

SYSTEM DESIGN COMPONENTS

Input Design

In an information system, input is the raw data that is processed to produce output. During the input design, the developers must consider the input devices they need to use.

Therefore, the quality of system input determines the quality of system output.

Objectives for Input Design

The objectives of input design are –

- To design data entry and input procedures
- To reduce input volume
- To design source documents for data capture or devise other data capture methods
- To design input data records, data entry screens, user interface screens, etc.
- To use validation checks and develop effective input controls.

Data Input Methods

It is important to design appropriate data input methods to prevent errors while entering data. These methods depend on whether the data is entered by customers in forms manually and later entered by data entry operators, or data is directly entered by users on the PCs.

A system should prevent a user from making mistakes by –

- Clear form design by leaving enough space for writing legibly.
- Clear instructions to fill form.
- Clear form design.
- Reducing key strokes.
- Immediate error feedback.

Some of the popular data input methods are –

- Batch input method (Offline data input method)
- Online data input method

- Computer readable forms
- Interactive data input

Input Integrity Controls

Input integrity controls include a number of methods to eliminate common input errors by end-users. They also include checks on the value of individual fields; both for format and the completeness of all inputs.

Audit trails for data entry and other system operations are created using transaction logs which gives a record of all changes introduced in the database to provide security and means of recovery in case of any failure.

Well-designed input forms and screens have following properties –

- It should serve specific purpose effectively such as storing, recording, and retrieving the information.
- It ensures proper completion with accuracy.
- It should be easy to fill and straightforward.
- It should focus on user's attention, consistency, and simplicity.
- All these objectives are obtained using the knowledge of basic design principles regarding
 - What are the inputs needed for the system?
 - How end users respond to different elements of forms and screens.

Output Design

The design of output is the most important task of any system. During output design, developers identify the type of outputs needed, and consider the necessary output controls and prototype report layouts.

Objectives of Output Design

The objectives of output design are –

- To develop output design that serves the intended purpose and eliminates the production of unwanted output.
- To develop the output design that meets the end user requirements.
- To deliver the appropriate quantity of output.
- To form the output in appropriate format and direct it to the right person.
- To make the output available on time for making good decisions.

Types of outputs

External Outputs

Manufacturers create and design external outputs for printers. External outputs enable the system to leave the trigger actions on the part of their recipients or confirm actions to their recipients.

Some of the external outputs are designed as turnaround outputs, which are implemented as a form and re-enter the system as an input.

Internal outputs

Internal outputs are present inside the system, and used by end-users and managers. They support the management in decision making and reporting.

There are three types of reports produced by management information –

- *Detailed Reports* – they contain present information which has almost no filtering or restriction generated to assist management planning and control.
- *Summary Reports* – they contain trends and potential problems which are categorized and summarized for managers who do not want details.
- *Exception Reports* – they contain exceptions, filtered data to some condition or standard before presenting it to the manager, as information.

Output Integrity Controls

Output integrity controls include routing codes to identify the receiving system, and verification messages to confirm successful receipt of messages that are handled by network protocol.

Printed or screen-format reports should include a date/time for report printing and the data. Multipage reports contain report title or description, and pagination. Pre-printed forms usually include a version number and effective date.

Output media

- Paper
- Screen
- Microfilm/Microfiche
- Video/Audio
- CD, DVD
- Other electronic media

Forms Design

Both forms and reports are the product of input and output design and are business document consisting of specified data. The main difference is that forms provide fields for data input but reports are purely used for reading. For example, order forms, employment and credit application, etc.

- During form designing, the designers should know –
 - who will use the forms
 - where would they be delivered
 - the purpose of the form or report
- During form design, automated design tools enhance the developer's ability to prototype forms and reports and present them to end users for evaluation.

Objectives of Good Form Design

A good form design is necessary to ensure the following –

- To keep the screen simple by giving proper sequence, information, and clear captions.
- To meet the intended purpose by using appropriate forms.
- To ensure the completion of form with accuracy.
- To keep the forms attractive by using icons, inverse video, or blinking cursors etc.
- To facilitate navigation.

The Components of Forms

The typical form has following five components:

- *Structure*. It includes ordering for fields, appearance on the page and logical connections between multiple fields.
- *Input fields*. They include text fields, password fields, check boxes, radio buttons, sliders and any other field designed for user input.
- *Field labels*. They tell users what the corresponding input fields mean.
- *Action buttons*. When user presses the button, the action is performed (such as submitting the data).
- *Feedback*. User understands the result of the input by a feedback. Most apps or sites use messages as a form of feedback. Messages notify the user about result; they can be positive (indicating that the form was submitted successfully) or negative (“The number you’ve provided is incorrect”).

Forms could also have following components:

- *Assistance*. Any help that explains how to fill out the form.
- *Validation*. Automatic check that ensures that user’s data is valid.

Rules While Designing Forms

1. Keep it short

Eliminating unnecessary fields requires time but increases the conversion rate.

2. Visually group related labels and fields

Labels should be close to the fields. Use some white space to group or separate elements visually.

If the form is long, you can break it down to steps. One step per page.

If your form asks about different topics, section it into separate groups of fields and name the groups accordingly.

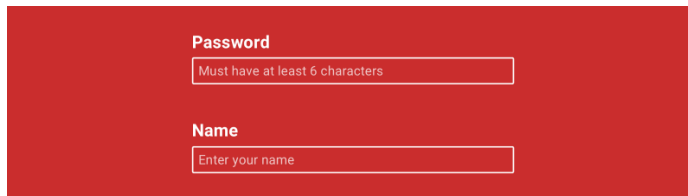
3. Present fields in a single column layout

Multiple columns interrupt the vertical momentum of moving down the form.

There are some exceptions to this rule: short and/or logically related fields such as City, State, and Zip Code can be presented on the same row.

4. Use logical sequencing

Stick to standard sequences and consider regional or cultural differences such as date format.



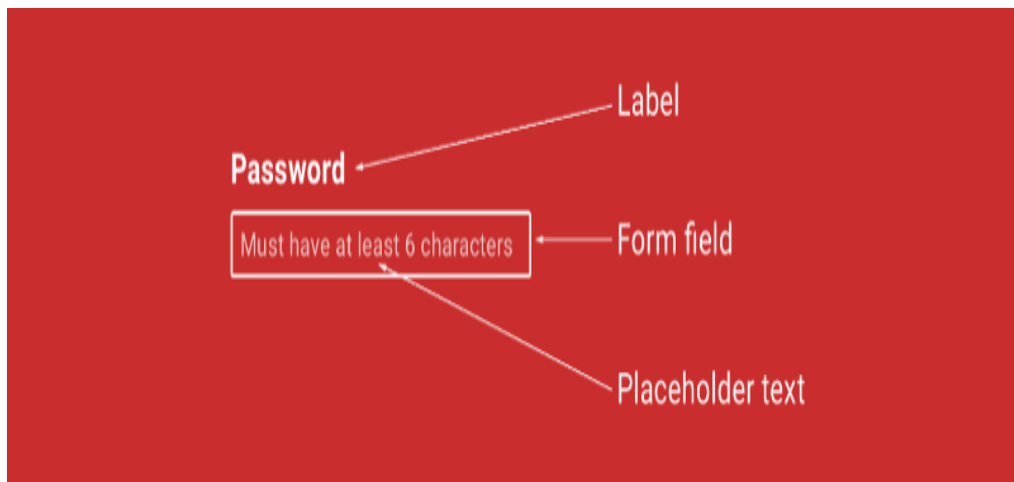
A form with two fields. The first field is labeled "Password" and has a placeholder text "Must have at least 6 characters". The second field is labeled "Name" and has a placeholder text "Enter your name".

It's important to consider the cultural difference in sequencing information. Dates as an example, Japan, Europe and the US all have a different way of displaying it.

5. Avoid placeholder text as a label

Without labels, users cannot check their work before submitting a form. Disappearing placeholder text strains users' short-term memory.

A placeholder is supposed to help users understand what's required from them.



A form with a Password field. The field has a placeholder text "Must have at least 6 characters". The field is labeled "Password". The placeholder text is also labeled "Placeholder text". The field is also labeled "Form field".

6. Match fields to the type and size of the input field

In this instance, if your form requires a 3 digits security number, design for that size. Imagine if the security number field was as big as the rest of the fields? This would easily create doubts and confusion.

7. Distinguish optional and required fields

No more asterisk symbol* in forms. If you don't need the information, don't ask for it.

If, for marketing reasons, you want to collect more data than what you need, you can always ask for it later.

8. Explain any input or formatting requirements

It's important to explain what you require from the user. This prevents errors which results on reducing the completion time.

PASSWORD:

Must be 6 or more characters and contain at least 1 number

9. Avoid Reset and Clear buttons

Thankfully this is not a common practice but you can still see it in old websites.

Card number:	<input type="text"/>	
Exp. date:	Month: <input type="text"/>	Year: <input type="text"/>
3-digit security code:	<input type="text"/> (on back of card)	
Name on card:	<input type="text"/>	
Billing Address:	<input type="text"/>	
City:	<input type="text"/>	
State:	<input type="text"/>	
Zip:	<input type="text"/>	

10. Provide highly visible and specific error messages

Be specific about the error message otherwise users won't know what went wrong and what's required from them.

User interface design

Types of user interface

1. Natural language

2. Question-and-answer
3. Menu
4. Form-fill
5. Command-language
6. Graphical
7. Touch-screen

Choosing Interface Elements

Interface elements include but are not limited to:

- *Input Controls*: buttons, text fields, checkboxes, radio buttons, dropdown lists, list boxes, toggles, date field
- *Navigational Components*: search field, pagination, slider, tags, icons
- *Informational Components*: tooltips, icons, progress bar, notifications, message boxes, modal windows
- *Containers*: accordion - An accordion is a vertically stacked list of items that utilizes show/hide functionality. When a label is clicked, it expands the section showing the content within. They can have one or more items showing at a time and may have default states that reveal one or more sections without the user clicking

Best Practices for Designing an Interface

Everything stems from knowing your users, including understanding their goals, skills, preferences, and tendencies. Once you know your user, make sure to consider the following when designing your interface:

- *Keep the interface simple*. The best interfaces are almost invisible to the user. They avoid unnecessary elements and are clear in the language they use on labels and in messaging.
- *Create consistency and use common UI elements*. By using common elements in your UI, users feel more comfortable and are able to get things done more quickly. It is also important

to create patterns in language, layout and design throughout the site to help facilitate efficiency. Once a user learns how to do something, they should be able to transfer that skill to other parts of the site.

- *Be purposeful in page layout.* Consider the spatial relationships between items on the page and structure the page based on importance. Careful placement of items can help draw attention to the most important pieces of information and can aid scanning and readability.
- *Strategically use color and texture.* You can direct attention toward or redirect attention away from items using color, light, contrast, and texture to your advantage.
- *Use typography to create hierarchy and clarity.* Carefully consider how you use typeface. Different sizes, fonts, and arrangement of the text to help increase scanability, legibility and readability.
- *Make sure that the system communicates what's happening.* Always inform your users of location, actions, changes in state, or errors. The use of various UI elements to communicate status and, if necessary, next steps can reduce frustration for your user.
- *Think about the defaults.* By carefully thinking about and anticipating the goals people bring to your site, you can create defaults that reduce the burden on the user. This becomes particularly important when it comes to form design where you might have an opportunity to have some fields pre-chosen or filled out.

Nielsen and Molich's 10 User Interface Design Guidelines

- *Visibility of system status.* Users should always be informed of system operations with easy to understand and highly visible status displayed on the screen within a reasonable amount of time.
- *Match between system and the real world.* Designers should endeavor to mirror the language and concepts users would find in the real world based on who their target users are. Presenting information in logical order and piggybacking on user's expectations derived from their real-world experiences will reduce cognitive strain and make systems easier to use.
- *User control and freedom.* Offer users a digital space where backward steps are possible, including undoing and redoing previous actions.

- *Consistency and standards.* Interface designers should ensure that both the graphic elements and terminology are maintained across similar platforms. For example, an icon that represents one category or concept should not represent a different concept when used on a different screen.
- *Error prevention.* Whenever possible, design systems so that potential errors are kept to a minimum. Users do not like being called upon to detect and remedy problems, which may on occasion be beyond their level of expertise. Eliminating or flagging actions that may result in errors are two possible means of achieving error prevention.
- *Recognition rather than recall.* Minimize cognitive load by maintaining task-relevant information within the display while users explore the interface. Human attention is limited and we are only capable of maintaining around five items in our short-term memory at one time. Due to the limitations of short-term memory, designers should ensure users can simply employ recognition instead of recalling information across parts of the dialogue. Recognizing something is always easier than recall because recognition involves perceiving cues that help us reach into our vast memory and allowing relevant information to surface. For example, we often find the format of multiple-choice questions easier than short answer questions on a test because it only requires us to recognize the answer rather than recall it from our memory.
- *Flexibility and efficiency of use.* With increased use comes the demand for less interactions that allow faster navigation. This can be achieved by using abbreviations, function keys, hidden commands and macro facilities. Users should be able to customize or tailor the interface to suit their needs so that frequent actions can be achieved through more convenient means.
- *Aesthetic and minimalist design.* Keep clutter to a minimum. All unnecessary information competes for the user's limited attentional resources, which could inhibit user's memory retrieval of relevant information. Therefore, the display must be reduced to only the necessary components for the current tasks, whilst providing clearly visible and unambiguous means of navigating to other content.
- *Help users recognize, diagnose and recover from errors.* Designers should assume users are unable to understand technical terminology, therefore, error messages should almost always be expressed in plain language to ensure nothing gets lost in translation.
- *Help and documentation.* Ideally, we want users to navigate the system without having to resort to documentation. However, depending on the type of solution, documentation may be

necessary. When users require help, ensure it is easily located, specific to the task at hand and worded in a way that will guide them through the necessary steps towards a solution to the issue they are facing.

Database design

Objectives	Guidelines
<ul style="list-style-type: none"><input type="checkbox"/> Purposeful information retrieval<input type="checkbox"/> Efficient data storage<input type="checkbox"/> Data availability<input type="checkbox"/> Efficient data update and retrieval<input type="checkbox"/> Data integrity	<ul style="list-style-type: none"><input type="checkbox"/> Use ER diagram for data modeling<input type="checkbox"/> All tables are normalized

Data integrity

- Entity integrity: the primary key of a table cannot have a null value
- Referential integrity: all foreign keys in a child table must have a matching record in the parent table
- Domain integrity: table entries must be of the same type, limit, range and other validation checks.