WHAT IS A USE CASE AND HOW DOES IT HELP YOU?

Defining use cases helps identify the features your solution must include and explains how the system should react to the different user’s actions.

It helps you also to think about what could go wrong, and how to mitigate issues (the so called failure path) as well as having alternative paths where different options in the user’s interaction could take place.

LET’S FOCUS ON CREATING A USE CASE:

- Based on your research, on the personas you defined and on the storyboard you created, identify the high priority use case you’re going to create.

- Use a sequence of action steps.

- Show the process moving forward.

- Write each action step in a way to show a simple active action.

- Two actions steps refer to an interaction between a user and a system (e.g., customer enters an address).

Here’s our template of a use case diagram (available in the Toolkit):
Let’s use the following storyboard as a basis. We will focus on one specific task (circled) for our first use case that will be describing the creation of a leave request.

1. Start by filling the main properties of the use case diagram:
   (*see related publication Matz and Germanakos, 2016)

   - **Use Case ID**: The unique reference number of this use case. This can help if you have several use cases and thus want to document the organization of these.

   - **Use Case Name**: Use concrete, short and meaningful title to the use case to avoid ambiguity. Go for an active wording (“Capture expense item details” rather than “Expense Item Detail Capturing”).
- **Primary Role**: Describe the persona who is using the system to fulfill a goal, and his/her role.

- **Secondary Role (optional)**: The persona who receives information from the system, but is not the primary user.

- **Use Case Goal**: Describe the goal the persona wants to achieve: use the Point of Views (User + need + insight) you created for the persona.

- **Background**: Create a short description and assumptions about the scenario.

- **Pre-conditions**: Describes what the system will guarantee is true before the use case starts (it could be the previous use case).

- **Trigger**: Describes what event or action that gets the use case started.

<table>
<thead>
<tr>
<th>Use Case ID:</th>
<th>123</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Case Name:</td>
<td>Create leave request</td>
</tr>
<tr>
<td>Primary Role:</td>
<td>Employee</td>
</tr>
<tr>
<td>Secondary Role (optional):</td>
<td>Frank needs a way to create a leave request in order to get approval that he can leave for his vacation in Hawaii. That makes him feel happy that he will be then able to start booking the trip. Frank had a very difficult year in the office with many meetings and critical deliveries he had to submit. He doesn't want to spend too much time in this activity. He wants to submit the request with the less possible interaction steps. Successfully authenticated to use the system. User knows about dates he wants to be on leave. He wants to go on vacation.</td>
</tr>
<tr>
<td>Use Case Goal (POV):</td>
<td></td>
</tr>
<tr>
<td>Background:</td>
<td></td>
</tr>
<tr>
<td>Pre-conditions:</td>
<td></td>
</tr>
<tr>
<td>Trigger:</td>
<td></td>
</tr>
</tbody>
</table>

2. **Add the information coming from your persona:**

   Picture and name of the persona, name of the application, and images to create empathy
Then write the sequence of steps:

- Always start with the big picture first! Do NOT write every detail at the first sitting, write down just an outline to start with. **Precision** should be low at the beginning, then greater closer to final construction.

- First describe the **Main Success Case**: The main success is case is the straight forward case (think of the 80/20 rule), so detail all the simple steps to achieve the user goal previously defined, and list the name of the interaction data needed to complete the different steps. The interaction data does not need to be in a technical format yet (data-structure binding).

- Capture **alternative success paths** and also **possible ways to fail** (internal failure of the system, wrong behavior of the user...).

---

**TIPS ON CREATING USE CASES:**

- **Work in teams and start writing on a whiteboard or flipchart, ideally use post-it so you can shuffle around the elements of the use case while discussing.**

- **Don’t be a perfectionist** (it is not necessary to make it perfect the first time).

- Developing use cases is an iterative process (work and refine your use cases), so don’t hesitate to iterate and refine your use cases as you know more about the different goals.

- **Use full sentences** (a subject and a predicate) in your action steps.

- Write in present tense.

- Write technology neutral.

- Show user’s intention (not UI details).
• Write technology neutral.

• Show user’s intention (not UI details).

• Show clearly “who has the ball.”

• Use verbs like “ensure, validate, verify” or similar, instead of “check whether.”

• Strive for 3 to 10 steps.
  • If your use case has more than 10 steps, take a step back and check whether there might be a sub-use case.

• Use the format you are most comfortable with (text, visual, process, table, etc).

• Create flow diagrams to visualize the organization of the different use cases and their interactions. This will help you to structure your use cases and keep an understanding of the larger picture.

• Validate your use cases with end-users, and iterate.

*CITATIONS: