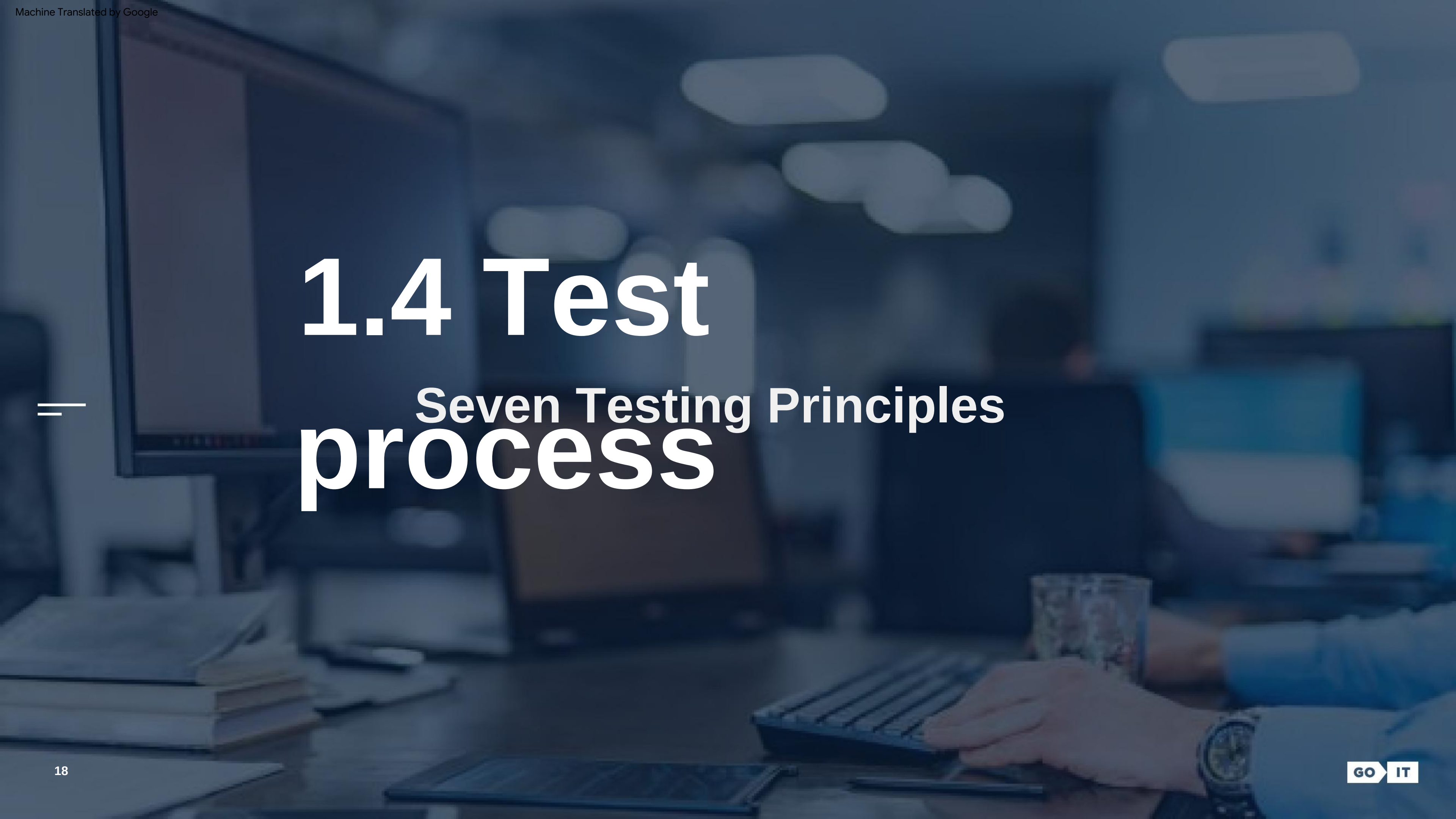
GO IT

Testing process



1.4 Test process

Seven Testing Principles

Testing shows the presence of defects, not the absence of them

Testing discovers the presence of defects, but it does not prove that there are no defects in the program. Testing reduces the risk of bugs that may be present in the product, but it doesn't matter if defects are not discovered - testing does not prove that our program was written without defects.

If the tester found no defects, it only says one thing...

... the tester found no defects.

If the tester found bugs in the program - it means that they are there.
If the tester found no bugs in the program, it also means that they are there.

Accurate testing is not possible

Testing everything (all combinations of data and prior conditions) is impossible except in trivial cases.

To run the calculator's accurate tests, it takes 4 human lives with an average lifespan of 60 years, not including time for eating and sleeping.

Tests that have been done beforehand save time and money

To find defects at an early stage, testing activities should be started as early as possible in the development lifecycle of a program or system and should be focused on certain objectives.

Defect clustering (bugs like to be friends :)

Most often, a minor part of the program's functionality contains most of the bugs, defects and inconsistencies with the documentation. If we skillfully determine the minimum of the most important areas of the program and prioritize them, then we can be sure with a high probability that most of the critical defects will be found in this minimum area.

The pesticide paradox

If the same tests are repeated again and again, the same set of test cases will not find any more defects. In order to overcome this "pesticide paradox", review test cases regularly. You also need to write new tests to check different parts of the program or system to potentially find more defects.

The tests depend on the context

Tests are performed differently depending on the context. We will test a banking application differently than a blog service.

False absence of errors

Disclosure and removal of defects does not mean that there are no more defects in the system. For example, reliably testing all marked requirements and fixing all discovered defects can lead to a system that is difficult to use, does not meet the needs and expectations of users, or is inferior to other competing systems.