**Many software vendors such as Microsoft, Adobe, and Apple ship software packages with both known and unknown defects in them. Do you feel that it is ethical for them to do so? Why or why not?**

Step 1:

When we have time, we'll change that. Simply continue to develop while waiting! How are you supposed to know how much future repairs will cost? — A manager of engineering who is keen to continue developing software

"Let's uncover every flaw before the system test. The client will hold out for the good. — An engineering manager was worried that shipping a defective product might discourage customers from purchasing it.

Many engineering managers are faced with difficult choices regarding which flaws should be fixed when. The type of product, the dangers involved in selling products with known or unknowable flaws, your development procedures, and the expense of rectifying the flaw all play a part in whether you decide to fix it or not.

Step 2:

When individuals are completely focused on detecting and correcting bugs during system testing, it is simplest to estimate this cost. Start by counting the number of fixes completed. You are aware of the quantity of workers (developers, testers, writers, and others), the cost per person-day, and the length of the system test. The fix value is crucial since the whole cost can be unexpectedly large.

Company A did track the engineering time needed to remedy a fault after the product was shipped, despite not tracking the time to find and fix a defect during the specifications, design, and development phases. After shipment, Company A encountered a typical issue:

Some clients were quite dissatisfied with the quality of the products, most notably in those instances where the engineers were unable to correct the issues prior to the product's delivery. Senior management was contacted by a disgruntled Very Important Customer, who then ordered that Engineering make the error as a "emergency" remedy.

Step 3:

In my opinion it is not ehical, because during the development of a software the developers have to put all effort to get a quality product., but some of them are to ready to test the product throughout the development process.

**Do you agree or disagree with the argument made in this case that many hardware and/or software upgrades are unnecessary? Why or why not?**

Step 1:

Upgrades to hardware and software are required to confirm compliance with PCI Data Security Standards and prevent the issuance of a PCI Assessment as a result of an incident.

Step 2:

The most recent version of the software is a software upgrade. It typically includes significant upgrades or completely new operating systems that significantly alter or update the application, operating system, or software.

A software upgrade would involve the addition of a brand-new security measure or functionality. For instance, every new iPhone often has a software upgrade that will soon be available, and sometimes older iPhone models can't handle them.

Upgrades to software are frequently so different from the prior version that you must pay for them. You might also get lifelong access to upgrades, depending on the service agreement that came with your purchase. You must carefully read the information you are given while making a purchase.

As an alternative, you can free download software updates, sometimes known as software patches. It's an improved version of the software, operating system, or programme you already have. On these systems, upgrades frequently claim to have fixed bugs or other issues that were present in a previous version of the software.

Every time you download or purchase new software, the purpose is typically to improve your experience using previously purchased or downloaded technologies. That isn't always the case, though. For instance, consumers have reported major troubles with the most recent Apple iPhone update.