

# Using charters in Exploratory Testing



## Warning: Content Compatibility Notice

Please note that the articles in this documentation feature media from app version 2.0.5. Interface elements may differ in newer versions. Refer to the latest app version for the most accurate information.

- [What is a charter, and why is it relevant?](#)
- [The structure of a charter](#)
  - [Examples](#)
- [Tips](#)
  - [Start broad and then go to specifics](#)
  - [Use and stay focused on the charter](#)
  - [Enumerate items you know in advance that you want to cover](#)
  - [Don't detail too much](#)
  - [Include quality attributes you aim to test](#)
  - [Discuss the charters with the team](#)
  - [Reuse the charter](#)
- [References](#)

## What is a charter, and why is it relevant?

In an Exploratory Testing approach, we don't follow predefined steps or check for well-known expectations. We also don't test in a random way.

While performing Exploratory Testing, we use our experience and knowledge to uncover new information. Exploring can be done at any moment, at any time, even without a specific, explicit reason whatsoever.

Usually, however, exploratory testing starts due to a reason and around some feature or area.

To focus our exploration activities, we use a charter. By having a charter, we can focus our attention on a certain question or concern we want to address. This way, we can explore in more depth and find information that otherwise would be hard to depict.

Before coming up with a charter, reflect a bit about the following:

- What is my main goal for this testing session?
- Do I want to deepen knowledge about a specific "thing" (e.g., feature, flow, behavior), or do I have to want a more high-level overview?
- Will I be using specific resources, tools, or heuristics?
- Will I be focused on certain quality criteria attributes? (e.g., correctness, performance, accessibility)
- Do I want to focus my attention more on positive test scenarios or negative test scenarios?

In sum, *Why am I going to do this session? What "exactly" will I be looking at and using what?*

## The structure of a charter

There is a common format that works for most teams.



### Charter template

Explore <area, feature, risk>

with/using <resources, restrictions, heuristics, dependencies, tools>

to discover <information>

*Adapted from Maaret Pyhäjärvi, Elisabeth Hendrickson*

However, the charter doesn't have a strict structure, and we can tailor it to our needs.

## Examples

Next, there are some examples of possible charters.

In this case, we have a high-level charter, to have an overview of the system.



#### Charter

*Explore the system*

*With typical usage scenarios*

*To discover how it works.*

We can also focus on a specific feature, or page, and around a specific concern/quality criteria (e.g., usability).



#### Charter

*Explore the login page*

*Using Chrome and different screen sizes*

*To discover problems with usability*

We can still be focused on a specific feature or page, but aim to test how it works for different contexts/environments (e.g., browsers).



#### Charter

*Explore the login page*

*Using Chrome, Firefox, Edge, and Opera*

*To discover problems with CSS styling and JavaScript*

Eventually, we may even have a more focused charter, for example, to depict problems around some specific processing.



#### Charter

Explore the REST API endpoint for importing test results in JUnit format

With different test reports

To discover problems in their processing and mapping to internal entities.

Our exploration can be focused on a certain feature/area (e.g., REST API endpoint), and we can explicitly enumerate a subset of items we wish to validate (e.g., operations).



#### Charter

Explore the REST API endpoint for managing bookings and the CRUD operations

- POST
- GET
- PUT
- DELETE

Using the "curl" command line tool and the REST API specification

To discover problems with the syntax

Our exploration can be made around specific personas relevant to the context of our product.



#### Charter

Explore the bookstore site

With a mid-age, non-technical teacher persona perspective

To discover problems around the easiness of the whole process of buying books

Sometimes we're concerned about product consistency (e.g., in terms of layout, terminology, options, and operations); we can have a charter for that.



#### Charter

Explore the Traffic Report

Using Chrome and different data sizes

To discover consistency problems with other reports

Charters can target specific testing types, such as security testing. We may want to focus our attention and concerns on specific features and ways of "attacking" them.



#### Charter

Explore the shopping cart

Using Chrome and RestMan extension to perform API calls

To discover if the shopping cart and its related properties can tamper somehow



#### Charter

Explore the product website

Using Chrome to craft and make API calls

To discover problems related to authorization

## Tips

### Start broad and then go to specifics

Start with more generic charters and common usage scenarios to have a more general idea about the system under test.

### Use and stay focused on the charter

While testing, we should stick with the charter; if we depict something a bit unrelated, we can take notes and then decide afterward if we want to create a charter around that.

### Enumerate items you know in advance that you want to cover

You may add a list of bullet points to your charter if you want to enumerate certain aspects you aim to test or subsets for your charter (e.g., operations, and user profiles).

### Don't detail too much

Avoid having long charters as, eventually, they will tend to become closer to test scripts and limit your exploration.

### Include quality attributes you aim to test

Usually, it is a good idea to know in advance the quality attributes/aspects you will be focused on, so you can target your exploration around those.

## **Discuss the charters with the team**

Provide visibility of what you aim to test. The team can give you feedback about areas or aspects they may consider relevant, among others.

## **Reuse the charter**

Charters can be reused; the findings of sessions with the same charter can and will be mostly different each time a person uses them during a testing session. A charter can be reused by another colleague, a tester, or even a developer.

## **References**

- [Exploratory Testing in an Agile Context by Elisabeth Hendrickson](#)