

User Requirements and Engineering Specifications

Good user requirements are one of the key factors that lead to a successful design. User requirements capture the stakeholders' needs, desires, and expectations for a product and are the basis for developing engineering specifications--the statements upon which a design will be verified against. Engineering specifications serve as a collection of criteria that the design must meet in order to fulfill the user requirements that were elicited from the stakeholders.

After completing this block you will be able to:

- elicit and develop user requirements from stakeholders
- identify data collection strategies to inform user requirements
- prioritize user requirements
- translate user requirements to engineering specifications
- write specific and unambiguous user requirements and engineering specifications

Table of Contents

[Section 1: What are User Requirements?](#)

[Section 2: What are Engineering Specifications?](#)

[Section 3: User Requirements Elicitation](#)

[Section 4 : Translating User Requirements to Engineering Specifications](#)

[Section 6: Writing Quality User Requirements and Engineering Specifications](#)

[Section 7: Organizing and Displaying your User Requirements and Specifications](#)

[Section 8: Common Pitfalls When Developing User Requirements](#)

[References](#)

Core Content

Section 1: What are User Requirements?

[**User requirements**, or] product requirements are any function, constraint, or other property required for a designed artifact to meet the needs or wants of stakeholders; the requirements are translated into engineering specifications that are both quantifiable and measurable in order to guide engineering design processes. [Mohedas]

Literature defines product or user requirements in multiple ways:

- Identifiable capabilities expressed as performance measurables of functions that the system must possess to meet the mission objectives
- Attributes of the final design that must be a part of any acceptable solution to the design problem
- An externally observable characteristic of a desired system

The term “user requirements” may also be referred to as: product requirements, design requirements, or customer requirements. [Mohedas, Dieter, Jiao]

Simply put, a user requirement is a statement that specifies WHAT a product should do, but it does not define HOW it should do it.

For example, the following user requirement specifies WHAT a product should do:

The device shall decrease the temperature of the skin

but not HOW it should do it:

The device shall apply moisture to cool down the skin.

In summary, requirements must be specific, clear, and without ambiguity.

Why are user requirements important?

Case studies have demonstrated that the success of new products depends upon how well the front-end design phases are executed. Studies have also shown that in many instances product failures are a result of critical decision errors made during the front-end design phases that could not be cost-effectively rectified later in the design process. A key component of front-end design involves eliciting and developing product requirements. [Mohedas]

The price is high for teams that fail to define requirements or that do it poorly. Ill-defined requirements result in requirements defects, and the consequences of these defects are ugly:

- Expensive rework and cost overruns.
- A poor quality product.
- Late delivery.
- Dissatisfied customers.
- Exhausted and demoralized team members.

To reduce the risk of ...project failure and the costs associated with defective requirements, project teams must address requirements early in [the design process] and they must define requirements properly.

[Gottesdiener]

Not only do poor forethought and planning in developing user requirements lead to the consequences described above, the importance of developing quality user requirements is even more important when we consider intercultural design. The following excerpt, from an article in the *Journal of Human-Computer Interaction*, describes how the perception of what a product should be can be highly influenced by the receiving culture:

“...Cultural diversity makes it impossible for designers to depend on instinctive knowledge or personal experience, therefore, many researchers have identified the need to explore cultural issues in web interface design. For example, Marcus & Gould [15] pointed out that web designers need to do much planning, research, analysis, design, evaluation, documentation, and training to deeply comprehend the requirements of the user, market, and business. Indeed, people from different cultures use web interfaces in different ways, hold different mental models for visual representations, navigation, interaction, and layouts, and have different communication patterns and expectations. In the context of globalisation, web localisation becomes a powerful strategy to acquire an audience in a global market. Therefore, web developers and designers have to make adaptations to fit the needs of people from different cultures...” **[Hsieh]**

“...A major finding from the existing literature on culture and [human-computer interaction (HCI)] is that there are cultural differences in the models that different user groups have of what HCI is. For example, Chinese users adapt a more holistic model of what it is to use software, compared to European users (Smith et al., 2004). The cultural differences imply the need for localization of the software design (Marcus & Gould, 2000) and for localization and cultural adaptation of established user experience and usability evaluation methods (Clemmensen, Hertzum, Hornbaek, Shi, & Yammiyavar, 2009; Hall, De Jong, & Steehouder, 2004; Smith & Yetim, 2004; Winschiers-Theophilus, 2009)...” **[Clemmensen]**

[Back to Table of Contents](#)

Section 2: What are Engineering Specifications?

Engineering specifications are the set of goals that, when achieved, necessarily meets the user requirements. Specifications are the restatement of the design problem in terms of parameters that can be measured and have target values.

The term “engineering specifications” may also be referred to as “product specifications” or “functional requirements”. **[Jiao]**

Engineering specifications are derived from the user requirements. Engineering specifications must be **[Makower]**:

1. user-driven
2. quantifiable
3. solution-neutral
4. specific

These attributes of engineering specifications are further described below:

1. User-Driven

Engineering specifications are developed based on the user requirements the team derives from stakeholders.

Establishing the engineering characteristics is a critical step toward writing the product design specification... The process of identifying the needs that a product must fill is a complicated undertaking... A major challenge of this step is to hear and record the fullness of customer ideas without applying assumptions. For example, if a customer is talking about carry-on luggage they may say, "I want it to be easy to carry." An engineer might interpret that phrase to mean, "make it lightweight," and set weight as a design parameter that should be minimized. However, the customer may really want a carry-on case that is easy to fit into the overhead luggage compartment of a plane. The carrying task is already easy due to the design innovation of wheeled luggage.

[Dieter]

The stakeholder's desire for luggage to be "easy to carry" may lead to several considerations in terms of user requirements and specifications. Does the luggage need to be lightweight? Does it need to be compact? What dimensions and weight (and other characteristics) fulfill these particular requirements? Should it have wheels or a carrying strap? There are many possibilities and these should be fully explored by the design team while engaging with stakeholders.

The above example is just one of many situations in which the design team will need to thoroughly explore the stakeholder's desires to create engineering specifications that *accurately* reflect the *stakeholder's* needs, aka are *user-driven*. This concept of being "user-driven" is not limited to the development of engineering specifications alone, rather, the bulk of the design process should be focused on the user as well.

2. Quantifiable

Engineering specifications should be quantifiable, or in other words, able to be measured in terms of engineering units. The following provides gives a good overview of how *quantifiable* engineering specifications can be defined:

- a. **Design Parameters.** Parameters are a set of physical properties whose values determine the form and behavior of a design. Parameters include the features of a design that can be set by designers and the values used to describe the performance of a design. Note: it must be clear that designers make choices in an attempt to *achieve* a particular product performance level, but they cannot *guarantee* they will succeed until embodiment design activities are finalized.
- b. **Design Variable.** A design variable is a parameter over which the design team has a choice. For example, the gear ratio for the RPM reduction from the rotating spindle of an electric motor can be a variable.
- c. **Constraints.** Constraints are limits on design freedom. They can take the form of a selection from a particular color scheme, or the use of a standard fastener, or a specific size limit determined by factors beyond the control of both the design team and the customers.10 Constraints may be limits on the maximum or minimum value of a design

variable or a performance parameter. Constraints can take the form of a range of values.
[Dieter]

In simple terms, engineering specifications that are quantifiable often have the following characteristics:

- Numeric
- Unit [i.e. inches, centimeters, kg, rpm...]
- Relational operator (=, <, ≥)
- Testable **[Makower]**

3. Specificity

Engineering specifications that show specificity accurately reflect the stakeholder's requirements and are able to be understood by outside designers who may not be familiar with your project:

Examples of nonspecific specifications:

- product weight ≥ weight of standard refrigerator
- product is aesthetically pleasing

For someone who may not be familiar with your project, they might have several questions in mind - what is the weight of a standard refrigerator? How do you determine what is "standard" in terms of refrigerator sizes? What does it mean to be "aesthetically pleasing"? Is there a certain standard one needs to meet to be deemed aesthetically pleasing, and what is that standard?

Examples of specific specifications:

- product weight ≥ 500 lbs.
- product is ranked a 4 out of 5 on a 5-point Likert scale by 75% of stakeholders.

While specifying their engineering specifications, the designers should always seek to create standards based on clear rationale. This rationale is achieved through a critical analysis of ethnographic data, engaging with stakeholders, and consultation of existing literature.

Specificity is especially important when trying to define vague requirements such as "safe" or "effective"

4. Solution Neutral

Engineering specifications should be "solution-neutral" meaning that:

"...the specification *at this time* should not be so complete as to suggest a single concept or class of concepts." **[Dieter]**

In other words, solution-neutral engineering specifications describe WHAT the design should achieve in order to fulfill the user requirements, but they do not dictate HOW the design should achieve these goals or WHAT the design should be.

Example: In response to the user's desire for their product to be "easy to carry" a student engineer came up with the following engineering specifications:

- product weighs \leq 10 lbs. and is made of lightweight aluminum alloy
- the product is able to be stowed in an airplane luggage compartment

Unless the user specifically requested that said product be constructed from aluminum alloy, the second statement is not a good engineering requirement because it dictates *how* the design should achieve the quality of being less than 10 pounds and "easy to carry"

[Back to Table of Contents](#)

Section 3: User Requirements Elicitation

To develop quality requirements, design experts have advocated the collection of information about end-users, stakeholders, and product-use environments from a variety of sources and using a variety of methods, such as interviews with end-users and other stakeholders, focus groups, surveys, customer complaints, sales data, and codes and standards. Newer information gathering methods based on the philosophies of human-centered and participatory design include focus group brainstorming techniques, consensus-building workshops, the use of prototypes during elicitation, protocol analysis, and comprehensive design ethnographies. These methods allow one to gain a better understanding of a product's stakeholders and its context of use in order to properly define product requirements. **[Mohedas]**

It is the customer's desires that ordinarily drive the development of the product, not the engineer's vision of what the customer should want. Information on the customer's needs is obtained through a variety of channels:

- **Interviews with customers:** Active marketing and sales forces should be continuously meeting with current and potential customers. Some corporations have account teams whose responsibility is to visit key customer accounts to probe for problem areas and to cultivate and maintain friendly contact. They report information on current product strengths and weaknesses that will be helpful in product upgrades. An even better approach is for the design team to interview single customers in the service environment where the product will be used. Key questions to ask are: What do you like or dislike about this product? What factors do you consider when purchasing this product? What improvements would you make to this product?
- **Focus groups:** A focus group is a moderated discussion with 6 to 12 customers or targeted customers of a product. The moderator is a facilitator who uses prepared questions to guide the discussion about the merits and disadvantages of the product. Often the focus group occurs in a room with a one-way window that provides for videotaping of the discussion. In both the interviews and the focus groups it is important to record the customer's response in his or her own words. All interpretation is withheld until the analysis of results. A trained moderator will follow up on any surprise answers in an attempt to uncover implicit needs and latent needs of which the customer is not consciously aware.
- **Customer complaints:** A sure way to learn about needs for product improvement is from customer complaints. These may be recorded by communications (by telephone, letter, or email) to a customer information department, service center or warranty department, or a return center at a

larger retail outlet. Third party Internet websites can be another source of customer input on customer satisfaction with a product. Purchase sites often include customer rating information. Savvy marketing departments monitor these sites for information on their products and competing products.

- **Warranty data:** Product service centers and warranty departments are a rich and important source of data on the quality of an existing product. Statistics on warranty claims can pinpoint design defects. However, gross return numbers can be misleading. Some merchandise is returned with no apparent defect. This reflects customer dissatisfaction with paying for things, not with the product.
- **Customer surveys:** A written questionnaire is best used for gaining opinions about the redesign of existing products or new products that are well understood by the public. (Innovative new products are better explored with interviews or focus groups.) Other common reasons for conducting a survey are to identify or prioritize problems and to assess whether an implemented solution to a problem was successful. A survey can be done by mail, e-mail, telephone, or in person.

Competitive performance benchmarking involves testing a company's product against the best-in-class that can be found in the current marketplace. It is an important step for making comparisons in the design and manufacturing of products. Benchmarking is used to develop performance data needed to set functional expectations for new products and to classify competition in the marketplace. Competitive performance benchmarking compares the performance of a company's product to the market's leading products. Benchmarking is a logical starting point in determining engineering characteristics for a product.

In addition to the above information sources, additional information can be sought out from academic literature, patent searches, and more. It is the designer's job to synthesize data into proper user requirements. Below is a table outlining a variety of information sources for designers **[Dieter]**:

TABLE 4.1
Sources of information for engineering design

I. Public sources	
A.	Federal departments and agencies (Defense, Commerce, Energy, NASA, etc.)
B.	State and local government (highway department, departments dealing with land use, consumer safety, building codes, etc.)
C.	Libraries—community, university, special
D.	Universities, research institutions, museums
E.	Foreign governments—embassies, commercial attaches
F.	Internet—Much information is free. Some requires fees.
II. Private sources	
A. Nonprofit organizations and services	
1.	Professional societies
2.	Trade and labor associations
3.	Membership organizations (motorists, consumers, veterans, etc.)
B. Profit-oriented organizations	
1.	Vendors (include manufacturers, suppliers, financiers). Catalogs, samples, test data, cost data and information on operation, maintenance, servicing and delivery
2.	Other business contacts with manufacturers and competitors
3.	Consultants
C. Individuals	
1.	Direct conversation or correspondence
2.	Personal friends, associates, "friends of friends"
3.	Faculty

What common design needs do user requirements need to address?

Often times, stakeholders may not completely express all of their desired requirements during the “elicitation” phase of developing user requirements.

To ensure **completeness** while developing user requirements, it is helpful to think about user requirements in terms of the framework described below:

Garvin identified the eight basic dimensions of quality for a manufactured product. These have become a standard list that design teams use as a guide for completeness of customer requirement data gathered in the PDP [product development process].

- **Performance:** The primary operating characteristics of a product. This dimension of quality can be expressed in measurable quantities, and therefore can be ranked objectively.
- **Features:** Those characteristics that supplement a product’s basic functions. Features are frequently used to customize or personalize a product to the customer’s taste.
- **Reliability:** The probability of a product failing or malfunctioning within a specified time period.
- **Durability:** A measure of the amount of use one gets from a product before it breaks down and replacement is preferable to continued repair. Durability is a measure of product life. Durability and reliability are closely related.
- **Serviceability:** Ease and time to repair after breakdown. Other issues are courtesy and competence of repair personnel and cost and ease of repair.
- **Conformance:** the degree to which a product’s design and operating characteristics meet both customer expectations and established standards. These standards include industry standards and safety and environmental standards. The dimensions of performance, features, and conformance are interrelated. When competing products have essentially the same performance and many of the same features, customers will tend to expect that all producers of the product will have the same quality dimensions. In other words, customer expectations set the baseline for the product’s conformance.
- **Aesthetics:** How a product looks, feels, sounds, tastes, and smells. The customer response in this dimension is a matter of personal judgement and individual preference. This area of design is chiefly the domain of the *industrial designer*, who is more an artist than an engineer. An important technical issue that affects aesthetics is *ergonomics*, how well the design fits the human user.
- **Perceived quality:** This dimension generally is associated with reputation. Advertising helps to develop this dimension of quality, but it is basically the quality of similar products previously produced by the manufacturer that influences reputation.

The challenge for the design team is to combine all the information gathered about customers’ needs for a product and interpret it. The customer data must be filtered into a manageable set of requirements that drive the generation of design concepts. The design team must clearly identify preference levels among the customer requirements before adding in considerations like time to market or the requirements of the company’s internal customers. **[Dieter]**

In addition to the list described above, to ensure completeness of user requirements, keep in mind that a single design need can also yield multiple requirements [Makower]:

Design Need	Specify as Requirement(s)
Good image quality	<ul style="list-style-type: none">• Brightness• Granularity
Easy to transport	<ul style="list-style-type: none">• Weight• Overall dimensions
Device sets up quickly	<ul style="list-style-type: none">• Time required to set-up• Number of steps required to set-up

[Back to Table of Contents](#)

Section 4 : Translating User Requirements to Engineering Specifications

Engaging stakeholders with little or no engineering or product design background can be challenging in settings with limited methodical engineering design tradition and experience.

Once the need for a new product is established, efficient and easy-to-administer methods that directly and systematically engage stakeholders to elicit user requirements, which define their need, are required. Hence, qualitative user requirements (URs) and their translation to quantitative engineering specifications are the major building blocks in an upstream product design process.

When designed appropriately, the elicitation process of URs and their subsequent mapping to engineering specifications should ensure customer satisfaction and willingness to choose, adopt, purchase, or use the final product. To achieve these objectives, there are three key strategies to preventing a mismatch between customer needs (requirements) and product specification:

1. identifying the “right” types of needs
2. eliciting “real” URs, which may involve qualitative information, and
3. translating the requirements into “effective” quantitative engineering specifications [5].

Utilizing qualitative and quantitative approaches, engineers address these three key strategies by reducing the ambiguity in user inputs and clarifying the obtained URs through careful communication with stakeholders to achieve completeness and consistency of URs.

Ethnography, free association, open-ended responses, and clustering techniques are some of the **qualitative methods** used to elicit implicit and explicit URs.

Ethnography, informed by research in anthropology, investigates tacit knowledge about the design subject [8,9]. To utilize ethnography for engineering design purposes, a multi-functional design team observes the actual behavior and environment of the potential end-users and records their interactions with their

environment. This method is suitable to identify end-users' problems and needs, especially for designers who are new to an environment.

In **free association**, elicitation stimulus probes or cues about requirements are presented to the end-users, who are asked to verbalize the concepts that immediately come to mind [10]. This approach is suitable for exploratory purposes and to capture open-ended inquiries.

Open-ended responses ask questions to elicit feedback about users' preferences as informed by their background and professional role. This approach is suitable when the design team is new to an environment or has limited background of a design task and is interested in capturing general information.

Clustering methods identify how stakeholders perceive and represent URs, such as which ones are viewed as similar and which are dissimilar. This approach is suitable for making comparisons across different stakeholder groups as well as individual users.

While these methods have their merits and are relatively simple to administer, their outcomes are often subjective, colloquial, context and linguistic dependent, and difficult to map to quantitative engineering specifications...

There are three keys to preventing a mismatch between customer needs and product attributes: identifying the "right" types of needs, eliciting "real" user requirements, and translating the requirements into "effective" engineering specifications (i.e., product attributes). Design engineering has developed several tools to map the acquired user requirements to quantified engineering specifications or attributes. For example, **quality function deployment (QFD)** is a tool that was developed in the 1970s to convert potential end-user and customer requirements to engineering attributes.

A review of the literature also reveals further challenges with inadequate methodologies for capturing complex requirements, the lack of expert guidance in eliciting and analyzing the requirements, and the application of quantitative evaluation for qualitative items. These problems compound when the captured requirements are difficult to define and intangible. For instance, while the customer may be able to explain and quantify some of the essential requirements such as "low-cost", they may run into difficulty explaining other requirements such as "aesthetically beautiful" or "user friendly". The difficulties stem from a term's subjective nature, based as it is upon each customer's diverse perspectives (knowledge, responsibilities, gender, experience, culture, etc.). Following is a description of a few subjective requirements and the methods used to establish the associated engineering attributes.

For instance, to understand *environmental friendliness* it needs to be broken down to sub-requirements. Reid *et al.* evaluate users' environmental friendliness perception in the auto industry based on vehicles' two-dimensional appearances. Ersal *et al.*, quantify perception of *craftsmanship* in vehicle interior design using a functional dependence table and statistical analysis methods such as cluster analysis of craftsmanship's perceptions and multidimensional scaling. In another study, evaluating *closeness* to customers is quantified and tested using multi-item scales. Witel *et al.* use qualitative methodologies to evaluate the performance of an e-service and its *attractiveness* using taxonomy methods.

Some designers use the **Kano model** to understand and analyze user needs and their impacts on user satisfaction. This model considers both the asymmetric and non-linear relationships between product performance and user satisfaction... [\[the Kano model is described in Section 5: Prioritizing User Requirements\]](#)

However, the Kano model does not provide a systematic and methodical quantification approach to translate user needs into measurable engineering parameters. Hence, recent attempts to assess and estimate engineering parameters based on the outcomes of the Kano model have led to the development of an “analytical Kano” model, which is combined with QFD in some cases. This analytical Kano creates a series of criteria to classify user requirements and a configuration index that provides a decision factor for selecting the functional requirements that contribute to product attributes. Even though the analytical Kano model attempts to quantify the elicited requirements, the designer’s subjective evaluation can still affect the quantification process.

Although requirements can vary depending on the type, experience level, knowledge, and interests of stakeholders, and the user context, purely quantitative methods of eliciting URs may fail to thoroughly engage stakeholders in order to resolve conflicting input, reveal nuanced differences among stakeholder input, and inadvertently promote limited iterations with stakeholders to establish accurate translations of requirements to engineering specifications. **[Sarvestani]**

Therefore, qualitative methods should be used in conjunction with quantitative methods to promote accuracy and thoroughness of UR’s. In addition, many of the same techniques used to elicit user requirements (interviews, literature reviews, benchmarking, etc.) may also be used in identifying engineering specifications.

For example, “safety” is often a user requirement cited for various applications, but it is a very broad term which can have multiple interpretations, sub-requirements, and specifications. The designer is likely to encounter conflicting viewpoints as to what constitutes a “safe” product from their various stakeholders. A review of relevant literature may present yet another view as to what constitutes a “safe” product for a certain application. Technical standards, such as the ISO ([International Organization for Standardization](#)), are often a useful reference to determine how “safety”, among other requirements, can be defined objectively in various applications. It is critical to thoroughly utilize all avenues of data, whether that be qualitative or quantitative, in order to have enough information to make accurate judgments on what engineering specifications will fulfill the respective user requirements.

ADDITIONAL RESOURCE: [Chapter 6 from Ullman’s *The Mechanical Design Process \(4th edition\)*](#) covers the importance of developing engineering specifications along with a clear explanation of **Quality Function Deployment (QFD)**--one of the most common ways used to generate engineering specifications.

[Back to Table of Contents](#)

Section 5: Prioritizing User Requirements

So far, we have covered the basic properties of user requirements, how to obtain them, and how to create user requirements that are clear and unambiguous. However, the list of user requirements isn’t complete until they are organized and prioritized. It is important to prioritize your user requirements such that resources needed for development can be planned, distributed, and used appropriately. **[Maguire]**

In addition, prioritizing user requirements is essential to the success of the design because user satisfaction is impacted by how well the design team addresses the user requirements that the stakeholders deem to be basic and important. If the team puts too much effort and resources into addressing a feature of the design

that is “nice to have” versus a feature of the design that a user deems “essential”, the satisfaction of the product can be negatively affected. The following excerpts give an in-depth analysis of prioritization:

Not all customer requirements are equal. This essentially means that customer requirements ...have different values for different people. The design team must identify those requirements that are most important to the success of the product in its target market and must ensure that those requirements and the needs they meet for the customers are satisfied by the product.

This is a difficult distinction for some design team members to make because the pure engineering viewpoint is to deliver the best possible performance in all product aspects. A Kano diagram is a good tool to visually partition customer requirements into categories that will allow for their prioritization. **[Dieter]**

[The Kano model] considers both the asymmetric and non-linear relationships between product performance and user satisfaction...Generally, the Kano model classifies product attributes into five categories:

1. Must-be: Attributes taken for granted by customers; their presence does not create customer satisfaction (CS), but their absence or poor performance will result in high levels of customer dissatisfaction.
2. One-dimensional: CS is positively proportional to the fulfillment level of these attributes; the higher the level of fulfillment, the higher the CS and vice versa.
3. Attractive: Attributes not generally expected by customers; their presence will create high levels of CS, but their absence will not result in customer dissatisfaction.
4. Indifferent: Customers do not care about these attributes; their presence or absence will not affect levels of customer satisfaction or dissatisfaction.
5. Reverse: Their presence causes customer dissatisfaction, but their absence creates CS.

Kano classifications are identified via a Kano questionnaire, which contains a pair of questions for each product attribute. The question pair includes a functional question that captures the user’s perception if the product has a certain attribute, and a dysfunctional one that captures the user’s perception if the product does not have that attribute. **[Sarvestani]**

An example of a Kano questionnaire is illustrated below **[Witell]**:

Q: If you can order cinema tickets online, how do you feel (functional form)

1. I like it that way
2. I am expecting it to be that way
3. I am neutral
4. I can accept it to be that way
5. I dislike it that way

Q: If you can not order cinema tickets online, how do you feel (dysfunctional form)

1. I like it that way
2. I am expecting it to be that way
3. I am neutral
4. I can accept it to be that way
5. I dislike it that way

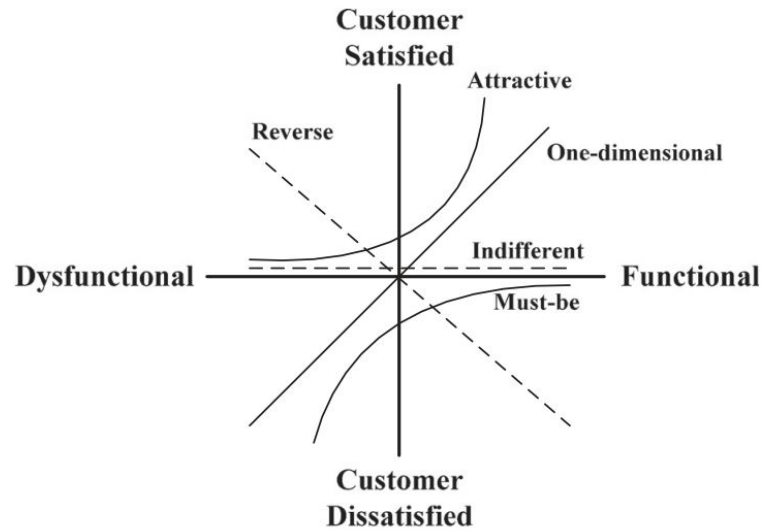


Fig. 1 Kano's model

Figure 1: Illustrating a generic Kano model, shows the impacts of the five attributes on a product's two-dimensional aspects (functionality and CS). [Sarvestani]

[Back to Table of Contents](#)

Section 6: Writing Quality User Requirements and Engineering Specifications

Now that you have developed a clearer idea of how to obtain user requirements and engineering specifications, it is time to consider how you will write and document *quality* statements that satisfy the basic properties of user requirements and engineering specifications described in earlier sections.

Some useful questions to ask yourself when creating user requirements:

- Is the requirement essential to the success of the product?
- Is the requirement clear?
 - If i gave the requirement to a designer or engineer, would he/she know what I meant by it?
- Is the requirement self-contained (i.e., can it stand alone and be understood without additional information)?
- Is each requirement actually a single requirement and not actually multiple requirements?
- Is the requirement solution neutral (i.e., does it not imply the use of one solution over another)?
- Is the requirement precise?
 - Does the requirement have only one interpretation? Is this interpretation obvious?
- Does the requirement avoid using potentially ambiguous words such as:
 - Vague subjects: "it" or "they"
- Does the requirement specify a "what" rather than a "how"?
- Is each requirement consistent with the other requirements? Are there direct conflicts?

- Does each requirement describe a unique aspect or are there requirements that describe the same thing?

[Makower]

Outside of thinking about the content of your user requirements, it is also important to make sure your collection of user requirements are *uniform*--that is, they are correct and consistent in terms of grammar and tense.

The following are a couple of examples of poorly written user requirements with a rationale for why they are not satisfactory:

- A. "The device should clean the floor by sucking debris off of the ground"
This requirement clearly specifies "how" the solution should clean the ground (by sucking debris), rather than "what" the solution should do (clean the ground).
- B. "The design should be easy to carry"
This requirement can actually be broken down into multiple requirements. "Easy to carry" can mean a number of things, including weight or dimensions. A broad user requirement such as this can be broken down by spending additional time understanding what the stakeholder(s) means by "easy to carry".

When creating engineering specifications, it is important to make sure the specifications are not:

- Ambiguous
- Incomplete
- Inconsistent
- Incorrect
- Infeasible
- Unusable
- Unverifiable

Some useful questions to ask yourself when creating engineering specifications **[Makower]**:

- Does each requirement have a value/number (engineering specification) associated with it?
- Does each engineering specification have units associated with it?
- Can the engineering specification be validated through testing?
 - Can you imagine a way to test the engineering specification?
- If all the engineering specifications were fulfilled, would the product be a success? If potentially not, then why?
- If the concept solution is [insert engineering specification(s)], is it necessarily [corresponding user requirement]?
 - i.e. If the suitcase is ≤ 2 lbs. is it necessarily "easy to carry"?

[Back to Table of Contents](#)

Section 7: Organizing and Displaying your User Requirements and Specifications

The templates and explanations below are two example of many ways to organize user requirements and specifications:

User Requirements Template:

- Column 1: Priority level—you must rank the user requirements in order of most important (ranked as 1) to least important. User requirements with the same level of importance may have the same priority level designation.
- Column 2: User requirement—provide a clear description of the user requirement that you have developed.
- Column 3: Justification—in the form of full sentences explain why the user requirement was included.
- Column 4: User requirement information sources—list the information source(s) that contributed to the user requirement developed.

Priority Level	User Requirement	Justification	User Requirement Information Sources
***Add as many rows as needed			

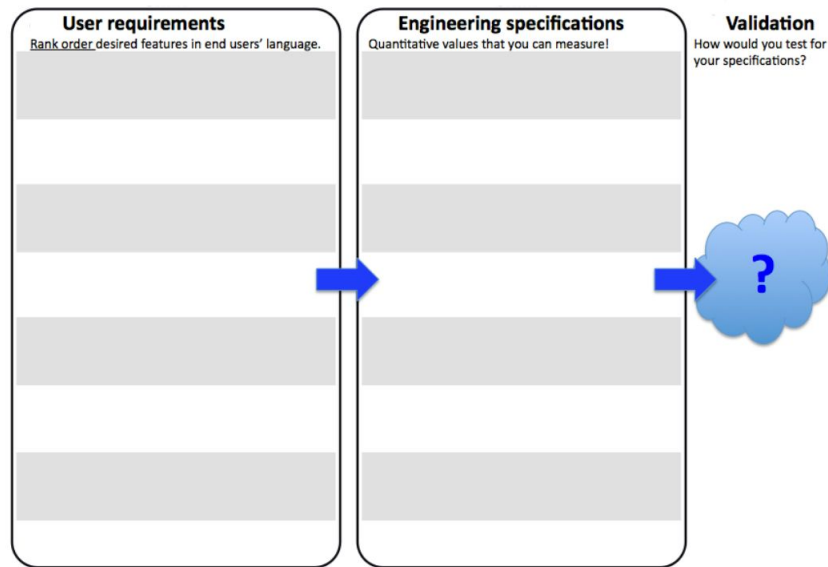
Engineering Specification Template:

- Column 1: User requirement—state the user requirement for which the engineering specification was developed.
- Column 2: Engineering specification(s)—state the engineering specification(s) that was developed, multiple engineering specifications can be used for a single requirement if needed.
- Column 3: Justification—indicate why this engineering specification is needed to satisfy the user requirement.
- Column 4: Engineering specification information sources—indicate what information was used to develop the specification, provide as much detail as possible so that future design engineers would know exactly what information went into the engineering specification. **[Mohedas]**

User Requirement	Engineering Specification	Justification	Engineering Specification Information Sources
***Add as many rows as needed			

The following template is another way to organize your requirements and specifications in relation to how you would validate your specifications:

User requirement / engineering specification table



[Back to Table of Contents](#)

Section 8: Common Pitfalls When Developing User Requirements

The [requirements elicitation] process represents a significant challenge for novice designers, as one must be prepared to use both technical and nontechnical skillsets. Prior studies comparing novice and expert designers have emphasized this challenge and its effect on final design quality. For example, a study of novices and experts performing a design task showed that novices spend less time gathering information and less time defining the scope of the design problem than experts. It has also been shown that novice designers who spend more time refining the scope of their design problems tend to produce higher quality designs.... Previous work has shown that novices understand the value and benefit of information gathering and synthesis while developing requirements; however, during execution they typically gather less information and perform less synthesis than originally planned. In addition, while novices understand the benefits of incorporating stakeholders' input and field-based observations into the requirements development process, they encounter obstacles and use stakeholder interactions to gain only superficial benefits.

Research has shown that novices do not tend to assess the quality and/or validity of the information they obtain prior to applying it to their problems. Similar results have been found for engineering students' use of internet sources through studies of design report bibliographies. [Mohedas]

Difficulties in Requirement Management

Customer requirements are normally qualitative and tend to be imprecise and ambiguous due to their linguistic origins. In most cases, requirements are negotiable and may conflict with one another, and thus tradeoffs are often necessary. Frequently, customers, marketing personnel, and designers employ different

sets of context to express the requirements. Differences in semantics and terminology always impair the ability to convey requirement information effectively from customers to designers. Distinguishing requirements in terms of CNs [customer needs] and FRs [functional requirements or product specifications] is of practical significance. An organization should put considerable effort into capturing the genuine or 'real' needs of the customers (CNs), rather than too much focus on the technological specifications (FRs) during the early stage of product development.

Second, there rarely exists any definitive structure of requirement information. Variables used to describe requirements are often poorly understood and expressed in abstract, fuzzy, or conceptual terms, leading to work on the basis of vague assumptions and implicit inference. A number of researchers have enforced a hierarchical structure or an AND/OR tree structure for the articulation of customer requirements, for example, the requirement taxonomy, the customer attribute hierarchy, and the FR topology. Nevertheless, the non-structured nature of requirement information itself coincides with those findings in natural language processing.

Third, the mapping relationships between CNs and FRs are often not clearly available at an early stage of design. Customers are often not aware of the underlying coupling and interrelationships among various requirements with regard to product performance. It is difficult, if not impossible, to estimate the consequences of specifying different requirements. Clausing discerns customer needs and product specifications and points out that the mapping problem between them is the key issue in the 'design for customers'.

Fourth, the specification of requirements results from not only the transformation of customer requirements from those of end-users, but also considerations of many engineering concerns. In practice, product development teams must keep track of a myriad of requirement information derived from different perspectives of the product life cycle, such as product technologies, manufacturability, reliability, maintainability, and environmental safety, to name a few. [Jiao]

The following article ([10 Requirement Traps to avoid by Karl Wiegers](#)) also describes major pitfalls when developing user requirements. Although the focus is on software engineering, the main points that Wiegers makes in his article can be applicable to any major design problem.

[Back to Table of Contents](#)

Section 9: Conclusion

In summary, user requirements and engineering specifications should do the following [Makower]:

- confirms the problem definition
- prioritizes stakeholder wants/needs
- provides data for decision making
 - concept selection
- provides design targets
 - engineering analysis
 - validation

[Back to Table of Contents](#)

References

1. Mohedas, I., Daly, S. R., & Sienko, K. H. (2015). Requirements development: Approaches and behaviors of novice designers. *Journal of Mechanical Design*. 137(7): 071407.
2. Makower, J., Yock, P., & Zenios, S. (2010). *Biodesign: The Process of Innovating Medical Technologies*. Cambridge, UK: Cambridge University Press. Retrieved from <http://mirlyn.lib.umich.edu/Record/013601217>
3. Gottesdiener, E. (March 2008). Good practices for developing user requirements. *The Journal of Defense Software Engineering*. 13-17.
4. Hsieh, H. C., Holland, R., & Young, M. (2009). A theoretical model for cross-cultural web design. *Human Centered Design*. 712-721.
5. Clemmensen, T. (2011). Templates for cross-cultural and culturally specific usability testing: Results from field studies and ethnographic interviewing in three countries. *International Journal of Human-Computer Interaction*. 27(7): 634-669
6. Dieter, G. E. (2009). *Problem definition and need identification*. Engineering design 4th ed. USA: McGraw-Hill Companies, Inc.
7. Sarvestani, A. S. (2015). *Designing task shifting medical devices for low-resource settings*. (Doctoral dissertation, The University of Michigan).
8. Mohedas, I., Daly, S. R., & Sienko, K. H. (2014). Design ethnography in capstone design: Investigating student use and perceptions. *International Journal of Engineering Education*. 30(4): 888-900.
9. Maguire, M., & Bevan, N. (August 2002). User requirements analysis. *Usability*. Proceedings of IFIP 17th World Computer Congress. *Springer US*. 133-148.
10. Tran, E. (1999). Requirements & specifications. Retrieved from http://users.ece.cmu.edu/~koopman/des_s99/requirements_specs/#traceability
11. Jiao, J., & Chen, C. (2006). Customer requirement management in product development: A review of research issues. *Concurrent Engineering: Research and Applications*. 14(3): 173-185.
12. Witell, L., & Lofgren, M. (2007). Classification of quality attributes. *Managing Service Quality: An International Journal*. 17(1): 54-73.

Some Additional Resources:

[Cultural influences on global health technologies](#)

[Software development article that highlights a list of user requirement models](#)

[Additional tips for user requirement development](#)