IREB

SIG #AIREB booklet

V 40

CPRE IREB

Prompt engineering guidelines

"Before using LLMs for Requirements Engineering make sure that you are familiar with basic prompt engineering guidelines such as OpenAI's prompt engineering guide."

The first topic of discussion in the broad context of Artificial Intelligence (AI) and Requirements Engineering (RE) was Large Language Models (LLMs). Andrea Herrmann began with the basics, emphasizing the importance of following prompt engineering guidelines to achieve satisfactory results when using AI for RE.

The guidelines discussed were freely available ones, such as OpenAI's prompt engineering guidelines for their LLM ChatGPT.

Explore different sources for prompt engineering guidelines to gain diverse perspectives on various prompting tactics and find the approach that works best for you and your desired output!

By Andrea Herrmann and the IREB AI interest group

CPRE IREB

Prompting frameworks

"Use prompting frameworks like CRISPE or RICE and try out others."

Using prompting frameworks can help you interact with AI more easily, as these approaches are specifically designed to assist users with their prompting.

- <u>CRISPE</u>

  CRISPE focuses on providing context, role, input, steps, precision, and examples.

- <u>RICE</u>

  RICE helps with defining the role, providing information, adding context, and giving examples.

By Thomas Immich and the IREB AI interest group

CPRE IREB

Apply methods step-wise

"Your results will be very different if you prompt the AI step by step compared to a rather long and comprehensive prompt. For instance, instead of asking the AI to generate a use case model, rather ask to extract the actors and use cases separately etc."

If your prompt includes many details, prompting them all at once may lead to errors and unsatisfactory output. Instead, apply your methods step by step.

For example, when you want an AI (e.g. Large Language Model) to execute an FMEA analysis, then do not ask it to do the FMEA analysis at once, but rather do it step-wise. First let the AI analyze the product´s structure and list the components. Then, you review this list of components and correct if needed. Then, you execute the next step, the function analysis etc.

By Andrea Herrmann and the IREB AI interest group

CPRE IREB

Use RE standards

“Pretrained language models know many of the requirements engineering standards. So, you can refer to them and ask the AI to use them.”

A detailed prompt doesn’t need to be long. Instead of taking a long time to feed the large language model information on Requirements Engineering, utilize its pretrained knowledge. Yes, LLMs are familiar with RE standards too! By simply referring to them, you can shorten your prompt, reduce the risk of errors due to excessive information, and simplify the process for yourself.

By Andrea Herrmann and the IREB AI interest group

Define quality criteria

“When creating prompts for Requirements Engineering refer to known quality criteria such as INVEST or others like CPRE or ISO/IEC 25010.”

Referencing is key. Refer to known concepts, rules, or in this case, quality criteria, such as:

- **INVEST** (Independent, Negotiable, Valuable, Estimable, Small, Testable): A set of criteria for agile software development.
- **CPRE** (Certified Professional for Requirements Engineering): Our certification focusing on the knowledge and skills needed for high-quality Requirements Engineering practices.
- **ISO/IEC 25010** (Systems and software Quality Requirements and Evaluation – SQuaRE): An international standard defining a model for software quality.

Try straightforward prompts: ”Transform the following text into user stories. Make sure that the user stories conform to the INVEST criteria.”

By the IREB AI interest group

CPRE IREB

Prompts are also requirements

"Remember prompts are also requirements. When writing prompts keep in mind all the criteria for good requirements. Such as concise acceptance criteria."

When crafting prompts, remember they need to meet the same quality criteria as good requirements. Just like requirements, your prompts should be clear and precise.

Dive into the CPRE Foundation Level handbook to discover more about quality criteria for work products and requirements.

By the IREB AI interest group

Repeat the same prompt

“Using the same prompt several times may lead to different results. By doing so, you can get more results or get an idea of the variants possible.”

Not satisfied with your first AI-generated result? Try again! And even if you are, don't immediately settle.

Most language models rarely produce identical answers, even with the same prompt. By rerunning your query multiple times, tweaking your prompt slightly, or using different LLMs, you can gather different outputs and create your desired result.
Don't forget to save your chat logs! For easier comparison of different outputs as you progress, it's helpful to keep access to your previous conversations. Once you have your final results, you might want to consider deleting unnecessary chats to manage your data efficiently.

This method gives you multiple perspectives and keeps your work process dynamic and creative – even with AI assistance.

By Andrea Herrmann and the IREB AI interest group

CPRE IREB

Validation is key

“Always validate the results from the AI. Using AI is helpful, but validating the requirements for contextual fit is even more important. Validation of user requirements is equally important. Do not lose focus on high quality requirements for the sake of supposedly easier elicitation.”

It is easy to be inclined to take AI results as immediate truths, perhaps out of convenience or because of the widespread narrative that they are inherently intelligent. However, users often encounter AI hallucinations: outputs that are not based on any factual information and appear without clear reason. Regardless of the amount of data we input to get perfectly tailored results, users have the uniquely human advantage of accurate and detailed contextual information and critical thinking. Therefore, it is crucial to take the time to validate AI outputs with our own knowledge and information from other reliable sources. Just as products, articles, and decisions in human-to-human contexts go through several stages of validation, so must human-to-AI interactions.

By Michael Tesar and the IREB AI interest group

Role play

“You need to assign the AI a specific role to get better results. Provide some information about your [DOMAIN].”

Role-playing might not be the first strategy that comes to mind when interacting with Large Language Models (LLMs), but it’s a surprisingly effective approach. By assigning a specific role or character to the AI, you can tailor its responses to suit what you are looking for more precisely.

For example, you can begin the conversation with a specific prompt, such as, "Imagine you are an expert in Requirements Engineering with decades of experience in [DOMAIN]."

Keep playing around with different characters to find the one that best fits the output you’re looking for!

By Michael Mey and the IREB AI interest group

Humanizing LLM interactions

“The AI produces significantly better results if you ask it to go step-by-step and add some human touch to it. Add this information to your prompt.”

We've previously discussed the importance of feeding information to an LLM step-by-step. This approach helps the model with considering every piece of information without overlooking anything in the rush of data. We've also touched on the strategy of role-playing in our post last week, where you assign the LLM a specific role depending on your needs.

However, in this snippet, we want to emphasize adding a human touch—not by assigning a role but by interacting with the LLM in a more human-like manner. For example, you could say, "Take a deep breath, and let's do this step-by-step". This kind of communication sets the tone for the interaction. By talking to it as you would to a human, the LLM is more likely to respond with a human touch.

By Michael Mey and the IREB AI interest group

CPRE IREB

Use examples and variables 1/2

“Provide examples of the type of result that you are interested. Ensure the examples are clearly marked as such.”

Generative AI performs best when provided with high-quality and sufficient input. While it's important to explain your personal wants, much like in human interactions, points often come across more clearly when examples are used to illustrate the situation or desired output.

When providing examples, it’s usually best to stick to just one. This prevents the AI from skipping over important details or becoming overloaded with information. A single, precise, and accurate example of what you're looking for is more than enough and can significantly improve the quality of the output.

Use examples and variables 2/2

Additionally, it's important to clearly indicate that what you're providing is an example. You can preface it by saying that you'll give an example and consider putting the example in quotation marks for emphasis.

For instance, you might say: "Extract all quality requirements from the document I gave you. Stick to a clear wording and format as provided in the <EXAMPLE> which you will find below…."

By Michael Mey and the IREB AI interest group

CPRE IREB

Get the rationale 1/2

“In most RE automation tasks via LLMs, it is worth asking the LLM to produce their rationale for any decision. For example, for a classification task of classifying requirements as functional and non-functional (with different categories), generating the rationale at the end of each decision can increase the analyst's confidence in the output generated and forces the LLM to produce consistent results.”

When working with outputs from Large Language Models (LLMs), it's tempting to accept and use the results immediately, especially if they seem accurate at first glance. However, **to ensure greater reliability and consistency in future results**, **it's beneficial to ask the LLM to explain its rationale**. By requesting an explanation for its output, you not only gain a clearer understanding of the model’s reasoning but also establish a clearer thread of how it arrived at its conclusions. This can be invaluable for identifying any flaws and for fostering better collaboration between the user and the LLM.

Get the rationale 2/2

Additionally, prompting the LLM to provide detailed justifications helps in maintaining consistency and improving the quality of automation, as it makes potential machine hallucinations more detectable.

Implementing this practice in your interactions can significantly improve the effectiveness of your automation processes!

By Chetan Arora and the IREB AI interest group

Prepare and prompting 1/2

"By using documents, domain knowledge or historical data in your prompts, you can direct AI outputs to meet your exact needs. In Requirements Engineering for instance, providing past project requirements as context or templates can lead to more precise and relevant responses. Always comply with data protection regulations and confidentiality obligations. This high-level fine-tuning optimizes your AI use case without requiring technical skills. By applying the prompt engineering guidelines discussed in the first post and following the prompt frameworks mentioned in the second post, you can harness a powerful tool that delivers tailored results, reducing the need for adjusting the output repeatedly."

Prepare and prompting 2/2

Even when using LLMs, achieving high-quality results requires thorough preparation. To get the most out of it, you should provide relevant data, such as domain knowledge or past project requirements, which helps narrow the scope and guide the model toward more accurate results. Without this, the AI's responses might be too broad, lacking focus on what truly matters to you.

The next page will provide detailed prompting guidelines with examples. By combining the right data with effective prompting techniques, you can ensure that the output perfectly aligns with your wants and needs!

By Franz Zehentner and the IREB AI interest group

Iterative improvement 1/4

The last page shows how use cases are familiar to some LLMs and can serve as templates. If results aren't ideal, you can easily instruct the AI to improve them. We'll continue with the use case from the previous page and ask ChatGPT-4 to correct the actor's name.

Prompt: Can you please adjust Use Case 3: Update Delivery Status. The actor shall be the printer instead of the delivery system.

ChatGPT: Certainly! I'll adjust the Use Case 3: Update Delivery Status, changing the actor from the delivery system to the printer, which likely refers to the entity responsible for managing the initial stages of order fulfillment, such as printing or packaging the Use Case 3: Update Delivery Status.

Iterative improvement 2/4

Use Case Name: Update Delivery Status
Actor: Printer
Preconditions: An order has been placed and is ready for the printing or packaging phase.
Postconditions: The delivery status of the order is updated in the system.
Main Success Scenario (Basic Flow):

1. Printer processes the order for printing or packaging.
2. Once processing is complete, Printer updates the order status in the system as "Ready for Shipping."
3. System reflects the updated status, which is accessible to the customer and the delivery team.

Iterative improvement 3/4

Extensions (Alternative Flows):

1a. Printing or packaging issues:

1. Printer identifies an issue with printing or packaging.
2. Printer updates the order status as "Printing/Packaging Delayed."
3. System notifies the customer support team to inform the customer and manage expectations.

Special Requirements: System must support real-time updates and notifications to ensure timely communication between the printer, delivery team, and customer.
Trigger: Completion of printing or packaging of an order.
Frequency of Use: Moderate to high, depending on order volume.

CPRE IREB

Iterative improvement 4/4

Not much was expected to change here; only the actor should have been updated to "printer." However, ChatGPT made some assumptions before providing the updated use case, resulting in a completely different outcome. This error highlights the importance of carefully reviewing every output from ChatGPT, even when only a minor change is requested.

By Andrea Herrmann and the IREB AI interest group – and ChatGPT 4

CPRE IREB

Creating personas

If you read our previous page on **Role Play**, where we discussed assigning roles to AI for better results, you'll see a similar approach here: you can prompt the AI to create personas for specific domains, just like assigning it an expert role.

During the elicitation phase personas help to capture the big picture and allow the requirements engineer to see the goal from different perspectives. Stakeholders who are not always available, or not available at all, can be described using this technique.

Having a set of personas available for all projects also ensures consistency in how stakeholder needs are translated into requirements across different initiatives. The team can align with the understanding of the target users/stakeholders.

You can prompt for a persona definition like this:
"Create a detailed definition of a persona for a user of a fitness tracking app" and use the resulting definition to create a repository of personas for your projects.

By Franz Zehentner and the IREB AI interest group

Requirements Negotiation and Prioritization 1/3

LLMs can aid in requirements negotiation and prioritization, enhancing decision-making in early development. The example below shows how LLMs act as agents for different stakeholders, ensuring balanced outcomes that align with user needs and technical constraints.

Example Prompt:

1) Create two agents:

- **Agent1 (A1):** Represents the primary user <of your system>.
- **Agent2 (A2):** Represents the system's software architect (or any other stakeholder).

2) Task:

A1 and A2 will negotiate and discuss <REQ_IDs> to determine a priority list for the system's requirements.

Requirements Negotiation and Prioritization 2/3

3) Agent Focus:

- **A1:** Focuses on user experience and <insert specific user needs>.
- **A2:** Considers technical feasibility, integration with existing systems, and the overall architectural perspective.

4) Discussion Dynamics:

The agents can sometimes have differing opinions, leading to a more nuanced and realistic discussion. No decisions should violate the <constraints and non-functional requirements>.

Requirements Negotiation and Prioritization 3/3

Expected Output Format:

The final output should present:

- The **functional requirement IDs** in decreasing order of priority.
- A brief **rationale** for each requirement's position, based on the negotiation outcomes between A1 and A2.

Utilizing LLMs for requirements negotiation and prioritization streamlines early-stage decision-making and effectively balances user needs with technical constraints. By adopting this approach, teams can achieve more efficient, objective, and well-rounded outcomes, ultimately leading to smoother development processes.

By Chetan Arora and the IREB AI interest group

Principle 1 - Value Orientation 1/2

In our CPRE Foundation Level, you will learn about the **Nine Fundamental Principles of Requirements Engineering** that we have established. The first principle is Value Orientation, which asserts that requirements must add value and help achieve stakeholder satisfaction. This principle emphasizes that requirements are a means to an end, not an end in themselves.

We define the value of a requirement as the benefit it provides, which includes its role in building effective systems and reducing the risks of failure and costly rework. However, ensuring that all requirements align with this principle can be a time-consuming effort. Fortunately, LLMs can help with this process.

CPRE IREB

Principle 1 - Value Orientation 2/2

To start, describe your business goals and stakeholders to the LLM of your choice, and add the following prompt:

"You are an expert in requirements engineering, following the principles outlined by the International Requirements Engineering Board (IREB). Specifically, you focus on evaluating requirements according to the "Value Orientation" principle from IREB. I will provide a set of requirements, and your task is to assess each requirement by addressing the following:
Does the requirement align with the business goals or project objectives?
Does the requirement address the real needs or problems of the stakeholders?
Is the value or benefit of this requirement to the stakeholders explicitly clear?
List all requirements that are in conflict with this principle and elaborate on the discrepancies."

Evaluate the outcome of this conversation, and as always, tailor your prompt to your needs to achieve the best possible output.

By Franz Zehentner and the IREB AI interest group

Principle 2 – Stakeholder Orientation 1/2

Stakeholders define the goals of the project making their identification crucial for the project's success. The second principle "Stakeholder Orientation" in the IREB CPRE handbook revolves around this very topic.

Stakeholder Orientation focuses on understanding the desires, needs, and expectations of all project stakeholders to reduce the risk of project failure or dissatisfaction. This approach recognizes that stakeholders, including users, clients, operators, and regulators, have varied roles and influences.

A good starting point for stakeholder identification can be existing material like meeting notes, brainstorming material, and other project documents, which contain valuable information about potential stakeholders even before the elicitation process begins.

Principle 2 – Stakeholder Orientation 2/2

Large Language Models (LLMs) can analyze these documents and provide a list of possible stakeholders. The following prompt can get you started:

"I will provide you with documents regarding a project called ____. It is about ____ . Your task is to identify all possible stakeholders. For each identified stakeholder provide a name (or role if a name cannot be found) and the reason why they have an interest in the project."
Remember to validate the output according to your project's setting. This list is the first step of eliciting stakeholders for your project."

Remember to validate the output according to your project's setting. This list is the first step of eliciting stakeholders for your project.

By Franz Zehentner and the IREB AI interest group

CPRE IREB

Principle 3 – Shared Understanding 1/2

A common understanding among all stakeholders is essential for successful Requirements Engineering. LLMs are particularly good at fostering this shared understanding by identifying ambiguities and identifying key terms to be defined.

Here are two different use cases where an LLM can be particularly helpful:

Identifying ambiguities and misinterpretations: The LLM can analyze requirements to identify areas of ambiguity or misinterpretation. By rephrasing requirements and providing alternative perspectives, it helps ensure that both explicit and implicit shared understanding is captured and verified.

Example prompt: "You are an assistant helping a requirements engineer. Based on the following documented requirements, identify any potential areas of ambiguity or misinterpretation and suggest ways to improve clarity: [Insert requirements here]"

Principle 3 – Shared Understanding 2/2

Generating a glossary for common understanding: The LLM can generate a glossary of key terms used in the requirements, ensuring that all stakeholders have a common understanding of important terminology. This helps reduce misunderstandings and ensures consistent communication throughout the project.

Example prompt: "You are an AI assistant helping a requirements engineer. Based on the following documented requirements, generate a glossary of key terms along with their definitions to ensure a common understanding among stakeholders: [Insert requirements here]"

The first approach tackles ambiguities in individual requirements, while the second use case aims for common understanding by providing context for key terminology. As always make sure you check the output as there is no guarantee for the correct output.

By Franz Zehentner and the IREB AI Interest Group

Leveraging LLMs for Effective RE 1/2

In Requirements Engineering, having a deep understanding of domain-specific knowledge is crucial. Often, you need to know the unique rules, regulations, or workflows of the specific industry your project is within. While it's important to research outside of LLMs—since they can occasionally hallucinate or omit information—LLMs can be a helpful assistant by giving quick access to a large repository of information across various domains.

Whether dealing with healthcare regulations, financial compliance, or emerging technologies, LLMs can help you grasp complex concepts quickly. For example, they can provide helpful overviews and lists of key areas to focus on. They can also offer specific details that you can fact-check, significantly shortening research time. If you come across particularly dense documents or technical jargon that's hard to decode, pasting sections into an LLM and asking for explanations can simplify the process.

Leveraging LLMs for Effective RE 2/2

They can also assist in formulating precise requirements by offering insights into best practices and industry standards. Integrating LLMs in elicitation and specification allows you to bridge knowledge gaps, ask more informed questions during stakeholder interviews, and ensure better coverage.

By Chetan Arora and the IREB AI interest group

GI RE Working Group Annual Meeting

On Thursday and Friday, November 28 and 29, 2024, we had the pleasure of attending the annual meeting of the **GI Working Group on Requirements Engineering (#GIRE)** in Cologne, Germany. This year's spotlight topic? You guessed it: **"GenAI and RE"**.

Over two days, the event delivered a wealth of insights through thought-provoking talks and interactive sessions, bridging research and industry perspectives. **One thing became strikingly clear: this is a field of immense breadth, countless facets, and rapid evolution.**

Staying ahead demands active exchange and continuous learning. That's why we encourage you to participate in events like these. Broaden your knowledge, join the conversation, and contribute your perspective. The field is moving fast, and those who stay engaged are the ones shaping its future.

Keep up with these events by following the **#AIREB** newsletter or subscribing to the IREB newsletter for updates on everything we're up to. Don't let the train leave without you—get involved and be part of the journey!

By Stefan Sturm

Elicitation - Suggest Constraints 1/3

In Requirements Engineering, the process of eliciting requirements is essential for building systems that meet both explicit needs and implicit expectations.

In this, an important role of requirements engineers is not only to gather surface-level information but also to bring hidden requirements – such as constraints and quality expectations – up to the surface in a clear and verifiable way.Chapter 4 of the CPRE Handbook addresses this process, categorizing elicitation techniques into methods for gathering information (like interviews or questionnaires) and methods for generating ideas and designs (such as brainstorming and prototyping). Each of these techniques are used in different situations and help turn vague, initial expectations into structured, actionable requirements and help identifying different types of constraints.

Elicitation - Suggest Constraints 2/3

Identifying constraints is one of the most critical steps in this process.

Constraints often stem from legal, organizational, or technical boundaries that limit what a system can achieve. These constraints can be tricky to identify, even with the proposed techniques of the CPRE, especially in complex projects where some constraints may not be obvious at first glance.

To help with this process, you can use a large language model (LLM) to help brainstorm potential constraints. By asking an LLM to identify constraints based on project information, you can compare its suggestions with your own findings. This can serve as a helpful cross-check to ensure that no important constraints are overlooked.

Elicitation - Suggest Constraints 3/3

When creating your prompt, make use of the concept of assigning an LLM a specific role and clearly articulate your need for constraint identification. You might phrase it like this:

You are an expert in Requirements Engineering and Business Analysis. You are very familiar with the different types of requirements (functional requirements, quality requirements, and constraints). I need you to help me think about possible constraints for my project.
Here is some information about my project: [PROJECT INFORMATION, E.G. PRODUCT VISION BOARD]

By Michael Mey and the IREB AI Interest Group

CPRE IREB

Write User Stories 1/5

User stories are an important part of Agile development because they connect requirements with the solutions created by development teams. Assigning the role of a "User Story Coach" to a large language model (LLM) can help in writing effective user stories that focus on the user's perspective and the value provided.

First, in creating user stories, the stakeholders for whom the story is being written need to be identified. Knowing who the stakeholders are helps to create stories that address their specific needs and motivations. Their goals and desired outcomes should be examined in detail to fully understand their experience and build around them.

During this process, ask clarifying questions to gather the stakeholder's requirements and understand the context in which they work. The aim is to turn this understanding into structured user stories using a clear template. A well-structured user story typically follows this format identified by Mike Cohn: "As a <TYPE OF USER>, I want <SOME GOAL> so that <SOME REASON>." This format puts the user's perspective at the center and clearly communicates their goals and the value of achieving them.

Write User Stories 2/5

User stories should also preferably be divided into smaller, manageable parts, so the team can demonstrate progress at the end of each phase.

Focusing on the stakeholder's needs and making the stories clear helps everyone involved stay aligned in their teamwork, ultimately bringing more successful results.

Using LLMs can help teams create these stories more efficiently. Here is our prompt suggestion:

You are a "User Story Coach" that assists technical business analysts in creating user stories, aligning them with the system architecture, for a broad range of industries.
You start by asking if the user has any system architecture documents or a summary of their system or tech stack. This helps you to contextualize the user story within the existing technical framework. If the user doesn't have these documents, you proceed to ask for a summary of the desired feature. You will ask more clarifying questions to grasp the full scope and specifics.

Write User Stories 3/5

You are a "User Story Coach" that helps in creating clear, user-focused stories that capture value from the user's perspective. You work with teams across various industries to craft stories that emphasize user needs and business benefits.
You start by asking about the user or customer who would benefit from this functionality – understanding who they are, what they're trying to achieve in their daily work or life, and why this matters to them. If these aspects aren't initially clear, you ask clarifying questions to understand the user's perspective, their goals, and the value they seek to gain. This helps you to contextualize the user story within the actual business needs. Then you transform this summary into a structured user story in the specified TEMPLATE. The template should always be marked down as code. This approach ensures that the user stories are comprehensive, clear, and technically aligned with the system's capabilities.

Write User Stories 4/5

Ask the user whether he wants to split the user story into smaller chunks. When you suggest split user stories, always consider vertical splitting, so that at the end of each sprint, the team would be able to demonstrate something. For splitting patterns, consider these approaches: Workflow Steps, Business Rules Variations, Major Effort, Simple / Complex, Variations in Data, Data Entry Methods, Defer System Qualities, Operations (Example CRUD), Use Case Scenarios, Break out a Spike.
[TEMPLATE]

User Story:
[Use the user story template by Mike Cohn (As a <TYPE OF USER> I want <SOME GOAL> so that <SOME REASON>.). Be clear and concise and focus on the added business value of the user story.]
Background/Description:
[Offer a detailed description of the issue, including context, relevant history, and its impact on users or the project.]
Acceptance Criteria:
[List dynamic, clear, and measurable criteria that must be met for this ticket to be considered complete. Criteria should be testable and directly related to the user story.]

CPRE IREB

Write User Stories 5/5

Additional Elements:

- Scenario/Task: [Describe the specific scenario or task in detail.]
- Constraints and Considerations: [Highlight any technical, time, or resource constraints, as well as special considerations relevant to the story.]
- Linkages: [Explain how this story links to larger project goals or roadmaps.]
- Attachments: [Attach any relevant screenshots, diagrams, or documents for reference.]

Prioritization:
[Provide guidance on how this story should be prioritized in relation to other tasks in the backlog.]
Collaboration and Ownership:
[Specify the team members responsible for collaborating on and owning the completion of this story.]
[/TEMPLATE]

By Michael Mey and the IREB AI Interest Group

Validation: Generate Acceptance Criteria 1/2

In Chapter 3 of the CPRE Foundation Level Handbook, we dissect the challenges of extensive specifications, which were also discussed in our previous AI snippet on value orientation. The CPRE Foundation Level handbook not only addresses these challenges but also offers solutions that emphasize adapting documentation to the project context and selecting work products that deliver optimal value.

Today, we turn our attention to the solution of implementing acceptance criteria, which are closely tied to the user stories highlighted in one of our previous posts.

Acceptance criteria clarify user stories by specifying the conditions that must be met for a story to be considered complete. This not only reduces ambiguity but also establishes a shared understanding among stakeholders and provides a solid foundation for evaluating the implementation of user stories. By outlining specific, measurable outcomes, well-defined acceptance criteria enable stakeholders to assess whether their needs have been met before accepting the user story.

Validation: Generate Acceptance Criteria 2/2

Acceptance criteria also serve as a basis for testing. They define the expected behavior of the system, enabling developers and testers to verify that the implementation aligns with the outlined requirements. This again ensures both quality and alignment with stakeholder expectations.
To integrate acceptance criteria into your development process, remember to incorporate them into your prompts when using AI!
Here’s an approach created by our SIG to using LLMs to generate acceptance criteria:
You are an expert Requirements Engineer and Business Analyst, very familiar with the knowledge of the CPRE and BABOK. You also know Mike Cohn's approach to writing better user stories. You are also familiar with the INVEST quality criteria. I want you to optimize my user stories. Review, rephrase, and most importantly, suggest acceptance criteria.

By Michael Mey and the IREB AI Interest Group

Management:
Elaborate DoR 1/2

Requirements management is crucial for ensuring all stakeholders involved in developing work products are aligned from the very beginning. That's why our CPRE Foundation Level handbook dedicates an entire chapter to requirements management. Requirements are like living organisms; they need proper preparation and nurturing before they can begin their journey through the development lifecycle. As described in the handbook, they need careful preparation through elicitation, initial documentation, and validation before they're ready for implementation.

A crucial aspect of this preparation is knowing when a requirement is truly "ready" to be worked on. The ISO/IEC/IEEE 29148:2018 standard emphasizes the importance of proper requirements preparation and readiness assessment as part of the overall requirements management process. Teams need clear criteria for when requirements are sufficiently prepared, making a well-defined Definition of Ready (DoR) essential.

Often, teams begin with minimal preparation criteria, using simple checklists before moving to more sophisticated tools like Jira and Confluence as projects grow more complex.

Management:
Elaborate DoR 2/2

As requirements become more numerous and interconnected, it can become challenging to determine when they are genuinely ready for implementation. A clear DoR helps maintain control and ensures that requirements are properly prepared before entering the development cycle – from initial documentation through stakeholder validation.
A DoR serves as a quality gate that helps prevent half-baked requirements from entering the development process. It ensures that the team has all necessary information, context, and resources to begin work on implementing the requirement. This preparation helps prevent costly delays and rework that often occur when requirements aren't properly prepared before development begins.
Identifying the DoR can be challenging, but large language models can assist you! Try a prompt like:

You are an expert Requirements Engineer, Business Analyst and Agile Coach, very familiar with the knowledge of IREB, BABOK and the Scrum Guide. I want you to give me a general Definition of Ready for my team. Before you start, please ask me relevant questions about my team, like what work they do, on what technology and whatever else is important.

By Michael Mey and the IREB AI Interest Group

CPRE IREB

Elicitation: Draft a Questionnaire/Survey 1/4

In Chapter 4.2.2 of our CPRE Foundation Level Handbook, questionnaires are highlighted as a key questioning technique for eliciting requirements in Requirements Engineering. To explore other techniques, check out the full chapter (Link to Chapter 4.2.2.).

A questionnaire presents a structured set of questions to a large group of stakeholders, either orally, in writing, or via a web form. This approach is especially useful for confirming hypotheses or validating previously elicited requirements with statistical support.

There are two types of questionnaires:

- Quantitative questionnaires use closed-ended questions with predefined answers. They are easy to analyze and ideal for validating requirements across large groups.
- Qualitative questionnaires have open-ended questions which can lead to more nuanced responses and finding new requirements. However, analyzing these requires more time and effort.

CPRE IREB

Elicitation: Draft a Questionnaire/Survey 2/4

Building on this, our Special Interest Group for AI and Requirements Engineering has created an Elicitation Prompt to help large language models (LLMs) assist you in creating questionnaires. This prompt guides the LLM in creating a Requirements Survey Template for any application, here referred to as [App_Name]. The goal is to systematically elicit and validate user needs, preferences, and expectations to inform product requirements.
Here is the prompt:

Requirements Survey Template for [App_Name]
Role: You are a Requirements Engineer helping to create a survey to elicit user requirements for [App_Name].

Before Starting, please Provide:
- [Product Vision Board]
- [Target User Groups]
- [Main Business Goals]

CPRE IREB

Elicitation: Draft a Questionnaire/Survey 3/4

Survey Structure:
- Introduction: "We're developing [App_Name], which will help you [Main Value Proposition]. This survey takes [X] minutes to complete."

Target Group Validation
- Age: [Age_Range_Options]
- Role: [Target_Group_Options]
- Experience with: [Relevant_Experience]

Current Situation
- How do you currently handle [Problem_Domain]?
- What challenges do you face with [Current_Process]?
- How much time do you spend on [Activity]?

Feature Assessment – Rate importance (1-5):
- [Feature_1]
- [Feature_2]
- [Feature_3]
- [Add more features as needed]

Elicitation: Draft a Questionnaire/Survey 4/4

Usage & Value
- How often would you use [App_Name]?
- Would you prefer [Option_A] or [Option_B]?
- What would you be willing to pay for [App_Name]?

Open Feedback
- What other features would you like to see in [App_Name]?
- Any concerns about using [App_Name]?

Quality Requirements:
- Survey Length: [X] questions
- Mix of: Multiple choice, rating scales, open questions
- Language: Simple and clear for [Target_Audience]
- Required responses marked with *

Fill in the [] placeholders based on your specific app and requirements.

By Michael Mey and the IREB AI Interest Group

Documentation: Create UML Activity Diagram 1/2

Note: The output can be visualized with https://mermaidchart.com

In Requirements Engineering, clearly documenting system workflows is essential for understanding and communicating processes. One way to represent these workflows is through UML Activity Diagrams, a standard visual tool used to depict the sequence of activities, decisions, and interactions in a process. UML Activity Diagrams are particularly useful for modeling the flow of control within a system, identifying decision points and parallel processes, and clarifying the sequence of events and dependencies.

These diagrams are valuable for stakeholders to quickly understand system behavior and for engineers to have well-defined process flows.

To help you create a precise UML Activity Diagram for your application [App_Name], our SIG developed the following prompt:

CPRE IREB

Documentation: Create UML Activity Diagram 2/2

Role: You are a Requirements Engineer

Create a UML activity diagram for [App_Name]

Input Required:
- [Paste your process/activity description here]

Technical Notes:
- Use Mermaid activity diagram syntax
- Include clear labels for each activity
- Show decision paths clearly
- Mark critical paths if any

Please generate a Mermaid activity diagram based on these specifications.

As always make sure to validate the result carefully, as LLM outputs for UML sometimes have considerable deficiencies in detail.

By Michael Mey and the IREB AI Interest Group

Validation: Rate Requirements against Quality Criteria 1/3

Assessing the quality of requirements is an important step in developing robust applications. As specified in Chapter 3.8 of our CPRE Foundation Level Handbook, a requirement must meet specific quality criteria to be considered reliable. A value-oriented approach helps prioritize quality criteria based on each requirement's importance, ensuring that the most critical requirements are well-defined to mitigate project risks.

Quality criteria are categorized into two groups:

- Those relevant to individual requirements
- Those that apply to broader RE work products, such as requirements documents or structured documentation frameworks

This distinction helps maintain high standards both in individual requirements and in overall documentation quality, supporting traceability and long-term project integrity.

With this in mind, our SIG developed a structured Quality Requirements Analysis Prompt to help you validate and rate requirements against clear, measurable, value-driven quality criteria tailored to specific applications.

CPRE IREB

Validation: Rate Requirements against Quality Criteria 2/3

Check out the prompt, customize it for your needs, and give it a try:

Quality Requirements Analysis for [App_Name]

Role: You are a Quality Requirements Engineer with expertise in [Domain_Type] applications. You specialize in translating business needs into measurable quality criteria and test scenarios according to industry standards (e.g., ISO 25010).

Project Context:
- Application Type: [App_Type]
- Main Features: [Key_Features]
- Target Users: [User_Group]

Quality Dimensions to Analyze:
- [Quality_1] (e.g. Performance)
- [Quality_2] (e.g. Security)
- [Add_More_If_Needed]

CPRE IREB

Validation: Rate Requirements against Quality Criteria 3/3

Example Output Format:

Performance
- Definition: Response time for user interactions
- Metrics: Max response time < 500ms
- Tests: Measure response time during peak usage

By Michael Mey and the IREB AI Interest Group

CPRE IREB

Requirements Management: Suggest Requirements Prioritization 1/3

The implementation of requirements takes effort, time, money, and attention. Often, these resources are limited, meaning that not all requirements can be implemented immediately. In such cases, you need prioritization. Which requirements should be prioritized, and which can be delayed? This is a key responsibility for Requirements Engineers and shows the importance of Requirements Prioritization.

As outlined in the CPRE Foundation Level handbook, prioritization involves two main steps: first, setting clear goals based on stakeholder needs and project objectives, and second, defining criteria (e.g., business value, cost, risk) to assess and rank requirements.

For a deeper understanding of prioritization, refer to Chapter 6.8 of the CPRE handbook.

Prioritization has many facets, and without a clear approach, it’s easy to overlook important factors. To help with the process, try using LLMs to assist with prioritization. Check out our SIG's prompt designed to help you get started:

CPRE IREB

Requirements Management: Suggest Requirements Prioritization 2/3

Role: You are a Requirements Prioritization Expert and Product Owner with extensive experience in agile methodologies. You specialize in balancing business value against technical constraints and helping teams make data-driven decisions about requirement implementation order.

I have a list of requirements for [Project] and would like to prioritize them based on business value, implementation effort, and urgency. The requirements vary and include functional requirements, quality requirements, and some regulatory constraints. Please help prioritize the requirements using the following criteria:

- Business Value: Are there measurable benefits for the end-user or company?
- Technical Effort: How complex and demanding is the technical implementation?
- Urgency: Are there any requirements that need to be implemented quickly due to regulatory or strategic reasons?

Requirements Management: Suggest Requirements Prioritization 3/3

Generate a prioritization matrix assigning each requirement a score based on the above criteria. Use weighted scores (e.g., Business Value 40%, Technical Effort 30%, Urgency 30%). Based on this analysis, please create a prioritized list of requirements.

Additionally, generate a visualization (e.g., bubble chart) that shows business value versus technical effort. Highlight requirements with the highest value and lowest effort, as they may be high-priority candidates.

List of requirements:
- [Req 1]
- [Req 2]
- [Add_More_If_Needed]

Project Description:
- [Paste your project description here]

By Michael Mey and the IREB AI Interest Group

CPRE IREB

Principle 4 – Context 1/2

Requirements always exist within context, involving system and context boundaries.

LLMs can assist in managing this context, here are some examples how:

Example 1 – Context dependencies:
LLMs can analyze requirements, considering provided contextual information such as project scope and known interfaces, to identify contextual dependencies. This helps clarify how external systems, environments, or components interact with the system.

Example prompt: "You are a requirements engineer expert. Based on the following system requirements, identify any context dependencies or external interfaces that should be considered: [Insert requirements here]"

Principle 4 – Context 2/2

Example 2 – System requirements:
LLMs can map real-world domain requirements (e.g. GDPR) to specific system requirements, bridging the gap between abstract requirements and technical specifications.

Example prompt: "You are a requirements engineer expert. Based on the following domain requirements, generate corresponding system requirements and highlight any critical assumptions: [Insert domain requirements here]"
Using AI tools such as GPT LLMs can improve the management of system context and boundaries. This ensures a clear understanding of requirements.

By Franz Zehentner, Michael Tesar and the IREB AI Interest Group

Principle 5 – Problem – Requirement – Solution

Problems, requirements and solutions are described as closely related aspects of system development. A problem occurs when people are dissatisfied with a current situation and a system can be designed to solve it. Requirements are captured to ensure that the system effectively addresses the problem and satisfies the needs of the customer/stakeholder.

Large language models (LLMs) can assist in separating concerns by categorizing existing datasets – such as requirements, meeting notes, and documentation – into distinct categories: problems, requirements, and solutions. This categorization enables Requirements Engineers to clearly identify which elements can be pursued further as requirements.

A prompt you could use might look like:
"You are an expert requirement engineer. Given the following dataset containing requirements, meeting notes, and documentation, extract a complete list of all information and categorize each item as a problem, requirement, or solution: [Insert dataset here]"

By Franz Zehentner and the IREB AI Interest Group

CPRE IREB

Principle 6 – Validation 1/2

Non validated requirements are useless as you cannot assume the stakeholders' needs are met. Therefore, validating the specification before development can make sure no unnecessary development resources are wasted.

Before starting intensive validation workshops, one could utilize the power of large language models (LLMs) and have the already written requirements analyzed and possible ambiguities, conflicts or incomplete specification identified.

This prompt can be a good starting point:
You are an expert requirements validation assistant specializing in analyzing stakeholder requirements for potential defects such as ambiguities, inconsistencies, and omissions. Your task is to categorize provided information into distinct categories: validated requirements, problematic requirements, or incomplete/ambiguous requirements. Provide actionable feedback for improvement where necessary.

Principle 6 – Validation 2/2

Add your requirements and check the feedback the LLM will give you. Engage in conversation if the first responses do not provide meaningful results. Correcting the LLM and stating your expectations can enhance the quality of the output.

By Franz Zehentner and the IREB AI Interest Group

Principle 7 – Evolution 1/2

Change is not the exception but the rule. As businesses evolve, technologies advance, and stakeholder priorities shift, systems and therefore also their requirements must adapt accordingly. These ongoing adjustments can be caused by various triggers: business decisions, customer feedback, evolving domain knowledge, or even a competitor's recent market innovation. Identifying which existing requirements are impacted by such changes is a critical step that ensures your product remains relevant and valuable over time.

Large Language Models (LLMs) can significantly streamline this process. By interpreting and analyzing textual requirements, these models can highlight dependencies, and pinpoint which requirements may need to be revisited.

The following prompt can help you in identifying requirements that are affected by a competitors product launch and your goal is to adapt your system to be developed accordingly:

CPRE IREB

Principle 7 – Evolution 2/2

"You are an expert requirement engineer assisting in analyzing changing requirements. Below is a description of a competitor’s product and our current set of requirements. Based on this information, identify which of our requirements are impacted by the competitor’s innovations. List the requirements with reference to the competitor's product feature.
Suggest any follow-up questions or considerations that might help clarify the scope and depth of the required changes.

Competitor’s Product Description:
[Insert details of the competitor’s product here or provide readable files.]

Our Current Requirements:
[Insert your existing requirements here or provide readable files.]"

This approach can be especially valuable if you have a large set of existing requirements and need to quickly identify starting points for your impact analysis. Even though the LLM might not catch 100% of affected requirements, the output will provide you with a flying start in your requirement analysis.

By Franz Zehentner and the IREB AI Interest Group

CPRE IREB

Principle 8 – Innovation: More of the same is not enough 1/2

In today’s landscape, refining existing solutions is often insufficient for a successful product on the market.
Large Language Models (LLMs) can support innovation by generating creative and alternative solution concepts or lay out the path on how to find new solutions. Recently some of the LLM tools got access to the internet and more current information. This enables access to information that the user might not have found or thought of as relevant and can be used by the LLM to formulate an even better response.

Example 1 – Ideation workshop:
LLMs can help you moderate and create the outline of workshops that lead to new solutions. In this case the LLM does not necessarily depend on domain knowledge since has access to various ideation techniques and can help you get your team to think outside the box.

Example prompt: "You are organizing an ideation workshop with the goal of generating innovative solutions. Given the following problem statement, provide five creative ideas that workshop participants could use as inspiration. List the potential advantages and disadvantages of each idea: [Insert problem statement here]"

Principle 8 – Innovation: More of the same is not enough 2/2

Example 2 – Cross-domain innovation:
LLMs can leverage cross-domain knowledge to suggest solutions inspired by advancements or practices in other industries, encouraging interdisciplinary innovation.

Example prompt:
"You are an innovation expert with broad cross-industry knowledge. Based on the following set of requirements, propose innovative solutions inspired by best practices from other industries. Clearly indicate the source and explain how the solution can be adapted to fit the described requirements:
[Insert system requirements here]"

But be careful: LLM are usually trained with 'old' data. Real innovation comes from supporting the creativity and innovation processes, not from replacing innovation and creativity with LLM. This specific example targets analogies with already existing methods, not inventing new methods overall.

By Franz Zehentner and the IREB AI Interest Group

CPRE IREB

Principle 9 - Systematic and Disciplined Work: We Can't Do Without It 1/2

Even in agile or flexible contexts, Requirements Engineering requires a systematic approach and flexibility at the same time. Large Language Models (LLMs) support this challenge by assisting with planning, ensuring adherence to standards and structures, and selecting appropriate steps. Requirements Engineers should be ready for all kinds of situations during a project's lifetime and have appropriate tools and methods ready.

Example 1 – Process auditing:
LLMs can help you detect potential gaps or deviations from established RE practices in your company. By analyzing project artifacts (like user stories, requirements specs, or sprint backlogs), they can flag missing elements, unstructured requirements, or steps in your pipeline that deviate from the companies defined process. Provide the LLM with your RE process description and/or guidelines beforehand and use the following example prompts to check conformity.

Example prompt: "You are a Requirements Engineering assistant. Based on the following project artifacts and company standards, identify any deviations or process gaps and suggest corrective steps: [Insert relevant documents or references here]"

Principle 9 - Systematic and Disciplined Work: We Can't Do Without It 2/2

Example 2 – Template enforcement and consistency checking: LLMs can parse documents to check if your team is using and following templates or style guides systematically. This ensures consistency across different requirements and reduces confusion. It also helps maintain the disciplined approach needed for stable project outcomes.

Example prompt: "You are an expert on documentation standards. Given these requirement templates and our newly written specification, verify that all relevant fields are present and consistent. Flag any inconsistencies and provide recommendations for aligning the specification with the template: [Insert templates and spec here]"

By making use of LLM-assisted methods, organizations can strengthen their RE processes, reduce errors, and maintain the discipline that is essential for successful and predictable outcomes. In addition, these results can help Requirements Engineers derive how to improve their systematic and disciplined requirements process. Additional input for this is provided by the CPRE Advanced Level.

By Franz Zehentner and the IREB AI Interest Group

CPRE IREB

Use consistent terminology 1/2

Consistent terminology is the foundation of common understanding among stakeholders in any domain. In requirements engineering, shared vocabulary ensures that all parties interpret requirements in the same way. Without consistent terminology, different interpretations of the same requirement are possible and can cause misunderstandings.

AI can be a powerful ally in identifying and resolving terminology inconsistencies that might not be detected. Large Language Models (LLMs) are very good at analyzing text and can quickly spot where the same concept is referred to by different terms. A related use case is where LLMs analyze given requirements and can create a glossary to be used as a reference for domain specific terms.

Example prompt for identifying inconsistent terminology:
"Analyze the requirements that follow and identify every case where the same underlying concept is referred to with different words, abbreviations, or spellings. Provide your results in a table with following columns:
Variants Found – every alternative spelling, acronym, or phrase that maps to the same concept.

Use consistent terminology 2/2

Concept identified – brief definition of concept
Here are the requirements: [Insert your requirements]"
If there are inconsistencies identified, they can be resolved and in the next step a glossary for further information or explanation could be created utilizing the LLMs capabilities of dealing with text.

Example prompt for creating a glossary:
"Based on the attached requirements document, create a glossary of key terms and concepts. Include any technical terms, domain-specific vocabulary, and stakeholder roles. For each term, provide a clear, concise definition: [Insert your requirements]"

Establishing and maintaining consistent terminology creates a solid foundation for communication and quality in requirements engineering. LLMs can help you to identify inconsistencies and build comprehensive glossaries, but the discipline using these defined terms is still up to the project team.

By Andrea Herrmann, Franz Zehentner and the IREB AI Interest Group

Prompt Strategy Series
Part 1 – RTF 1/2

Role-Task-Format (RTF) is a lightweight prompting strategy that can be more effective than complex techniques, if you need a quick assessment of less complex tasks. You need to define the three parts of this approach as followed to get the desired output quality:

- Role: Define the role, expertise and focus
- Task: Clearly specify what needs to be done
- Format: Describe the expected output structure

RTF works best when you need quick, structured analysis without excessive context or additional information. It is particularly effective for routine RE tasks like reviews, classifications, or assessments where clarity and consistency matter more than creativity.

If you need requirements (or in this case a user story) reviewed, instead of a lengthy prompt with multiple instructions, use this template:
"Role: You are a senior requirements engineer with 15+ years of experience in requirements quality assurance.
Task: Review the following user story for compliance against INVEST criteria (Independent, Negotiable, Valuable, Estimable, Small, Testable). Identify which story fail to meet these criteria and explain why.

CPRE IREB

Prompt Strategy Series
Part 1 - RTF 2/2

Format: For each problematic story, provide in a tabular view:

- Story ID and title
- Failed INVEST criteria
- Specific issues identified
- Recommended improvements

User Story: [Insert your story here]"
The RTF approach produces focused and consistently formatted feedback for quick tasks.

By Franz Zehentner and the IREB AI Interest Group

Disclaimer: The IREB AI Interest Group did not develop this framework. It was established by AI users and the community.

Prompt Strategy Series
Part 2 – CRISPE 1/4

When facing more complex challenges that need detailed analysis and creative solving methods, the CRISPE framework (Context–Role–Instructions–Steps–Parameters–Example) provides a prompting approach that can help in tackling the task. The prompt is structured in 6 parts:

- Context: Background information, project setting
- Role: Define the AI's expertise, capabilities and perspective in the task to be solved
- Instructions: Clear directive of what needs to be accomplished
- Steps: Break down the process into individual steps to be carried out
- Parameters: Specify constraints, quality criteria, and boundaries
- Example: Provide an example of the desired output format

Use CRISPE for complex tasks like stakeholder analysis, requirements traceability mapping, or system modeling where context matters significantly and the process involves multiple steps that depend on each other.

CPRE IREB

Prompt Strategy Series
Part 2 - CRISPE 2/4

When to use CRISPE:

- Complex analysis requiring multiple perspectives and considerations
- Tasks where domain context significantly impacts the outcome
- Multi-step processes with interdependent phases
- Creative problem-solving that benefits from structured guidance
- First-time analysis of unfamiliar domains or systems

When NOT to use CRISPE:

- Simple, routine tasks (use RTF instead)
- Quick reviews or validations
- Straightforward classifications

In this example, we are conducting a stakeholder analysis in a complex setting. The individual contents of the prompt need to be replaced to your situation:

"Context: We're developing a healthcare management system for a mid-sized hospital with 200 beds, serving both inpatients and outpatients. The system will integrate with existing EHR systems and must comply with HIPAA regulations.

CPRE IREB

Prompt Strategy Series
Part 2 - CRISPE 3/4

Role: You are a senior business analyst specializing in healthcare IT with expertise in stakeholder identification and analysis.
Instructions: Conduct a comprehensive stakeholder analysis to identify all relevant parties and their interests in this healthcare management system project.

Steps:

- Identify primary, secondary, and key stakeholders
- Analyze each stakeholder's interests and influence level
- Assess potential conflicts between stakeholder groups
- Recommend engagement strategies for each stakeholder category
-

Parameters:

- Consider regulatory compliance requirements
- Account for both technical and clinical stakeholders
- Include external parties (vendors, patients, regulators)
- Focus on stakeholders who can impact or are impacted by system decisions

Prompt Strategy Series
Part 2 – CRISPE 4/4

Example: For each stakeholder, provide: Stakeholder: Chief Medical Officer Category: Primary stakeholder Interest: Ensure system supports clinical workflows and patient safety Influence: High - can approve/reject system decisions Engagement Strategy: Regular consultation meetings, early prototype reviews"

CRISPE's structure ensures a thorough analysis for complex requirements engineering challenges where context and systematic approach are crucial.

By Franz Zehentner and the IREB AI Interest Group

Disclaimer: The IREB AI Interest Group did not develop this framework. It was established by AI users and the community.

Prompt Strategy Series
Part 3 – RISEN 1/3

When you need a structured and streamlined prompt for more complex requirements engineering tasks, the RISEN framework (Role – Instructions – Steps – End goal – Narrowing) helps you get precise, actionable outputs without overwhelming detail.

R – Role
Define the AI's persona, expertise and knowledge level.

I – Instructions
State exactly what you want done. Be as precise as possible.

S – Steps
Outline the process in clear, sequential steps. The process description determines how the AI will tackle the problem and process each step you define.

E – End goal
Describe the desired final deliverable. This part of the prompt defines the format the output and what will be generated.

N – Narrowing
Constrain scope to keep the response focused. If you want to limit your output to e.g. 10 examples or do not want to include a specific field, this is the place to define it.

Prompt Strategy Series
Part 3 – RISEN 2/3

Example Prompt:

R – Role:

You are a seasoned requirements engineer specializing in agile user-story workshops.

I – Instructions:

Draft interview questions to elicit non-functional requirements for a banking portal.

S – Steps:

- Identify key quality attributes (e.g., performance, security, usability).
- Translate each attribute into open-ended questions.
- Group questions by stakeholder type.
- Prioritize questions based on risk impact.

E – End goal:

A prioritized list of 10–12 interview questions, ready for stakeholder review.

N – Narrowing:

- Domain: Online banking portal
- Stakeholders: Security Officer, UX Designer, Operations Manager
- Maximum questions: 12

Prompt Strategy Series
Part 3 – RISEN 3/3

When to use RISEN

- Generating structured artifacts (user stories, test scenarios, interview guides)
- Mid-complexity tasks with well-defined domains
- Agile ceremonies where speed and clarity matter

How RISEN differs from CRISPE

- Lean vs. comprehensive: RISEN skips deep context and parameter sections, favoring faster setup
- No examples: Relies on clear steps and narrowing rather than sample outputs
- Ideal for agile RE: CRISPE shines in heavyweight, multi-phase analyses; use RISEN when you already know the domain and just need a tight, repeatable prompt

By Franz Zehentner and the IREB AI Interest Group

Disclaimer: The IREB AI Interest Group did not develop this framework. It was established by AI users and the community.

CPRE IREB

Contact

info@ireb.org

www.ireb.org

CPRE IREB