# iOS Developer Technical Task

This is a simple technical challenge that is intended to assess your technical ability as an iOS developer and how you solve problems.

## The Task

The task is to build an iOS application that will make a network request and display a credit score to the user, and then a detail page that opens when the donut is tapped by the user, displaying some additional information from the endpoint with UI you prefer.

The dashboard page should be something like the screenshot below:

**A picture containing application

Description automatically generated**

A request can be made to the following endpoint. It will respond with a JSON payload providing credit score information for the user.

<https://5lfoiyb0b3.execute-api.us-west-2.amazonaws.com/prod/mockcredit/values>

We understand that your time may be limited so we are not expecting a perfect solution. Where you would have done something differently given more time, please explain what you would have done and why.

## How to submit your solution

Your solution should be an Xcode project that we can open and run with the latest version of Xcode without requiring any other steps.

You can provide the solution to us either as a zipped file, or ideally as a link to a public Git repository.

You can provide any supporting comments as a readme file. This can be used to explain your approach, assumptions and any decisions you have made.

Have fun with the task, it is your opportunity to show us your personal programming style. Good luck!

## Criteria and Tips

We do not expect a perfect solution, but we do expect your solution to be **production-grade**.

Here are some criteria and tips:

* The project should contain only what is needed.
* The code should be well structured and easy to follow.
* The solution should be well tested with UnitTests.
* You should consider error cases.
* You should consider other form factors.
* You can make any technical decisions that you think are appropriate provided you’re willing to explain them.
* You can use any native framework or tool you are comfortable with, like SwiftUI rather than UIKit, as well as Combine or Async/Await
* Your solution should be written entirely in **Swift** – we do not use Objective C.

## Bonus Points

For bonus points, consider some of the following extras:

* Add a circular progress indicator inside of the circle that animates as it appears, rather like the donut in the ClearScore app.
* Make the app ready for localization
* Implement useful UITests