

Building large products from reusable components on iOS

Introduction - concept

- ▶ Large product with 100+ screens
- ▶ Reusing UI Components
- ▶ Layout management
- ▶ Solutions
- ▶ UI Testing

New wave Mobile banking solution

- ▶ Open account
- ▶ PFM
- ▶ Pay bills
- ▶ Send money
- ▶ Split bill

The screenshot shows a mobile banking app interface for splitting a bill. At the top, there's a teal header with a back arrow and the title 'Split Bill'. Below this, the 'Bill amount' is displayed as '75.000 IDR' in a large, teal font. Underneath, it says 'Split between you and 2 more' with a 'RESET' link. The main section lists three recipients, each with a circular profile picture, the name 'Budi Hamdani', and a split amount of '25.000' with a delete 'x' icon. Below the list is a green circular button with a plus sign and the text 'ADD RECIPIENT'. Further down is a 'Comment' section with '(optional)' text and a placeholder text: 'Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce et nibh nibh.' At the bottom, there is a large green button labeled 'REQUEST FOR MONEY'.

How do screens add up to 100?

- ▶ New account creation – 16 screens
- ▶ Authentication – 6 screens
- ▶ Send money – 20 screens
- ▶ Profile – 12 screens
- ▶ Savings – 12 screens
- ▶ PFM – 10 screens
- ▶ Etc ... - 36 screens

Building blocks - components

- ▶ Form fields
 - ▶ Input field
 - ▶ Selection box
 - ▶ Checkbox
 - ▶ Button
- ▶ Tabbar
- ▶ Screen parts

Correspondence Address

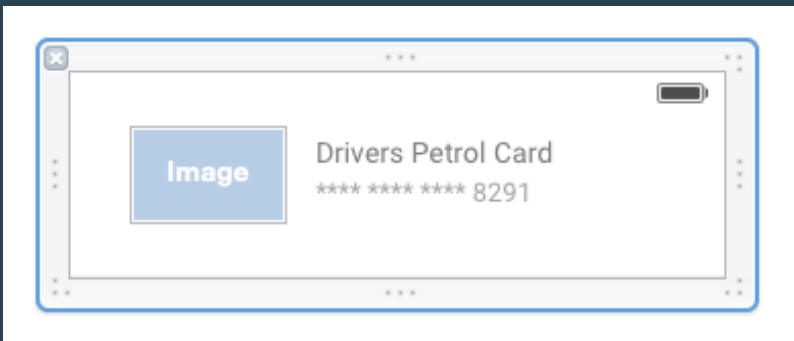
Street Address
Jalan Kebulan #900

Zip
283799

District
Kuningan

City
Jakarta

Building blocks - components



- ▶ Views (IBDesignable)
- ▶ Components (Xib)
- ▶ Screen

What is the challenge?

- ▶ 100+ screens
- ▶ Reusable UI components
- ▶ Different screen sizes
- ▶ We need to build these screens fast

We need a layout manager

What should a layout manager do?

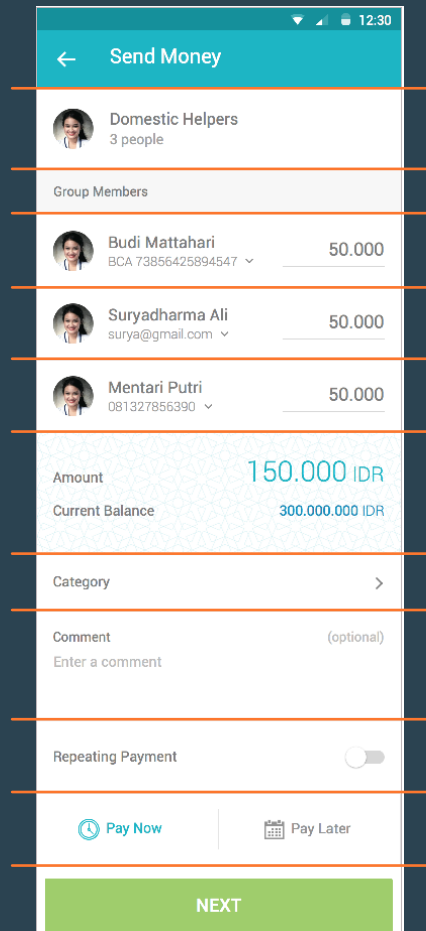
- ▶ Manage Auto Layout constraints
- ▶ Add / insert / remove component
- ▶ Arrange components vertically
 - ▶ Automatically scroll
- ▶ Arrange components horizontally
 - ▶ Automatically page
- ▶ Stick a component to the bottom of the screen
- ▶ Distribute content to full screen

Solutions

- ▶ Calculating coordinates
- ▶ Auto Layout
- ▶ `CSLinearLayout` (no Auto Layout)

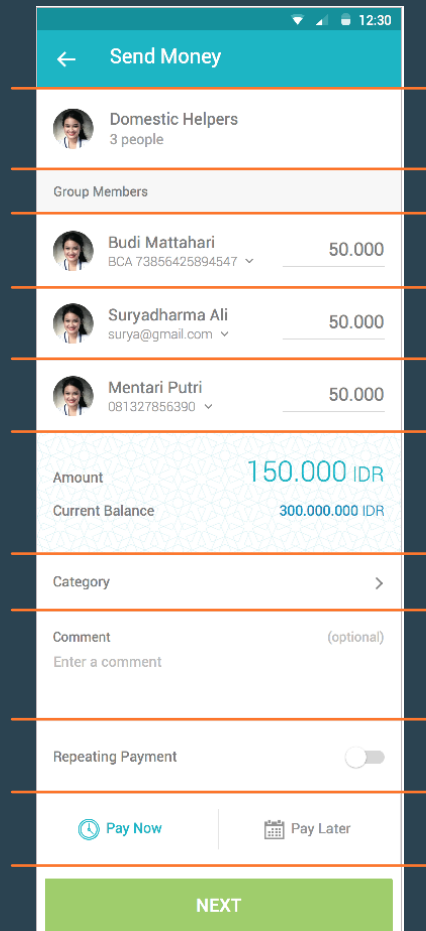
Our solution:
SCComponentCollection

SCComponentCollection



- ▶ Components are the building blocks
- ▶ Components are placed inside `ComponentCollections`
- ▶ Auto Layout constraints are managed automatically
- ▶ iOS7+

Features



- ▶ Add / Insert / Remove
- ▶ Vertical / Horizontal
- ▶ Automatic scrolling
- ▶ ComponentCollections can be embedded
- ▶ Move component to the bottom
- ▶ Resize component to fit the screen

SCComponentCollection+FormBuilder

The screenshot shows a mobile app interface with a status bar at the top displaying 'Carrier', signal strength, '8:39 PM', and battery level. Below the status bar is a navigation bar with a back arrow, a progress indicator (10 dots, the first is filled), and '1 of 10'. The main content area has a title 'Contact information' in blue. Below the title is a message: 'First, we need to verify your basic information. Please tell us your mobile number and e-mail to proceed.' There are two input fields: 'Mobile number' with the value '+62123123123' and 'Email' with the value 'asdf@asdf.com'. Below the email field is a note: 'This will be used to log in to Generation 1.0.' At the bottom is a large green button labeled 'NEXT'.

- ▶ Components are form fields
- ▶ Validation
- ▶ Move to next field
- ▶ Next / Done button

Example

Carrier 8:46 AM

← ● ○ ○ ○ ○ ○ ○ ○ ○ ○ ○ 1 of 10

Email cannot be empty.

Contact information

First, we need to verify your basic information.
Please tell us your mobile number and e-mail to proceed.

Mobile number
+62123123

Email

This will be used to log in to Generation 1.0.

NEXT

- ▶ 4 Components:
 - ▶ Header
 - ▶ Mobile text
 - ▶ Email text
 - ▶ Next button

Code

```
self.header = [self.onboardingModule
                onboardingContactInfoHeaderComponentWithTitle:FWLocalizedString(CONTACTINFO_TITLE)
                andDescription:FWLocalizedString(CONTACTINFO_DESCRIPTION)];

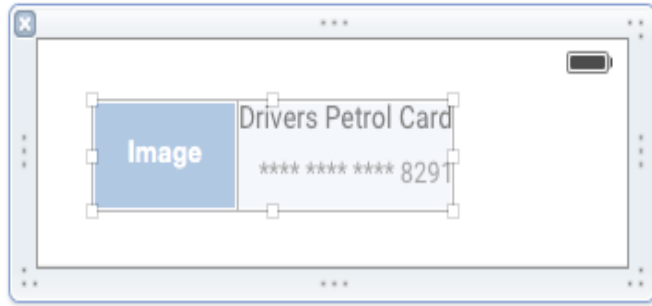
self.emailInputField = [self.activationModule inputFieldFormItemWithTitle:CONTACTINFO_EMAIL_INPUT
                        validatorDelegate:self
                        validators:@[
                            [EmptyValidator validatorWithMessage:CONTACTINFO_EMAIL_EMPTY_ERROR],
                            [RegexValidator validatorWithMessage:CONTACTINFO_EMAIL_FORMAT_ERROR regex:EMAIL_REGEX]]];

self.formNextButton = [self.activationModule formNextButtonWithTitle:CONTACTINFO_NEXT_BUTTON
                        clickedDelegate:self];

[self.componentCollection registerComponent:_header];
[self.componentCollection registerFormItem:_mobileInputField];
[self.componentCollection registerFormItem:_emailInputField];
[self.componentCollection registerFormButtonToTheBottom:_formNextButton];
```


iOS9: UIStackView

UIStackView



- ▶ Horizontal / Vertical
- ▶ Alignment
- ▶ Distribution
- ▶ IB Support

Testing

The key to automatic testing

- ▶ Stability: No false negative test runs!
- ▶ Easy to write / maintain
- ▶ Environment independent
- ▶ Take small steps

Stability: Build Environments

- ▶ MOCK – Tests can run 100%
- ▶ Integrated
 - ▶ Staging
 - ▶ Production

Testing

- ▶ KIF – easy to write
- ▶ MOCK – navigation, UI can be tested
- ▶ Integrated envs:
 - ▶ NSURLConnectionVCR
 - ▶ Record network communication
 - ▶ Run integrated tests on CI without real network communication

Summary

- ▶ We need a layout manager to build screens with speed
- ▶ `SCComponentCollection` will be open sourced
- ▶ iOS9 – `UIStackView` is a big thing
- ▶ Testing: stability, maintainability, small steps

References

- ▶ <https://github.com/scalessec/CSTableView>
- ▶ <https://github.com/dstnbrkr/VCRURLConnection>
- ▶ <http://ocmock.org/reference/>
- ▶ <http://typhoonframework.org/>
- ▶ <https://github.com/team-supercharge/SCConfiguration>
- ▶ <https://github.com/specta/expecta/>
- ▶ <https://github.com/facebook/ios-snapshot-test-case>
- ▶ <https://github.com/luisobo/Nocilla>
- ▶ <https://wiki.jenkins-ci.org/display/JENKINS/Home>
- ▶ <http://oclint.org/>
- ▶ <https://github.com/Cue/ocstyle>
- ▶ <http://semver.org/>
- ▶ <https://developer.apple.com/videos/play/wwdc2015-218/>

Thanks for your attention!

Questions?



David Kovacs

Supercharge

CTO

@davidkovaccs

david.kovacs@supercharge.io // www.supercharge.io



Supercharge