

Requirements Cheat Sheet

HAVE YOU CONSIDERED THESE?



IS THERE A TIME CONSTRAINT?

Does any part of your requirement depend on time? Should the system behave differently based on time parameters?



HAVE YOU HANDLED ALL THE ERRORS?

What are the possible errors and failure points with these requirements? What should the system do when each of these errors occur? Is it an error or a warning or a recommendation? Can you help the user recover?



WHAT IS THE NEXT ACTION?

When the user is achieving this requirement, what is the next step they will likely take? Can you make the next action obvious so they do not have to search to find it? Do they have a choice not to take the next step action without losing their work?



IS THIS THE BEST REPRESENTATION OF THE DATA

Can you show this requirement in a better way? Is this the best visualization to fit the type of data and what it will be used for?



HAVE YOU HANDLED LARGE /EMPTY DATA SETS?

How will your requirements handle very large data? How can you show large data so that it is still consumable? If the data is empty, how does that affect your requirements?



DO YOU NEED A SPECIFIC ORDER?

Do you need your requirements to follow a certain order? Is there a default sort order for your tables? Should some actions happen before others?



DO YOU NEED DEFAULTS?

Have you anticipated the user's actions to see what are the best defaults? Do defaults apply in these requirements or should the user be more thoughtful about the input?



HAVE YOU HANDLED INTEGRATIONS?

Do you have data that will be pushed/pulled to other systems? Has your requirements handled getting data from these end points?



ARE AUXILIARY FUNCTIONS AFFECTED?

Will your requirement affect auxiliary functions such as: imports/exports, reports, change logs, audit logs, zoom capabilities etc....



HOW DO YOU ONBOARD?

How can you help users get started better? Can your requirement make it easier for them to be up and running?