## **Use case template**

An adaptation of the standard Cockburn template will be used. The template and examples follow:

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | UC-2 The tester reports a bug | | |
| Primary actor | Tester | Secondary actors | Company Software |
| Description | The tester can register a bug by giving it a name and a description; immediately after the bug is registered, all programmers see the updated list of bugs with the newly added entry. | | |
| Trigger | The tester submits a bug report to the platform by filling it out and pressing "Send." | | |
| Preconditions | Pre-1. Tester is logged into the Company Software  Pre-2. There is a software to report a bug on. | | |
| Postconditions | Post-1. All programmers see the updated list of bugs with the newly added entry. | | |
| Normal flow | **2.0 Reporting a Bug**   1. The tester accesses the software code for testing. 2. They open the bug report submission window. 3. They specify the program and the possible cause of the bug. 4. They provide details and examples of the anomaly or malfunction they found. 5. They complete the report and submit it to the platform's bug list. | | |
| Alternative flows | **2.1 Multiple Bugs in One Program**   1. The tester identifies multiple bugs in the code. 2. They separate them in the bug report. 3. Return to step 4 of the normal flow. | | |
| Exceptions | **2.0 E1 The Bug(s) Was Already Reported**   1. The tester is informed that the bug is already in the list.   2a. They exit the form. 2b. They indicate that it is a different bug from the ones already listed. | | |
|  |  | | |

Descriptions of template fields:

* **ID and name:** Title should be descriptive and should usually begin with a verb, e.g. order, calculate, input, etc. ID can have any format but must be unique among all use cases.
* **Primary actor:** Person that wishes to accomplish a goal through the use of the system. Only a single primary actor per use case.
* **Secondary actors:** Actors that have an interest in the completion of the goal but that do not directly interact with the system.
* **Description:** Concise description of the purpose of the use case.
* **Trigger:** Condition internal or external to the system that prompts the use case to start.
* **Preconditions:** Conditions that must be true before the use case starts. Each should be labeled with an ID unique to the use case.
* **Postconditions:** Conditions that must be true after the use case ends normally. Each should be labeled with an ID unique to the use case.
* **Normal flow:** Detailed step-by-step description of the logical flow of the use case. It should describe an explicit two way interaction, with the system prompting for input and the actor responding accordingly. Each step should be numbered.
* **Alternative flows:** Flows that achieve the same goal as the normal flow but are expected to be less common or lower priority.
* **Exceptions:** Conditions that result in the normal flow ending prematurely due to an unrecoverable condition in the system. The condition that causes the flow should be clearly stated, as should be any other decisions that the actor must make in this situation.