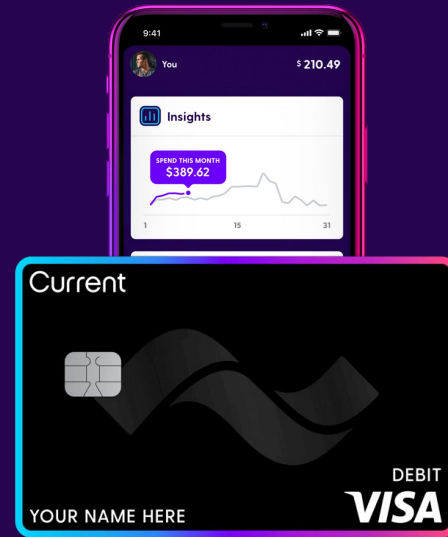




iOS Code Challenge

“Fast Foodz” app



PROJECT OVERVIEW

The goal of the assignment is to build a simple app that finds fast food places nearby in 4 popular categories: Burgers, Pizza, Mexican and Chinese. The app is using the user's location and searches Yelp API to display them in two different views: on the map and in the list. When user selects a place, the app will display driving directions in another screen using Apple's MapKit APIs.

In assignment files, you will find a video of completed app for your reference and a starter Xcode project with all necessary resources included.

PROJECT GUIDELINES

- You can use either Objective-C or Swift (highly preferred)
- You can use either REST or GraphQL API provided by Yelp
- You can use any 3rd party dependencies (Alamofire, Kingfisher, etc)
- You should use default UIKit components unless specified otherwise
- You are **not** required to write any tests
- You are **not** required to handle network connection errors
- **Make sure the project builds before submitting it for review**

... and feel free to contact us if you have any additional questions!

YELP API

To search nearby fast food places, you will use Yelp's **Business Search** API.

Documentation: [REST](#) and [GraphQL](#)

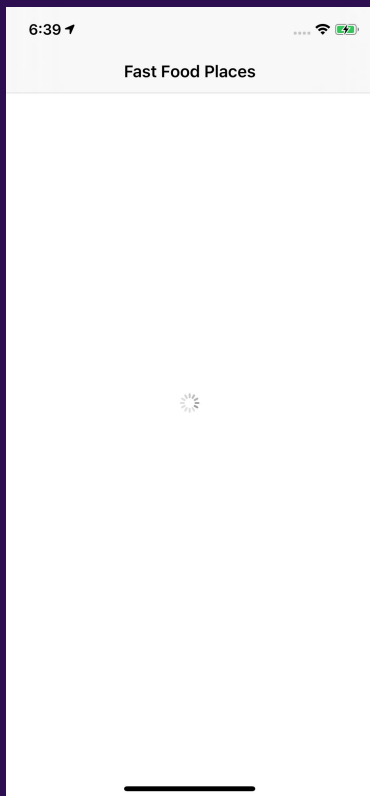
To **authenticate** ([guide](#)), you will need an API key which is located in `APIKey.swift` file inside provided Xcode project. Please, **do not** share this key with anyone and **do not** abuse Yelp's [API Terms of Use](#).

In addition to obtained from user's location latitude and longitude values, you should use the following parameters to query places:

radius = 1000, sort_by = distance, categories = "pizza,mexican,chinese,burgers"

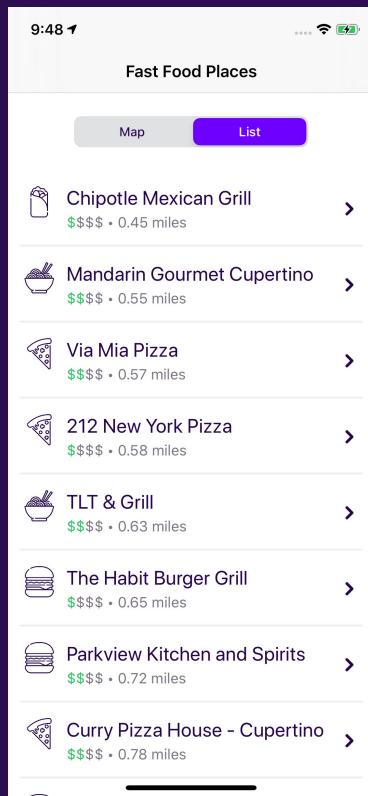
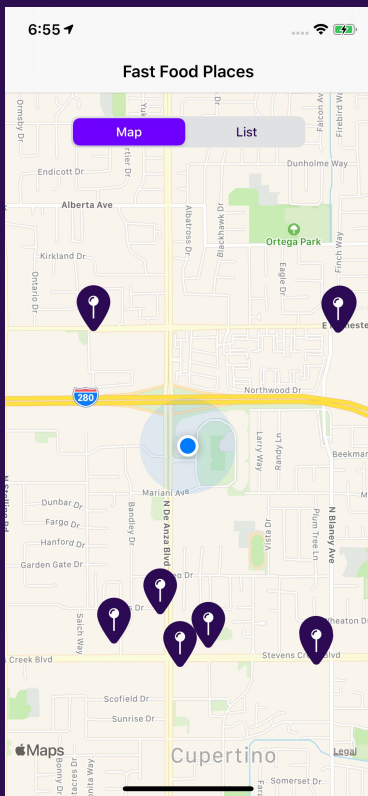
Use (40.758896, -73.985130) as the default value.

LOADING SCREEN



During location retrieval and network request, you should show a medium-style loading indicator centered in the screen.

HOME SCREEN



Once fast food places are fetched, you should show a Home screen which consists of two subscreens: Map and List. Segment control is used to switch between view modes. Last selected mode should be persisted between app launches (map is default).

Segmented control style:

Background color: *london sky*

Selected indicator color: *competition purple*

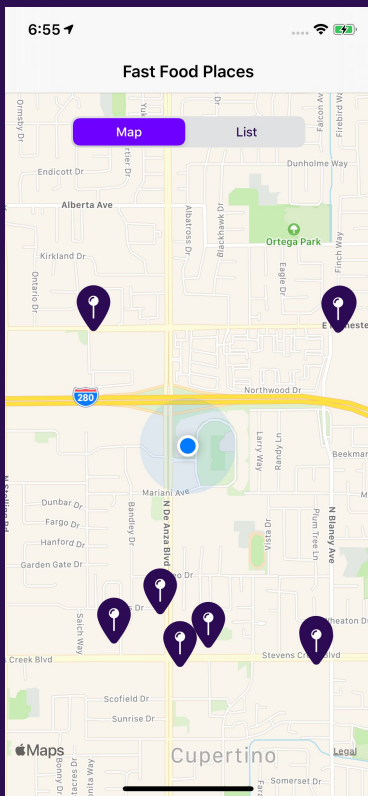
Default text color: *deep indigo*

Selected text color: *white*

Top margin: *24 points*

Horizontally centered

MAP VIEW SCREEN



Map style:

All map controls should be hidden

Initial span in meters: *1000*

Initial center: *user location or default location provided*

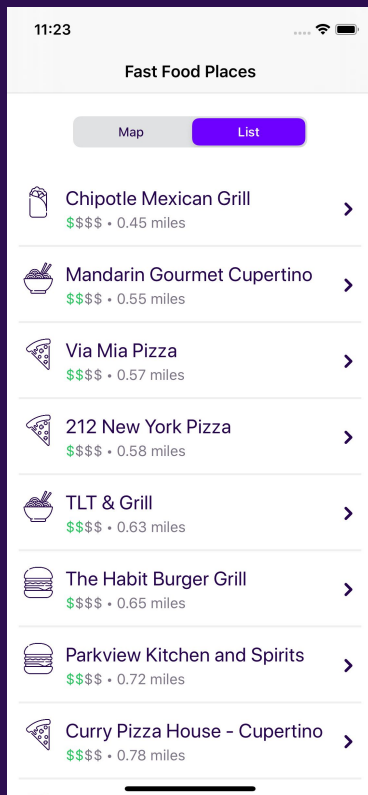
Map pin view style:

Icon image name: *"pin"*

On tap: *medium intensity device feedback (vibration)*

Pins are not draggable and do not show callouts

LIST VIEW SCREEN



List item style:

Item height: *dynamic*

Content padding: *16 points*

Highlight color: *powder blue*

Separator color: *london sky*

Separator height: *2 points*

Separator horizontal margins: *12 points*

Icon size: *32x32 points*

Icon default color: *deep indigo*

Title font and color: *title3, deep indigo*

Info font and color: *subheadline, lilac grey*

Info dollar color: *pickle green*

Spacing between labels: *6 points*

Spacing between labels and icon: *12 points*

Arrow indicator color: *deep indigo*

Arrow vertical position: *centered*

DETAILS SCREEN

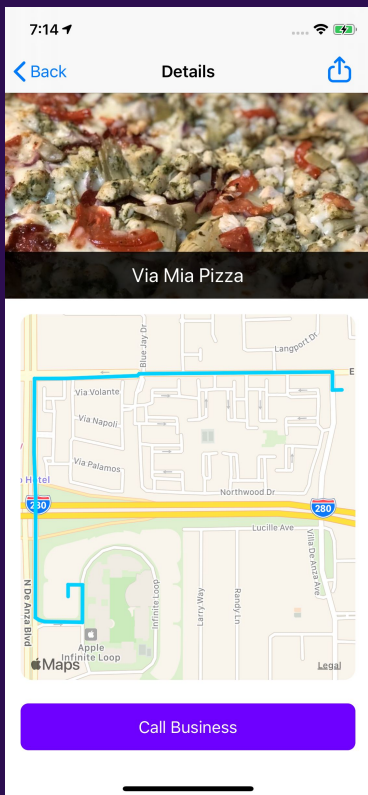


Image style:

Aspect ratio: *16:9* Background color: *london sky*

Name label style:

Background color: *black, 0.85 alpha* Font and color: *title3, white*
Margins: *horizontal - 16 points, vertical - 12 points*

Map style:

Corner radius: *12 points* Directions line color and width: *blu cepheus, 4 points*
Map should show driving directions from user's location to place location using MapKit API
([Documentation](#))

Button style:

Height: *48 points* Corner radius: *6 points* Background color: *competition purple* Title color: *white*

Share button style:

Image name: *"share"*
Should open default iOS Share sheet with business's Yelp page URL

Spacing:

Map's left, top, right margins: *16 points* Map and button spacing: *24 points*
Button's left, bottom, right margins: *16 points*

EXTRA CREDIT TASKS

1. Add a cross dissolve transition when switching between Map and List views **without** using *UIView.transition* methods
2. (For iOS Ninjas) Color icons in places list view based on provided gradient colors (in *Colors.swift* file, colors are in top to bottom order).

Imagine a static vertical line colored with evenly distributed gradient colors having the same height as the list view screen. Based on icon's center Y-axis position in the list, it will take a color on the gradient line, which should be dynamically applied during scrolling.

(see included example video for reference)

GOOD LUCK!

