Introduction to Agile

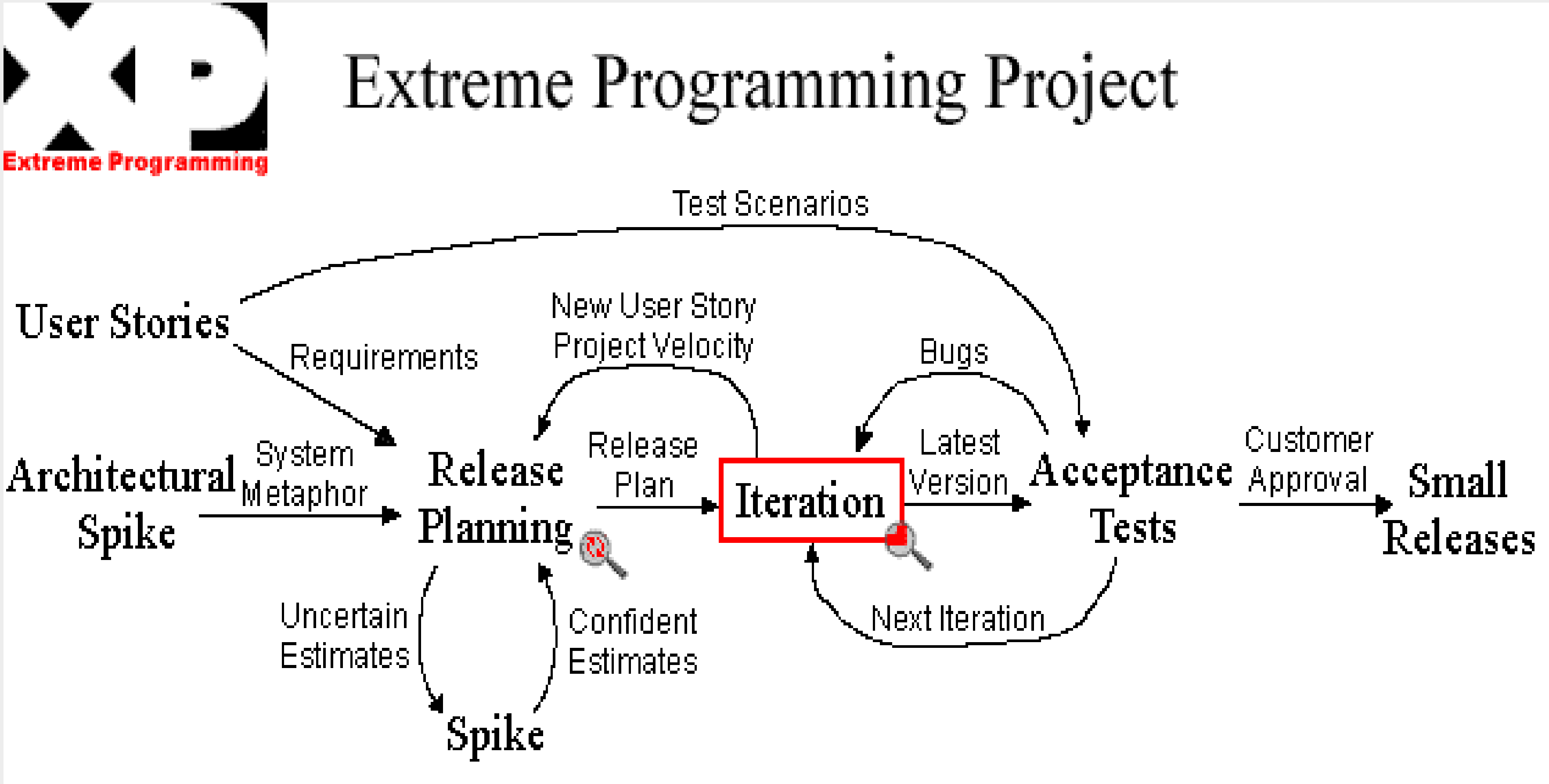
Lesson 3: Agile Methods and Practices - Extreme Programming (XP),Lean Software Development & Kanban

Lesson Objectives

- Introduction to Extreme Programming
- The Rules of Extreme Programming
- Extreme Programming (XP) - Principles
- Extreme Programming (XP) – Key Terms
- Introduction to Lean Software Development
- Principles of Lean Software Development
- What is Kanban?

3.1: Agile Methods and Practices – Extreme Programming (XP)

Introduction to Extreme Programming



3.1: Agile Methods and Practices – Extreme Programming (XP)

Introduction to Extreme Programming (Cont.)

- Extreme programming (XP) was created by Kent Beck where he applied a light weight methodology to deliver a financial system in 2 years which previously had been undelivered over a number of years with a team of 30 people

Wikipedia definition:

Extreme Programming (XP) is a software development methodology which is intended to improve software quality and responsiveness to changing customer requirements. As a type of agile software development, it advocates frequent "releases" in short development cycles, which is intended to improve productivity and introduce checkpoints where new customer requirements can be adopted.

Introduction to Extreme Programming (Cont.)

www.xprogramming.com definition:

Extreme Programming is a discipline of software development based on values of simplicity, communication, feedback, and courage. It works by bringing the whole team together in the presence of simple practices, with enough feedback to enable the team to see where they are and to tune the practices to their unique situation.

The Rules of Extreme Programming

- Planning
  - User Stories User stories are written
  - Release planning creates the release schedule
  - Make frequent small releases
  - The project is divided into iterations
  - Iteration planning starts each iteration
- Managing
  - Give the team a dedicated open work space
  - Set a sustainable pace
  - A stand up meeting starts each day
  - The Project Velocity is measured
  - Move people around
  - Fix XP when it breaks

The Rules of Extreme Programming (Cont.)

- Coding
  - The customer is always available
  - Code must be written to agreed standards
  - Code the unit test first
  - All production code is pair programmed
  - Only one pair integrates code at a time
  - Set up a dedicated integration computer

- Designing
  - Simplicity
  - Choose a system metaphor
  - Use CRC cards for design sessions
  - Create spike solutions to reduce risk
  - No functionality is added early
  - Refactor whenever and wherever possible

The Rules of Extreme Programming (Cont.)

- Testing
  - All code must have unit tests
  - All code must pass all unit tests before it can be released
  - When a bug is found tests are created.
  - Acceptance tests are run often and the score is published

Extreme Programming (XP) - Principles

- The Planning Game
- Small Releases
- Metaphor
- Simple Design
- Testing
- Refactoring
- Pair Programming
- Collective Ownership
- Continuous Integration
- 40-hour Week
- On-site Customer
- Coding Standards

Extreme Programming (XP) – Key Terms

- User Stories
- Code the unit test first
- Choose a System Metaphor
- Spike
- CRC Cards

Introduction to Lean Software Development

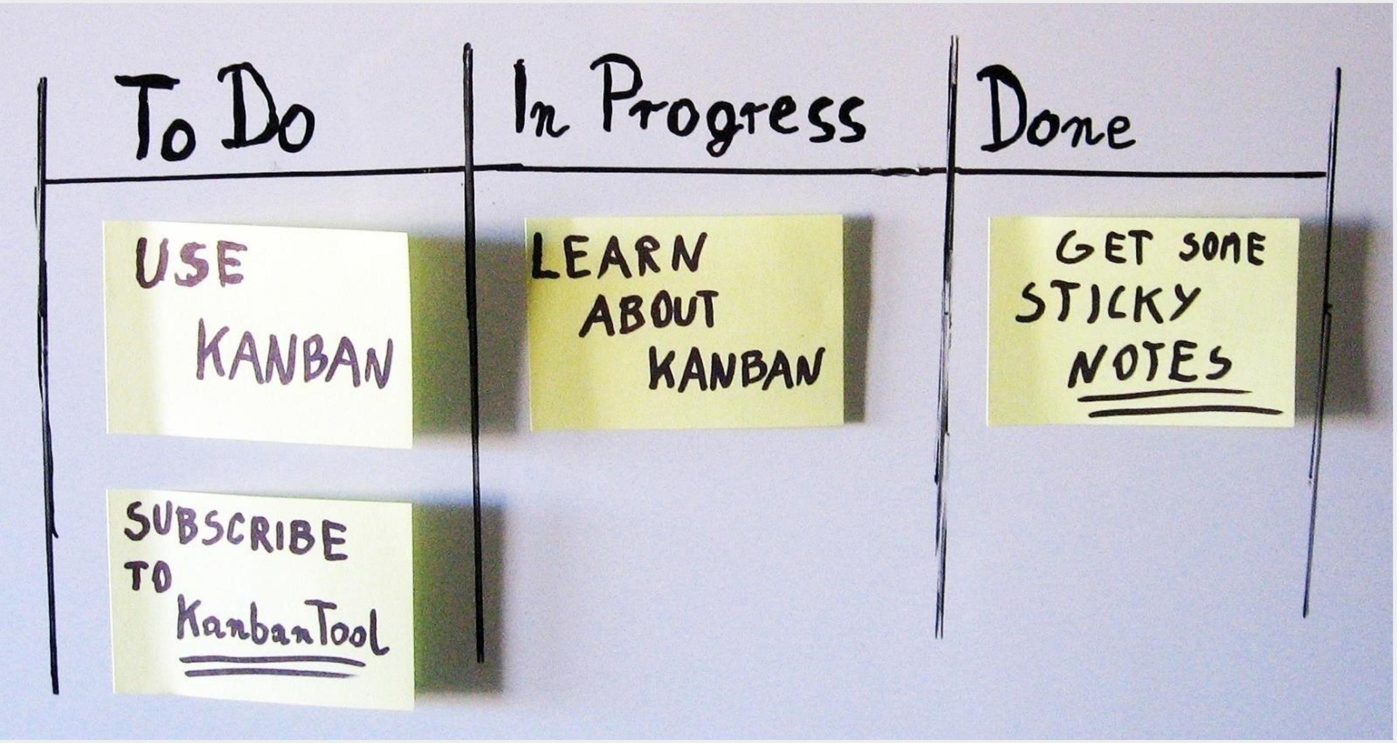
- Lean Software Development is the application of Lean Thinking to the software development process
- Lean Software Development is more strategically focused than other Agile methodology
- The goals are to develop software in one-third the time, with one-third the budget, and with one-third the defect rate
- "Lean Software Development" is not a management or development methodology in itself, but it offers principles that are applicable in any environment to improve software development"

Principles of Lean Software Development

- Eliminate waste: Do only what adds value for a customer, and do it without delay
- Amplify learning: Use frequent iterations and regular releases to provide feedback
- Decide as late as possible: Make decisions at the last responsible moment
- Deliver as fast as possible: The measure of the maturity of an organization is the speed at which it can repeatedly and reliably respond to customer need
- Empower the team: Assemble an expert workforce, provide technical leadership and delegate the responsibility to the workers
- Build integrity in: Have the disciplines in place to assure that a system will delight customers both upon initial delivery and over the long term
- See the whole: Use measurements and incentives focused on achieving the overall goal

What is Kanban?

- The word Kan means "visual" in Japanese and the word "ban" means "card". So Kanban refers to "visual cards“



3.2: Agile Methods and Practices – Kanban

What is Kanban? (Cont.)

- Kanban is way for teams and organizations to visualize their work, identify and eliminate bottlenecks and achieve dramatic operational improvements in terms of throughput and quality
- Kanban is a method to gradually improve whatever you do – whether software development, IT/ Ops, Staffing, Recruitment, Marketing and Sales
- in fact, almost any business function can benefit from applying Kanban to bring about significant benefits such as reduced lead time, increased throughput and much higher quality of products or services delivered

Summary

- In this lesson, you have learnt
  - An introduction to Extreme Programming
  - Lean Software Development
  - Kanban