Working for Julian

A Guide to Understand How I Think and Manage Teams

Leadership Principles

- **Serving Others**. I lead by putting the welfare of the team first. I have a service-oriented mindset to ensure the well-being (and, thus, effectiveness) of my reports, which leads to higher morale and better performance. Effective leaders make sure the work is done, yet they also prioritize empathy, understanding the needs, challenges, and perspectives of their team members. This allows leaders to build stronger relationships and inspire loyalty.

- **Coach mode**. While I put a big value on being a Servant Leader, sometimes managed reports need a different approach at crunch-time. I help reports create a plan of attack on a problem or task, to meet a challenge. This also means demonstrating values such as integrity, resilience, and hard work, which sets a standard for others. I call this "coach mode" because I am leading on the field (working tasks) with the field, and resolving performance issues that I see as-they-happen. Team leaders who can pivot to "coach mode" foster a culture where people feel secure, valued, and motivated, leading to a more cohesive high-performing team.

- **Collaboration over competition**. I want a collaborative environment, rather than one driven by internal competition, as it helps strengthen bonds among team members. A unified team is more resilient and effective in facing external challenges. I also tend to respond to any discovered office politics situations like an immune system,

- **The WHY is key** (Vision and Purpose). People are often more inspired by the reasons behind actions than by the actions themselves. Leaders who communicate a clear *Why* build deeper connections, trust, and loyalty among employees and clients or customers. Leaders who prioritize the *Why* are more likely to motivate and create a following that's driven by shared purpose and values. Leaders need to inspire a sense of purpose to align the team's efforts toward a common goal.

Management Style Preferences

- **Empowering Independence.** Over two decades in IT informs my management philosophy to emphasize self-reliance, initiative, and continuous learning from my reports. I teach my reports that ***taking initiative*** when starting a new task, and not needing a lot of immediate task guidance, increases their value on the team. Try to solve as much of a problem independently as you can, before reaching out for help. Remember getting "stuck" is *good*, so long as you worked the problem first, a necessary step toward self-reliance. Asking questions *before* you work a problem robs you of growth and experience. When you come to me for help, know that I will *always* ask what you've tried up to that point. If you get stuck early and often on tasks, know that means we'll be having a one-on-one to discuss your blind spots and skill gaps to determine where you are struggling.

- **Setting Clear Expectations and Providing Support.** We're all adults and professionals here, so I trust you to own your project tasks and escalations. I will set clear expectations and provide you all necessary tools and support for you to succeed. I do expect task work to get underway without requiring deep assistance at the *start* of task work, but I am here to help when you are stuck. If you require a lot of guidance to begin work on tasks, a one-on-one meeting may occur to assess your skill blind spots. I want a team culture where we all are eager to help each other, but we also trust each other to work our assigned tasks until stuck. And if training needs are found, I want to get you whatever you need to succeed.

- **System Documentation should provide the WHERE and the WHAT, *<u>not</u>* the HOW.** Good system documentation always provides the **WHERE** and **WHAT**— server information, IP addresses, and other essential information; ***<u>however</u>***, it is ineffective and poor strategy for documentation to spend pages on the **WHY** or the **HOW**. It can be a time sink, reflecting skill gaps. Process documentation is helpful, but also skilled and experienced team members usually only need the WHERE, in order to begin work on something. Skilled and experienced staff should need the **WHERE** in order to begin work on something. Anything needed for the **HOW** is ***process documentation*** and, while helpful for new team members, it should be concise and outcome oriented. Documentation should almost always be about Systems, Infra, and assets is what the team needs moves, for ongoing Ops and Project deliverables (the HOW and WHAT). Good Documentation already is a time-consuming activity, so confusing process for systems documentation only leads to excess documentation iteration cycles and, ultimately, ineffective cross-training.

- **Process Documentation should be concise and *extremely* brief.** Process documentation should only come *after* system documentation, and is never a replacement for developing skills and experience within the team. Process documentation should *not* contain endless verbiage or paragraph-sized steps for any process. If a team is busy writing process documentation that contains myriad steps explaining a lot of ***HOW*** information, it usually reflects a lot of skill-gap among team members, or incorrectly assigned IT roles. Unless your team or department has a full-time trainer or documentation role in the group, process docs should be very concise and brief.