

# Software Engineering

**Lesson #07 - Lecture**



Lesson #07 - Lecture

Your KBTU 202309 Software Engineering  
class information is updating ...

Lesson #07 update is in progress

This will take around 2 hours to complete

Please, don't turn off your head



# Software Testing

---

# Software Testing

# Software Testing

---

```
#include<iostream>
Using namespace std;

int main()
{
    cout << "Software Testing" << endl;

    return 0;
}
```

## Introduction

---

# Software Testing

Testing is intended to show that a program does what it is intended to do and to discover program defects before it is put into use

You check the results of the test run for errors, anomalies, or information about the program's non-functional attributes

**ALPHA**  
TESTING

**BETA**  
TESTING

**RELEASE**



# Software Testing

When you test software, you are trying to do two things:

Demonstrate to the developer and the customer that the software meets its requirements

For custom software, this means that there should be at least one test for every requirement in the requirements document



# Software Testing

When you test software, you are trying to do two things:

Find inputs or input sequences where the behavior of the software is incorrect, undesirable, or does not conform to its specification. These are caused by defects (bugs) in the software

# Software Testing

The first is validation testing, where you expect the system to perform correctly using a set of test cases that reflect the system's expected use

# Software Testing

The second is defect testing, where the test cases are designed to expose defects

Introduction

# Software Testing

“Testing can only show  
the presence of errors,  
not their absence”

Edsger Dijkstra

Turing award  
winner 1972



# Software Testing

Verification and validation processes are concerned with checking that software being developed meets its specification and delivers the functionality expected by the people paying for the software

- Validation: Are we building the right product?
- Verification: Are we building the product right?

## Introduction

---

# Software Testing

Verification & Validation - Georgia Tech - Software Development Process

<https://www.youtube.com/watch?v=gQrSxbfUjug>





# Verification & Validation



Are we building the product right?



Are we building the right product?

## Verification

- Verify the intermediary products like requirement documents, design documents, ER diagrams, test plan and traceability matrix
- Developer point of view
- Verified without executing the software code
- Techniques used: Informal Review, Inspection, Walkthrough, Technical and Peer review



## Validation

- Validate the final end product like developed software or service or system
- Customer point of view
- Validated by executing the software code
- Techniques used: Functional testing, System testing, Smoke testing, Regression testing and Many more

# Software Testing

Software verification is the process of checking that the software meets its stated functional and non-functional requirements



# Software Testing

Validation is a more general process

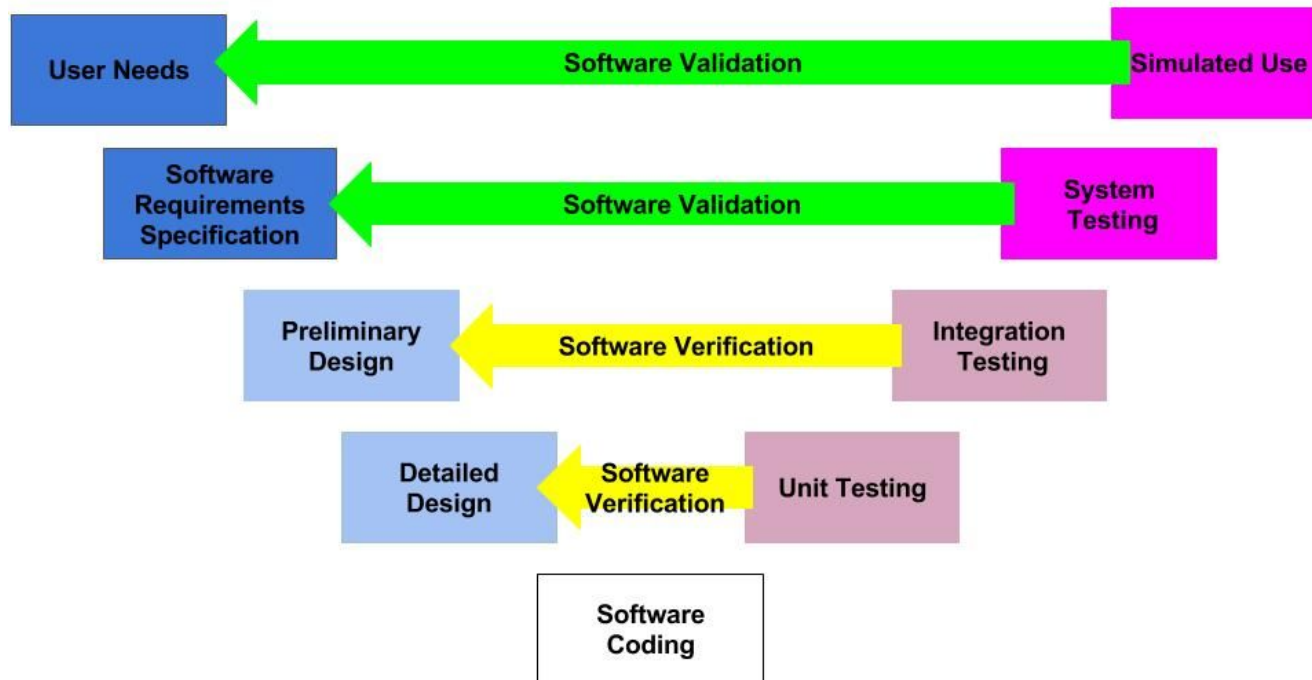
The aim of software validation is to ensure that the software meets the customer's expectations

It goes beyond checking conformance with the specification to demonstrating that the software does what the customer expects it to do

## Introduction

---

# Software Testing



# Software Testing

Typically, a commercial software system has to go through three stages of testing:

- Development testing, where the system is tested during development to discover bugs and defects
- Release testing, where a separate testing team tests a complete version of the system before it is released to users
- User testing, where users or potential users of a system test the system in their own environment

# Introduction

---



I don't  
always test  
my code  
But when  
I do, I do it  
in  
production





I ran some  
tests in  
dev and  
left the  
rest for QA



# Introduction

---



ГОСУДАРСТВЕННЫЕ УСЛУГИ  
И ИНФОРМАЦИЯ ОНЛАЙН



Портал қазір қолжетімсіз, біз оны қалпына келтіру бойынша жұмыстар жүргізудеміз. Келтірілген қолайсыздықтар үшін кешірім сұраймыз!

Портал сейчас недоступен, мы работаем над его восстановлением. Приносим свои извинения!

The Portal is temporary unavailable, the restoration is in progress! Sorry for any inconveniences caused!



**По причине большого количества обратившихся на сайт, в данный момент все свободные ресурсы заняты.**

Все обращения будут обязательно приняты.  
Пожалуйста обратитесь на сайт чуть позже.

**Сайтқа өтініш берушілердің көп болуына байланысты, қазіргі уақытта барлық**



ГЛАВНАЯ

КАЗАХСТАН

В МИРЕ

ФАКТЫ

СПОРТ

ЗДОРОВЬЕ

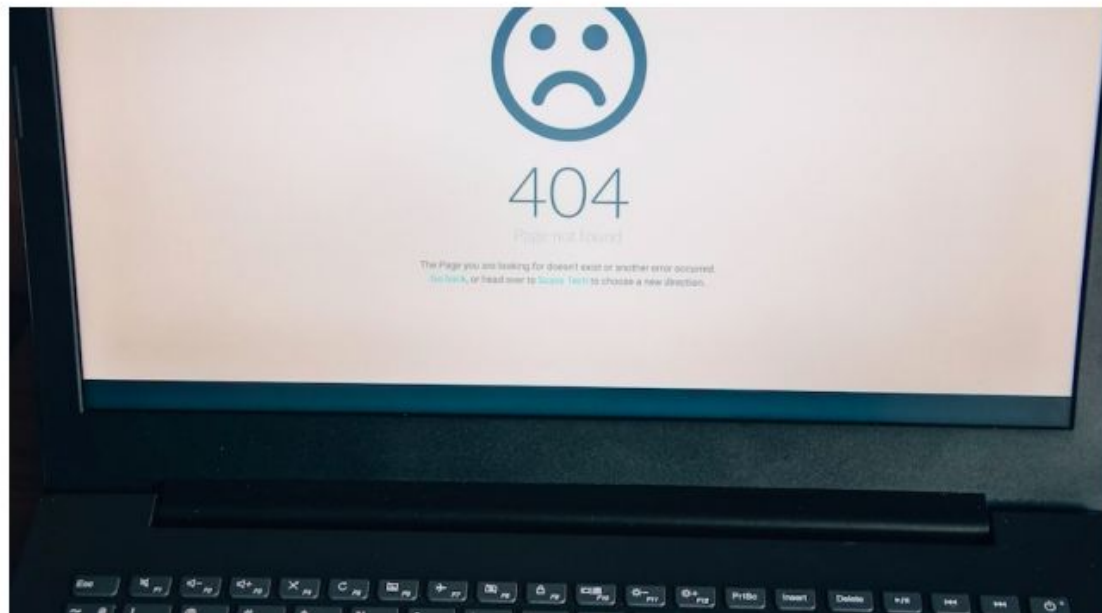
ЭТО ИНТЕРЕСНО

ВИДЕО

## Сайт для подачи заявок на 42500 тенге "упал" из-за перегрузок



17-ИЮЛ-2020, 11:14



### ГОРЯЧАЯ ТЕМА

- 1** Запретить незастрахованным по ОСМС казахстанцам водить машину предлагает Минздрав
- 2** Air Astana ищет варианты для возвращения казахстанцев из Израиля
- 3** Крупные предприятия не могут открыть в Актобе из-за дефицита газа
- 4** 15 октября в Актобе состоится прощальный матч Маркоса Пиззелли
- 5** 54 гражданина Израиля и 163 казахстанца Air Astana привезёт в Алматы
- 6** Разбитые дороги, изношенные автобусы - что

## Agenda: Lesson #07 - Software Engineering - Lecture

---

1 Development testing

2 Test-driven development

3 Release testing

4 User testing

## Development testing

---

# Development testing

Development testing includes all testing activities that are carried out by the team developing the system

The tester of the software is usually the programmer who developed that software

# Development testing

There are three stages of development testing:

- Unit testing, where individual program units or object classes are tested
- Component testing, where several individual units are integrated to create composite components
- System testing, where some or all of the components in a system are integrated and the system is tested as a whole

## Development testing

---

# Development testing

Development testing is primarily a defect testing process, where the aim of testing is to discover bugs in the software

It is therefore usually interleaved with debugging—the process of locating problems with the code and changing the program to fix these problems

# Development testing

## Unit Testing

- When you are testing object classes, you should design your tests to provide coverage of all of the features of the object
- This means that you should test all operations associated with the object; set and check the value of all attributes associated with the object; and put the object into all possible states
- This means that you should simulate all events that cause a state change

# Development testing

## Choosing unit test cases

Testing is expensive and time consuming, so it is important that you choose effective unit test cases. Effectiveness, in this case, means two things:

- The test cases should show that, when used as expected, the component that you are testing does what it is supposed to do
- If there are defects in the component, these should be revealed by test cases

# Development testing

## Choosing unit test cases

Two strategies that can be effective in helping you choose test cases are:

- Partition testing, where you identify groups of inputs that have common characteristics and should be processed in the same way
- Guideline-based testing, where you use testing guidelines to choose test cases



## Development testing

---

# Development testing

## Component testing

Testing composite components should therefore focus on showing that the component interface or interfaces behave according to its specification

You can assume that unit tests on the individual objects within the component have been completed

# Development testing

## Component testing

There are different types of interface between program components and, consequently, different types of interface error that can occur:

- **Parameter interfaces** These are interfaces in which data or sometimes function references are passed from one component to another

# Development testing

## Component testing

There are different types of interface between program components and, consequently, different types of interface error that can occur:

- Shared memory interfaces These are interfaces in which a block of memory is shared between components

# Development testing

## Component testing

There are different types of interface between program components and, consequently, different types of interface error that can occur:

- **Procedural interfaces** These are interfaces in which one component encapsulates a set of procedures that can be called by other components

# Development testing

## Component testing

There are different types of interface between program components and, consequently, different types of interface error that can occur:

- Message passing interfaces These are interfaces in which one component requests a service from another component by passing a message to it

## Development testing

---

# Development testing

## System testing

System testing during development involves integrating components to create a version of the system and then testing the integrated system

System testing checks that components are compatible, interact correctly, and transfer the right data at the right time across their interfaces

## Development testing

---

# Development testing

## System testing

System testing should focus on testing the interactions between the components and objects that make up a system

You may also test reusable components or systems to check that they work as expected when they are integrated with new components

Development testing

---

# Development testing

Meet Test Engineers at Google

<https://www.youtube.com/watch?v=C7OLZf5099Y>





## Agenda: Lesson #07 - Software Engineering - Lecture

---

1

Development testing

2

**Test-driven development**

3

Release testing

4

User testing

## Test-driven development

---

# Test-driven development

Test-driven development (TDD) is an approach to program development in which you interleave testing and code development

You develop the code incrementally, along with a set of tests for that increment

Test-driven development was introduced as part of the XP agile development method. However, it has now gained mainstream acceptance and may be used in both agile and plan-based processes

## Test-driven development

---

# Test-driven development

Test-driven development helps programmers clarify their ideas of what a code segment is actually supposed to do

To write a test, you need to understand what is intended, as this understanding makes it easier to write the required code

## Test-driven development

---

# Test-driven development

As well as better problem understanding, other benefits of test-driven development are:

- Code coverage
- Regression testing
- Simplified debugging
- System documentation

## Test-driven development

---

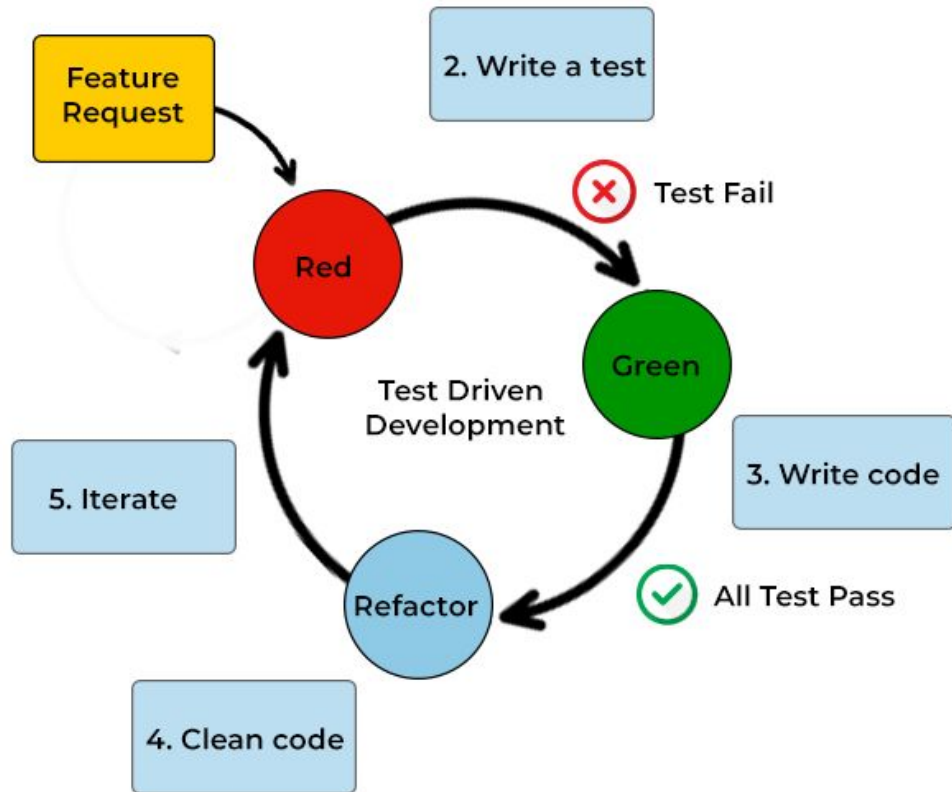
# Test-driven development

It is also claimed that use of TDD encourages better structuring of a program and improved code quality

However, experiments to verify this claim have been inconclusive

# HOW TDD WORKS

1. Start here



## Test-driven development

---

# Test-driven development

It is a “break first, build later” approach.



## Agenda: Lesson #07 - Software Engineering - Lecture

---

1

Development testing

2

Test-driven development

3

**Release testing**

4

User testing



## Release testing

---

# Release testing

Release testing is the process of testing a particular release of a system that is intended for use outside of the development team. Normally, the system release is for customers and users

In a complex project, however, the release could be for other teams that are developing related systems

For software products, the release could be for product management who then prepare it for sale

# Release testing

There are two important distinctions between release testing and system testing during the development process:

- The system development, team should not be responsible for release testing
- Release testing is a process of validation checking to ensure that a system meets its requirements and is good enough for use by system customers

## Release testing

---

# Release testing

The primary goal of the release testing process is to convince the supplier of the system that it is good enough for use

If so, it can be released as a product or delivered to the customer

Release testing, therefore, has to show that the system delivers its specific functionality, performance, and dependability, and that it does not fail during normal use

## Release testing

---

# Release testing

Release testing is usually a black-box testing process whereby tests are derived from the system specification

The system is treated as a black box whose behavior can only be determined by studying its inputs and the related outputs

# Release testing

## Requirements-based testing

- A general principle of good requirements engineering practice is that requirements should be testable
- That is, the requirement should be written so that a test can be designed for that requirement
- A tester can then check that the requirement has been satisfied

# Release testing

## Scenario testing

- Scenario testing is an approach to release testing whereby you devise typical scenarios of use and use these scenarios to develop test cases for the system
- A scenario is a story that describes one way in which the system might be used
- Scenarios should be realistic, and real system users should be able to relate to them

# Release testing

## Performance testing

- Once a system has been completely integrated, it is possible to test for emergent properties, such as performance and reliability
- Performance tests have to be designed to ensure that the system can process its intended load
- This usually involves running a series of tests where you increase the load until the system performance becomes unacceptable

**Release testing**

---

# Release testing

Release Testing

<https://www.youtube.com/watch?v=tX41leXiTQw&t=75s>





## Agenda: Lesson #07 - Software Engineering - Lecture

---

1 Development testing

2 Test-driven development

3 Release testing

**4 User testing**

# User testing

User or customer testing is a stage in the testing process in which users or customers provide input and advice on system testing

This may involve formally testing a system that has been commissioned from an external supplier

Alternatively, it may be an informal process where users experiment with a new software product to see if they like it and to check that it does what they need

# User testing

There are three different types of user testing:

1. Alpha testing
2. Beta testing
3. Acceptance testing

# User testing

## User testing



It was beta when Gmail was open to public



2013 - 2020



2020 - Present

## User testing

# User testing

Some companies, like Apple, offer some beta software program(s) users to attend to beta testing activities



## Apple Beta Software Program

Help make the next releases of iOS, iPadOS, macOS, tvOS and watchOS our best yet. As a member of the Apple Beta Software Program, you can take part in shaping Apple software by test-driving pre-release versions and letting us know what you think.

## User testing

---

# User testing

Some software products, like Ubuntu, will be provided beta version to public for users to attend to beta testing activities

## UBUNTU LTS 20.10

Beta version is now available to  
Download



## User testing

---

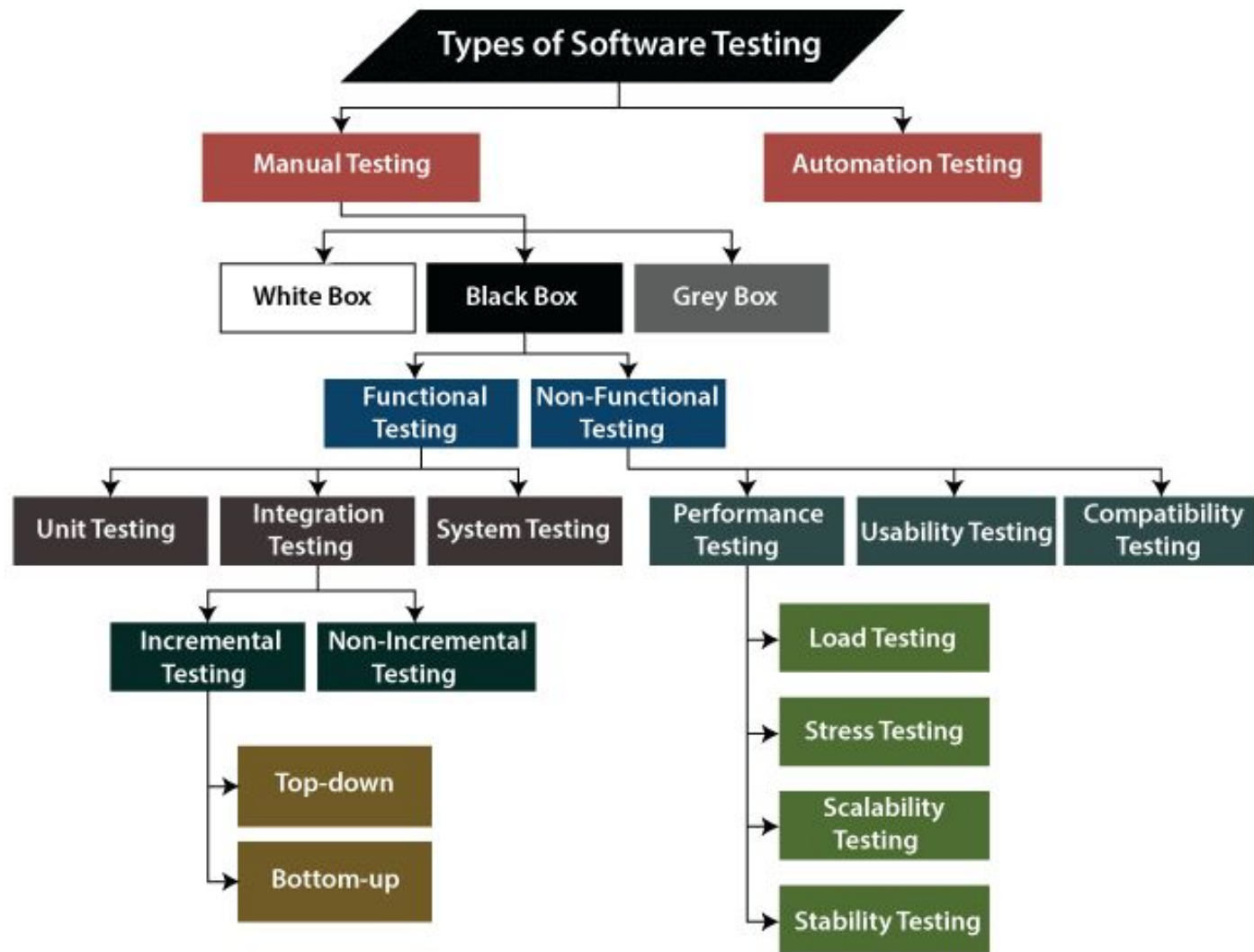
# User testing

Alpha and Beta Testing - Georgia Tech - Software Development Process

[https://www.youtube.com/watch?v=Yo5-0Q0S\\_ks](https://www.youtube.com/watch?v=Yo5-0Q0S_ks)



# Advanced Software Engineering

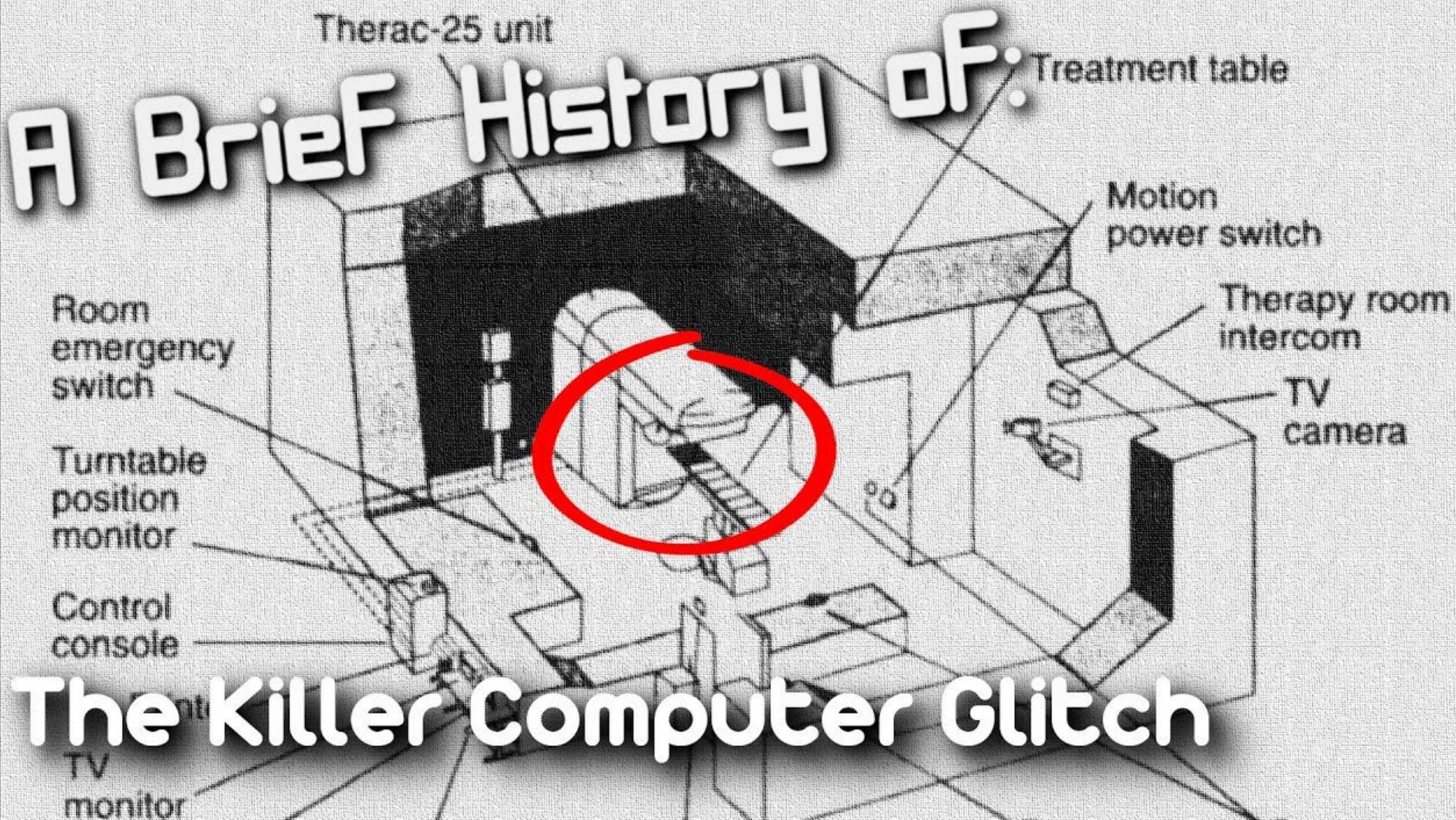




**BUGS**



**BUGS EVERYWHERE**



Therac-25 unit

Treatment table

Motion  
power switch

Therapy room  
intercom

TV  
camera

Room  
emergency  
switch

Turntable  
position  
monitor

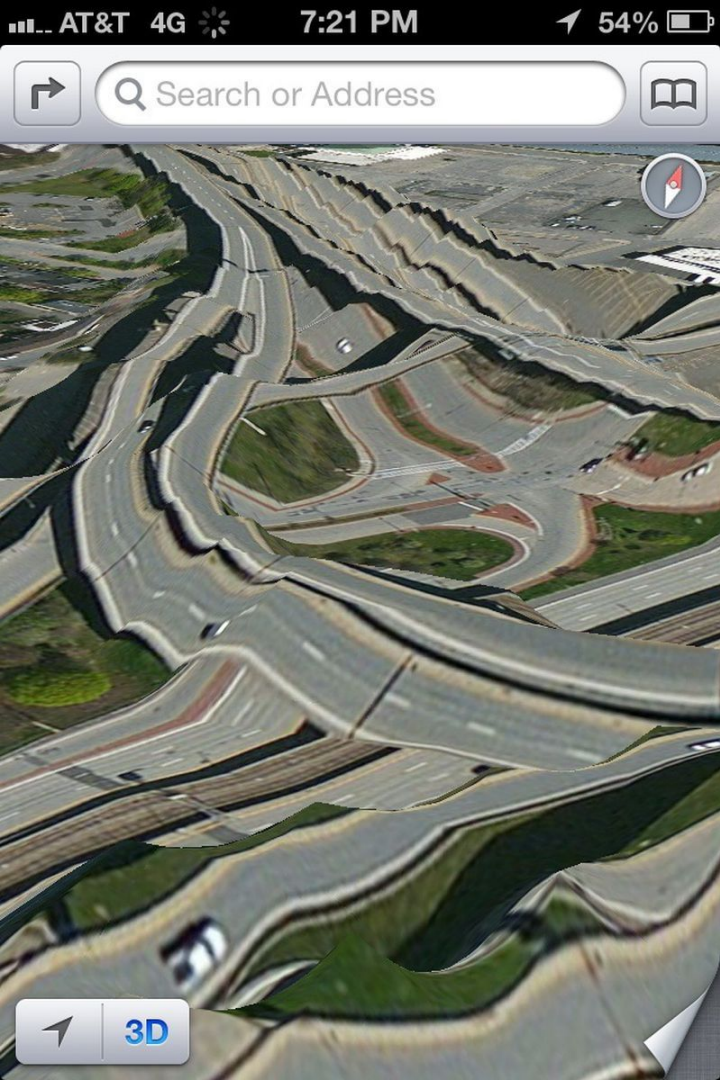
Control  
console

TV  
monitor

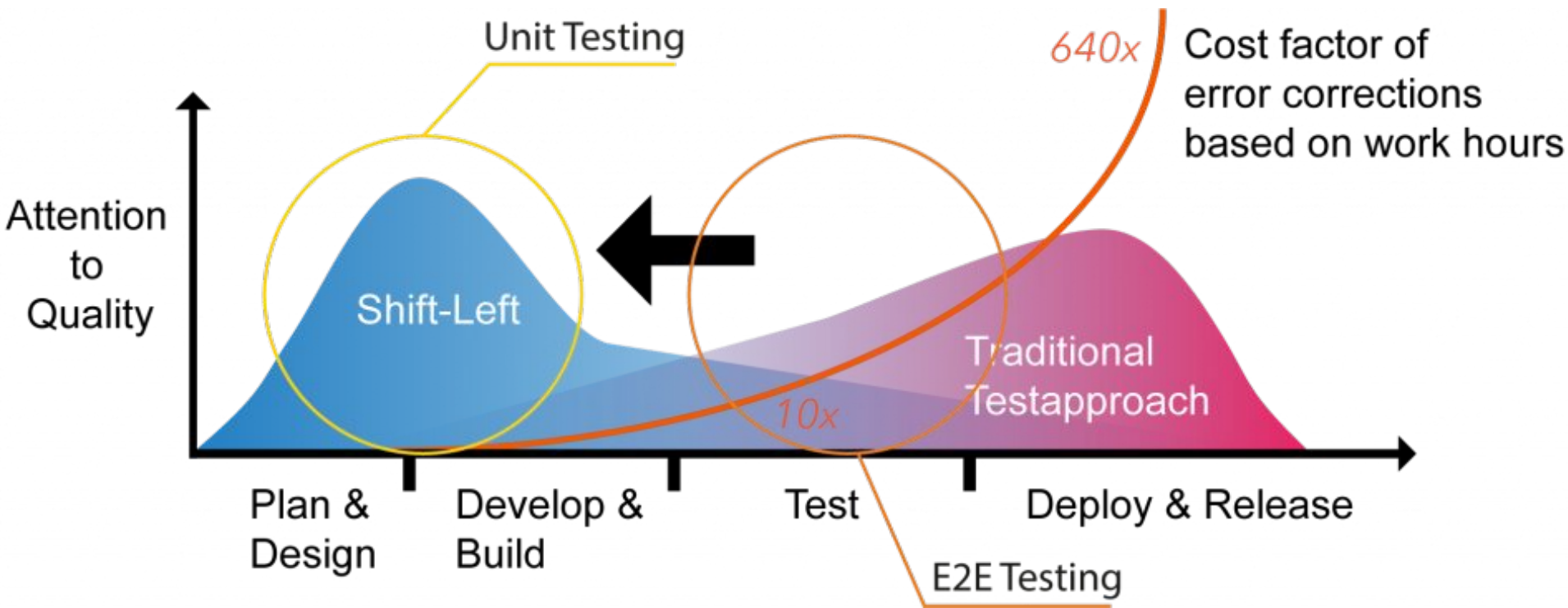
# A Brief History of:

## The Killer Computer Glitch





# User testing

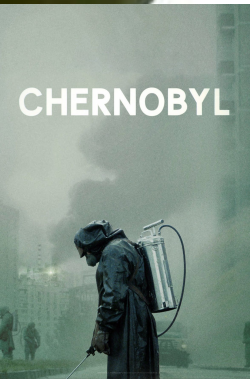




**WHEN YOU LET THE NEW INTERN**



**RUN A TEST ON PRODUCTION**



CHERNOBYL

## **Agenda: Lesson #07 - Software Engineering - Lecture**

---

**Q & A**