Fundamentals of Agile PM

Agile PM 101
Seminar Overview

Module 1
METHODOLOGIES & FRAMEWORKS:
Agile vs. Traditional

Module 2
PLAN & ESTIMATE:
Why, When, and How To Do It

Module 3
EXECUTE & CONTROL:
Tools, Techniques and Case Study Exercise

Module 4
ENTERPRISE ADOPTION:
How To “On-Board” Agile Pilot Projects

Module 5
TIPS & TRICKS:
The Rest of the Story
Introduction:
John Stenbeck, PMP, PMI-ACP, CSM, CSP

- Sr. PM Consultant for client-side ERP implementations
- Experience in Accounting, IT, Aerospace, Construction, and Manufacturing
- Train technical professionals in Aerospace, Defense, High Technology, Financial Services, Bio-medical and Life-science fields
- Adjunct instructor at the University of California San Diego (UCSD)

Introduction:
John Stenbeck, PMP, PMI-ACP, CSM, CSP

- Featured speaker at HP’s National Conference (2002), Oracle’s Annual Conference (2003), and PeopleSoft’s International Conference (2004).
- One project reached the Federal Supreme Court.
- Front page feature in the San Diego Union and LA Times; Guest on Oprah and the Today Show.
- Past President and VP of Prof. Dev. for PMI-SD
GR8PM Introduction:

- PMP® and PMI-ACP® Exam Prep classes
- Scrum Master Certification classes
- Corporate On-site seminars:
  - Command Course in Agile Project Management
  - Project Management Boot Camp
  - Crash Course in Leadership
  - Masters Course in Estimating and Risk Management
- Organizational Support Services:
  - Agile Enterprise Roll-out Consulting
  - Contract Project Managers

Partial List of Past Clients

- Booz Allen Hamilton, Inc., McLean, VA
- Guinness Bass Import Company, Greenwich, CT
- Lucent Technologies – Bell Labs, Allentown, PA
- Nike Corp., Beaverton, OR
- Oracle Corp., Redwood Shores, CA
- Orange County Public Works, Orange, CA
- Qualcomm Inc., San Diego, CA
- U.S. Army – Space & Terrestrial Comms., Fort Monmouth, NJ
- U.S.D.A. – National Finance Center, New Orleans, LA
- Visa – Smart Cards, Foster City, CA
Fundamentals of Agile PM:

Module 1:
METHODOLOGIES & FRAMEWORKS

Agile PM 101 – Best Practices:
Module 1: METHODOLOGIES & FRAMEWORKS

Introduction: Simple Example
Scenario: You + 3 Friends hold a dinner party.
Project Objective: Successful party, and still be friends!
Cross-functional Team: Each a specialist in either Cocktails, or Appetizers, or Entrees, or Desserts.
Deadline & Constraints: Friday night ; Budget (so no hired Labor), Mutual Accountability, and Integration.

Sprint 1: Planning and Shopping.
Sprint 2: Preparation and Greeting.
Sprint 3: Dinner and Conversation.
Sprint 4: Dessert and Good-byes.
Are You Ready For Some Key Questions?

Is Agile Really Needed?

The last major tool recognized in the PMBOK (Second Edition) was Critical Chain in 1997.

What has changed since then?

- Google launched in September, 1998
- The iPod was unveiled in October, 2001
- The BlackBerry “smartphone” was released in January, 2002
- NASA’s Phoenix lander extracted Martian ice in June, 2007
- The iPad was released in April, 2010
Has Complexity Increased?

*Iron Triangle* transformed into *Hell-of-a-Hexagon*
From *Three* to *Fifteen* Interrelationships.

Agile PM 101 – Best Practices:
Module 1: METHODOLOGIES & FRAMEWORKS

Is Agile Being Used?

Organizations Using Agile
Source: PMI.org (2011)

- Select Projects: 33%
- Extensively: 50%
- Not At All: 17%
How Has Requirements Complexity Increased?

**FEATURE USAGE**

- **Never**: 45%
- **Rarely**: 19%
- **Sometimes**: 16%
- **Often**: 13%
- **Always**: 7%

Feature Usage reported by Jim Johnson, Chairman, Standish Group International, Inc., XP2002 Conference, Sardinia, Italy

Organizations

- **TRADITIONAL PROJECT MANAGEMENT**
  - PROJECT MANAGEMENT INSTITUTE
  - PRINCE²®, APM, IPMA, and Various Universities

- **AGILE PROJECT MANAGEMENT**
  - SCRUM ALLIANCE
  - Agile Alliance, Scrum.org, PMI, and Various Universities
Certifications

PMI

PgMP

PMP

ACP

CAPM

SCRUM ALLIANCE

Intellectual Schema

Methodology = Philosophical Foundation
(PMBOK or Agile)

Framework = Logical Foundation
(Scrum, Extreme, FDD, Lean or Spiral)

Processes = Practical, How To Protocols
(Sponsoring, Organizing, Funding, Controlling)

“I estimate that 75% of those organizations using Scrum will not succeed in getting the benefits they hope for from it.”

Ken Schwaber, Scrum Co-Creator
Agile PM 101 – Best Practices:
Module 1: METHODOLOGIES & FRAMEWORKS

We 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

We Would Add… not a …
Agile PM 101 – Best Practices:
Module 1: METHODOLOGIES & FRAMEWORKS

Similar Taxonomy for Each Methodology
Definitions are completely arbitrary logical devices.

**PMBOK:**
1. Objective
2. Phase
3. Work Pkg.
4. Activity
5. Task

**AGILE:**
1. Product
2. Theme
3. Epic
4. Story
5. Task

**DEFINITION:**
1. Business-level Full Function Vision
2. What a User Class wants to see or experience (Sub-function; End-to-end workflow)
3. What a User will do and the result(s) they will see
4. Workflow component in User words
5. Technical job plus acceptance criteria

---

**Agile PM 101 – Best Practices:**
Module 1: METHODOLOGIES & FRAMEWORKS

Similar Roles

**Traditional vs. Agile/Scrum**

**Traditional:**
- Stakeholders & Sponsor
- Program or Sr. Project Manager
- Jr. PM or Team Lead
- Team and SME’s
- Everybody else

**Agile / Scrum:**
- Stakeholders & Sponsor
- Product Owner
- Scrum Master
- Team and SME’s
- Everybody else
Agile PM 101 – Best Practices:
Module 1: METHODOLOGIES & FRAMEWORKS

Agile’s Best Known Framework

GR8PM
Traditional • Agile • Hybrid
Training • Coaching • Consulting
Fundamentals of Agile PM:

Module 2:
PLAN & ESTIMATE

Agile PM 101 – Best Practices:
Module 2: PLAN & ESTIMATE

Enterprise-level planning

- Strategic Planning
- Tactical Planning
- Operational Planning
- Production Planning
Enterprise-level Development Planning

Market Development

Product Development

Operational Planning

Production Planning

Granularity unlocks the Agile-Enterprise!

Product Functionality (Themes)

Roadmaps (Feature Stories)

Releases (User Stories)

Iterations (Stories & Tasks)
Vision Statement Example:

For music lovers who want a simple way to listen to and manage songs, the iPod is a portable digital music player that provides a user friendly experience with intuitive, easy to use controls. Unlike other MP3 players, our product provides seamless integration with a world class music store (iTunes).

Best Practice: Ideally, a vision statement gives the team the ability to explain the project to someone “in an elevator”.

Project Initiation: Help Sponsor Format the Vision

- FOR (target customer)
- WHO WANTS (statement of the need)
- THE (product name)
- IS A (product category)
- THAT (product key benefit, compelling reason to buy)
- UNLIKE (primary competitive alternative)
- OUR PRODUCT (final statement of primary differentiation)

Best Practice: Using this Vision Statement format helps Stakeholders clearly articulate previously vague thoughts.
Project Initiation:

Product Vision Questions:

- Do you know what you want developed?
- Who is it for? Why are we developing this for you?
- How does this align with your strategic objectives?
- What benefits are expected from a successful deployment?
- What are the consequences of failure?
- What are your competitors doing in this area?

EXERCISE SCENARIO:

You have been hired by a small technology company to create a new iPhone App to complement their marketing strategy – funny, interactive novelties. The App must be something that clients find clever and valuable because it is entertaining and quirky!

The last iPhone App – the “Practical Joker” - was a huge success because it played on people’s enjoyment of practical jokes. Using the iPhone’s ability to “hear and identify” songs, it “listened” to whomever the iPhone was pointed at and then announced whether they were “honest and beautiful” or “misled and homely” or “sinister and hot” among a host of other entertaining descriptions.

Your customer needs another “first” to keep their competitive advantage… and that is your mission!
Exercise Instructions:

This is an *Iteration Planning* meeting

- I am the customer/proxy
- You are the development team
- You have 2 deliverables
- *Listen to both* parts of the Instructions

---

Exercise Instructions:

Part 1:

- Discuss the Scenario provided to identify “value” factors
- *Brainstorm* *WILD* ideas for your new iPhone App
- Review the **Appendix**: iPhone App Vision Statement Examples
- As a Team, choose whether to use:
  - Your own wild idea
  - Your own tame idea
  - One of the Vision examples provided
Agile PM 101 – Best Practices:
Module 2: PLAN & ESTIMATE

Instructions:
Part 2: Your team has 2 deliverables
- Write the Vision in “standard” format
- Write a Press Release
- You have 15 minutes

Do you commit to the Iteration Goal?
- Help each other, have fun, and remember “good enough”
- Your time is limited, so focus!

ANY QUESTIONS?

Review & Retrospective Meetings:
1. If your team completed BOTH deliverables please stand up.
   (Notice that the last 10% often takes 90%!)  
2. Review Meeting – Volunteer to Present
   - Share the Vision
   - Read the Press Release
3. Retrospective Meeting
   - As a Team, discuss how you could improve your process
   - Large group debrief – Share your insights
When is it best to do detailed estimating, (a) when you know very little or (b) when you know a lot?

The amount of estimating done in Traditional versus Agile is similar. It just happens at different times, with different assumptions.

### Fundamental Premise of Planning

<table>
<thead>
<tr>
<th>Estimate Types</th>
<th>% Design</th>
<th>Estimate Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROM</td>
<td>0 – 10 %</td>
<td>+100% to – 50%</td>
</tr>
<tr>
<td>Budget</td>
<td>15 – 25 %</td>
<td>+30% to – 15%</td>
</tr>
<tr>
<td>Definitive</td>
<td>45 – 100%</td>
<td>+15% to – 5%</td>
</tr>
</tbody>
</table>

**Estimate Descriptions**

**ROM**: Made without detailed engineering data using tools like capacity curves, scaling factors, and feature/cost ratios.

**Budget**: Made for owner’s planning not project control using tools like flow sheets, layouts, and equipment details.

**Definitive**: Based on detailed engineering data, approved drawings and specifications. Used for project planning and control.
Cost-effective Planning Despite Uncertainty

Level 1 - Product: iPad, v1.0, Web Access & Communication

Level 2 - Themes:
- Video Watching
- Game Playing
- Music Listening
- Traveling
- E-Mailing

Level 3 – Epics (for E-Mailing):
- Manage Contacts
- Create Messages
- Store & Retrieve Messages
- Attach & Link Content
- Filter Viruses & Spam
Level 3 – Epics (for E-Mailing):
  • Create Messages
  • Manage Contacts
  • Store & Retrieve Messages
  • Attach & Link Content
  • Filter Viruses & Spam

Level 4 – Stories:
  • Create Contact
  • Update Contact
  • Delete Contact
  • Sort Contacts

Level 5 – Tasks:
  • Define Fields
  • Define DB
  • Define GUI
  • Check Duplicates
  • Validate Format
  • Import Function

Does this remind you of Progressive Elaboration and Rolling Wave Planning?
### Agile PM 101 – Best Practices:
Module 2: PLAN & ESTIMATE

<table>
<thead>
<tr>
<th>Priority</th>
<th>User Story Description</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>As a:</strong> Email User</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>I want to:</strong> Sort my emails</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>So that:</strong> I can find emails from specific people</td>
<td></td>
</tr>
</tbody>
</table>

#### Acceptance Criteria

**Given:** I am logged into my email account

**When:** I want to sort my emails

**Then:** I can sort them by sender
Properly defining “Done” is critical to success.
• Detailed Lessons Learned are a great resource.

**Definition of “Done” – Example:**
• Cleanly coded into an executable
• Refactored for usability
• Unit and Regression tests are error free
• Includes User Documentation

**Definition of “Done” – More Examples:**
Install the Radio Broadcast System (RBS) equipment:
• Cabinet anchors properly drilled and set in concrete slab
• Electrical supply and grounding tested without fault
• Equipment rack, UPS, and signal equipment installed and tested error free
• RF jumpers connected and swept
• RBF integrated to core and test calls 100% error free
• As installed documentation, and photos, loaded to facilities repository
Definition of “Done” – More Examples:

Facilitate Emergency Response Tabletop Exercise:

- Identify, recruit, and confirm participants
- Validate time blocked in participants’ calendars
- Preparation information delivered and expectations communicated to participants
- Attendance confirmed 24 hours beforehand
- Welcome participants and execute exercise
- Conduct Lessons Learned and collect evaluations

Aircraft hardware installation and certification:

- All materials received and certified per Bill of Materials (BOM)
- All processes have been certified against FAA protocols
- All technicians have proper certification on record
- Each component replaced by certified technician and reviewed/approved by Supervisor
- Each component is unit tested and no errors are detected
- Each system is regression tested and no errors are detected
- Entire aircraft is run through complete ground safety test protocol and no errors are detected
- All documentation is properly filed
Fibonacci Sizing:

1. The Fibonacci series = 1, 2, 3, 5, 8, 13, 21, 34, 55…
   • Defined by the non-linear recurrence equation: \( F_n = F_{n-1} + F_{n-2} \)

2. Process first establishes a “midpoint” and then the relative size of other items in the group

3. Leverages our physiological “wiring” because it uses Size instead of Time!

4. First we will learn HOW to do it, then we will APPLY it.

Exercise Instructions – Round 1:
1. As a Team pick the “medium” dog and assign it a size of 8.
2. Individually, put a relative size to the other dogs (2, 3, 5, 8, etc.)
3. You can use any number more than once, or not at all.
4. After everyone has finished, as a Team review and size each dog.

Use the following process:
(a) Start at top of list.
(b) Share each one’s number.
(c) Highest number, then lowest, explain why.
(d) Use the explanations to re-vote.
(e) Negotiate the final team number.

<table>
<thead>
<tr>
<th>Breed</th>
<th>Individual</th>
<th>Team</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black Lab</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dachshund</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Great Dane</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Terrier</td>
<td></td>
<td></td>
</tr>
<tr>
<td>German Shepherd</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Poodle (std.)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>St. Bernard</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bulldog</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Exercise Instructions – Round 2 (Planning Poker):
1. Write the Numbers below on each of 9 Index Cards.

2. Create an index card with a ? and an ∞ symbol.
   
   How to use the ? and ∞ cards.

3. Size the Cars on the next slide.

Planning Poker Exercise:
1. As a Team pick the “medium” car and assign it a size of 8.
2. Then start at the top and individually choose a card for the size.
3. Use only the number cards for now.
4. On the count of 3, everyone shows their card at once!
5. Highest, then lowest, explain why.
6. Use the explanations to re-vote.
7. Negotiate final number.
Exercise Instructions – Round 3:

1. Evaluate **Your** User Stories and choose a “midsize” one.
2. Use Planning Poker to size all of your User Stories.
   - **Write the size** on each story.
   - You have to **focus**!
   - **Remember**, good enough! Don’t stress.

---

**BONUS CONTENT??**
Implications for Project leaders:
- Understand multiple Agile frameworks and tools
- Tailor Agile to the organizational context
- Integrate Traditional tools when appropriate
- Embrace change and create hybrid approaches

Sixth Annual "State of Agile Development"

Key Changes (since last year):
- Companies using Scrum: **down 10%**
- Companies using Hybrids: **up 25%**
Why is Agile Mainstream
Your PMI-ACP® Qualifications

Project Management Process Groups

<table>
<thead>
<tr>
<th>Knowledge Areas</th>
<th>Initiation</th>
<th>Planning</th>
<th>Executing</th>
<th>M &amp; C</th>
<th>Closing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Management Integration</td>
<td>Initiation</td>
<td>Planning</td>
<td>Executing</td>
<td>M &amp; C</td>
<td>Closing</td>
</tr>
<tr>
<td>▪ Develop Project Charter</td>
<td>▪ Develop Project Management Plan</td>
<td>▪ Direct and Manage Project Execution</td>
<td>▪ Monitor and Control Project Work</td>
<td>▪ Perform Integrated Change Control</td>
<td>▪ Close Project or Phase</td>
</tr>
<tr>
<td>Project Scope Management</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
</tr>
<tr>
<td>Project Time Management</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
</tr>
<tr>
<td>Project Cost Management</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
</tr>
<tr>
<td>Project Quality Management</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
<td>▪ Init</td>
</tr>
</tbody>
</table>

Inputs
- Project Statement of Work (Light version)
- Contract (Agile version)

Outputs
- Project Vision Statement
- Feature Descriptions

All projects – Traditional and Agile – use a kick-off ceremony to describe the engagement and manage expectations. It usually includes reviewing the Project Charter that describes Why, What, When, Where, and for Whom the project is being done.

The key difference lies in the approach to the question, “How will it be done?” Agile uses iterative development to increased business involvement, team commitment, and process improvement.
Why is Agile Mainstream
Your PMI-ACP® Qualifications

Project Management Process Groups

<table>
<thead>
<tr>
<th>Knowledge Areas</th>
<th>Initiation</th>
<th>Planning</th>
<th>Executing</th>
<th>M &amp; C</th>
<th>Closing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Scope Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quality Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Inputs**
- Collect Requirements
- Define Scope
- Create WBS

**Outputs**
- Define Activities
- Sequence Activities
- Estimate Activity Duration
- Develop Schedule

The PMBOK identifies Define Scope and many PMs interpret that as a one-time activity even though it conflicts with Progressive Elaboration. Agile embraces Progressive Elaboration by establishing a framework – the Product Backlog – for scope management. The Backlog is prioritized and frequently reviewed and revised to manage project scope.

Given the high uncertainty about requirements and/or high technological risks facing most projects it is a wise approach.

Why is Agile Mainstream
Your PMI-ACP® Qualifications

Project Management Process Groups

<table>
<thead>
<tr>
<th>Knowledge Areas</th>
<th>Initiation</th>
<th>Planning</th>
<th>Executing</th>
<th>M &amp; C</th>
<th>Closing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Project Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integration</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project Scope Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Time Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cost Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Quality Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Inputs**
- Prioritized Product Backlog

**Tools**
- Prioritization techniques
- Sizing and Estimating techniques
- Iteration Planning techniques

**Outputs**
- Prioritized Iteration Backlog
- User Stories

Traditional project management uses a hierarchy of Program, Project and sub-Project plans to sequence work and measure progress. Agile uses a hierarchy of Roadmap, Release, and Iteration plans to do the same thing.

Agile project sequencing is more dynamic because of its Lean-driven focus to eliminate the waste of non-value-added work.
What the Exam Tests

- The ACP exam is based on 11 books from agile authors.
- That body of work covers 3,888 pages.
- The 11 authors sometimes have differing and/or opposing opinions regarding agile practices.

<table>
<thead>
<tr>
<th>Title</th>
<th>Price</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agile Estimating and Planning</td>
<td>$54.99</td>
<td>313</td>
</tr>
<tr>
<td>Agile Retrospectives: Making Good Teams Great</td>
<td>$29.95</td>
<td>200</td>
</tr>
<tr>
<td>Agile Project Management: Creating Innovative Products</td>
<td>$49.99</td>
<td>403</td>
</tr>
<tr>
<td>Agile Project Management with Scrum</td>
<td>$39.99</td>
<td>402</td>
</tr>
<tr>
<td>The Software Project Manager’s Bridge to Agility</td>
<td>$49.99</td>
<td>300</td>
</tr>
<tr>
<td>Agile Software Development: The Cooperative Game</td>
<td>$59.99</td>
<td>513</td>
</tr>
<tr>
<td>Coaching Agile Teams</td>
<td>$49.99</td>
<td>392</td>
</tr>
<tr>
<td>Becoming Agile: In an Imperfect World</td>
<td>$44.99</td>
<td>308</td>
</tr>
<tr>
<td>The Art of Agile Development</td>
<td>$39.99</td>
<td>300</td>
</tr>
<tr>
<td>User Stories Applied For Agile Software Development</td>
<td>$51.99</td>
<td>313</td>
</tr>
<tr>
<td></td>
<td>507.85</td>
<td>3,888</td>
</tr>
</tbody>
</table>

What about PDUs?

- Continuing Certification Requirements (CCR):
  - PMI-ACP® holders are required to earn 30 professional development units (PDUs) in agile topics every 3 years.

- Counting PMI-ACP® PDUs for other PMI credentials:
  - The 30 PDUs in agile topics also count towards the PMP® or PgMP® certification(s).
  - For general project management classes, only the hours that pertain to agile topics count towards the PMI-ACP®.
Fundamentals of Agile PM:

BONUS MATERIAL

Agile PM 101 – Best Practices:
Module 3: EXECUTE & CONTROL

Where Does Agility Come From?

AVOIDING WASTE
(100% PRODUCTIVE)
What is the Agility “Challenge”? 

How is Agility Increased?
How is the Cone of Uncertainty narrowed?

Affinity Estimates

<table>
<thead>
<tr>
<th></th>
<th>Smaller</th>
<th>Larger</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGH</td>
<td>XS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>S</td>
<td>L</td>
</tr>
<tr>
<td></td>
<td>M</td>
<td>XL</td>
</tr>
<tr>
<td>LOW</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Agile PM 101 – Best Practices:
Module 3: EXECUTE & CONTROL

Enterprise Planning

Roadmap – 1 Year

| Release #1 | 4 Months | | Release #2 | 4 Months | | Release #3 | 4 Months |
|------------|----------|------------|------------|----------|------------|----------|
| Feature #1 | Feature #2 | Feature #3 | Feature #4 | Feature #5 | Feature #6 | Feature #7 | Feature #8 |

BACKLOG:
Stories
#1 thru #n

Create a Release Plan

Iteration #1 = 44 days
Iteration #2 = 43 days
Iteration #3 = 45 days
Iteration #4 = 23 days

<table>
<thead>
<tr>
<th>PRIORITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOW</td>
</tr>
<tr>
<td>HIGH</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SMALLER</th>
</tr>
</thead>
<tbody>
<tr>
<td>XS</td>
</tr>
<tr>
<td>S</td>
</tr>
<tr>
<td>M</td>
</tr>
<tr>
<td>L</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LARGER</th>
</tr>
</thead>
<tbody>
<tr>
<td>XL</td>
</tr>
</tbody>
</table>

Priority:

HIGH:

LOW:

8

5

3
Iteration Cycle

<table>
<thead>
<tr>
<th>CALENDAR – 4 WEEK ITERATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>MON.</td>
</tr>
<tr>
<td>STAND-UP &amp; WORK</td>
</tr>
<tr>
<td>STAND-UP &amp; WORK</td>
</tr>
<tr>
<td>STAND-UP &amp; WORK</td>
</tr>
<tr>
<td>STAND-UP &amp; WORK</td>
</tr>
</tbody>
</table>

Agile PM 101 – Best Practices:
Module 3: EXECUTE & CONTROL

Iteration Planning: Best Practices

Cross-Functional Teams

- Increase organizational competency/capacity through cross-training; reduce risk for same reason
- Must have clear Product Vision in order to self-organize
- Do “a little of everything” all the time, rather than doing “all of 1 thing” at a time

Team Size = 7 +/- 2

- Too Big = Communications break down
- Too Small = Scrum Meeting Overhead is Not Justified