From UI Design to UIKit

How do I build U1?

No Interface Builder

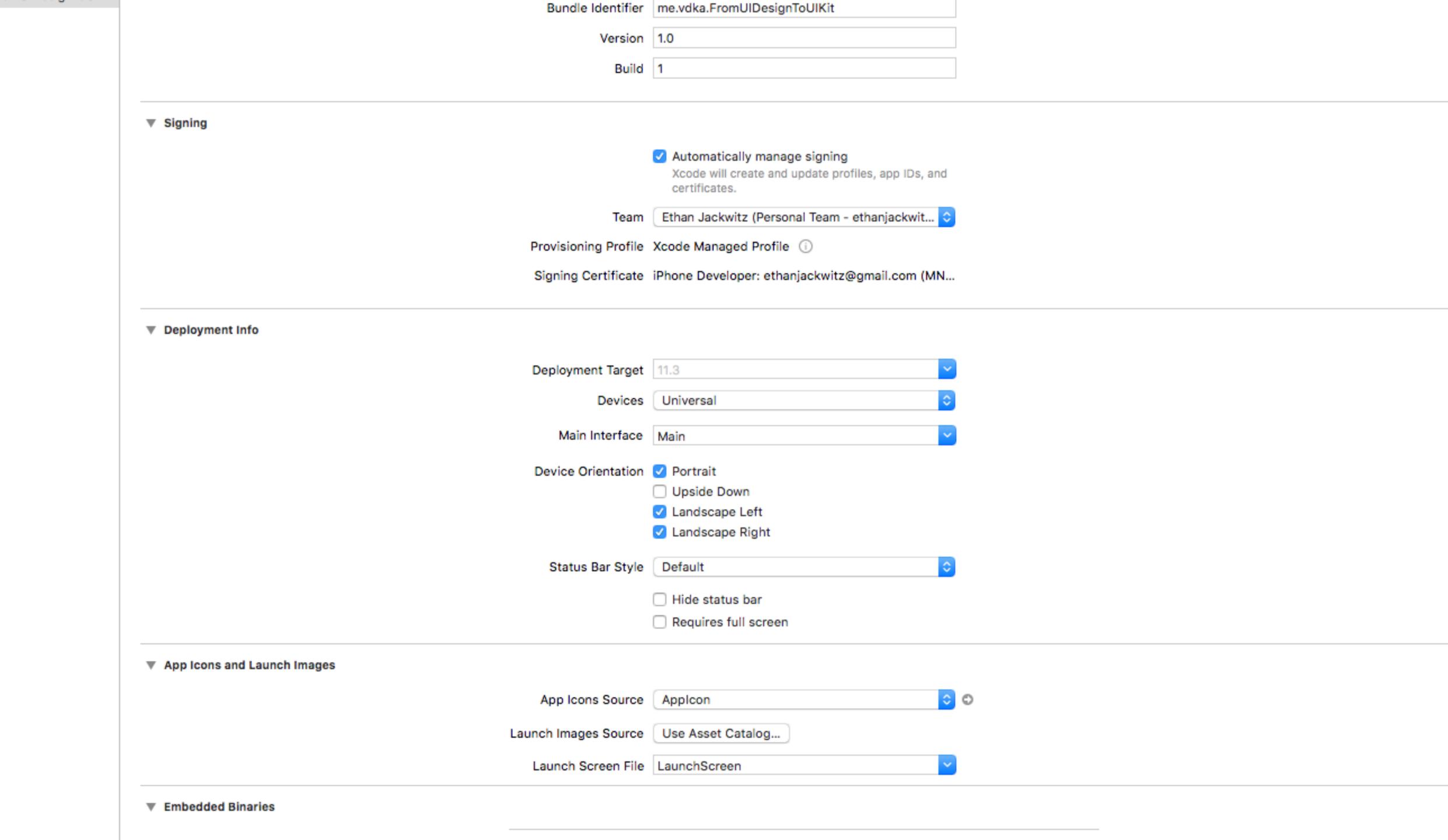
No Interface Builder Helpers for AutoLayout

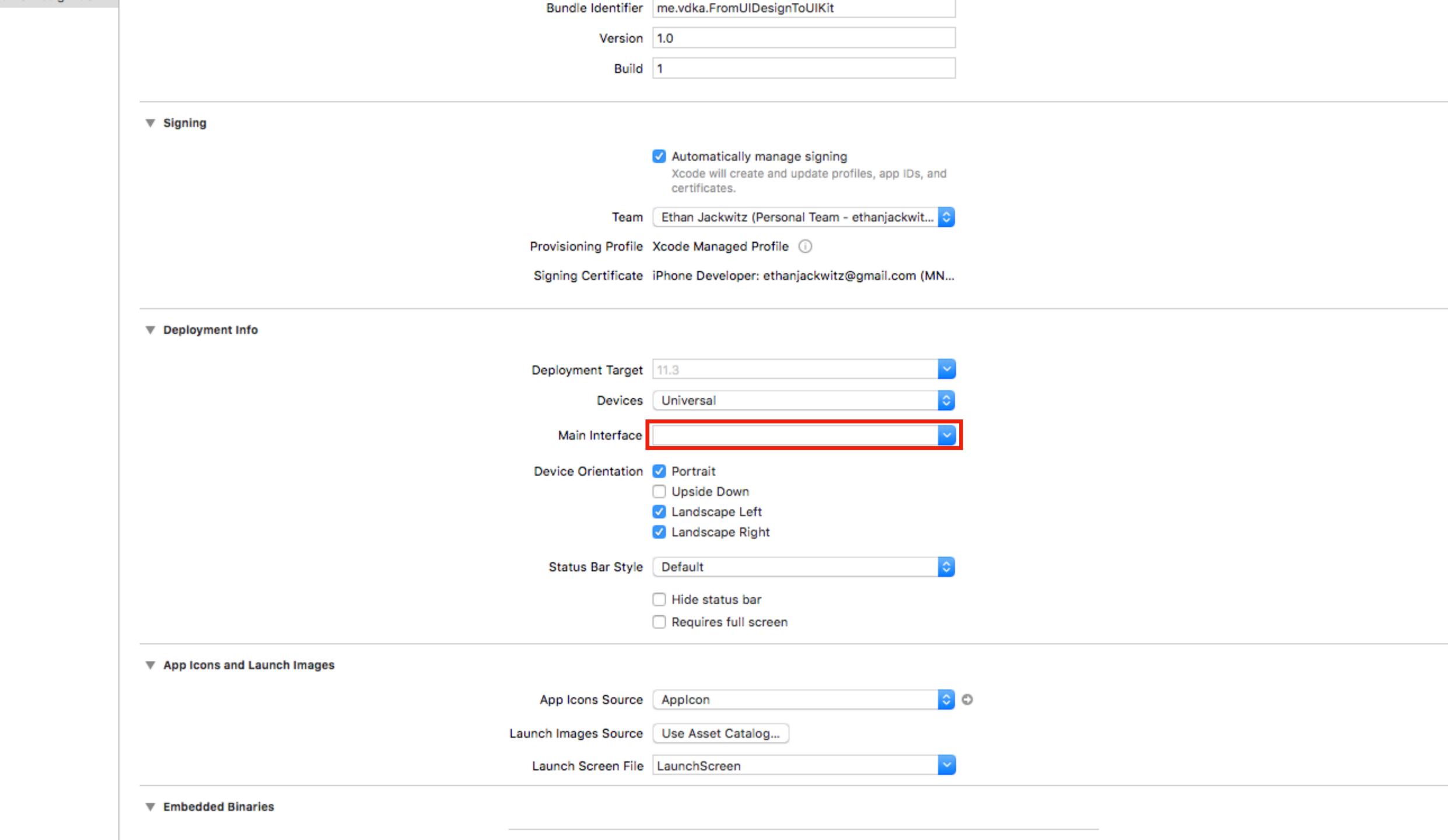
Helpers for AutoLayout

Debugging using Reveal.app

Debugging using Reveal.app

Using UlLayoutGuides





```
extension UIView {
    func addSubviewEnablingAutoLayout(_ subview: UIView) {
        self.addSubview(subview)
        subview.translatesAutoresizingMaskIntoConstraints = false
    }
}
```

```
extension NSLayoutConstraint {
    func withPriority(_ priority: UILayoutPriority) → NSLayoutConstraint {
        self.priority = priority
        return self
    }
}

extension UILayoutPriority: ExpressibleByIntegerLiteral {
    public init(integerLiteral value: Int) {
        self.init(rawValue: Float(value))
    }
}
```

```
protocol ReusableView {
    static var reuseIdentifier: String { get }
}

extension ReusableView where Self: UIView {
    static var reuseIdentifier: String {
        return String(describing: self)
    }
}
```

```
extension UITableView {
    // register for the Class-based cell
    func register<T: UITableViewCell>(_: T.Type) {
        register(T.self, forCellReuseIdentifier: T.reuseIdentifier)
    func dequeueReusableCell<T: UITableViewCell>(forIndexPath indexPath: IndexPath) → T {
        guard let cell = dequeueReusableCell(withIdentifier: T.reuseIdentifier, for: indexPath) as? T else {
            fatalError(" ... ")
       return cell
    // register for the Class-based header/footer view
    func register<T: UITableViewHeaderFooterView>(_: T.Type) {
        register(T.self, forHeaderFooterViewReuseIdentifier: T.reuseIdentifier)
    func dequeueReusableView<T: UITableViewHeaderFooterView>() → T? {
        let v = dequeueReusableHeaderFooterView(withIdentifier: T.reuseIdentifier) as? T
        return v
```

```
extension UITableViewCell: ReusableView {}
extension UITableViewHeaderFooterView: ReusableView {}
```

Reveal

revealapp.com

Peek

github.com/shaps80/Peek