



iOS Application Development

Lecture 9: Working with the Web and Threading

Simon Völker & Philipp Wacker
Media Computing Group
RWTH Aachen University

hci.rwth-aachen.de/ios



RWTHAACHEN
UNIVERSITY

Seminar Review Meetings

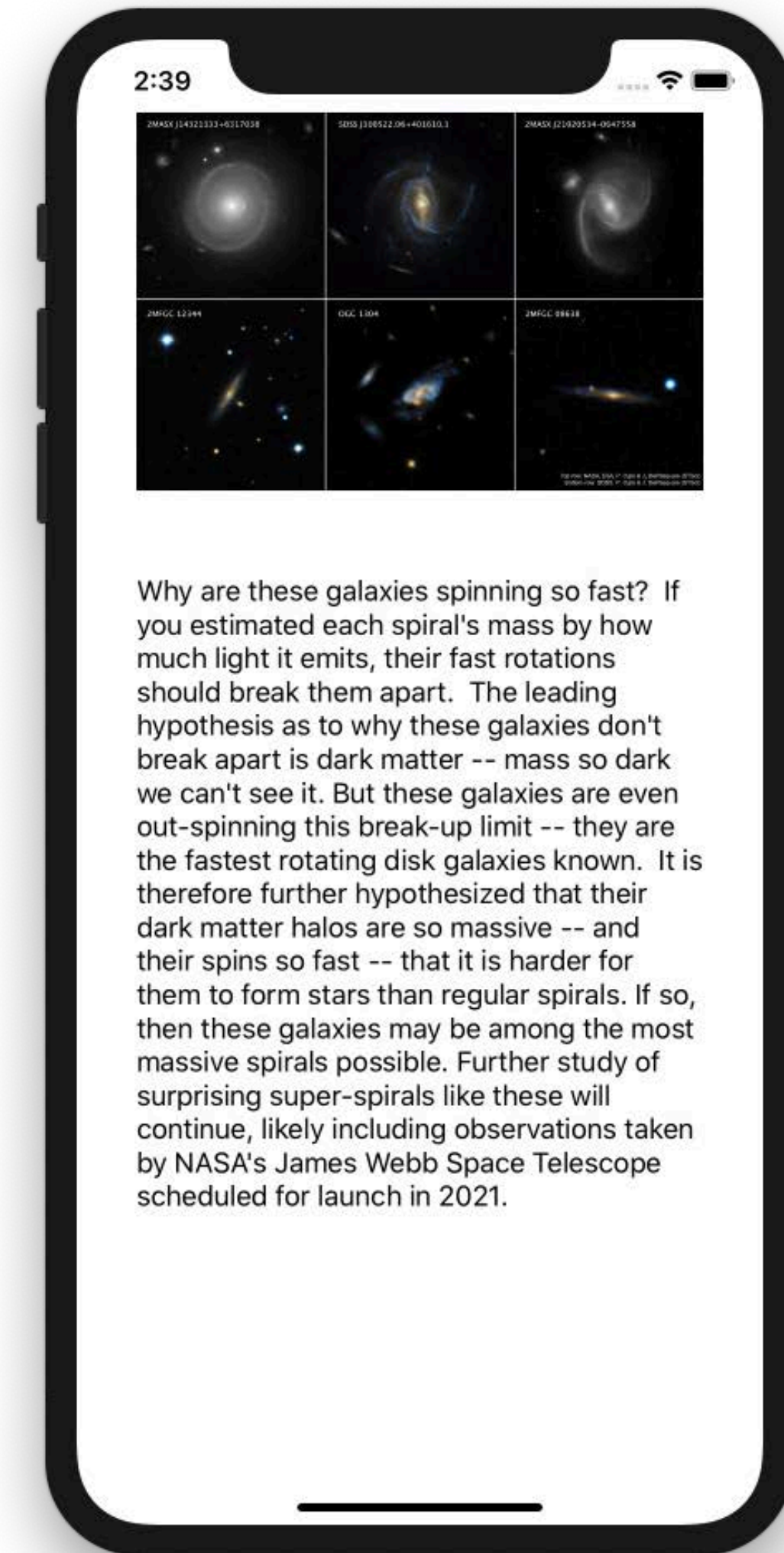
- Next weeks seminar meetings:

Topic	Group	Presentation	Discussions
MapKit, CoreLocation	Group 8	18.11.19, first group	11.11.19
Debugging in Xcode and Instruments	Group 4	18.11.19, second group	11.11.19
SiriKit	Group 2	19.11.19, second group	12.11.19
Extensions & Inter-App communication	Group 5	19.11.19, first group	12.11.19
WatchOS	Group 14	25.11.19, second group	18.11.19
Swift UI	Group 11	25.11.19, first group	18.11.19
Scene Kit	Group 7	26.11.19, first group	19.11.19
GameplayKit	Group 9	26.11.19, second group	19.11.19
RealityKit & Reality Composer	Group 1	09.12.19, first group	02.12.19
Networking on iOS	Group 6	2.12.19, first group	25.11.19
ClassKit	Group 3	2.12.19, second group	25.11.19
Bringing People into AR	Group 13	09.12.19, second group	02.12.19
Core Data	Group 10	3.12.19, first group	26.11.19
Core ML + Create ML	Group 12	3.12.19, second group	26.11.19



Querying Web Content

1. Create Url
2. Request Data with API keys
3. Create a Swift model
4. Decode JSON
5. Update UI



Working with the Web

- URL: `https://itunes.apple.com/us/app/keynote/id409183694?mt=12`
Protocol Subdomain Domain Path Query

- Request Types:
 - Most common are GET and POST
- Headers (e.g. User-Agent)
- Body

Create a URL

- URL:

```
let url = URL(string: "https://www.apple.com")!
```

- Create a data task and the request:

```
let task = URLSession.shared.dataTask(with: url) { (data,  
response, error) in  
  
    if let data = data {  
        print(data) // 46472 bytes  
    }  
}  
task.resume()
```

runs on a background thread

Create a URL

- URL:

```
let url = URL(string: "https://www.apple.com")!
```

- **Create a data task and the request:**

```
let task = URLSession.shared.dataTask(with: url) { (data,  
    response, error) in  
  
    if let data = data {  
        if let string = String(data: data, encoding: .utf8) {  
            print(string) // html code  
        }  
    }  
}  
task.resume()
```

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en-US" lang="en-US" prefix="og: http://ogp.me/ns#" class="no-js">
<head>

<meta charset="utf-8" />
<link rel="canonical" href="https://www.apple.com/" />

<link rel="alternate" href="https://www.apple.com/" hreflang="en-US" /><link rel="alternate" href="https://www.apple.com/ae-ar/" hreflang="ar-AE" /><link rel="alternate" href="https://www.apple.com/ae/" hreflang="en-AE" /><link rel="alternate" href="https://www.apple.com/am/" hreflang="en-AM" /><link rel="alternate" href="https://www.apple.com/at/" hreflang="de-AT" /><link rel="alternate" href="https://www.apple.com/au/" hreflang="en-AU" /><link rel="alternate" href="https://www.apple.com/be-fr/" hreflang="fr-BE" /><link rel="alternate" href="https://www.apple.com/benl/" hreflang="nl-BE" /><link rel="alternate" href="https://www.apple.com/bg/" hreflang="bg-BG" /><link rel="alternate" href="https://www.apple.com/bh/" hreflang="en-BH" /><link rel="alternate" href="https://www.apple.com/br/" hreflang="pt-BR" /><link rel="alternate" href="https://www.apple.com/bw/" hreflang="en-BW" /><link rel="alternate" href="https://www.apple.com/ca/" hreflang="en-CA" /><link rel="alternate" href="https://www.apple.com/ca-fr/" hreflang="fr-CA" /><link rel="alternate" href="https://www.apple.com/cf/" hreflang="fr-CF" /><link rel="alternate" href="https://www.apple.com/che-de/" hreflang="de-CH" /><link rel="alternate" href="https://www.apple.com/cht-fr/" hreflang="fr-CH" /><link rel="alternate" href="https://www.apple.com/ci/" hreflang="fr-CI" /><link rel="alternate" href="https://www.apple.com/cl/" hreflang="es-CL" /><link rel="alternate" href="https://www.apple.com/cm/" hreflang="fr-CM" /><link rel="alternate" href="https://www.apple.com/cn/" hreflang="zh-CN" /><link rel="alternate" href="https://www.apple.com/co/" hreflang="es-CO" /><link rel="alternate" href="https://www.apple.com/cz/" hreflang="cs-CZ" /><link rel="alternate" href="https://www.apple.com/de/" hreflang="de-DE" /><link rel="alternate" href="https://www.apple.com/dk/" hreflang="da-DK" /><link rel="alternate" href="https://www.apple.com/ee/" hreflang="et-EE" /><link rel="alternate" href="https://www.apple.com/eg/" hreflang="en-EG" /><link rel="alternate" href="https://www.apple.com/es/" hreflang="es-ES" /><link rel="alternate" href="https://www.apple.com/fin-fi/" hreflang="fi-FI" /><link rel="alternate" href="https://www.apple.com/fr/" hreflang="fr-FR" /><link rel="alternate" href="https://www.apple.com/gn/" hreflang="fr-GN" /><link rel="alternate" href="https://www.apple.com/gg/" hreflang="fr-GQ" /><link rel="alternate" href="https://www.apple.com/gr/" hreflang="el-GR" /><link rel="alternate" href="https://www.apple.com/gw/" hreflang="en-GW" /><link rel="alternate" href="https://www.apple.com/hk/" hreflang="zh-HK" /><link rel="alternate" href="https://www.apple.com/hk/en/" hreflang="en-HK" /><link rel="alternate" href="https://www.apple.com/hr/" hreflang="hr-HR" /><link rel="alternate" href="https://www.apple.com/hu/" hreflang="hu-HU" /><link rel="alternate" href="https://www.apple.com/id/" hreflang="id-ID" /><link rel="alternate" href="https://www.apple.com/ie/" hreflang="en-IE" /><link rel="alternate" href="https://www.apple.com/il/" hreflang="en-IL" /><link rel="alternate" href="https://www.apple.com/in/" hreflang="en-IN" /><link rel="alternate" href="https://www.apple.com/it/" hreflang="it-IT" /><link rel="alternate" href="https://www.apple.com/jo/" hreflang="en-JO" /><link rel="alternate" href="https://www.apple.com/jp/" hreflang="ja-JP" /><link rel="alternate" href="https://www.apple.com/ke/" hreflang="en-KE" /><link rel="alternate" href="https://www.apple.com/kr/" hreflang="ko-KR" /><link rel="alternate" href="https://www.apple.com/kw/" hreflang="en-KW" /><link rel="alternate" href="https://www.apple.com/la/" hreflang="es-419" /><link rel="alternate" href="https://www.apple.com/lae/" hreflang="en-419" /><link rel="alternate" href="https://www.apple.com/li/" hreflang="de-LI" /><link rel="alternate" href="https://www.apple.com/lt/" hreflang="lt-LT" /><link rel="alternate" href="https://www.apple.com/lu/" hreflang="fr-LU" /><link rel="alternate" href="https://www.apple.com/lv/" hreflang="lv-LV" /><link rel="alternate" href="https://www.apple.com/ma/" hreflang="fr-MA" /><link rel="alternate" href="https://www.apple.com/md/" hreflang="en-MD" /><link rel="alternate" href="https://www.apple.com/me/" hreflang="en-ME" /><link rel="alternate" href="https://www.apple.com/mg/" hreflang="fr-MG" /><link rel="alternate" href="https://www.apple.com/mk/" hreflang="en-MK" /><link rel="alternate" href="https://www.apple.com/ml/" hreflang="fr-ML" /><link rel="alternate" href="https://www.apple.com/mo/" hreflang="zh-MO" /><link rel="alternate" href="https://www.apple.com/mt/" hreflang="en-MT" /><link rel="alternate" href="https://www.apple.com/mu/" hreflang="fr-MU" /><link rel="alternate" href="https://www.apple.com/mx/" hreflang="es-MX" /><link rel="alternate" href="https://www.apple.com/my/" hreflang="en-MY" /><link rel="alternate" href="https://www.apple.com/mz/" hreflang="en-MZ" /><link rel="alternate" href="https://www.apple.com/ne/" hreflang="fr-NE" /><link rel="alternate" href="https://www.apple.com/ng/" hreflang="en-NG" /><link rel="alternate" href="https://www.apple.com/nl/" hreflang="nl-NL" /><link rel="alternate" href="https://www.apple.com/no/" hreflang="no-NO" /><link rel="alternate" href="https://www.apple.com/nz/" hreflang="en-NZ" /><link rel="alternate" href="https://www.apple.com/om/" hreflang="en-OM" /><link rel="alternate" href="https://www.apple.com/ph/" hreflang="en-PH" /><link rel="alternate" href="https://www.apple.com/pl/" hreflang="pl-PL" /><link rel="alternate" href="https://www.apple.com/pt/" hreflang="pt-PT" /><link rel="alternate" href="https://www.apple.com/qa/" hreflang="en-QA" /><link rel="alternate" href="https://www.apple.com/ro/" hreflang="ro-RO" /><link rel="alternate" href="https://www.apple.com/ru/" hreflang="ru-RU" /><link rel="alternate" href="https://www.apple.com/sa-ar/" hreflang="ar-SA" /><link rel="alternate" href="https://www.apple.com/sa/" hreflang="en-SA" /><link rel="alternate" href="https://www.apple.com/se/" hreflang="sv-SE" /><link rel="alternate" href="https://www.apple.com/sg/" hreflang="en-SG" /><link rel="alternate" href="https://www.apple.com/si/" hreflang="en-SI" /><link rel="alternate" href="https://www.apple.com/sk/" hreflang="sk-SK" /><link rel="alternate" href="https://www.apple.com/sn/" hreflang="fr-SN" /><link rel="alternate" href="https://www.apple.com/th/" hreflang="th-TH" /><link rel="alternate" href="https://www.apple.com/tn/" hreflang="fr-TN" /><link rel="alternate" href="https://www.apple.com/tr/" hreflang="tr-TR" /><link rel="alternate" href="https://www.apple.com/tw/" hreflang="zh-TW" /><link rel="alternate" href="https://www.apple.com/ug/" hreflang="en-UG" /><link rel="alternate" href="https://www.apple.com/uk/" hreflang="en-GB" /><link rel="alternate" href="https://www.apple.com/vn/" hreflang="en-VN" /><link rel="alternate" href="https://www.apple.com/za/" hreflang="en-ZA" />

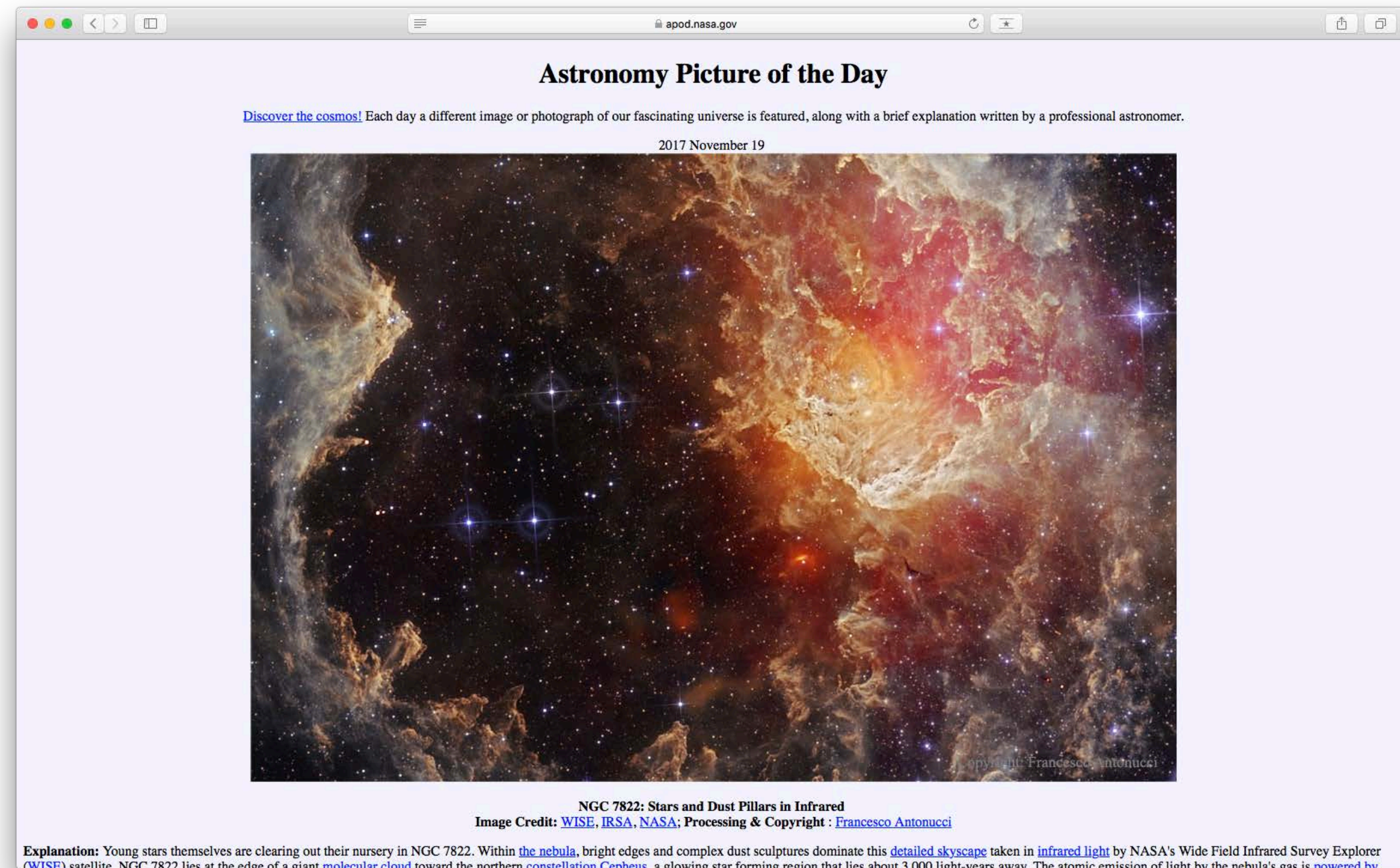
<meta name="viewport" content="width=device-width, initial-scale=1, viewport-fit=cover" />

<link rel="stylesheet" type="text/css" href="/ac/globalnav/3/en_US/styles/ac-globalnav.built.css" />
<link rel="stylesheet" type="text/css" href="/ac/localnav/3.0/styles/ac-localnav.built.css" />
<link rel="stylesheet" type="text/css" href="/ac/globalfooter/3/en_US/styles/ac-globalfooter.built.css" />
<link rel="stylesheet" type="text/css" href="/ac/localeswitcher/1/styles/localeswitcher.built.css" />

<title>Apple</title>
<meta property="analytics-track" content="Apple - Index/Tab" />
<meta property="analytics-s-channel" content="homepage" />
<meta property="analytics-s-bucket-0" content="appleglobal,applehome" />
<meta property="analytics-s-bucket-1" content="apple{COUNTRY_CODE}global,apple{COUNTRY_CODE}home" />
<meta property="analytics-s-bucket-2" content="apple{COUNTRY_CODE}global" />
<meta property="analytics-s-bucket-store" content="applestoreww,applestoreus" />
<meta name="Description" content="Discover the innovative world of Apple and shop everything iPhone, iPad, Apple Watch, Mac, and Apple TV, plus explore accessories, entertainment, and expert device support." />
<meta property="og:title" content="Apple" />
<meta property="og:description" content="Discover the innovative world of Apple and shop everything iPhone, iPad, Apple Watch, Mac, and Apple TV, plus explore accessories, entertainment, and expert device support." />
<meta property="og:url" content="https://www.apple.com/" />
<meta property="og:locale" content="en_US" />
<meta property="og:image" content="https://www.apple.com/ac/structured-data/images/open_graph_logo.png?20170911434" />
<meta property="og:type" content="website" />
<meta property="og:site_name" content="Apple" />
<link rel="stylesheet" href="/wss/fonts?families=SF+Pro,v1|SF+Pro+Icons,v1" type="text/css" media="all" />
<link rel="stylesheet" href="https://images.apple.com/v/home/dn/built/styles/main.built.css" type="text/css" />
<script src="https://images.apple.com/v/home/dn/built/scripts/head.built.js" type="text/javascript" charset="utf-8"></script>
</head>
<body class="page-home ac-nav-overlap">
<aside id="ac-localewitcher" data-analytics-region="locale switcher">
<div class="ac-ls-content">
<div class="ac-ls-copy">Actualmente estás en el sitio de EE.UU. Elige otro país o región para ver contenido específico según tu ubicación.</div>
</div>
</aside>
</body>
</html>
```



Working with an API



NASA Astronomy Picture of the Day (APOD) API

<https://api.nasa.gov/api.html#apod>

Working with an APOD API

- Use the demo key **DEMO_KEY** to get the image of the day

```
let url = URL(string: "https://api.nasa.gov/planetary/apod?api_key=DEMO_KEY")!
let task = URLSession.shared.dataTask(with: url) { (data, response, error) in
    if let data = data,
        let string = String(data: data, encoding: .utf8) {
        print(string)
    }
}
task.resume()
```

```
{
  "copyright": "Francesco Antonucci",
  "date": "2017-11-19",
  "explanation": "Young stars themselves are clearing out their nursery in NGC 7822. Within the nebula, bright edges and complex dust sculptures dominate this detailed skyscape taken in infrared light by NASA's Wide Field Infrared Survey Explorer (WISE) satellite. NGC 7822 lies at the edge of a giant molecular cloud toward the northern constellation Cepheus, a glowing star forming region that lies about 3,000 light-years away. The atomic emission of light by the nebula's gas is powered by energetic radiation from the hot stars, whose powerful winds and light also sculpt and erode the denser pillar shapes. Stars could still be forming inside the pillars by gravitational collapse, but as the pillars are eroded away, any forming stars will ultimately be cut off from their reservoir of star stuff. This field spans around 40 light-years at the estimated distance of NGC 7822. Free APOD Lectures: January 5 in NYC & January 9 in DC",
  "hdurl": "https://apod.nasa.gov/apod/image/1711/ngc7822_WISEantonucci_1824c.jpg",
  "media_type": "image",
  "service_version": "v1",
  "title": "NGC 7822: Stars and Dust Pillars in Infrared",
  "url": "https://apod.nasa.gov/apod/image/1711/ngc7822_WISEantonucci_1080c.jpg"
}
```



Modify a URL with URL Components

```
let url = URL(string: "https://api.nasa.gov/planetary/apod?api_key=DEMO_KEY")!
```

```
let baseURL = URL(string: "https://api.nasa.gov/planetary/apod")!
```

```
let query: [String: String] = [  
    "api_key": "DEMO_KEY"  
]
```

```
extension URL {  
    func withQueries(_ queries: [String: String]) -> URL? {  
        var components = URLComponents(url: self,  
                                         resolvingAgainstBaseURL: true)  
        components?.queryItems = queries.flatMap  
            { URLQueryItem(name: $0.0, value: $0.1) }  
        return components?.url  
    }  
}
```

```
let url = baseURL.withQueries(query)!
```

Working with an APOD API

```
let baseURL = URL(string: "https://api.nasa.gov/planetary/apod")!

let query: [String: String] = [
    "api_key": "DEMO_KEY"
]

let url = baseURL.withQueries(query)!
let task = URLSession.shared.dataTask(with: url) { (data,
response, error) in
    if let data = data,
        let string = String(data: data, encoding: .utf8) {
        print(string)
    }
}

task.resume()
```

```
{
  "copyright": "Francesco Antonucci",
  "date": "2017-11-19",
  "explanation": "Young stars themselves are clearing out their nursery in NGC 7822. Within the nebula, bright edges and complex dust sculptures dominate this detailed skyscape taken in infrared light by NASA's Wide Field Infrared Survey Explorer (WISE) satellite. NGC 7822 lies at the edge of a giant molecular cloud toward the northern constellation Cepheus, a glowing star forming region that lies about 3,000 light-years away. The atomic emission of light by the nebula's gas is powered by energetic radiation from the hot stars, whose powerful winds and light also sculpt and erode the denser pillar shapes. Stars could still be forming inside the pillars by gravitational collapse, but as the pillars are eroded away, any forming stars will ultimately be cut off from their reservoir of star stuff. This field spans around 40 light-years at the estimated distance of NGC 7822. Free APOD Lectures: January 5 in NYC & January 9 in DC",
  "hdurl": "https://apod.nasa.gov/apod/image/1711/ngc7822_WISEantonucci_1824c.jpg",
  "media_type": "image",
  "service_version": "v1",
  "title": "NGC 7822: Stars and Dust Pillars in Infrared",
  "url": "https://apod.nasa.gov/apod/image/1711/ngc7822_WISEantonucci_1080c.jpg"
}
```



1. Create Url ✓



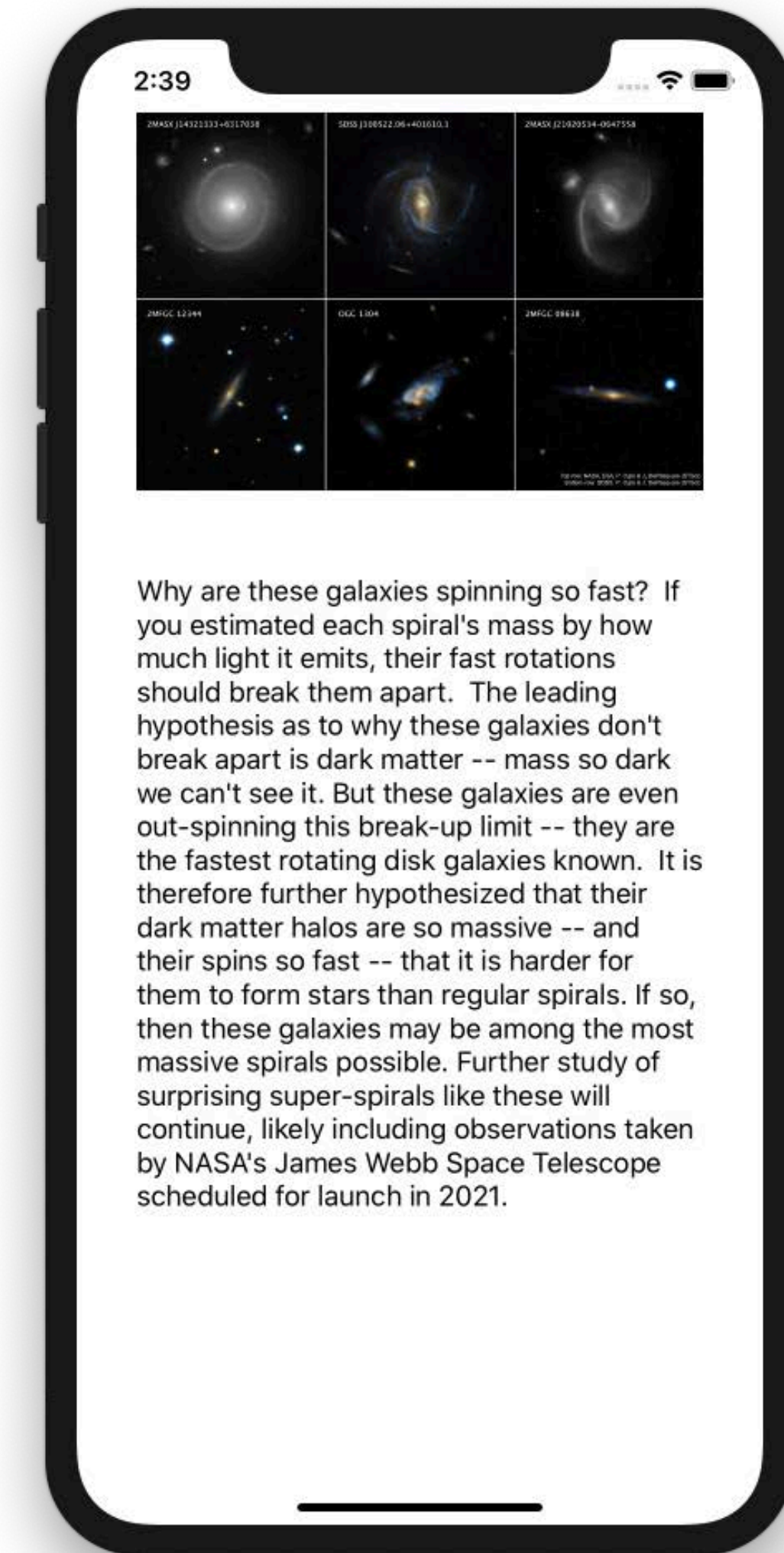
2. Request Data with API keys ✓



3. Create a Swift model

4. Decode JSON

5. Update UI



The Swift model

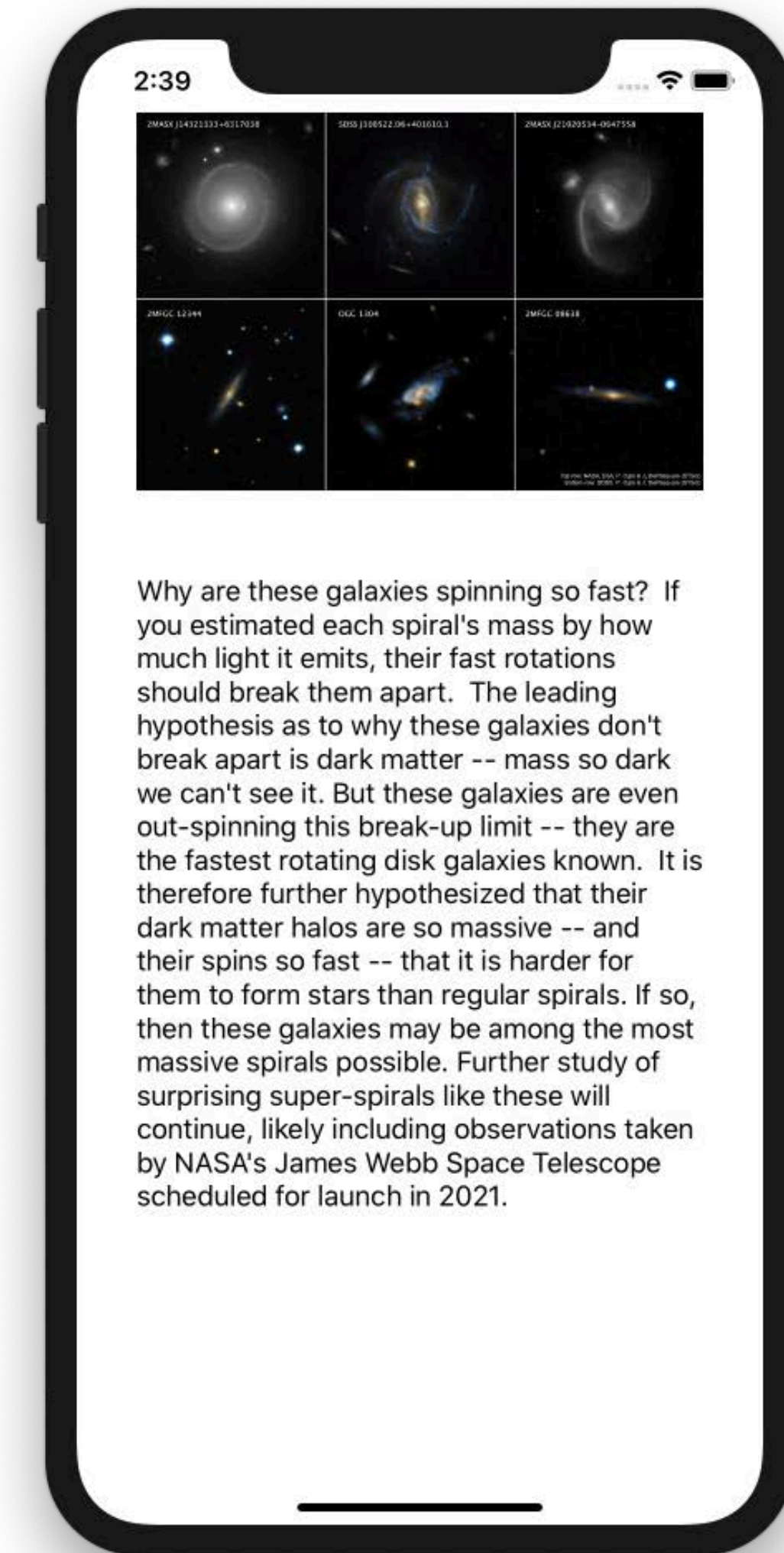
- The PhotoInfo model:

```
struct PhotoInfo: Codable {
    var title: String
    var description: String
    var url: URL

    enum Keys: String, CodingKey {
        case title
        case description = "explanation"
        case url
    }

    init(from decoder: Decoder) throws {
        ...
    }
}
```

1. Create Url ✓
2. Request Data with API keys ✓
3. Create a Swift model ✓
4. Decode JSON
5. Update UI



Decode JSON

```
let baseURL = URL(string: "https://api.nasa.gov/planetary/apod")!

let query: [String: String] = [
    "api_key": "DEMO_KEY",
]

let url = baseURL.withQueries(query)!
let task = URLSession.shared.dataTask(with: url) { (data,
response, error) in
    if let data = data,
        let photoInfo = try? jsonDecoder.decode(PhotoInfo.self, from: data) {
            print(photoInfo)
        }
    }
}

task.resume()
```


Decode JSON

```
func fetchPhotoInfo() -> PhotoInfo {
    let baseURL = URL(string: "https://api.nasa.gov/planetary/apod")!

    let query: [String: String] = ["api_key": "DEMO_KEY",]

    let url = baseURL.withQueries(query)!
    let task = URLSession.shared.dataTask(with: url) { (data, response, error) in
        let jsonDecoder = JSONDecoder()
        if let data = data,
            let photoInfo = try?
                jsonDecoder.decode(PhotoInfo.self, from: data) {
            return photoInfo
        }
    }
    task.resume()
}
```



Unexpected non-void return value in void function

Completion Handler

```
func performLongRunningOperation(completion: @escaping () ->
Void) {
    // Code that performs a long running operation
    completion()
}
```

```
func performLongRunningOperation(completion: @escaping (String) -> Void) {
    // Code that performs a long running operation that creates a `result: String`
variable
    completion(result)
}
```

```
func performLongRunningOperation(completion: @escaping (String,
Int, Data) -> Void) {
    // Code that performs a long running operation that creates `result: String`,
`code: Int`, `data: Data` variables
    completion(result, code, data)
}
```

Decode JSON

```
func fetchPhotoInfo(completion: @escaping (PhotoInfo?) -> Void)
{
    let baseURL = URL(string: "https://api.nasa.gov/planetary/apod")!
    let query: [String: String] = ["api_key": "DEMO_KEY",]

    let url = baseURL.withQueries(query)!
    let task = URLSession.shared.dataTask(with: url) { (data,
        response, error) in
        let jsonDecoder = JSONDecoder()
        if let data = data,
            let photoInfo = try? jsonDecoder.decode(PhotoInfo.self, from: data) {
                completion(photoInfo)
            }
        else {
            print("Either no data was returned, or data was not serialized.")
            completion(nil)
        }
    }
    task.resume()
}
```

Decode JSON

```
override func viewDidLoad() {  
    super.viewDidLoad()  
  
    self.fetchPhotoInfo { (photoInfo) in  
        if let photoInfo = photoInfo {  
  
            // Do Something with the photoInfo  
  
        }  
    }  
}
```

1. Create Url



2. Request Data with API keys



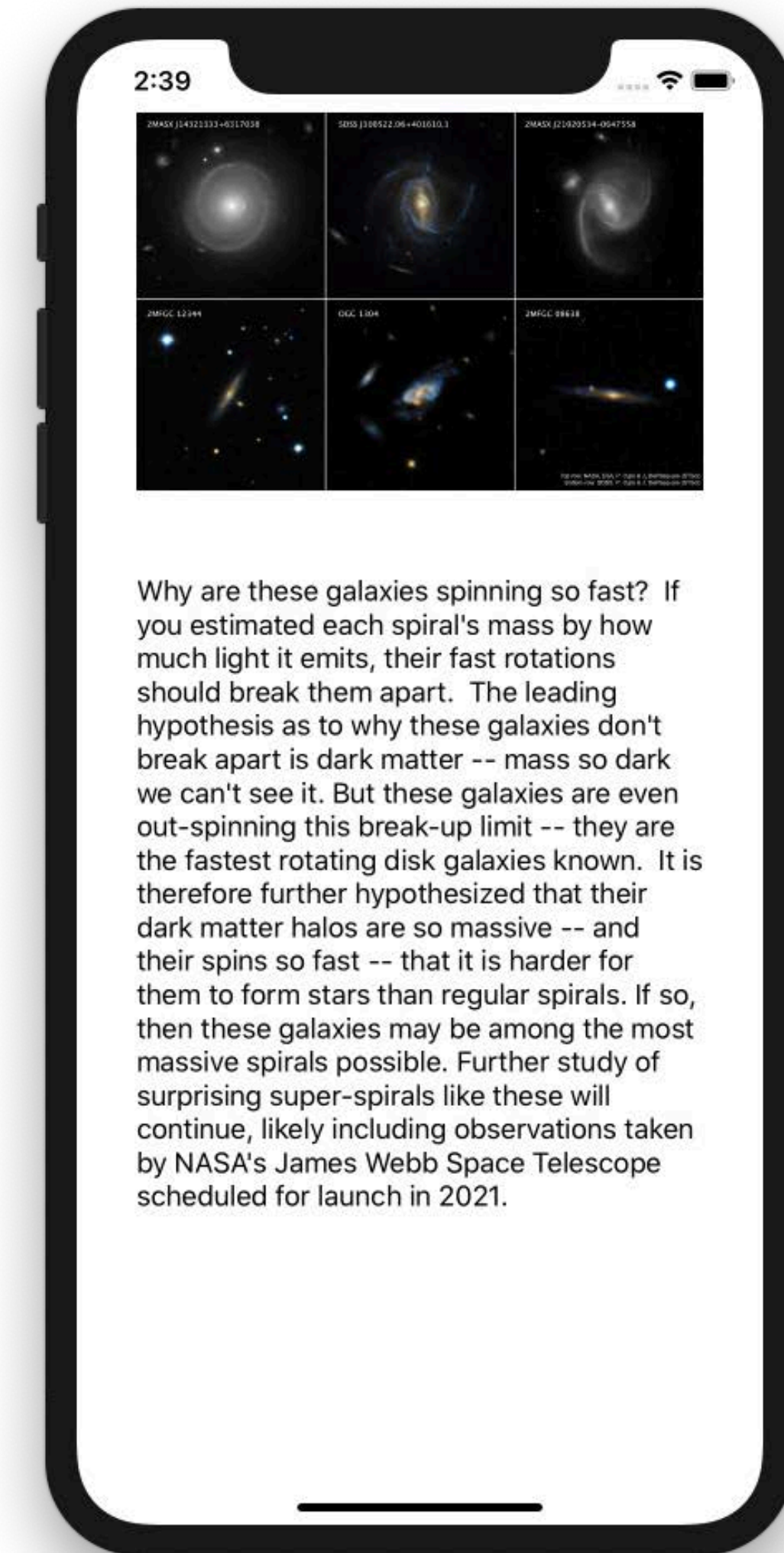
3. Create a Swift model



4. Decode JSON



5. Update UI



Update UI

```
NSURLSession.shared.dataTask(with: url) { (data, response, error) in
...

self.fetchPhotoInfo { (photoInfo) in
    if let photoInfo = photoInfo {
        self.title = photoInfo.title
        self.descriptionLabel.text = photoInfo.description
    }
}
```



Main Thread Checker: UI API called on a background thread: : -[UILabel setText:] ...

Multi Threading in iOS

- Run multiple tasks at the same time
- Run slow or expensive tasks in the background
- Free the main thread so it responds to the UI



Synchronous and Asynchronous

- Synchronous
 - One task completes before another begins
 - Ties up the main thread (main queue)
- Asynchronous
 - Multiple tasks run simultaneously on multiple threads (concurrency)
 - Tasks run in the background thread (background queue)
 - Frees up the main thread



Grand Central Dispatch

- Allows your app to execute multiple tasks concurrently on multiple threads
- Assigns tasks to "dispatch queues" and assigns priority
- Controls when your code is executed



Grand Central Dispatch

- Main queue
 - Created when an app launches
 - Highest priority
 - Used to update the UI and respond quickly to user input
- Background queues
 - Lower-priority
 - Used to run long-running operations



Dispatch Queue

```
DispatchQueue.global(qos: .background).async {  
    // Do some background work  
    DispatchQueue.main.async {  
        // Update the UI to indicate the work has been completed  
    }  
}
```

Grand Central Dispatch Demo



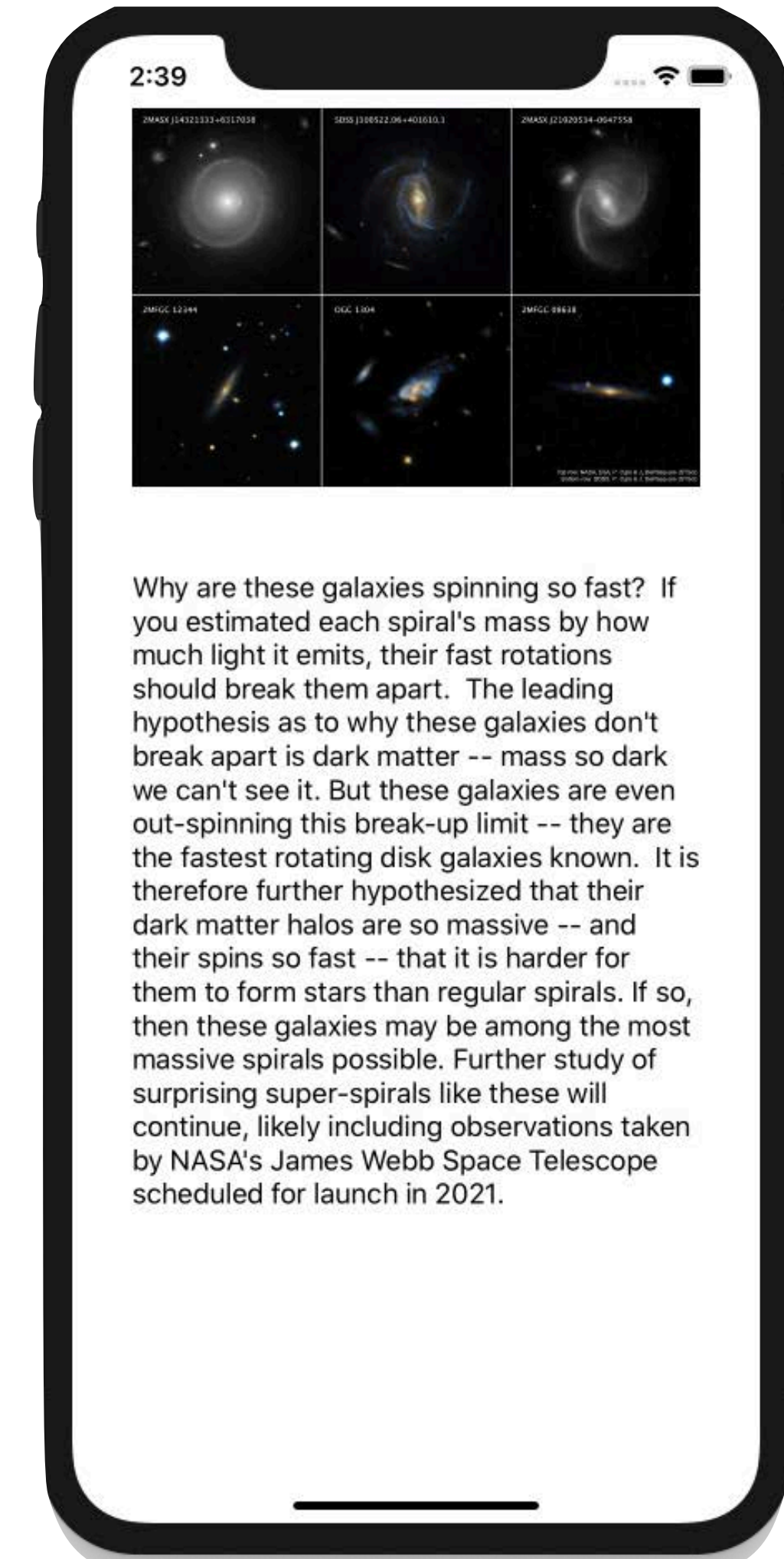
Update UI

```
self.fetchPhotoInfo { (photoInfo) in
    if let photoInfo = photoInfo {
        DispatchQueue.main.async {
            self.title = photoInfo.title
            self.descriptionLabel.text = photoInfo.description
        }
    }
}
```

Update UI

```
func updateUI(photoInfo: PhotoInfo) {
    let task = URLSession.shared.dataTask(with: photoInfo.url,
        completionHandler: { (data, response, error) in
            if let data = data, let image = UIImage(data: data) {
                DispatchQueue.main.async {
                    self.title = photoInfo.title
                    self.imageView.image = image
                    self.descriptionLabel.text = photoInfo.description
                }
            }
        })
    task.resume()
}
```

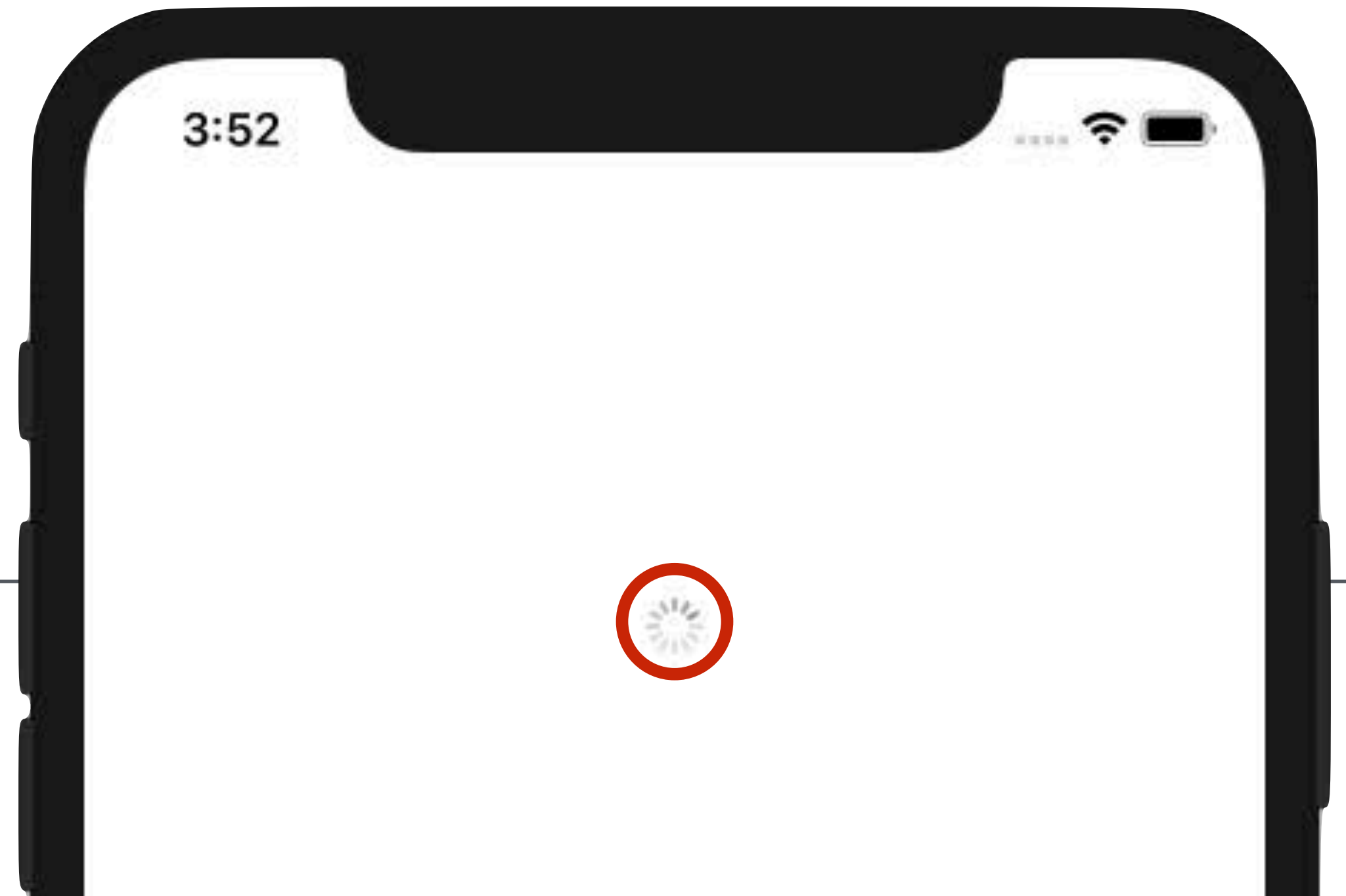
1. Create Url ✓
2. Request Data with API keys ✓
3. Create a Swift model ✓
4. Decode JSON into swift type ✓
5. Update UI ✓



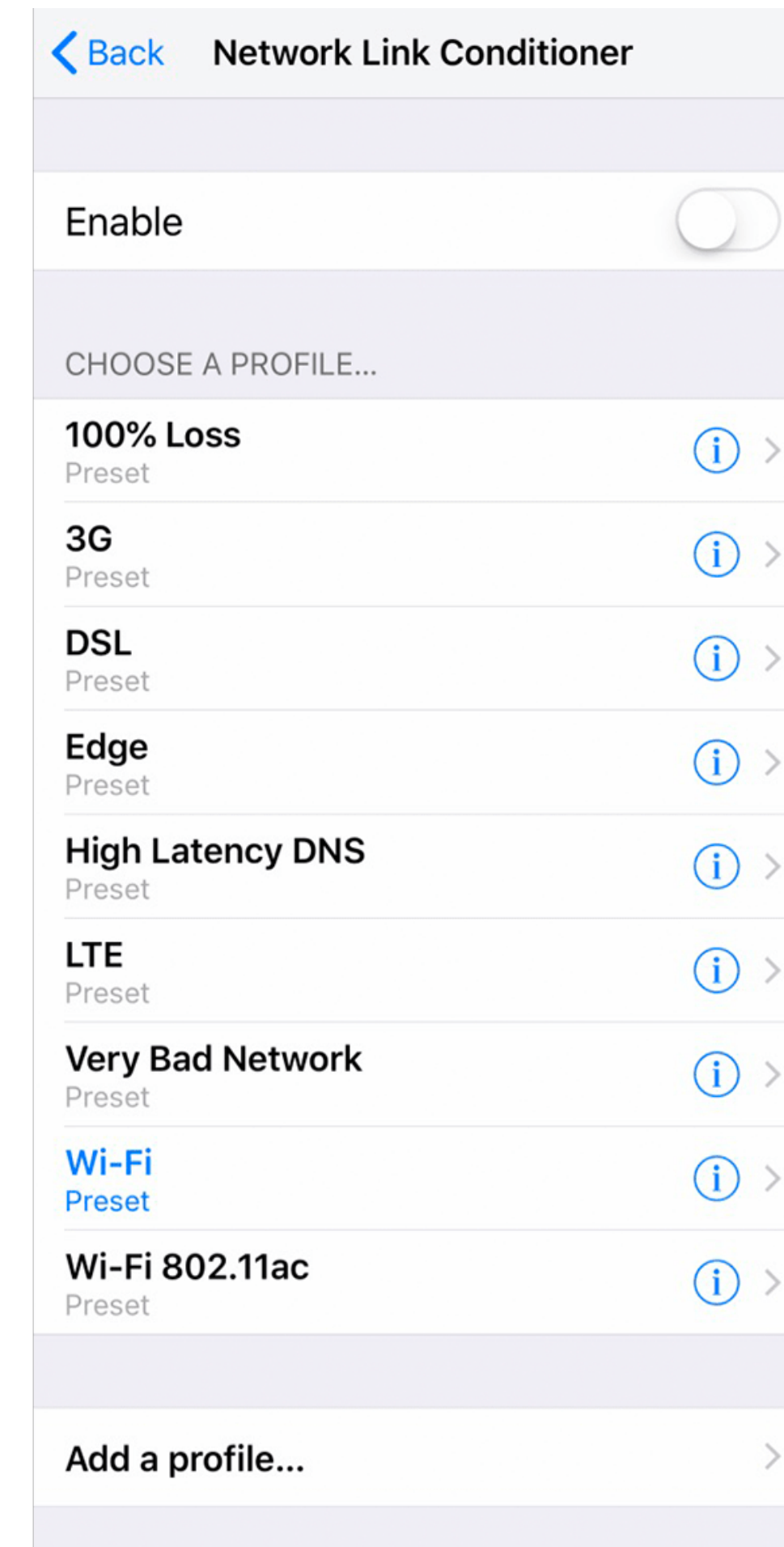
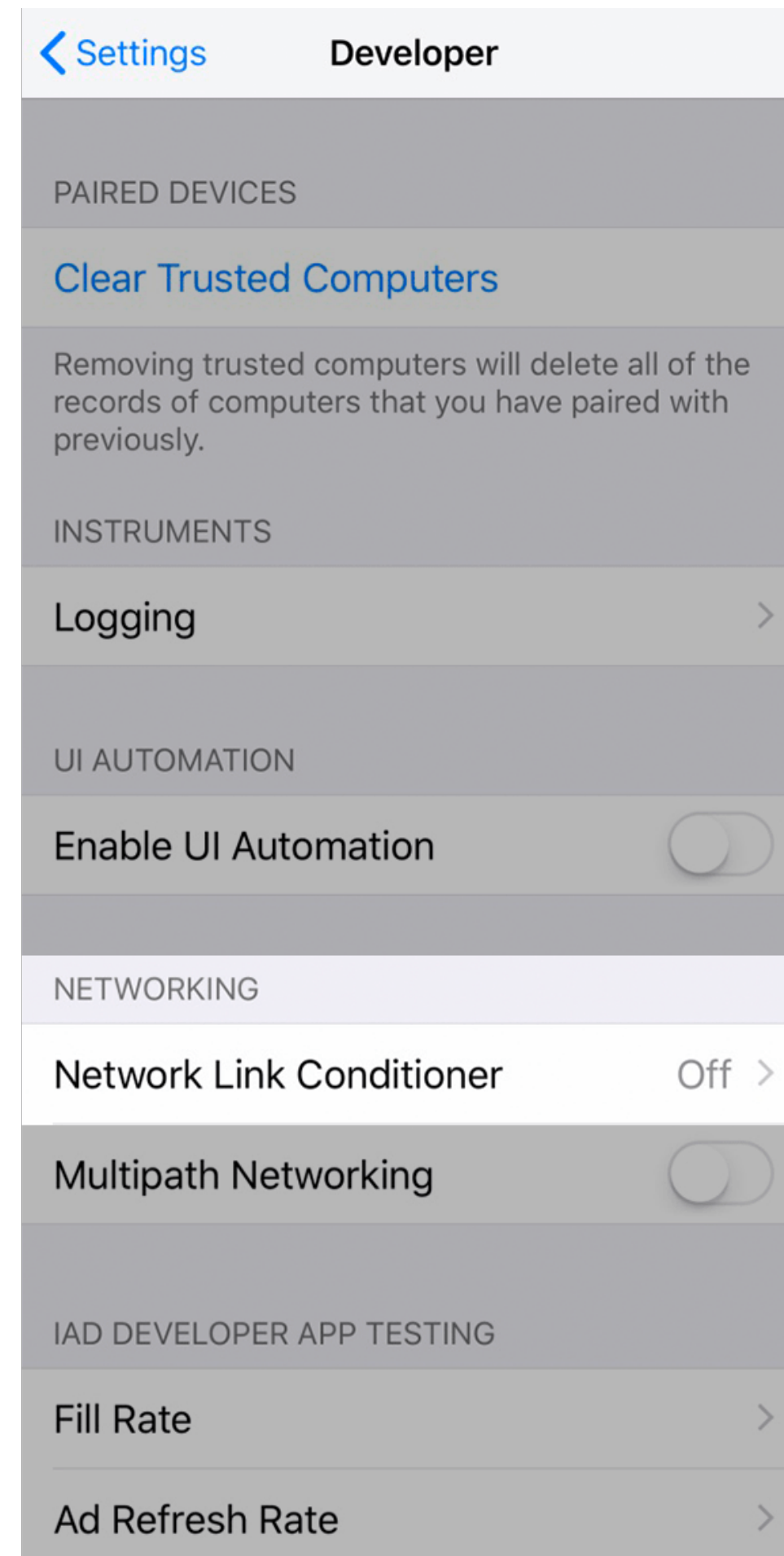
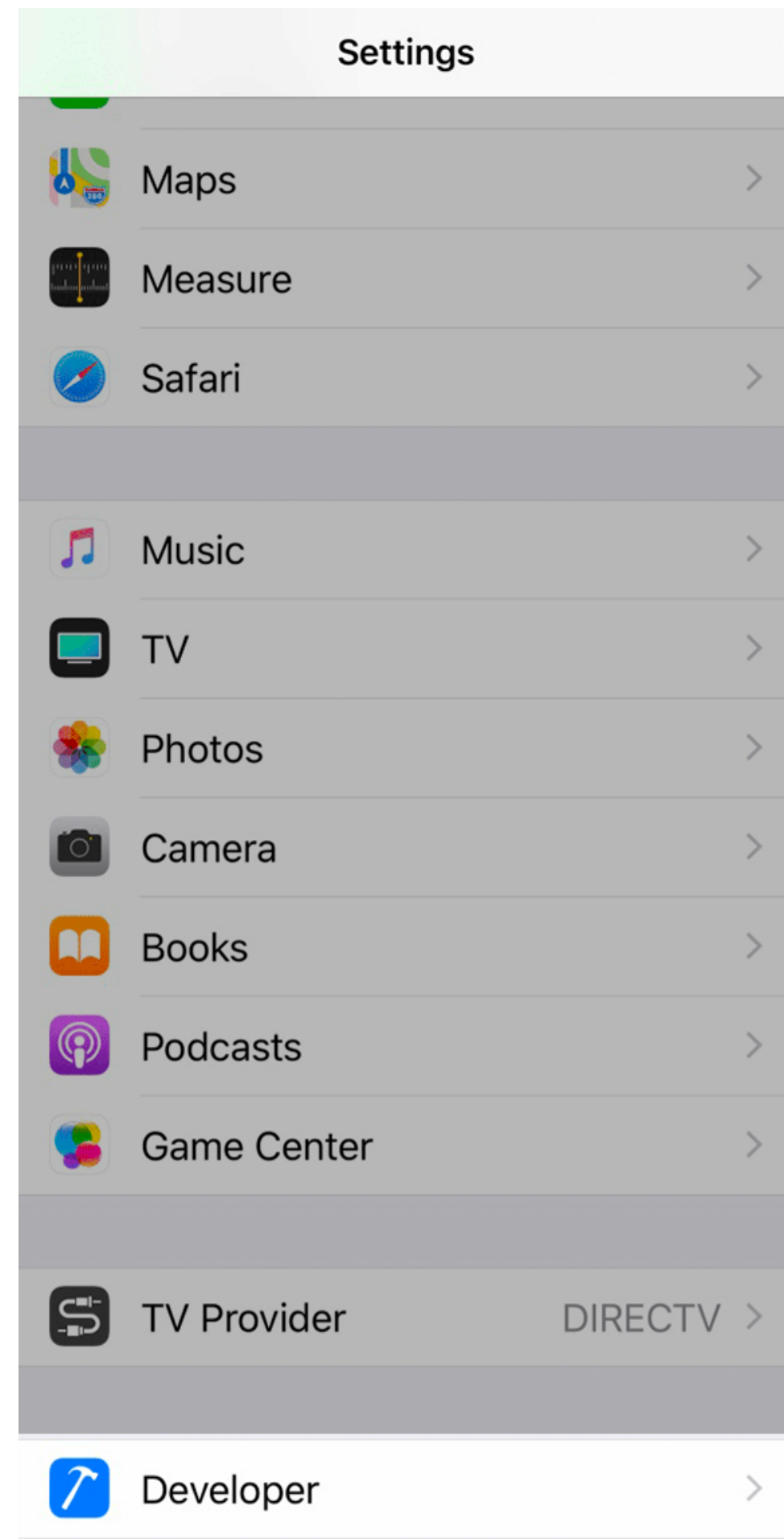
Work with a slow Network connection

- Show an activity Indicator:

```
@IBOutlet weak var activityIndicatorView: UIActivityIndicatorView!  
  
...  
  
// While Loading  
self.activityIndicatorView.startAnimating()  
  
...  
  
// Loading completed  
self.activityIndicatorView.stopAnimating()  
self.activityIndicatorView.isHidden = true
```



iOS: Network Link Conditioner



MacOS: Network Link Conditioner

The Network Link Conditioner is part of the “Additional Tools for Xcode” package.

You can download it from the [Apple Developers](#) page.

