



Objectives

- ✓ Ipad and applications
- ✓ To Understand Agile Methodology
- ✓ Why Agile is so successful ?

Topics covered

- ✓ Rapid Software Development
- ✓ Agile software development
- ✓ Agile SDLC
- ✓ Phases of agile method
- ✓ Manifesto for Agile Software Development
- ✓ Software companies using Agile Methods

Question Answer Session Lecture 3

Audience Q & A
Session/Discussion

Copyright 2011 @ M.Fahad Khan

iPad

The **iPad** is a line of tablet computers designed, developed and marketed by Apple Inc., primarily as a platform for audio-visual media including books, periodicals, movies, music, games, and web content. The iPad was introduced on January 27, 2010 by Apple's then-CEO Steve Jobs. Its size and weight fall between those of contemporary smart phones and laptop computers

Copyright 2011 @ M.Fahad Khan


iPad Functionality

Copyright 2011 @ M.Fahad Khan

Mac iOS 5

http://en.wikipedia.org/wiki/iOS_version_history
Copyright 2011 @ M.Fahad Khan

iPad Applications



Copyright 2011 @ M.Fahad Khan

Rapid Software Development

- Because of rapidly changing business environments, businesses have to respond to new opportunities and competition.
 - This requires software rapid development and delivery is not often the most critical requirement for software systems.
- Businesses may be willing to accept lower quality software if rapid delivery of essential functionality is possible.
- Due to the changing environment
 - it is often impossible to arrive at a stable, consistent set of system requirements.
 - A waterfall model of development is impractical
 - An approach to development based on iterative specification and delivery is the only way to deliver software quickly.


Copyright 2011 @ M.Fahad Khan



Copyright 2011 @ M.Fahad Khan

Scrum part of Agile


SCRUM is a software development methodology, which is a part of Agile. In SCRUM, the development life cycles are fast paced, usually each cycle is around 21 days.



Copyright 2011 @ M.Fahad Khan

Agile software development

Agile software development is a group of software development methodologies based on iterative and incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional teams.



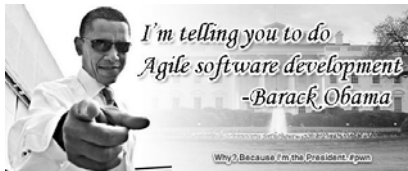
Copyright 2011 @ M.Fahad Khan

Agile methods

- Dissatisfaction with the overheads involved in design methods led to the creation of agile methods.
 - What if:
 - the design is simple and the code quality is high
- These methods:
 - Focus on the code rather than the design;
 - Are based on an iterative approach to software development;
 - Are intended to deliver working software quickly and evolve this quickly to meet changing requirements.
 - What if:
 - the customer can change their mind, substitute functionality, and change priorities
- Agile methods are probably best suited to small/medium-sized business systems or PC products.

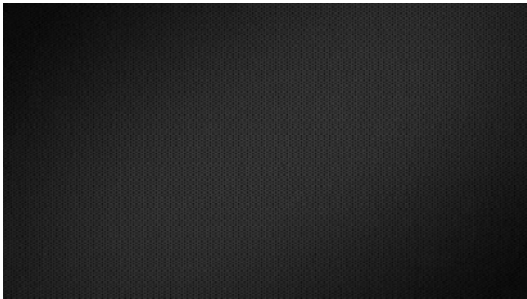
13 Copyright 2011@ M.Fahad Khan

Agile methods are probably best suited

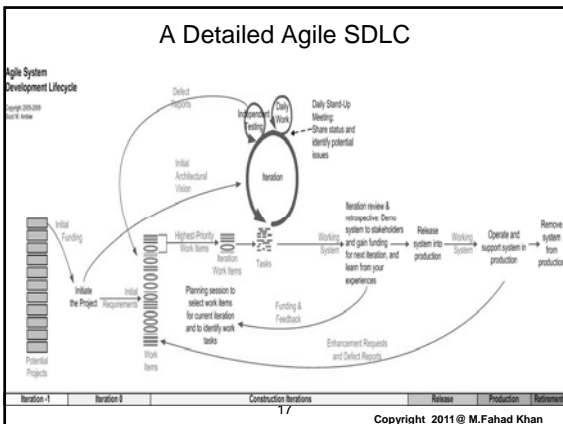
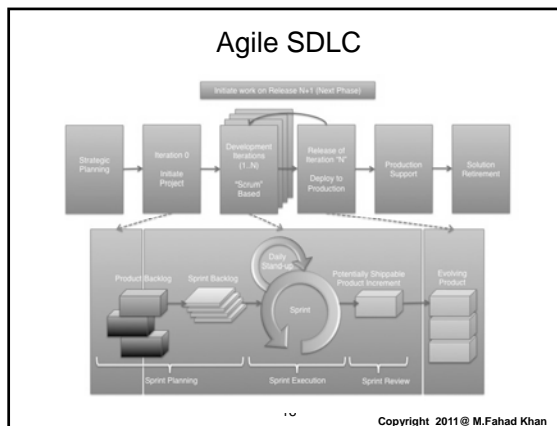


Copyright 2011@ M.Fahad Khan

Agile methods



15 Copyright 2011@ M.Fahad Khan



Phases of Agile method

1. **Requirements.**

The first step in the Agile Software Development Process is to identify some high-level requirements as well as the scope of the release. This 30,000 foot view allows developers to quickly begin coding in order to find out what works even quicker. While the requirements developed within a Waterfall Software Development Process are considered 'law', requirements within an agile process are more or less 'suggestions', and are open to more conversation by the team during other phases. Because requirements are not set in stone, the agile method is more adaptable to changes in requirements as the project grows.
2. **Architecture and Design.**

The goal of the architecture and design phase is to try to identify an architecture that has a good chance of working. The architecture is often defined using free-form diagrams which explore the technical infrastructure, and the major business entities and their relationships. The design is derived in a modeling session, in which issues are explored, until the team is satisfied that they understand what needs to be delivered.

Copyright 2011@ M.Fahad Khan

Phases of Agile method

3. Development.

The development phase uses an evolutionary method that is an iterative and incremental approach to software development. Instead of creating a comprehensive prerequisite such as a requirements specification, that you review and accept before creating a complete design model; the critical development piece evolves over time in an iterative manner. The system is delivered incrementally over time, in small modules that have immediate business value, rather than building and then delivering a system in a single "big bang" release. By focusing development on smaller modules, agile projects are able to control costs despite the seeming lack of planning.

4. Test & Feedback.

One of the key principles of the Agile Methodology is to conduct the testing of the software as it is being developed. The software development is test driven. The unit testing is achieved from the developer's perspective and the acceptance testing is conducted from the customer's perspective

Copyright 2011 @ M.Fahad Khan

DesignMethodology

Copyright 2011 @ M.Fahad Khan

Agile Vs Waterfall

a tale of **two teams**

Copyright 2011 @ M.Fahad Khan

Agile Methodology

Copyright 2011 @ M.Fahad Khan

Agile practices that were easiest to learn.

Agile Practices Easiest to Learn (Top 10 out of 30)

Daily Stand Up Meeting	70%
Continuous Integration	38%
Retrospectives	35%
Iteration Demo	32%
Iteration Planning	31%
Burndown Tracking	27%
Pair Programming	25%
Release Planning	21%
Product Backlog	21%
Code Refactoring	21%

Source: Amblysoft Agile Practices 2009 Survey, <http://www.amblysoft.com/surveys/practices2009.html>
Copyright 2009 Amblysoft Inc.

Copyright 2011 @ M.Fahad Khan

Agile methods

Well-known agile software development methods include:


- Agile Modeling
- Agile Unified Process (AUP)
- Dynamic Systems Development Method (DSDM)
- Essential Unified Process (EssUP)
- Feature Driven Development (FDD)
- Open Unified Process (OpenUP)
- Scrum
- Velocity tracking

Copyright 2011 @ M.Fahad Khan

Principles of Agile Methods	
Principle	Description
Customer involvement	The customer should be closely involved throughout the development process. Their role is provide and prioritise new system requirements and to evaluate the iterations of the system.
Incremental delivery	The software is developed in increments with the customer specifying the requirements to be included in each increment.
People not process	The skills of the development team should be recognised and exploited. The team should be left to develop their own ways of working without prescriptive processes.
Embrace change	Expect the system requirements to change and design the system so that it can accommodate these changes.
Maintain simplicity	Focus on simplicity in both the software being developed and in the development process used.

25
Copyright 2011 @ M.Fahad Khan

Agile Tales



Agile Tales

Presents...

Copyright 2011 @ M.Fahad Khan

Software companies using Agile Methods

<http://www.ifsworld.com/>

Copyright 2011 @ M.Fahad Khan

Software companies using Agile Methods

http://wn.com/Agile_Scrum_XP_Software_Development_On-Demand/

Copyright 2011 @ M.Fahad Khan

Manifesto for Agile Software Development

“uncovering better ways of developing software by doing it and helping others do it”

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

29
Copyright 2011 @ M.Fahad Khan

Individuals and Interactions over Processes and Tools

- Focus on good people who work well together.
 - A good team with no process (or, worse, a bad one) is better than a bad team with a good process...
- Ideally the team should be made up of “generalizing specialists” –
 - people who specialize in something, but are conversant(having frequent or familiar association) in all the disciplines the team needs.

30
Copyright 2011 @ M.Fahad Khan

A Good Team



Copyright 2011 @ M.Fahad Khan

Working Software over Comprehensive Documentation

- We're not paid to develop documentation, we're paid to develop software.
 - Of course that doesn't mean that documentation is unimportant, but it has to be considered relative to its cost in delivering functionality (i.e, software).
 - Traditional development stress on flawless documentation.

Copyright 2011 @ M.Fahad Khan

Working Software over Comprehensive Documentation



Copyright 2011 @ M.Fahad Khan

Agile Board



Copyright 2011 @ M.Fahad Khan

Agile Board



http://www.xqa.com.ar/visualmanagement/tag/task-boards/


Copyright 2011 @ M.Fahad Khan

Working Software Principles

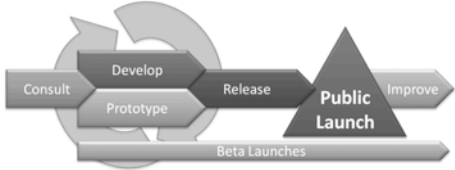
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
 - Working software is the primary measure of progress.
 - No denial of the importance of the software that meets user needs today- but exhibit the quality characteristics in the future .

Copyright 2011 @ M.Fahad Khan

Working Software Principles



"The function of good software is to make the complex appear to be simple."
-Andy Beck



Consult Develop Prototype Release Public Launch Improve
Beta Launches

Copyright 2011 @ M.Fahad Khan

Customer Collaboration Principles

- Business people and developers must work together throughout the project.
 - Give highest priority to satisfy the customer through early and continuous delivery of valuable software.
- Customer Collaboration over Contract Negotiation
 - The primary purpose of collaboration is to work with the customer to make sure that they are involved in delivering the final product they need.
 - The primary purpose of a contract is to cover your weaknesses

Copyright 2011 @ M.Fahad Khan

Customer Collaboration Principles



Product Owner (role)
Available Customer (role)
Developer Team
Technical Owner (role)
Developer (role)

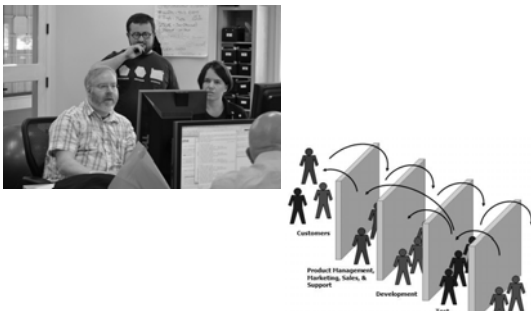
Copyright 2011 @ M.Fahad Khan

Responding to Change Principles

- Continuous attention to technical excellence and good design enhances agility.
 - Welcome changing requirements, even late in development.
 - Agile processes harness change for the customer's competitive advantage.
- A changed requirement late in the lifecycle is a competitive advantage as long as you can act on it.
 - If you froze requirements 18 months ago and deliver tomorrow, your software is delivering on the business needs of 18 months ago.

Copyright 2011 @ M.Fahad Khan

Responding to Change Principles



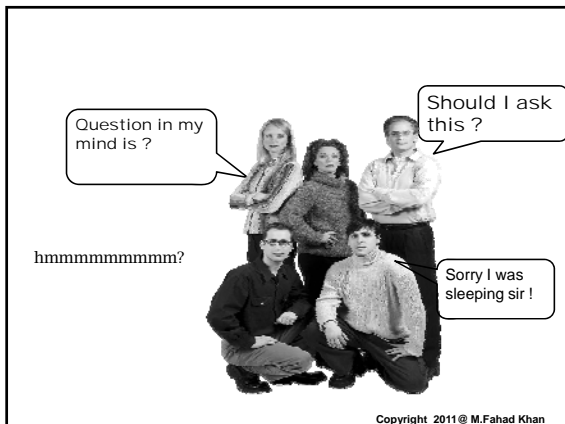
Customers
Product Management, Marketing, Sales, & Support
Development
Test

Copyright 2011 @ M.Fahad Khan

Miscellaneous Principles

- Simplicity – the art of maximizing the amount of work done – is essential.
 - The best architectures, requirements, and designs emerge from self-organizing teams.
- Agile processes promote sustainable development.
 - The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
 - Also known as the "40 Hour Week Rule."

Copyright 2011 @ M.Fahad Khan



Assignment 1

Give one practical example of real time system which is developed through agile development

Hint:

www.google.com

www.wikipedia.org

Due Date: 27-12-2011(Omega)
29-12-2011(alpha)

If you have any query please feel free to ask

Phone: +92-51-9047-592

Fax: +92-51-9047-420

Email: fahad.khan@uettaxila.edu.pk

University Of Engineering & Technology Taxila Pakistan

Copyright 2011 @ M.Fahad Khan