

# AGILE SOFTWARE DEVELOPMENT

Presented By: Anusha

# Topics

- Introduction
- Terminology
- The Agile Manifesto
- Some Agile Methodologies
- Summary
- References

# What Is Agile

- Agile --readiness for motion, nimbleness, activity, dexterity in motion
- Agility
  - The ability to both create and respond to change in order to profit in a turbulent business environment
    - Companies need to determine the amount of agility they need to be competitive
- Chaordic
  - Exhibiting properties of both *chaos* and *order*
  - The blend of chaos and order inherent in the external environment and in people themselves, argues against the prevailing wisdom about predictability and planning
  - Things get done because people adapt, not because they slavishly follow processes

# Agile Software Development

- **Agile software development** is a conceptual framework for software engineering that promotes development iterations throughout the life-cycle of the project.
- Software developed during one unit of time is referred to as an iteration, which may last from one to four weeks.
- Agile methods also emphasize working software as the primary measure of progress

# Agile Software Development: Intro

- Characteristics of Agile Software Development
  - Light Weighted methodology
  - Small to medium sized teams
  - vague and/or changing requirements
  - vague and/or changing techniques
  - Simple design
  - Minimal system into production



## **Manifesto for Agile Software Development**

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

**Individuals and interactions** over processes and tools

**Working software** over comprehensive documentation

**Customer collaboration** over contract negotiation

**Responding to change** over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

# Characteristics

- Modularity
- Iterative
- Time-bound
- Incremental
- Convergent
- People-oriented
- Collaborative

# Existing Agile Methods

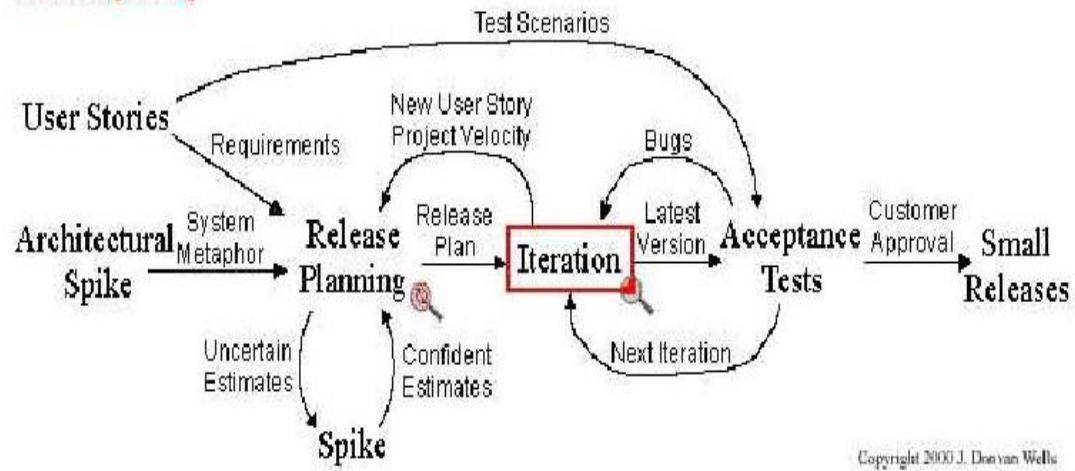
- Extreme Programming (“XP”)
- Agile Unified Process
- Scrum

# Extreme Programming

- Most prominent Agile Software development method
- Prescribes a set of daily stakeholder practices
- “Extreme” levels of practicing leads to more responsive software.
- Changes are more realistic, natural, inescapable.



Extreme Programming Project



Copyright 2000 J. Don van Wells

# Agile Unified Process

- AUP is a simplified version of RUP

## Phases of AUP

- Inception
- Elaboration
- Construction
- Transition

# Disciplines of AUP

- Model
- Implementation
- Test
- Deployment
- Configuration Management
- Project Management
- Environment

# Scrum

- It is an Agile S/w development method for project management

## Characteristics:

- Prioritized work is done.
- Completion of backlog items
- Progress is explained
- Agile Software Development

# Conclusion

- Synthesizes the existing literature.
- Each method is described in terms of process, roles, responsibilities, practices, adoption and experiences.
- Enables a selection criteria for comparing methods and pointing out their differences.

# References

- [1]. Abrahamsson P, Salo O and Ronkainen J. Agile software development methods (Review and analysis).
- [2]. Scott W Ambler. Agile model driven development.
- [3]. Cohen D, Lindvall M, Costa P. Agile software development.
- [4]. [http://en.wikipedia.org/wiki/Agile\\_Modeling](http://en.wikipedia.org/wiki/Agile_Modeling).
- [5]. [http://en.wikipedia.org/wiki/Extreme\\_Programming](http://en.wikipedia.org/wiki/Extreme_Programming).
- [6]. [http://en.wikipedia.org/wiki/Agile\\_Unified\\_process](http://en.wikipedia.org/wiki/Agile_Unified_process).
- [7].  
[http://en.wikipedia.org/wiki/Scrum\\_2&development29](http://en.wikipedia.org/wiki/Scrum_2&development29).