



iOS Coding Challenge

When we start developing a new feature at Veryable, we implement UI prototypes and follow technical specifications. For this exercise, we have provided you with some wireframes and specs (see below) which you will need to develop a new feature. We have also included the required assets (images, colors, etc.) and a base project for this exercise.

The Base Project

We set up an example XCode project which uses Cocoapods for dependency management. You'll notice we already included the SnapKit pod which can be used for programmatic Auto Layout constraints. You can choose to use SnapKit if you want or go a different route. You can also add any other dependencies you feel are necessary (HTTP client, etc.). If you don't already have a favorite HTTP client (and you want to use one) then we recommend [Alamofire](#).

For the base project setup, we've started with an MVC pattern for the view layer and created files for you to implement a Repository pattern for the networking layer. You can choose to follow these patterns or implement something completely different — up to you! Just make sure your code is readable and consistent.

Instructions

Please read the following instructions along with the [Wireframe.pdf](#) file to understand the requirements of the application. The wireframes clearly indicate what the views should look like and how the user should interact with them. Make sure you follow the details for the fonts and colors!

Please start from the attached base project and add your code. All the necessary colors and fonts are included in the project.

1. The purpose of this exercise is to build an app where the user can see the list of their banking and card accounts (AccountList) and select an individual account to view some details (AccountDetail).
2. In order to get the list of accounts, you should make an HTTP request to retrieve [this JSON file](#) hosted on S3. The application should parse the response to the appropriate model and display in order.
3. The list of accounts should be vertically scrollable. When a user touches one of the list items, the app should navigate to the next page (AccountDetail) where the user can see the details.
4. To get back to AccountList from AccountDetail, the user can touch either the physical Done button or the back arrow in the navigation bar.

Best of luck from the Veryable team and we hope to hear back from you soon! Thanks for your time!