**Cazuri de utilizare pentru firma BugFixes**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID and name | UC-1: Autentificare in sistem | | | | |
| Primary actor | | Utilizator (Tester/Programator) | Secondary actors | - |
| Description | | Utilizatorul acceseaza aplicatia de gestiune a bug-urilor, urmand sa se autentifice in contul sau de programator sau tester. Se verifica username-ul si parola utilizatorului, iar daca sunt corecte acesta va fi redistribuit in interioul aplicatiei. | | |
| Trigger | | Utilizatorul doreste sa acceseze aplicatie de gestiune a bug-urilor | | |
| Preconditions | | Utilizatorul are un cont creat in cadrul firmei si a aplicatiei. | | |
| Postconditions | | Utilizatorul si-a accesat contul si are disponibilitate in cadrul functionalitatilor din cadrul aplicatiei. | | |
| Normal flow | | * Utilizatorul acceseaza aplicatia. * Se solicita credentialele de logare ale user-ului (username si parola) pentru a-i oferi disponibilitate la cont. * Utilizatorul introduce username-ul si parola contului sau. * Sistemul de gestiune a conturilor verifica corectitudinea datelor introduse. * Daca datele introduse sunt valide, sistemul de gestiune a conturilor redirectioneza utilizatorul catre pagina principala, permitandu-i accesul la functionalitatile acestuia. | | |
| Alternative flows | | Daca username-ul si/sau parola sunt introduse gresit, sistemul va afisa un mesaj de eroare si il va ruga pe utilizator sa introduca din nou credentialele. | | |
| Exceptions | | Daca datele de logare sunt gresite, sistemul afiseaza un mesaj de eroare specificand faptul ca username-ul si/sau parola sunt incorecte, transmitand utilizatorul inapoi la pagina de autentificare. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID and name | UC-2: Adaugare bug | | | |
| Primary actor | | Tester | Secondary actors | Programator |
| Description | | Testerul poate sa inregistreze un bug in sistemul de urmarire a bug-urilor, atribuindu-i o denumire, o descriere si un grad de risc ( valori de la 1 la 10). Dupa inregistrearea bug-ului, toti programatorii vor vedea pe fereastra lor noul bug introdus in sistem. | | |
| Trigger | | Testerul incepe procesul de adaugare a unui bug in sistem. | | |
| Preconditions | | Testerul sa fie logat in contul sau de tester din cadrul firmei. | | |
| Postconditions | | * Bug-ul a fost inregistrat cu succes in sistem. * Lista de bug-uri atat a testerior cat si a programatorilor va fi actualizata cu noul bug introdus. | | |
| Normal flow | | * Testerul se logheaza in contul sau si selecteaza optiunea de adaugare nou bug. * Sistemul solicita completarea a 3 campuri: denumire, descriere si grad de risc(1-10). * Testerul introduce denumirea, descrierea si gradul de risc al bugului * Daca datele de intrare sunt valide, sistemul actualizeaza lista bugurilor, adaugand noul bug. | | |
| Alternative flows | | * Daca testerul introduce un nume, descriere sau grad de risc invalid, sistemul va afisa un mesaj de eroare si il va ruga pe utilizator sa reintroduca datele bug-ului. * Daca bug-ul este existent, sistemul va afisa un mesaj de eroare specificand ca testerul incearca sa introduce un bug deja existent in sistemul de gestiune a bug-urilor, rugandu-l sa reintroduca datele bug-ului. | | |
| Exceptions | | Daca sistemul va avea eroare la inregistrarea bug-ului in sistem, se va afisa un mesaj de eroare, iar bug-ul nu se salveaza. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID and name | UC-3: Adaugare bug pentru programator | | | |
| Primary actor | | Tester | Secondary actors | Programator |
| Description | | Testerul poate sa inregistreze un bug in sistemul de urmarire a bug-urilor, atribuindu-i o denumire, o descriere si un grad de risc ( valori de la 1 la 10), dar mai ales poate sa ofere un bug strict unui anumite programator din sistem, completand un camp special prin care specifica username-ul programatorului dorit.. Dupa inregistrearea bug-ului, toti programatorii vor vedea pe fereastra lor noul bug introdus in sistem, dar acesta va avea o informatie in plus, username-ul programatorului care sa resolve acel bug. | | |
| Trigger | | Testerul doreste sa adauge un bug in sistem special unui anumit programator. | | |
| Preconditions | | * Testerul sa fie logat in contul sau de tester din cadrul firmei. * Trebuie sa existe cel putin un programator in sistem, pentru a avea cui sa ii ofere testerul programatorului un bug. | | |
| Postconditions | | * Bug-ul a fost inregistrat cu succes in sistem, acesta avand disponibilitatea de a fi rezolvat doar de catre programatorul asignat. * Lista de bug-uri atat a testerior cat si a programatorilor va fi actualizata cu noul bug introdus. | | |
| Normal flow | | * Testerul se logheaza in contul sau si selecteaza optiunea de adaugare nou bug. * Sistemul solicita completarea a 4 campuri: denumire, descriere, grad de risc(1-10) si programator. * Testerul introduce denumirea, descrierea, gradul de risc al bugului si programatorul caruia doreste sa ii asigneze acest nou bug. * Daca datele de intrare sunt valide, sistemul actualizeaza lista bugurilor, adaugand noul bug. | | |
| Alternative flows | | * Daca testerul introduce un nume, descriere, grad de risc invalid sau/si un username de programator inexistent, sistemul va afisa un mesaj de eroare si il va ruga pe utilizator sa reintroduca datele bug-ului. * Daca se incearca asignarea unui bug nou unui programator inexistent, se afiseaza un mesaj de eroare prin care se specifica ca programatorul caruia dorim sa ii asignam bugul nu este existent. * Daca bug-ul este existent, sistemul va afisa un mesaj de eroare specificand ca testerul incearca sa introduce un bug deja existent in sistemul de gestiune a bug-urilor, rugandu-l sa reintroduca datele bug-ului. | | |
| Exceptions | | * Daca sistemul va avea eroare la inregistrarea bug-ului in sistem, se va afisa un mesaj de eroare, iar bug-ul nu se salveaza. * Daca nu exista programatorul dorit, se afiseaza un mesaj de eroare prin care se specifica ca programatorul caruia dorim sa ii asignam bugul nu este existent. | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID and name | UC-4: Vizualizare bug-uri tester | | | |
| Primary actor | | Tester | Secondary actors | - |
| Description | | Testerul acceseaza sistemul de gestiune a bug-urilor, sistemul oferindu-i in urma autentificarii cu succes, lista de buguri existenta in baza de date, in care se observa denumirea, descrierea si gradul de risc a fiecarui bug. | | |
| Trigger | | Testerul doreste sa vizualizeze lista de bug-uri din cadrul sistemului. | | |
| Preconditions | | Testerul are acces la sistemul de gestiune a bug-urilor. | | |
| Postconditions | | Testerul vizualizeaza bug-urile din sistem. | | |
| Normal flow | | * Testerul se logheaza in contul sau din cadrul sistemului. * Sistemul incarca lista de buguri, ordonate descrescator in functie de gradul de risc al fiecarui bug. * Testerul vizualizeaza lista de bug-uri. | | |
| Alternative flows | | None | | |
| Exceptions | | None | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID and name | UC-5: Vizualizare bug-uri programator | | | |
| Primary actor | | Programator | Secondary actors | - |
| Description | | Programatorul acceseaza sistemul de gestiune a bug-urilor, sistemul oferindu-i in urma autentificarii cu succes, lista de buguri existenta in baza de date, in care se observa denumirea, descrierea si gradul de risc a fiecarui bug. | | |
| Trigger | | Programatorul doreste sa vizualizeze lista de bug-uri din cadrul sistemului. | | |
| Preconditions | | Programatorul are acces la sistemul de gestiune a bug-urilor. | | |
| Postconditions | | Programatorul vizualizeaza bug-urile din sistem. | | |
| Normal flow | | * Programatorul se logheaza in contul sau din cadrul sistemului. * Sistemul incarca lista de buguri, ordonate descrescator in functie de gradul de risc al fiecarui bug. * Programatorul vizualizeaza lista de bug-uri. | | |
| Alternative flows | | None | | |
| Exceptions | | None | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID and name | UC-6: Filtrare bug-uri dupa grad de risc | | | |
| Primary actor | | Programator | Secondary actors | - |
| Description | | Programatorul acceseaza sistemul de gestiune a bug-urilor, sistemul oferindu-i in urma autentificarii cu succes, lista de buguri existenta in baza de date. Astfel el poate sa filtreze aceasta lista cu toate bug-urile din sistem in fucntie de anumite criterii, aceasta filtrare fiind intuitive si usor de utilizat. | | |
| Trigger | | Programatorul doreste sa vizualizeze lista de bug-uri care contine doar bug-uri cu un anumite grad de risc din cadrul sistemului. | | |
| Preconditions | | * Programatorul are acces la sistemul de gestiune a bug-urilor. * Exista cel putin un bug in sistem pentru a reusi filtrarea. | | |
| Postconditions | | * Programatorul are afisata lista de bug-uri care au gradul de risc droit de acesta * Programatorul poate sa modifice gradul de risc al acestor bug-uri afisate, sau sa modifice statusul bugului daca doreste | | |
| Normal flow | | * Programatorul se logheaza in contul sau din cadrul sistemului. * Sistemul incarca lista de buguri, ordonate descrescator in functie de gradul de risc al fiecarui bug. * Programatorul vizualizeaza lista de bug-uri. * Programatorul scrie intr-o casuta speciala gradul de risc a bug-urilor pe care doreste sa le vada, cu scopul de exemplu de a modifica gradul de risc daca e ridicat * Prin completarea casutei de grad de risc, lista se actualizeaza, afisandu-se doar bug-urile cu gradul de risc introddus. | | |
| Alternative flows | | Daca nu exista bug-uri cu gradul de risc introdus de catre programator, se afiseaza un mesaj de eroare ca nu exista bug-uri cu respectivul grad de risc. | | |
| Exceptions | | * Daca programatorul filtreaza in functie de un grad de risc inexistent, se afiseaza un mesaj de eroare ca nu exista bug-uri cu respectivul grad de risc. * Daca exista eroare la conectarea de baze de date, in care se aflat lista bug-urilor, se afiseaza un mesaj de eraore care sa ajute programatorul sa stie in ce punct al procesului a aparut o problema. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| ID and name | UC-7: Update status bug | | |
| Primary actor | Programator | Secondary actors | Tester |
| Description | Programatorul vizualizeaza lista bug-urilor si poate selecta un bug pentru a-i modifica statusul bug-ului, de exemplu din bug daunator in bug rezolvat. | | |
| Trigger | Programatorul doreste sa declare ca a rezolvat un bug, schimbandu-i starea acestuia. | | |
| Preconditions | 1. Programatorul are acces la sistemul de gestiune a bug-urilor. 2. Lista bug-urilor este afisata pe ecran. 3. Exista cel putin un bug caruia sa ii modificam statusul. | | |
| Postconditions | 1. Statusul bug-ul este modificat din lista tuturor utilizatorilor. 2. Utilizatorii pot vizualiza lista actualizata, cu statusul bug-urilor schimbat. | | |
| Normal flow | * Programatorul se logheaza in contul sau din cadrul sistemului. * Sistemul incarca lista de buguri, ordonate descrescator in functie de gradul de risc al fiecarui bug. * Programatorul selecteaza bug-ul pe care doreste sa il modifice. * Programatorul declara ca bug-ul a fost rezolvat, schimbandu-I statusul * Programatorul confirma ca a rezolvat bug-ul respective printr-un mesaj de confirmare oferit de sistem. * SIstemul ofer un mesaj, prin care se sepccifica ca bug-ul si-a schimbat starea din daunator in rezolvat. * Se afiseaza lista bug-urilor, cu bug-ul respective modificat, acesta nefiind accesibil din acest moment. | | |
| Alternative flows | * Programatorul incearca sa schimbe starea unui bug rezolvat, facandu-l din nou danator, ceea ce nu se poate intampla. | | |
| Exceptions | * Daca programatorul incearca sa modifice un bug deja rezolvat, sistemul va afisa un mesaj de eroare prin care se specifica faptul ca un bug rezolvat nu mai poate fi modificat. | | |

Iteratia 1:

* Autentificare in sistem + interfata pentru autentificare si afisare bug-uri.
* Crearea bazei de date pentru bug-uri, programatori si testeri.
* Afisarea bug-urilor existente in sistem (tester/programator).

Iteratia 2:

* Adaugare bug in sistem
* Filtrare bug-uri dupa grad de risc descrescator
* Realizarea repository-ului pentru implementarea operatiilor CRUD.
* Filtrare bug dupa status daca riscul este egal

Iteratia 3:

* Modificarea statusului unui bug.
* Adaugarea prioritatii bug-urilor, acesta putandu-se modifica numai daca au grad de risc maxim.
* Finalizarea aplicatiei.