1. **What criteria should be used in choosing an appropriate requirements engineering tool?**

Carrillo de Gea et al. (2021) provided an alternative list of 17 items to be used in the evaluation of RE tool capabilities:

* Organization of requirements with metadata, attributes, and reuse
* Reports, database queries, and open interface language
* Internal checks, that is, consistency, dependencies, and history
* Traceability support, that is, drag and drop (horizontal and vertical)
* Providing support for reuse
* Remote working, cloud only
* Multiple views of requirements
* Performance
* Collaboration, workflow management
* Easily adapted and integrated into business processes
* Federation and notification with ALM/PLM tools
* Export/import with standard formats
* Macros for repeated commands
* Training and learning curve effort
* Agile, CI/CD, and DevOps
* Intelligent support
* Scalability

This checklist (or an adapted version) and an appropriate consensus management approach (e.g., Wideband Delphi, AHP) can be used to select the right tool for a team or enterprise.

1. **Are there any drawbacks to using certain tools in requirements engineering activities?**
2. **When selecting an open-source tool, what characteristics should you look for?**
3. **How can tools enable distributed, global requirements engineering activities? What are the drawbacks in this regard?**
4. **If an environment does not currently engage in solid requirements engineering practices, should tools be introduced?**
5. **What sort of problems might you find through a traceability matrix that you might not see without one?**

Requirements Source Traceability Matrix

Yet another kind of traceability matrix links requirements to their sources. Aside from those coming directly from users, many requirements are derived from governmental regulations and from standards. Linking the requirements to these sources can be very helpful when the sources change. Table 9.7 shows the typical format for such a traceability matrix.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Requirement ID | Federal Regulation #1 | Federal Regulation #2 | State Regulation #1 | State Regulation #1 | International Standard #1 |
| 3.1.1.3 | X |  |  |  |  |
| 3.1.2.9 | X | X |  |  |  |
| 3.2.1.8 |  |  | X | X |  |
| 3.2.2.5 |  |  | X |  |  |
| 3.2.2.6 |  |  |  |  | X |
| 3.3.1 |  | X |  |  |  |
| 3.3.2 |  | X |  |  |  |
| 3.4.1 |  | X |  |  |  |
| 3.4.3 |  | X |  |  |  |
| 3.4.4 |  | X |  |  |  |
| 3.6.5.1 |  |  |  | X |  |
| 3.6.6.4 |  |  |  |  | X |

*Table 9.7 Traceability Matrix Showing Requirements and Their Sources*

This kind of traceability matrix is especially useful for tracking nonfunctional requirements. Failure to trace nonfunctional requirements throughout the project life cycle can be a significant problem (Kassab et al. 2008).

1. **How is AI being proposed for knowledge acquisition and representation in requirements specifications?**

In the past few years, the thread of work on Artificial Intelligence (AI) for RE has made strides in rigorously investigating how general-purpose AI tools can be tailored best for RE tasks (Dalpiaz and Niu 2020). For example, recent research studies are investigating how human intervention in the requirement gathering processes can be reduced by using “Speech Understanding Methodology” techniques with the capability to “listen in” on a conversation and suitably collect stakeholders’ declarations into a distinct vision (Spoletini and Ferrari 2017; Sharma and Pandey 2014).

Speech understanding methodology can be combined then with “Automatic Keywords Mapping,” another AI technique being investigated that can enhance the requirements elicitations. Many requirements issues are related to not having the stakeholders able to depict their requirements properly, or the ignorance of the domain experts and developers to “observable” words that lead essentially to system requirements. These issues can be eliminated by automatically mapping every keyword spoken by each stakeholder. The earlier studies released a keyword mapping technique for designers so that they can recognize the keywords used by stakeholders to assist them in making ideal requirements (Sharma and Pandey 2013).

Case-based reasoning is also being investigated for requirement elicitation, which can reduce the problem of natural language understanding as well as save the time of the requirement expert. There has also been recent research on the use of machine learning algorithms to identify user preferences based on their sentiment (Li et al. 2018).