**Module 10.1 Discussion Post**

**Julio Pochet – The Requirements Pit**

I chose **Topic 45: The Requirements Pit,** because I’ve experienced how confusing it can be when a project doesn’t have clear expectations. Whether it’s coding, work, or personal projects, **not knowing exactly what’s needed leads to wasted time, frustration, and constant rework**. Since we’re modifying an existing **Tkinter To-Do List GUI**, understanding what needs to be added or changed is important. If we don’t clarify the requirements first, we might spend time coding useless or unnecessary features.

**Main Points of the Topic:**

This topic focuses on one of the biggest challenges in software development—**unclear, incomplete, or constantly changing requirements**. Without asking questions early, developers risk creating something that **doesn’t meet user needs**. The best way to avoid this is by **clarifying expectations, seeking feedback, and building in small, manageable steps** rather than trying to complete everything simultaneously. Another key takeaway is that **projects should be flexible**—unexpected changes happen, and the ability to adjust is just as important as having a plan.

**How This Applies to My Tkinter To-Do List Project:**

* Making sure I **fully understand what’s required** before modifying the code.
* **Building one feature at a time** and testing it instead of trying to finish the whole project in one go.
* Keeping the design **simple and adaptable** so changes can be made if needed.

**Additional Resource:**  
📌 **"How Poor Requirements Lead to Catastrophic Software Failures"** – [Medium](https://medium.com/@sogekebosun/how-poor-requirements-lead-to-catastrophic-software-failures-aab25399f398)