

Build-In Quality into Your Continuous Delivery Pipeline



Ken Pugh

Chief Consultant
Ken Pugh, Inc.



Build Quality Into Your Continuous Delivery Pipeline

An immersive application of ASE/BDD/TDD to transform teams

- The Hypothesis
- The Approach
- The Results
- The Pitch

The Context



The Hypothesis

The image features a dark blue background with a white wavy line pattern that creates a grid-like effect. Two orange triangles are positioned on the page: one in the top right corner and another in the bottom center.

The Hypothesis

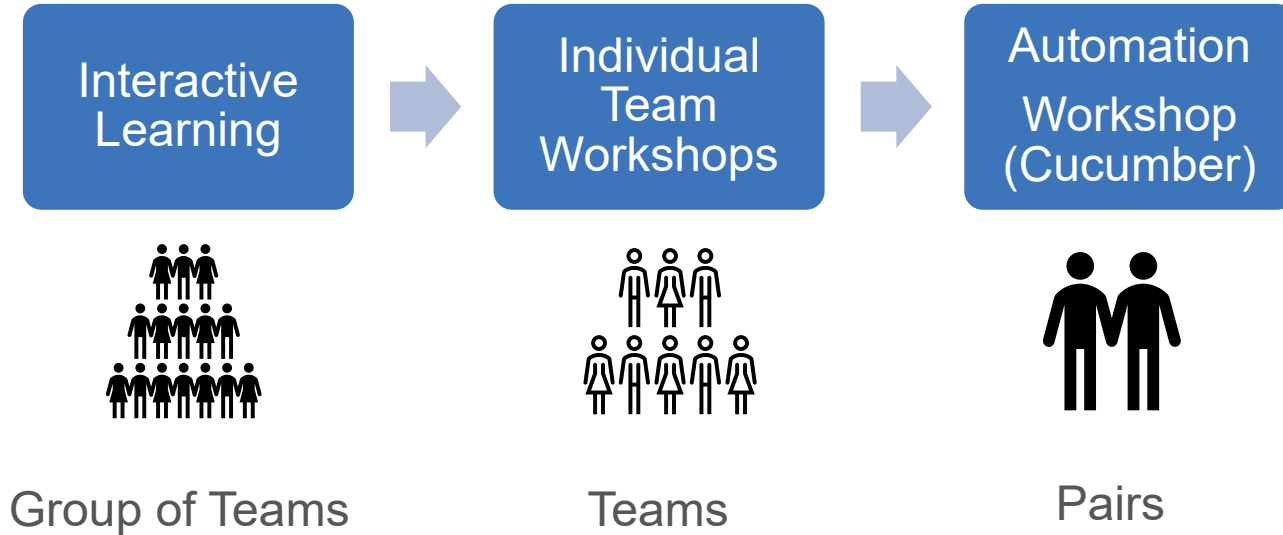
Establish a new way of working
through full team *immersive learning*
focused on behavior of the system from the outside world

- ▶ We believe that immersive learning of Behavior Driven Development will result in:
 - Amplifying flow, collaboration and quality
 - Changing the culture
 - Creating empathy
 - Better requirements
- ▶ As measured by feedback from the teams and reduction in defects and rework

The Approach

The image features a dark blue background with white wavy lines that create a grid-like pattern. Two orange triangles are positioned on the page: one in the top right corner and another in the bottom center.

Flow of BDD Immersion



Agile Software Engineering

*eXtreme
Programming*

*Behavior Driven
Development*

*Lean Agile Principles
and Practices*

*Design Patterns and
Practices*

Agile Modeling

Code Quality

Value: Build-In Quality



Principle: #4 – Build incrementally with fast, integrated learning cycles

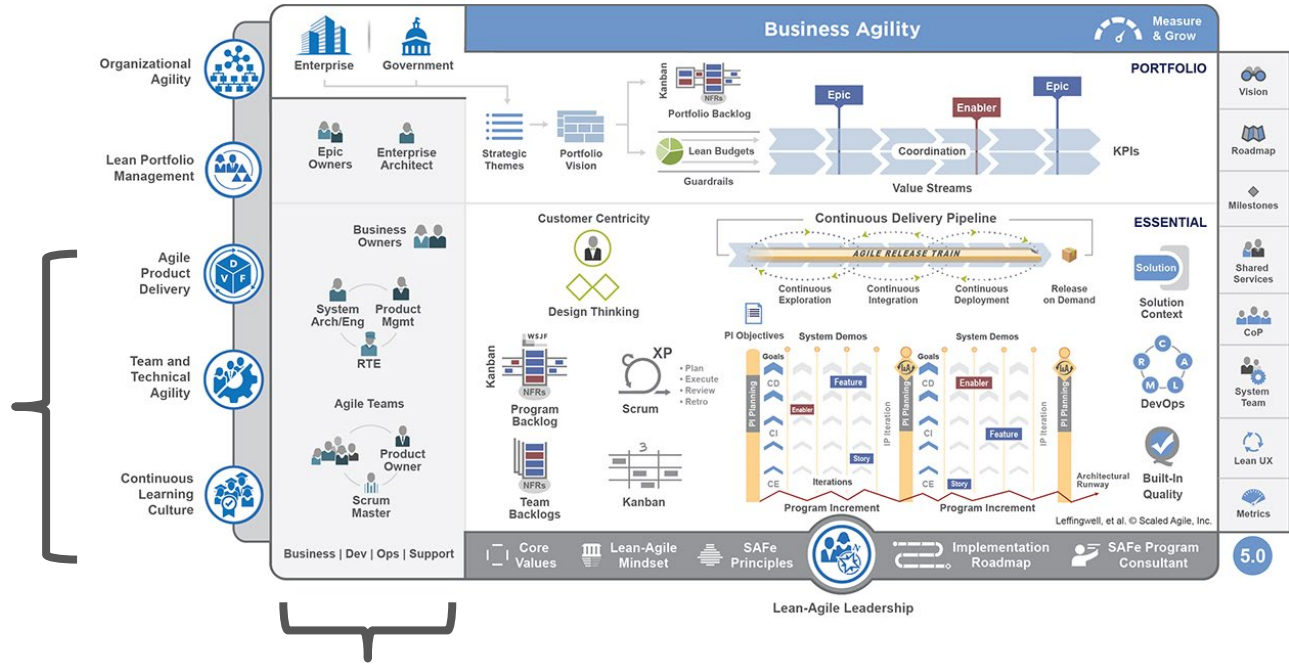


Practice: Test-First

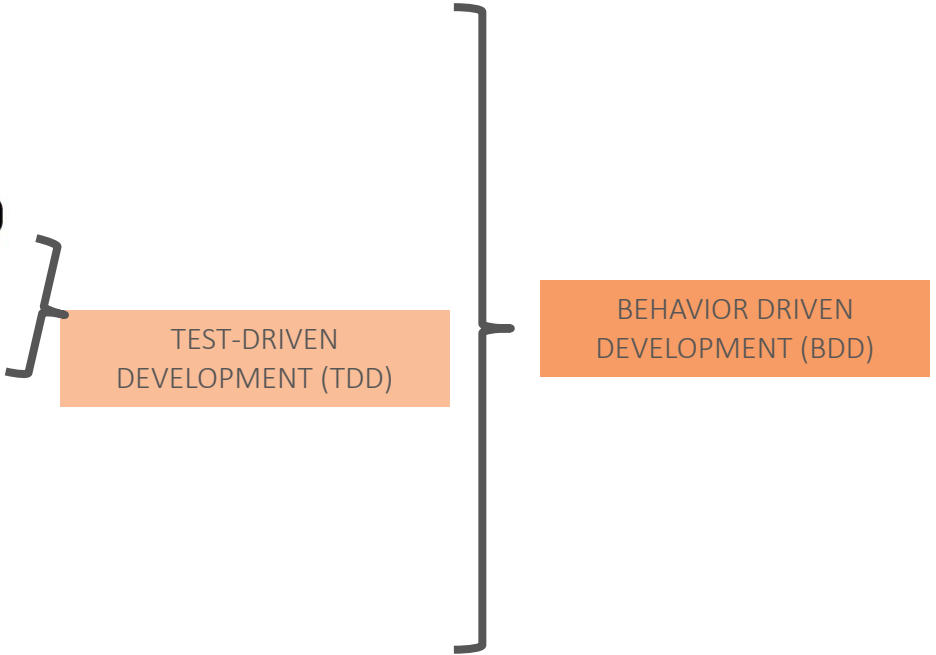
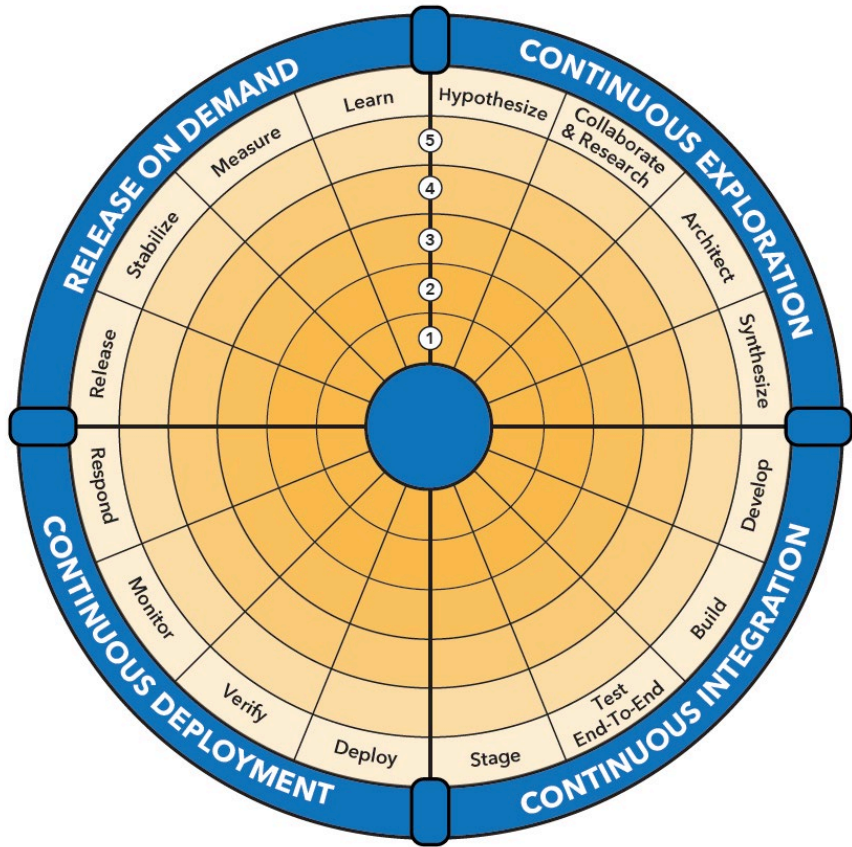
Behavior Driven Development

Test Driven Development

BEHAVIOR DRIVEN DEVELOPMENT (BDD)



TEST-DRIVEN DEVELOPMENT (TDD)

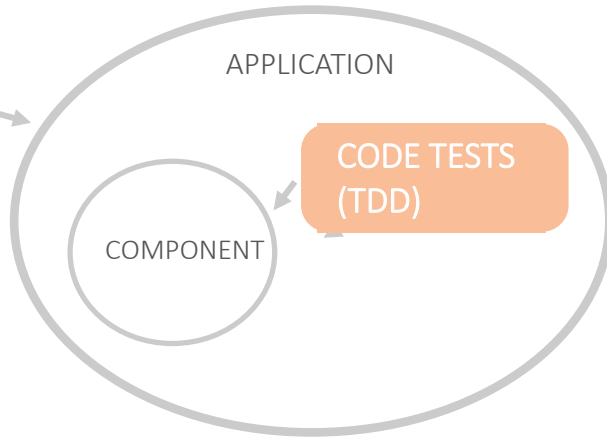


The Context

THE TRIAD



STORY TESTS
(BDD)



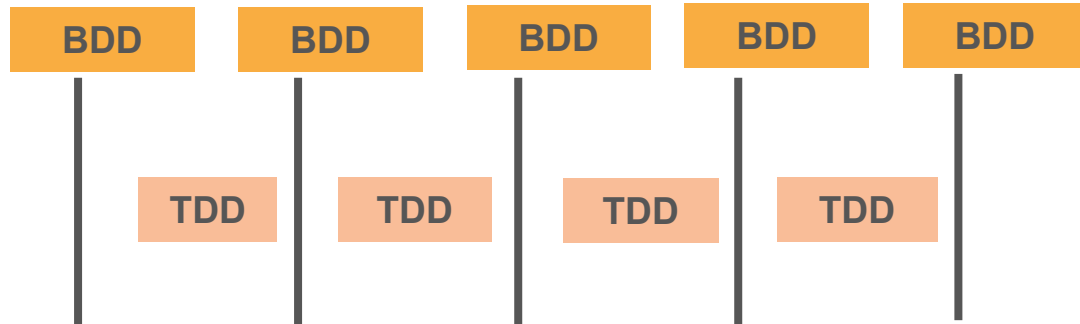
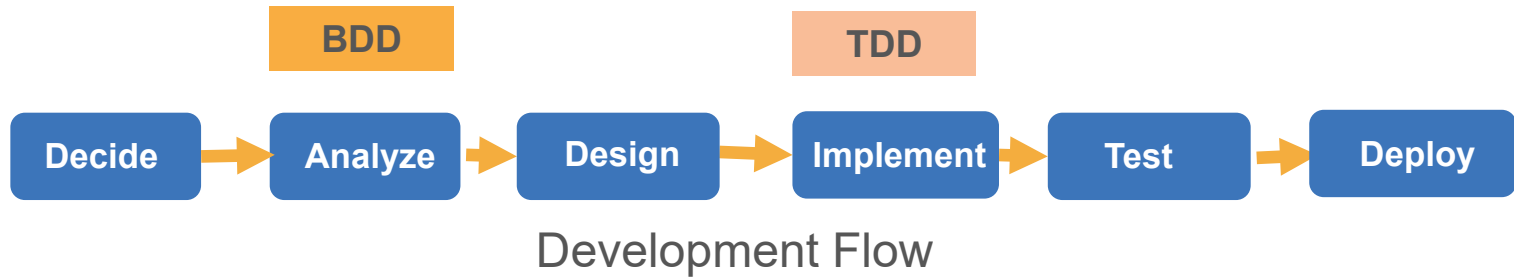
PAIRS



- ▶ The Triad documents desired behavior of the system.
 - That expected behavior becomes the tests
- ▶ Every test is a requirement that a system must meet
- ▶ Every requirement should have a test for it



When are behaviors / tests defined?



The Story

As an automobile policy reviewer,

I do not want to re-review a policy up for renewal if no new incidents have occurred since the last renewal

so that reviewer time is saved

Domain Terms

Incidents - Major

Moving violation
At-Fault Accident

Incidents – Minor

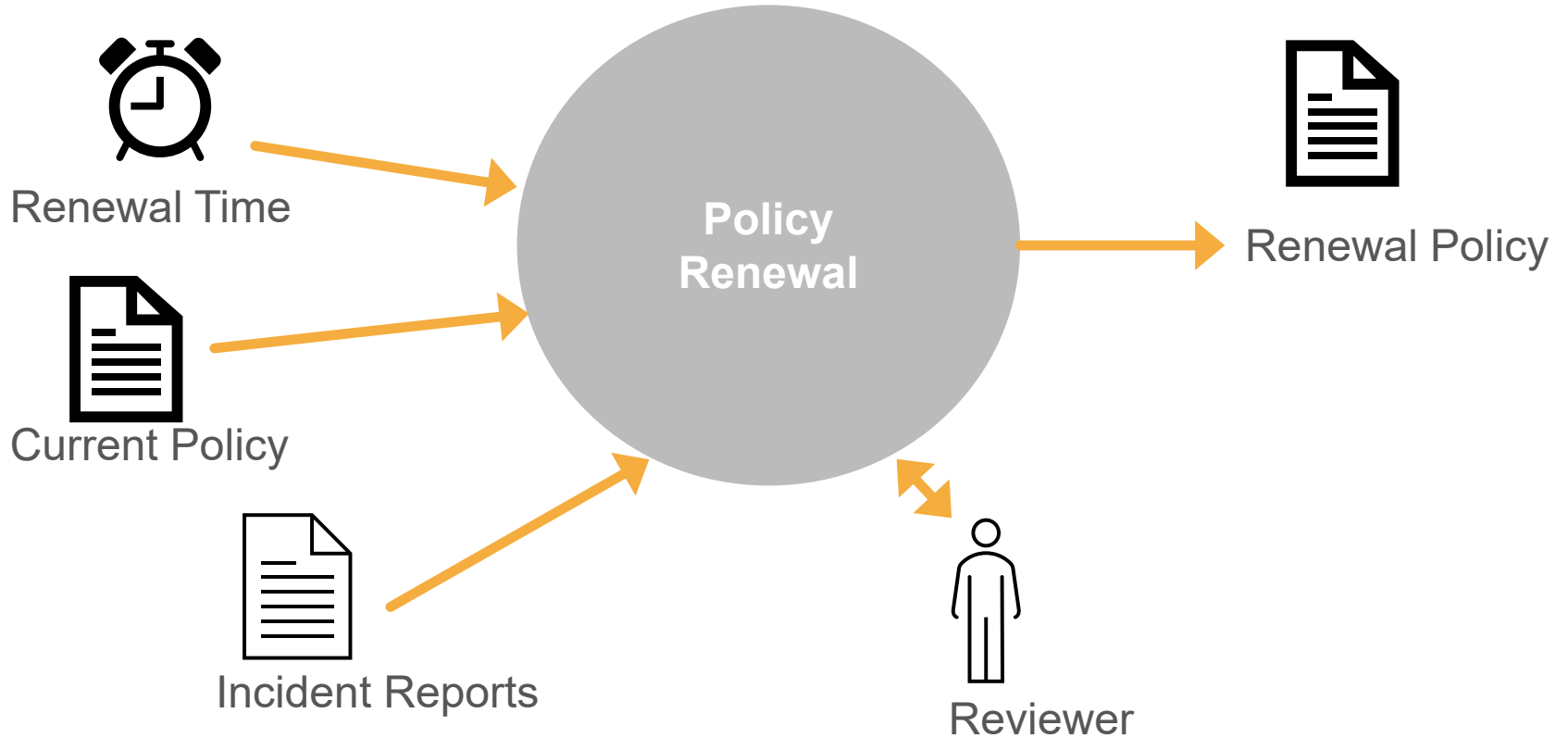
Parking ticket

Incident

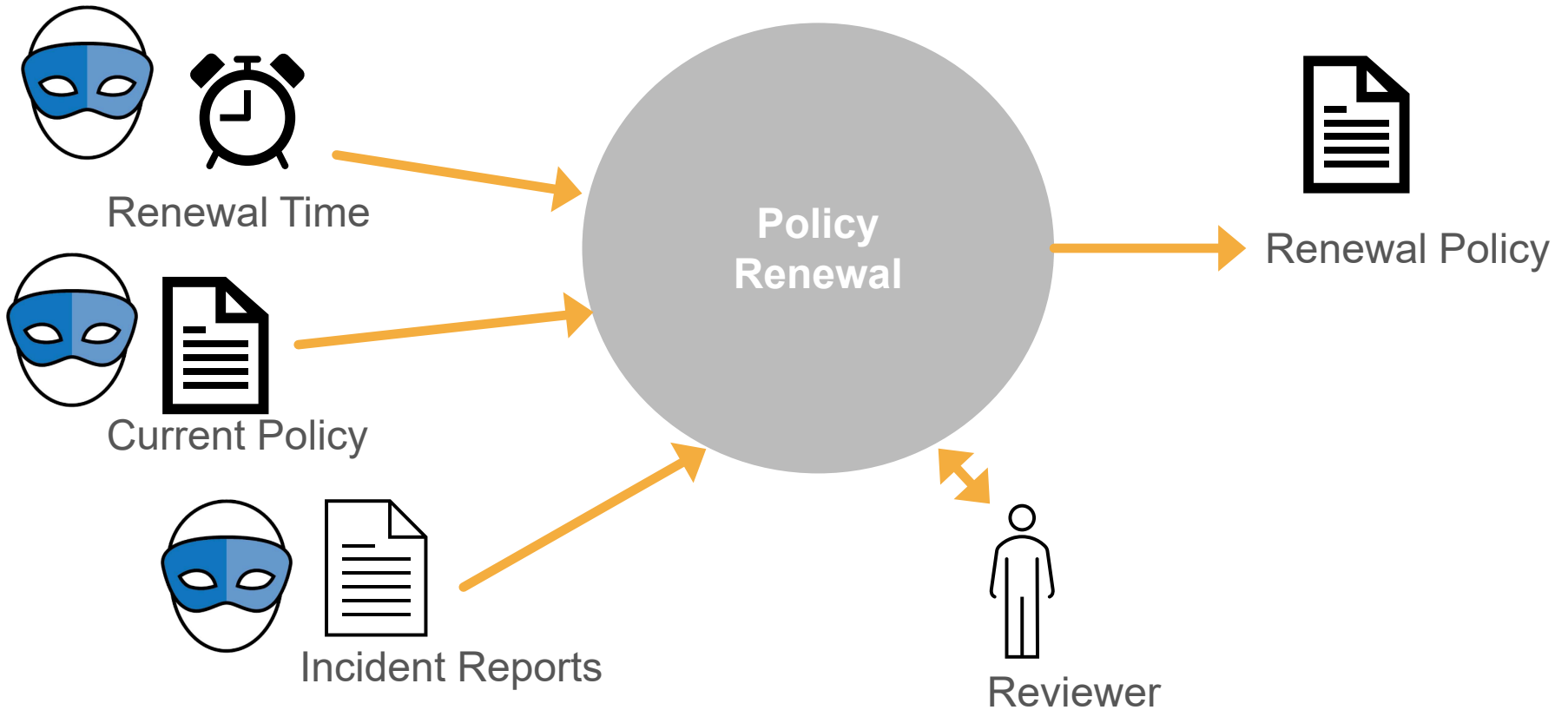
Type	Date Occurred	Date Reported	Description
Major	January 1, 2020	January 31, 2020	Speeding

Renewal Preparation Date
– one month prior to Renewal Date

Context of the story



Test Doubles for Automation



Acceptance Criteria as a Scenario

Acceptance Criteria:

- Request for review is not sent for policies with no new incidents

Template for Scenario of a Flow

```
Given current state  
When action or event occurs  
Then state change or output
```

Scenario

```
Given a policy with no new incidents  
When policy is renewed  
Then no review is requested
```

Stories have many scenarios

Scenario with trivial incidents

```
Given a policy with only trivial incidents  
When policy is renewed  
Then NO review is requested
```

Scenario with major incident

```
Given a policy with one or more major incidents  
since the last renewal  
When policy is renewed  
Then a review is requested
```

Acceptance Criteria into Testable Scenarios

```
Given a policy with one or more major incidents
since the last renewal
When policy is renewed
Then a review is requested
```

Test defines specific pass/fail behavior – a story's details

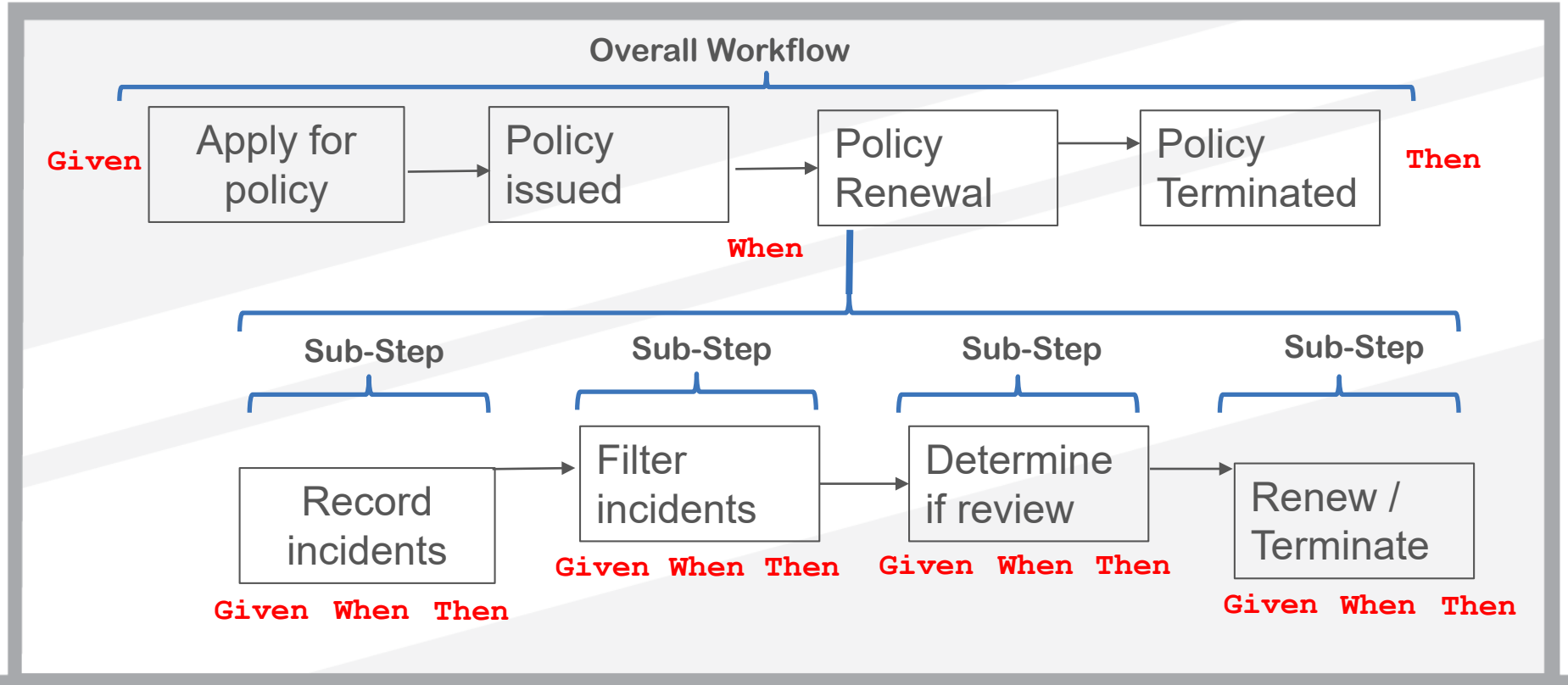
```
Given a policy
|Owner | Renewal Date |
|Sam   | March 1, 2021 |
And incidents are
| Type | Date Occurred | Date Reported |
| Major| January 1,2020 | February 2, 2020 |
When policy is prepared for renewal on
| Preparation Date |
| February 1, 2021 |
Then a review is requested
```

Another Acceptance Criteria into Testable Scenario

```
Given a policy with no new incidents
When policy is renewed
Then NO review is requested
```

```
Given a policy
|Owner | Renewal Date |
|Sam   | March 1, 2021 |
And incidents are
| Type | Date Occurred | Date Reported |
| Major| January 1,2020 | January 31, 2020 |
When policy is prepared for renewal on
| Preparation Date |
| February 1, 2021 |
Then a review is NOT requested
```

End-to-end and step scenarios



Business Rules

Incidents	Renewal Preparation Date	Review Required
Type Date Occurred Date Reported Major January 1,2020 January 31, 2020	February 1, 2021	No - reviewed last time
Type Date Occurred Date Reported Major January 1,2020 January 31, 2020	January 31, 2021	Yes ??
Type Date Occurred Date Reported Major January 1,2020 January 31, 2020	January 30, 2021	Yes
Type Date Occurred Date Reported Major January 1,2018 January 31, 2020	January 30, 2021	No - more than 3 years

What to Do With Scenarios / Tests

Tests written in Gherkin

Store in source code control

Can usually be automated (Cucumber, SpecFlow, etc.)

Test Driven Development

- ▶ Design components with responsibilities specified by tests
 - Components work together to pass the BDD tests
- ▶ No code goes in until the test goes on
 - Don't test code, code to the test



Incident

Type	Date Occurred	Date Reported
Major	January 1, 2020	January 31, 2020

```
class Incident {  
    ...  
}
```

Incidents

Type	Date Occurred	Date Reported
Major	January 1, 2020	January 31, 2020
Minor	January 2, 2020	February 1, 2020

```
class IncidentCollection {  
    Boolean reviewRequired()  
}
```

The Results

The image features a dark blue background with a white wavy line pattern that creates a grid-like effect. Two orange triangles are positioned on the page: one in the top right corner and another in the bottom center.

The Teams

Existing Agile
Teams with
SAFe

Collocated
when possible

Require full
cross-functional
team
participation

Use “real work”

Retro and adapt

Workshop Variations

- Remote PO participation
- 

- Everyone attended Cucumber
- 

- Really sophisticated stories
- 

Observer Notes

Every team went from zero to having solid usable examples of how to automate

There was so much discussion where they were trying to get on the same page that it made you wonder how they were working before – it was amazing

Immersive element is a huge differentiator –
not short scenarios and not examples, real work –
and fully cross functional –
lays the groundwork for future requirements clarity

Some Feedback

BDD makes
communicating
details and
requirements
MILES easier

Wondered how
they ever
worked
together before

This class allows us
to reduce ambiguity
and redundancy, and
better map tests.

We're rarely on the
same page. This
training requires and
reinforces it...this is
how training should be.

Now I know
why developers
hate me! (From
the Business)

Some More Feedback

I saw the whole perspective of a software lifecycle in a different and better way

Communication and understanding will be improved

Help me and the team to understand when something is “done”

Improve everyone’s understanding of our business requirements and produce better documentation

This will help us find issues earlier in the process

Still More Feedback

Help make sure
our code is
written to the
correct
specification

Better
implementation
and testing
resulting in fewer
defects

Use good structure to
reduce redundancy
and better mapping
to automated tests

Provide better
understanding to save
time and reduce
frustration

Frames the way
our team will
create stories
and tests

A few months after BDD/ATDD adoption, one team reported

- **Team “happiness factor” increased**
 - **Specifically, lead developer and tester are much happier**
 - **Less stress on testers**
- **More distributed testing effort across the sprint**
- **Helped to create/enhance “we are a team” feeling**
- **Fewer production defects**
- **Fewer test environment defects**
- **Less rework due to miscommunication**

Considerations for Teaching Remotely

Collaboration –
provide virtual tools
(e.g. Mural, Miro)

Remote working agreements –
necessary to create and protect a
rich interaction

TDD – solve how two
developers do virtual
pair programming

Technical Prework – do tech check
of tools / breakout plan

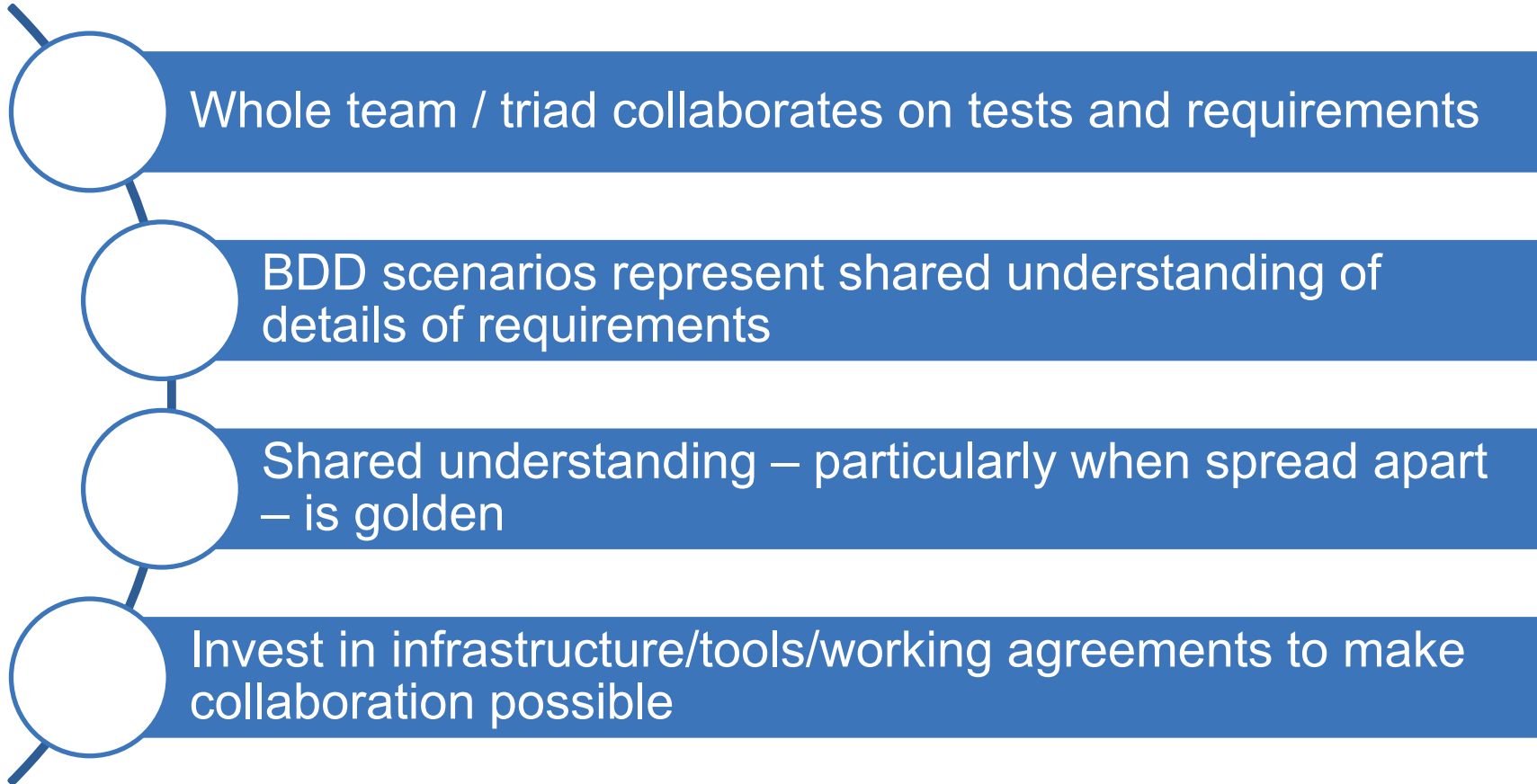
Prepare for remote troubleshooting –
how do students indicate they need help

Internal leaders to
promote engagement

The Pitch

The image features a dark blue background with a white wavy line pattern that creates a grid-like effect. Two orange triangles are positioned on the page: one in the top right corner and another in the bottom center. The text "The Pitch" is centered in a white, sans-serif font.

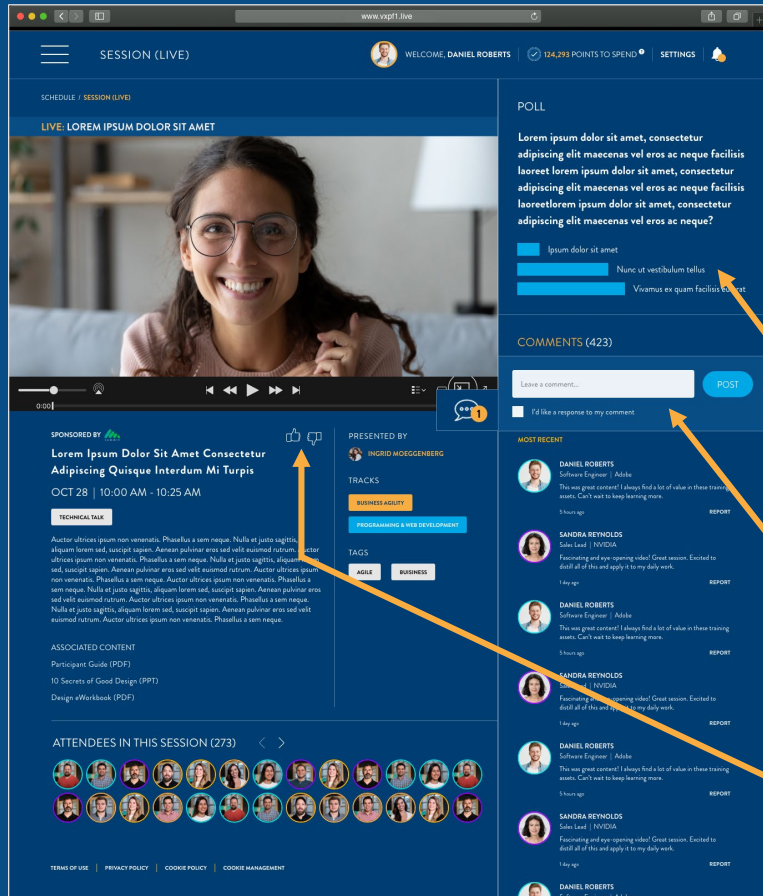
The Pitch – Enable your business and connect your teams



Join us at the Meet the Speaker Session!



Please refer to the agenda for scheduled times



Participate in polling, post comments, and rate sessions

1

Polling

2

Comment

3

Thumbs up or down

Thank you!