

Kamcord iOS Programming Challenge

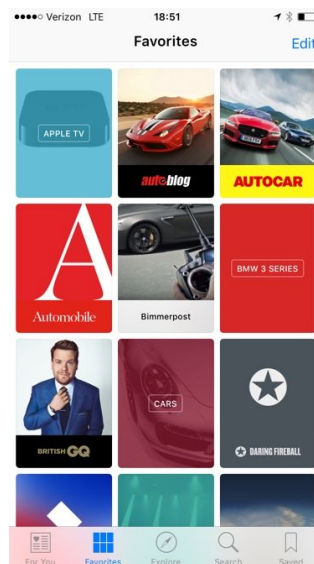
Introduction:

One of our challenges at Kamcord is content discovery. We've added a new tab for users to discover short content videos that we call Shots. The Shots tab allows users to browse through all the featured shots available on Kamcord. You can download the Kamcord app and play around with it @ <http://apple.co/1kT8yXu>.

Specifications:

We would like you to build a simple Shots feed. The shots view should use a UICollectionView, where each item displays one shot. Each cell should display the shot's thumbnail and number of hearts. Keep in mind that on iPad there are 5 items per row, while iPhone only has 3 items per row. Tapping on a cell should play an mp4 video.

Basically similar to iOS News app



The following REST API URL is the end-point for the shots feed shown above:

<https://api.staging.kamcord.com/v1/feed/set/featuredShots>

End-Point Requirements:

- "count" is a required URL parameter for the endpoint. You can utilize that property any way you want.
- A device-token is required in the HTTP Headers. ANY value for that field will suffice (for example: "device-token: abc123;").

You may view the result by using the following command in terminal:

```
curl -X GET --header 'Accept: application/json' --header 'accept-language: en' --header 'device-token: abc123' --header 'client-name: ios' 'https://api.kamcord.com/v1/feed/set/featuredShots?count=20'
```

The response is a JSON dictionary where the value of **response** => **group of cards** is an array of dictionaries. Each dictionary contains the information for one card on Kamcord called **shotCardData**. The keys you will care about are:

response => group => card => shotCardData:

- **heartCount** - number of hearts.
- **shotThumbnail** The URL of the thumbnail image.
- **play => mp4** - video link

The feed information and thumbnails should be loaded on app start. Please use best UI practices when building this shots view (i.e. the app should feel responsive at all times, etc.).

Bonus:

In our sample request, we ask for 20 shots back from the server. Kamcord has tens of thousands of shots available. Therefore, as a **bonus** we'll be adding pagination for when the user can not find a shot they like from the ones fetched.

response:

- **feedId** - The value of this of this key is used to fetch the feed of shots using the Feed API
- **nextPage** - The value of this of this key is used to fetch the next list of shots. For example, if the value is "40.FEATURED_SHOTS.subfeed_featured_shots.40.40", the REST API URL Example will be:

```
curl -X GET --header 'Accept: application/json' --header 'accept-language: en' --header 'device-token: abc123'
--header 'client-name: ios'
'https://api.kamcord.com/v1/feed/ZmVIZElkPWZIZWRfZmVhdHVyZWRFc2hvdCZ1c2VySWQmdG9waWNJZCZzdHJlYW1TZXNzaW9uSWQmbGFuZ3VhZ2VDb2RI?count=20&page=40.FEATURED_SHOTS.subfeed_featured_shots.40.40'
```

Note: you are now calling the `feed` endpoint rather than `feed/set`

What we're looking for:

- How well you follow the design spec
- How you architect your code (Hint: Encapsulation)
- Utilization of Apple's newest APIs and Auto-Layout
- How creative you get in the process
- Consistency and cleanliness of the code
- Performance of the overall app, as well as while **scrolling the collection view**

This exercise should take between 2-3 hours. **Please refrain from using third party frameworks (no cocoapods).** This is an opportunity to show off your iOS skills and we'd hate for other frameworks to steal your spotlight! **You may use the language of your preference, or a mix and match of the two (Obj-C & Swift.)**

Submission:

When you are finished, zip up the Xcode project and send it to us - or use dropbox, google drive, etc (no public GitHub) and send us the link for review.