

Software Engineering, Assignment #04

20B030299
Rymbayeva Anelya,
FIT, 3 course

Exercise: 8.3 - 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

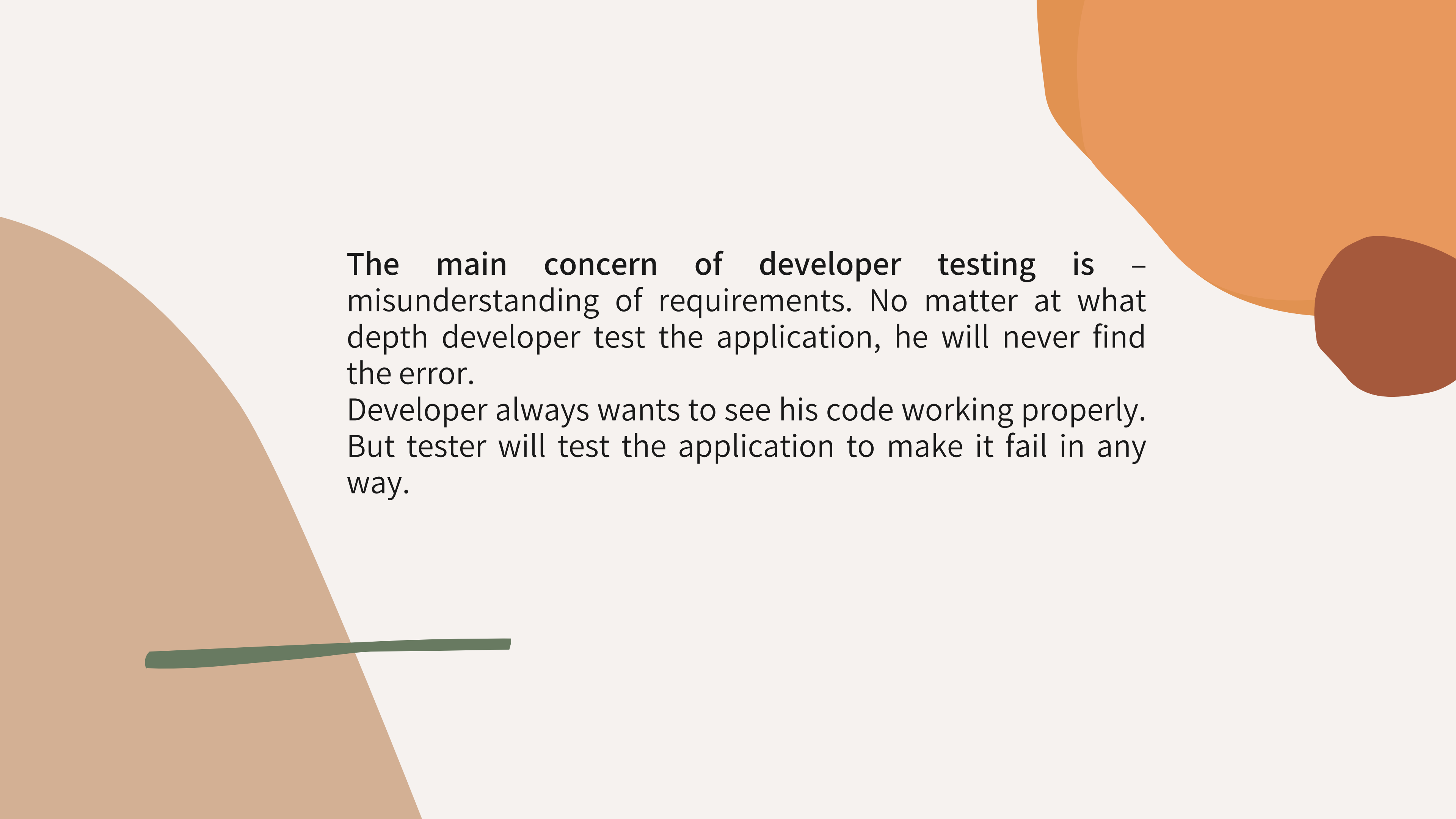


PRO for Inside Team:

- The team understands the problems that their program is tackling, and thus understands the problems that it might run into.
- They can write tests that accurately reflect the programs' future challenges.
- An outsider team may try to verify the code, without fully understanding the initial problem.
- Testing becomes easy as they know the areas of code which need testing
- Saves time, as they need not learn about the code


CON for Inside Team:

- Depending on the team, and often pressured by time, they could write test units that insufficiently test the quality of the code, or are unable to think of challenges to their code beyond the solution that they've already tested.
- One may not easily identify their own errors
- Testing may be compromised so as to reduce their error rate
- Increases the effort of developers and may compromise with quality







The main concern of developer testing is – misunderstanding of requirements. No matter at what depth developer test the application, he will never find the error.

Developer always wants to see his code working properly. But tester will test the application to make it fail in any way.



Exercise 9.9 - Explain the differences between software re-engineering and refactoring

Refactoring is the process of changing a software system in such a way that it does not alter the external behavior of the code, yet improves its internal structure.

-  clean up code that minimizes the chances of introducing bugs.
-  improving the design of the code after it has been written
-  refactoring does not add features or functionalities in a software system
-  It makes a software system easier to understand and cheaper to modify without changing its observable behavior by changing its internal structure.

Re-engineering	Refactoring
It is a maintenance process, so that the understandability and the structure of the program can be improved	The original structure of the program is improved. By doing so, the complexity of the program can be reduced
It can be applied even to legacy software	It is limited to object oriented development programs
You can add any new functionality to the already existing system	Addition of new functionality is not allowed
Reverse engineering is allowed	It is based on agile methods, so reverse engineering is not allowed

THE PURPOSE OF REFACTORING:

- 1. Refactoring Improves the Design of Software
- 2. Refactoring Makes Software Easier to Understand
- 3. Refactoring Helps Finding Bugs
- 4. Refactoring Helps Programming Faster

Software Re-engineering means - reorganizing or restructuring or modifying existing software systems to make them more maintainable.