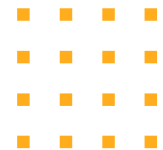


# Software Engineering

## Introduction

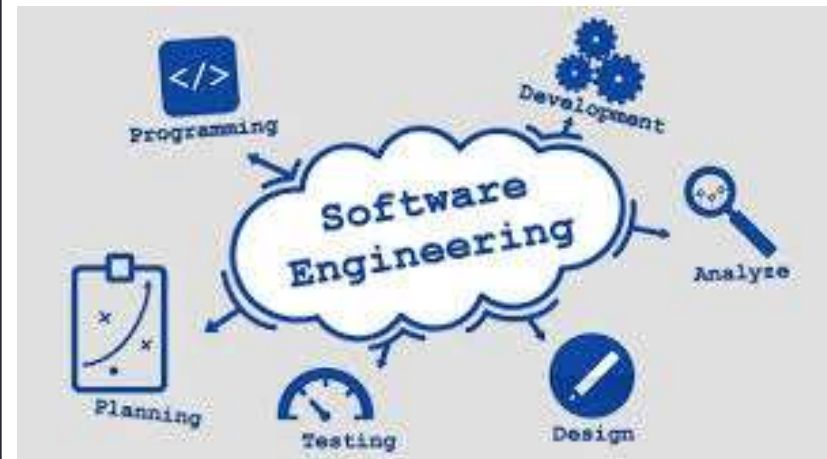




## What is Software Engg??

It is the application of a **systematic, disciplined, quantifiable approach** for the **development, operation, and maintenance** of software

*Definition by IEEE*





# Software Engineering



## Software engineering overview

1. Requirements
2. Design
3. Construction/Coding
4. Testing

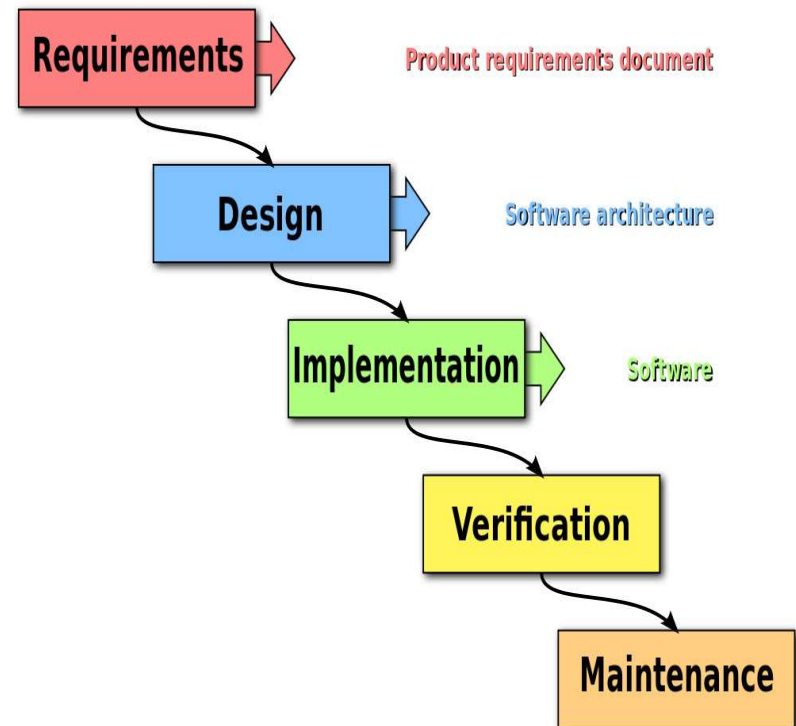
## Development methodologies

The Waterfall development process

Iterative Methodologies

Heavyweight methodologies

Agile methodologies and XP



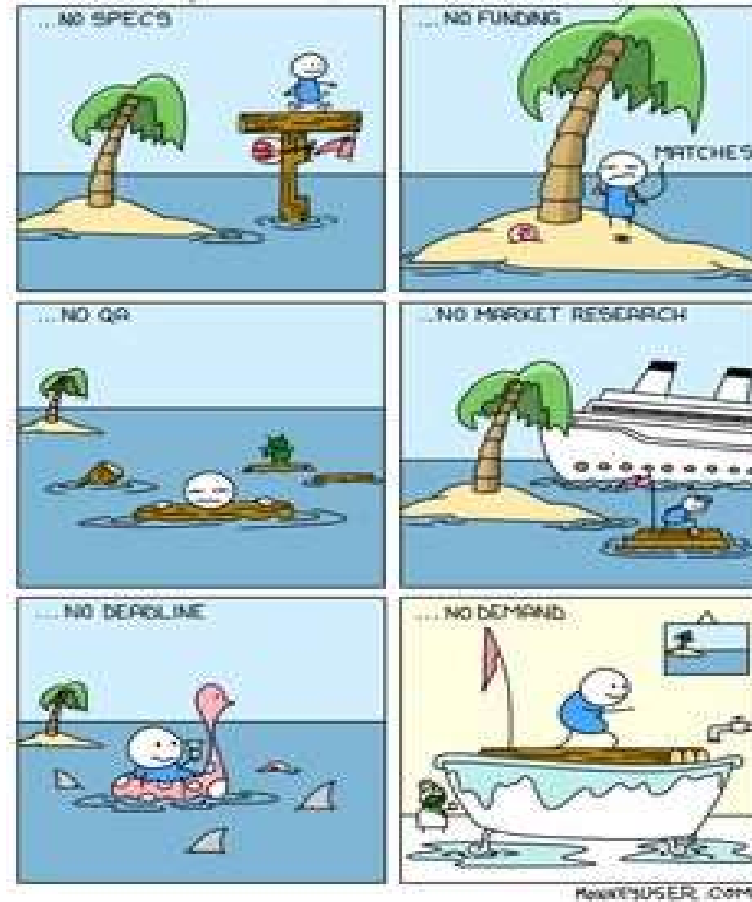


# Software Development Activities

- Software development always includes the following activities (to some extent):
  - i. Requirements analysis
  - ii. Design
  - iii. Construction(Coding)
  - iv. Testing
  - v. Maintenance
- These activities do not follow strictly one after another!  
Often overlap and interact



### BUILDING (A RAFT) WITH



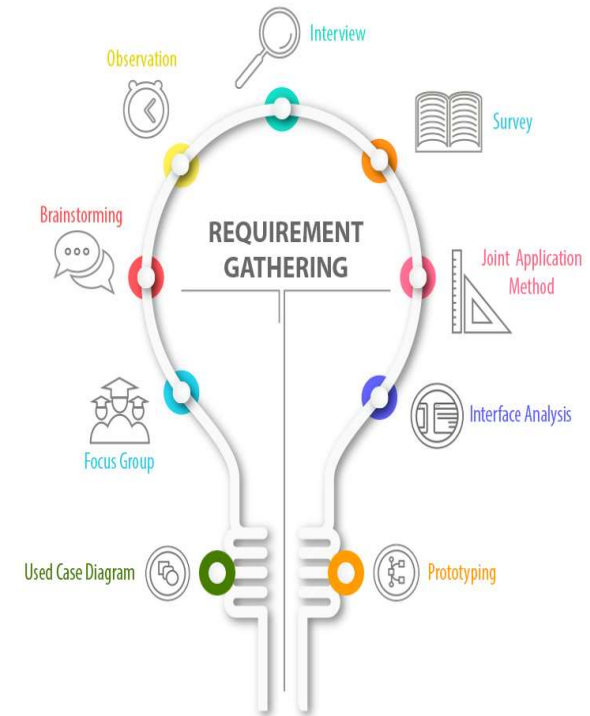


# Software Requirements

- defines the functionality of the system
  - Answer the question "what?", not "how?"
  - Define constraints on the system

Two kinds of requirements

- Functional* requirements
- Non-functional* requirements



Source: NMG Technologies.com



## Requirements Analysis

- *Requirements analysis* starts from a vision about the system
- Customers don't know what they need!
- Requirements come roughly and are specified and extended iteratively
- *Prototyping* is often used, especially for the user interface
- The outcome is the Software Requirements Specification (SRS)

*How stakeholders think requirement gathering works.*

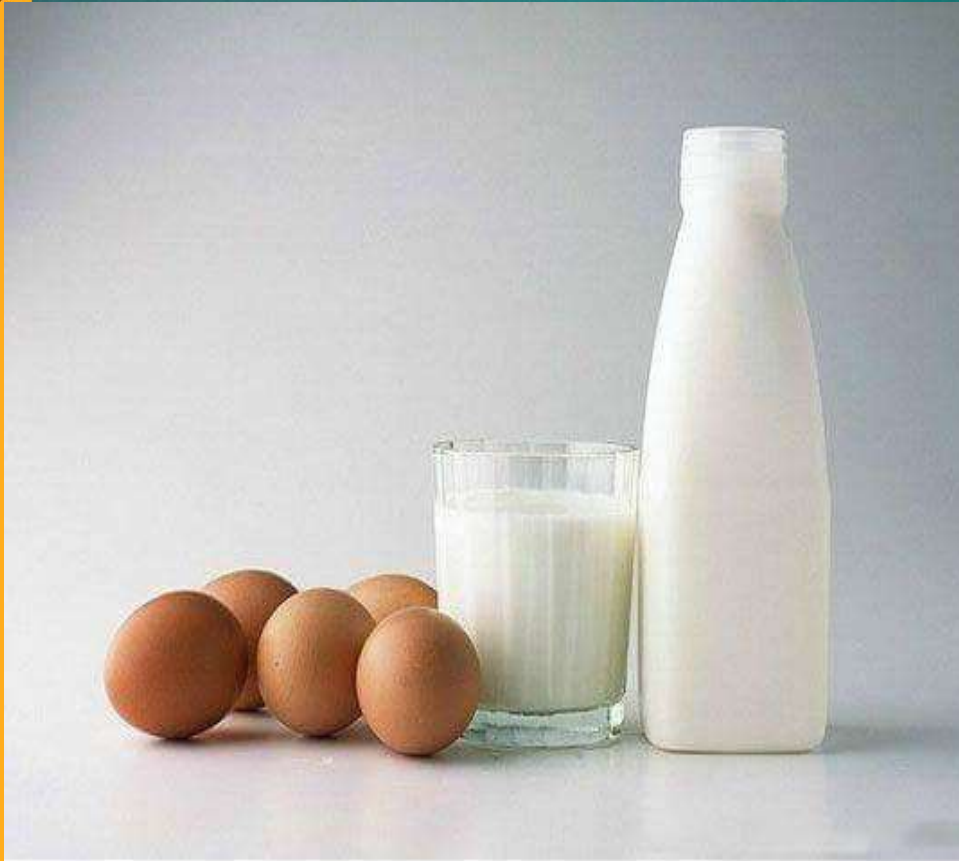


*How requirement gathering really works.*





## (Mis)understanding the Requirements



### BEING A PROGRAMMER

My mom said:

"Honey, please go to the market and buy 1 bottle of milk. If they have eggs, bring 6"

I came back with 6 bottles of milk.

She said: "Why the hell did you buy 6 bottles of milk?"

I said: "BECAUSE THEY HAD EGGS!!!!"





## Software Design

*Software design* is a technical description about how the system will implement the requirements

- The *system architecture* describes:
  - How the system will be decomposed into subsystems (modules)
  - Responsibilities of each module
  - Interaction between modules
  - Platforms and technologies



When your client asks



...if you can do it cheaper?



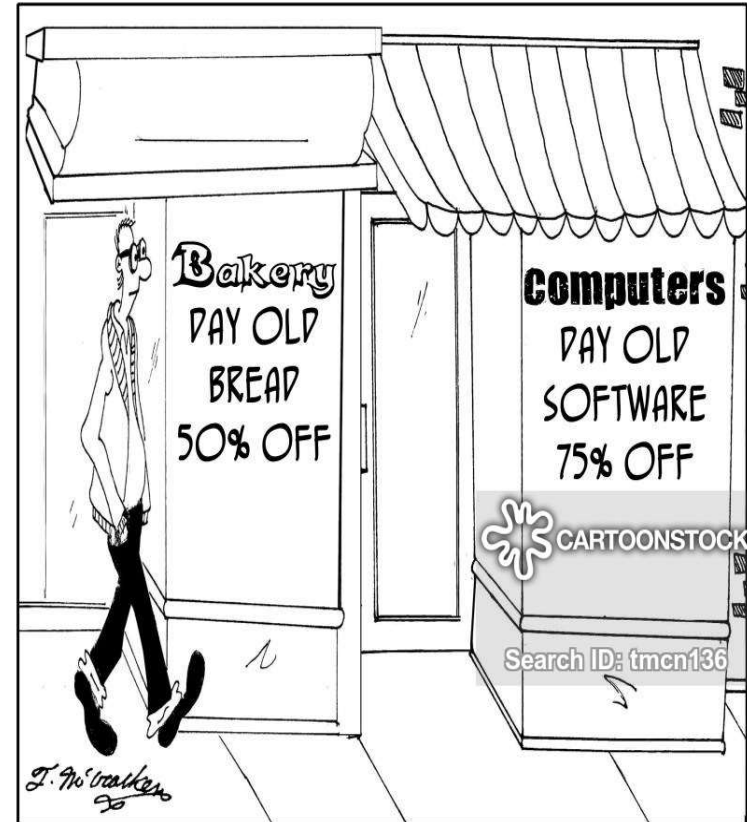
# Software Construction

- During the *software construction* phase developers create the software

(*implementation* phase)

It includes:

- Internal method design
- Writing code
- Writing unit tests (sometimes)
- Testing and debugging
- Integration





# Testing



- *Testing* checks whether the developed software conforms to the requirements

Aims to identify defects (bugs)

Developers test the code after write it

At least run it to see the results

- *Unit testing* is even better
- Units tests can be repeated many times by developers
- *Integration testing* (Units integrated)
- *System testing* is done by QA engineers





# Debugging



*Debugging* aims to find the source of already identified defect and to fix it

- Performed by developers

## Steps in debugging:

- Find the defect in the code
- Identify the source of the problem
- Identify the exact place in code causing it
- Fix the defect
- Test to check if the fix is correct



## Coding != Software Engineering



- Inexperienced developers consider coding the core of development
- In most projects **coding is only 20% of the project activities!**
- The **important decisions** are taken during the **requirements analysis and design**
- Documentation, testing, integration, maintenance, etc. are often disparaged
- Software engineering is not just coding!

*Programmer != software engineer*



## Assessment



1. Requirements phase defines the functionality of the system (True / False)
2. What are the 2 main types of requirements?
3. \_\_\_\_\_ answer the question "what?", not "how?".
4. Software Construction phase is also called as \_\_\_\_\_ phase
5. Outcome of requirements analysis phase is \_\_\_\_\_
  - a. Software Requirements Specification
  - b. Requirements Document
  - c. Analysis Report
  - d. Software Requirements Report



# Thanks!

Any questions?

You can also find me at:  
[nithyasnsct@gmail.com](mailto:nithyasnsct@gmail.com)