

## PROJECT

## Virtual Tourist

A part of the iOS Developer Nanodegree Program

## PROJECT REVIEW

## CODE REVIEW 6

## NOTES

## Meets Specifications

SHARE YOUR ACCOMPLISHMENT



## Core Data

- ✓ The app uses a managed object model created in the Xcode Model Editor. A `.xcdatamodeld` model file is present.

Concurrency tested and the project is thread-safe. Excellent!

- ✓ The object model contains both Pin and Photo entities.

- ✓ The object model contains a one-to-many relationship between the Pin and Photo entities, with an appropriate inverse.

## Travel Locations Map View

- ✓ The app contains a map view that allows users to drop pins with a touch and hold gesture.

- ✓ When a pin is tapped, the app transitions to the photo album associated with the pin.

- ✓ When pins are dropped on the map, the pins are persisted as Pin instances in Core Data and the context is saved.

## Photo Album View

- ✓ When a Photo Album View is opened for a pin that does not yet have any photos, it initiates a download from Flickr.

- ✓ The code for downloading photos is in its own class, separate from the `PhotoAlbumViewController`.

- ✓ Images display as they are downloaded. They are shown with placeholders in a collection view while they download, and displayed as soon as possible.

- ✓ The specifics of storing an image is left to Core Data by activating the "Allows External Storage" option.

✓ Once all images have been downloaded, the user can remove photos from the album by tapping the image in the collection view. Tapping the image removes it from the photo album, the booth in the collection view, and Core Data.

✓ The Photo Album view has a button that initiates the download of a new album, replacing the images in the photo album with a new set from Flickr.

✓ If the Photo Album view is opened for a pin that previously had photos assigned, they are immediately displayed. No new download is needed.

 [DOWNLOAD PROJECT](#)

 [CODE REVIEW COMMENTS](#)



[RETURN TO PATH](#)

[Student FAQ](#)