



## Cucumber

### Components

- Feature files
- Step definitions
- Support files

### Directory Structure

The folder structure follows the same format:

```
| features
| - step_definitions
| - todo_item_steps.rb
| - support
| - factory_girl.rb
|- user_creates_todo.feature
|- user_completes_todo.feature
```

## Features

### Components

- Feature name and description
- Background (optional)
- Scenarios

### Format

Features follow a simple format:

```
Feature: [Topic]
  In order to [goal]
  As a [user role]
  I should be able to [action]
```

## Scenarios

### Components

- Given (places system in a specific state)
- When (describes the action being taken)
- Then (performs assertions against the outcome)

```
Scenario: Complete a todo
  Given I have signed in as "john@example.com"
  And I have a todo "Pick Up the Milk" due today
  When I complete the todo "Pick Up the Milk"
  Then all my todos should be complete
```

## Steps

### Components

A step consists of the step type (Given, When, or Then), a string or regular expression, and a block of Ruby code to be run for that step.

```
Given /^I have signed in as "([^"]*)"$/ do |email|
  visit sign_in_path
  fill_in "Email", with: email
  click_button "Sign In"
end

When "I create a todo without a title" do
  visit new_todo_path
  click_button "Create Todo"
end

Then %I should see an error do
  page.should have_css("form.errors", text: "Title must be present")
end
```

## Support

### Components

Support files provide developers a way to extend Cucumber's functionality by providing an area to require third-party libraries and add methods available to each step definition.

```
# features/support/factory_girl.rb
require "factory_girl/step_definitions"

# features/support/global_user.rb
module GlobalUser
  def current_user=(new_user)
    @current_user = new_user
  end

  def current_user
    @current_user
  end
end

World(GlobalUser)

# features/step_definitions/user_steps.rb
Given %I am operating as "$email" do |email|
  self.current_user = User.find_by_email!(email)
end
```