

Published: March 21, 2018

Updated on: Sep 11, 2018

What is BDD? An Introduction to Behavioral Driven Development

Category: Test Automation Types of Software Testing

What is BDD?

Behavioral Driven Development (BDD) is a software development approach that has evolved from **TDD** (Test Driven Development). It differs by being written in a shared language, which improves communication between tech and non-tech teams and stakeholders. In both development approaches, tests are written ahead of the code, but in BDD, tests are more user-focused and based on the system's behavior.



Choosing BDD

TDD works satisfactorily, as long as the business owner is familiar with the unit test framework being used and their technical skills are strong enough, which is not always the case. In these

circumstances, BDD has the advantage because the test cases can be written in a common language used by the stakeholders such as, for example, English. This access to clearer, low-jargon communication is probably the biggest advantage to using BDD, making it possible for collaboration between the technical and non-technical teams to run with improved efficiency.

Characteristics of BDD

As described above, the advantage to BDD test cases being written in a common language is that details of how the application behaves can be easily understood by all. For example, test cases can be written using real-time examples of the actual requirements, to explain the behavior of the system.

Essentials to have in place before implementing BDD

- Requirements should be converted into user stories that can define concrete examples.
- Each example should be a valid user scenario, rather than a mere test case.
- An understanding of the 'role-feature-reason' matrix and the 'given-when-then' formula.
- An awareness of the need to write 'the specification of the behavior of a class' rather than 'the unit test of a class'.

The 'Given-When-Then' formula BDD example

This is the proposed template for writing BDD test cases for a user story, which can be defined as:

Given a certain scenario

When an action takes place

Then this should be the outcome.

A practical example would be:-

Given the User has not entered any data on the form

When they click the submit button

Then proper validation messages should be show.

Role-Feature-Reason matrix

This template is also used in BDD, in order to aid user story creation. This template is defined as:-

- As a
- I want
- So that

An example of such a user story would be: – **As a** retail customer, **I want** to return an electronically purchase merchandise within 14 days, **so that** the refund will be processed

BDD Tools

Cucumber

Cucumber is a test framework that supports BDD. In Cucumber, the BDD specifications are written in plain, simple English which is defined by the Gherkin language. In other words, Gherkin is a language that Cucumber understands. Gherkin presents the behavior of the application used, from which Cucumber can generate the acceptance test cases. Cucumber is a framework developed by Ruby that can work across different technologies.

Specflow

Specflow evolved from the Cucumber framework using Ruby on Rails, and is used mainly for .Net projects. SpecFlow also uses the Gherkin language.

Some benefits to using BDD

If you plan to implement BDD, here are a few points that will benefit the software team.

1. You are no longer defining 'test', but are defining 'behavior'.
2. Better communication between developers, testers and product owners.
3. Because BDD is explained using simple language, the learning curve will be much shorter.
4. Being non-technical in nature, it can reach a wider audience.
5. The behavioral approach defines acceptance criteria prior to development.

Disadvantages of BDD

Even the best development approaches can have pitfalls and BDD is no exception. Some of them are:

1. To work in BDD, prior experience of [TDD](#) is required.
2. BDD is incompatible with the waterfall approach.
3. If the requirements are not properly specified, BDD may not be effective.
4. Testers using BDD need to have sufficient technical skills.

Summary

BDD lets us develop, test and think about the code from the view of the business owner. You need to have the mindset to implement 'real time examples' rather than implementing just 'functionalities' So try it out!



Author:

Jithin Nair

Jithin is an experienced Lead QA Engineer with a global product development a...

[Read bio in full](#)

Similar articles:

Published on: September 19, 2018

What is Unit Testing? An Introduction to Unit Testing

[Read more](#)

Published on: April 18, 2018

TDD vs BDD – What's the Difference Between TDD and BDD?

[Read more](#)

Published on: February 13, 2018

What is TDD? – An Introduction to Test Driven Development

[Read more](#)

30-day free trial - no credit card required

Pay monthly, no long-term contracts & cancel at any time.

[See Pricing & Sign Up](#)

Follow us

[Feature Tour](#)

[Our Customers](#)

[Pricing & Sign Up](#)

[API](#)

[Blog](#)

[Contact Us & Support](#)

© Copyright TestLodge - Test management software. All rights reserved. Terms of Service. DPA.
Privacy Policy.