ASSIGNMENT 3

DEV TRADE

# OBJECTIVE

In this assignment, you will create the UI for the DevTrade app scene.

The goal of this assignment is to master the use of stack view and size classes concept.

# MATERIALS

For this assignment, you’ll need the presentation slides from session 6 (you can find it on GitHub, session 6).

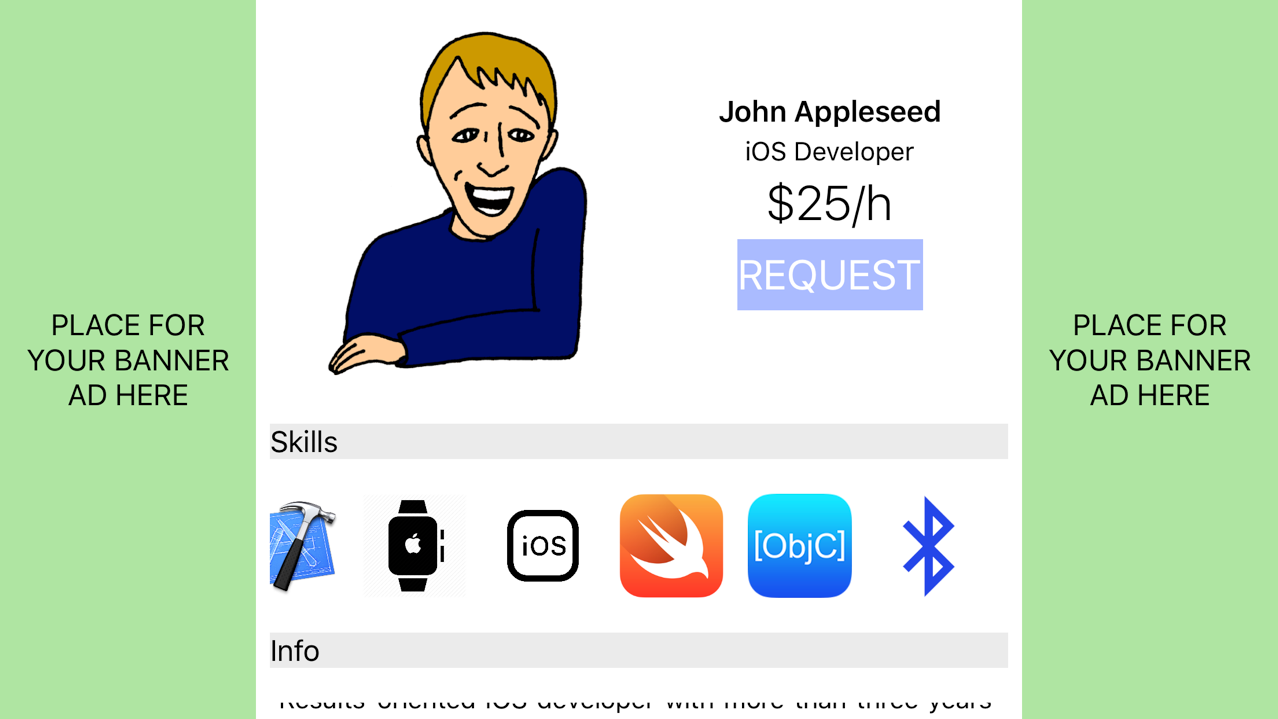
Also, the start-up project with all necessary assets is provided. Short demo video is also provided as reference. You can found both on the GitHub.

DevTrade app in short:

* 1. Developer CV is presented on screen
  2. User can easily scroll up and down to see all relevant info about developer
  3. User can be redirected to developer’s apps link
  4. Ad banners are presented to user in some screen sizes

# Required tasks

1. Create a **complete** scene in **storyboard**.
2. App scene should be properly presented in both **portrait and landscape** mode for all device screen sizes (iphone4, SE, 7, 7+, iPad). In case of **regular width, banner ad** placeholder views should be presented to user (the green views in demo video). In other width size (**compact**) banners should be removed from the screen.
3. Use of **UIStackView** and **UIScrollView** are **highly recommended** for this assignment.
4. Use of **UITableView** is forbidden.
5. Try to replicate scene from the demo video. Some basic scene guidelines:
   1. Skills icon size is 60x60 pt
   2. Banner width is 20% of screen width (in case if banner is presented)
   3. In **compact width** size, developer image is presented above the developer name and price. **In regular width** size, developer image is placed left to developer name/price (see demo video)
   4. For all the other distances and sizes, be free to set your own but try to replicate the visual appearance from the demo as much as possible.



0.2\*A

AA

A

0.2\*A

AA

# HINTS

1. Try to first layout those green banners and central view. Create all necessary constraints for various size classes. Recall that **Vary for Traits** can’t help us with view installation/uninstallation – this must be done manually, remember those **+** signs left to view properties in attribute inspector menu?
2. Take a **UIScrollView** and used it as the view for central part of the screen. You will certainly need several scroll views (**UIScrollView**) in this assignment so get familiar with scroll views. Take a look at <http://stackoverflow.com/questions/31668970/is-it-possible-for-uistackview-to-scroll>
3. Try to visualise in how many horizontal/vertical containers views can be grouped. Each group of views can be embedded in stack view. Stack views can be nested, too.
4. Stack view creates its own set of constraints in order to satisfy given alignment, distribution and spacing. Try not to break those constraints. Generally speaking, you are just putting your views in stack view and try to constraint the stack view against other views (the views not contained by the stack view itself).
5. Look for **UITextView**. It is very powerful view for presenting text, it is like a **UILabel** on steroids. Text view can be easily tweak to automatically detect emails, links, phone numbers and similar in containing text. Also, text view can manage text scroll.