



# STAKEHOLDER ENGAGEMENT

## **STAKEHOLDERS, WHY THE FOCUS?**

Projects are undertaken by people, for people. Knowledge worker projects often have no tangible product; therefore, effective communication and stakeholder engagement are critical to ensure that team members know what they are building and what the customer is asking for.

Projects won't be successful without the right stakeholders, so we should push hard to get the people we need. It is essential for stakeholders to stay engaged with the project, so we need to do all we can to make stakeholder involvement 'stick'. To keep stakeholders engaged, we should take actions to recognize and reward stakeholder involvement, such as celebrating project milestones with the stakeholders, talking to their managers about how to recognize their contributions, and making sure project-related feedback becomes part of their performance reviews.

## **STAKEHOLDERS, WHY THE FOCUS?**

Since knowledge worker projects often create intangible products and services, there is a lot of potential for a gap between what the customer wants and describes and what the development team hears and interprets. It is critical to bridge that gap early and often, with each new idea presented on the project.

Projects take a long time, and people generally want things as soon as possible after they have described them. For these reasons, it is important to frequently show stakeholders what has been built on the project. One advantage of engaging the customer and showing them pieces of the project as those pieces are built is that the true rate of project progress is highly visible.

The practices necessary to work successfully with project stakeholders include aligning the stakeholder's understanding of the project, communicating with stakeholders, using critical soft skills, and leading effectively.

## **ALIGNING STAKEHOLDERS' UNDERSTANDING**

Agile methods recognize the importance and challenge of overcoming the “semantic gap” between what customers ask for and how the development team interprets what they ask for. On agile projects, failing is okay as long as the failure is fast and cheap and it’s still possible to recover the project. We ideally want to uncover disconnects in understanding at the lowest possible cost. By doing so, we save the organization money that it can then invest in other projects.

### **WIREFRAMES**

Wireframe models are a popular way of creating a quick mock-up of the product. If there are discrepancies in understanding, the wireframe serves as a useful visual for stakeholders to refer to and adjust until they achieve consensus. Wireframes are a form of “low-fidelity prototyping”. In other words, they are quick and cheap ways to get feedback on something.

## WIREFRAMES CONTD

Agile teams may use wireframe models created in tools like PowerPoint, or they may draw the models freehand on whiteboards or sheets of paper stuck to a wall that allow for easy repositioning to change workflows. The purpose of these tools is to help clarify what 'done' looks like and validate the approach the team plans to take before they commit large amounts of time to building (potentially wrong) increments of the product.

## PERSONAS

Personas are quick guides or reminders of the key stakeholders on the project and their interests. Personas may be based on profiles of real people or composites of multiple users.

## PERSONAS CONTD

When they are used as a project tool, personas should:

- Provide an archetypal description of users
- Be grounded in reality
- Be goal-oriented, specific, and relevant
- Be tangible and actionable
- Generate focus

Personas help the team prioritize their work, stay focused on the users, and gain insight into who the users will be. These tools help team members empathize with users of the product or solution. Personas can help keep a team focused on delivering the features that users will find valuable, and this leads to better decision-making on the project.

## ALIGNING STAKEHOLDERS' UNDERSTANDING

<b>Name:</b> Hassan the Movie expert	
<p><b>Description:</b> Hassan loves movies. On average, he rents 5 movies a week from his local rental store.</p> <p>His two children also like to watch children's TV shows. They often like to watch the same shows more than once, which means that Hassan sometimes has to pay late fees.</p> <p>Hassan's wife has different movie tastes than Hassan and often spends a lot of time choosing a movie.</p>	<p><b>Values:</b> Hassan would like to be able to order movies from the comfort of his home. He would like to be able to search for movies by title, actor, genre, and director. He would also be interested in knowing how other viewers rated the movie.</p> <p>He is looking forward to unlimited movies so his children can watch shows multiple times without having to pay additional fees.</p> <p>He would also appreciate a "recommended" feature to help him and his wife choose movies.</p>

**Figure 1: Sample Persona**

# ALIGNING STAKEHOLDERS' UNDERSTANDING

## USER STORIES/BACKLOGS

User stories are bite-sized, understandable chunks of business functionality. Agile project teams commonly rely on user stories and a backlog of these stories to help align team priorities with the needs of the business. User stories are often written in the following format:

“As a *<Role>*, I want *<Functionality>*, so that *<Business Benefit>*.”

Example: “As a MoviesOnline customer, I want to search movies by actor, so that I can more easily find movies I would like to rent.”

The advantage of this role, function, and business benefit template is that it forces the project to identify the user (Who is asking for this?) and the business benefit (Why are we doing this?) for every required piece of functionality.



## ALIGNING STAKEHOLDERS' UNDERSTANDING

### USER STORIES/BACKLOGS CONTD

Another format, often used for non-functional or system-based requirements, is “Given, When, Then.” For example:

**Given** the account is valid and the account has a MovieCredit balance of greater than ~~≠~~0,

**When** the user redeems credit for a movie,

**Then** issue the movie and reduce the user’s MovieCredit balance.

The way user stories are phrased is important, but there are also other aspects of user stories that contribute to their value on a project. The INVEST mnemonic, is often used as a reminder of the characteristics of effective user stories.

# ALIGNING STAKEHOLDERS' UNDERSTANDING

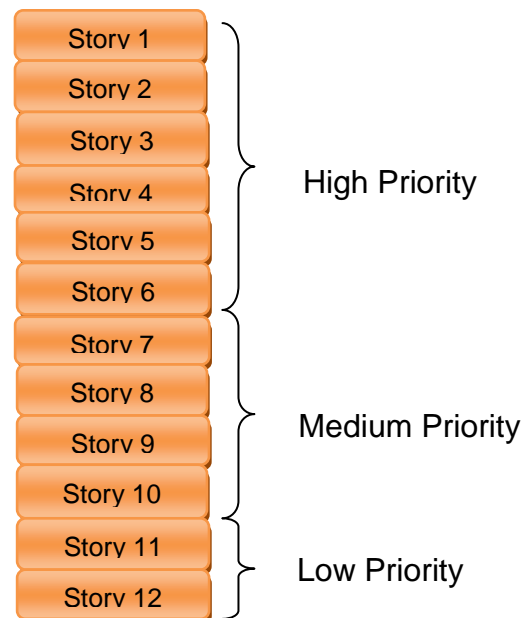
## USER STORIES/BACKLOGS CONTD

- **I – INDEPENDENT:** We try to create independent user stories that can be selected on merit, rather than dragged into the release because other user stories are dependent on them.
- **N – NEGOTIABLE:** Negotiating user stories leads to an improved understanding of the true requirements, costs, and acceptable compromises.
- **V – VALUABLE:** User stories without clearly understood business benefits will be difficult to prioritize, since backlogs are usually ranked on business value.
- **E – ESTIMATABLE:** We have to be able to estimate the effort of a user story.
- **S – SMALL:** Small user stories are easier to estimate and test than large user stories. As work units get larger, people's ability to estimate them reliably decreases.
- **T – TESTABLE:** Having testable user stories is important for tracking progress because we measure progress based upon how many user stories have been successfully accepted.

# ALIGNING STAKEHOLDERS' UNDERSTANDING

## USER STORY BACKLOG

After the user stories are created, they are organized into a backlog. This backlog of user stories is a visible list of the work to be done, as shown in the figure below:



**Fig. 2: User Story Backlog**

# ALIGNING STAKEHOLDERS' UNDERSTANDING

## STORY MAPS

Story maps are great tools for explaining what will go into the first release, second release, and subsequent releases. As such, they are often used in stakeholder communications and can be posted on a wall as an information radiator of the project plan.

## INCORPORATING STAKEHOLDER VALUES

This is about bringing project priorities into alignment with stakeholder priorities. Incorporating stakeholder values means we make sure we do not plan and initiate work that the stakeholders do not support or value at this time.

One way to incorporate stakeholder values on a project is to engage business representatives in the prioritization of the backlog. Another way to incorporate stakeholder's values and interests into the project activities is to invite our stakeholders to retrospectives and planning meetings.

# ALIGNING STAKEHOLDERS' UNDERSTANDING

## STAKEHOLDER MANAGEMENT

Stakeholder management is about comprehensively looking after stakeholders, including in the initial efforts to identify who the stakeholders are. It is important to identify all the stakeholders and effectively manage their involvement in the project, because excluding or alienating any of them will put the successful execution of the project at risk.

If project stakeholders are new to agile methods, they may need some basic education about how agile projects operate to help them understand the approach, address any myths about agile projects, and manage their expectations. This education should include the goals, values, practices, and benefits of the agile approach so they understand why the project will be executed in this manner.

# ALIGNING STAKEHOLDERS' UNDERSTANDING

## STAKEHOLDER MANAGEMENT CONTD

In addition to the initial efforts to identify stakeholders, educate them, and address their concerns, stakeholder management involves continuing to engage them in the project as it progresses.

One key reason to keep stakeholders engaged is to ensure we will hear about change requests as soon as possible. An ongoing dialogue with our stakeholders will also help us identify potential risks, defects, and issues. Agile methods provide multiple touch points with stakeholders and deliberate communication events designed to facilitate early and continuous feedback.

Another aspect of stakeholder management is establishing a process for escalating issues that need a high level of authority to resolve. Agreement on such a process is essential to keep the project operating smoothly.

# ALIGNING STAKEHOLDERS' UNDERSTANDING

## VENDOR MANAGEMENT

Vendor management focuses on stakeholders who are external to the organization but are involved in the project because they are providing some product or service. As with internal stakeholders, we should be careful in selecting vendors and we may need to educate them about the unique aspects of working in an agile environment.

If a vendor needs to use an agile approach to participate in the project, then this requirement should be outlined in the request for proposal (RFP).

Since agile projects acknowledge that requirements may change and scope is negotiable, traditional vendor engagement contracts based on formal specifications are problematic. Instead, agile projects typically use the agile contracting models.

# COMMUNICATING WITH STAKEHOLDERS

## COMMUNICATIONS MANAGEMENT

The preferred way to communicate on agile projects is through face-to-face (F2F) communications. F2F communications have the highest bandwidth of all forms of communication; this means that F2F communications can transfer the most information in a given time period. F2F communications also allow for immediate questions and answers, whereas static methods, like paper documentation, do not.

In F2F communication, the two participants can quickly shortcut information when both people understand the concept, and they can ask each other questions and get immediate feedback. Such conversations also convey a lot of emotional bandwidth through nonverbal communication such as gestures, facial expressions, and tone of voice. In F2F communication, we can quickly tell if people are puzzled, angry, or passionate about a topic. However, while we prefer F2F communications for their efficiency and high bandwidth, we cannot always rely on them; there are times when information does need to be recorded on paper.



# COMMUNICATING WITH STAKEHOLDERS

## INFORMATION RADIATORS

“Information radiator” is an umbrella term for a number of highly visible ways to display information, including large charts, graphs, and summaries of project data. These tools, also sometimes referred to as “visual controls”, quickly inform stakeholders about the project’s status, and they are usually displayed in high-traffic areas to maximize exposure.

The sort of data that might be displayed on an information radiator includes:

- The features delivered to date versus the features remaining to be delivered
- Who is working on what
- The features selected for the current iteration
- Velocity and defect metrics
- Retrospective findings
- Risk registers

Story maps and burn down and burn up charts are common examples of information radiators.

## COMMUNICATING WITH STAKEHOLDERS

### BURN DOWN AND BURN UP CHARTS

Burn down and burn up charts are used to show progress and to help determine when the project (or release within the project) should be complete. Burn down charts show the estimated effort remaining on the project, and burn up charts show what has been delivered.

This means as more work is completed, a burn down chart will show a progress indicator moving downward to indicate the reduced amount of work that still needs to be done. In contrast, the progress indicator on a burn up chart will move upward, to show the increasing amount of work completed.

Burn down charts typically show the estimated time remaining.

# COMMUNICATING WITH STAKEHOLDERS

## VELOCITY

Velocity is the measure of a team's capacity for work per iteration. This metric helps us gauge how much work the team is able to do, based on the number of user stories completed in past iterations. This metric provides a way to communicate what we have accomplished, what we will likely be able to accomplish, and when we expect the project (or release) to be done. Velocity is measured in whatever units the team uses for its work, so those units could be hours, days, points, etc.

When velocity is tracked over multiple iterations, this metric can be used to determine when a project will likely be completed. For example, if a project's velocity averages 50 points per iteration, and the backlog contains 500 points' worth of undeveloped work, then we would divide 500 by 50 to get 10. The data indicates that the project will likely be completed in 10 more iterations.

# COMMUNICATING WITH STAKEHOLDERS

## AGILE MODELING

The term “agile modeling” refers to the various modeling techniques that are commonly used on agile projects. While models are important in agile methods, their main value often lies in the discussion and creation of the model, rather than the final output. As a reflection of this, agile models are often sketched on whiteboards and then photographed as a means of recording them. The value is in the creation, not the beautification and preservation of the model in a specialized modeling tool.

Agile models are typically lightweight capturing the design without a need for further polish.

## AGILE MODELING CONTD

The types of agile models that can be created during agile modeling include:

- Use case diagrams
- Data models
- Screen designs

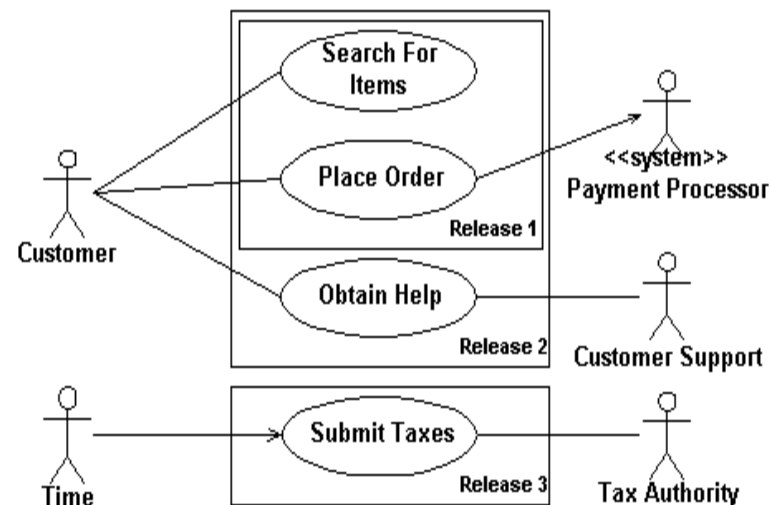


Fig 3: Sample Use Case Diagram

## AGILE MODELING CONTD

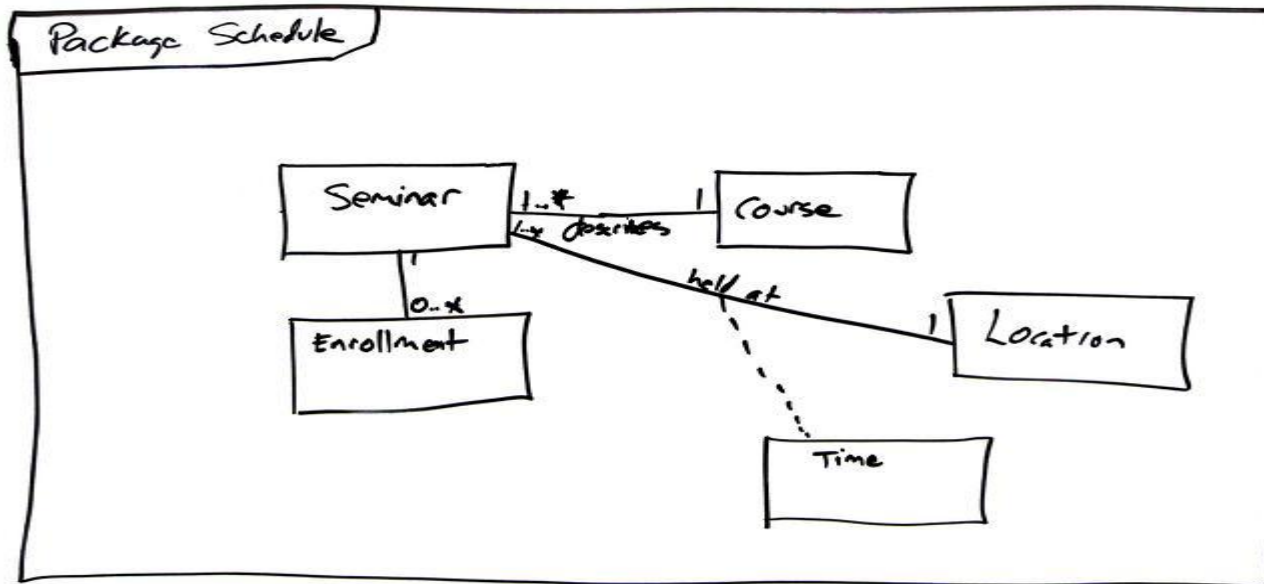
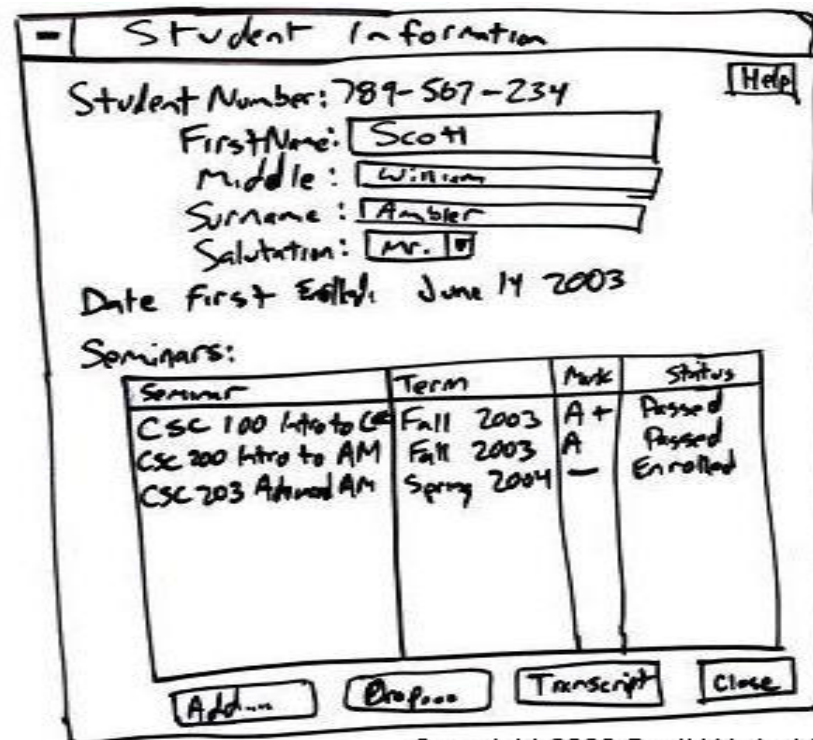


Fig 4: Sample Data Model

# COMMUNICATING WITH STAKEHOLDERS

## AGILE MODELING CONTD



Student Information [Help]

Student Number: 789-567-234

First Name:

Middle:

Surname:

Salutation:

Date first Enrolled: June 14 2003

Seminars:

Seminar	Term	Mark	Status
CSC 100 Intro to CS	Fall 2003	A+	Passed
CSC 200 Intro to AM	Fall 2003	A	Passed
CSC 203 Advanced AM	Spring 2004	-	Enrolled

Copyright 2002 Scott W. Ambler

Fig 5: Sample Screen Design

## USING CRITICAL SOFT SKILLS

Soft skills are often more difficult to master than technical skills. Poor soft skills can quickly demoralize and disenchant a technically strong team, while someone who effectively uses soft skills can extract amazing results from average teams.

We use the term “critical soft skills” instead of “critical soft talents” because unlike talents, which we are largely born with, skills can be acquired and improved through education and practice. The following critical soft skills will be tested on the PMI-ACP exam:

- Negotiation
- Active listening
- Facilitation methods
- Globalization, culture, and team diversity
- Conflict resolution
- Distributed teams
- Participatory decision models



# USING CRITICAL SOFT SKILLS

## NEGOTIATION

Negotiations happens throughout an agile project, especially when discussing the requirements or priorities of features and what “done” would best look like.

Negotiating on agile projects does not have to be – and typically should not be – a zero-sum game (i.e. with a winner and a loser). Instead, healthy negotiations allow each party to investigate the trade-offs and present alternative perspectives. There should be an opportunity for each viewpoint or business case to be fully described, noting the pros and cons of the different options. Negotiations are most effective when the interactions between participants are positive and there is some room for give and take on each side.

## USING CRITICAL SOFT SKILLS

### ACTIVE LISTENING

Active listening is hearing what someone is really trying to convey, rather than just the meanings of the words they are speaking. On agile projects, we need to listen for the message, not just the string of words being spoken.

Active listening is a skill that can be improved with practice. According to the book *'Co-Active Coaching: Changing Business, Transforming Lives'*, our listening skills progress through three levels:

**LEVEL 1: INTERNAL LISTENING** – We hear the words being spoken, and although we may be very attentive, we interpret them through our own lens. When listening, we are thinking “How is this going to affect me?” and miss the speaker’s real message.

## USING CRITICAL SOFT SKILLS

### ACTIVE LISTENING CONTD

**LEVEL 2: FOCUSED LISTENING** – When listening at this level, we let go of our own thoughts and put ourselves in the mind of the speaker. We empathize with their thoughts, experiences, and emotions as they tell us about the situation. We look for emotional indicators in the speaker’s words and pauses, their voice and tone, and their facial expressions for more information about how the person feels about what they are describing.

**LEVEL 3: GLOBAL LISTENING** – When listening at this level, we build on the approach taken in level 2, adding a higher level of awareness, like an antennae function, to pick up on subtle physical and environmental indicators. These indicators can include the speaker’s movements or posture, their energy level, and the atmosphere or vibe in the room. We notice factors like whether the speaker is voicing the information openly in front of others or privately, the mannerisms of others who are within earshot, and many other subtle clues to help us understand a fuller context of the information being shared.

# USING CRITICAL SOFT SKILLS

## FACILITATION METHODS

This is essentially testing whether you understand how to run effective meetings and workshops. When facilitating a meeting or session, keep the following in mind:

- **Goals:** Establishing a clear goal for each meeting or workshop session can help people get engaged in the discussion from the start.
- **Rules:** Establishing some basic ground rules is another important technique for holding effective sessions.
- **Timing:** Timing is always important when we are trying to get a group of people together, and it can be easy to lose track of the time once the session is going. Therefore, the duration of the session should be established ahead of time, and someone should be designated as the timekeeper.
- **Assisting:** The session facilitator needs to make sure the meeting is productive and that everyone has a chance to contribute.

## **USING CRITICAL SOFT SKILLS**

### **GLOBALIZATION, CULTURE, AND TEAM DIVERSITY**

It is not uncommon to have team members from three or four continents working on the same project. Since knowledge work is invisible, we rely primarily on communication to share project information. Different cultures bring communication challenges, but they can also bring efficiency to a project because you have a broader pool of resources to choose from.

The XP practice of pairing can help overcome local issues related to diversity. Pairing gives team members a great opportunity to learn about how others communicate progress, raise problems, and brainstorm.

## USING CRITICAL SOFT SKILLS

### CONFLICT RESOLUTION

Conflict is an inevitable part of project work. Whenever people come together to solve problems, there will be differences of opinion and competing interests. Some degree of conflict is healthy, to ensure that ideas are sufficiently tested before they are adopted. However, we need to make sure the conflict does not escalate beyond healthy skepticism and friendly teasing, or we will end up with a negative and repressive team environment.

Creating an environment in which people can use conflict constructively is a key part of successfully engaging stakeholders on a project. We must watch for instances when conflict moves beyond normal, healthy debate and becomes destructive and harmful to the relationships and the team.

## USING CRITICAL SOFT SKILLS

### CONFLICT RESOLUTION CONTD

Level	Name	Characteristic	Language Type	Atmosphere/Environment
Level 1	Problem to solve	Information sharing and collaboration	Open and fact-based	<ul style="list-style-type: none"> <li>➤ People have different opinions or misunderstandings</li> <li>➤ Conflicting goals or values</li> <li>➤ Not comfortable, but not emotionally charged either</li> </ul>
Level 2	Disagreement	Personal protection trumps resolving the conflict	Guarded and open to interpretation	<ul style="list-style-type: none"> <li>➤ Self-protection becomes important</li> <li>➤ Team members distance themselves from the debate</li> <li>➤ Discussions happen off-line (outside of the team environment)</li> <li>➤ Good-natured joking moves to half-joking barbs</li> </ul>
Level 3	Contest	Winning trumps resolving the conflict	Includes personal attacks	<ul style="list-style-type: none"> <li>➤ The aim is to win</li> <li>➤ People take sides</li> <li>➤ Blaming flourishes</li> </ul>
Level 4	Crusade	Protecting one's own group becomes the focus	Ideological	<ul style="list-style-type: none"> <li>➤ Resolving the situation is not good enough</li> <li>➤ Team members believe that people "on the other side" will not change and need to be removed</li> </ul>
Level 5	World War	Destroy the other!	Little or nonexistent	<ul style="list-style-type: none"> <li>➤ "Destroy!" is the battle cry</li> <li>➤ The combatants must be separated</li> <li>➤ No constructive outcome can be had</li> </ul>

**Table 1: Leas's framework on the stages of conflict**

## USING CRITICAL SOFT SKILLS

### CONFLICT RESOLUTION CONTD

In summary, conflict is normal and inevitable when people work closely together. Leas's model can help us objectively assess the severity of a conflict. We should pay attention to the language being used and give the team an opportunity to resolve the conflict themselves. If we do need to intervene, we should focus on de-escalating the problem by separating facts from emotions and looking for ways to help people move forward, despite their differences.



# USING CRITICAL SOFT SKILLS

## DISTRIBUTED TEAMS

Distributed stakeholders present a challenge to both face-to-face communication and tools such as information radiators that rely on co-location to work best.

To help address this challenge, distributed teams can use communication technologies like video conferencing, live chat, Skype, and other tools to simulate a shared team environment and allow distributed stakeholders to chat and interact as if their colleagues were within earshot.

## USING CRITICAL SOFT SKILLS

### PARTICIPATORY DECISION MODELS

Participatory decision models present different ways to engage the team in the decision-making process. The speed at which we make decisions and our team's level of agreement with these decisions will impact both project performance and team cohesion. Also, since knowledge worker projects have no tangible, emerging product moving down a production line, communication and decision-making processes become more critical to keep everyone informed and engaged.

The first thing to realize is that it is not realistic to expect the team to achieve total agreement on all issues and decisions. The mechanisms put in place for making tough decisions in agile projects are called participatory decision models. These include:

## USING CRITICAL SOFT SKILLS

### PARTICIPATORY DECISION MODELS CONTD

**SIMPLE VOTING:** This is a simple approach that lets the team vote “for” or “against” an idea by a show of hands. Although this is an easy technique, it limits our opportunities to refine the resulting decision. This method can prevent the team from discussing better alternatives.

**THUMBS UP/DOWN/SIDEWAYS:** Asking for a show of thumbs up, down, or sideways around the room is a more efficient way of achieving a simple vote while still allowing some time to discuss other options. With this technique, we ask those who are holding their thumb sideways why they cannot make up their mind. This approach is quicker than polling everyone in the group for input, since most people will have no concerns and will just want to move forward.

**JIM HIGHSMITH’S DECISION SPECTRUM:** Team members indicate how they feel about a decision by placing a check mark on a spectrum ranging from “fully in favor” to “mixed feelings” to “absolutely no, or veto.”

## USING CRITICAL SOFT SKILLS

### PARTICIPATORY DECISION MODELS CONTD

**JIM HIGHSMITH'S DECISION SPECTRUM CONTD:** Highsmith's model is effective because it allows people to both indicate their support for a decision and express their reservations at the same time. This method invites those who are not entirely in favor of an option to share their concerns.



**Fig 6: Jim Highsmith's Decision Spectrum**

## USING CRITICAL SOFT SKILLS

### PARTICIPATORY DECISION MODELS CONTD

**FIST-OF-FIVE VOTING:** Using the fist-of-five approach, people vote by showing number of fingers that indicates their degree of support. A popular version of this method uses finger votes to register the amount of resistance or objection to an idea. With this method, the number of fingers raised indicates the following:

- **One finger:** “I totally support this option.”
- **Two fingers:** “I support this option with some minor reservations that we probably don’t need to discuss.”
- **Three fingers:** “I have concerns that we need to discuss.”
- **Four fingers:** “I object and want to discuss the issue.”
- **Five fingers:** “Stop, I am against this decision.”

With participatory decision models, a key point to remember is, “not involved means not committed.” If people are not involved, they will not be committed to the decision and, ultimately, will not be committed to the project.

## LEADING EFFECTIVELY

Leadership is about tapping into people's intrinsic motivations. To be effective leaders, we need to discover why our people want to do things, understand what motivates them, and then align their project tasks and goals accordingly. It is when we align project objectives with personal objectives that we get higher levels of productivity.

### MANAGEMENT VERSUS LEADERSHIP

Management has a more mechanical focus than leadership; it is concerned with tasks, control, and speed. In contrast, leadership assumes a humanistic focus on people and purpose; it is more concerned with empowerment, effectiveness, and doing the right things.

## LEADING EFFECTIVELY

### MANAGEMENT VERSUS LEADERSHIP CONTD

The following table illustrates the differences between a management focus and leadership focus:

<b>Management Focus</b>	<b>Leadership Focus</b>
Tasks/things	People
Control	Empowerment
Efficiency	Effectiveness
Doing things right	Doing the right things
Speed	Direction
Practices	Principles
Command	Communication

We need the mechanics of management but to be truly effective, we need to layer leadership on top of those mechanics. We can best amplify team productivity through a combination of management and leadership.

# LEADING EFFECTIVELY

## SERVANT LEADERSHIP

Agile promotes a servant leadership model that recognizes that it is the team members, not the leader, coach, ScrumMaster, or project manager, who get the technical work done and achieve the business value.

The servant leadership approach redefines the leader's role in relation to the team. It focuses the leader on providing what the team members need, removing impediments to progress, and performing supporting tasks to maximize the team's productivity.



# LEADING EFFECTIVELY

## SERVANT LEADERSHIP CONTD

There are four primary duties a leader performs in this role of serving the team:

1. **Shield the team from interruptions:** Servant leaders need to isolate and protect the team members from diversions, interruptions, and requests for work that are not part of the project.

When business representatives are closely involved in a project, they tend to make side requests directly to the developers that would sidetrack the planned development effort. These requests should be done during the iteration planning meeting or submitted to the product owner, who manages the backlog. A servant leader reminds people about the designated channels so the team can maintain their focus on the iteration.

Although it is important to shield the team from internal diversions, the project manager must be especially vigilant in protecting the team from external diversions. Time fragmentation – breaking people away from focusing on the project work and moving them back and forth between initiatives – saps productivity.

## LEADING EFFECTIVELY

### SERVANT LEADERSHIP CONTD

**2. Remove impediments to progress:** Servant leaders need to clear obstacles from the team's path that would cause delay or nonvalue-adding work. These obstacles may include wasted work or compliance activities (efforts that do not directly contribute toward delivering business value) that divert the team from completing the objectives of the current iteration.

Removing or easing such impediments will allow the development team to work faster and ultimately deliver more value to the business.

**3. (Re) Communicate project vision:** Communicating and re-communicating the project vision is critical to successfully leading a team. Only if stakeholders have a clear image of the goals for the completed product and project can they align their decisions with, and work toward, the common project objective.

# LEADING EFFECTIVELY

## SERVANT LEADERSHIP CONTD

The most effective leaders (level 5 leaders) dedicate a much higher percentage of their work time to communicating and re-communicating project and corporate vision than do people in the lower leadership levels.

So agile projects should not just have a vision exercise at the project kickoff or when developing the iteration goals, servant leaders need to continually look for opportunities to communicate the project vision and find new ways to illustrate and reinforce that vision.

**4. Carry food and water:** This is about providing the essential resources a team needs to keep them nourished and productive. Such resources could include tools, compensation, and encouragement. Leaders need to learn what motivates their team members as individuals and find ways to reward them for good work.

# LEADING EFFECTIVELY

## SERVANT LEADERSHIP CONTD

Leaders also need to celebrate victories – the large ones, of course, but also the small ones – as the project progresses. Celebrations and recognition help build momentum, and leaders need to nourish their teams with such rewards frequently to keep the project moving forward productively.

Training and other professional development activities are also examples of resources the team may need to be productive. By building the team members' skills, the project will not only gain the benefits of their new knowledge, but such actions also show that we want the team members to grow as individuals, not just extract work and information from them.

## LEADING EFFECTIVELY

### TWELVE PRINCIPLES FOR LEADING AGILE PROJECTS

1. Learn the team member's needs.
2. Learn the project's requirements.
3. Act for the simultaneous welfare of the team and the project.
4. Create an environment of *functional accountability*.
5. Have a vision of the completed project.
6. Use the project vision to drive your own behavior.
7. Serve as the central figure in successful project team development.
8. Recognize team conflict as a positive step.
9. Manage with an eye toward ethics.
10. Remember that ethics is not an afterthought, but an integral part of our thinking.
11. Take time to reflect on the project.
12. Develop the trick of *thinking backwards*.

Thinking backwards means we visualize the end goal and then work backwards to determine what needed to happen to get there and what problems and risks may have occurred.

# LEADING EFFECTIVELY

## LEADERSHIP TOOLS AND TECHNIQUES

The leadership tools and techniques employed on agile projects involve taking a soft-skills approach, rather than a directing, command-and-control project structure. Leadership is getting people to want to do what needs to be done. Instead of telling people what to do, we need to create an environment where people want to do what needs to be done.

As leaders, we can help create a productive project environment by using practices like modeling the behavior we want the team to follow, using different communication tools to express the project vision, enabling stakeholders to act, and being willing to challenge the status quo.

# LEADING EFFECTIVELY

## LEADERSHIP TOOLS AND TECHNIQUES

### MODELING DESIRED BEHAVIOR

**HONESTY:** People will not follow leaders they know are deceptive, since doing so undermines their own feelings of self-worth. Therefore, leaders should pay special attention to transparency and make sure they follow through on what they say they will do. So we shouldn't hide our mistakes – we should admit them openly. This is not only a healthy approach for us as leaders, but it also sets an example for how we want our team to operate.

**FORWARD-THINKING:** People expect those who lead them to understand where they are going. Leaders should be able to paint the picture for the team so everyone understands what they are ultimately aiming for.

# LEADING EFFECTIVELY

## LEADERSHIP TOOLS AND TECHNIQUES

### MODELING DESIRED BEHAVIOR

**COMPETENT:** Leaders do not need to have the strongest technical skills on the team, since the team members are typically happy to provide specialist knowledge when required. But leaders should be competent and not be an embarrassment or liability to the group.

**INSPIRING:** People want to be inspired in their work, rather than be met each day with a sense of doom and gloom. Therefore, leaders need to find ways to explain the project's vision and journey with genuine enthusiasm and spirit.

When we embody these traits as leaders, not only do we encourage people to follow us, we also model the behaviors that we want our team members to emulate. We are, in effect, leading by example.



# LEADING EFFECTIVELY

## LEADERSHIP TOOLS AND TECHNIQUES

**COMMUNICATING THE PROJECT VISION:** A leader can use a variety of practices to communicate and re-communicate the project vision to keep stakeholders aligned with the project objectives, based on what is most effective for the particular team. Whatever the method used, it is important to frequently communicate the project's goals and objectives to ensure that all stakeholders are aware of and aligned with the vision.

**ENABLING OTHERS TO ACT:** In order to enable our team members to feel confident in making decisions and taking actions that move the project forward in a productive way, we need to foster a collaborative environment. This involves building trust among team members and strengthening others by sharing power. We also need to create a safe working environment where people are not afraid to ask what they may think are dumb questions. People learn much more quickly when they can raise questions without fear of reprisal or ridicule.

# LEADING EFFECTIVELY

## LEADERSHIP TOOLS AND TECHNIQUES

Strengthening others by sharing power means the project manager, ScrumMaster, or leader does not keep the project plan or estimates to him- or herself. Instead, the leader makes sure that information and knowledge are spread throughout the team.

**BEING WILLING TO CHALLENGE THE STATUS QUO:** Challenging the status quo means we search for innovative ways to change, grow, and improve and then experiment and take risks by constantly generating small wins and learning from our mistakes. Iterations are perfect microcosms for experimentation.

Allowing stakeholders to suggest new ideas for improvement and then giving them a chance to try out those ideas is one way to cement the concept that everyone's ideas have value.

As leaders, we need to encourage our team to challenge the status quo of how we operate, not only because the team members are in a great position to suggest process improvements, but also because doing so helps to motivate them.