

Завдання 2

Severity - Critical/Priority - Low

Severity - Minor / Priority - Highest

1. Severity - Critical / Priority - Low:

а) У кавовій машині може бути баг, коли машина продовжує працювати, навіть якщо кава вже приготовлена та її не вийняли з машини. Це може привести до перегріву та може бути небезпечним для здоров'я користувача. Проте, користувач може відключити машину вручну, тому цей баг може мати низький пріоритет для вирішення.

б) При використанні програмного забезпечення для фінансових транзакцій, можливий баг, який дозволяє зловмиснику здійснити переказ коштів з рахунку користувача без його згоди та авторизації. Це може бути надзвичайно критично для користувача, але при цьому баг може бути невідомий для компанії, яка надає послугу, тому що такі транзакції можуть бути невеликі та не привертати увагу.

2. Severity - Minor / Priority - Highest:

а) У електрочайнику може бути баг, коли під час завантаження води до ємності вода може проливатися через щілини. Це може бути незначним багом, оскільки він може бути відчутний тільки під час завантаження води, але цей баг може бути високого пріоритету для вирішення, оскільки може призвести до серйозної пожежі.

б) У тостері може бути баг, коли тостер нагрівається нерівномірно, що може призвести до перепалених або згорілих тостів. Це може мати високий пріоритет для вирішення, оскільки впливає на якість продукту.

Завдання 3

Як керівниця, я маю створити життєвий цикл багу для ефективного керування процесом його виявлення, відстеження та виправлення. Основні статуси, які входять в життєвий цикл багу, зазвичай такі:

1. Виявлення багу - цей статус відображає період, коли баг виявлено та зареєстровано в системі управління помилками.

2. Підтвердження багу - після виявлення багу потрібно підтвердити, що це справді баг, а не помилка в системі.
3. Присвоєння пріоритету - цей статус вказує на те, наскільки важливо виправити цей баг. Зазвичай пріоритет визначається відповідно до впливу багу на користувачів та бізнес-процеси.
4. Призначення відповідальності - на цьому етапі призначається технічний спеціаліст, який буде працювати над виправленням багу.
5. Виправлення багу - на цьому етапі технічний спеціаліст виправляє баг і перевіряє, чи він більше не повторюється.
6. Перевірка виправлення - після виправлення багу його необхідно перевірити, щоб підтвердити, що баг було успішно виправлено.
7. Закриття багу - коли баг виправлено та перевірено, він закривається в системі управління помилками.

Ці статуси можуть відрізнятися в залежності від особливостей процесу розробки програмного забезпечення та конкретних потреб команди, але загальна послідовність має допомогти забезпечити ефективне керування цим процесом та підвищити якість програмного забезпечення.