MVVM in Practice

Vytis Šibonis,
Wahanda

Recap: MVC

- Model - data, no logic
- View - UI, no logic
- Controller - the rest, aka everything

Model - View - View Model

- Compatible with usual MVC
- Reduces code in controller

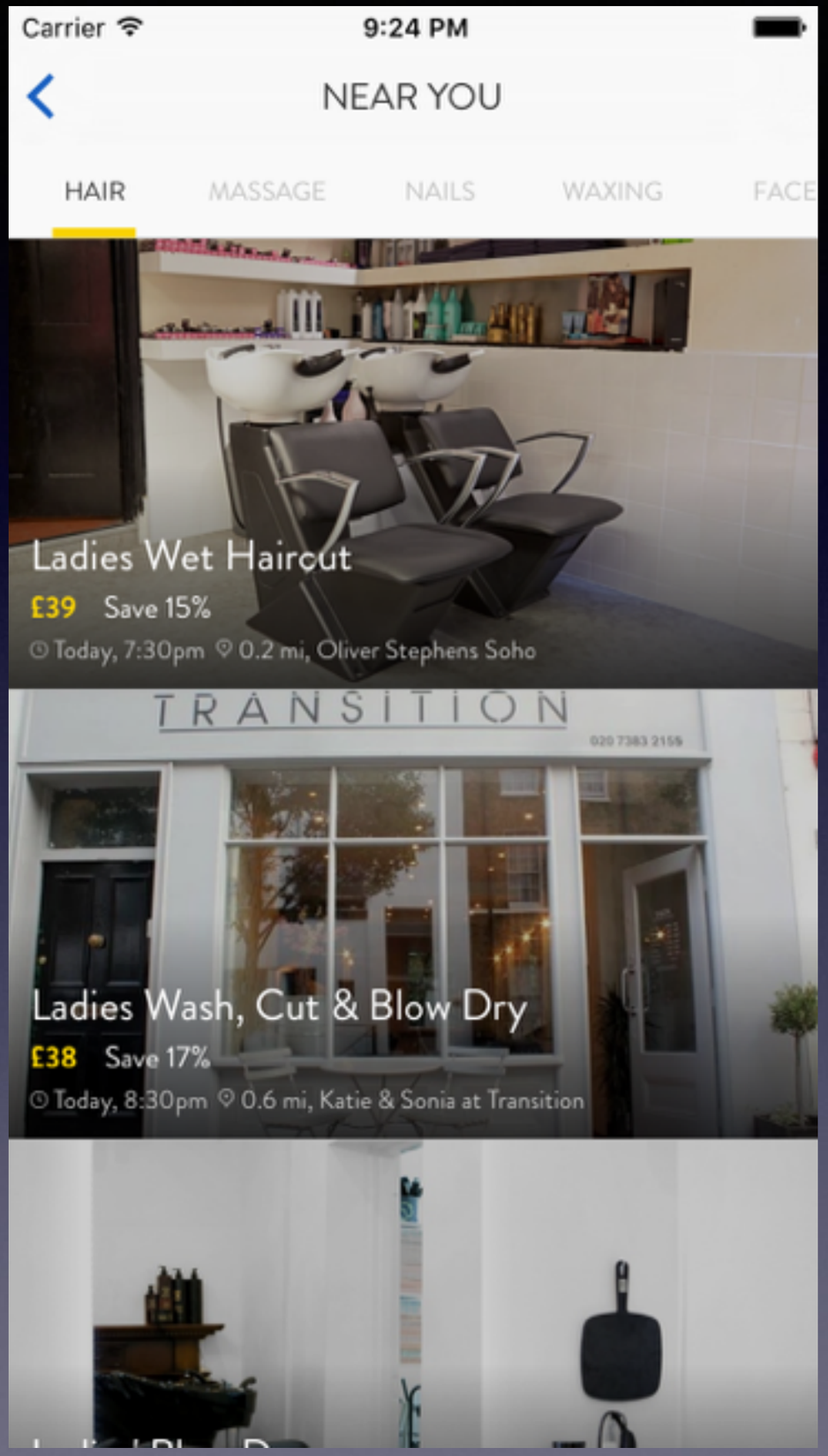
View Model

- Glue between view and model
- Created from model
- Provides content for each element in the view

Wahanda

Buy any beauty service:

- Haircut
- SPA weekend
- Snail facial (??)

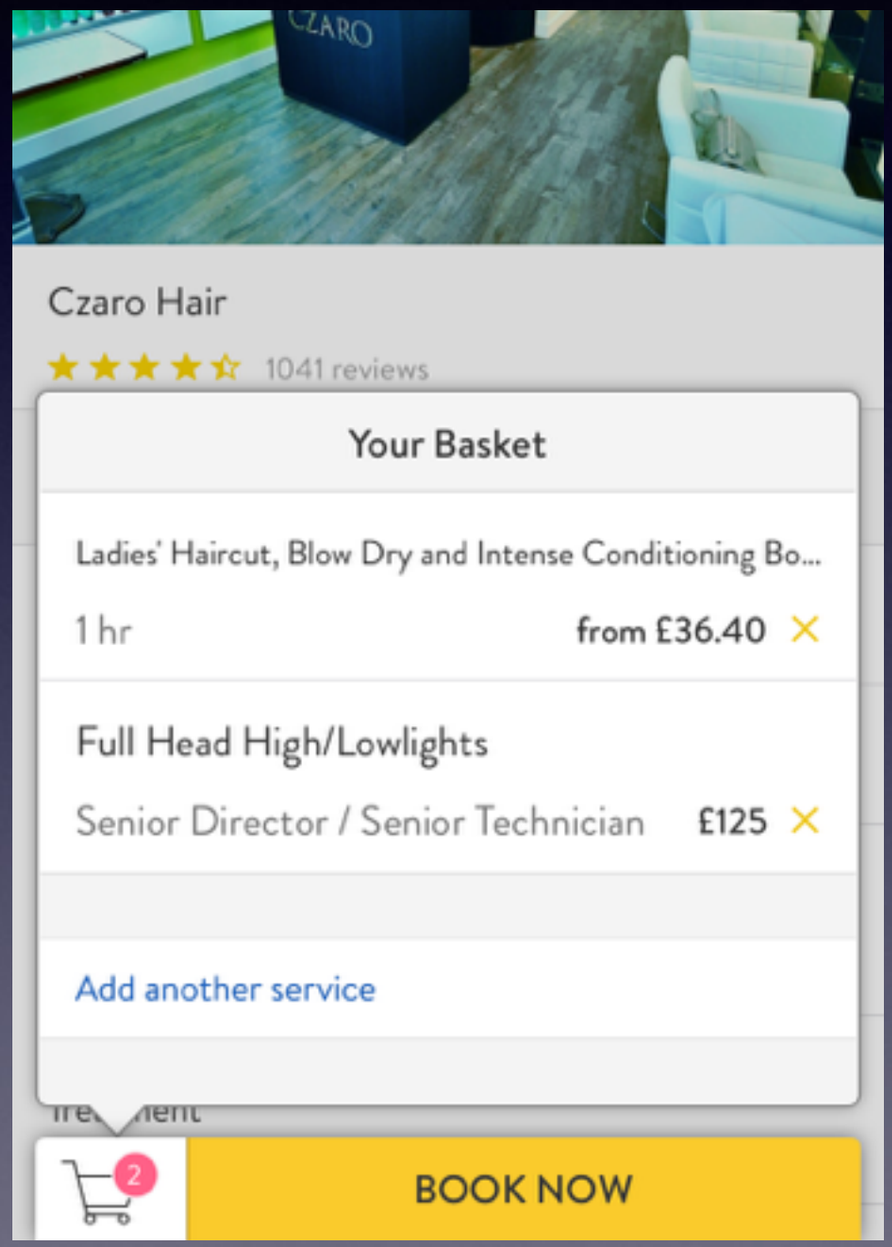


Challenges

- Lots of presentation logic in the app
- Different combinations
- Manual testing doesn’t scale…and boring

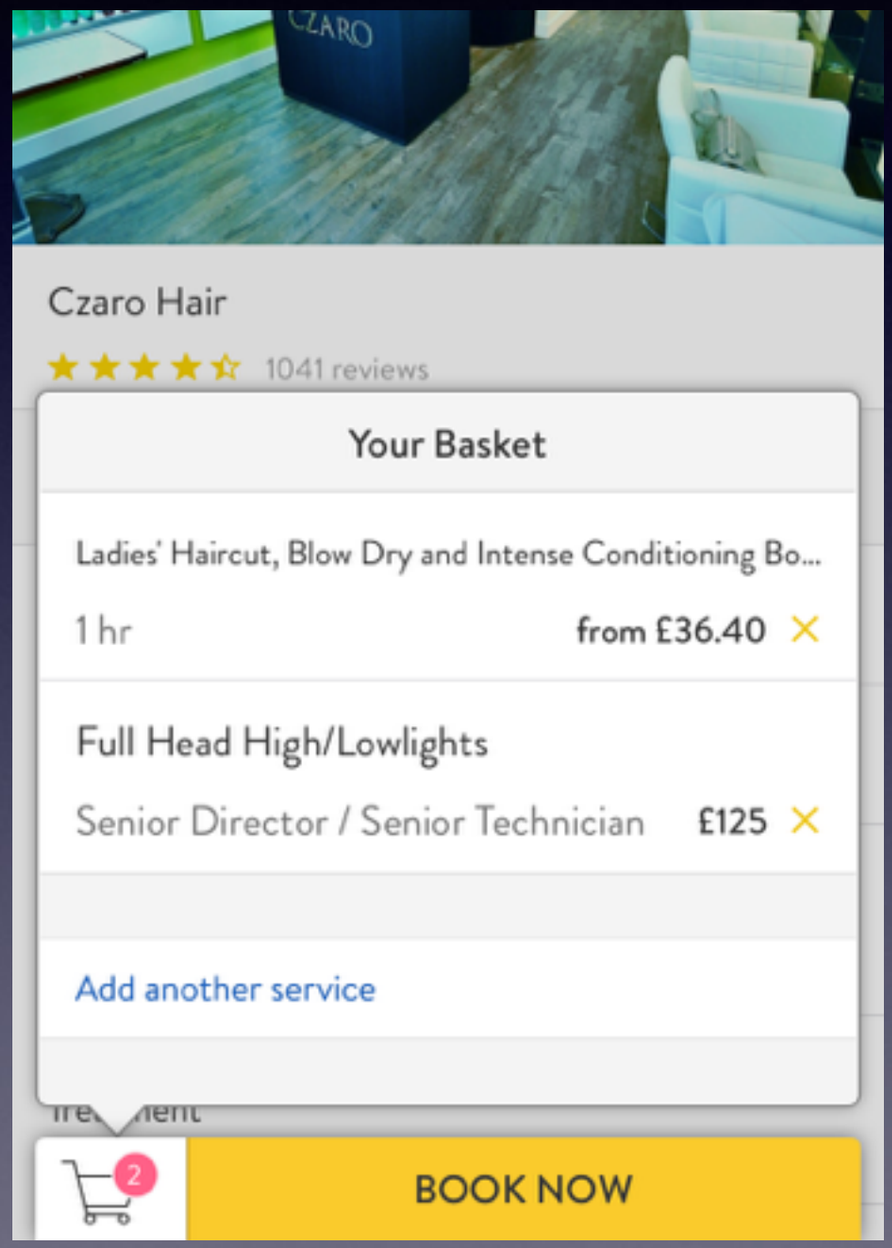
Example: Basket

- Duration
- Employee category
- Various prices
- Number of nights etc.



Example: Basket

- Data comes from API
- Not optimized for UI
- Different models for each entry



View model to the rescue!

```
@interface BasketEntryViewModel : NSObject

+ (instancetype)viewModelWithServiceEntry:(BasketServiceEntry *)serviceEntry;

@property (nonatomic) NSString *serviceTitle;
@property (nonatomic) NSString *serviceDuration;
@property (nonatomic) NSString *priceString;

@end
```

UITableViewCell

```objc
@implementation BasketServiceCell

- (void)configureWithViewModel:(BasketEntryViewModel *)viewModel
{
    self.serviceTitleLabel.text = viewModel.serviceTitle;
    self.durationLabel.text = viewModel.serviceDuration;
    self.priceLabel.text = viewModel.priceString;
}

@end
```

Testing

```
- (void)testSimpleServiceSkuRow
{
    BasketEntryViewModel *model = [self modelFixtureWithId:@"2" ];
    XCTAssertEqualObjects(model.serviceTitle, @"Short hair", @"Wrong title");
    XCTAssertEqualObjects(model.priceString, @"£65", @"Should show full price");
}

- (void)testMultiSkuMultiAddedService
{
    BasketEntryViewModel *model = [self modelFixtureWithId:@"3"];

    XCTAssertEqualObjects(model.serviceTitle, @"Permy wavy", @"Wrong title");
    XCTAssertEqualObjects(model.serviceDuration, @"3 hrs", @"Duration should be
sum of all sku durations");
}
```

Results

- Presentation code lives only in View Model
- Easy to test
- When a bug is found it can be added to tests

Reading list

- https://www.objc.io/issues/13-architecture/mvvm/
- http://artsy.github.io/blog/2015/09/24/mvvm-in-swift/

Questions?

vytis@wahanda.com
@vytis0