SDLC - Review

Class 6 - 9

Introduction

- Software applications can be complex
- Our aim is to build **successful** software products
- Large software projects are challenging
- Ad hoc software development can result in failures
- Engineering approach is essential

Engineering Approach for Software

- Estimate the cost and effort involved
- Should plan and schedule the work.
- Should involve users in defining requirements, what exactly is expected from the software.
- Should identify the stages in the development.
- Should define clear milestones
- Should track the progress and have alternatives
- Should have clear quantitative analysis of whether the project is successful or not

Software Process

- Process defines a set of steps
- These steps need to be carried out in a particular order
- Different types of processes in a software domain
  - process for **software development**
  - Process for managing the project
  - Process for change and configuration management
  - Process for managing the above processes

Step in a Software Process

- Well defined objective
- Well defined inputs and outputs
- Why/What/How/When

- Entry and Exit criteria

Software Development Process

- Software Development Life Cycle.
- SDLC is the process which helps to develop good quality software products
- SDLC is composed of a number of clearly defined and distinct work steps or phases
- A number of SDLC models or process models have been created such as Waterfall, Spiral, Agile etc.

Software Development Life Cycle

- Problem Definition
- Feasibility Study
- Requirements Analysis
- Design
- Implementation
- Testing
- Maintenance
- Archival

Problem Definition

What is the problem?

- Ensure there exists a problem to be solved
- Define goal
- Usually short and quick
- Categorizing problem
- Avoid misunderstanding
- Identifying cause of the problem
- Checking cost-effectiveness

Problem Definition Document

- Problem statement
- Project objective
- Preliminary Ideas
- Time and Cost for Feasibility Study

Deliverable 1A

Next --- Feasibility Study

Try to answer:

- Are there feasible solutions?
- Is the problem worth solving?

Look at cost-benefit analysis;
even for sorting, for example.

Types of Feasibility Study

- Economical
- Technical
- Operational

Cost – Benefit Analysis

- Types of costs
- Types of benefits
- Estimation in early stage; challenging

The SW provider should not make the decision.

Feasibility Report

- A brief statement of the problem; System environment
- Important findings and recommendations
- Alternatives
- System description
- Cost-benefit analysis
- Evaluation of technical risk
- Legal consequences

Deliverable 1B

Software Development Life Cycle

- Problem Definition
- Feasibility Study
- Requirements Analysis
- Design
- Implementation
- Testing
- Maintenance
- Archival

Requirements Analysis

- Knowing user's requirement in detail
- Objective is to determine what the system must do to solve the problem (without describing how)
- Produces SRS document
- Incorrect, incomplete, inconsistent, ambiguous SRS often results in project failure

Requirement Analysis

- Challenging
  - Users may not know exactly what is needed
  - Users may change their mind over time
  - Users may have conflicting demands
  - Analyst has no or limited domain knowledge
  - Client may be different from the user
  - Users may not be capable to differentiate between what is possible and what is impractical

SRS

- First and most important baseline
- What the system will be able to do
- **Basis for validation and final acceptance**
- Cost increases rapidly after this step
- Should be reviewed in detail by user and other analyst
- Should be adequately detailed
- It identifies all functional and performance requirements
- Decides the Acceptance Criteria too.

Requirements Analysis Process

- Interviewing clients
- Studying existing things
- Long process – should be organized systematically
- Identifies users and business entities
- Get functional or domain knowledge

Organizing Findings

- Massive amount of information through study
- Need to be organized, recorded and classified
- Ensure consistency and completeness
- Prepare SRS
- Get it reviewed
- Get it accepted

Deliverable 2

Software Development Life Cycle

- Problem Definition
- Feasibility Study
- Requirements Analysis
- Design
- Implementation
- Testing
- Maintenance
- Retirement

Design

- Deals with “ How “
- Consider several technical alternatives
- Input is the SRS
- May follow “design techniques” / “design paradigms”
- Need to study “design patterns”
- Prepare for technical management review
- Deliverable : design document

Why so many plans?

Requirements - Based on ?

Design - Studio / 1 / 2 / 3 BHK

Now, why you, a novice, should push your stupid plans?

Design Goals

- Processing component
  - Business Rules
  - Algorithms
- Data component
  - Database Design
- Different design paradigms
- System structure [Software Architecture]
  - Decomposes the complex system
  - Defines the subsystems or modules

Deliverable 3

Implementation

- Coding are done
- Translating design specification into the source code
- **Source code along with internal documentation**
- To reduce the cost of later phases
- Making the program more readable
- General coding standards
- Walk-through & Review

Deliverable 4

Testing

- Testing is important because software bugs could be expensive or even dangerous.
- Process of evaluating whether the current software product meets the requirements or not.
- Checks for missing requirements, bugs or errors, security, reliability and performance
- Many types as it is a very important phase in practical software development - black/white/gray - unit/integration/regression/UAT

Who knows/decides what to test?

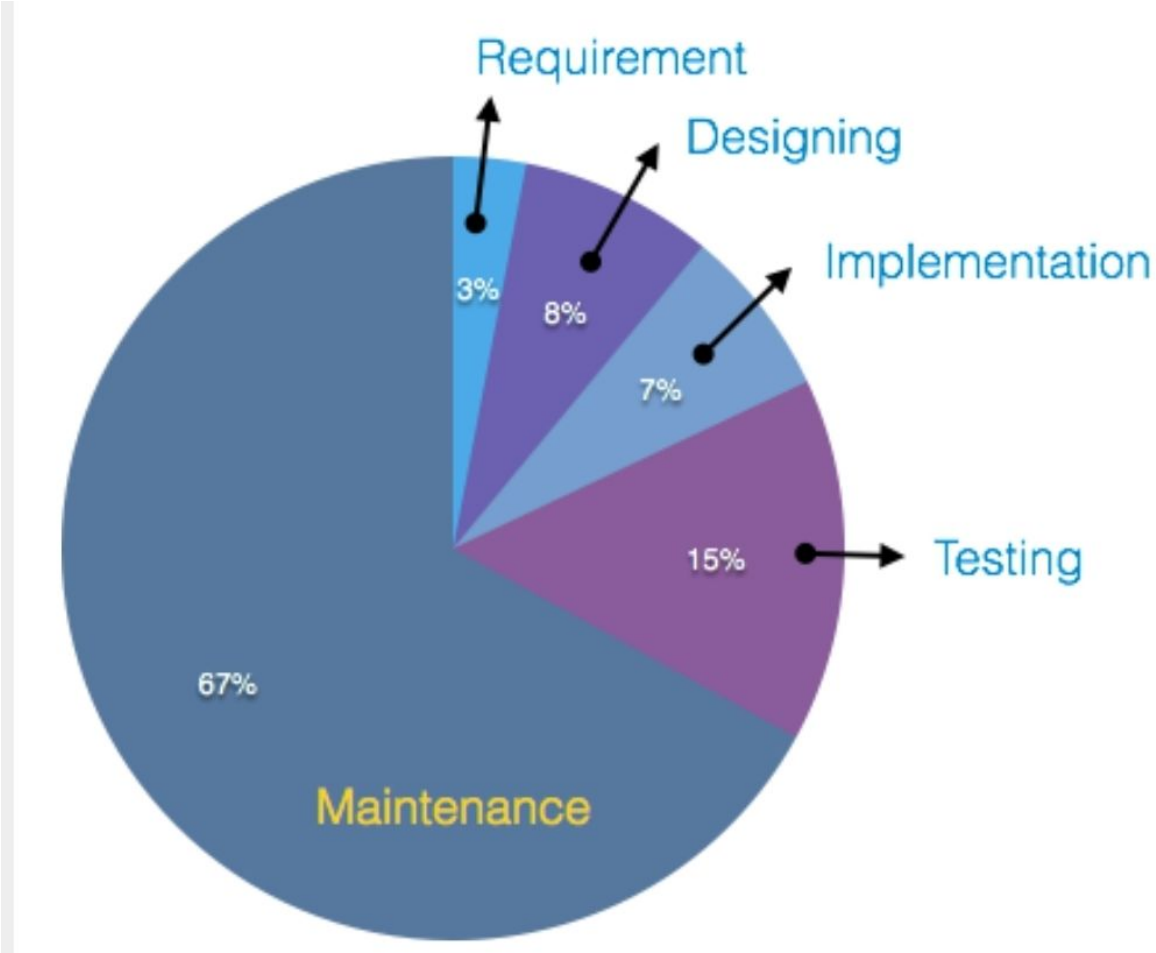
How to test effectively? [Load Testing, Integrations]

Deliverable? [Test Plan, Test Report etc.]

Maintenance

- Goal is to modify and update software after delivery
  - Correcting errors
  - Improving performance or capabilities
  - Deletion of obsolete features
  - Optimization
- Types of Software Maintenance
  - Corrective
  - Adaptive
  - Preventive
  - Perfective

Cost Comparison over Phases



Software Retirement Process

- Application Decommission or Application sunsetting
- Final stage of life cycle
- Shutting down
- **Archival**

- Reasons
  - Replaced
  - Release no longer supported
  - Redundant
  - Obsolete

What about?

- Estimate the cost and effort involved
- Should plan and schedule the work.
- Should involve users in defining requirements, what exactly is expected from the software.
- Should identify the stages in the development.
- Should define clear milestones
- Should track the progress and have alternatives
- Should have clear quantitative analysis of whether the project is successful or not

Deliverable - PROJECT PLAN?

Remember!

Firms need to have business (clients)

Firms decide processes

Firms decide tools / software to use

Firms make decisions - some are wrong

Firms meet failures - some learn from failures

Engineers make / break the firms