

No description, website, or topics provided.

Edit

Manage topics

 8 commits

 4 branches

 0 releases

 1 contributor

Your recently pushed branches:

 postGeotags (less than a minute ago)

 Compare & pull request

Branch: postGeotags ▾



New pull request








Create new file



Upload files

Find file

Clone or download ▾

This branch is even with LambdaSchool:postGeotags.  Pull request  Compare

 SpencerCurtis Update README.md	Latest commit fb36dc3 on Oct 18, 2018
 LambdaTimeline.xcodeproj	Add starter project 4 months ago
 LambdaTimeline	Add starter project 4 months ago
 .gitignore	Create .gitignore 4 months ago
 Podfile	Add starter project 4 months ago
 Podfile.lock	Add starter project 4 months ago
 README.md	Update README.md 4 months ago

 README.md 

# Lambda Timeline

## Introduction

The goal of this project is to take an existing project called LambdaTimeline and add features to it throughout this sprint.

Today you will be adding geotagging to posts.

## Instructions

Create a new branch in the repository called `postGeotags` and work off of it from where you left off yesterday.

You're welcome to fulfill these instructions however you want. If you'd like suggestions on how to implement something, open the disclosure triangle and there are some suggestions for you. Of course, you can also ask the PMs and instructors for help as well.

1. Add a `geotag: CLLocationCoordinate2D?` property as well. The user should have the choice to geotag posts or not.

2. You will need to make an `MKAnnotation` that represents a `Post`. The annotation's title should be the post's title and the subtitle should be the name of the post's author. You have at least two options on how to do this: a. Change the `Post` object to adopt and conform to the `MKAnnotation` protocol. b. Create a separate object called `PostAnnotation` or something similar that conforms to `MKAnnotation`. The `PostAnnotation` could get its values from a `Post` that is passed into an initializer.

Either way you decide to do this has its pros and cons, so choose whatever makes the most sense to you.

3. Update the `dictionaryRepresentation` and both initializers to account for the property (or properties).
  - **Note:** Firebase will not store a `CLLocationCoordinate2D`, so you must break it up into a key-value pair for both the latitude and longitude in the `dictionaryRepresentation`.
4. Update the `PostController` to account for creating posts with and without geotags.
5. Create a helper class that will take care of requesting location usage authorization from the user, as well as getting their current location in order to geotag a post that is being created. This can be done through using `CLLocationManager` and `CLLocationManagerDelegate`.
  - **Note:** For the base requirements of this project, every post that should be geotagged will just use the user's current location.
6. Update the UI for creating **image and video** posts to allow the user to choose whether to geotag their posts or not.
7. In the `Main.storyboard`, embed the navigation controller in a tab bar controller. Add a new view controller scene with a map view on it. There are a few things that you will have to change now that the tab bar is essentially the first view controller to be displayed once the user is authenticated.

#### ▼ Hints

- As the map view controller is going to need access to the same instance of `PostController` as the rest of the app uses, consider creating a subclass of `UITabBarController` and initializing a `PostController` there instead of the `PostsCollectionViewController`. That way, the tab bar controller can pass references to it to both the `PostsCollectionViewController` and the new map view controller.
  - In the `AppDelegate` the way the navigation controller holding the `PostsCollectionViewController` becomes the initial view controller if the user is authenticated is by initializing it from the storyboard with a `Storyboard ID`. You will need to give the tab bar controller a storyboard ID and use it instead of the navigation controller's that is currently used. If you are unfamiliar with how this works, [this Stack Overflow question](#) gives a straight answer.
8. The map view controller should take the posts from the post controller that have geotags, and place annotations on the map view for each of them.

## Go Further

- Add the ability for the user to select a place to geotag the post at. This can be done a number of ways:
  - Use a map view and let them drop a pin (annotation) on it. You can then get the coordinate from the pin.
  - Use `MKLocalSearch` to let the user search for a location.

You are encouraged to implement both methods if you feel up to it.

- Customize the annotations that are shown on the map view controller to also show the media (image or video) associated with it. **Note:** if you chose to create a `PostAnnotation` object, you may need to modify it so you can have more information than just the post's title and author.

- Add the ability to go directly to the detail view controller of a post (the one with the post's comments) from the annotation.