**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. |