

# Software Development Lifecycle

Hovag Abramian



# Session III

## Outline

We are going to learn about:

- Software Development Lifecycle
- Software Project Management
- Related Tools

## Learning Objectives

At the end of the session, you will be able to:

- Identify the stages of SDLC
- Categorize processes and artifacts relevant to each stage
- Define the meanings of common terminology

# Software Development Lifecycle

- Defines steps and responsibilities
- The steps can overlap
- The steps might need revisiting

# Software Development Methodologies

- Evolution and Necessity
  - No methodology - Cowboy coding
  - Traditional methodology - Waterfall
  - Agile methodologies - Scrum, Kanban
- Artifacts

# Waterfall

- Concept

- Requirements
- Design
- Implementation
- Integration
- Maintenance

## Feasibility Analysis

Can it be done?

Are the resources available?

Should it be done?

Buying vs. in-house

# Waterfall

- Concept
- Requirements
- Design
- Implementation
- Integration
- Maintenance

Functional Requirements

Technical Requirements

Use Cases

Terms of Reference

# Waterfall

- Concept
- Requirements
- Design
- Implementation
- Integration
- Maintenance

Architecture

Hierarchy

Diagrams

Design Level Use Cases

# Waterfall

- Concept
- Requirements
- Design
- Implementation
- Integration
- Maintenance

Coding

Unit Testing



# Waterfall

- Concept
- Requirements
- Design
- Implementation
- Integration
- Maintenance

Integration Tests

Acceptance Tests

Load Test

Phased Deployment

A/B Testing

# Waterfall

- Concept
- Requirements
- Design
- Implementation
- Integration
- Maintenance

Feedback

Feature Requests

Bug Reports

# Waterfall Methodology

- Best Practices
- Example Projects
- Strengths
- Drawbacks
  - Changing requirements!
- Conclusion: Waterfall remains relevant

# Agile Methodologies - Scrum

- Requirements
- Inside-out vs Linear
- Sprint
- Stand-up meetings
  - Reflection and feedback
  - Planning
  - Obstacles
- Scrum master

# Testing

- Unit testing
- Integration Testing
- Subsystem Testing
- System Testing
- Regression Testing
- Alpha Testing
- Beta Testing
- Acceptance Testing

# Terminology

- Development Environment
- Development Server
- Test Environment
- Code Freeze
- Staging
- Production
- Deployment: Rollout, Rollback

# Lean Methodologies - Kanban

- Kanban is about tasks, scrum is about deliverables
- Philosophy
  - Continuous flow
  - In progress -> Done
  - Waste Reduction
- Kanban Board
- Ticketing Systems

# Lean Methodologies - Kanban

- Visualize the Workflow
- Limit your Work in Progress (WIP)
- Manage the Flow
- Implement Feedback Loops
- Make Process Policies Explicit
- Improve Collaboratively



# Tools

- Jira
- Git
- Microsoft Project

# References

1. Bennatan, E.M., (2000) "*On Time, Within Budget: Software Project Management Practices and Techniques*", Wiley