

## Lecture 04

### Skills and Characteristics of an Effective RA

This chapter describes the desired skills and characteristics of an effective RA. As emphasized in the previous chapter, the RA fulfills several critical project roles. On many projects, the RA is a part-time individual who is otherwise engaged as PM, product manager, system engineer, developer, or in some other capacity. On other projects, there may be a full time RA or even several RAs and a requirements manager. The size of the project and the perceived complexity of the needed requirements-related activities, as well as the funding available, are the major determinants of the number of RAs and their needed skill levels. The roles of the RA may be divided among those available to do the needed work, also considering current skills, interests, and desired development needs. Whatever the situation, RAs should consider themselves key resources, able to contribute to the project in the roles described in the previous chapter. The RA requires a unique blend of skills that reflects knowledge and real-world orientation, as well as the ability to interpret and satisfy customers', users', and management's intent. Some of the skills are intrinsic in the way an individual works (such as analytical and interpersonal skills) and others are learned (e.g., facilitation skills).

### Skills of the RA

As a framework for this chapter, refer to Table 3.1, RA's Skills Matrix.<sup>1</sup> A list of RA skills is provided. Three levels of RAs are shown:

1. With thanks to senior RA Michael Davis of Northrop Grumman IT for providing this artifact. The responsibility for modifications to the original version is mine.

**Table 3.1** RA's Skills Matrix

Line Number	RA's Skills Matrix	Reference	Entry/Junior-Level Analyst	Mid-Level Analyst	Senior-Level Analyst
1	Types of requirements	Ch. 4	K	X	X

## The islamia university of Bahawalpur

2	Criteria of a good requirement	Ch. 1	K	X	X
3	Customer/user involvement with requirements joint team	Ch. 1	K	X	X
4	Identifying real requirements (from the stated requirements)	Ch. 1	K	X	X
5	Anticipating and controlling requirements changes	Ch. 1	K	X	X
6	Office automation tools	Tutorials	X	X	X
7	References concerning requirements (books, articles, standards)	See Bibliography	K	X	X
8	Requirements attributes	Ch. 5	K	X	X
9	Requirements baseline	Ch. 6	K	X	X
10	Training in systems engineering (e.g., life cycles, risk management)	Ch. 5	K	X	X
11	Requirements justification/rationale	Ch. 5	K	X	X
12	Requirements management tools (e.g., DOORS, RequisitePro)	Ch. 5	K	X	X
13	Requirements peer review/inspection/walk-through	Ch. 5	K	X	X
14	Requirements syntax	WBR, Ch. 7	K	X	X
15	Requirements traceability	Ch. 5	K	X	X
16	Requirements verification and validation (V&V)	Ch. 5	K	X	X
17	System/subsystem/software-level requirements	Ch. 5	K	X	X
18	Developing and using metrics for requirements activities/processes	Ch. 2	K	X	X
19	Technical writing of requirements deliverables (RTM, SRS, IRS)	Ch. 4	K	X	X
20	Development, implementation, and use of requirements processes	Ch. 5		K	X
21	Familiarity with Microsoft Project	Tutorial		K	X
22	QA of requirements	Ch. 9		K	X
23	Requirements allocation (to components, applications, packages)	Ch. 4		K	X
24	Requirements change control and change notification	Ch. 6		K	X
25	Requirements repository	Ch. 5		K	X
26	Requirements errors (missing, incorrect, infeasible, out of scope)	Ch. 6		K	X
27	Requirements defect notification	Ch. 6		K	X
28	Requirements dissemination to customers/users/developers/testers	Ch. 4		K	X

**Table 3.1** RA's Skills Matrix (continued)

<i>Line Number</i>	<i>RA's Skills Matrix</i>	<i>Reference</i>	<i>Entry/Junior Level Analyst</i>	<i>Mid-Level Analyst</i>	<i>Senior-Level Analyst</i>
--------------------	---------------------------	------------------	-----------------------------------	--------------------------	-----------------------------

29	Requirements elicitation	Ch. 5	K	X	
30	Requirements identification	Ch. 5	K	X	
31	Use case development (with Xcustomer/user and based on user's guides)	Ch. 7	K		
32	Requirements in customer/user Xdecision-making process	Ch. 1	K		
33	Requirements interaction with	CM	Ch. 6	X	X
34	Requirements negotiation	SL, EG1	X	X	
35	Requirements ownership	WBR, EG2	X	X	
36	Requirements prioritization	Ch. 5	X	X	
37	Requirements review board X(RRB)/configuration review board (CRB)/configuration control board (CCB)	Ch. 7	X		
38	Requirements Xrough-order-of-magnitude (ROM) costs	Ch. 7	X		
39	Requirements specifications	Ch. 7	X	X	
40	Evaluating requirements for risks	Ch. 7		X	
41	Training the requirements Xprocesses	Ch. 5			
42	Requirements impact estimation Gilb X(IE) table				

Knowledge of = K Experience with = X

References:

REH= Young, R. R., *The Requirements Engineering Handbook*, Norwood, MA: Artech House, 2004.

WBR= Alexander, I. F., and R. Stevens, *Writing Better Requirements*, Boston: Addison-Wesley, 2002.

SL= Lauesen, S. *Software Requirements: Styles and Techniques*, pp. 346–347.

EG1= Gottesdiener, E., *Requirements by Collaboration: Workshops for Defining Needs*, Reading, MA: Addison-Wesley, 2002, pp. 122–128.

EG2= Gottesdiener, E., *Requirements by Collaboration: Workshops for Defining Needs*, Reading, MA: Addison-Wesley, 2002, pp. 89–94.

Gilb: See material at [www.result-planning.com](http://www.result-planning.com).

1. Entry/junior-level analyst;
2. Mid-level analyst;
3. Senior-level analyst.

A “K” is used in this table to suggest that knowledge of the skill is needed at a particular level of analyst expertise. An “X” suggests that experience in using the skill is needed at a particular level. A mapping is provided to

sections in this book or to other sources where each skill is addressed. Additional references are provided in those places.

As with any framework or model, use this matrix as a guide, not as a specification. It will help you evaluate your suitability for a project role and

provide a guide to and resource for strengthening and improving your skills.

Obviously, one could add other skills. However, the matrix serves as a guide and suggests that there is a lot to learn in order to be an effective RA. It is one thing to read about (gain knowledge of) skills and quite another thing to garner experience in applying the skills in a project environment that involves actual customers and users.

A junior or entry-level analyst (those with less than two years of experience) should be familiar with the following:

The types of requirements (described in detail in the next chapter);

The criteria of a good requirement (provided in Chapter 1);

Office automation tools [e.g., Microsoft (MS) Office or Corel WordPerfect suite];

The concept of using a requirements process;

Some of the references concerning requirements-related activities;

The purpose of requirements verification, and so forth.

She should understand that a rationale should be provided for each requirement (why the requirement is needed in the system or software).

A mid-level analyst (those with two to four years of experience) should have knowledge of more of the aspects and activities of requirements engineering, coupled with additional experience in applying this knowledge. High on this list are requirements activities involving customers and users (such as the concept of a joint team), utilizing a requirements process, and familiarity with an industry-strength requirements tool. The mid-level analyst should be proficient with peer reviews and inspections and should ensure that all of her own work products are peer reviewed. She should understand the value of bidirectional traceability of the requirements and be learning how to develop a requirements traceability matrix (RTM).

A senior-level analyst (those with five or more years of experience performing requirements-related activities) should have both knowledge of and experience with using all of the skills in the matrix. She should be familiar with all of the roles described in the previous chapter and have well-developed interpersonal skills and characteristics as described later in this chapter. She should understand the value and importance of independent QA and have a thorough understanding of CM activities. She should be able to recommend and use requirements metrics and be able to apply metrics to requirements processes. She should be able to provide training sessions for more junior RAs and for other members of the project team. She should have a good familiarity with systems engineering and the system life

2 to 4 years  
of experience

cycle and an understanding of the many requirements-related activities that need to be performed throughout the system life cycle.

Figure 3.1 summarizes the progression of the RA.

Another important document is the RA's position or job description, provided in Table 3.2.<sup>2</sup>

4 or more years  
of experience

This is a concise and useful summary of the role of the RA. It describes the position, summarizes skills that are needed (although not in as much detail or precision as Figure 3.1), indicates knowledge that is needed, suggests several responsibilities, indicates some measures of performance, and provides three useful references on which this artifact is based. I suggest that you utilize this artifact to clarify your role and to develop position requisitions of prospective RAs. Tailor it to reflect your responsibilities. Utilize it in your performance reviews to discuss professional-development activities that will enhance your skills with your manager.

Senior-level analyst

Mid-level analyst

# The islamia university of Bahawalpur

- Is familiar with a requirements process and an RTM;
- Is familiar with automated requirements tools;
- Is able to facilitate requirements definition activities between developers and customers/users;
- Applies peer reviews and/or inspections to requirements development efforts;
- Understands the value of bidirectional requirements traceability.

Junior or entry-level analyst

- Knows the types of requirements
- Knows the criteria of a good requirement
- Understands how to provide rationale for a requirement;
- Has studied related references;
- Knows the purpose of requirements verification
- Is familiar with office automation tools
- Has a good understanding of the roles of the RA;
- Is familiar with all roles described in Chapter 2;
- Experienced in full life cycle activities;
- Well-developed interpersonal skills and characteristics;
- Has a through understanding of CM activities;
- Understands the value and importance of independent QA;
- Able to provide requirements-related training to more junior RAs and other project members.

0 to 2 years

of experience

**Figure 3.1** Professional growth of the RA is based on cumulative experiences. (*Adapted from: Michael Davis.*)

2. With thanks to Karl Wieggers for allowing me to participate in the development of versions of this artifact.

**Table 3.2** RA Job Description

<i>Description</i>	<p>The RA or engineer is the individual who has the primary responsibility to elicit, analyze, validate, specify, verify, and manage the real needs of the project stakeholders, including customers and end users. The RA/engineer is also known as a requirements manager, business analyst, system analyst, or, simply, analyst.</p> <p>The RA serves as the conduit between the customer community and the software development team through which requirements flow.</p> <p>An RA is involved at some level throughout the entire system or software development life cycle. Upon establishment of the requirements baseline, the focus is shifted towards the management of the requirements specification and verifying the fulfillment of all requirements.</p> <p>The requirements engineering function is a project role, not necessarily a job title. The role may be performed by a dedicated RA or split among multiple team members who have other primary job functions, such as a PM or product developer. The RA is responsible for ensuring that the tasks are performed properly.</p>
<i>Skills Needed</i>	<p>Interviewing skills to talk with individuals and groups about their needs and ask the right questions to surface essential requirements information.</p> <p>Listening skills to understand what people say and to detect what they might be hesitant to say.</p> <p>Analytical skills to evaluate critically the information gathered from multiple sources, reconcile conflicts, decompose high-level information into details, abstract up from low-level information to a more general understanding, distinguish presented user requests from the underlying true needs, and distinguish solution ideas from requirements.</p> <p>Facilitation skills to lead requirements elicitation workshops.</p> <p>Observational skills to validate data obtained via other techniques and expose new areas for elicitation.</p> <p>Writing skills to communicate information effectively to customers, marketing, managers, and technical staff.</p> <p>Organizational skills to work with the vast array of information gathered during elicitation and analysis and to cope with rapidly changing information.</p> <p>Interpersonal skills to help negotiate priorities and to resolve conflicts among project stakeholders (such as customers, product management, and engineering).</p> <p>Modeling skills to represent requirements information in graphical forms that augment textual representations in natural language, including using modeling languages already established in the development organization.</p>
<i>Knowledge Needed</i>	<p>An understanding of contemporary requirements elicitation, analysis, specification, verification, and management practices and the ability to apply them in practice.</p> <p>Familiarity with requirements engineering tools and other resources.</p> <p>An understanding of how to practice requirements engineering according to several software development life cycles in a team environment.</p> <p>Knowledge of product management concepts and how enterprise software products are positioned and developed.</p> <p>Application domain knowledge is a plus to have credibility with user representatives and be able to work effectively with them.</p>
<i>Responsibilities</i>	<p>Work with the PM, product manager, or project sponsor to document the product's vision and scope.</p> <p>Identify project stakeholders and user classes, document user class characteristics, and identify appropriate representatives for each user class and negotiate their responsibilities.</p>

**Table 3.2** RA Job Description (continued)

<i>Responsibilities</i>	<p>Elicit requirements using interviews, document analysis, requirements workshops, storyboards, surveys, site visits, business process descriptions, use cases, scenarios, event lists, business analysis, competitive product analysis, task and workflow analysis, and viewpoints.</p> <p>Write requirements specifications according to standard templates, using natural language simply, clearly, unambiguously, and concisely.</p> <p>Decompose high-level business and user requirements into functional requirements and quality requirements, specified in an appropriate level of detail suitable for use by those who must base their work on the requirements.</p> <p>Define quality attributes, external interfaces, constraints, and other nonfunctional requirements.</p> <p>Represent requirements using alternative views, such as analysis models (diagrams), prototypes, or scenarios, where appropriate.</p> <p>Lead requirements analysis and verification, ensuring that requirement statements are complete, consistent, concise, comprehensible, traceable, feasible, unambiguous, and verifiable and that they conform to standards.</p> <p>Participate in requirements prioritization.</p> <p>Participate in peer reviews and inspections of requirements documents.</p> <p>Participate in peer reviews of work products derived from requirements specifications to ensure that the requirements were interpreted correctly.</p> <p>Enter, manipulate, and report on requirements stored in a commercial requirements tool.</p> <p>Define requirement attributes and facilitate their use throughout the project.</p> <p>Manage requirements traceability information and track requirements status throughout the project.</p> <p>Identify requirements errors and defects, and write requirements defect identification and notification reports.</p> <p>Manage changes to baselined requirements through effective application of change control processes and tools.</p> <p>Establish and implement effective requirements practices, including use and continuous improvement of a requirements process.</p> <p>Assist with the development of the organization's requirements engineering policies, procedures, and tools.</p> <p>Implement ways to reuse requirements across projects.</p> <p>Identify ways to assist product management in product planning through requirements development and analysis.</p> <p>Propose new product features and updates.</p>
<i>Measures of and Performance</i>	<p>Evaluation from product and project management on overall product quality effectiveness in the marketplace of the requirements after the product has been developed.</p> <p>Feedback from key customer or marketing representatives on the way in which the requirements engineering process was conducted.</p> <p>Customer satisfaction measures.</p> <p>Satisfying or exceeding requirements development schedules, resource constraints, and quality goals.</p> <p>Control of requirements creep attributable to missed requirements and leakage of "unofficial" requirements into the project.</p>

**Table 3.2** RA Job Description (continued)



- |            |  |
|------------|--|
| References | Ferdinandi, Patricia L., <i>A Requirements Pattern: Succeeding in the Internet Economy</i> , Boston: Addison-Wesley, 2002, Chapter 8.<br>Wiegiers, Karl, "The Habits of Effective Analysts," <i>Software Development</i> 8(10) (October 2000): 62–65.<br>Young, R. R., <i>Effective Requirements Practices</i> , Boston, MA: Addison-Wesley, 2001, Chapters 4 and 5. |
|------------|--|

Notes:

Each team that uses this job description needs to weight the various skills and knowledge that are pertinent to its job. Certain skills listed might be critical for one requirements engineer job and unimportant for another.

Each person considering hiring an individual to be a requirements engineer needs to consider which of these skills are intrinsic to the way the individual works (e.g., analytical and interpersonal skills) and which can be learned (e.g., facilitation and listening skills).

The users of this generic job description will need to modify some of the terminology to reflect their specific environments (e.g., corporate information-systems development, commercial product development, contract development).

This job description needs to be tailored to match the experience level for the position. Source: Karl Wiegiers et al.

## Characteristics of an Effective RA

In addition to learned, or “hard,” skills, there is a set of personal characteristics that will serve the RA well. You may feel that some of these characteristics are themselves really skills. I won’t argue this with you—let’s agree that all of the skills and characteristics noted are helpful and useful. Table 3.3 summarizes the desired characteristics described below as countermeasures you can apply to overcome barriers you are likely to encounter.

Table 3.4 provides suggestions for how to strengthen these characteristics.

Consider the following characteristics, which you may choose to continue to refine, as well as the suggestions and resources proffered to help.

1. Engage in continuing education to acquire expert knowledge of requirements engineering and requirements practices. Chapter 1 described the many components of the requirements process, a set of activities that are performed throughout the system life cycle of a project. My earlier book [1] provides a comprehensive set of references in the requirements literature as of 2001 (many more recent references are provided in this book). For each of the 10 recommended practices described in the earlier book, a few key references are provided at the end of each chapter, together with a short summary of the information provided by the reference. Ongoing study, such as attending training seminars in areas related to expertise, assignment, and activities, is helpful. Journals such as *IEEE Software*, *CrossTalk*, *Software Development Magazine*, and *INSIGHT* provide informative articles and reviews of related books that you might purchase and study. These are both informative and motivational—they

**Table 3.3** Desired RA Characteristics as Countermeasures to Likely Barriers

Barriers You May Encounter	Characteristics as Countermeasures
----------------------------	------------------------------------

[A] Lacking a thorough knowledge of requirements engineering, requirements processes, and requirements errors methodologies can cause the RA to be less effective than is needed.	[1] Engage in continuing education to acquire expert knowledge of requirements engineering and practices. [7] Initiate learning, applying, and using effective practices; seek sponsorship for requirements-related activities from the PM; be committed to project success. [13] Maintain a good knowledge of evolving technology and how it can be applied to meet customer needs.
[B] The end product does not meet the Carefully customer's needs. There are different ideas and opinions as to what the real negotiation skills. requirements are.	[2] Be a good listener, communicator, and writer. document decisions and action items. [3] Have good facilitation and
	[4] Be persistent and persevering. [5] Be proactive in engaging customers and users, coworkers, and project management. [15] Desire to make a difference in your professional work.
[C] Management does not always with understand what is being built what resources are needed to achieve other the end product.	[6] Develop the ability to communicate effectively and management. [10] Develop the ability to estimate the time and resources required to accomplish technical work.
[D] The requirements process does not continuous support the project's needs.	[8] Develop and maintain an attitude of improvement. [14] Set achievable goals and meet them. define and describe methods to achieve the project's goals in a requirements plan. [16] Develop your ability to contribute to the project's risk process.
[E] Strong personalities and strong Carefully opinions can derail the effectiveness of good requirements development and management.	[2] Be a good listener, communicator, and writer. document decisions and action items. [9] Take responsibility for your views, attitudes, relationships, and actions, and maintain respect for others.
[F] People often try to do more than is the main called for and to make ad hoc changes new requirements during work product development. independently and	[11] Maintain focus on keeping the main thing thing. Install a mechanism to control and changes. Don't invent requirements avoid gold plating. Avoid requirements creep.
[G] Project personnel can become too provide vested in the work product solution to people who are analyze and decompose requirements close to the problem and the legacy system. effectively.	[12] Develop the ability to think outside the box to creative approaches that might not occur to

Source: Richard Raphael.

provide encouragement to strengthen one's own understanding. Also, there are several Web sites that offer reviews of requirements-related books (see for example Ian Alexander's Web site [2]) and "goodies" (reusable requirements-related artifacts) available at Karl Wiegers' Web site [3]. Attending conferences such as the annual Institute of Electrical and Electronics Engineers (IEEE) Conference on Requirements Engineering [4] or the annual conference

**Table 3.4** Characteristics of an Effective RA and Suggested Activities to Strengthen Them

## The islamia university of Bahawalpur

<i>Characteristic</i>	<i>Suggested Activities</i>
1. Engage in meetings and continuing education. professional associations.	Read requirements engineering literature, attend professional conferences, visit Web sites, hold office in
2. Good listener, practice communicator, and writer.	Attend seminars in listening skills, communications, and writing; making presentations and writing.
3. Good facilitation managing and negotiation skills.	Practice facilitating meetings, coordinating workshops, and process design sessions.
4. Persistent and requirements. persevering.	Practice evolving real requirements from stated
5. Proactive i.n suggestions engaging others. approaches for	In performing daily assignments, think deliberately about (1) for making things better, and (2) appropriate venues and making them. Practice. Ask for feedback, and act on it.
6. Ability to your communicate perspective and effectively with modify management.	Practice looking at your responsibilities from the perspective of manager and senior management. Write down your management's perspective. Work to understand differences and your communications accordingly.
7. Learn, apply and use effective practices.	Select a practice that you believe will improve a work situation. Gather support for trying it out ("piloting" it). Consider steps you and the project or organization can take to give the pilot the best chance of success. Implement the practice. Follow through to ensure it takes. Assess the value of implementing the practice after one month.
8. Develop and of maintain an attitude up on of continuous other improvement. such as	Practice the Plan-Do-Check-Act (PDCA) cycle at the conclusion meetings. Document the suggestions that are offered. Follow suggestions to the extent feasible and possible. If this works, explore opportunities to inculcate an attitude of continuous improvement, by documenting processes and improving them.
9. Take responsibility for your views, good attitudes, relationships, and that you have worked toward learning from the error. Also, work to and actions.	Make known to coworkers an error you have made, in the spirit of contributing to doing things better. Convey that your intentions were recognize the value provided by all coworkers.
10. Develop the ability work tasks to estimate work assigned to you. Track the actual time consumed, noting distractions. requirements. Consider changes you might make in your work habits to be more	Estimate the time you think you will require to accomplish assigned to you. Track the actual time consumed, requirements. Consider changes you might make in your productive. Over time, try to have estimates be closer to actuals.
11. Maintain focus. are	Embrace the concept of real requirements. Understand how these different from stated requirements. Suggest prioritizing requirements on your project, and evolve an approach to collaboratively prioritizing a set of requirements. Evaluate the impact of this approach.
12. Strengthen your ability to think outside three the box. Consider the	Meet with stakeholders to consider previously unconsidered potential solutions to vexing problems. Use the brainstorming technique to get ideas from each participant. Multivote on the ideas suggested. potential value in seriously pursuing one or more ideas.
13. Strengthen your system knowledge of available architect and other "technologists." Discuss possible ways to accomplish technology.	Schedule a brown bag to consider technology possibilities. Invite a some system objectives by incorporating new technologies.
14. Set achievable some goals and meet them. accomplish. Keep these	Plan your work for the next month. Set a few specific objectives for things that you believe are really important to specific objectives foremost in mind over the next month. Manage to the specific objectives (i.e., ensure that you accomplish them).

15. Strive to make a difference in your work. Explore with your manager how the tasks for which you are responsible at work might make a difference to the project or organization. Identify some work situations, specific achievements, and then pursue them in earnest. Solicit your coworkers' and your manager's support in achieving them.

**Table 3.4** Characteristics of an Effective RA and Suggested Activities to Strengthen Them (continued)

Characteristic	Suggested Activities
16. Contribute to your project's risk process. Initiating a risk	Volunteer to serve on the project's risk management team. If your project does not have one, suggest that the project consider process. Identify the top risks, prioritize them, and develop risk mitigation plans for those considered to be the most serious risks. Monitor the risks.

Source: Richard Raphael.

of the International Council on Systems Engineering (INCOSE) [5] is another way to strengthen knowledge, learn about old and new techniques, meet others who are working in this area, and find out about the latest offerings available from vendors. Consider becoming a member and active participant in professional associations and societies such as INCOSE, IEEE, the American Society for Quality (ASQ), the Society for Software Quality (SSQ), the International Association of Facilitators (or associated local organizations), and the Requirements Engineering Specialist Group (RESG) in Europe [6]. Often, professional organizations offer evening meetings, lunchtime sessions, or Saturday tutorials that provide opportunities to learn and to meet colleagues. For example, the Washington, D.C., chapter of INCOSE provides superb opportunities to share experiences, glean lessons learned, and find sources of information, as do many other local and regional chapters [7]. Actively seek opportunities to write articles and make presentations. The old adage that no one learns more than the author, teacher, or presenter is true.

2. Be a good listener, communicator, and writer. Good communication skills are important. It is important to understand the needs and expectations of different stakeholders. Learn to listen carefully so as to hear what users and customers are trying to say, even if they aren't very good at expressing it. You need to be able to verify understanding by repeating back your interpretation of statements. You need to be able to write clearly and concisely so that the requirements are documented according to the criteria of a good requirement provided in Chapter 1. Steven Gaffney's Web site and his seminars have been valuable resources for me—see the workshops and materials available at his Web site [8].
3. Have good facilitation and negotiation skills. Among the most effective requirements gathering techniques is the requirements workshop. See Ellen Gottesdiener's *Requirements by Collaboration* [9] for a thorough treatment of this important technique. The RA often may find him- or herself facilitating groups of people in such venues. It's important to be able to encourage the identification of ideas, while not allowing one or a few people to dominate the discussion. Often, you'll find yourself needing to negotiate to achieve consensus

among individuals with divergent views. There are workshops you can attend to learn and hone these skills. As noted above, there are professional resources available to strengthen facilitation skills. See

the Web site for the International Association of Facilitators for ideas [10].

4. Be persistent and persevering. Since customers and users provide us their stated requirements, it's vital that RAs be persistent and persevering so that the real requirements are evolved. It's not enough to depend on being able to proffer the excuse that "we built the system that you requested." If the stated requirements are not acceptable, do not wait until you have completed the system and the users reject it. Cut the risks to your project by improving the requirements as early as possible. The risks include doing wasted technical work, and of course having the system rejected (with all the legal and commercial risks that result). Identifying the real requirements is the one most important thing that the RA can do to contribute the most to customers.
5. Be proactive in engaging customers and users, coworkers, and project management. You'll soon appreciate that it's not enough just to go with the flow. The performance of the RA's roles demands that you be proactive. Customers and users need your initiative and persistence to help them evolve the real requirements. Your coworkers need your proactive support to help them select and use effective processes, practices, methods, techniques, and tools. Project management needs you to speak for approaches that will best serve the project, for example, investing more in the requirements process, identifying the real requirements, and providing a mechanism to control new requirements and changes to requirements.
6. Develop the ability to communicate effectively with management. Too often, differences in perspective prevent good communication. Management views information technology (IT) as a means to achieve business objectives. Systems and software engineers view their work in terms of work products that must meet specified requirements. As noted earlier, it furthers neither your career nor the project or organization to say yes, when impossible commitments only guarantee failure in the future. Dorothy McKinney offers suggestions in her article "Six Translations between Software-Speak and Management-Speak" [11]. Another insight is that sometimes RAs must concern themselves not only with their own management, but also with the customer's management.
7. Initiate learning and applying effective practices, and be committed to project success. One needs to be willing to learn and use effective practices. Learning comes from experience and study. Applying practices on projects requires training to familiarize people with them; mentoring people in their use; tracking their effectiveness; and ensuring that their deployment and use is effective. A serious

problem is that, most often, new and improved practices aren't given a real chance because it is human nature to revert to using practices that are already in place.<sup>3</sup> One needs to be committed to project success and to advising when things need to be done differently or

better.

8. Develop and maintain an attitude of continuous improvement. Related to the commitment described above is the idea of maintaining an attitude of continuous improvement. The RA should encourage the project to embrace mechanisms to instill the attitude of “getting ever better.” At my company, we end every meeting with an evaluation of how the meeting went: what worked and what could have been done better. We call this “doing PDCA” in honor of Dr. Deming and Walter Shewhart’s contributions to continuous process improvement and our adoption of their teachings. Similarly, at the end of each cycle of activities, consider having a workshop to gather feedback concerning how things went. Use the ideas and suggestions generated to improve how the work is done (i.e., to improve the process that is being used). These mechanisms (in addition to providing good ideas) serve to help everyone buy in to the procedures used, because participants help shape them—“I helped improve that process!”
9. Take responsibility for your views, attitudes, relationships, and actions. By taking responsibility, one establishes a sense of accountability. You’ll tend to exhibit pride in your work. You’ll not let personalities and individual characteristics deter you from having good relationships with everyone. Your actions will make a valued contribution. You will be setting an example for others. You’ll be a leader.
10. Develop the ability to estimate the time and other resources required to accomplish technical work. One of the difficulties in making estimates of technical work is that these estimates are needed early in order to develop projections of the number of staff required to complete the project. (The number of staff, their seniority, and their roles are required to develop an estimate of the cost of the project.) The difficulty is compounded by the fact that the real requirements are not yet known. So, we often find ourselves making estimates without an accurate basis for them. This can lead to a lot of work that is not productive and also to confusion caused by the inability downstream to meet the estimates.

The RA can contribute to the estimation process by (1) working with users in the joint team environment to identify the real

3. See Watts Humphrey’s “Why Don’t They Practice What We Preach” for insights concerning this problem and suggestions for how to deal with it. See [www.sei.cmu.edu/publications/articles/practice-preach/practice-preach.html](http://www.sei.cmu.edu/publications/articles/practice-preach/practice-preach.html).

requirements, and (2) working with PMs and the development staff to make estimates of the time and other resources required to accomplish the technical work. Using data based on previous experiences (“managing by fact”) is best.

Coworker John E. Moore in the Defense Enterprise Solutions



(DES) business unit at Northrop Grumman IT is a valuable resource. As project management (project planning, project tracking, and integrated product management) “process owner,” Dr. Moore has developed a “Brickchart” capability within MS Project that facilitates tracking progress for tasks.<sup>4</sup> Another coworker, Rich Raphael, developed the Risk Manager’s Assistant (RMA), a straightforward database tool that supports standard risk management processes and programs.<sup>5</sup> Both of these are easily learned and useful project-management tools. See the discussion in Chapter 5 concerning risk management. As an RA gets more experienced performing requirements analysis, the RA should also be evaluating each new or changed requirement for any risk that it may add to the project. As projects get more complex and as customers become more mature in specifying their needs, each new or changing requirement risks adverse impact on the project. Note that the CMMI<sup>®</sup>, as part of the introduction to the requirements management (REQM) process area (PA), specifies that one should “refer to the risk management (RI) process area for more information about identifying and handling risks associated with requirements.”

11. Maintain focus on keeping the main thing the main thing. One of the pitfalls in developing systems and software is that we try to do too much; another is that we try to incorporate changes as we work. Customers and users will ask, Can you do this? Will the new system do that? We don’t like to say no. We participate in creating a perception that the new system will be all things to all stakeholders. In so doing, we jeopardize our ability to fulfill these commitments and the success of the effort.

The RA can serve a critical role here. Very early, the RA should facilitate establishing the concept that all requirements are not equally important and that it’s the responsibility of all stakeholders to prioritize needs collaboratively and to focus the intent of the project (to keep the main thing the main thing). As suggested by Neal Whitten in “Meet Minimum Requirements: Anything More Is Too Much” [12], the RA should work to identify the minimum set of requirements required to accomplish the business objectives. This goal can be facilitated by doing the following:

4. Contact Dr. Moore at [john.moore@ngc.com](mailto:john.moore@ngc.com).

5. Contact Mr. Raphael at [RRaphael@ngc.com](mailto:RRaphael@ngc.com).

Establishing and following a process of prioritizing all requirements;

Establishing the concept of follow-on releases or versions that will address lower-priority requirements and requirements that are identified during development of increments of functionality later in the development process;

Ensuring that a mechanism is put in place and used to control new



requirements and changes to requirements.

These activities can have an enormous positive impact on keeping the train on the tracks. One of the major causes of rework is changes introduced after technical work has been completed or is well underway.

Another aspect of this desired characteristic is that the RA should not invent requirements independently and should avoid “gold plating,” that is, adding features and capabilities to systems and software when they are not required by the real requirements. The RA or developer might think he knows something that will be “way cool” for the users that could turn out to be unwanted or very disruptive to the project (e.g., if increased costs are incurred to provide it).

12. Develop the ability to think outside the box to provide creative approaches that might not occur to people who are close to the problem and the legacy system. One of the advantages an RA brings to a new assignment is that he does not have the same vested interest that a user or customer has and, therefore, can act as an impartial or unbiased agent. The RA arrives without expectations, without necessarily having much knowledge of the domain, and without being attached to any particular outcome. Unhindered by years of association with a problem domain and unconfined by the constraints of the legacy system, you are free to think more freely about what needs to be done and how it can be addressed best. Leverage these opportunities to think of new and different ways the system objectives might be addressed.
13. Maintain a good knowledge of evolving technology and how it can be applied to meet customer needs. Some experienced RAs believe that a strong technical background is very helpful for an RA. As mentioned earlier, understanding current technologies is not solely the responsibility of the RA, but we can contribute to the system design by involving architects in reviews of the requirements and by assisting them in developing technical solutions. Another reason this is important is that incorporating some new technologies creates new requirements that must be considered. Other experienced RAs believe that a strong technical background is not as important to the RA as the other characteristics, especially when eliciting requirements and understanding the real needs and expectations of customers and users. These people believe that a strong technical

perspective may actually inhibit the RA and that utilizing a person with a more general background is a better approach.

14. Set achievable goals and meet them. This is related to the characteristic of maintaining focus. The RA should set achievable goals and meet them. Having a documented requirements plan and process and following them will help.
15. Desire to make a difference in your professional work. We shouldn't be content to just go to work or to put in a set number of hours at our

jobs. Rather, it should be one of our values to want to make a difference in our professional work. It's vital that control of the project be maintained. Having this value affects others and inspires us to become increasingly effective in our own roles.

Occasionally, we find ourselves in a situation in which we are powerless to make a difference. For example, I participated on a project for a period of several months during which I sensed that I made an important, needed, and valued contribution. Suddenly, the PM seemed to withdraw his support for my role. I discussed the situation with him and was unable to change it. It was time for me to move on to a different project. Sometimes we need to take the responsibility for change and act on it.

16. Develop your ability to contribute to the project's risk process. Every project should have a risk process to identify, evaluate, prioritize, and mitigate existing or potential risks. Consider participating in your project's risk management team and process. Requirements-related risks are important to the project. You can contribute to the dialogue that will help your project deal with its risks successfully.

## Summary

Suggested skills of the RA are listed and categorized in an RA's skills matrix (Table 3.1) according to those needed by a junior-level, mid-level, or senior-level analyst. This matrix will help you evaluate your suitability for a specific project role. You may use it as a guide to further strengthen and improve your skills or as a reference to sources of information concerning each skill. Table 3.2 provided an RA job or position description that should help you clarify the many ways in which the role of the RA can be leveraged to benefit both your project and your organization. Making the RA's role explicit helps a project to run more smoothly. The RA's role needs to be understood and valued in the minds of PMs and the technical community—this job description should help! Sixteen characteristics of an effective RA were presented and described. Suggestions are provided concerning how to strengthen these characteristics. Consider these in the context of your own personal and professional development, as well as of your current assignments and responsibilities, and select one or a few characteristics to strengthen each year. Yes, being an effective RA involves learning many

skills and having many desired personal characteristics. This chapter, coupled with thoughtful introspection, should provide a useful road map.

## Case Study

A requirements engineering consultant was invited to assist a particular location of a large U.S. Government organization. Senior management at that location indicated, "millions of dollars had been wasted" in repeated efforts

to develop systems and software solutions internally. The consultant met with senior management, managers, users, and developers to gather information and gain an understanding of the situation. Following analysis and development of a tailored requirements course, he presented formal training for all stakeholders that addressed existing problems in the organization from the perspective of relevant industry experience. On the surface, there seemed to be a sincere desire on the part of all stakeholders to improve the situation, although many issues existed. The training addressed how these issues could be resolved. At the conclusion of the training, the senior manager concluded that the situation could not be improved. Many of the other participants in the training were perplexed by this conclusion: they felt they were off to a fresh start.

Analysis: The senior manager himself was the key issue that prevented the situation from improving. Although there were issues relating to all parties, management was willing to allow parochial interests of users, an overly bureaucratic development process, and power struggles of some key stakeholders to paralyze efforts and render improvement of the situation impossible. In Dr. Deming's framework, there were "too many red beads."<sup>6</sup> Users and the development organization were powerless to improve the situation without management's support and management's expectation for better results. Management must enable and empower its workers (all the rest of us) in order for work to be productive and effective. Industry studies report that lack of appropriate senior management support is a factor in most IT failures. This vignette has much to offer senior managers. Industry experience is that senior management must sponsor and support IT and systems/software development initiatives if they are to be successful. See the discussion in Chapter 8 and a recent *Harvard Business Review* article, "Six Decisions Your IT People Shouldn't Make," [13] for further insights and specific suggestions. The RA can be helpful here by offering these insights, suggestions, and industry experience to his or her management team and by helping to clarify the specific roles that senior management should provide.