Studio apps. For more information about creating App Studio issues, see "*Creating an App Studio issue*."

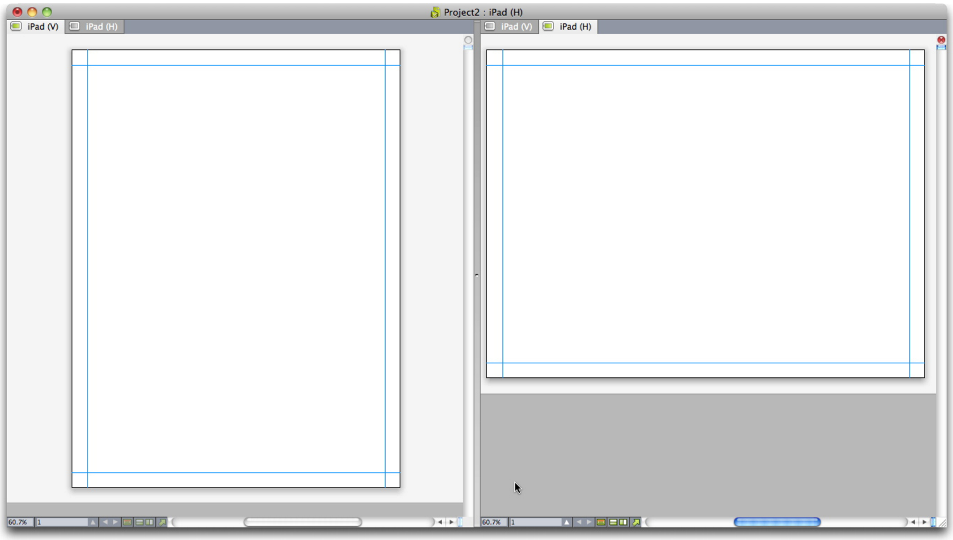
The App Studio Publishing Portal also provides a way for you to create your own App Studio app. This is basically a matter of supplying descriptive information about your app and uploading some graphic resources. For more information, see "*Creating an App Studio app*."

Understanding App Studio apps

To create an App Studio app, you simply enter some descriptive information and upload your developer credentials. The Publishing Portal builds two versions of the app: a build for testing on your device, and a build for submission to the App Store. For more information, see "*Creating an App Studio app*."

Understanding layout families

A QuarkXPress project file can contain multiple layouts (for more information, see "Projects and layouts" in *A Guide to QuarkXPress*). When you create an App Studio layout, QuarkXPress creates a separate layout for each device/orientation combination you checked in the **New Project** dialog box. All of these layouts are members of a single *layout family*.



A layout family displayed in split view. Each tab across the top represents a layout that is part of the layout family.

A layout family is a group of layouts that represents a single App Studio article. One layout family might include a horizontal layout for the iPad, a vertical layout for a different type of tablet, and both vertical and horizontal layouts for a third type of tablet. All of the layouts in a layout family are bound to one another in terms of page count; if you add or delete a page in one of a layout family's layouts, that page is added or deleted from *all* of the layouts in the layout family.

Layout families make it easy for you to keep content synchronized between the various editions of each page of content. For example, assume that you're working on a 30-page issue, and you discover that you need to add a new page after page 12. You can go to any of the layouts in the layout family and add the new page, and the page is automatically inserted in every other layout in the layout family. This is much easier than manually adding the new page to each layout one-at-a-time.

When you create a layout family, QuarkXPress displays the project in a split view. You don't have to work on the layout in split view, but it can be helpful when you're trying to keep different editions of a page consistent.

Another helpful feature for App Studio issues is the shared content feature. This feature lets you put copies of a picture or a chunk of text into different layouts, and then automatically keep the copies synchronized. So, for example, if you correct a name on page five of one layout, the name can be automatically corrected in all of the other layouts too. For more information, see "Working with shared content" in *A Guide to QuarkXPress* and "*Synchronizing content between orientations*."

Creating an App Studio issue

App Studio issues are stored on the App Studio Publishing Portal. Before you can start creating content, you must create a free account on the Publishing Portal and set up an organization, a publication, and an issue for that account. Once you have created an issue, you can begin adding articles to it. To create an App Studio issue:

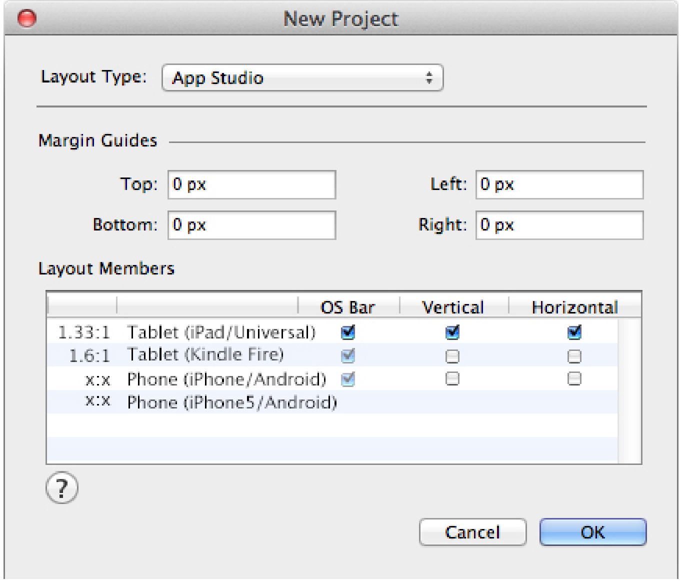
1. Go to *http://my.appstudio.net*.
2. If you don't have an account, create a free account.
3. Log in with your user name and password.
4. If you have not registered your organization, register it now.
5. If you have not yet created a publication for your issues, create it now.
6. Select the publication you want to add an issue to.
7. Click **Create New Issue**.

- In the **Title** field, enter a title for the issue. You can use this field to enter an issue number.
- In the **Subtitle** field, enter a subtitle for the issue. You cna use this field for whatever you want.
- In the **Deadline** field, enter a deadline. This field is optional.
- In the **Apple ID** field, enter the Apple ID for the in-app purchase (if any) that your customers can use to purchase the issue. (For more information, see "*Setting up in-app purchases*."
- In the **Description** field, enter a description for your issue.

8. Click **Create**. The issue is created on the server. You can now upload artices to this issue from within QuarkXPress. (For more information, see "*Exporting an App Studio article*.")

Creating an App Studio layout

To create an App Studio layout, choose **File** > **New Project** and select **App Studio** from the **Layout Type** drop-down menu.



New Project

Layout Type: App Studio

Margin Guides

Top: 0 px

Bottom: 0 px

Left: 0 px

Right: 0 px

Layout Members

		OS Bar	Vertical	Horizontal
1.33:1	Tablet (iPad/Universal)	☑	☑	☑
1.6:1	Tablet (Kindle Fire)	☑	□	□
x:x	Phone (iPhone/Android)	☑	□	□
x:x	Phone (iPhone5/Android)			

New Project dialog box for an App Studio layout

The options in the Layout Members field let you create separate layout families for a variety of different devices. For example, if you check **Vertical** and/or **Horizontal** for both **Tablet (iPad/Universal)** and **Phone (iPhone/Android)**, QuarkXPress will create a project with two layout families: One for tablets, and one for phones. (For more information about layout families, see "*Understanding layout families*.") This allows you to customize one version of the layout for tablet use, and another version for phone use.

You do not have to create layouts for more than one device. If you check **Tablet (iPad/Universal)**, the issue will be automatically scaled down for smaller devices, and the issue aspect ratio will be maintained. You should create a layout for a particular device only if you want to tailor the layout to fit that device precisely.

To specify which orientations you want to support, check **Vertical** and/or **Horizontal** for the desired devices in the **Layout Members** list.

The **OS Bar** check box controls whether the layout allows room for the status bar at the top of the screen. Check this box if you are planning to export in App Studio format, and uncheck it if you are planning to export the layout as an AVE issue.

When you click **OK**, QuarkXPress creates a project with a layout family that contains layouts for each device/orientation combination you checked. For more information, see "*Understanding layout families*."

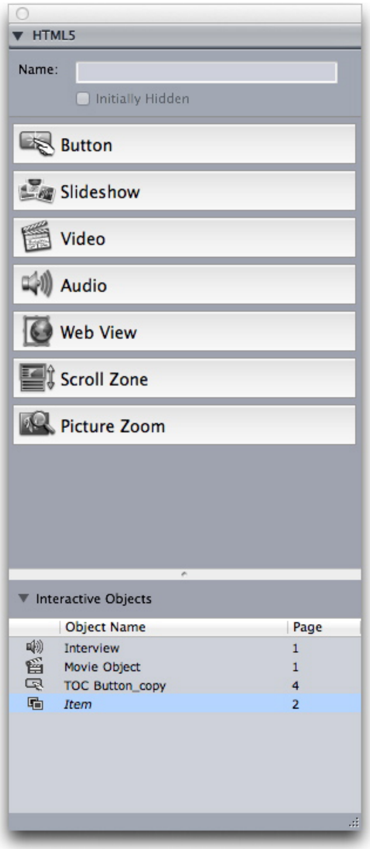
QuarkXPress displays the new project in a split view, so that you can work with multiple members of the layout family at the same time. For more information, see "Splitting a window" in *A Guide to QuarkXPress*.

By default, all new App Studio layouts include an RGB version of black named **Black (RGB)**. You should use this version of black instead of the default CMYK **Black** color to achieve the darkest black in the exported App Studio issue, especially for text.

By default, all App Studio layouts export with a white background. To create a different-colored background, draw a colored box that completely covers the page and then send it to the back. To avoid selecting the box as you work on the page, put the box on its own layer and then lock that layer.

Adding interactivity to an App Studio issue

You can add a variety of types of interactivity to an App Studio issue, including slideshows, movies, buttons, sound, and HTML with the **HTML5** palette (**Window** menu). This palette lets you name and add interactivity to whichever object is selected in the layout.



	Object Name	Page
	Interview	1
	Movie Object	1
	TOC Button_copy	4
	Item	2

HTML5 palette

Picture boxes, text boxes, anchored boxes, and no-content boxes support different kinds of interactivity. Options that are not available for the selected item are disabled.

At the bottom of the palette, there is a list of all of the interactive objects in the active layout, including each enrichment type, object's name, and page number. You can navigate to any object listed here by double-clicking it.

Once you apply interactivity to a box, the application adds an icon to the box to show what kind of interactivity it has. To view these icons, make sure **View > Visual Indicators** is checked. The icons are as follows:

- : Button
- : Slideshow
- : Video
- : Audio
- : Web View
- : Scroll Zone
- : Picture Zoom

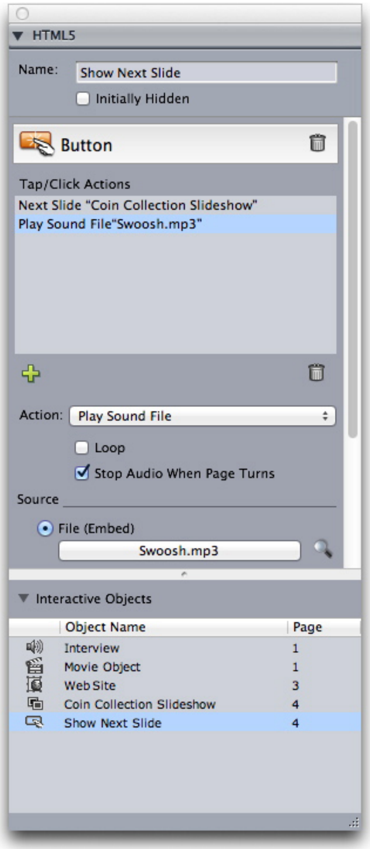
The **HTML5** palette includes a number of file selection controls. If you are logged into Quark Publishing Platform, you can choose whether to select a file from the file system or from Quark Publishing Platform. If you choose **Platform** from such a button, the **Asset Picker** dialog box lets you choose a Platform asset.

Quark Publishing Platform is a dynamic publishing platform that provides a broad range of functionality to streamline and automate your publishing process. For information about using App Studio with Quark Publishing Platform, see *A Guide to Quark Publishing Platform*. For general information about the Platform, visit *http://www.quark.com*.

Adding a button to an App Studio issue

To add a button to an App Studio issue:

1. Select the rectangular picture box you want to make into a button.
2. In the **HTML5** palette, click **Button**.



Name: Show Next Slide

☐ Initially Hidden

Tap/Click Actions

Next Slide "Coin Collection Slideshow"
Play Sound File"Swoosh.mp3"

Action: Play Sound File

☐ Loop

☑ Stop Audio When Page Turns

Source

● File (Embed)

Swoosh.mp3

Object Name	Page
Interview	1
Movie Object	1
Web Site	3
Coin Collection Slideshow	4
Show Next Slide	4

Actions added to a button

3 Optionally, enter an internal name for the picture in the **Name** field. This name displays in the **Interactive Objects** area in the bottom of the palette.

4 If you want the button to be initially invisible, check **Initially Hidden**.

5 To add an action for the button, click the ✚ button. Choose an action from the **Action** drop-down menu at the bottom, then use the controls below to configure the action. You can add multiple actions to the same button. For more information, see "*Working with interactivity actions*."

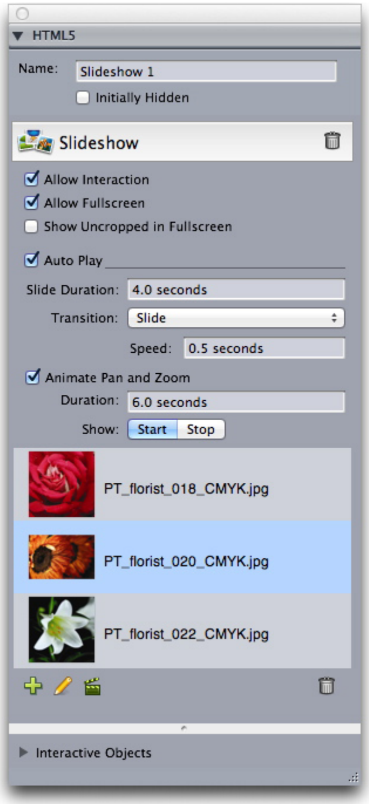
6 To finish configuring the button, click a different object or a blank part of the layout.

Adding a slideshow to an App Studio issue

Slideshows support both picture files and pages from QuarkXPress layouts. To add a slideshow to an App Studio issue:

1 Select the picture box you want to contain the slideshow.

2 In the **HTML5** palette, click **Slideshow**.



HTML5

Name: Slideshow 1

□ Initially Hidden

Slideshow

☑ Allow Interaction

☑ Allow Fullscreen

□ Show Uncropped in Fullscreen

☑ Auto Play

Slide Duration: 4.0 seconds

Transition: Slide

Speed: 0.5 seconds

☑ Animate Pan and Zoom

Duration: 6.0 seconds

Show: Start Stop

PT_florist_018_CMYK.jpg

PT_florist_020_CMYK.jpg

PT_florist_022_CMYK.jpg

Interactive Objects

Slideshow options

3 Optionally, enter a name for the slideshow in the **Name** field. This name displays in the **Interactive Objects** area in the bottom of the palette.

4 If you want the slidehow to be initially invisible, check **Initially Hidden**.

5 To allow the user to interact with the slideshow, check **Allow Interaction**.

6 To allow the user to switch the slideshow to full-screen mode and back by double-tapping it, check **Allow Fullscreen**.

7 To display slides uncropped when the slideshow is in fullscreeen mode, check **Show Uncropped in Fullscreen**. If this box is not checked, the slides use their picture-box crop in fullscreen mode.

8 To make the slideshow start playing immediately when the user displays the page, check **Auto Play**. If this box is unchecked, the user must manually change the slides with finger swipes or with buttons.

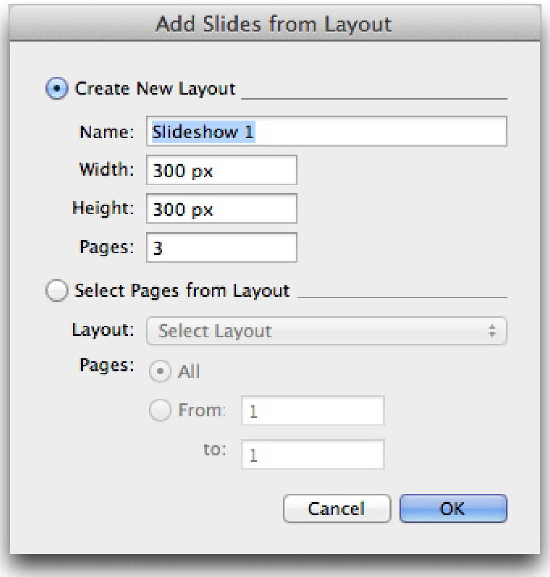
- **Slide Duration**: Lets you control how long each slide is on the screen.
- **Transition**: Lets you control the transition between slides.
- **Speed**: Lets you specify how long each transition lasts.

9 To make the slide pan and/or zoom at the beginning of its display, check **Animate Pan and Zoom**.

- **Duration**: Lets you control how long the pan/zoom lasts. At the end of this duration, the slide stops and stays in its final position until the next slide is displayed.
- **Start** and **Stop**: These buttons allow you to set the beginning and ending crop of each image. Click **Start** and scale/position the picture for the initial position, then click **Stop** and scale/position the image for the final position.

10 To add a slide, click ✚ at the bottom of the slide list and choose one of these options:a

- **Add Slides from Picture Files**: Lets you add picture files to the slideshow. You can use picture files in PNG, JPEG, TIFF, PDF, and EPS formats. You can use the Shift or Command/Ctrl key to select multiple files.
- **Add Slides from QuarkXPress Layout**: Displays the **Add Slides from Layout** dialog box. This dialog box lets you add pages from a QuarkXPress layout to the slideshow. You can create a new layout or choose pages from an existing layout.



Add Slides from Layout

● Create New Layout

Name: Slideshow 1

Width: 300 px

Height: 300 px

Pages: 3

○ Select Pages from Layout

Layout: Select Layout

Pages: ● All

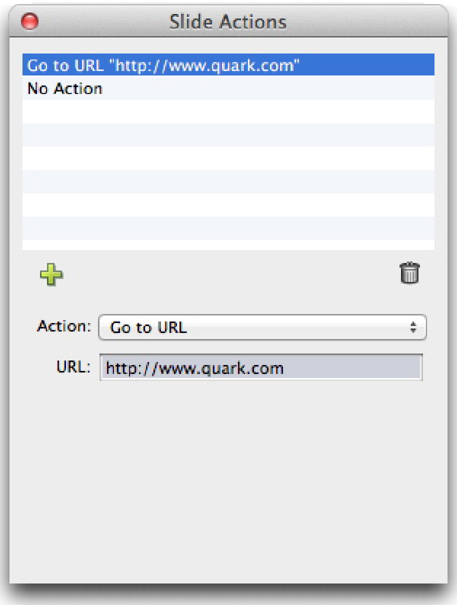
○ From: 1

to: 1

Add Slides from Layout dialog box

11 To edit a slide, select it and click ✎. If the slide is a picture file, that files opens in the default image editing application. If the slide is a page from a QuarkXPress layout, the layout opens and scrolls to that page.

12 To execute an action when the user taps a slide, select it and click 🎬. The **Tap Actions for Slide** dialog box displays.



Go to URL "http://www.quark.com"
No Action

Action: Go to URL

URL: http://www.quark.com

Tap Actions for Slide dialog box

To add an action, click ✚, then choose the action from the **Action** drop-down menu and configure it the way you want it. (For more information, see "*Working with interactivity actions*."

13 To set the cropping for a slide, select the slide's icon in the list, then scale and reposition the preview of the slide in the picture box.

14 To delete a slide, select the slide's icon in the list and click 🗑.

15 To finish configuring the slideshow, click a different object or a blank part of the layout.

Adding a video to an App Studio issue

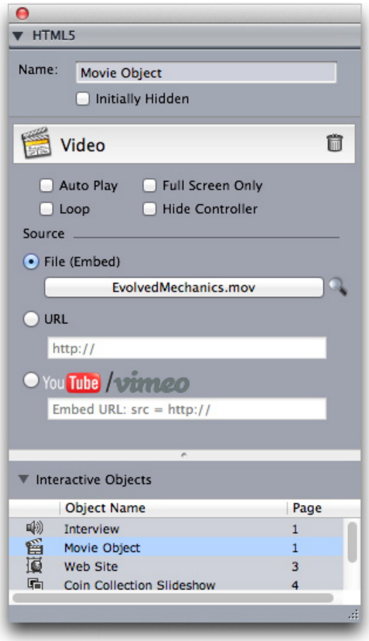
App Studio issues support only H.264 video at up to 720p, 30 frames per second, Main Profile level 3.1, with AAC-LC stereo audio at up to 160 Kbps per channel, 48kHz, in the .mp4 file format.

➡ You can start and stop the playback of video files with actions. For more information, see "*Working with interactivity actions*."

To add a video to an App Studio issue:

1 Select the picture box you want to contain the video.

2 In the **HTML5** palette, click **Video**.



HTML5

Name: Movie Object

□ Initially Hidden

Video

□ Auto Play □ Full Screen Only

□ Loop □ Hide Controller

Source

● File (Embed)

EvolvedMechanics.mov

○ URL

http://

○ YouTube/vimeo

Embed URL: src = http://

Interactive Objects

Object Name	Page
Interview	1
Movie Object	1
Web Site	3
Coin Collection Slideshow	4

Video options

3 Optionally, enter a name for the picture in the **Name** field. This name displays in the **Interactive Objects** area in the bottom of the palette.

4 If you want the video to be initially invisible, check **Initially Hidden**.

5 To make the video play automatically when the page it is on displays, check **Auto Play**.

6 To make the video switch to fullscreen mode as soon as it begins to play, check **Full Screen Only**.

7 To make the video play repeatedly, check **Loop**.

8 To hide the default video controls, check **Hide Controller**. Note that if you want the user to be able to control the movie, you will need to provide a way of controlling the video with actions.

9 To specify the location of the video, click an option in the **Source** area.

- To use a local video file, click **File (Embed)**, then click the button and select the video file.
- To use a video from a URL, click **URL** and enter the URL in the field.
- To use a video file from Quark Publishing Platform, click **Quark Publishing Platform**, then click **Browse QPP** and select the video file.
- To use a video file from YouTube or Vimeo, click **YouTube/Vimeo**, then paste the url of the video in the **src=** field. (To get this value for a YouTube video, Control+click/right-click the video and choose **Copy embed URL**. To get this value for a Vimeo video, Control+click/right-click the video, choose **Copy embed code**. Then paste the code into a text editor, and then select the portion of the code that looks like this: `http://www.[site].com/embed/XXXXXXXXXX` and paste it into the field.)

10 To finish configuring the video, click a different object or a blank part of the layout.

Adding audio to an App Studio issue

You can associate an audio file with a picture box. When the end user views the issue, the box is replaced with audio controls that allow the sound to be played. You can also configure sound files to play in the background and to continue playing when the user changes pages.

You can start and stop the playback of audio files with actions. For more information, see "*Working with interactivity actions*."

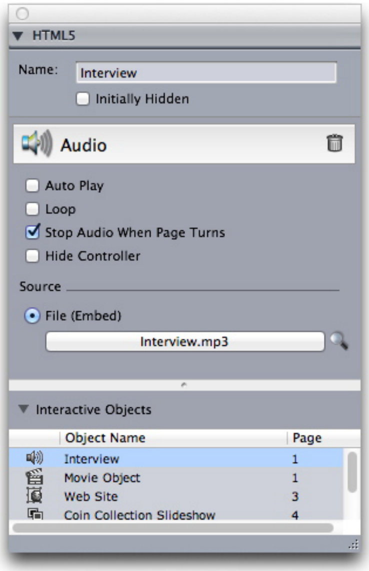
App Studio issues support only .mp3 files.

Only one audio file can play at a time. If one sound is playing, and another sound starts, the first sound is paused.

To add audio to an App Studio issue:

1 Select the picture box you want to replace with the audio controls.

2 In the **HTML5** palette, click **Audio**.



HTML5

Name: Interview

□ Initially Hidden

Audio

□ Auto Play

□ Loop

☑ Stop Audio When Page Turns

□ Hide Controller

Source

● File (Embed)

Interview.mp3

Interactive Objects

Object Name	Page
Interview	1
Movie Object	1
Web Site	3
Coin Collection Slideshow	4

Audio options

3 Optionally, enter a name for the audio controller in the **Name** field. This name displays in the **Interactive Objects** area in the bottom of the palette.

4 If you want the audio object to be initially invisible, check **Initially Hidden**.

5 To make the audio play automatically when the page it is on displays, check **Auto Play**.

6 To make the audio play repeatedly, check **Loop**.

7 To make the audio stop playing when the user goes to a different page, check **Stop Audio When Page Turns**. If you leave this box unchecked, the audio will continue to play until the user moves to the next article.

Sounds that play at the article level do not play when you preview the article in a Web browser.

8 If **Stop Audio When Page Turns** is checked, you can hide the default audio controls by checking **Hide Controller**.

9 To specify the location of the audio file, click an option in the **Source** area.

- To use a local audio file, click **File (Embed)**, then click the button and select the audio file.
- To use an audio file from Quark Publishing Platform, click **Quark Publishing Platform**, then click **Browse Platform** and select the audio file.

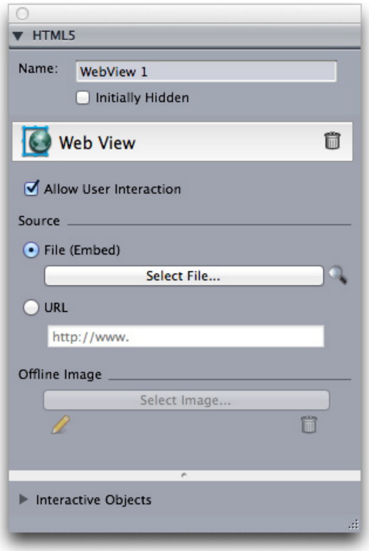
10 To finish configuring the audio object, click a different object or a blank part of the layout.

Adding a Web view to an App Studio issue

You can use a Web view to include changeable content (such as ads) or custom interactivity to an App Studio issue. You can use a Web view to display HTML, PDF, Word, Excel®, PowerPoint®, RTF, and several other types of files. On the output device, the WebKit framework is used to render the content. To add a Web view to an App Studio issue:

1 Select the picture box you want to contain the Web content.

2 In the **HTML5** palette, click **Web View**.



HTML5

Name: WebView 1

☐ Initially Hidden

Web View

☑ Allow User Interaction

Source

● File (Embed)

○ URL

http://www.

Offline Image

Web view options

3 Optionally, enter a name for the Web view in the **Name** field. This name displays in the **Interactive Objects** area in the bottom of the palette.

4 If you want the Web view to be initially invisible, check **Initially Hidden**.

5 To allow the user to interact with the content, check **Allow User Interaction**.

6 To specify the location of the content, click an option in the **Source** area.

- To use a local file, click **File (Embed)**. Put the file and any other files it uses (such as pictures) into a folder by themselves, then click the button and select the file. To ensure that links work, all files in the folder that contains this file will be added to the article.
- To use a URL, click **URL** and enter the URL in the field.

7 If the content in the Web view is not embedded, you can specify an image to display when the Internet connection is not available. To do so, click **Select Image** in the **Offline Image** area and choose a picture file. You can use the icons under this button to edit or delete the offline image.

➡ Offline images must be in PNG or JPEG format.

8 To finish configuring the Web view, click a different object or a blank part of the layout.

Adding a scroll zone to an App Studio issue

The scroll zone feature lets you add a scrollable area to a page of an App Studio issue. The content for the scrollable area comes from a different layout (the scrollable layout). Once you set up a scroll zone, you can populate it with whatever you want, whether it's a long run of text, a large panoramic picture, or a series of interactive elements. You can then use that scrollable layout in multiple layouts within a layout family. (For more information, see "*Understanding layout families*.")

➡ Referenced scrollable layouts are exported as HTML in the same way the host layout is. That means they work the same way in terms of font usage and the **Convert To Graphic on Export** option for text boxes.

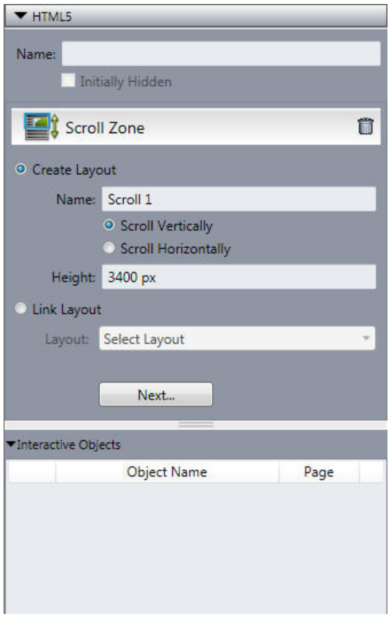
➡ Referenced layout can contain their own interactive objects, and they will work as they would in the host layout.

To set up a scroll zone for an App Studio issue:

1 Navigate to a layout that is part of a layout family.

2 Draw a picture box to represent the size and the location of the scroll zone. Make sure the box is selected.

3 In the **HTML5** palette, click **Scroll Zone**.



Name:

□ Initially Hidden

● Create Layout

Name: Scroll 1

● Scroll Vertically

○ Scroll Horizontally

Height: 3400 px

○ Link Layout

Layout: Select Layout

Scroll zone controls

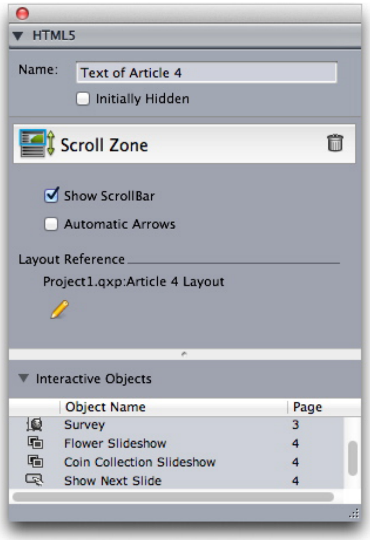
4 Optionally, enter a name for the scroll zone in the **Name** field. This name displays in the **Interactive Objects** area in the bottom of the palette.

5 If you want the scroll zone to be initially invisible, check **Initially Hidden**.

6 Choose whether you want to create a layout or link to an exising layout.

- If you want to create a new layout to serve as the scrollable layout, click **Create Layout** and enter a name for the layout in the **Name** field. You can decide whether you want to create a horizontal or vertical scrollbox. A vertical scrollbox can be as long as you want it to be, a horizontal scrollbox is restricted to one page. Enter the height of the scrollable layout in the **Height** field.
- If you want to use an existing layout in the active project as the scrollable layout, click **Link Layout** and choose that layout's name from the **Layout** drop-down menu. The size of the selected box is adjusted to fit the layout.

7 Click **Next**. The **HTML5** palette displays the following controls.



HTML5

Name: Text of Article 4

□ Initially Hidden

Scroll Zone

☑ Show ScrollBar

□ Automatic Arrows

Layout Reference

Project1.qxp:Article 4 Layout

Interactive Objects

Object Name	Page
Survey	3
Flower Slideshow	4
Coin Collection Slideshow	4
Show Next Slide	4

Scroll zone layout controls in **HTML5** palette

8 To provide scrollbars as a visual cue that the area is scrollable, check **Show ScrollBar**.

9 To automatically display arrows that indicate the direction in which the scroll zone can be scrolled, check **Automatic Arrows**.

10 To edit the scrollable layout, click the **Edit** button ✎ under **Layout Reference**.

11 To finish configuring the scroll zone, click a different object or a blank part of the layout.

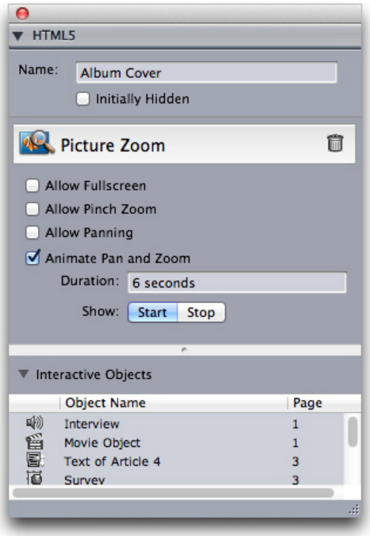
Adding a zoomable picture to an App Studio issue

A zoomable picture initially displays in a box, but takes over the entire screen when double-tapped. You can use this feature to add an animated pan-and-zoom effect to an image, or allow it to be zoomed and panned directly in its box.

App Studio issues support interactive pictures in PNG, JPEG, GIF, TIFF, and EPS formats. To add an interactive picture to an App Studio issue:

1 Select the picture box that contains the picture you want to make zoomable.

2 In the **HTML5** palette, click **Picture Zoom**.



▼ HTML5

Name: Album Cover

□ Initially Hidden

Picture Zoom

□ Allow Fullscreen

□ Allow Pinch Zoom

□ Allow Panning

☑ Animate Pan and Zoom

Duration: 6 seconds

Show: Start | Stop

▼ Interactive Objects

Object Name	Page
Interview	1
Movie Object	1
Text of Article 4	3
Survey	3

Picture zoom controls

3 Optionally, enter a name for the zoomable picture in the **Name** field. This name displays in the **Interactive Objects** area in the bottom of the palette.

4 If you want the zoomable picture to be initially invisible, check **Initially Hidden**.

5 To allow the user to switch back and forth between cropped view and fullscreen view by double-tapping the picture, check **Allow Fullscreen**.

6 To allow the user to zoom in and out with pinch gestures, check **Allow Pinch Zoom**.

7 To allow the user to pan around in the picture with one finger, check **Allow Panning**.

8 To make the picture pan and/or zoom at the beginning of its display, check **Animate Pan and Zoom**.

- **Duration**: Lets you control how long the pan/zoom lasts. At the end of this duration, the slide stops and stays in its final position until the next slide is displayed.
- **Start** and **Stop**: These buttons allow you to set the beginning and ending crop. Click **Start** and scale/position the picture for the initial position, then click **Stop** and scale/position the image for the final position.

9 To finish configuring the zoomable picture, click a different object or a blank part of the layout.

Working with interactivity actions

Actions let you add interactivity to items in a layout that you will be exporting as an App Studio app. The actions available include:

- **No Action**: The default action. Does nothing.
- **Go to URL**: Navigates to a URL when the user touches the selected item.

- **Go to Page**: Jumps to a different page. For more information, see "*Creating a Go to Page action*."
- **Go to Next Page**: Jumps to the next page.
- **Go to Previous Page**: Jumps to the page before this one.
- **Go to First Page of Issue**: Jumps to the first page in the issue.
- **Go to Slide**: Displays the specified slide of the specified slideshow.
- **Next Slide**: Displays the next slide of the specified slideshow.
- **Previous Slide**: Displays the previous slide of the specified slideshow.
- **Play Object**: Lets you start playing the specified audio or video object.
- **Pause Object (Toggle)**: Lets you pause and resume the playing of the specified audio or video object.
- **Play Sound File**: Lets you play a sound file. For more information, see "*Creating a Play Sound File action*."
- **Pause Sound Files (Toggle)**: Lets you pause and resume the playback of all sound files.
- **Show Pop-Up**: Lets you display a pop-up. For more information, see "*Creating a Show Pop-Up action*."
- **Hide Pop-Up**: Lets you hide the currently displayed pop-up.
- **Show Object**: Lets you show the specified object (if it is hidden).
- **Hide Object**: Lets you hide the specified object (if it is visible).
- **Open**: Lets you open files and display them within your application. This must be a format that the device understands how to display. For example, you can open a PDF file and display that within your app on a device that supports PDF file viewing.

Creating a Go to Page action

A **Go to Page** action lets the user jump to a different page. To create a **Go to Page** action:

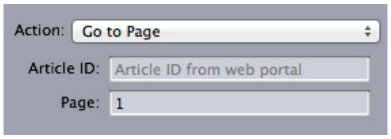
1. Display the article that contains the target page on the App Studio Publishing Portal.
2. Click the **Edit** button on the thumbnail of the article. The **Edit Content** dialog box displays, including the ID of the article in this format:

   ```
   [IssueName]_XX/X
   ```

   For example, if your issue name is "MyIssue", it might be something like this:

   ```
   MyIssue_01/1
   ```

3. In QuarkXPress, Choose **Go to Page** from the **Action** drop-down menu.



Action: Go to Page

Article ID: Article ID from web portal

Page: 1

4. Enter the article ID in the **Article ID** field.

5. Enter the absolute page number of the target page in the **Page** field. (For example, to go to the second page in the target article, enter 2.)

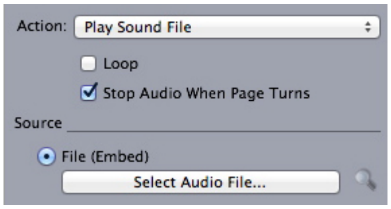
Creating a Play Sound File action

A **Play Sound File** action lets you play a sound file.

➡ App Studio issues support only .mp3 files.

To create a **Play Sound File** action:

1. Choose **Play Sound File** from the **Action** drop-down menu.



Action: Play Sound File

☐ Loop

☑ Stop Audio When Page Turns

Source

● File (Embed)

2. To make the sound play over and over again, check **Loop**.

3. To make the sound stop when the user navigates to a different page, check **Stop Audio When Page Turns**.

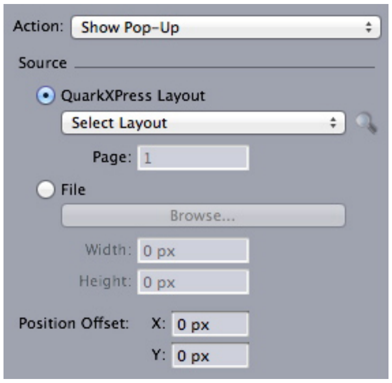
4. To specify the location of the sound file, click **File (Embed)**, then click the button and select the sound file.

Creating a Show Pop-Up action

A **Show Pop-Up** action lets you display a pop-up view containing either the contents of a QuarkXPress layout in the same project file or the contents of an external file. If you choose a QuarkXPress layout, any interactivity in that layout should work in the pop-up. If you choose to display an external file, you can use any file that can be displayed in the native mobile browser on your device.

To create a **Show Pop-Up** action:

1. Choose **Show Pop-Up** from the **Action** drop-down menu.



Action: Show Pop-Up

Source

● QuarkXPress Layout

Select Layout

Page: 1

○ File

Width: 0 px

Height: 0 px

Position Offset: X: 0 px

Y: 0 px

2 Do one of the following things:

- To display the contents of a different QuarkXPress layout, click **QuarkXPress Layout**, then choose the target layout from the drop-down menu or choose **Create New Layout** and create a new layout. Enter a page number in the **Page** field to indicate which page to display in the pop-up. Click the **Edit** ✏ button to display the layout.
- To display the contents of an external file, click **File**, then click **Browse** and navigate to the file you want to display, then specify the width and height at which you want it to display. Use the **Width** and **Height** values to specify the size at which to display the file. If you use an external file, the pop-up includes a built-in close button.

3 To offset the content of the pop-up, enter values in the **Position Offset** fields.

Text in App Studio layouts

By default, text boxes in an App Studio layout are set to export as HTML text. This allows them to be searchable and selectable. However, it also means that you cannot apply some of the higher-end text controls (such as tracking, kerning, hyphenation, and baseline shift) to text in such boxes.

If you want to have complete typographical control over text in a box, choose **Item** > **Modify** and check **Convert to Graphic on Export**. The box will export as an image rather than as text, so its text won't be searchable or selectable, but you can make it look exactly the way you want it to.

Because the auto-leading settings in WebKit differ from the auto-leading settings in QuarkXPress, you should avoid using auto leading in App Studio layouts. To achieve maximum similarity between line spacing in the layout and on the device, always use absolute leading values in an App Studio layout.

Fonts in App Studio layouts

Unless you convert a text box to a graphic (see "*Text in App Studio layouts*"), you must make sure that you use fonts that is available on the devices where they will be displayed. For a list of the fonts supported on iOS, see *http://iosfonts.com*. Android devices, by default, support only Droid Serif, Droid Sans, and Droid Sans Mono, but additional fonts can be installed.

➡ When uploading layouts to App Studio, you can now use custom fonts, if they are in the TTF or OTF format. These will be embedded when uploading and therefore can be displayed within your app. If you are embedding large font files, this will increase the file size of your app content substantially.

Hyperlinks in App Studio layouts

To apply a hyperlink to text, select the text and use the **Hyperlinks** palette (for more information, see *A Guide to QuarkXPress*).

To apply a hyperlink to an entire box, select the box and use the **HTML5** palette (see "*Adding a button to an App Studio issue*").

Groups in App Studio layouts

If you have multiple overlapping picture boxes and graphic text boxes, you may be able to streamline your export by grouping the boxes and then checking **Convert to Graphic on Export** (**Item** > **Modify**) for the group. If you do not group such boxes, they all export as separate HTML elements. If you group them, however, they export as a single graphic object.

Synchronizing content between orientations

The commands described in this topic let you synchronize a page's content between the various layouts in a layout family. For example, if you create a table of contents on page 2 of the vertical layout for the iPad, you can use the commands described here to automatically turn that table of contents into shared content and copy it to page 2 of the horizontal layout for the iPad and/or the vertical layout for the Galaxy Tab. You may need to reposition and/or resize the content in the other layouts, but these commands automate the process of getting the content to the appropriate pages in each device and orientation, and they help ensure that that content stays consistent on all of the pages where it appears.

➡ For more information about shared content, see "Working with shared content" in *A Guide to QuarkXPress*. To understand how the commands described below work, you must first understand how shared content works.

Item > **Copy to Other Layouts**: This command turns all selected boxes into shared content and places copies of them on the same page of other layouts in the layout family. The submenu for this command lets you be more specific about how this copy is performed:

- **Same Position**: Puts the copies in the same position relative to the upper-left corner of the page, if possible. This option copies items to all other layouts in the layout family.
- **Use Relative Positioning**: Puts the copies in the same position relative to the page size and shape overall. This option copies items to all other layouts in the layout family.

Updating Missing Files

Multiple missing digital files can now be updated all at once using the **Usage** dialog.

Reviewing Digital Publishing asset usage

To see which assets have been used in App Studio interactivity in the active layout, display the **Digital Publishing** pane of the **Usage** dialog box (**Utilities** menu). Like the **Pictures** pane, the **Digital Publishing** pane lists the page number (a dagger symbol or PB indicates pasteboard) for each Digital Publishing asset in the layout. It also includes a **Type** column that indicates the asset type and a **Source** column that indicates whether the asset is a file or a folder. To view more information about the selected asset, expand the **More Information** area at the bottom.

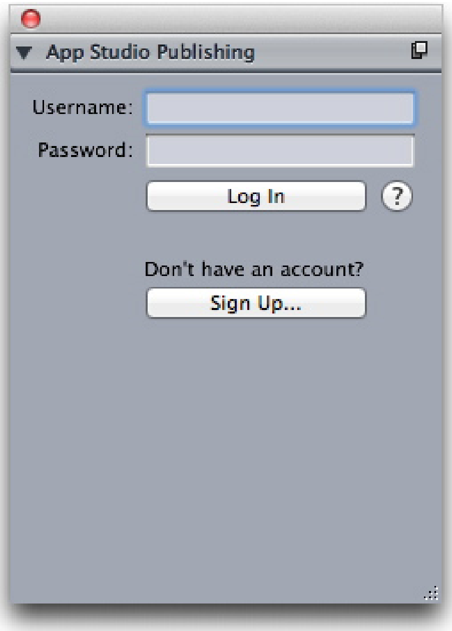
Exporting an App Studio article

Export works differently for App Studio issues than it does for other formats. Rather than creating a single local exported issue file on your hard disk, you upload the exported articles that make up an issue to the App Studio Publishing Portal. Once an article has been uploaded to the server, you can preview it on the App Studio Publishing Portal and in your copy of App Studio Issue Previewer. Once the content is final, you can serve it to your customers directly from the App Studio Publishing Portal.

➡ Before you can export and upload an App Studio article, you must have an account on the App Studio Publishing Portal. For more information, see "*Understanding the App Studio Publishing Portal*" and "*Creating an App Studio issue*."

To export the active layout as an article in an App Studio issue:

1 Display the **App Studio Publishing** palette (**Window** menu).

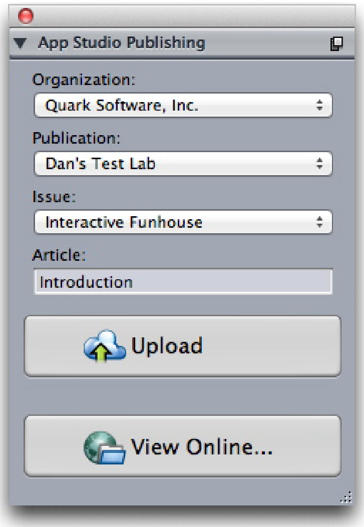


App Studio Publishing palette - login version

2. If you use a proxy server, choose **Proxy Settings** from the palette menu. The **Proxy Settings** dialog box displays.

- To automatically detect the network's proxy server, check **Auto Detect Proxy Settings For This Network**. If you uncheck this box, you must enter the proxy server's host address and port number manually.
- Enter the user name and password for the proxy server.
- Click **OK**.

3. If you do not yet have a free App Studio Publishing Portal account, click **Sign Up** and create one.
4. Enter your user name and password, and then click Log In. The **App Studio Publishing** palette changes to the upload interface.



Organization: Quark Software, Inc.

Publication: Dan's Test Lab

Issue: Interactive Funhouse

Article: Introduction

App Studio Publishing palette - upload version

5. Choose your organization from the **Organization** drop-down menu.
6. Choose your publication from the **Publication** drop-down menu.
7. Choose the target issue from the **Issue** drop-down menu.
8. Enter a name for the article in the **Article Title** field. This name will display in the app's thumbnail table of contents.
9. To upload the article, click **Upload**.
10. To view the uploaded article in your default Web browser, click **View Online**. The article's issue displays in your default Web browser.
11. To make the issue that the article is part of available in the test environment so that you can preview it, click **Test** in the App Studio Publishing Portal.

Previewing an App Studio issue

There are two ways to preview an App Studio issue.

The most accurate way to preview your issues is with the free Quark App Studio Previewer app. Download this app from the app store, launch it, tap **Change Publication**, and enter your login credentials for your App Studio Publishing Portal account. Then choose the publication you want to log on to.

There are two versions of each publication: the published version and the test version. The test version is provided so that you can test your issues on a physical device before you release them. An issue shows up in the test publication only if you have clicked **Test** for that issue in the App Studio Publishing Portal, and shows up in the regular publication only if you have published it from the App Studio Publishing Portal. (For more information, see "*Publishing an App Studio issue*.")

The second way to preview your issues is on the App Studio Publishing Portal. You can click on any issue and view it in a compatible Web browser. (App Studio supports Google Chrome, Firefox, Safari, and any other browser that uses the WebKit browser engine. It does not support Internet Explorer.) You should consider this a secondary way to preview your issues, because not all types of interactivity can be previewed in a browser.

Publishing an App Studio issue

To make an App Studio issue available in your customers' apps, click **Publish** for that issue in the App Studio Publishing Portal. This issue automatically becomes available on your customers' iPads.

Creating an App Studio app

This chapter describes the process of preparing to submit an App Studio app to Apple or to an Android app store.

Preparing to submit an App Studio app to Apple

The process in general for preparing to submit an App Studio app to Apple is:

1. Create your Apple iOS Developer account.
2. Get your iPad's device ID.
3. Create your iOS Development Certificate.
4. Create your iOS Distribution Certificate.
5. Register your iPad so you can test on it.
6. Create your app ID.
7. Create your Development Provisioning Profile.
8. Create your App Store Distribution Provisioning Profile.
9. Create your app description.
10. If you are charging for your content, set up your in-app purchases.

The following topics explain this process in detail. For more information, see the iTunes Connect Developer Guide, which is available to Apple developers at *itunesconnect.apple.com*.

Creating your developer account

Before you can proceed, you must create an Apple iOS Developer account. To do so:

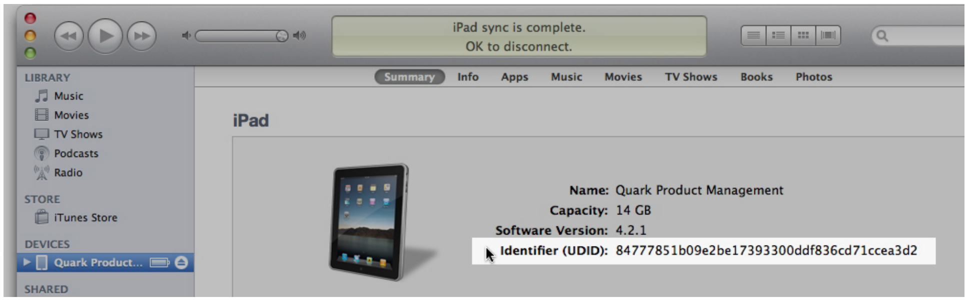
1. Go to *http://developer.apple.com/programs/ios/* and click **Enroll Now**.
2. Click **Continue** and follow the instructions on the screen. When you are asked to enroll as an individual or a company, choose the appropriate option.

3. Go to *itunesconnect.apple.com*, click **Contracts, Tax, and Banking**, and enter into an iOS Paid Applications contract with Apple. If you plan to charge for your app or issues, this contract must be in effect, with valid bank info entered, before you submit your app to the Apple app store.

Getting your iPad's device ID

To test your app on your iPad, you must specify its device ID (that is, its UDID).To get your iPad's UDID:

1. Plug your iPad into your computer.
2. Launch iTunes 7.7 or later.
3. Select the iPad in iTunes.
4. Click the **Serial Number** label. The label changes to **Identifier (UDID)**.



Name: Quark Product Management
Capacity: 14 GB
Software Version: 4.2.1
Identifier (UDID): 84777851b09e2be17393300ddf836cd71ccea3d2

5. Press Command+C to copy the device ID to the clipboard.
6. Paste the device ID into a text file. It should be a very long code with both numbers and lower-case letters, but no hyphens or other characters.

Creating your iOS Development Certificate

You must have an iOS Development Certificate to develop apps and test them on an iPad. To create this Certificate:

1. Go to *http://developer.apple.com/devcenter/ios/* and log in.
2. Under **iOS Developer Program** in the sidebar, click **iOS Provisioning Portal**.
3. Click **Certificates** in the left sidebar.
4. Click the **Development** tab.
5. Click **Request Certificate**.
6. Follow the instructions on the screen to create, download, and install your Development Certificate.

Creating your iOS Distribution Certificate

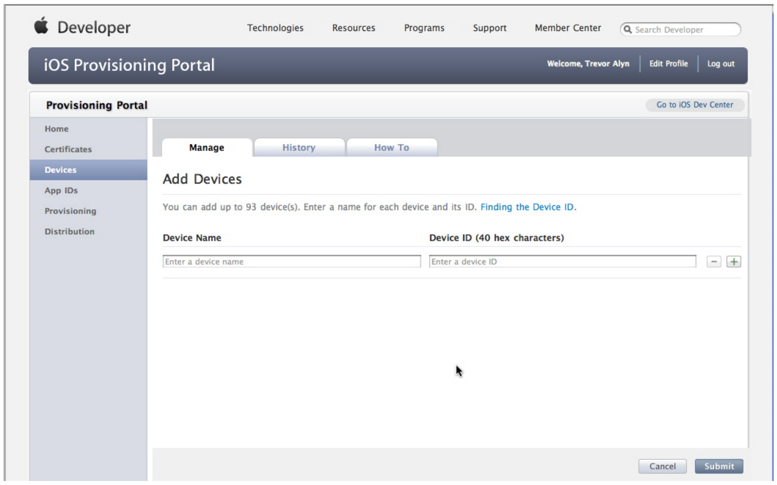
You must have an iOS Distribution Certificate to submit an app to the App Store. To create this Certificate:

1. Go to *http://developer.apple.com/devcenter/ios/* and log in.
2. Under **iOS Developer Program** in the sidebar, click **iOS Provisioning Portal**.
3. Click **Certificates** in the left sidebar.
4. Click the **Distribution** tab.
5. Click **Request Certificate**.
6. Follow the instructions on the screen to create, download, and install your Distribution Certificate.

Registering devices

Until they are in the App Store, apps can be run only on devices that they are registered to run on. To register your iPad for testing:

1. Click **Devices** in the left sidebar.
2. Click **Add Devices**. The **Add Devices** screen displays.



Add Devices

You can add up to 93 device(s). Enter a name for each device and its ID. Finding the Device ID.

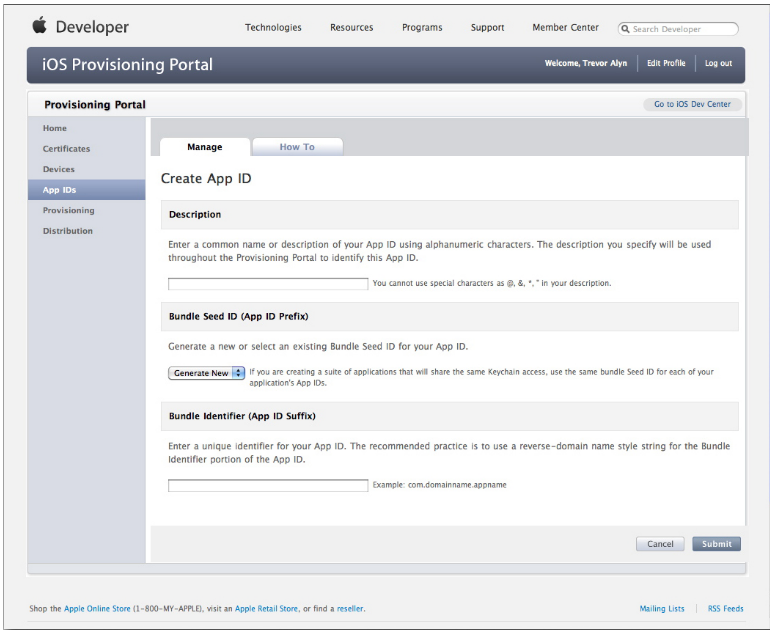
3. Enter a name for your iPad (for example, "Jane's iPad") in the **Device Name** field, then paste the iPad's UDID in the **Device ID** field.
4. To add an additional device, click the + button to the right of the **Device ID** field.
5. Click **Submit**.

Creating an app ID

Every app must have a unique ID. To create your app ID:

1 Click **App IDs** in the left sidebar. The **App IDs** screen displays.

2 Click **New App ID**. The **Create App ID** screen displays.



Create App ID

Description

Enter a common name or description of your App ID using alphanumeric characters. The description you specify will be used throughout the Provisioning Portal to identify this App ID.

You cannot use special characters as @, &, *, " in your description.

Bundle Seed ID (App ID Prefix)

Generate a new or select an existing Bundle Seed ID for your App ID.

If you are creating a suite of applications that will share the same Keychain access, use the same bundle Seed ID for each of your application's App IDs.

Bundle Identifier (App ID Suffix)

Enter a unique identifier for your App ID. The recommended practice is to use a reverse-domain name style string for the Bundle Identifier portion of the App ID.

Example: com.domainname.appname

Shop the Apple Online Store (1-800-MY-APPLE), visit an Apple Retail Store, or find a reseller.

3 Enter the name of your app in the **Description** field. Note that this is NOT the name that will display for the app in the App Store.

4 Leave the **Bundle Seed ID (App ID Prefix)** drop-down menu set to **Generate New**.

5 In the **Bundle Identifier (App ID Suffix)** field, enter a unique identifier for your app, using the form `com.[your organization's name].[name of your app]`. For example, if the app is being created for 123 Productions, the bundle identifier might be `com.123productions.123magazine`.

➡ Bundle IDs are case-sensitive.

6 Click **Submit**.

Setting up for push notifications

You can use push notifications to notify your customers when a new issue of your publication becomes available.

➡ You must complete this step before you create your Provisioning Profiles. (For more information, see the two topics after this one.)

To set up for push notifications:

1 Click **App IDs** in the left sidebar. The **App IDs** screen displays.

2 Click the name of the app you want to enable notifications for. The **Configure App ID** page displays.

3 Check **Enable for Apple Push Notification Service**.

➡ Do not check **Enable for iCloud**, **Enable for Passes**, **Enable for Data Protection**, or **Enable for Game Center**.

4 Click the **Configure** button for **Development Push SSL Certificate**. The **Apple Push Notification Service SSL Certificate Assistant** screen displays.

5 Follow the instructions on the screen. The instructions will direct you to create a certificate signing request, upload it in the **Apple Push Notification Service SSL Certificate Assistant** screen, and download an Apple Push Notification Service SSL certificate. This file should have a name like "aps_development_identity.cer."

6 Double-click the certificate file to install it.

7 Repeat the last three steps for **Production Push SSL Certificate**.

8 Launch the Keychain Access application (Applications > Utilities) and click **My Certificates** in the list on the left. A list of certificates displays.

9 Expand the certificate named **Apple Development iOS Push Services**. A key icon displays.

10 Select both **Apple Production iOS Push Services** and the line with the key icon.

11 Control+click the two selected lines and choose **Export 2 items**, then save the items in Personal Information Exchange (.p12) format with the file name `myapp_development_push.p12`. Do not enter a password. If a dialog box displays, stating that Keychain Access wants to export a key, enter your login password and then click **Allow**.

12 Repeat the last three steps for **Apple Production iOS Push Services** and save the .p12 file as `myapp_production_push.p12`.

13 Set up an account at Urban Airship (*http://urbanairship.com/*). You will use this account to send your push notifications.

14 Keep the .p12 files nearby. You will need them when you submit your app request on the App Studio Publishing Portal. For more information, see "*Requesting an App Studio app*."

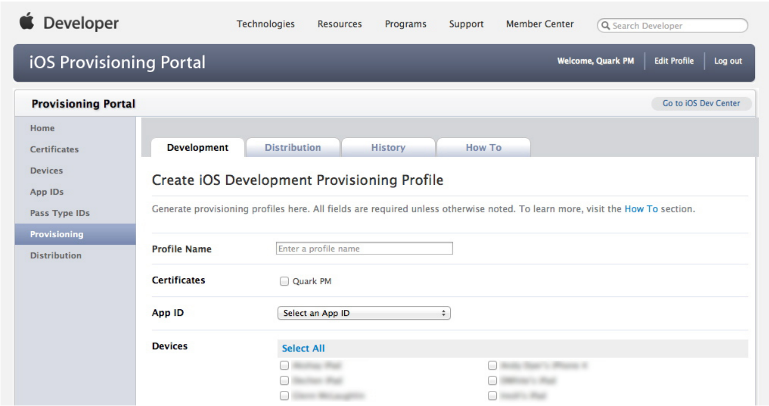
Creating a Development Provisioning Profile

You must have a Development Provisioning Profile to run your app on a physical iPad. To create a Development Provisioning Profile:

1 In the left sidebar, click **Provisioning**.

2 Click the **Development** tab.

3 Click **New Profile**.



Create iOS Development Provisioning Profile

Generate provisioning profiles here. All fields are required unless otherwise noted. To learn more, visit the How To section.

Profile Name: Enter a profile name

Certificates: □ Quark PM

App ID: Select an App ID

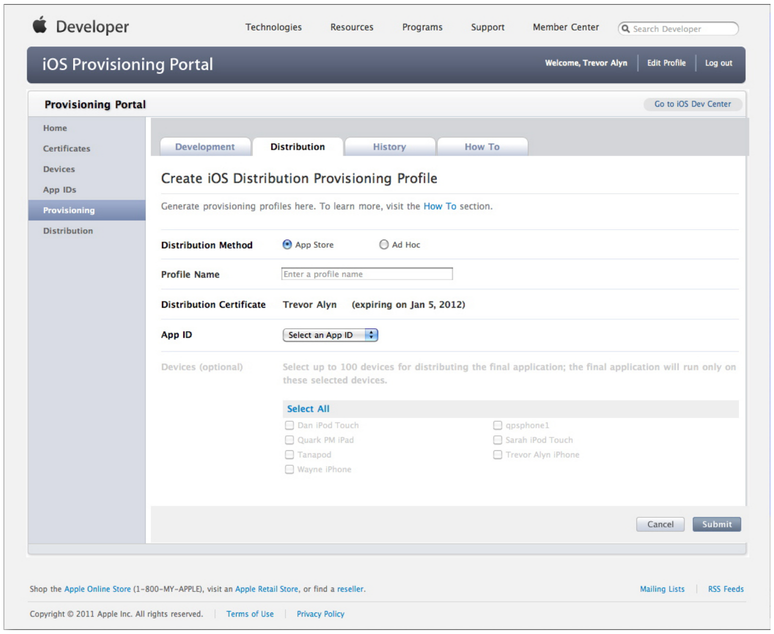
Devices: Select All

4. In the **Profile Name** field, enter "[App Name] Ad Hoc Provisioning Profile".
5. In the **Certificates** area, check your certificate's box.
6. Choose your app's app ID from the **App ID** drop-down menu.
7. In the **Devices** area, click **Select All**, or click the devices you want to test on.
8. Click **Submit**. The **Development Provisioning Profiles** screen displays.
9. Before you can download the Development Provisioning Profile, you must reload the page. After you reload the page, the **Download** button should become available for the Development Provisioning Profile you just created.
10. Click **Download** to download the Development Provisioning Profile to your desktop. This file will have a name like "App_Name_Development_Provisioning_Profile.mobileprovision."
11. Double-click on the downloaded file to install it.
12. Keep the .mobileprovisioning file. You will need it when you request your app.

Creating an App Store Provisioning Profile

You must have an App Store Distribution Provisioning Profile to build an app for submission to the App Store. To create an App Store Provisioning Profile:

1. In the left sidebar, click **Provisioning**.
2. Click the **Distribution** tab.
3. Click **New Profile**.
4. Under **Distribution Method**, click **App Store**.



Create iOS Distribution Provisioning Profile

Generate provisioning profiles here. To learn more, visit the How To section.

Distribution Method ● App Store ○ Ad Hoc

Profile Name: Enter a profile name

Distribution Certificate: Trevor Alyn (expiring on Jan 5, 2012)

App ID: Select an App ID

Devices (optional): Select up to 100 devices for distributing the final application; the final application will run only on these selected devices.

Select All

□ Dan iPod Touch	□ qpsphone1
□ Quark PM iPad	□ Sarah iPod Touch
□ Tanapod	□ Trevor Alyn iPhone
□ Wayne iPhone	

Shop the Apple Online Store (1-800-MY-APPLE), visit an Apple Retail Store, or find a reseller. Mailing Lists | RSS Feeds

Copyright © 2011 Apple Inc. All rights reserved. | Terms of Use | Privacy Policy

5 In the **Profile Name** field, enter "[App Name] Dist Provisioning Profile".

6 Choose your app's app ID from the **App ID** drop-down menu.

7 Click **Submit**. The **Distribution Provisioning Profiles** screen displays.

8 Before you can download the Distribution Provisioning Profile, you must reload the page. After you reload the page, the **Download** button should become available for the Distribution Provisioning Profile you just created.

9 Click **Download** to download the Distribution Provisioning Profile to your desktop. This file will have a name like "App_Name _Dist_Provisioning_Profile.mobileprovision."

10 Double-click on the downloaded file to install it.

11 Keep the .mobileprovisioning file. You will need it when you request your app.

Creating an app description in iTunes Connect

Before you can submit your app to Apple, you need to create a description for that app on the Apple developer Web site.

➡ If you plan to sell your issues, you must create in-app purchases before you execute the following steps. For more information, see "*Setting up in-app purchases*."

To create an app description:

1 Go to *http://itunesconnect.apple.com* and log in (if necessary). The **iTunes Connect** screen displays.

2 Click **Manage Your Applications**. The **Manage Your Apps** screen displays.

3 Click **Add New App**. The **App Information** screen displays.

4 In the **App Name** field, enter the name of your app as you want it to display in the App Store.

5 In the **SKU Number** field, enter a unique value that can be used to identify your app.

6 Choose the ID of your app from the **Bundle ID** drop-down menu. (This is the App ID you created in "*Creating an app ID*.") Make sure you DO NOT select **Xcode: Wildcard AppID - ***.

7 Click **Continue**. The availability date and price tier screen displays. Follow the instructions on the screen to set the price point and availability date of your app. You can choose the current date to get the app posted as soon as possible.

8 Click **Continue.** The metadata and rating screen displays. Complete the forms on this screen.

9 If you are charging for issues, you must associate your in-app purchases with the app. To do this, click **Edit** in the **In-App Purchases** area, then check all of the in-app purchases you want associated with the app and click **Save**.

10 In the **Uploads** area, upload a 1024 x 1024 pixel PNG file where it says **Large App Icon**, and upload a dummy screen shot where it says **iPad Screenshots**. You can change both of these pictures later.

11 Click **Ready to Upload Binary**.

Setting up in-app purchases

If you plan to charge for your issues or offer subscriptions, you must set up an in-app purchase for each issue with Apple. For information on how to do this, see "Managing Your In-App Purchases" in the *iTunes Connect Developer Guide* (available to registered Apple iOS developers at *http://developer.apple.com*).

The process, as it exists as of this writing, is as follows:

1 Go to *http://itunesconnect.apple.com/* and log in (if necessary).

2 Click **Manage Your Applications**. The **Manage Your Apps** page displays.

3 Click your app. The app's page displays.

4 Click **Manage In-App Purchases**. The **In-App Purchases** page displays.

5 Click **Create New**. The **Select Type** screen displays.

6 To specify what type of in-app purchase you're creating, click one of the following buttons.

- **Consumable**: Do not use this option.
- **Non-Consumable**: Click this option to create an in-app purchase for a single issue.
- **Auto-Renewable Subscription**: Check this option to create an automatically renewing paid subscription.
- **Non-Renewing Subscription**: Check this option to create a non-renewing paid subscription.

- **Free Subscription**: Check this option to create a free subscription. (Use this option if you want to use Newsstand with free issues.)

  The next screen displays.

7 In the **Reference Name** field, enter the title and date of the in-app purchase issue. For example:

- If you are creating a Non-Consumable in-app purchase, enter something like `123 Magazine - Single Issue`.
- If you are creating a subscription in-app purchase, enter something like `123 Magazine Subscription`.

8 If you are creating a Non-Consumable in-app purchase or subscription, enter your bundle identifier, followed by the issue name and date, in the **Product ID** field. For example: `com.123productions.123magazine.price.issue.default`.

- To associate this in-app purchase with an App Studio issue, display the issue's main page on the App Studio Publishing Portal, then enter this product ID in the **Apple ID** field.
- To associate this in-app purchase with a subscription, you will need to enter the subscription's information when you submit your app request on the App Studio Publishing Portal. For more information, see "*Requesting an App Studio app*."

9 To specify the language or languages of your issue, click **Add Language** and follow the instructions on the screen.

10 If you are creating a Non-Consumable in-app purchase, check **Cleared for Sale** and choose the price tier you want from the **Price Tier** drop-down menu. (For a corresponding list of prices, click **View Pricing Matrix**.).

11 If you are creating an subscription in-app purchase, you must specify one or more available durations for the subscription. Click **Add Duration** and follow the instructions on the screen. The **Create New In-App Purchase** page displays.

- In the **Product ID** field, enter your bundle identifier, followed by the issue duration. For example: `com.123productions.123magazine.6months`.
- Check **Cleared for Sale**.
- If you are creating a paid subscription, choose the price tier you want from the **Price Tier** drop-down menu. (For a corresponding list of prices, click **View Pricing Matrix**.)

12 In the **Screenshot for Review** area, click **Choose File** and upload a screen shot of what an issue of your app will look like when viewed on the iPad. Note that this image is only for review and will not be displayed in the App Store.

➡ This screen shot can be changed later.

13 If you plan to offer subscriptions, click **View or generate a shared secret** on the in-app purchases screen for your app. You will need your Shared Secret when you submit your app request on the App Studio Publishing Portal. For more information, see "*Requesting an App Studio app*."

14 Click **Save**.

➡ If you want to change the price of an issue after it has been approved, change the price of its corresponding in-app purchase. Apple will not need to review the change.

Preparing to submit an Android App Studio app

There are a number of Android app stores that you can submit an app to, including the following:

- **Google Play** (*https://play.google.com/apps/publish*): As of this writing, you can submit apps to the Google Play app store for a $25 registration fee.
- **Amazon Appstore** (*https://developer.amazon.com/welcome.html*): As of this writing, you can submit apps to the Amazon Appstore for a yearly fee of $99 (which is currently being waived). Apps submitted to the Amazon Appstore are automatically considered for inclusion in the Kindle Fire Appstore.

The process for submitting an app to the above app stores is similar to the process of submitting an app to Apple. You must create a developer account, then create an app description with supporting pictures and other resources. You can currently submit the same app to both of the above app stores, and both of them offer an in-app purchase feature that allows you to charge for your issues.

For more information, see the Web site for each app store.

Requesting an App Studio app

To get a build of an iOS or Android App Studio app, you must fill out a form online and submit any custom graphics you would like to add to the app. When Quark receives the form, they will use that information to create a custom build for testing and another custom build for app store submission. If you are requesting an Apple app, make sure you have:

- Your Development Certificate
- Your Distribution Certificate
- Your Ad Hoc Provisioning Profile
- Your App Store Provisioning Profile
- Your app's App ID

- Your home screen icon in PNG format, at 144 x 144 pixels
- Your landscape splash screen in PNG format, at 2048 x 1536 pixels
- Your portrait splash screen in PNG format, at 1536 x 2048 pixels
- The IDs of any in-app purchases you want to use
- Your .p12 files (if you are using notifications)

To request an App Studio app:

1. On the App Studio Publishing Portal (*http://my.appstudio.net*), go to the publication for which you want to create an app.
2. Locate the entry for the application you want to build and click View for that application.
3. Click **App Manager**.
4. Click **Build** for the platform you want to build for. A form displays.
5. Fill out the form and provide the various resources it requests. If you have questions about a part of the form, click the corresponding question mark in the form.
6. When the build is ready, you will receive an e-mail. Download the builds per the instructions in the e-mail.
7. To test the app, drag the test version of the build onto your iTunes icon, then sync with your iPad.

➡ If the app does not display on the iPad, verify that the iPad's UDID is associated with your Development Provisioning Profile. For more information, see "*Registering devices*" and "*Creating a Development Provisioning Profile*."

Updating an App Studio app

When you create a new version of an App Studio app, you must increment the version number or Apple will reject the app.

To prepare to upload a new version of an app on the Apple Developer site, log in at *http://itunesconnect.apple.com*, click **Manage Your Applications**, click the icon for your application, click **Add Version**, and then follow the instructions on the screen. When you specify the new version number, be sure it is greater than the current version number.

To update an App Studio app, enter the new version number in the form when you request a build of the new version of the app.

Submitting an app to Apple

Before you submit your app to the App Store, make sure that you have followed all of the instructions in the "*Submitting an app to Apple*" section. When you are finished:

1 Log in to your Apple developer account at *http://itunesconnect.com*.

2 Click **Manage Your Applications**. The **Manage Your Apps** screen displays.

3 Click the icon for your app. The app description screen displays.

4 Make any desired changes. In the **Uploads** area, upload final screen shots. (To take a screen shot on the iPad, press the Home button and then, while keeping it pressed, press the Power button once. The screen shot will be available in the Photos app.

➡ If you are charging for issues, you must associate your in-app purchases with the app. To do this, click **Edit** in the **In-App Purchases** area, then check all of the in-app purchases you want associated with the app and click **Save**.

5 When you're sure everything is final, click **Save**.

6 Click **Ready to Upload Binary**. If you skip this step, you will not be able to upload your app in step 9.

7 Launch Xcode and choose **Xcode** > **Open Developer Tool** > **Application Loader**.

8 Click **Deliver Your App**.

9 Follow the instructions on the screen. When the **Choose an application** screen displays, choose the name of your app.

10 When the **Application Information** screen displays, click **Choose** and select the .ipa file you downloaded from the App Studio Publishing Portal and click **Open**.

11 Follow the instructions on the screen. If you are asked whether your app contains cryptography, click **No**.

After your app has been successfully submitted, Apple will review it and contact you to let you know when it is approved.

Submitting an Android app

For Android app submission instructions, see the developer site for the target app store.

Legal notices

Index