



/dev/world
10th anniversary

Behaviour Driven Development for iOS

Mira Kim

ASB / Software Specialist

@mrk144



In Partnership With



WORKSHOP AGENDA

- ▶ Introduction
- ▶ Part 1: Understanding BDD
- ▶ (Short break)
- ▶ Part 2: BDD with Calabash
- ▶ Conclusion

INTRODUCTION

PART 1: UNDERSTANDING BDD

WORKSHOP PART 1

- ▶ Part 1: Understanding BDD
 - ▶ What is Behaviour Driven Development?
 - ▶ How does BDD fit into app development workflow?
 - ▶ What are the benefits?
 - ▶ What are the challenges?

BEHAVIOUR DRIVEN DEVELOPMENT

Behaviour Driven Development (BDD) is a synthesis and refinement of practices stemming from [Test Driven Development](#) (TDD) and [Acceptance Test Driven Development](#) (ATDD).

CAKE

An item of soft sweet food made from a mixture of flour, fat, eggs, sugar, and other ingredients, baked and sometimes iced or decorated.



TASTE OF BDD

SOCIAL NETWORKING FOR FOODIES



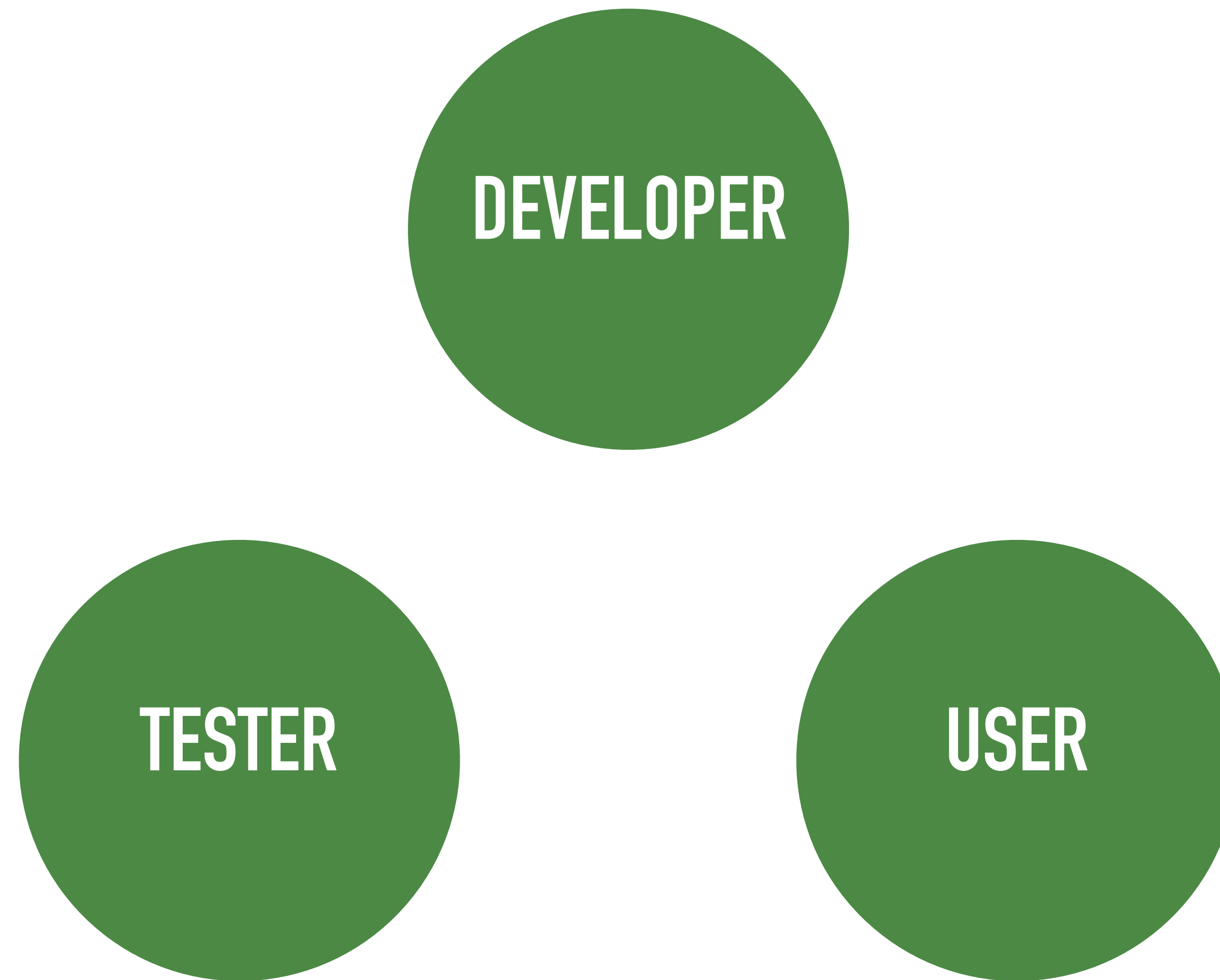
10 MIN

- ▶ How would you make this app?
- ▶ What features do we want?
- ▶ What questions would you ask?
- ▶ As a group compile a list of questions

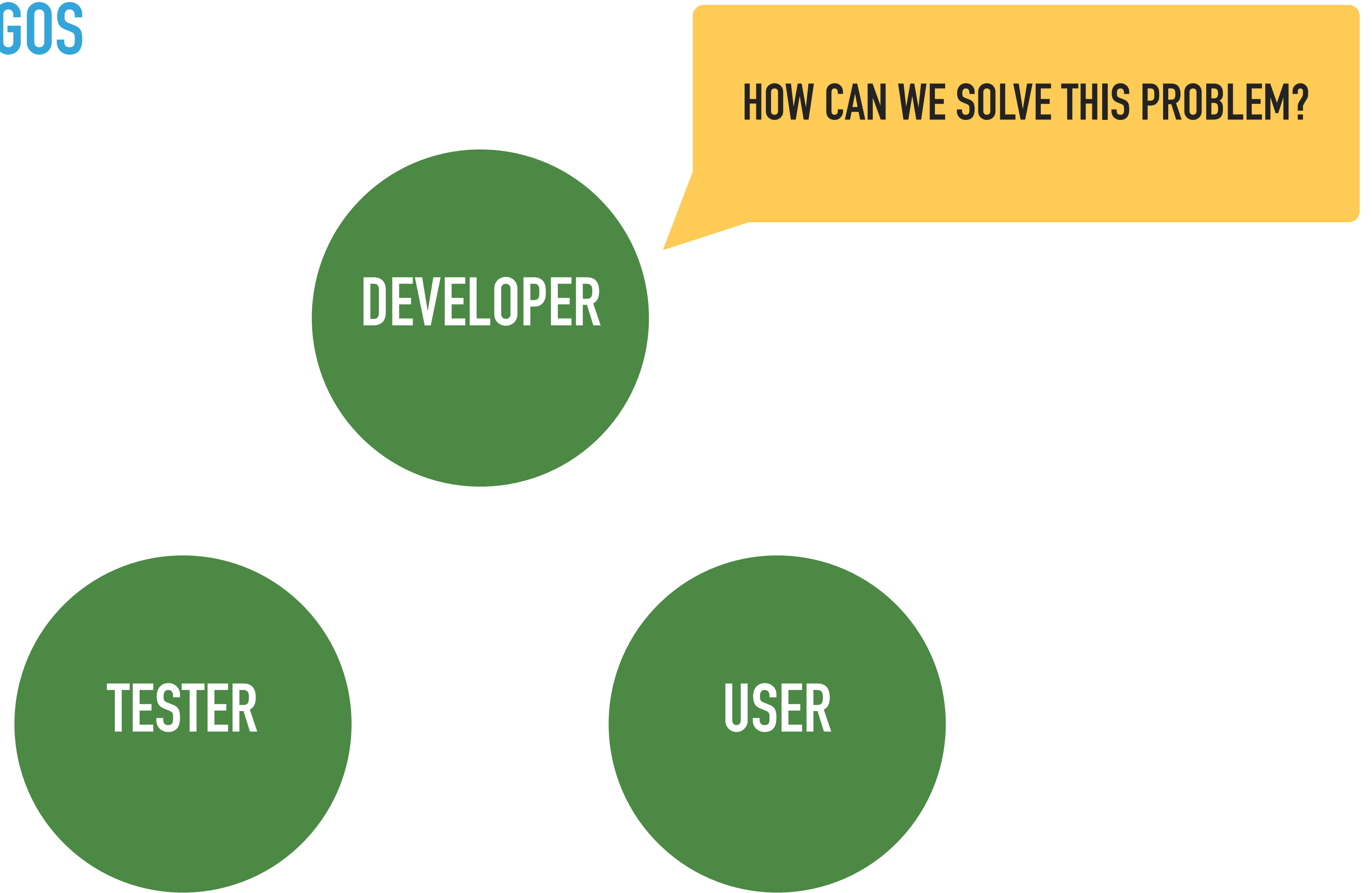
PERSPECTIVE

- ▶ Developers' perspective
 - ▶ Technical requirements
 - ▶ Frameworks
 - ▶ Platform specific information
- ▶ Centred around "how do we write code"

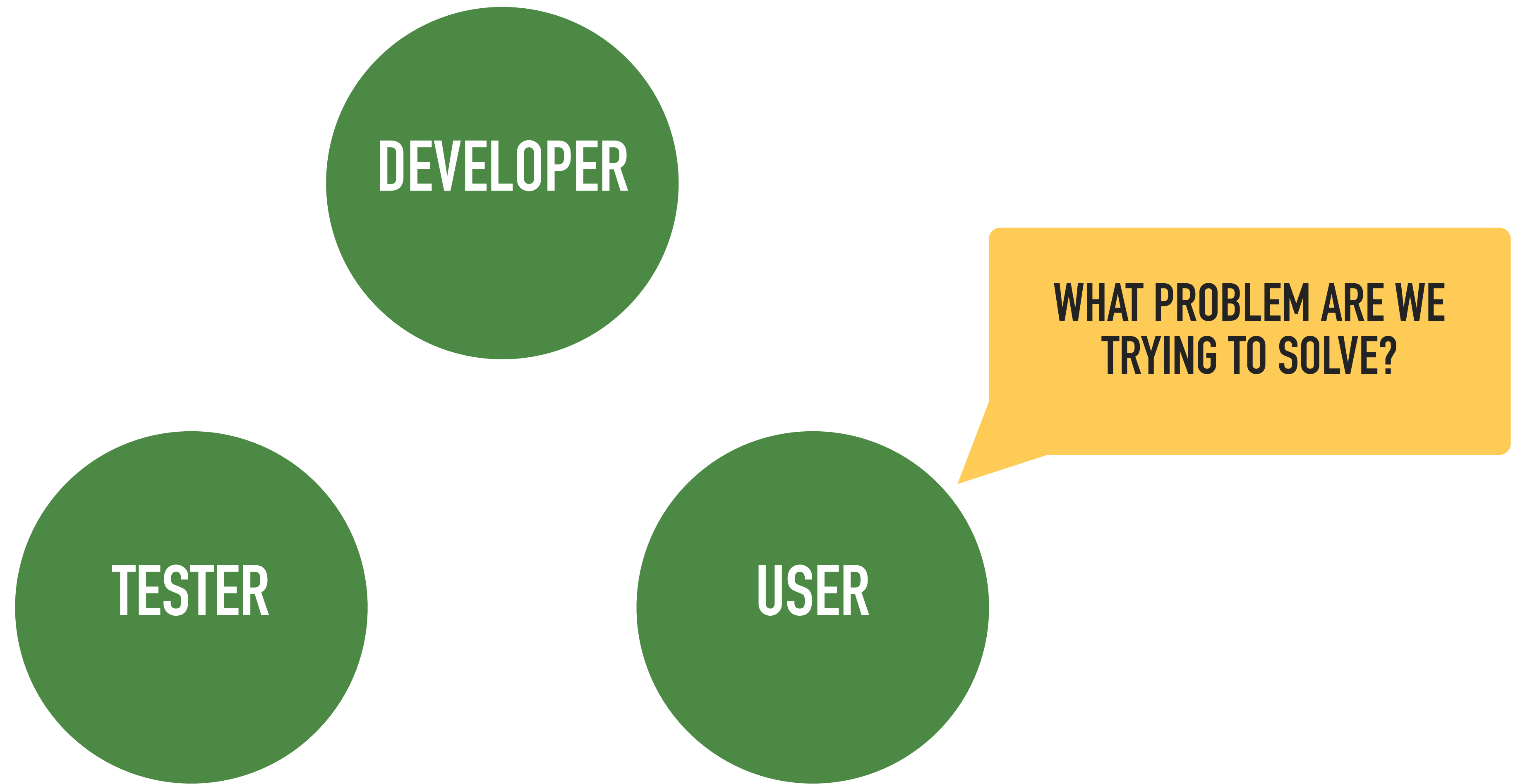
THREE AMIGOS



THREE AMIGOS



THREE AMIGOS



THREE AMIGOS

DEVELOPER

WHAT ABOUT ...

TESTER

USER

USER STORY

- ▶ Short, simple descriptions of a feature
 - ▶ As a <type of user>
 - ▶ I want <some goal>
 - ▶ So that <some reason>

USER STORY

- ▶ As a person who likes food
- ▶ I want to upload pictures of my brunch
- ▶ So that my friends can like it

ACCEPTANCE TESTS

- ▶ Description of the behaviour of a software
- ▶ Expressed as an example or a usage scenario

ACCEPTANCE TESTS

- ▶ Given ...
- ▶ When ...
- ▶ Then ...

ACCEPTANCE TESTS

- ▶ Given that user is logged in
- ▶ When the photo button is pressed
- ▶ Then Photo picker should be shown

ACCEPTANCE TESTS

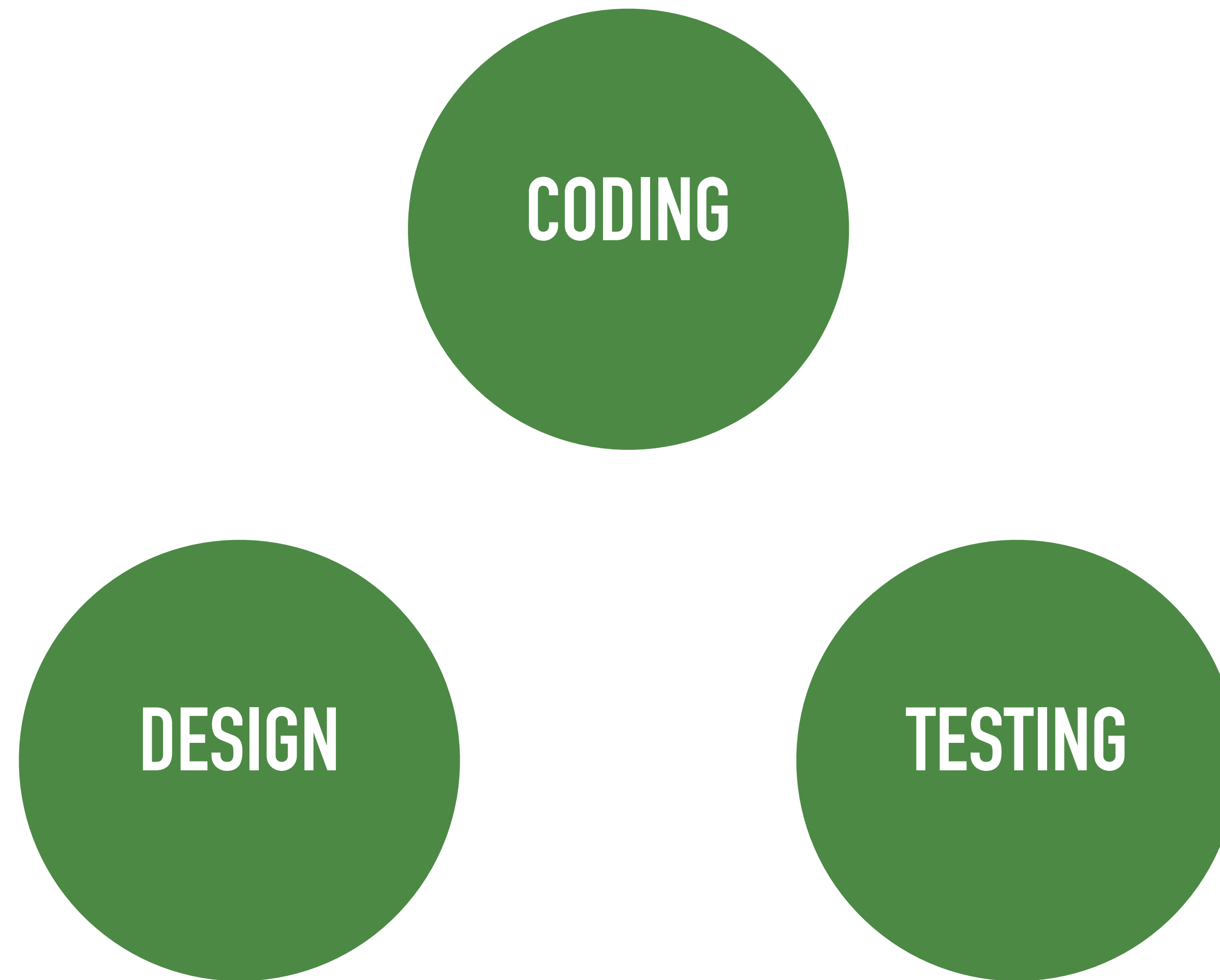
- ▶ Represent the user's point of view
- ▶ Form of requirements
- ▶ Way of verifying that app functions as intended
- ▶ May be automated

ACTIVITY

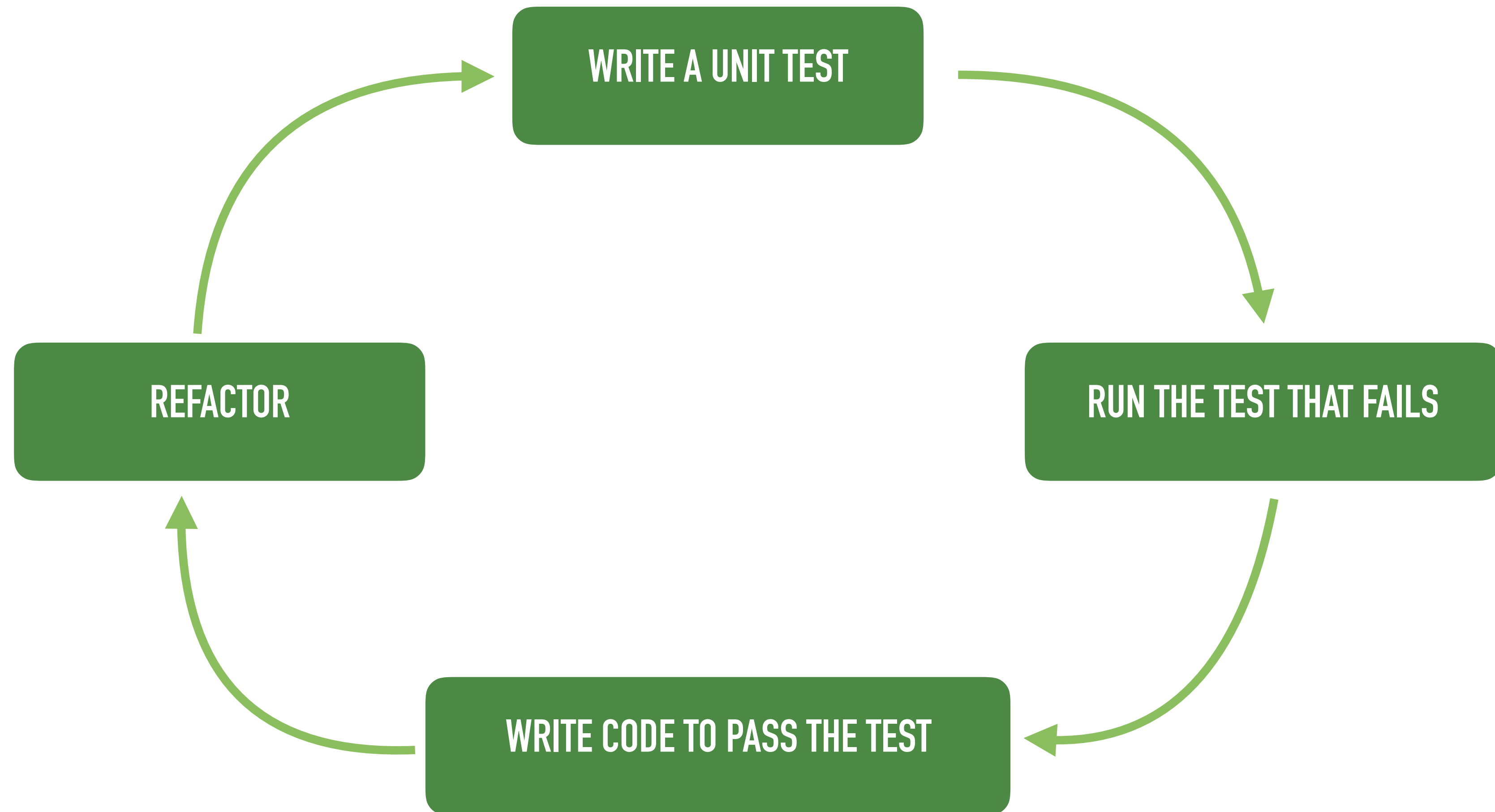
20 MIN

- ▶ Social Networking for Foodies
- ▶ Write up user stories and acceptance tests for each feature

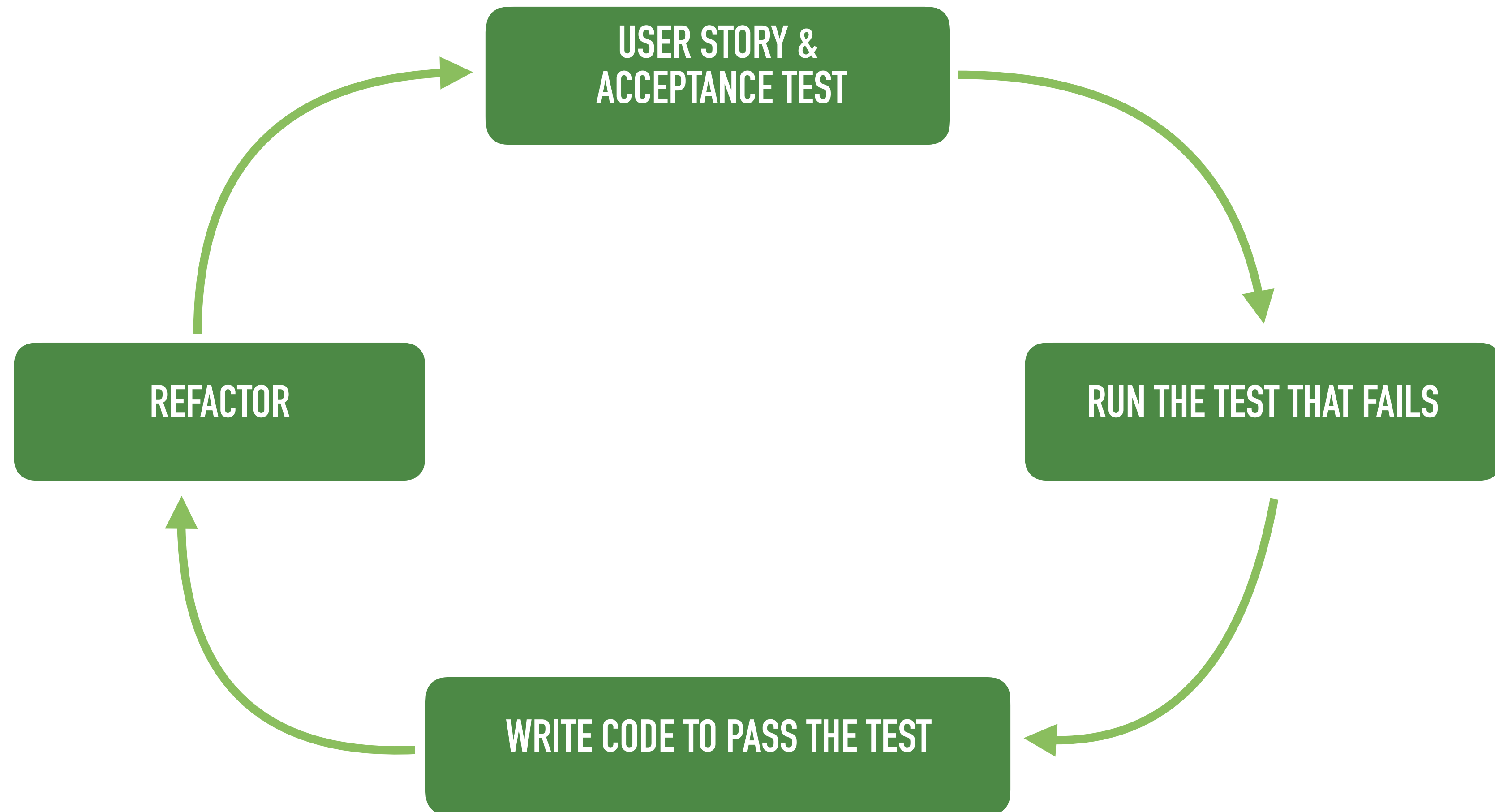
TEST DRIVEN DEVELOPMENT



TEST DRIVEN DEVELOPMENT



BEHAVIOUR DRIVEN DEVELOPMENT



BDD = TDD AND ATDD + MORE

- ▶ Apply the "Five Why's" to each user story

FIVE WHY'S

- ▶ Repeating the question "Why?"
- ▶ Cause and effect relationship underlying a problem

BDD = TDD AND ATDD + MORE

- ▶ Apply the "Five Why's" to each user story
- ▶ Thinking "from the outside in"

THINKING FROM OUTSIDE IN

- ▶ What are we trying to do with the app?
- ▶ What is the purpose of this app?
- ▶ Think in terms of the app's behaviour from the user's perspective

BDD = TDD AND ATDD + MORE

- ▶ Apply the "Five Why's" to each user story
- ▶ Thinking "from the outside in"
- ▶ Describe behaviours in a easy to understand language

BETTER COMMUNICATION

- ▶ Can be understood by testers and non-technical people

BDD BENEFITS

- ▶ Better planning at the beginning of the development process
- ▶ Minimise writing code that never gets used
- ▶ Avoid developing features nobody uses
- ▶ Assurance of the functional integrity of the app

BDD BENEFITS

- ▶ Works with agile
- ▶ Small multifunctional team
- ▶ Individual developer - use different perspectives

BDD CHALLENGES

- ▶ Setup can be difficult
- ▶ UI Automation
- ▶ Framework dependent - UI testing framework
- ▶ Some hardware functionalities are difficult to automate or simulate
- ▶ Realistically some features are better to be manually tested

~~BDD CHALLENGES~~ UI AUTOMATION CHALLENGES

- ▶ Setup can be difficult
- ▶ UI Automation
- ▶ Framework dependent - UI testing framework
- ▶ Some hardware functionalities are difficult to automate or simulate
- ▶ Realistically some features are better to be manually tested

PART 2: BDD WITH CALABASH

WORKSHOP PART 2

- ▶ Part 2: BDD with Calabash
 - ▶ What is Calabash
 - ▶ Calabash, Cucumber, Gherkin?!
 - ▶ Features & Scenarios
 - ▶ Setup Calabash
 - ▶ Explore Calabash
 - ▶ Try some BDD

WHAT IS CALABASH

CALABASH

<http://calaba.sh>

- ▶ Framework
- ▶ Consists of libraries that enable test code to interact programmatically with the app
- ▶ Supports both Android and iOS
- ▶ Open source and free
- ▶ Developed and maintained by Xamarin

CUCUMBER

<https://cucumber.io>

- ▶ Testing tool
- ▶ Runs automated acceptance tests
- ▶ Written in Ruby
- ▶ Uses plain language parser Gherkin

[https://en.wikipedia.org/wiki/Cucumber_\(software\)](https://en.wikipedia.org/wiki/Cucumber_(software))

GHERKIN

- ▶ Designed to be non-technical and human readable
- ▶ Script for automated acceptance tests
- ▶ Simple documentation for the code under test
- ▶ Understood by all three amigos

FEATURES & SCENARIOS

- ▶ Cucumber tests are divided into Features
- ▶ Features are subdivided into Scenarios
- ▶ Scenarios are sequence of Steps

FEATURES & SCENARIOS

FEATURE

SCENARIO 1

SCENARIO 2

SCENARIO 2

USER STORY

ACCEPTANCE TEST 1

ACCEPTANCE TEST 2

ACCEPTANCE TEST 2

FEATURES & SCENARIOS

- ▶ Feature = user story
- ▶ Scenarios = acceptance tests

FEATURE

- ▶ Use case that describes a specific function of the app
- ▶ Keyword is **Feature**:

SCENARIOS

- ▶ Flow of events
- ▶ Mapped to a single test case
- ▶ Keyword is **Scenario:**

STEPS

- ▶ Scenario is defined by a sequence of steps
- ▶ Following keywords are used:
 - ▶ **Given** - preconditions
 - ▶ **When** - actions taken by a user
 - ▶ **Then** - outcome
 - ▶ **And** - logical and
 - ▶ **But** - Logically same as **And**, but used in the negative form

EXAMPLE

Feature: Upload food pic for share

Scenario: Picture is picked from Photos

Given the user is logged in

When the upload is pressed

And user has selected a photo

Then the picture should start uploading

And progress bar is displayed

ACTIVITY

- ▶ Convert some of the user stories into Features and Scenarios
- ▶ Feature = user story
- ▶ Scenarios = acceptance tests

Feature: Upload food pic for share

Scenario: Picture is picked from Photos

Given the user is logged in

When the upload is pressed

And user has selected a photo

Then the picture should start uploading

And progress bar is displayed

CALABASH SETUP

SETTING UP CALABASH

- ▶ Calabash and cucumber are both installed and run using bundler
- ▶ Install bundler

```
$ gem install bundler
```

DON'T HAVE GEM?

- ▶ Install ruby
- ▶ Recommended to use managed ruby environment such as rbenv
- ▶ Install using homebrew

INSTALL RUBY

```
$ brew update  
$ brew install rbenv  
$ rbenv init  
$ rbenv install 2.4.1  
$ rbenv global 2.4.1
```

DON'T HAVE HOMEBREW?

- ▶ Install homebrew

<https://brew.sh/>

```
/usr/bin/ruby -e "$(curl -fsSL https://  
raw.githubusercontent.com/Homebrew/install/  
master/install)"
```

TROUBLESHOOTING



Gem::FilePermissionError

```
$ gem install bundler
Fetching: bundler-1.15.4.gem (100%)
ERROR:  While executing gem ... (Gem::FilePermissionError)
    You don't have write permissions for the /Library/Ruby/Gems/
2.0.0 directory.
```

```
$ which gem
/usr/bin/gem
```


TROUBLESHOOTING



Gem::FilePermissionError

- ▶ <https://github.com/rbenv/rbenv/issues/879>
- ▶ Add the following in ~/.bash_profile

```
export PATH="$HOME/.rbenv/bin:$PATH"  
eval "$(rbenv init -)"  
export PATH="$HOME/.rbenv/plugins/ruby-build/bin:$PATH"
```

INSTALL CALABASH FRAMEWORK

- ▶ Step 1: Clone example project
- ▶ Step 2: Update bundle
- ▶ Step 3: Run the app in simulator
- ▶ Step 4: Run Calabash interactive console

<https://github.com/calabash/calabash-ios-example>

<https://github.com/calabash/calabash-ios/wiki/Getting-Started>

STEP 1: CLONE EXAMPLE PROJECT

```
Git clone https://github.com/summermk/  
dw17-bdd-workshop.git
```

<https://github.com/summermk/dw17-bdd-workshop>

STEP 2: SETUP CALABASH FRAMEWORK

```
# Install the necessary gems  
$ bundle update
```

STEP 3: RUN THE APP

- ▶ Open the project bdd-workshop.xcodeproj in Xcode
- ▶ Update the signing profile
- ▶ Build and run on a simulator

STEP 3: RUN THE APP

In the Console you should see output like this:

```
DEBUG CalabashServer:222 | Creating the server: <LPHTTPServer: 0x7fe97a507ef0>  
DEBUG CalabashServer:223 | Calabash iOS server version: CALABASH VERSION: 0.16.4
```

STEP 4: RUN INTERACTIVE CONSOLE

```
$ bundle exec calabash-ios console
```

STEP 4: RUN INTERACTIVE CONSOLE

```
$ bundle exec calabash-ios console
```

```
Running irb...
```

```
/Users/mirakim/.rbenv/versions/2.4.1/lib/ruby/gems/2.4.0/gems/edn-1.1.1/lib/edn/core_ext.rb:97:
```

```
warning: constant ::Bignum is deprecated
```

```
##### Useful Methods #####
```

```
ids => List all the visible accessibility ids.
```

```
labels => List all the visible accessibility labels.
```

```
text => List all the visible texts.
```

```
marks => List all the visible marks.
```

```
tree => The app's visible view hierarchy.
```

```
flash => flash(<query>); Disco effect for views matching <query>
```

```
verbose => Turn debug logging on.
```

```
quiet => Turn debug logging off.
```

```
copy => Copy console commands to clipboard.
```

```
clear => Clear the console.
```

```
Calabash says, "Det ka æn jå væer ei jált"
```

```
Attached to: #<Launcher: DeviceAgent/ios_device_manager>
```


STEP 4: RUN INTERACTIVE CONSOLE

```
> start_test_server_in_background
```

STEP 4: RUN INTERACTIVE CONSOLE

```
calabash-ios 0.20.5> start_test_server_in_background
```

```
INFO: Detected app at path:
```

```
INFO: /Users/mirakim/Library/Developer/Xcode/DerivedData/bdd-workshop-gwtglvxmlldzhifirspoqmrwajgc/  
Build/Products/Debug-iphonesimulator/bdd-workshop.app
```

```
INFO: Modification time of app: Sun 20 Aug 2017 18:03:38 NZST
```

```
INFO: If this is incorrect, set the APP variable and/or rebuild your app
```

```
INFO: It is your responsibility to ensure you are testing the right app.
```

```
#<Launcher: DeviceAgent/ios_device_manager>
```

```
calabash-ios 0.20.5>
```

STEP 4: RUN INTERACTIVE CONSOLE

```
> query("button")
```

STEP 4: RUN INTERACTIVE CONSOLE

```
calabash-ios 0.20.5> query("button")
[
  [0] {
    "id" => nil,
    "description" => "<UIButton: 0x7fdb5870a160; frame = (330.5 6; 40 30); opaque = NO; layer = <CALayer: 0x618000035800>>",
    "label" => "Add",
    "frame" => {
      "y" => 6,
      "x" => 330.5,
      "width" => 40,
      "height" => 30
    },
    "accessibilityElement" => true,
    "value" => nil,
    "alpha" => 1,
    "enabled" => true,
    "visible" => 1,
    "selected" => false,
    "class" => "UIButton",
    "rect" => {
      "y" => 26,
      "center_x" => 350.5,
      "center_y" => 41,
      "x" => 330.5,
      "width" => 40,
      "height" => 30
    }
  }
]
```

STEP 4: RUN INTERACTIVE CONSOLE

```
> query("button")
```

```
> touch("button")
```

STEP 4: RUN INTERACTIVE CONSOLE

```
> touch("textField")
```

```
> keyboard_enter_text("Hello DevWorld")
```

EXPLORE INTERACTIVE CONSOLE

10 MIN

Calabash iOS Wiki - [Explore Interactively](https://github.com/calabash/calabash-ios/wiki/Getting-Started#explore-interactively)

```
> query("button")  
> query("button marked: 'Add'")  
> touch("button")  
> keyboard_enter_text("Hello DevWorld")
```

ADD TASK FEATURE

Feature: Add Task Feature

As a user

I want to add a task

So that I can be reminded to do it

SCENARIOS

@add_task

Scenario: Add task screen

Given the app has launched

When I press the "Add" button

Then I see the screen "New task"

SCENARIOS

@add_task

Scenario: Add a new task

Given that I am in the new task screen

When I type in the task "buy some bread"

And I press the "Done" button

Then I see the screen "To Do List"

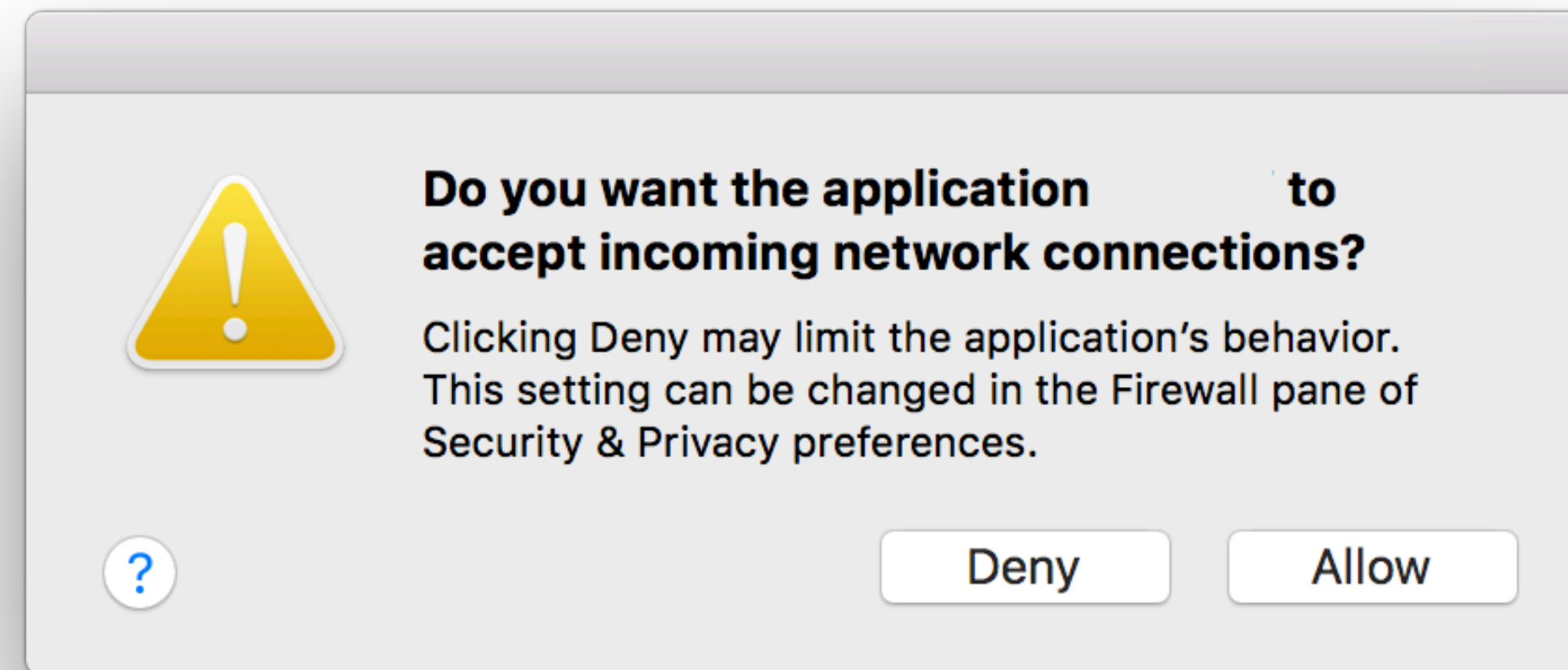
And I see the task with title "buy some bread"

RUN CUCUMBER

```
$ bundle exec cucumber
```

TROUBLESHOOTING

- ▶ Firewall warning keeps showing



TROUBLESHOOTING

#temporarily shut firewall off:

```
sudo /usr/libexec/ApplicationFirewall/socketfilterfw --setglobalstate off
```

#put Xcode as an exception:

```
/usr/libexec/ApplicationFirewall/socketfilterfw --add /Applications/  
Xcode.app/Contents/MacOS/Xcode
```

#put iOS Simulator as an exception:

```
/usr/libexec/ApplicationFirewall/socketfilterfw --add /Applications/  
Xcode.app/Contents/Developer/Applications/Simulator.app/Contents/MacOS/  
Simulator
```

#re-enable firewall:

```
sudo /usr/libexec/ApplicationFirewall/socketfilterfw --setglobalstate on
```

<https://stackoverflow.com/questions/9845502/how-do-i-get-the-mac-os-x-firewall-to-permanently-allow-my-ios-app>

RUN TESTS WITH TAGS

```
$ bundle exec cucumber --tags @add_task
```

WRITE CUSTOM STEPS

```
@add_task
```

```
Scenario: Add task screen
```

```
# features/add_task.feature:7
```

```
Given the app has launched
```

```
# features/steps/sample_steps.rb:1
```

```
When I press the "Add" button
```

```
# calabash-cucumber-0.20.5/features/  
step_definitions/calabash_steps.rb:30
```

```
Then I see the screen "New task"
```

```
# features/add_task.feature:10
```

HELPER MESSAGE

You can implement step definitions for undefined steps with these snippets:

```
Then(/^I see the screen "([^"]*)"$/) do |arg1|  
  pending # Write code here that turns the phrase above into  
  concrete actions  
end
```


TRY WRITING THE STEPS

BDD BENEFITS

- ▶ Minimise writing code that never gets used
- ▶ Avoids implementing features that don't add value
- ▶ Better communication between all three amigos
- ▶ Can verify the app's functional integrity
- ▶ Closely mimics the real world usage

BDD CHALLENGES

- ▶ Can be difficult to learn for people who are not familiar with TDD
- ▶ Automating all the acceptance tests may be difficult or time consuming
- ▶ Can become tightly coupled with the automation testing framework or tools

The use of BDD requires no particular tools or programming languages, and is primarily a conceptual approach; to make it a purely technical practice or one that hinges on specific tooling would be to miss the point altogether

Common Pitfalls for BDD, From Agile Alliance

WHAT'S NEXT

- ▶ Check out the examples from calabash
 - ▶ <http://calaba.sh/>
 - ▶ <https://github.com/calabash/calabash-ios>
- ▶ Try writing User Stories and Acceptance Tests for the new app or new functionalities
- ▶ Integrate calabash with your app

MIRA KIM



COME AND SAY HI!



@mrk144



summermk



mirakim135@gmail.com

