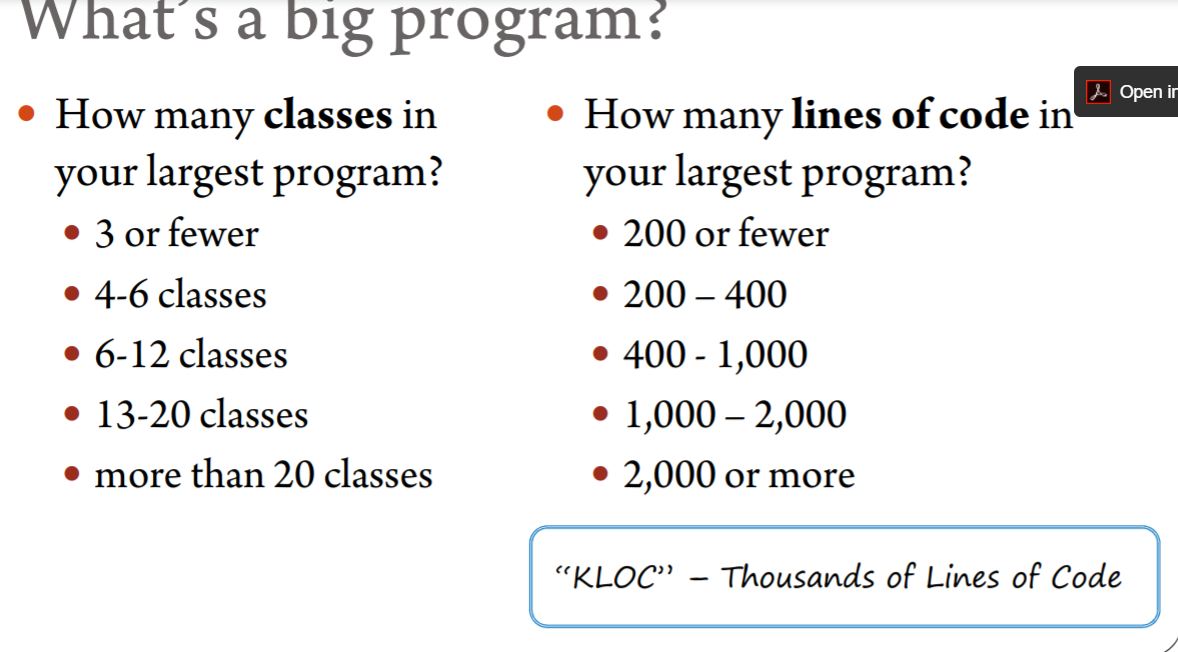
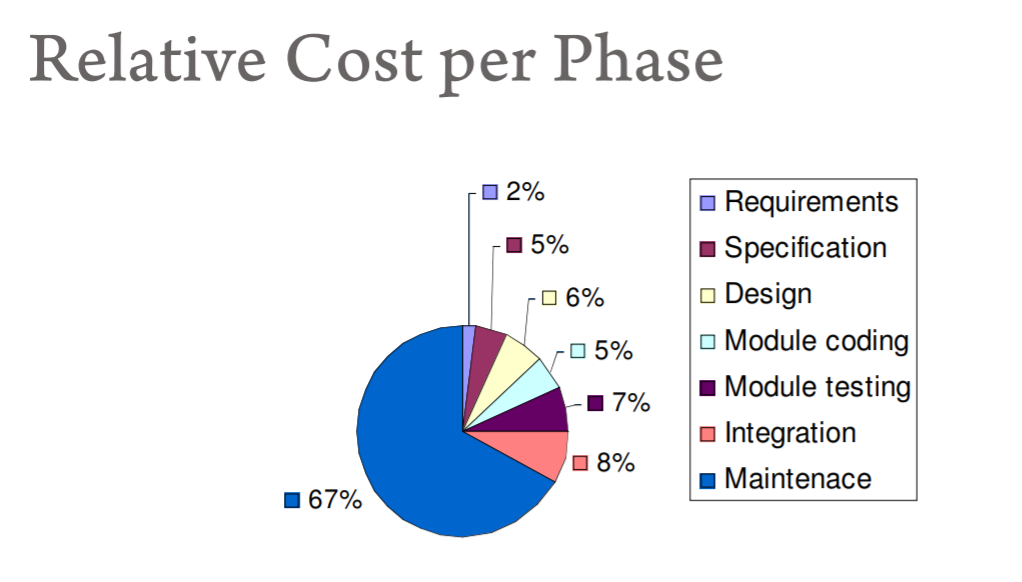
# Module 06 Lecture Notes





* MSD: Chapter 1 (see resources on Collab)

Software Design (cat drawing)

1. Some areas better designed than others

Development Time

1. Every aspect takes a different amount of time. Most on the tail end.

Beta Version

1. Missing features

After Beta Test

1. Missing different features

Advertised

1. Like a lion

What the Customer Wanted

1. Big Question Mark

What software is two versions later

1. A mutant cat

Communication is hard.

Client communication is hard.

What the customer wants is an important question.

# In Data Science

There will be some level of software development.

# Cost of Failure

1. Financial
2. Loss of life
3. Loss of equipment
4. Inconvenience

Cost of failure is becoming very high.

Ariane 501 launch explosion

* Specification and design errors
* No test of inertial reference system
* No test of complete flight control system
* If they had followed standard process the error would have been detected.

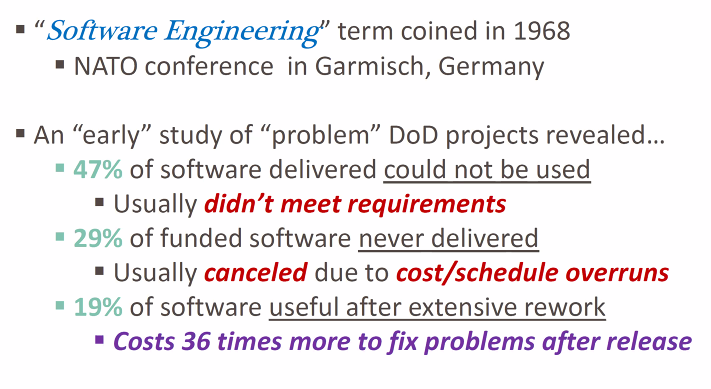
Therac Radiation Therapy Machine

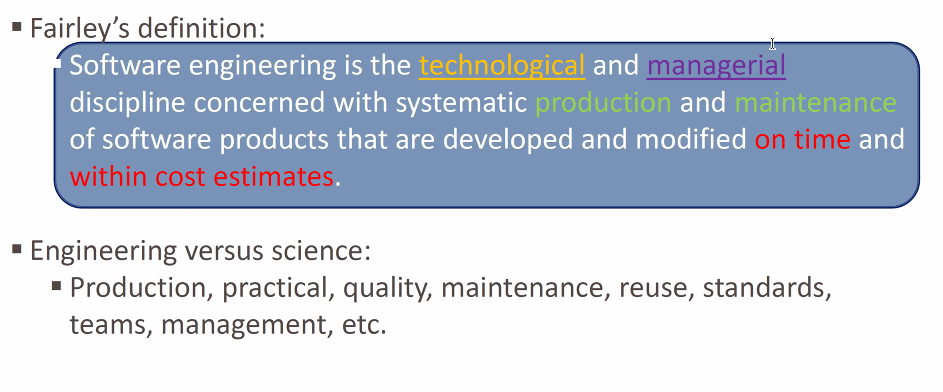
* Designed in a way that it could not be tested
* 3 people died

USS Yorktown

* Division by zero error brought down all machines on network of ships propulsion system.

# “Programming in the Large”



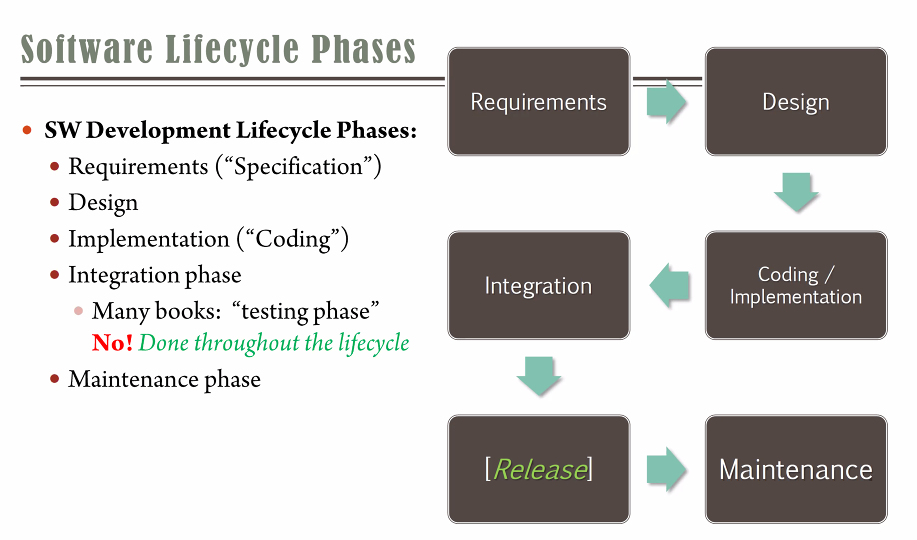


Crossing a small creek == just a log.

Crossing a large body of water == much more complicated.

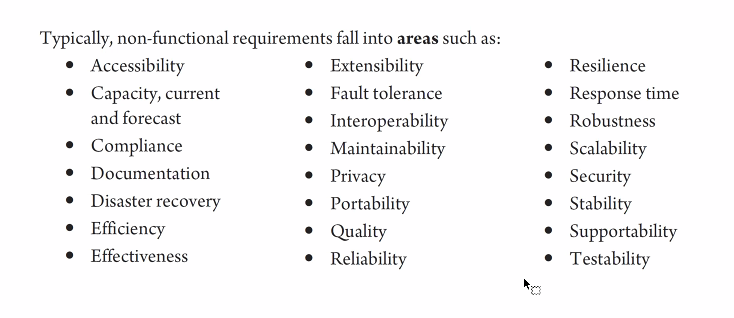
# “Programming in the Large”

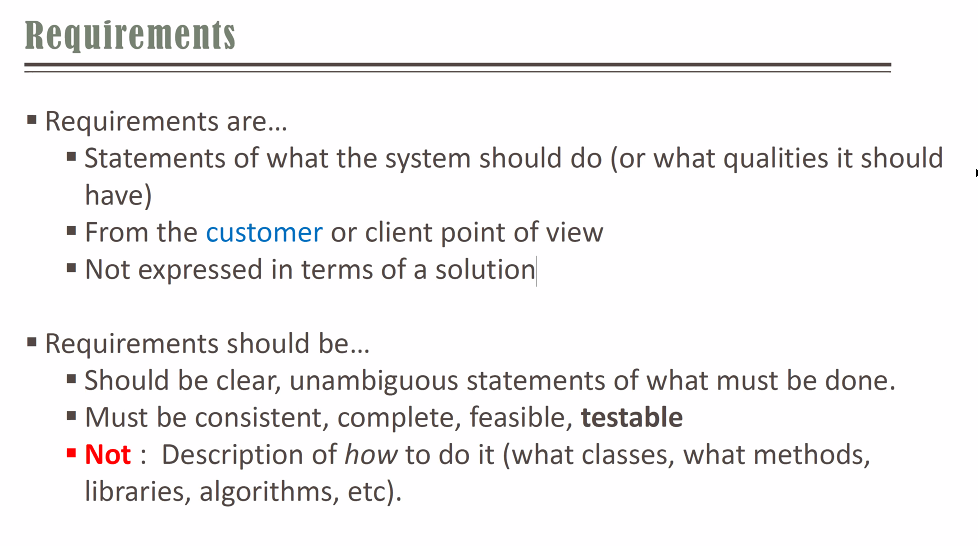
* Needs are different
* Risk larger
* Number of ppl involved
* How long the project takes
* Economics
* Consequences of failure/success
* How long will solution last?
* Maintainability
* Extensible



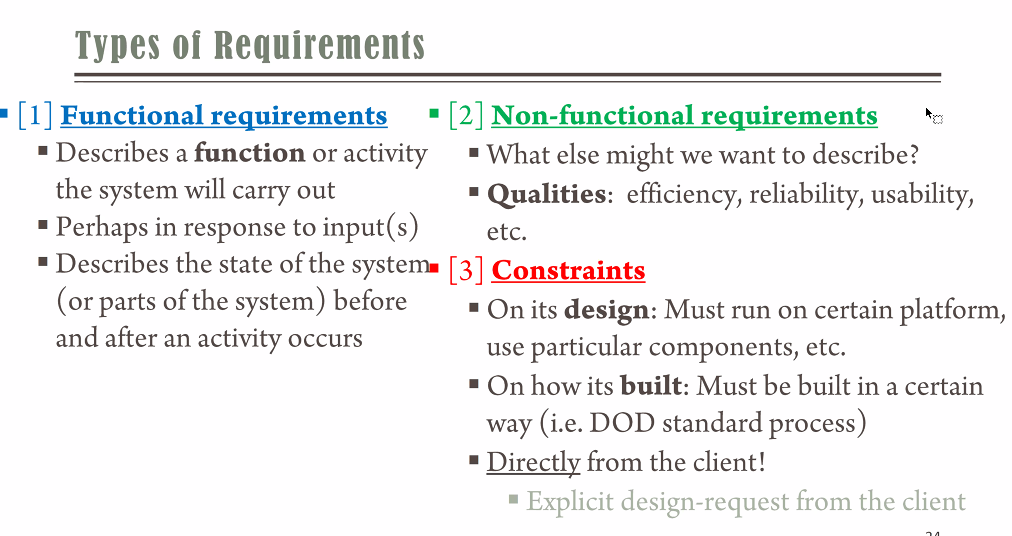
Complete life cycle testing

Test Driven Development





Requirements are NOT **HOW**, but they are **WHAT**.



**Example 1**:

Functional: I want users to log into the system

Non-functional: I want users to log into the system securely

Constraint: The client says the user to log into the system using UVa NetBadge

**Example 2**:

I want my users to be able to use the system after 1 hour or less of training.

**Example 3:**

Functional: I want to project the material on my laptop on the screen

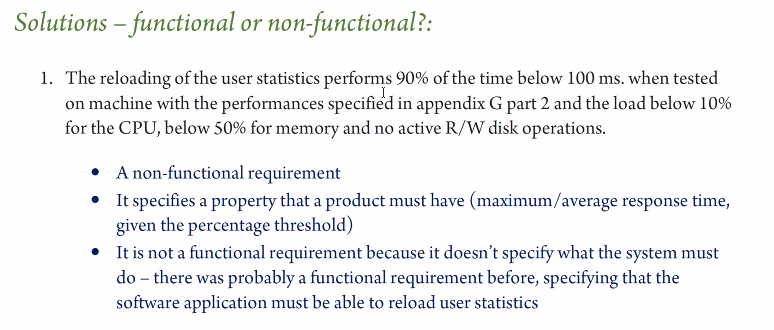
Non-Functional: I want to project in HD

Non-Functional: I want to project on 3 screens

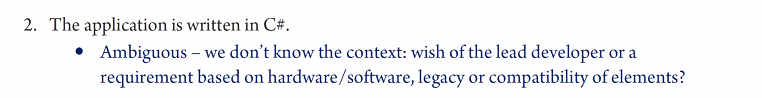
Non-Functional: I want to project tin black and white

***Non-Functional MUST be based on an existing Functional Requirement***

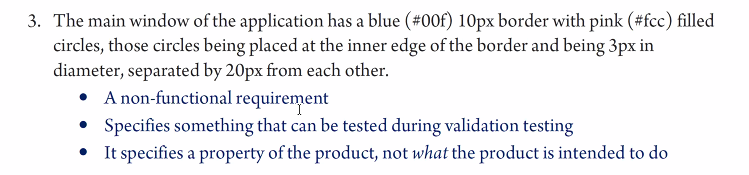
When prioritizing the functional get prioritized first, then non-functional.



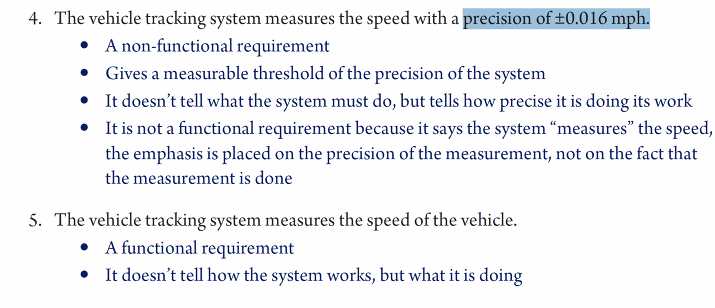
#2 is DESIGN



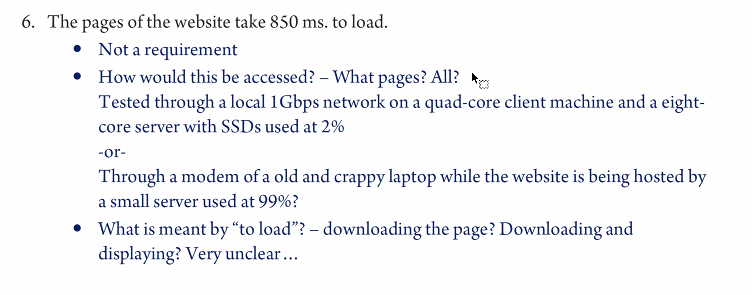
#3 This is non-functional – This gives “qualities” of the software



# 4 is NON functional (or functional requirement. Could be argued either way. If the 0.016 is absolutely necessary (and no more or no less) and the system does not function at all without it, then functional), # 5 is Functional



#6 is ambiguous



# 

# For the Exams

The exam will ask us to State Assumptions or Make an Argument and describe why you selected a particular choice.