TOPIC 2 – TYPES OF REQUIREMENTS

LEARNING OUTCOMES

By the end of this topics, you will be able to:
1. Describe the FR and NFR
2. Discuss the techniques might be use to elicit requirement
3. Explain on how to modelling in the requirement analysis

INTRODUCTION

The analysis phase is so named because the term analysis refers to breaking a whole into its parts with the intent of understanding the parts' nature, function, and interrelationships. In the context of the SDLC, the outputs of the planning phase (the system request, feasibility study, and project plan), outline the business goals for the new system, define the project's scope, assess project feasibility, and provide the initial work plan. These planning phase deliverables are the key inputs into the analysis phase. In the analysis phase, the systems analyst works extensively with the business users of the new system to understand their needs from the new system.

2.1 Functional Requirement And Non- Functional Requirement

A requirement is simply a statement of what the system must do or what characteristics it needs to have. During a systems development project, requirements will be created that describe what the business needs (business requirements); what the users need to do (user requirements); what the software should do (functional requirements); characteristics the system should have (nonfunctional requirements); and how the system should be built (system requirements). Although this list of requirement categories may seem intimidating at first, the categories merely reflect the purpose of the requirements and the stage in the SDLC in which they are defined.

We have already discussed the creation of the systems request in the planning phase of the SDLC. In the system request, there are statements that describe the reasons for proposing the systems development project. These statements reflect the business requirements that this system, if built, will fulfill. These business requirements help define the overall goals of the system and help clarify the contributions it will make to the organization's success. Examples of business requirements include: "Increase market share"; "Shorten order processing time"; "Reduce customer service costs"; "Lower inventory spoilage"; "Improve responsiveness to customer service requests"; and "Provide account access to mobile customers."

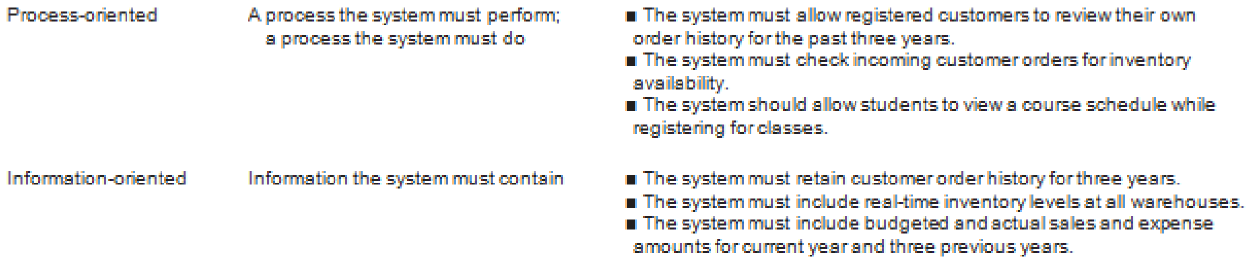
When the systems development project is complete, success will be measured by evaluating whether the stated business requirements have actually been achieved; therefore, they provide the overall direction for the project. During the analysis phase, requirements are written from the perspective of the business, and they focus on what the system needs to do in order to satisfy business user needs. A good starting place is to concentrate on what the user actually needs to accomplish with the system in order to fulfill a needed job or task. These user requirements describe tasks that the users perform as an integral part of the business' operations, such as: “Schedule a client appointment”; “Place a new customer order”; “Re-order inventory”; “Determine available credit”; and “Look up account balances.” Use cases (discussed in Chapter 4) are tools used to clarify the steps involved in performing these user tasks. By understanding what the user needs to do in terms of tasks to perform, the analyst can then determine ways in which the new system can support the users' needs.

Determining ways in which the new system can support user needs leads to statements of the system's functional requirements.

A functional requirement relates directly to a process the system has to perform as a part of supporting a user task and/or information it needs to provide as the user is performing a task. The International Institute of Business Analysis (IIBA) defines functional requirements as “the product capabilities, or things that a product must do for its users.” Functional requirements begin to define how the system will support the user in completing a task. For example, assume the user requirement is “Schedule a client appointment.” The functional requirements associated with that task include: “Determine client availability,” “Find available openings matching client availability,” “Select desired appointment,” “Record appointment,” and “Confirm appointment.” Notice how these functional requirements expand upon the user's task to describe capabilities and functions that the system will need to include, allowing the user to complete the task.

As the analyst works with the business users of the system to discover user andfunctional requirements, the user may reveal processes that will be needed or information that will be needed. For example, as shown in Figure 2, the user may state “The system must retain customer order history for three years” (an information need). The analyst should probe for the reasoning behind this statement, such as “The system should allow registered customers to review their own order history for the past three years” (a process need). Similarly, the user may state “The system should check incoming customer orders for inventory availability” (a process need).

An alert analyst will recognize the related information need, “The system should maintain real-time inventory levels at all warehouses.” All of these requirements are necessary to fully understand the system that is being developed.



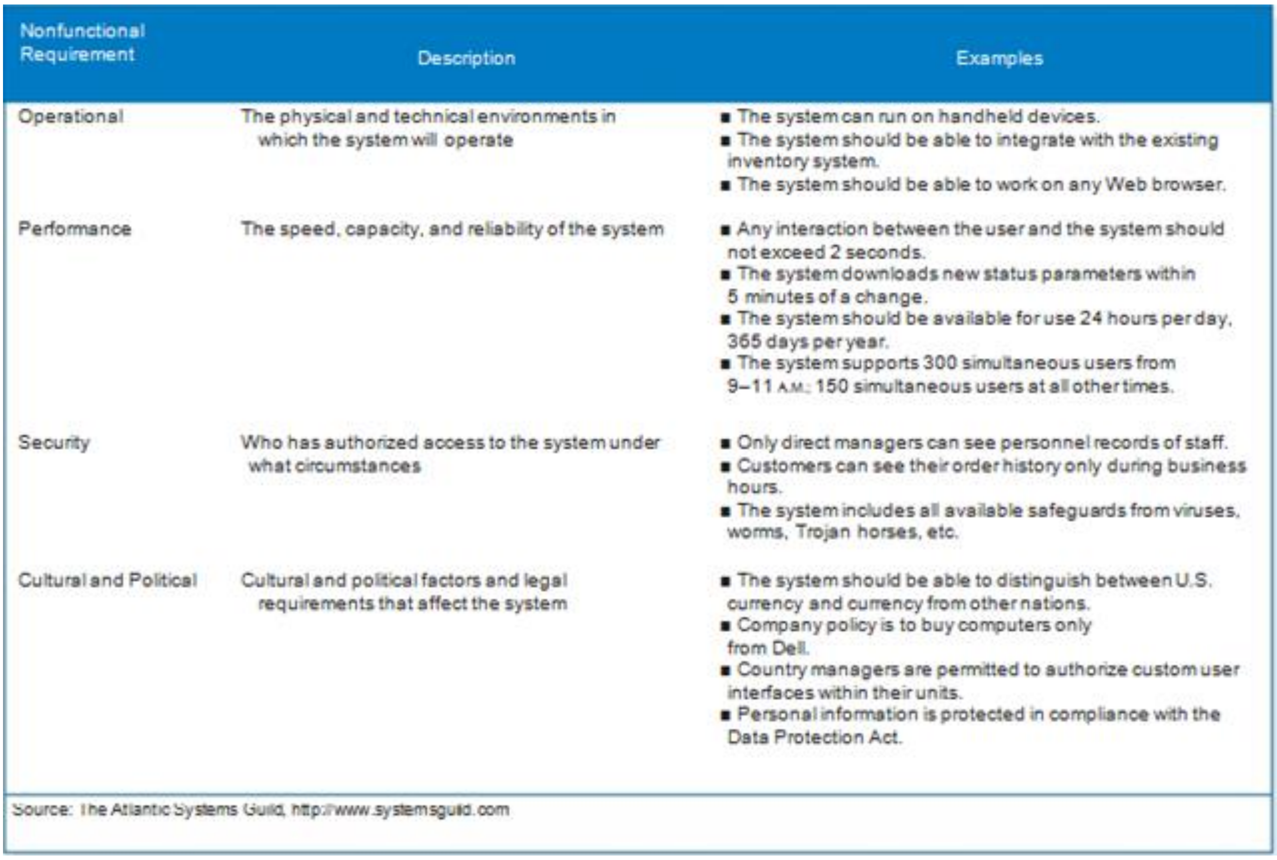
Process-oriented	A process the system must perform; a process the system must do	■ The system must allow registered customers to review their own order history for the past three years. ■ The system must check incoming customer orders for inventory availability. ■ The system should allow students to view a course schedule while registering for classes.
Information-oriented	Information the system must contain	■ The system must retain customer order history for three years. ■ The system must include real-time inventory levels at all warehouses. ■ The system must include budgeted and actual sales and expense amounts for current year and three previous years.

Figure 2: Functional Requirements

The final category of requirements is non-functional requirements. The IIBA defines this group of requirements as "the quality attributes, design, and implementation constraints, and external interfaces which a product must have." Although the term "nonfunctional" is not very descriptive, this requirement category includes important behavioral properties that the system must have, such as performance and usability. The ability to access the system through a mobile device would be considered a nonfunctional requirement. Nonfunctional requirements are primarily used in the design phase when decisions are made about the user inter face, the hardware and software, and the system's underlying architecture. Many of these requirements will be discovered during conversations with users in the analysis phase, however, and should be recorded as they are discovered.

Figure 2.1 lists different kinds of nonfunctional requirements and examples of each kind. Notice that the nonfunctional requirements describe a variety of system characteristics: operational, performance, security, and cultural and political. These characteristics do not describe business processes or information, but they are very important in understanding what the final system should be like. For example, the project team needs to know whether a system must be highly secure, requires subsecond response time, or has to reach a multilingual customer base. These requirements will affect design decisions that will be made in the design phase, particularly architecture design,

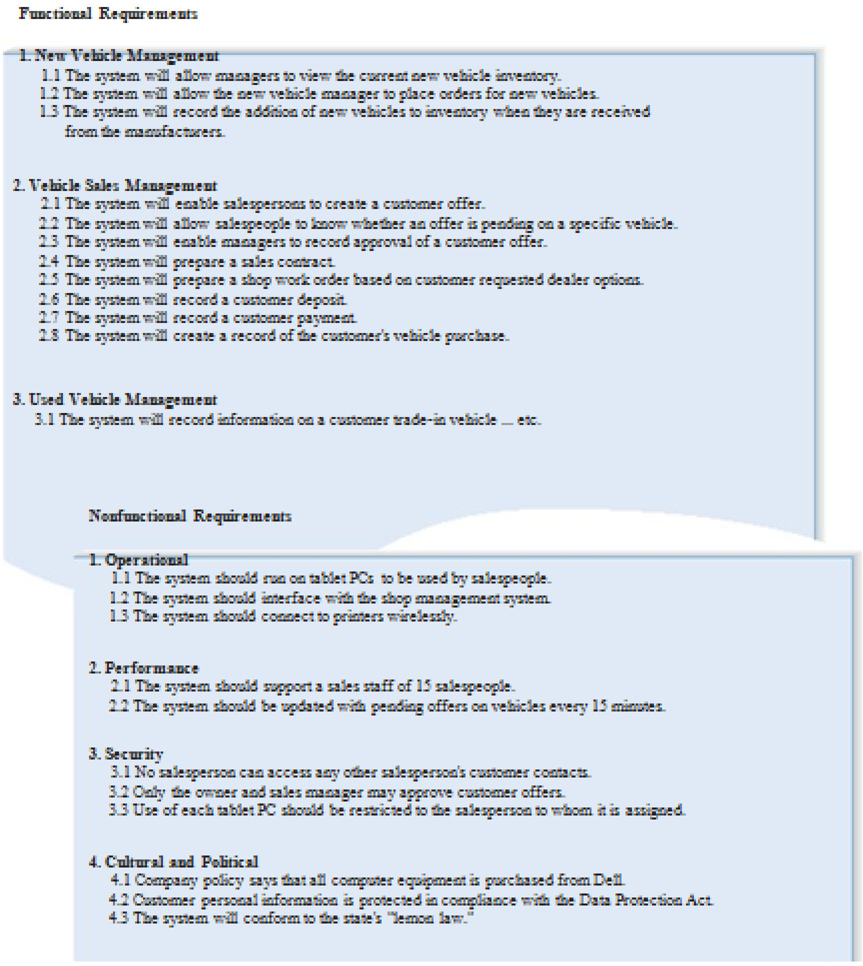
In addition, if the methodology in use includes developing test plans during analysis, then these requirements will be important in establishing testing benchmarks that will be needed later.



Nonfunctional Requirement	Description	Examples
Operational	The physical and technical environments in which the system will operate	■ The system can run on handheld devices. ■ The system should be able to integrate with the existing inventory system. ■ The system should be able to work on any Web browser.
Performance	The speed, capacity, and reliability of the system	■ Any interaction between the user and the system should not exceed 2 seconds. ■ The system downloads new status parameters within 5 minutes of a change. ■ The system should be available for use 24 hours per day, 365 days per year. ■ The system supports 300 simultaneous users from 9–11 A.M.; 150 simultaneous users at all other times.
Security	Who has authorized access to the system under what circumstances	■ Only direct managers can see personnel records of staff. ■ Customers can see their order history only during business hours. ■ The system includes all available safeguards from viruses, worms, Trojan horses, etc.
Cultural and Political	Cultural and political factors and legal requirements that affect the system	■ The system should be able to distinguish between U.S. currency and currency from other nations. ■ Company policy is to buy computers only from Dell. ■ Country managers are permitted to authorize custom user interfaces within their units. ■ Personal information is protected in compliance with the Data Protection Act.

Source: The Atlantic Systems Guild, http://www.systemsguild.com

Figure 2.1: Non Functional Requirements



Functional Requirements

1. New Vehicle Management

1.1 The system will allow managers to view the current new vehicle inventory.
1.2 The system will allow the new vehicle manager to place orders for new vehicles.
1.3 The system will record the addition of new vehicles to inventory when they are received from the manufacturers.

2. Vehicle Sales Management

2.1 The system will enable salespersons to create a customer offer.
2.2 The system will allow salespeople to know whether an offer is pending on a specific vehicle.
2.3 The system will enable managers to record approval of a customer offer.
2.4 The system will prepare a sales contract.
2.5 The system will prepare a shop work order based on customer requested dealer options.
2.6 The system will record a customer deposit.
2.7 The system will record a customer payment.
2.8 The system will create a record of the customer's vehicle purchase.

3. Used Vehicle Management

3.1 The system will record information on a customer trade-in vehicle ... etc.

Nonfunctional Requirements

1. Operational

1.1 The system should run on tablet PCs to be used by salespeople.
1.2 The system should interface with the shop management system.
1.3 The system should connect to printers wirelessly.

2. Performance

2.1 The system should support a sales staff of 15 salespeople.
2.2 The system should be updated with pending offers on vehicles every 15 minutes.

3. Security

3.1 No salesperson can access any other salesperson's customer contacts.
3.2 Only the owner and sales manager may approve customer offers.
3.3 Use of each tablet PC should be restricted to the salesperson to whom it is assigned.

4. Cultural and Political

4.1 Company policy says that all computer equipment is purchased from Dell.
4.2 Customer personal information is protected in compliance with the Data Protection Act.
4.3 The system will conform to the state's "lemon law."

Figure 2.2: Sample of Requirement Definition

2.2 Technique to Elicit Requirements from Different Users

An analyst is very much like a detective (and business users sometimes are like elusive suspects). He or she knows that there is a problem to be solved and therefore must look for clues that uncover the solution. Unfortunately, the clues are not always obvious (and often are missed), so the analyst needs to notice details, talk with witnesses, and follow leads, just as Sherlock Holmes would have done. The best analysts will thoroughly search for requirements using a variety of techniques and make sure that the current business processes and the needs for the new system are well understood before moving into design. You don't want to discover later that you have key requirements wrong—surprises like this late in the SDLC can cause all kinds of problems.

Before discussing the five requirements elicitation techniques in detail, a few practical tips are in order. First, the analyst should recognize that important side effects of the requirements definition process include building political support for the project and establishing trust and rapport between the project team and the ultimate users of the system. Every contact and interaction between the analyst and a potential business user or manager is an opportunity to generate interest, enthusiasm, and commitment to the project. Therefore, the analyst should be prepared to make good use of these opportunities as they arise during the requirements definition process.

Second, the analyst should carefully determine who is included in the requirements definition process. The choice to include (or exclude) someone is significant; involving someone in the process implies that the analyst views that person as an important resource and values his or her opinions. You must include all of the key stakeholders (the people who can affect the system or who will be affected by the system). This might include managers, employees, staff members and even some customers and suppliers. Also, be sensitive to the fact that some people may have significant influence within the organization even if they do not rank high in the formal organizational hierarchy. If you do not involve a key person, that individual may feel slighted, causing problems during implementation.

Finally, do everything possible to respect the time commitment that you are asking the participants to make. The best way to do this is to be fully prepared and to make good use of all the types of requirements elicitation techniques. Although, as we will see, interviewing is the most commonly used technique, other indirect methods may help the analyst develop a basic understanding of the business domain so that the direct techniques are more productive. In general, a useful strategy for the analyst to employ is to begin requirements gathering by interviewing senior managers to gain an understanding of the project and get the “big picture.” These preliminary interviews can then be followed by document analysis and, possibly, observation of business processes to learn more about the business domain, the vocabulary, and the as-is system. More interviews may then follow to collect the rest of the information needed to understand the as-is system.

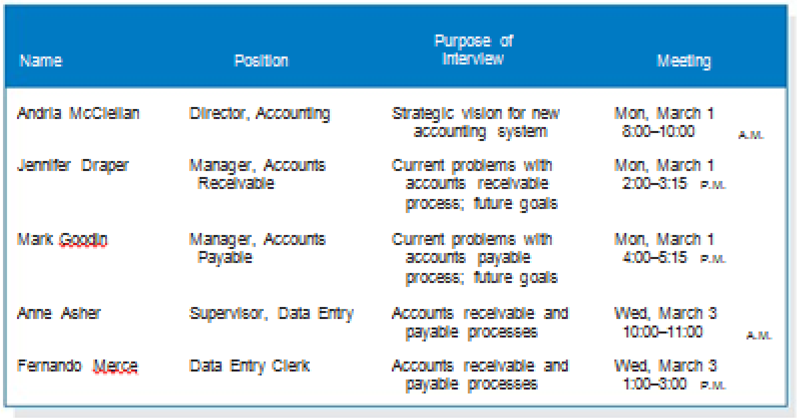
In our experience, identifying improvements is most commonly done through JAD sessions because these sessions enable the users and key stakeholders to work together and create a shared understanding of the possibilities for the to-be system. Occasionally, these JAD sessions are followed by questionnaires sent to a much larger group of users or potential users to get a broad range of opinions. The concept for the to-be system is frequently developed through interviews with senior managers, followed by JAD sessions with users of all levels, to make sure that the key requirements of the new system are well understood.

In this section, we focus on the five most commonly used requirements elicitation techniques: interviews, questionnaires, document analysis/ reading, and observation.

2.3 Interview

The interview is the most commonly used requirements elicitation technique. After all, it is natural—usually, if you need to know something, you ask someone. In general, interviews are conducted one on one (one interviewer and one interviewee), but sometimes, due to time constraints, several people are interviewed at the same time. There are five basic steps to the interview process: selecting interviewees, designing interview questions, preparing for the interview, conducting the interview, and post-interview follow-up.

Selecting Interviewees- An interview schedule should be created, listing who will be interviewed, the purpose of the interview, and where and when it will take place. (See Figure 2.3.) The schedule can be an informal list that is used to help set up meeting times or a formal list that is incorporated into the work plan. The people who appear on the interview schedule are selected on the basis of the analyst’s information needs. The project sponsor, key business users, and other members of the project team can help the analyst determine who in the organization can best provide important information about requirements. These people are listed on the interview schedule in the order in which they should be interviewed.



Name	Position	Purpose of Interview	Meeting
Andria McClellan	Director, Accounting	Strategic vision for new accounting system	Mon, March 1 8:00–10:00 A.M.
Jennifer Draper	Manager, Accounts Receivable	Current problems with accounts receivable process; future goals	Mon, March 1 2:00–3:15 P.M.
Mark Goodin	Manager, Accounts Payable	Current problems with accounts payable process; future goals	Mon, March 1 4:00–5:15 P.M.
Anne Asher	Supervisor, Data Entry	Accounts receivable and payable processes	Wed, March 3 10:00–11:00 A.M.
Fernando Merce	Data Entry Clerk	Accounts receivable and payable processes	Wed, March 3 1:00–3:00 P.M.

Figure 2.3: Sample interview schedule

People at different levels of the organization will have different viewpoints on the system, so it is important to include both managers who manage the processes and staff who actually perform the processes to gain both high-level and low-level perspectives on an issue. Also, the kinds of interview subjects that you need may change over time. For example, at the start of the project the analyst has a limited understanding of the as-is business process. It is common to begin by interviewing one or two senior managers to get a strategic view and then move to mid-level managers who can provide broad, overarching information about the business process and the expected role of the system being developed. Once the analyst has a good understanding of the big picture, lower-level managers and staff members can fill in the exact details of how the process works. Like most other things about systems analysis, this is an iterative process—starting with senior managers, moving to mid-level managers, then staff members, back to mid-level managers, and so on, depending upon what information is needed along the way.

It is quite common for the list of interviewees to grow, often by 50%–75%. As you interview people, you likely will identify more information that is needed and additional people who can provide the information.

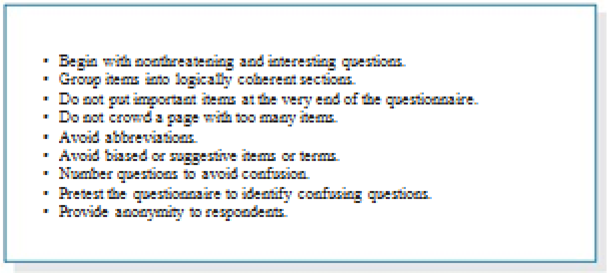
2.4 Questionnaire

A questionnaire is a set of written questions for obtaining information from individuals. Questionnaires often are used when there is a large number of people from whom information and opinions are needed. In our experience, questionnaires are commonly used for systems intended for use outside of the organization (e.g., by customers or vendors) or for systems with business users spread across many geo- graphic locations. Most people automatically think of paper when they think of questionnaires, but today more questionnaires are being distributed in electronic form, either via e-mail or on the Web. Electronic distribution can save a significant amount of money, compared with distributing paper questionnaires.

Developing good questions is critical for questionnaires because the information on a questionnaire cannot be immediately clarified for a confused respondent. Questions on questionnaires must be very clearly written and must leave little room for misunderstanding; therefore, closed-ended questions tend to be most commonly used. Questions must enable the analyst to clearly separate facts from opinions. Opinion questions often ask the respondent the extent to which they agree or disagree (e.g., "Are network problems common?"), while factual questions seek more precise values (e.g., "How often does a network problem occur: once an hour, once a day, or once a week?"). See Figure 2.4 for guidelines on questionnaire design.

Perhaps the most obvious issue—but one that is sometimes overlooked—is to have a clear understanding of how the information collected from the questionnaire will be analyzed and used. You must address this issue before you distribute the questionnaire, because it is too late afterward.

Questions should be relatively consistent in style so that the respondent does not have to read instructions for each question before answering it. It is generally a good practice to group related questions together to make them simpler to answer. Some experts suggest that questionnaires should start with questions important to respondents, so that the questionnaire immediately grabs their inter- est and induces them to answer it. Perhaps the most important step is to have several colleagues review the questionnaire and then pretest it with a few people drawn from the groups to whom it will be sent. It is surprising how often seemingly simple questions can be misunderstood.



- Begin with nonthreatening and interesting questions.
- Group items into logically coherent sections.
- Do not put important items at the very end of the questionnaire.
- Do not crowd a page with too many items.
- Avoid abbreviations.
- Avoid biased or suggestive items or terms.
- Number questions to avoid confusion.
- Pretest the questionnaire to identify confusing questions.
- Provide anonymity to respondents.

Figure 2.4: Good Question Design

2.5 Observation

Observation, the act of watching processes being performed, is a powerful tool to gain insight into the as-is system. Observation enables the analyst to see the reality of a situation, rather than listening to others describe it in interviews or JAD sessions. Several research studies have shown that many managers really do not remember how they work and how they allocate their time. (Quick, how many hours did you spend last week on each of your courses?) Observation is a good way to check the validity of information gathered from other sources such as interviews and questionnaires.

In many ways, the analyst becomes an anthropologist as he or she walks through the organization and observes the business system as it functions. The goal is to keep a low profile, to not interrupt those working, and to not influence those being observed. Nonetheless, it is important to understand that what analysts observe may not be the normal day-to-day routine because people tend to be extremely careful in their behavior when they are being watched.10 Even though normal practice may be to break formal organizational rules, the observer is unlikely to see this. (Remember how you drove the last time a police car followed you?) Thus, what you see may not be what you really want.

Observation is often used to supplement interview information. The location of a person's office and its furnishings gives clues as to their power and influence in the organization, and such clues can be used to support or refute information given in an interview. For example, an analyst might become skeptical of someone who claims to use the existing computer system extensively if the computer is never turned on while the analyst visits. In most cases, observation will support the information that users provide in interviews. When it does not, it is an important signal that extra care must be taken in analyzing the business system.

OBSERVATION PRACTICE

Visit the library at your college or university and observe how the book check-out process occurs. First, watch several students checking books out, and then check one out yourself. Prepare a brief summary report of your observations. When you return to class, share your observations with others. You may notice that not all the reports present the same information. Why? How would the information be different had you used the interview or JAD technique?

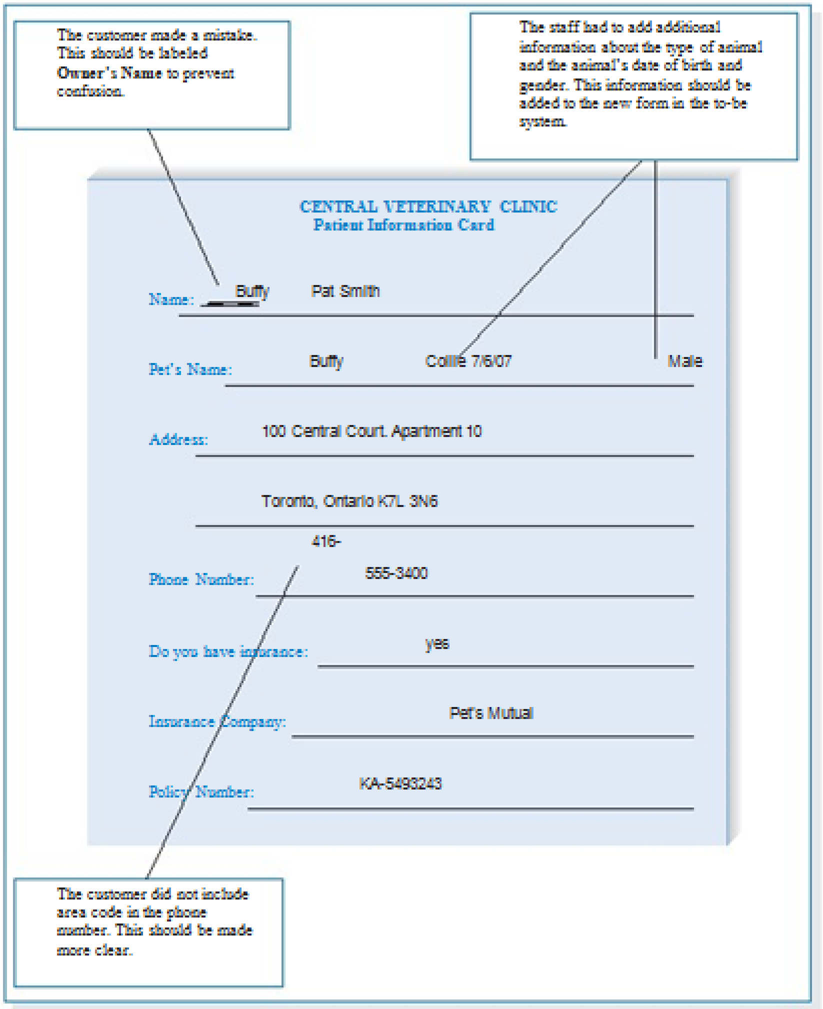
2.6 Document Reading/ analysis

Project teams often use document analysis to understand the as-is system. Under ideal circumstances, the project team that developed the existing system will have produced documentation, which was then updated by all subsequent projects. In this case, the project team can start by reviewing the documentation and examining the system itself.

Unfortunately, most systems are not well documented, because project teams fail to document their projects along the way, and when the projects are over, there is no time to go back and document. Therefore, there may not be much technical documentation about the current system available, or it may not contain updated information about recent system changes. However, there are many helpful documents that do exist in the organization: paper reports, memorandums, policy manuals, user training manuals, organization charts, and forms. Problem reports filed by the system users can be another rich source of information about issues with the existing system.

But these documents (forms, reports, policy manuals, organization charts) only tell part of the story. They represent the formal system that the organization uses. Quite often, the "real," or informal system differs from the formal one, and these differences, particularly large ones, give strong indications of what needs to be changed. For example, forms or reports that are never used likely should be eliminated. Likewise, boxes or questions on forms that are never filled in (or are used for other purposes) should be rethought. See Figure 2.5 for an example of how a document can be interpreted.

The most powerful indication that the system needs to be changed is when users create their own forms or add additional information to existing ones. Such changes clearly demonstrate the need for improvements to existing systems. Thus, it is useful to review both blank and completed forms to identify these deviations. Likewise, when users access multiple reports to satisfy their information needs, it is a clear sign that new information or new information formats are needed.



The customer made a mistake. This should be labeled Owner's Name to prevent confusion.

The staff had to add additional information about the type of animal and the animal's date of birth and gender. This information should be added to the new form in the to-be system.

CENTRAL VETERINARY CLINIC
Patient Information Card

Name: Buffy Pat Smith

Pet's Name: Buffy Collie 7/6/07 Male

Address: 100 Central Court. Apartment 10

Toronto, Ontario K7L 3N6

Phone Number: 416- 555-3400

Do you have insurance: yes

Insurance Company: Pet's Mutual

Policy Number: KA-5493243

The customer did not include area code in the phone number. This should be made more clear.

Figure 2.5: Performing Document Analysis

2.7 Requirement Specification and Validation

Technically feasible to confirm that the requirements specification document is of the desired

Quality. However, it not easy to ascertain whether the requirements expressed in the specification are really what the client wants and needs. The client may not know what he or she wants, what is

wanted may be completely different from what is needed. Requirements validation is the process of

checking that the requirements as specified are a true representation of the client’s needs and wishes.

And it was an essential from the earliest stages of requirements engineering.

2.8 Example model required system aid in the analysis of requirement

Requirement analysis is the most important part of job. It will help you determining the actual needs of stakeholders. At the same time, enable you to communicate with the stakeholders in a language they understand (like charts, models, flow-charts,) instead of complex text.

- Business process modeling notation (BPMN)

  BPMN (Business Process Modeling & Notation) is a graphical representation of your business process using simple objects, which helps the organization to communicate in a standard manner. Various objects used in BPMN includes:

  - Flow objects
  - Connecting objects
  - Swim lanes
  - Artifacts.

- UML (Unified Modeling Language)

  UML is a modelling standard primarily used for specification, development, visualization and documenting of software system. To capture important business process and artifacts UML provides objects like State, Object, Activity, Class diagram. There are 14 UML diagrams that help with modelling like the use case diagram, interaction diagram, class diagram, component diagram, sequence diagram, etc. UML models are important in the IT segment as it becomes the medium of communication between all stakeholders. A UML-based business model can be a direct input to a requirements tool. A UML diagram can be of two type's Behavioral model and Structural model. A behavioral model tries to give information about what the system do while a structural model will give what is the system consist of.

- Flow chart technique

A flowchart is a visual representation of the sequential flow and control logic of a set of related activities or actions. There are different formats for flowcharts which include Linear, Top-down and cross-functional (swim lanes). A flow chart can be used for different activities like representing data flows, system interactions, etc. The advantage of using Flowchart is that it can be easy to read and write even for non-technical team members, and can show the parallel process by function, critical attributes of a process.

- Data flow diagram

  Data flow diagrams show how data is processed by a system in terms of inputs and outputs. Components of data flow diagram includes Process, Flow, Store and Terminator
- A logical data flow diagram shows system's activities while a physical data flow diagram shows a system's infrastructure. A data flow diagram can be designed early in the requirement elicitation process of the analysis phase within the SDLC (System Development Life Cycle) to define the project scope. For easy analyzing a data flow diagram can be drilled down into its sub-processes known as "levelled DFD".