

A Cucumber is a software tool based on **Behavior Driven Development** (BDD) framework which is used to write **acceptance tests** for the web application. The tests are written in easily readable and understandable format for Business Analysts, Developers and Testers.

It is like reading plain English texts!

Syntax

Cucumber tests are divided into individual **Features**. These Features are subdivided into **Scenarios**, which are sequences of **Steps**:

- **Given** - Describes the initial state before the start of a test.
- **When** - Describes user actions during a test.
- **Then** - Describes results from **When** actions.

Occasionally, you might want to write a combination of Given-When-Then. You can use the following **conjunctions keywords**:

- **And** - Logical AND
- **But** - Negative form of AND

Before we start, you might want to consider using [VCR and Webmock](#) for Cucumber testing. Once we are ready, let's try implementing the tests!

Action steps

First, we will start with the most common actions users do every day when using a web application.

Sign in

Sign in is an action everyone does multiple times per day. To sign in a user in our application we will create a **new user** and use **login_as function** provided by [warden test helpers](#) (assuming we are using [devise](#) for authentication).

```
# And I am signed in
And(/^I am signed in$/) do
  @user = create(:user, email: 'user@example.com', password:
'password')
  login_as(@user, scope: :user)
end
```

Clicking on an element

Every so often we **click on an element** in our browser. All this clicking has to be covered with tests, you can describe them in many different ways. Here are some examples: