

DEVELOPING SOFTWARE APPS for MOBILE DEVICES

Session 1

Prof. Barry Dwolatzky

Director: Joburg Centre for Software Engineering (JCSE)

Wits University



Introductions

- My Name
- Experience in software development
- Something interesting about what I do
- My expectations for this course



Introduction to mobile app development

- How does mobile app development differ from other forms of software development?



The Software Development Process

- Phases

Plan

Estimate duration and cost for the project and develop a plan of action

Design

Make decisions about how the software will be built and tested

Code

Write the software in some programming language

Compile

Use a compiler to translate into executable code. Fix syntax errors until a “clean compile” is achieved.

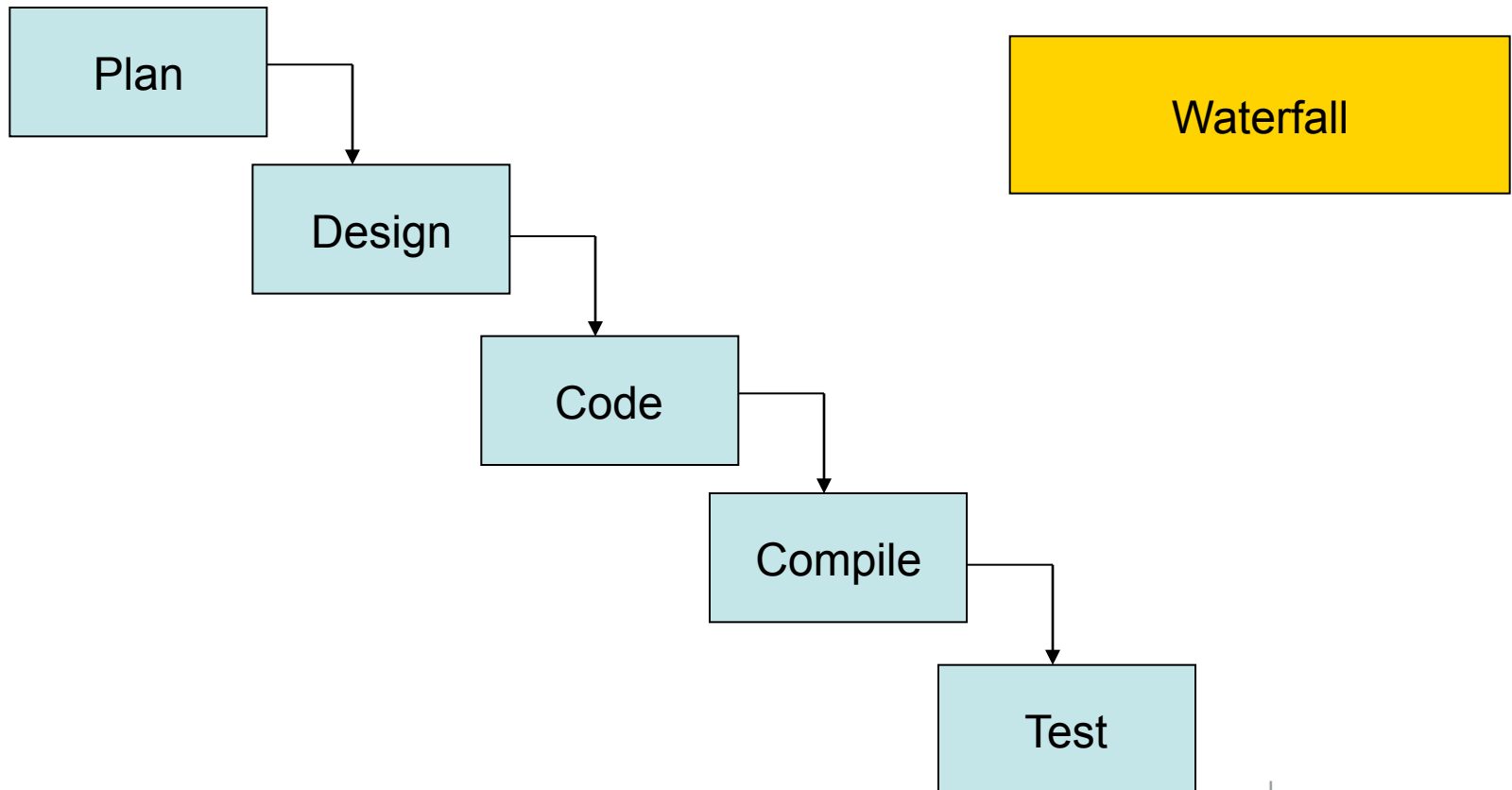
Test

Ensure that software operates as specified and meets original requirements. Rework to fix errors.



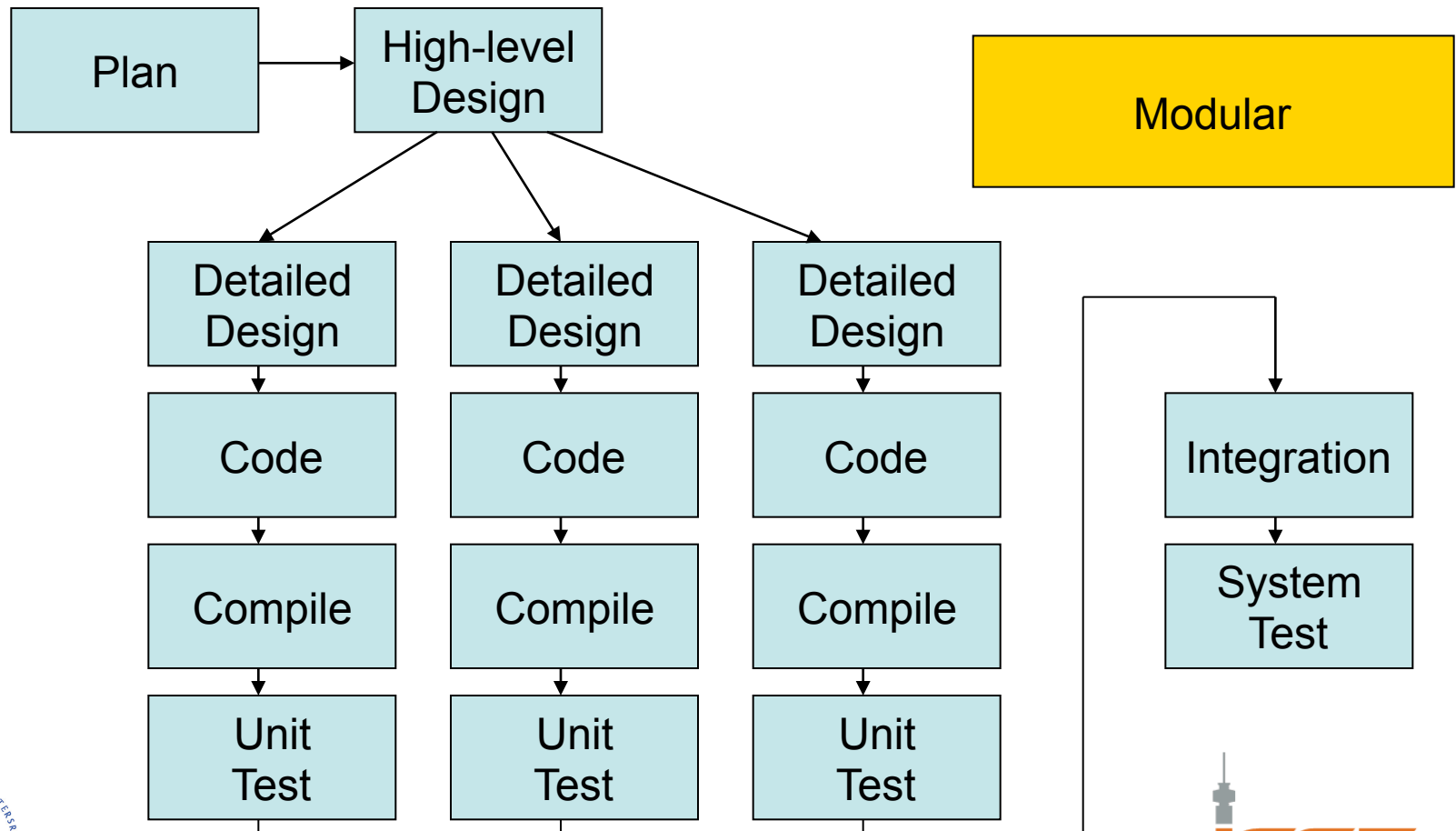
The Software Development Process

- Sequence



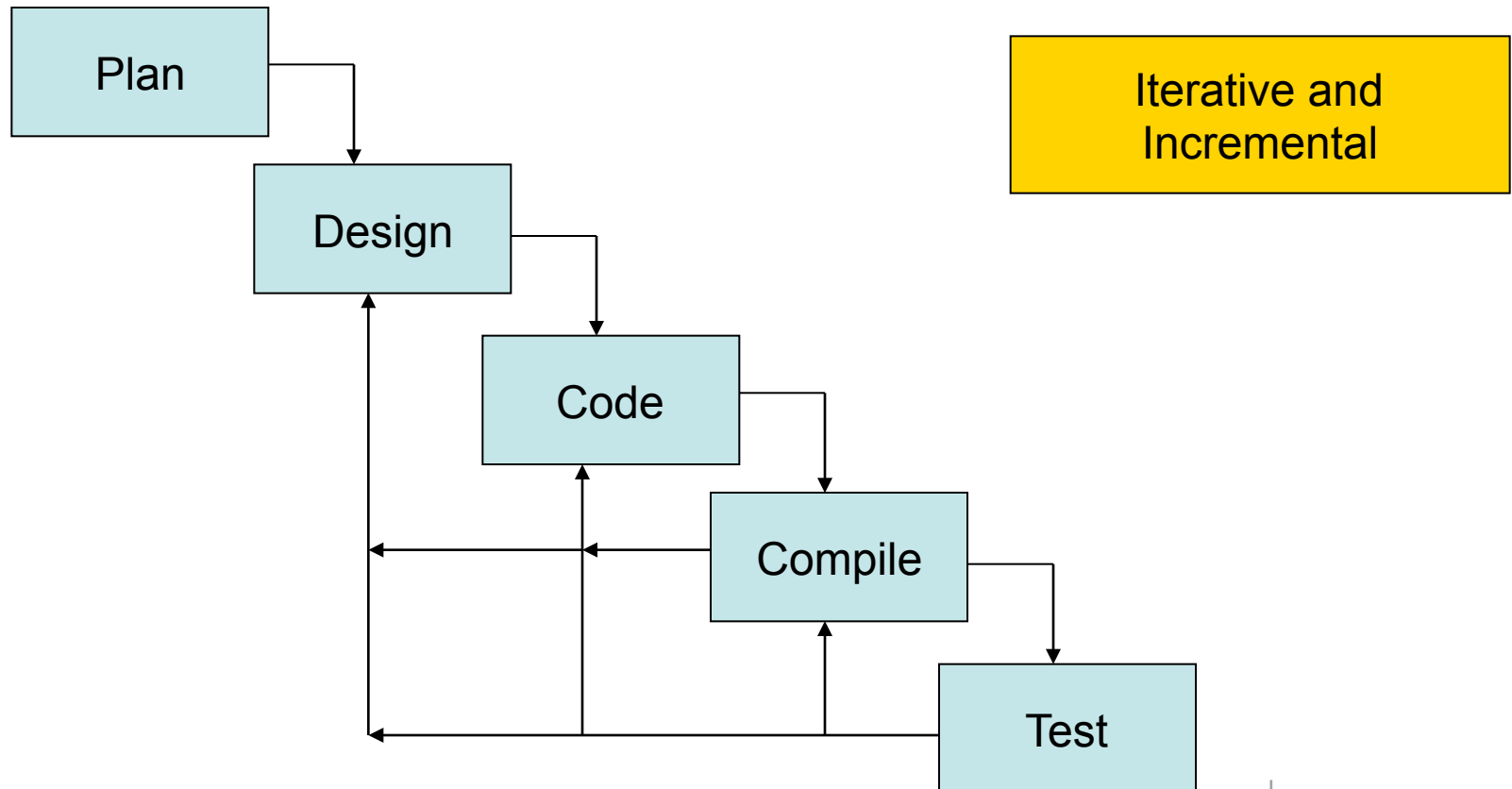
The Software Development Process

- Sequence



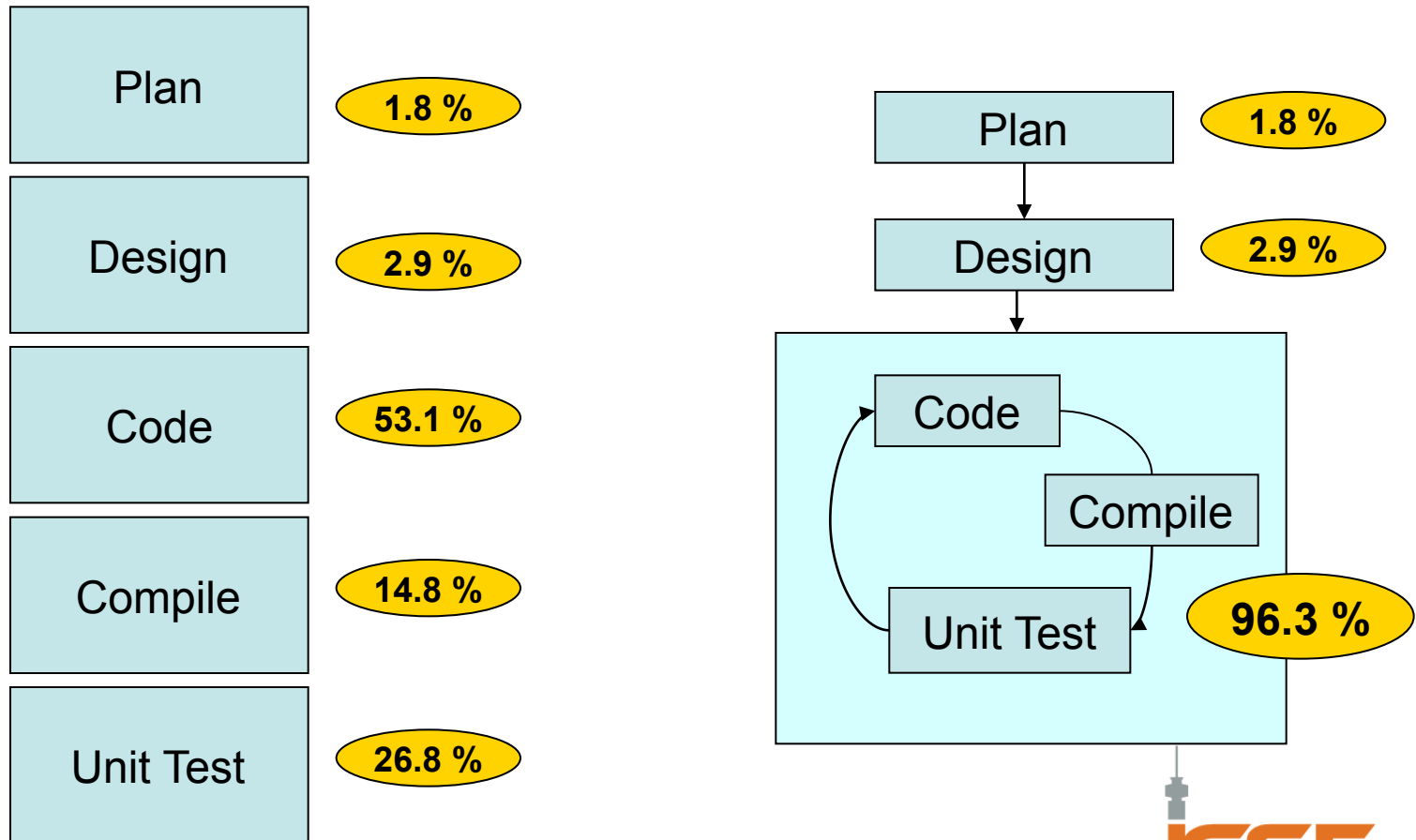
The Software Development Process

- Sequence



Software Development Process

- Barry Dwolatzky October 2009



Some key issues in (all) software development

- The hardest part about software development is knowing WHAT to build
 - Requirements Engineering is crucial
- The role of architecture and design
 - The power of abstraction
- Maintainability & Reusability as the key “ilities”
- Software testing issues
- The importance of planning



Achieving High Software Quality

Testing vs. Inspection



Defects and Software Quality

- Software Quality = Meeting the user's needs
- There are different levels of software quality:
 - Firstly there is a quality threshold – it is essential that the product works. Without this any other quality attributes are meaningless;
 - Beyond this threshold the user has other requirements (e.g., safety, security, privacy, usability)
- It is defects that prevent software from functioning. It is also defects that result in software not being completely safe or secure.



Defect Injection and Removal

- Programs are complex products.
 - Small programs have thousands of instructions.
 - Large programs have millions of instructions.
 - These instructions are individually produced.
 - Each instruction must be precisely correct.
- Any simple mistake can cause the program not to work.
- When people do complex tasks, they inevitably make mistakes.
- On average, experienced programmers inject a defect about every 10 instructions.
- Current practices rely heavily on testing to remove these defects

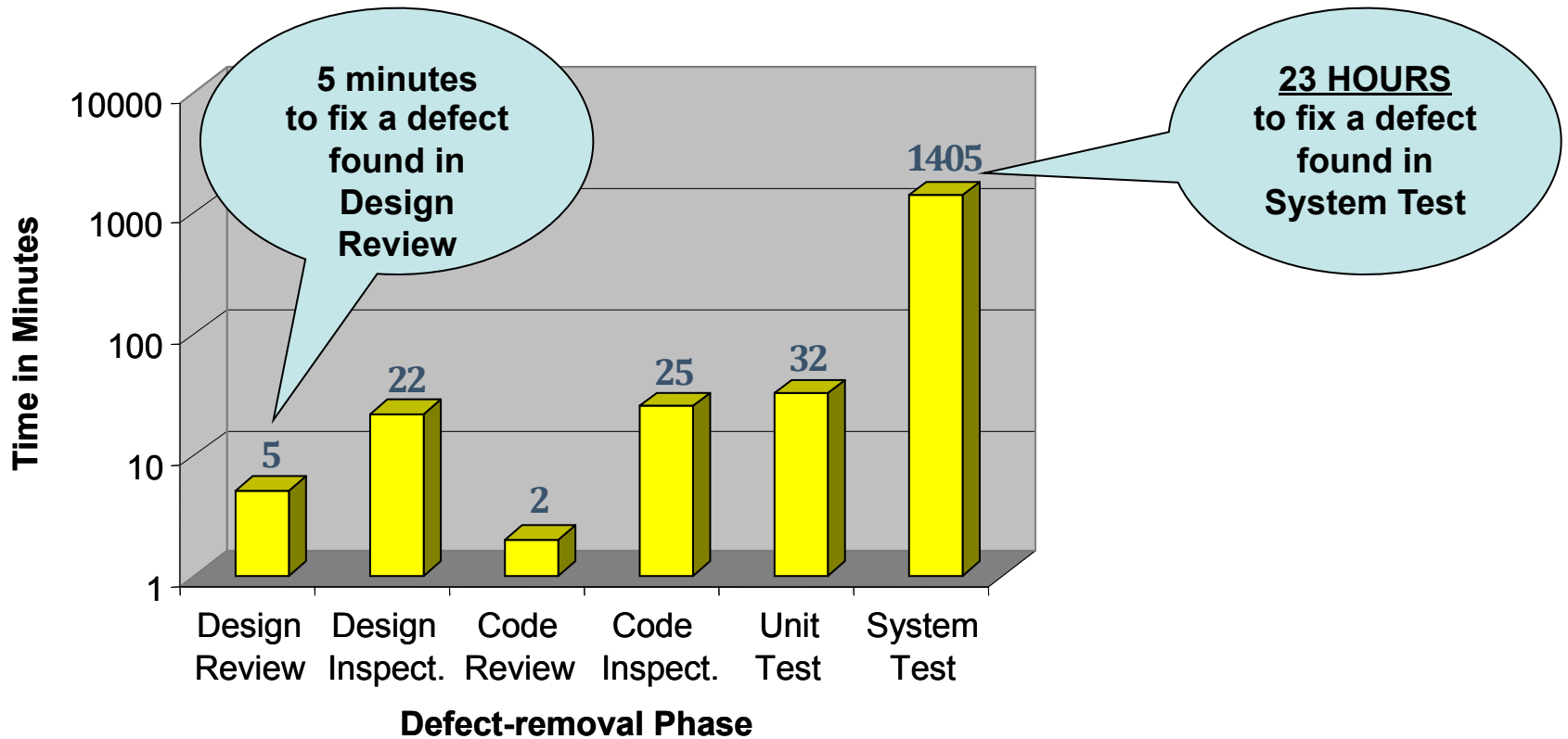


The Economics of Software Quality

- Since each new test exposes new defects it is hard to know when to stop testing
- Since each test costs money and takes time the quality of software becomes an economic decision
- The cost of finding and fixing defects increases sharply in later life-cycle phases



Average Defect Removal Times



Source: Xerox



Defect Injection and Removal

- A single test
 - exercises the product under one set of conditions
 - produces correct or incorrect results
- If there is a problem, developers must find the defect, fix it, and then test the fix.
- For products with many possible operating conditions, many tests are required.
- Projects that rely on testing for quality spend a lot of time and money on testing.

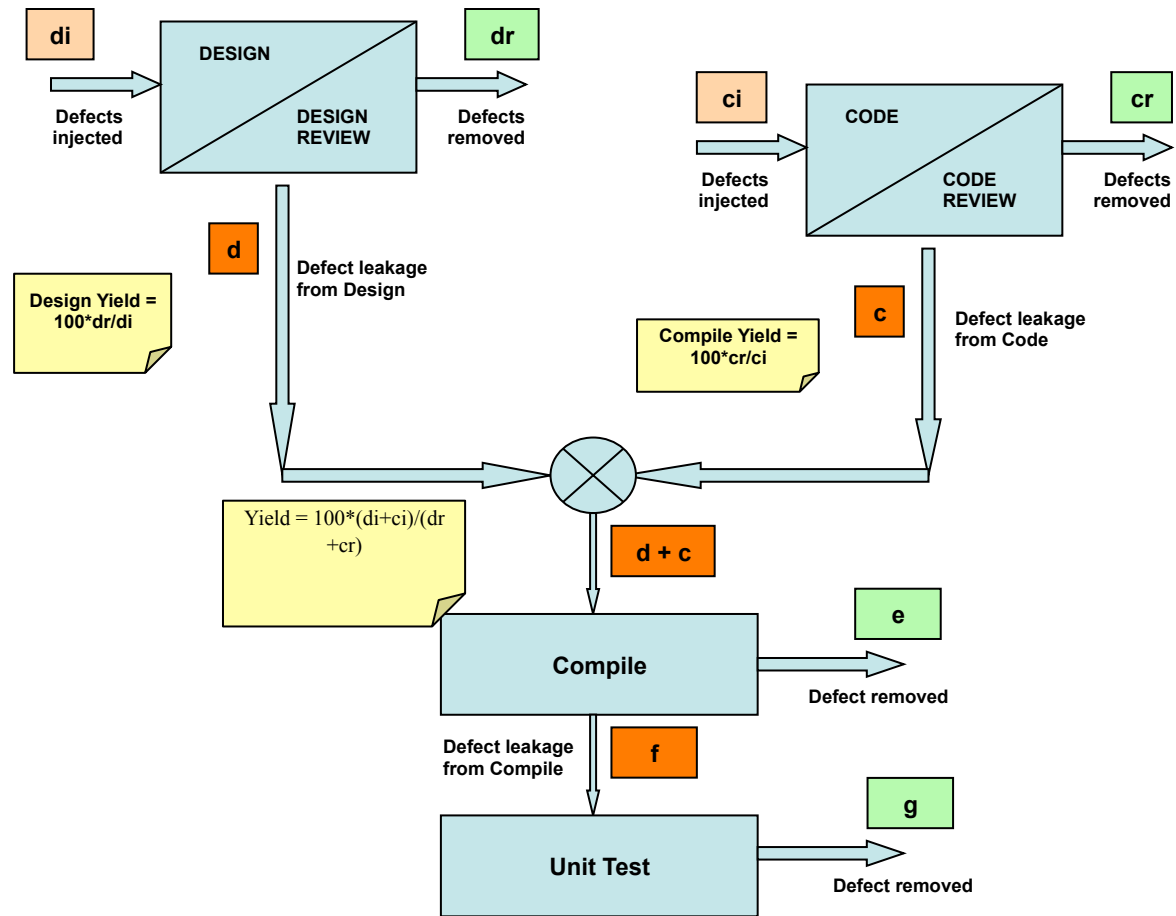


Some Important Principles

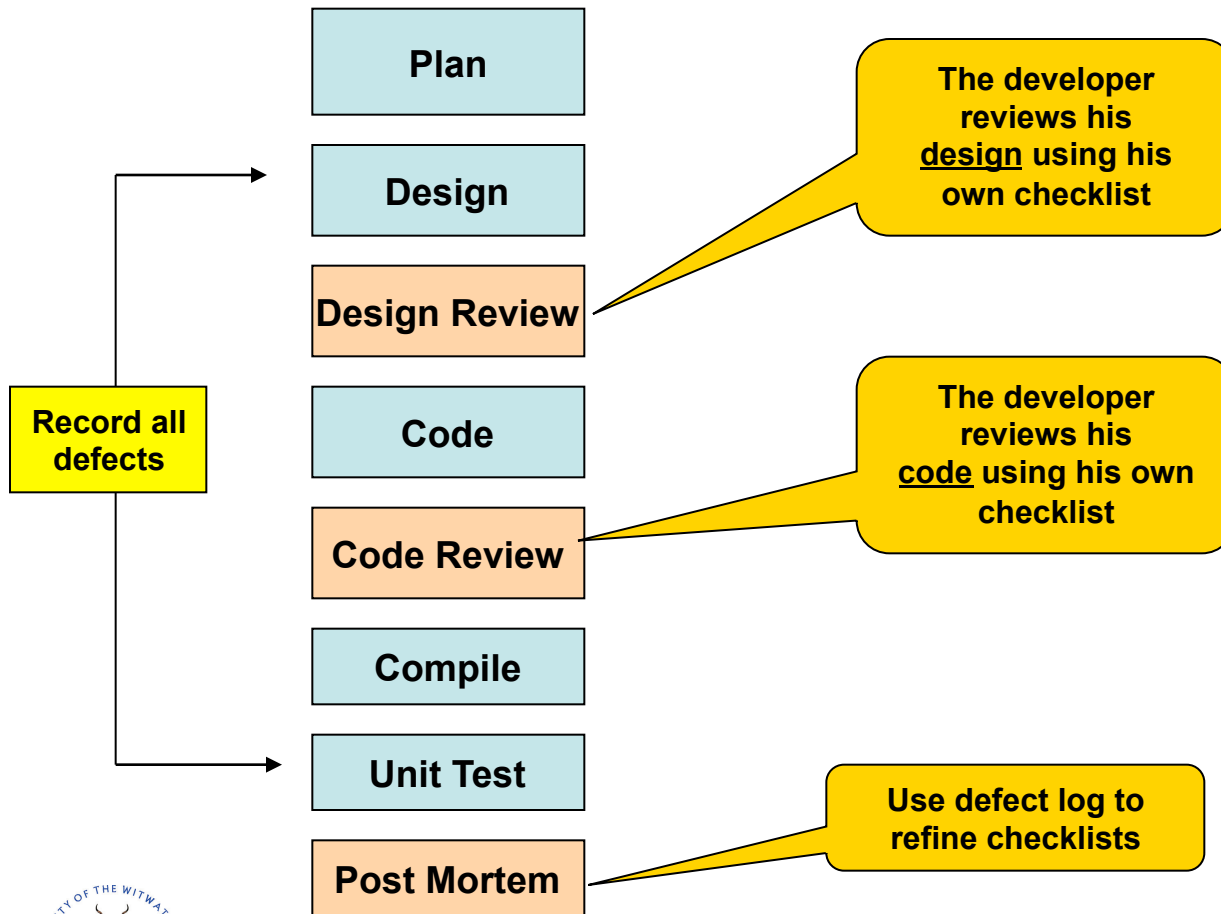
- A heavy reliance in testing as a way to achieving quality is:
 - Inefficient
 - Time-consuming, and
 - Unpredictable
- Because defects result from errors by individuals one should manage personal behaviour (i.e. process) to effectively manage software quality
- The person who introduced the defects is the best person to find and fix them
- The sooner one finds and removes the defect the better



The Personal Software Process (PSP)



PSP Software Development Process



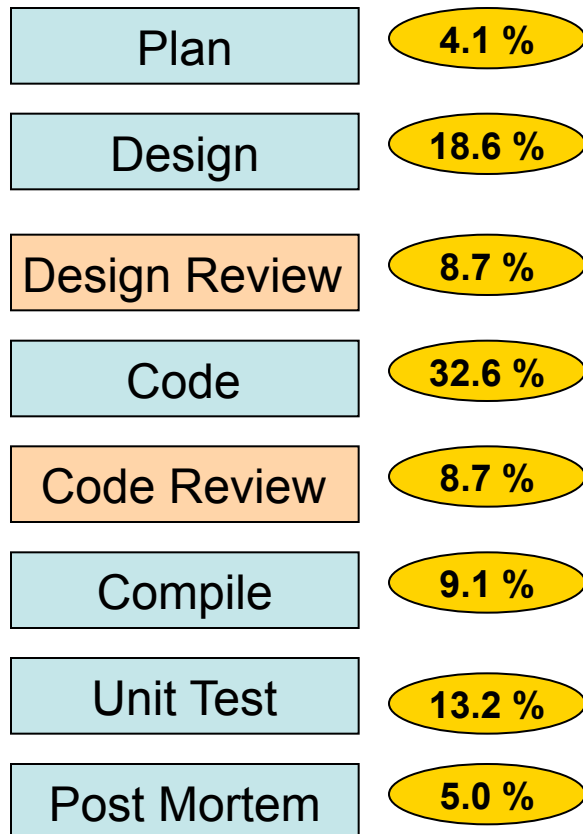
PSP Software Development Process

- Time in each phase and defects are logged
- Aim to have:
 - design time = coding time
 - design review time = $\frac{1}{2}$ design time
 - code review time = $\frac{1}{2}$ coding time
 - Yield (% of defects removed before compile + test) as close to 100% as possible

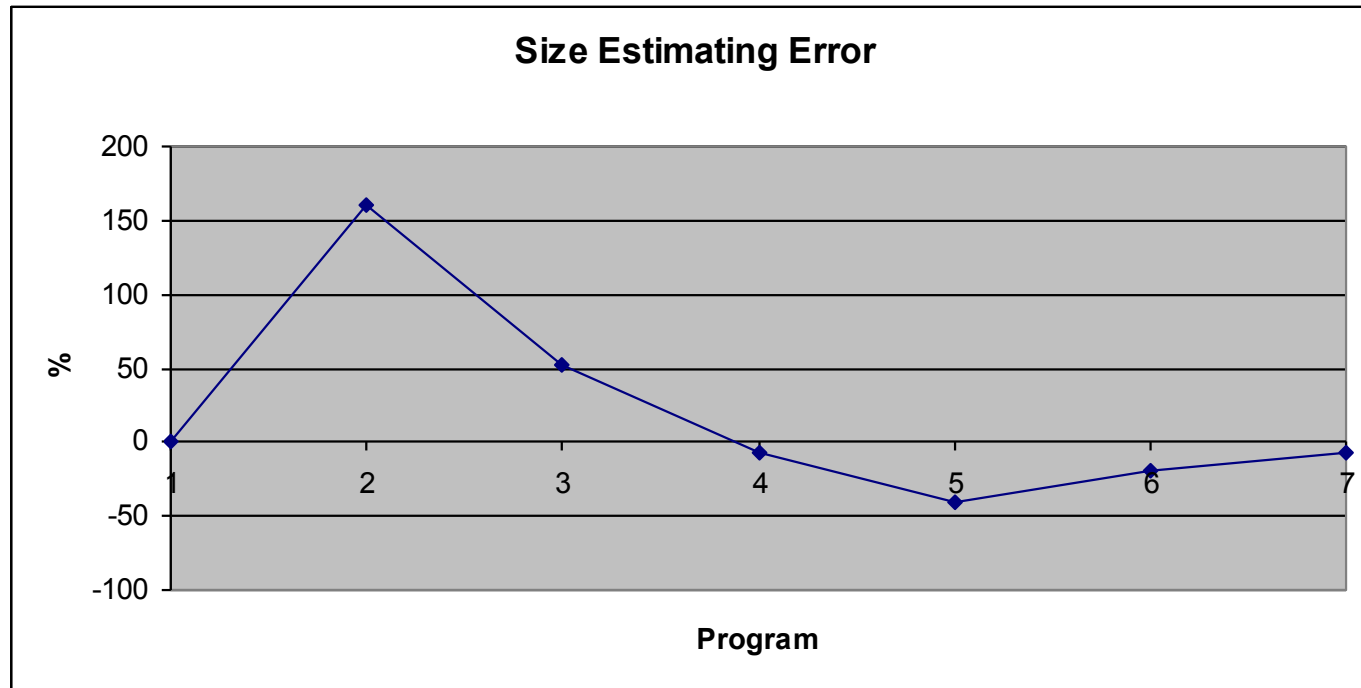


PSP: Barry Dwolatzky

December 2009 ...



PSP: Barry Dwolatzky



Team Software Process (TSP)

- An Agile approach in which a team of PSP developers work together with a coach
- Introduce Design and Code Inspections as a way of increasing team Yield
- The higher the quality of code taken into System Test, the more predictable and efficient testing is



So – what about developing mobile apps?

- 8 things to consider (www.getelastic.com/want-to-build-mobile-app)

① Which Platform?

- iPhone/iPad
- Symbian
- Android
- Windows Mobile
- Blackberry
- Others?

Consider your customers.



8 things to consider (continued)

② One step at a time

- Build for one platform first, release it and measure your success.

④ Who will build it?

- In our case its you 😊
- If you were to outsource an app development in the USA it would cost \$20K to \$400



8 things to consider (continued)

④ Don't rush

- Don't get pressured by the release of competitor apps
- If you mess up on your first release you may lose some customers forever



8 things to consider (continued)

- ⑤ Make the experience consistent with your web store
 - Your customers have been users of your web store for years (months?)
 - Make sure that they experience your app in the same way that they've experienced others
- ⑦ Think about integration with existing e-Commerce platforms



8 things to consider (continued)

⑦ Have a roadmap

- Create a roadmap that clearly identifies and prioritizes future functionality
- Plan to release new versions approximately every 3 months

8 things to consider (continued)

⑧ Measure

- Make sure that at minimum you're tracking the following analytics
 - Number of app downloads and installations
 - Number of app removals
 - Number of app starts
 - Average usage time
 - Where the app is being used (GPS data)

APIs exist to support the collection of these analytics

