Finding the Sweet Spot

The Art of Writing Scenarios

What is a Scenario?

An example of a system behavior

Written in the language of the business

Concrete: Real, Solid

One example of how your system will be used to satisfy a need

A collection of scenarios describe the behavior of an entire feature

Scenario
Scenario
Scenario
Scenario
Scenario
Scenario
Scenario
Scenario
Scenario

‘Good Enough’
representation of
a feature’s behavior

Why Scenarios?

The needs of the stakeholders are kept top-of-mind

Deepens shared understanding and leads to shared language

Quality is discussed from the outset

May lead to valuable automated tests

Feature: *feature title*

narrative

Scenario: *scenario title*

Given *this state*
When *I change state*
Then *I expect this outcome*

The Bones

Feature: *feature title*

Feature Title

Describes the behavior you should expect to see in the scenarios

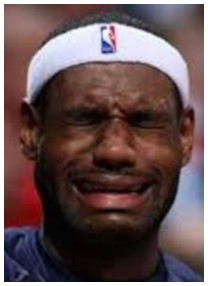
Gives a sense of all that is getting done

Tells whose behavior it is

Crisp

Not generalized

Account Management



ATM Withdrawals

Account Holder Withdraws Cash

Feature Narrative

Names the role

Identifies the need that will be satisfied

Identifies the benefit that will be received

As a [role] I want [feature] so that [benefit]

In order to [benefit] as a [role] I want to [feature]

Feature: Account Holder withdraws cash

As an Account Holder
I want to withdraw cash from an ATM
So that I can get money when it is convenient for me

Don’t be constrained by the template

Go beyond it to ensure intent is clear

Feature: *feature title*

narrative

Scenario: *scenario title*

Scenario Title

Short and descriptive

Is in support of the feature

See consistency but a difference with the other scenario titles

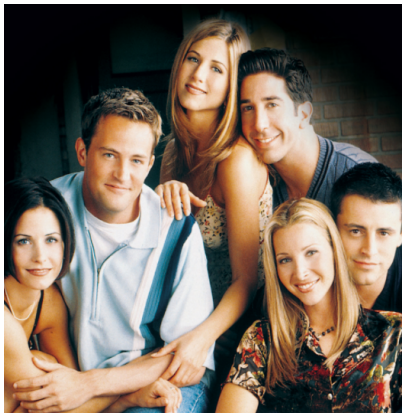
Generating Scenario Titles

1) Identify the things in the scenarios

2) Discover rules related to the things

Generating Scenario Titles

3) Brainstorm



The one where Underdog gets away

The one where the monkey gets away

The one where Ross hugs Rachel

F·R·I·E·N·D·S

Happy Path
Happy Path
Edge Case
Edge Case
Edge Case
Rule's Outcome
Rule's Other Outcome
That One Strange Case

Full representation
of the feature's behavior

As a Team

Build and deepen your shared understanding

Then turn you shared understanding into your shared language

Then drive that shared language into everything

Exercise

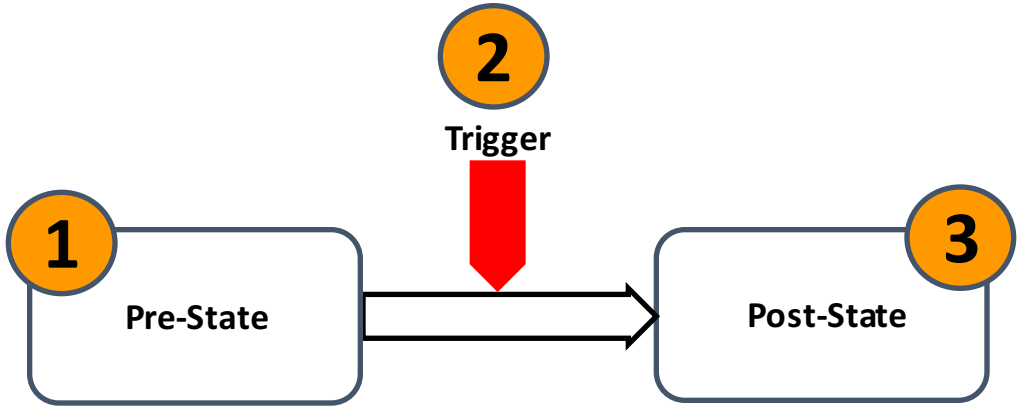
As a team (*table*):

- Write the feature title and narrative for the *withdrawal* feature
- Then generate as many scenario titles as you can

Exercise Review

The 3 Steps

1. Put the system in a particular state / create context
2. Do something to change that state
3. Examine the new state / check the outcome



Given *this state*
When I *change the state*
Then *examine the new state*

The '**Given**' should define all of, and no more than, the necessary context

Extra Givens are **distractions**.

Missing Givens are **assumptions**.

If you can get different outcomes from the Givens that you provided, then you're missing some

Scenario: Account has sufficient funds

Given the account balance is $500

And my card is valid

And the machine contains enough money

Scenario: Card has been disabled

Given the card is disabled

Keep the '**When**' simple

- It should describe the feature
- Most scenarios in the feature should revolve around the same event
- If you Give the When a different context you may get a different outcome
- Ensure the When is what triggers the change of state
  - Try to isolate on one trigger

Not every user action is part of the trigger

Scenario: Account has sufficient funds

Given my account balance is $500
And my card is valid
And my machine contains enough money

When I enter my PIN
And I select ‘withdrawal’
And I request $50

Not every user action is part of the trigger

Scenario: Account has sufficient funds

Given my account balance is $500

And my card is valid

And my PIN is valid

And my machine contains enough money

~~**When** I enter my PIN~~

~~**And** I select 'withdrawal'~~

When I request $50

The '**Then**' is the user's 'Then'

The outcomes examined must be ones **observable by the user,** not something buried inside the system.

Should be related to the business value found in the feature narrative.

Relative to the **Given** and the **When**.

Scenario: Account has sufficient funds

Given my account balance is $500
And my card is valid
And my PIN is valid
And my ATM contains enough money

When I request $150

Then the ATM should dispense $150
And my account balance should be $350
And my card should be returned

Exercise

- For the feature your table selected
- Find a partner
- Write the scenarios (Given-When-Then) for the scenario titles you identified

Having Trouble Getting Started?
Here's Some Inspiration

Shitty First Drafts

Now, practically even better news than that of short assignments is the idea of shitty first drafts. All good writers write them. This is how they end up with good second drafts and terrific third drafts. People tend to look at successful writers, writers who are getting their books published and maybe even doing well financially, and think that they sit down at their desks every morning feeling like a million dollars, feeling great about who they are and how much talent they have and what a great story they have to tell; that they take in a few deep breaths, push back their sleeves, roll their necks a few times to get all the cricks out, and dive in, typing fully formed passages as fast as a court reporter. But this is just the fantasy of the uninitiated. I know some very great writers, writers you love who write beautifully and have made a great deal of money, and not *one* of them sits down routinely feeling wildly enthusiastic and confident. Not one of them writes elegant first drafts. All right, one of them does, but we do not like her very much. We do not think that she has a rich inner life or that God likes her or can even stand her. (Although when I mentioned this to my priest friend Tom, he said you can safely assume you've created God in your own image when it turns out that God hates all the same people you do.)

Very few writers really know what they are doing until they've don it. Nor do they go about their business feeling dewy and thrilled. They do

"Shitty First Drafts," *Bird by Bird: Some Instructions on Writing and Life*, by Anne Lamott. Copyright © 1994 by Anne Lamott. Used by permission of Pantheon Books, a division of Random House, Inc.

For me, and most of the other writers I know, writing is not rapturous. In fact, the only way I can get anything written at all is to write **really,** really shitty first drafts.

The first draft is the child's draft, where you let it all pour out and then let it romp all over the place, knowing that no one is going to see it and that you can shape it later. You just let this childlike part of you channel whatever voices and visions come through and onto the page. If one of the characters wants to say, "Well so what, Mr. Poopy Pants!," you let her. No one is going to see it. If the kid wants to get into really sentimental, weepy, emotional territory you let him. Just get it all down on paper, because there may be something great in those six crazy pages that you would never have gotten to by more rational, grown-up means. There may be something in the very last line of the very last paragraph on page six that you just love that is so beautiful or wild that you now know what you're supposed to be writing about, more or less, or in what direction you might go—but there was no way to get to this without first getting through the first five and a half pages.

The SFD

Exercise

- For the feature your table selected
- Find a partner
- Write the scenarios (Given-When-Then) for the scenario titles you identified

Having Trouble Getting Started?
Here's Some Inspiration

Exercise Review

Improving the SFD

```
Feature: adding a task

  Scenario: Adding a task
    Given the network is established
    And there are other members in the network
    And the caregiver is logged in
    And and the system is on the manage my task page
    When I click the Add new task button
    And I enter the "Game 7 - NBA finals" into the task name field
    And I enter "June 19th, 2016" into the date text field
    And I enter "8:00 pm" into the time text field
    And I enter "Go Cavs!" into the comments text field
    And I click the Save Todo button
    Then I should see "Your task has been saved."
    And the task database should be used
    And an email is sent to the network memebers
```

Improving the SFD

1. Cohesive Scenario Titles

- Do they all support the feature?
- Does the feature title include the actor?
- Are the words consistent?

2. Narrative

- Can you find all of these?
- the user
- the need
- the benefit

3. Remove the noise

- Remove unnecessary data and details.

4. Focus on the behavior

- Remove implementation details and scriptiness.
- Imperative: Instructing how to do it.
- Versus
- Declarative: Describing what the goal is
- Tips:
- Ask, "How would a non-technical user describe this step?"
- Try and remove any details of the underlying technology
- Look for steps that can be combined into a single domain action

5. The Pre-Condition / Context - Given

- Are all really necessary?
- Anything missing?
- Are there common pre-conditions that could be part of a background?
- Look for steps that can be combined into a single domain pre-condition.

6. The Action - When

- Isolate the trigger
- Are my whens cohesive?
- Are there given's hiding in your action?
- Is the when focused on only changing the state of the given pre-conditions? It should describe the feature?
- Look for steps that can be combined into a single domain action.

7. The Outcome - Then

- Should focus on the outcome of the action
- The outcomes examined must be ones observable by the user, not something buried inside the system

Additional notes

The words matter

- Create a vocabulary shared by everyone.
- Use the language of the business.
- Be consistent
- Be disciplined
- The words will end up in your codebase. (Domain-Driven Design)