**BDD-Behaviour driven development:**

Behaviour-driven development (BDD) is **an Agile software development methodology in which an application is documented and designed around the behaviour a user expects to experience when interacting with it**.

EX:

**The following is a really simple example of how to write a BDD feature for a login button**. A user should be able to login by entering their credentials and clicking on a button. their homepage. As you can see, there are several keywords here: Feature, Scenario, Given, When, Then, And.

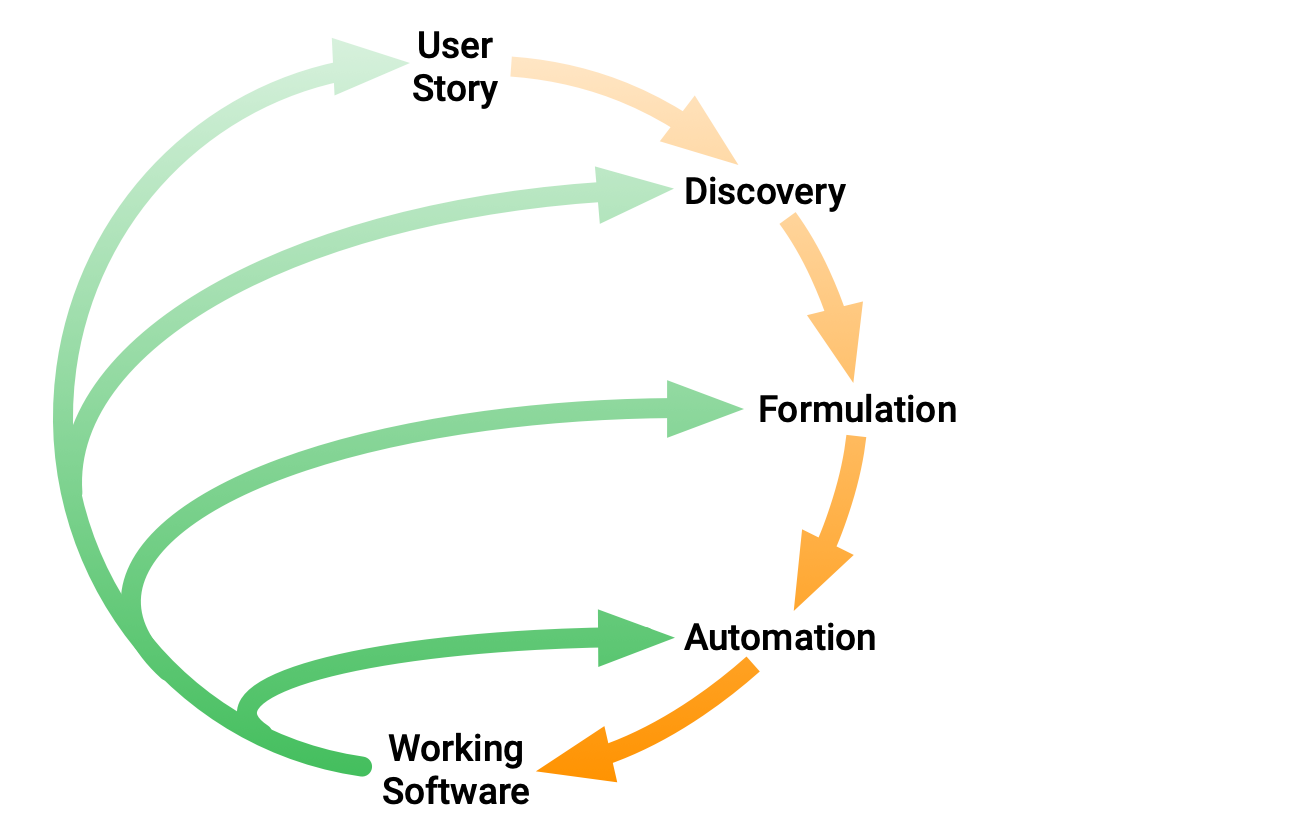
**BDD STEPS:**

Essentially, day-to-day BDD activity is a three-step, iterative process:

1. First, take a small upcoming change to the system – a [User Story](https://cucumber.io/docs/terms/user-story/) – and talk about concrete examples of the new functionality to explore, discover and agree on the details of what’s expected to be done.
2. Next, document those examples in a way that can be automated, and check for agreement.
3. Finally, implement the behaviour described by each documented example, starting with an automated test to guide the development of the code.

The idea is to make each change small and iterate rapidly, moving back up a level each time you need more information. Each time you automate and implement a new example, you’ve added something valuable to your system, and you’re ready to respond to feedback.

We call these practices *Discovery*, *Formulation*, and *Automation*.

Discovery, Formulation and Automation

Over time, the documented examples become an asset that enables your team to continue confidently and rapidly making changes to the system. The code reflects the documentation, and the documentation reflects the team’s shared understanding of the problem domain. This shared understanding is constantly evolving.