## **Use case**

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

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.

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | Login | | |
| **Primary actor** | AngajatFirma | **Secondary actors** | System |
| **Description** | Un angajat vrea sa foloseasca aplicatia pentru a vedea/rezolva buguri | | |
| **Trigger** | Actorii trbuie sa porneasca aplicatia. | | |
| **Preconditions** | Angajatul trebuie sa aibe cont, sa fie inregistrat in sistem | | |
| **Postconditions** | Actorii primesc un mesaj care indica stadiul autentificarii(reusita/nereusita). | | |
| **Normal flow** | 1.Utilizatorul introduce usernameul in campul corespunzator  2.Utilizatorul introduce parola in campul “Password”  3.Dupa introducerea datelor userul apasa butonul de Login  4.Sistemul valizeaza datele introduse si afiseaza un mesaj in functie de starea validarii(reusit/esuat)  5.In caz de autentificare cu succes utilizatorului I se v adeschide meniul corespunzator accesului sau(tester/developer). In caz de esec se va relua autentificarea. | | |
| **Alternative flows** | **None** | | |
| **Exceptions** | **Cont inexistent** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **ID and name** | Inregistrare bug | | |
| **Primary actor** | Tester | **Secondary actors** | Developer |
| **Description** | Actorul principal introduce datele in campurile corespunzatoare care descriu proprietatile unui bug(descriere, nume, etc.) apoi inregistreaza acel bug pentru a fi vizualizat de toti programatorii | | |
| **Trigger** | Testerul apasa pe butonul de “Inregistrare” | | |
| **Preconditions** | Testerul este logat in aplicatie  Exista o baza de date in care bug-urile pot fi stocate | | |
| **Postconditions** | Bugul este inregistrat cu succes  Bugul este disponibil pentru vizualizare de catre programatori  Testerul primeste o notificare in legatura cu starea inregistraii(esuata/reusita) | | |
| **Normal flow** | 1.Testerul deschide aplicatia de inregistrare a bugurilor  2.Testerul completeaza informatiile necesare despre bug (denumire, decriere, etc.)  3.Testerul trimite bugul pentru inregisrare in sistem | | |
| **Alternative flows** | Daca campurile necesare nu sunt completate sistemul afiseaza un meaj de eroare si cere completarea lor | | |
| **Exceptions** | Testerul introduce date invalide care duc la respingerea inregistrarii | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | Eliminare bug | | |
| Primary actor | Developer | Secondary actors | Developer |
| Description | Procesul prin care un programator elimină un bug din sistem după ce a fost remediat. | | |
| Trigger | Programatorul apasa pe butonul de “Eliminare bug” | | |
| Preconditions | Programatorul trebuie sa fie logat in siste.  Bugul pe care dorete sa-l elimine exista in sistem. | | |
| Postconditions | Bugul este eliminat din sistem  Lista de buguri este actualizata pentru toti programatorii | | |
| Normal flow | 1.Programatorul deschide aplicatia de gestionare a bugurilor  2.Programatorul aceseaza lista de buguri disponibile pentru eliminare.  3.Programatorul selecteaza bugul pe care doreste sa-l elimine  4.Programatorul confirma eliminarea bugului.  5.Se actualizeaza lista de buguri pentru toti programatorii. | | |
| Alternative flows | Anularea Eliminarii: Programatorul poate renunta la eliminarea unui bug inainte de a confirma operatia. În acest caz, bug-ul nu este eliminat din sistem, iar procesul revine la pasul anterior. | | |
| Exceptions | Sistemul intampina o eroare interna in timpul procesului de eliminare a bugului ceea ce duce la esecul operatiei(ex. Nu este selectat un bug inainte de apasarea butonului de “Eliminare”) | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | Vizualizare lista buguri | | |
| Primary actor | Developer | Secondary actors | none |
| Description | Procesul prin care un programator accesează lista de bug-uri disponibile în sistem pentru a le vizualiza și a identifica bug-urile care necesită remediere. | | |
| Trigger | Programatorul se autentifica si intra in aplicatie | | |
| Preconditions | Exista cel putin un bug in sistem  Programatorul este autentificat | | |
| Postconditions | Programatorul poate vedea lista bugurilor disponibile in sistem  Programatorul poate selecta un bug pentru a vedea detalile sale | | |
| Normal flow | 1.Programatorul deschide aplicația de gestionare a bug-urilor.  2.Programatorul accesează opțiunea de vizualizare a listei de bug-uri.  3.Sistemul afișează lista de bug-uri disponibile, inclusiv detalii precum denumirea și starea lor (de exemplu, "în așteptare", "în lucru", "rezolvat").  4.Programatorul explorează lista de bug-uri pentru a identifica bug-urile relevante. | | |
| Alternative flows | Căutare sau Filtrare: Programatorul poate căuta sau filtra lista de bug-uri după criterii specifice, cum ar fi prioritățea sau statusul, pentru a reduce numărul de bug-uri afișate și a se concentra pe cele relevante. | | |
| Exceptions | none | | |