**Difference between BDD vs TDD :**

|  |  |  |
| --- | --- | --- |
| S.NO. | Behavior Driven Development | Test Driven Development |
| 01. | Behavior Driven Development is a development technique which focuses more on a software application’s behavior. | Test Driven Development is a development technique which focuses more on the implementation of a feature of a software application/product. |
| 02. | In BDD the participants are Developers, Customer, QAs. | In TDD the participants are developers. |
| 03. | Mainly it creates an executable specification that fails because the respective feature doesn’t exist, then writing the simplest code that can make the specification pass and as a result we get the required behavior implemented in the system. | Mainly it refers to write a test case that fails because the specified functionality doesn’t exist and after that update the code that can make the test case pass and as a result we get the feature implemented in the system. |
| 04. | Its main focus is on system requirements. | Its main focus is on unit test. |
| 05. | In BDD the starting point is a scenario. | In TDD the starting point is a test case. |
| 06. | It is a team methodology. | It is a development practice. |
| 07. | Here language used to write behavior/scenarios is simple English language. | Here language is used is similar to the one used for feature development like programming language. |
| 08. | In BDD collaboration is required between all the stakeholders. | In TDD collaboration is required only between the developers. |
| 09. | It is a good approach for project development which are driven by user actions. | It is a good approach for projects which involve API and third-party tools. |
| 10. | Some of the tools used are  Cucumber, Dave, JBehave, Spec Flow,  Concordian, BeanSpec etc. | Some of the tools used are  JBehave, JDave, Cucumber, Spec Flow, BeanSpec, FitNesse etc. |

Test-driven development is the act of first deciding what you want your program to do (the specifications), formulating a failing test, then writing the code to make that test pass. It is most often associated with automated testing. Although you could apply the principals to manual testing as well.

What is Test-Driven Development (TDD)?

[Test-Driven Development](https://www.browserstack.com/guide/what-is-test-driven-development) is a testing methodology or a programming practice implemented from a developer’s perspective. In this technique, a QA engineer starts designing and writing test cases for every small functionality of an application.

### What is Behavioral-Driven Development (BDD)?

[Behavioral-Driven Development](https://www.browserstack.com/guide/advanced-bdd-test-automation) (BDD) is a testing approach derived from the Test-Driven Development (TDD) methodology. In BDD, tests are mainly based on systems behavior. This approach defines various ways to develop a feature based on its behavior.

**Benefits of TDD**

* Unit test proves that the code actually works
* Can drive the design of the program
* Refactoring allows improving the design of the code
* Low-Level regression test suite
* Test first reduce the cost of the bug
* Reduces the amount of time required for rework
* Explores bugs or errors very quickly
* Faster feedback
* Encourages the development of cleaner and better designs
* Enhances the productivity of the programmer
* Allows any team member to start working on the code without a specific team member. This encourages knowledge-sharing and collaboration.
* It gives the programmer confidence to change an application’s large architecture quickly.
* Results in the creation of extensive code that is flexible and easy to maintain

**Drawbacks of TDD**

♣ Developer can consider it as a waste of time

♣ The test can be targeted on verification of classes and methods and not on what the code really should do

♣ Test become part of the maintenance overhead of a project

♣ Rewrite the test when requirements change