icco INTERNATIONAL CORPORATE CONSULTING OLOFSSON

GUIDE · NOVEMBER 2026

Example Mapping: slice requirements into bite-sized pieces in 25 minutes

How business, development and test clarify every story together. With process, the tasks of each role, example, the way to the BDD story and a printable card template.

ICCO LTD · Quality management for complex IT transformations
www.icco-online.com · info@icco-online.com

THE PRINCIPLE

Four card colours, one map

Example Mapping is a short, structured session in which business, development and test clarify a story together. The result is a map of cards in four colours.

Card	Meaning	Note
Yellow: story	The user story under discussion	Exactly one per session, at the top of the map
Blue: rule	An acceptance criterion of the story	Place under the story, the what and not the how
Green: example	A concrete case for a rule	Positive case, negative case and edge cases, placed under the rule
Red: question	Something open that cannot be clarified right away	Place at the edge instead of discussing at length

Why it pays off

- **Slice right:** stories of a handy size, with a clear boundary.
- **Estimate reliably:** small, clarified stories are easy to estimate.
- **Develop with focus:** bite-sized requirements, no assumptions or follow-up questions.
- **Describe efficiently:** examples turn directly into test cases.
- **Cover completely:** every rule has its examples, gaps show up early.

> **INVEST checks, Example Mapping clarifies.** INVEST is the checklist for a well-sliced story: independent, negotiable, valuable, estimable, small, testable. Example Mapping is the method with which the team clarifies the story and meets these criteria. Many blue cards show that the story is not small enough. Open red cards show that it cannot be estimated yet. Green examples make it testable.

THE PROCESS

The session in five steps

One story, 25 minutes at most, five people at most. The times are guide values.

2 min	**Read the story** One person reads the story aloud. The yellow card goes at the top of the map.
8 min	**Collect rules** Which rules apply? Each rule goes on a blue card under the story.
10 min	**Find examples** Concrete cases for each rule, including negative and edge cases, on green cards.
ongoing	**Note questions** Anything unclear goes on a red card at the edge.
3 min	**Vote** Everyone shows a thumb: is the story ready for implementation?

Rules of the session

- Business unit or product owner, development and test take part. Others only with relevant knowledge, five people at most.
- There are no stupid questions.
- No Gherkin yet: the formalism would distract from the content.
- If the discussion goes in circles, information is missing. Note questions instead.

> **If time runs out.** Possible reasons: too little practice with the method, the story has too many rules or too many questions. A postponed session is not a failure, it makes the unknown visible before it costs money.

WHO DOES WHAT

Three roles, one task

All three clarify the same story. Each role brings something the others lack. The next three pages list what to do for each role. Print, put next to you, tick off.

	Business	Development	Test
You bring	Knowledge of what is needed and why	Knowledge of how it can be built	An eye for everything that can go wrong
Before	Pick the story, collect everyday examples	Read the story, anticipate dependencies	Read the story, gauge testability
In the session	Explain the value, name rules and real cases	Ask questions, name effort and risks	Look for error and edge cases, check coverage
After	Clarify open questions, read and approve the BDD story	Write the BDD story with test, build	Write the BDD story with development, create test cases

> **Who leads the session?** One person moderates: reads the story aloud, watches the time, keeps the map tidy and asks for the thumbs at the end. Any role can do it, in turns, or at first someone from outside.

For all three

- If you do not understand, ask.
- If you do not know, write a red card.
- Nobody guesses or simply assumes.

YOUR ROLE

Business unit or product owner

You know what is needed and why.

You explain it until everyone understands, and you decide what belongs to the story and what does not.

BEFORE

- [ ] Pick a story whose value is clear
- [ ] Have real everyday examples ready, including exceptions
- [ ] Bring documents: forms, rule books, e-mails

IN THE SESSION

- [ ] Read the story aloud and explain what it is needed for
- [ ] Name business rules, each on a blue card
- [ ] Name concrete cases, including awkward ones, on green cards
- [ ] Answer the questions you can answer
- [ ] Do not know? Red card, do not guess
- [ ] Decide what does not belong
- [ ] Show your thumb at the end, honestly

AFTER

- [ ] Clarify open red questions or name someone, with a date
- [ ] Read the BDD story: is the business language right? Is a case missing?
- [ ] Approve the story

These questions help you

- Which problem does it solve, and for whom?
- What happens in the normal case, and when not?
- How will I know it is done?

Not your job

- Dictating the technical solution (“put a button here”)
- Writing Gherkin
- Hiding exceptions because they are annoying

YOUR ROLE

Development

You know how things can be built.

You say early where it gets expensive, risky or unclear, so nobody has to guess later.

BEFORE

- [ ] Read the story
- [ ] Think briefly: which systems and interfaces are affected?

IN THE SESSION

- [ ] Ask until every rule is unambiguous
- [ ] Look for edge cases: “What happens if …?”
- [ ] Name effort and dependencies
- [ ] A rule with a lot of effort: suggest splitting it into its own story
- [ ] Capture every assumption as a red card
- [ ] Do not discuss the solution yet, that comes later
- [ ] Show your thumb at the end, honestly

AFTER

- [ ] Write the BDD story together with test
- [ ] Estimate the story and start implementing
- [ ] Automate scenarios where it pays off and add them to the pipeline

These questions help you

- What am I still missing to build without guessing?
- What does this depend on, and what depends on it?
- Does it fit into a handy size?

Not your job

- Setting business rules yourself
- Filling gaps with your own assumptions
- Turning the session into a technical discussion

YOUR ROLE

Test

You think of everything that can go wrong.

You make sure every rule is checkable and backed by examples.

BEFORE

- [ ] Read the story
- [ ] Gauge: can this be checked unambiguously?

IN THE SESSION

- [ ] Ask about negative and edge cases: empty, too large, wrong order, no permission
- [ ] Check: does every blue rule have at least one green example?
- [ ] Ask for concrete values instead of “a valid value”
- [ ] Ask: “How do we know it is met?”
- [ ] Capture anything unclear as a red card
- [ ] Show your thumb at the end, honestly

AFTER

- [ ] Write the BDD story together with development: one scenario per example
- [ ] Create test cases in Xray and link them to the story
- [ ] Clarify test data and test environment
- [ ] Before UAT check: did all BDD tests pass?

These questions help you

- How do I check this?
- Where is the limit, and what happens just before and after it?
- What happens on an error?

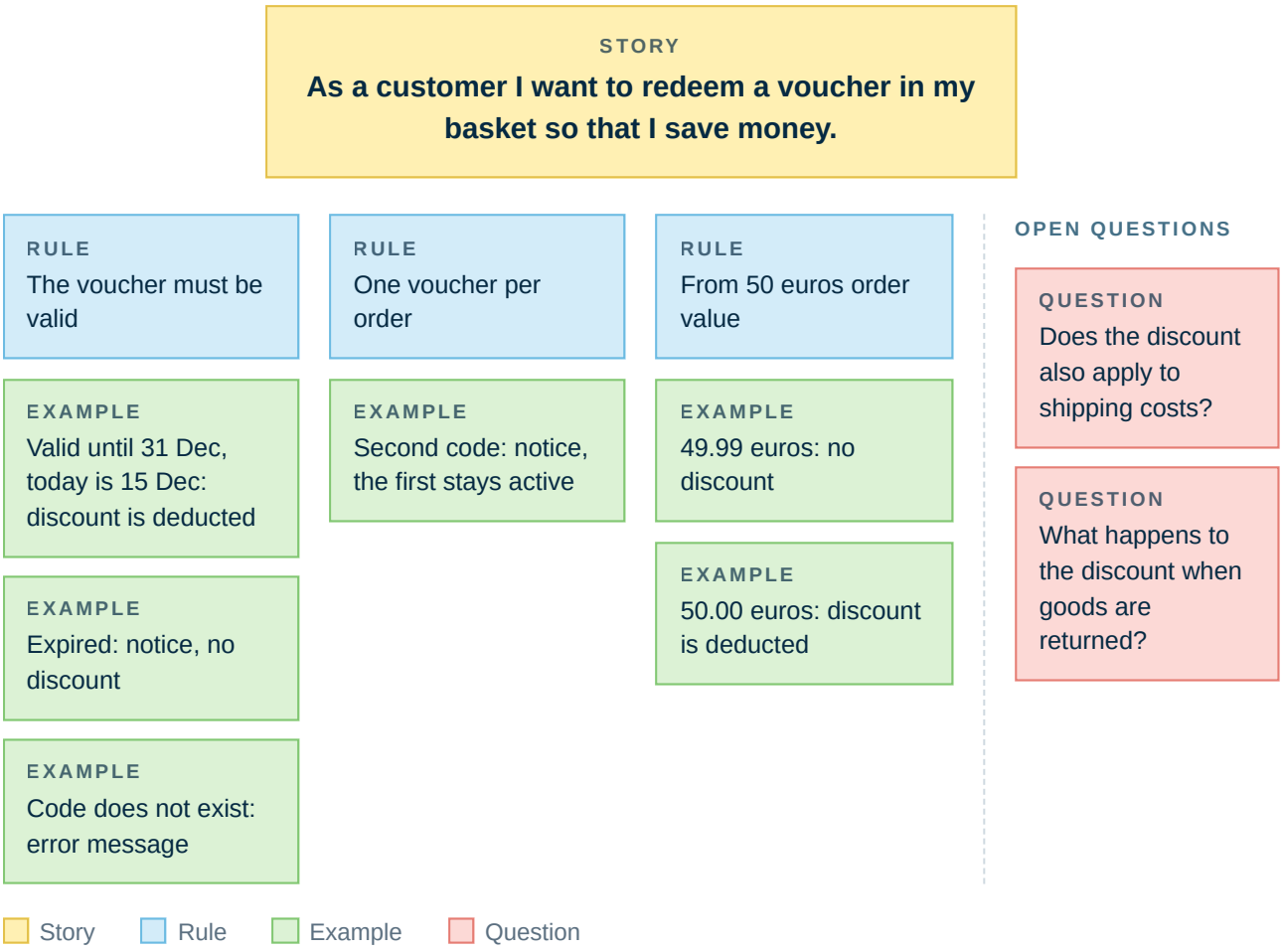
Not your job

- Starting to think only after implementation
- Dictating business rules or solutions
- Checking what nobody described

EXAMPLE

A map for the voucher

The example is fictitious and shows an online order.



Three possible outcomes

Outcome	How to recognise it	What happens next
Ready	Few rules, no open questions	The story goes into the sprint backlog.
Split	Many rules	Bundle rules into new stories, each estimable and testable.
Clarify	Many open questions	Answer the questions, then hold a second session.

AFTER THE SESSION

The map becomes a BDD story

A developer and a tester write the BDD story from the map, and the business unit gives feedback. Every example becomes a scenario, and the scenarios are at the same time the test cases.

- **Yellow story** becomes the feature: title and the sentence “As a … I want …”.
- **Blue rule** becomes a rule, under which the scenarios sit.
- **Green examples** become scenarios. Examples with different values become rows of an examples table.
- **Red question** stays open. Clarify first, then write the scenario.

```
Feature: Redeem voucher
  As a customer I want to redeem a voucher in my basket
  so that I save money.

  Rule: The voucher must be valid

    Scenario: Expired voucher
      Given the voucher has expired
      When I redeem it in the basket
      Then I see a notice
      And no discount is deducted

  Rule: From 50 euros order value

    Scenario Outline: Minimum order value
      Given the order value is <value> euros
      When I redeem a valid voucher
      Then the result is "<result>"

      Examples:
        | value | result               |
        | 49.99 | no discount          |
        | 50.00 | discount is deducted |

  # Open from the map: Does the discount also apply to shipping costs?
```

- **Given, When, Then.** Context, action, outcome. A scenario without action and check is not a test.
- **One behaviour per scenario.** A meaningful title, one behaviour. Readable and checkable on its own.
- **The language of the business.** Describe behaviour, not clicks and field names.

CARD TEMPLATE

To print and cut out

Print on coloured paper or use as a model for sticky notes. Page 11 has the green example cards.

STORY (YELLOW)

STORY (YELLOW)

STORY (YELLOW)

RULE (BLUE)

RULE (BLUE)

RULE (BLUE)

RULE (BLUE)

RULE (BLUE)

RULE (BLUE)

RULE (BLUE)

QUESTION (RED)

QUESTION (RED)

QUESTION (RED)

QUESTION (RED)

QUESTION (RED)

QUESTION (RED)

QUESTION (RED)

CARD TEMPLATE

Example cards (green)

EXAMPLE	EXAMPLE	EXAMPLE
EXAMPLE	EXAMPLE	EXAMPLE
EXAMPLE	EXAMPLE	EXAMPLE
EXAMPLE	EXAMPLE	EXAMPLE
EXAMPLE	EXAMPLE	EXAMPLE
EXAMPLE	EXAMPLE	EXAMPLE

TO TAKE AWAY

Checklist and next steps

Before the session

- Pick a story that is already roughly sliced
- Invite business, development and test, five people at most
- Prepare cards in four colours and pens, or set up a digital whiteboard
- Fix the time: 25 minutes at most

After the session

- Photograph or save the result
- Story ready: write the BDD story, developer and tester together
- Story too big: split it by rules into new stories
- Assign open questions with an owner and a date

Source of the method: Matt Wynne, Cucumber community, “Introducing Example Mapping” (2015), cucumber.io/blog/example-mapping-introduction. All product and brand names mentioned belong to their respective owners. The example is fictitious. As of November 2026.

Let's start with one story

We moderate the first sessions in your team, coach moderators and review your backlog for stories that are too big or unclear.

ICCO LTD · info@icco-online.com · www.icco-online.com
Peiraios No. 16, Office 201, 3016 Limassol, Cyprus