**COMP 3322**

|  |  |
| --- | --- |
| ***Assignment 05*** | ***Student Name : Meerim Samakova*** |
| ***Assignment Date: 11/30/2023*** | ***Student id : 10003737*** |

**1. Explain how the number of known defects remaining in a program at the time of delivery affects product support.**

The number of defects or bugs that are already known to exist in a software program when it is delivered can have an impact, on aspects of product support;

Customer Satisfaction: When there are known defects it can greatly affect the user experience and lead to customer dissatisfaction. Users who encounter problems might not be happy with how the product performs, which can result in reviews and a loss of trust in the software.

Support Volume: Having a number of defects often leads to an increase in customer support demands. Users who experience issues may seek help resulting in support tickets, calls or inquiries. This increased workload can strain support teams. Affect their efficiency and response times.

Allocation of Resources: Dealing with known defects may require resources to be redirected from development tasks or improvements. Support teams may need to spend more time troubleshooting, debugging and resolving issues, which can impact their ability to focus on implementing features or enhancements.

**2. Testing is meant to show that a program does what it is intended to do. Why may testers not always know what a program is intended for?**

Testers may not always have an understanding of the intended purpose of a program due, to reasons.

Insufficient Documentation or Requirements: In cases the project might lack documentation or clear requirements. Testers rely on specifications, user stories or documentation to comprehend the intended functionalities and behaviors of the software. If these are incomplete, ambiguous or outdated testers may face difficulties in grasping the purpose or expected behavior of the program.

Ambiguous or Evolving Goals: Software development projects can evolve and objectives may change during the development process. Testers may not always be kept up to date about the changes or updates in project goals leading to confusion about the intended functionalities or desired outcomes.

Communication Challenges: In development teams or organizations with dispersed teams communication gaps can arise. Testers might not have access to stakeholders or subject matter experts who define the programs objectives resulting in a lack of guidance or understanding.

**3) Some people argue that developers should not be involved in testing their own code but that all testing should be the responsibility of a separate team. Give arguments for and against testing by the developers themselves.**

Advantages of Developers Conducting Testing, on Their Code;

Immediate Issue Detection: Developers possess an in depth understanding of the code they have written. By testing their work they can quickly identify problems leading to faster debugging and resolution. This approach helps them catch and rectify bugs at a stage minimizing the chances of these Disadvantages of Developers Testing Their Own Code;

Bias and Blind Spots: Due to their familiarity, with the code developers may unintentionally overlook issues or test scenarios that align with their assumptions potentially missing out on edge cases or user perspectives.

Conflict of Interest: Developers may prioritize ensuring that their code functions as intended than thoroughly exploring failure points or negative scenarios. As a result there is a possibility of evaluation which could lead to overlooking defects in their own work.

**4) What are the benefits of involving users in release testing at an early stage in the testing process? Are there disadvantages in user involvement?**

Including users, in the testing process at a stage provides advantages:

User Feedback on Experience: Users offer insights into the softwares usability and functionality from an end user perspective. Involving them early allows them to test the user interface, navigation and overall experience helping identify areas for improvement.

Validation of Requirements: User involvement helps validate if the software meets their requirements and expectations. Their feedback ensures that the software aligns with user needs resulting in satisfaction upon release.

Identification of Critical Issues: Users may uncover issues or bugs that internal testing might have missed. Early involvement allows us to identify these issues before a release, enabling resolution and minimizing their impact on the final product.

While there are advantages there might be some drawbacks when it comes to involving users in release testing;

Limited Technical Knowledge; Users may not have the expertise, which could make it challenging for them to effectively communicate issues or provide detailed bug reports. This could potentially impact the accuracy and clarity of the feedback they provide.

Biased Feedback; Users feedback might be influenced by preferences or limited exposure, to the software leading to opinions. This could result in prioritizing changes based on preferences, than considering the broader needs of all users.

**5) What is regression testing? Explain how the use of automated tests and a testing framework such as JUnit simplifies regression testing.**

Regression testing is a form of software testing that aims to verify that recent code modifications or updates, to an application haven't negatively impacted its existing functionalities. It involves retesting functioning parts of the software to uncover any side effects or problems caused by new code implementations, updates or fixes.

Automated tests and testing frameworks like JUnit make regression testing more convenient in ways;

Automation of Test Suites: Automated tests, developed using frameworks like JUnit can be designed to cover aspects of the softwares functionalities. Once created these test suites can be repeatedly executed with intervention simplifying the process of efficient regression testing.

Reproducibility: Automated tests ensure reproducible test scenarios. They guarantee that the same tests are executed in a manner every time reducing errors and providing reliable results—something crucial for effective regression testing.

Execution: Automated tests run faster compared to manual testing methods. They can cover a range of test cases. Execute them swiftly providing quicker feedback, on whether recent code changes have introduced any unexpected issues.