



Riños, Christopher Jan

Software Engineer

Manager: Arjay Gallentes

Evaluated By: Arjay Gallentes

July Impact Statement

Organization: Axos Business Center Team VII (Arjay Gallentes)

Location: ABC Manila Office

01/01/2025 - 06/30/2025

Acknowledgement

Manager

Entered by: Arjay Gallentes Date: 08/27/2025
Status: Acknowledge Review
Comment:

Employee

Entered by: Christopher Jan Riños Date: 08/27/2025
Status: Acknowledge Review
Comment:

Questions

What do you do? How do you describe your role, duties, and responsibilities? Please be specific.

Manager Evaluation

Response:

Employee Evaluation

Response: I'm a Software Engineer currently working on the Data team. My role involves refining and maintaining our data models to ensure they align with both technical requirements and business needs. Specifically, my key responsibilities are the following:

- **Refine the data model** based on the current setup in ALF, making sure it remains scalable and efficient.
- **Support the OS development team** by assisting with data-related issues, ensuring smooth integration and consistency across systems.
- **Create Entity-Relationship Diagrams (ERDs)** to clearly present the refined data model, helping both technical and non-technical stakeholders understand the structure and relationships within our data.
- **Handle data conversion and transformation**, ensuring that data is accurately migrated and formatted to fit new models or systems.

- **Identify and implement opportunities to automate repetitive tasks** and streamline workflows, making our processes faster and more efficient.
- **Create and maintain documentation**, including how-to guides and reference materials, to support team members and stakeholders.

What processes do you perform? Please refer to the specific process maps or procedures for the tasks you perform, manage, or have impacted. If no process map exists, please describe the task and write "no process map or procedure".

Manager Evaluation

Response:

Employee Evaluation

Response: In my role as a Software Engineer, I'm involved in several key processes that support data modeling, automation, and team collaboration. I also take an active role in improving and maintaining the tools and workflows we rely on. Below are the main processes I perform or have directly impacted, along with references to the relevant procedures or documentation:

1. Data Modeling and Database Design

Process Description: Designing and maintaining data models based on business requirements and legacy documentation.

Key Activities:

- Analyzing legacy sheets and translating them into relational models.
- Creating ERDs and aligning them with naming conventions and table classifications (Main, Multivalued, Dimension, Reference).

2. SQL Script Generation and Automation

Process Description: Automating the generation of SQL scripts to reduce manual effort and improve consistency.

Key Activities:

- Developed a tool that converts legacy Excel sheets into SQL scripts.
- Ensures table relationships and constraints are accurately reflected.
- Deploy the SQL scripts to the appropriate DB environment.

3. Environment Schema Comparison

Process Description: Ensuring consistency of database schemas across development, staging, and production environments.

Key Activities:

- Created a script to compare SQL files and highlight differences across the different environments.
- Used for validating deployments and identifying missing changes.

4. Team Collaboration and Knowledge Management

Process Description: Facilitating team communication and documentation of technical discussions.

Key Activities:

- Initiated daily syncs for the Internal Data Model Team.
- Created and maintained a Loop workspace for meeting notes and decisions.

5. Process Improvement and Tooling Suggestions

Process Description: Identifying inefficiencies and proposing solutions to streamline workflows.

Key Activities:

- Suggested improvements in ERD readability and standardization.
- Proposed enhancements to automation tools and documentation practices.

NO PROCESS MAP OR PROCEDURE

What metrics do you use to manage your performance? How do you know you are doing a good job? Please be specific (number of calls per day, number of closed loans per month).

Manager Evaluation

Response:

Employee Evaluation

Response: To manage my performance and ensure I'm contributing effectively to the team, I rely on a combination of **quantitative metrics** and **qualitative feedback**. Here are the ways I track my performance:

Closed PBIs in Azure DevOps (ADO):

- One of my main metrics is the number of PBIs related to **data model refinement** that I've successfully closed. This includes tasks such as schema updates, ERD revisions, and implementation of new data structures.

Cycle Time and Lead Time:

- I monitor how long it takes to complete a task from the moment it's picked up to when it's closed. Shorter cycle times, especially for complex data-related tasks, indicate efficiency and focus.

Quality of Deliverables:

- I track the number of revisions or rework requests received after submitting a data model or ERD. Fewer revisions typically reflect a higher quality of work and better alignment with requirements.

How do you report progress to your supervisor? (What reports, meetings, etc?)

Manager Evaluation

Response:

Employee Evaluation

Response: I report my progress through a combination of **daily meetings**, **task tracking**, and **open communication channels**:

Daily Internal Data Team Alignment:

- We hold a daily sync within the data team where we ask questions, share ideas, and discuss blockers. It's a collaborative space that helps us stay aligned and support each other.

Daily AUC Project Stand-Up:

- I also participate in a daily stand-up meeting for the AUC project. I share what I worked on the previous day, what I plan to work on today, and raise any issues that need attention. This keeps the broader team informed and coordinated.

Open Group Chat in Microsoft Teams:

- We maintain an open and active group chat where we can voice out questions, share quick updates, and collaborate informally. It's a helpful space for real-time communication and quick problem-solving.

Task Updates in Azure DevOps:

- I regularly update the status of my tasks in Azure DevOps. This provides visibility into my progress and allows my supervisor and team members to track deliverables, timelines, and dependencies.

Which processes / routine tasks have you improved / suggested improvement? What are the specific improvements suggested or made? a. Please refer to specific process maps and procedures for the processes you perform, manage, or have impacted, or if you have created a new process, please describe it. b. If suggested, please outline action steps to implement, include dependencies.

Manager Evaluation	Employee Evaluation
Response:	<p>Over the past few months, I've been actively looking for ways to make our workflows smoother, faster, and more collaborative. Here are some of the key changes I've contributed to:</p> <p>1. Automating SQL Script Generation</p> <ul style="list-style-type: none">What I noticed: Manually writing SQL scripts from the Legacy sheet was not only repetitive and time-consuming, but also impractical at scale—especially since Copilot couldn't efficiently handle the generation of a large number of complex table definitions.What I did: I built an application that takes the Legacy sheet as input and automatically generates the SQL needed to create tables and define relationships.Why it matters: This has significantly reduced manual work, improved accuracy, and sped up the process of setting up database structures. <p>2. Simplifying DB Schema Comparisons</p> <ul style="list-style-type: none">What I noticed: Comparing database schemas across environments was a manual and error-prone task.What I did: I created a script that compares SQL files and highlights the differences, making it easier to spot which changes have or haven't been deployed.Why it matters: It's made our deployment process more reliable and helped us catch inconsistencies early. <p>3. Improving ERD Readability</p> <ul style="list-style-type: none">What I noticed: Our ERDs weren't always easy to interpret at a glance.What I did: I suggested a visual alignment standard to clearly distinguish between Main, Multivalued, Dimension, and Reference tables.

- **Why it matters:** This small change has made our diagrams more intuitive and easier to navigate, especially for new team members or stakeholders.

What other improvements have you suggested (sales, product, group interactions, etc.)? Please be specific.

Manager Evaluation

Response:

Employee Evaluation

Response: I've also been thinking about how we can enhance our team dynamics and make our group interactions more inclusive and engaging. Here's one suggestion I'd like to propose:

Encouraging Broader Participation During Retrospectives

- **What I noticed:** During retrospectives, only a few team members tend to speak up, while others remain quiet. With a large team, this limits the variety of perspectives we hear and can make some members feel less involved.
- **What I suggest:** Assign one representative from each team to share their 4Ls (Liked, Learned, Lacked, Longed for) during each retrospective. This role would rotate every sprint to ensure everyone gets a chance to contribute.
- **Why it matters:** This approach gives everyone a structured opportunity to be visible, encourages more balanced participation, and helps surface a wider range of feedback. It also fosters a more inclusive and supportive team culture.

What improvement opportunities exist to make your role more efficient? What ideas do you have to reduce waste or inefficient work? What are the required action steps?

Manager Evaluation

Response:

Employee Evaluation

Response: In any role, finding ways to work more efficiently and reduce wasted effort is essential for both personal and team success. By identifying improvement opportunities and taking practical steps to address them, we can streamline our workflows, enhance productivity, and create a more positive work environment. Below are some key areas for improvement, along with actionable steps to help achieve these

goals.

1. Automate Repetitive Tasks

- Identify tasks performed frequently and manually (e.g., data entry, data modeling, ERD creation, SQL script creation and validation).
- Evaluate available automation tools (e.g., scripts, applications, etc.).
- Develop or adopt automation solutions for these tasks.
- Test automated processes to ensure accuracy.
- Train team members on using automation tools.
- Monitor and refine automation for continuous improvement.

2. Standardize Processes

- Review current workflows for inconsistencies or variations.
- Create standardized templates, checklists, or documentation for common tasks.
- Communicate standards to all team members.
- Integrate standards into onboarding and training materials.
- Periodically review and update standards as needed.

3. Continuous Training

- Assess current skill levels and identify knowledge gaps.
- Research and provide access to relevant training resources (e.g., online courses, workshops).
- Schedule regular training sessions or 1-on-1.
- Encourage knowledge sharing among team members.
- Track training progress and update learning paths as technology evolves.

What improvements have you made or suggested?: How did you positively influence the attitude of your colleagues? Please be specific.

Manager Evaluation

Response:

Employee Evaluation

Response: In my role, I've focused not only on improving technical processes but also on fostering a more collaborative and solution-oriented team culture. Here are a few

improvements I've made or suggested, along with how they positively influenced my colleagues:

1. Strengthening Team Communication

- **What I noticed:** Our team discussions were scattered and sometimes lacked continuity.
- **What I did:** I suggested we hold a daily internal sync for the Data Model Team to keep everyone aligned. To support this, I also set up a Loop workspace where we can document our discussions and decisions.
- **Why it matters:** It's helped us stay on the same page, track progress more easily, and ensure that no important detail slips through the cracks.

2. Improving Documentation for Custom Tools and Scripts

- **What I noticed:** Team members often had to ask for help or clarification when using the scripts and applications I developed, which slowed down adoption and created unnecessary dependencies.
- **What I did:** I created clear, easy-to-follow guides for each tool and script I built. These guides include usage instructions, input/output formats, and troubleshooting tips.
- **Why it matters:** This empowered others to use the tools independently, reduced repetitive questions, and improved the overall efficiency and confidence of the team when working with custom solutions.

3. Documenting Key Processes (e.g., ADO Pipeline Deployment)

- **What I noticed:** There was no centralized, step-by-step guide for deploying database changes through the ADO pipeline, which led to confusion and inconsistent execution.
- **What I did:** I wrote a detailed how-to guide that walks through the entire deployment process, including prerequisites, pipeline configuration, and validation steps.

- **Why it matters:** This helped standardize the deployment process, reduced errors, and made it easier for both new and existing team members to follow best practices with confidence.

4. Organizing Internal Documentation

- **What I noticed:** Our internal documentation was scattered across different platforms and folders, making it hard to find relevant information quickly.
- **What I did:** I took the initiative to organize and centralize our documentation, grouping related content and ensuring everything was easy to navigate and up to date.
- **Why it matters:** This improved knowledge sharing, reduced time spent searching for information, and encouraged a more structured and collaborative documentation culture within the team.

What are your goals? What would you like to accomplish in the next six months? Please make your goals "SMART" - specific, measurable, attainable, relevant, and time-sensitive.

Manager Evaluation

Response:

Employee Evaluation

Response:

1. Expand Technical Skillset

- **Specific:** Enroll in and complete online courses to improve skills in Golang, cloud services, and databases.
- **Measurable:** Complete at least **2 Udemy courses** (one on Golang, one on cloud services or databases).
- **Attainable:** Courses are self-paced and accessible.
- **Relevant:** Enhances technical capabilities directly related to current and future projects.
- **Time-sensitive:** Complete both courses by **December 2025**.

2. Strengthen Cross-Team Collaboration

- **Specific:** Actively participate in cross-functional meetings and support the OS team.
- **Measurable:** Attend at least **90% of scrum meetings** and contribute to **at least 3 OS-related tasks or discussions**.
- **Attainable:** Participation is already part of the current workflow.

- **Relevant:** Builds stronger relationships and improves project alignment.
- **Time-sensitive:** Track contributions through **December 2025**.

3. Contribute to Improve Documentation Practices

- **Specific:** Help improve internal documentation for data models and processes.
- **Measurable:** Create or update **at least 3 documentation entries** (e.g., ERDs, process guides).
- **Attainable:** Can be done alongside regular tasks.
- **Relevant:** Supports team efficiency and knowledge sharing.
- **Time-sensitive:** Complete updates by **November 2025**.

4. Contribute to Automate a Repetitive Task

- **Specific:** Identify and automate a repetitive task related to data modeling.
- **Measurable:** Deliver **one working automation script or tool**.
- **Attainable:** Leverages existing skills and tools.
- **Relevant:** Improves productivity and reduces manual effort.
- **Time-sensitive:** Complete automation by **October 2025**.

What are your goals? What would you like to accomplish in the next 2 years? Goals over the longer term can be more aspirational.

Manager Evaluation

Response:

Employee Evaluation

Response:

1. Become a Senior Software Engineer

- **Goal:**
Reach a senior-level engineering role by consistently delivering high-quality work, demonstrating ownership, and contributing to team and project success.
- **Reality:**
I have a solid technical foundation and contribute effectively, but I need more experience in leadership, architectural decision-making, and strategic thinking.

- **Options:**
 - Lead small features or initiatives within the team.
 - Take initiative in solving complex problems or improving processes.
 - Attend leadership training, tech talks, or mentorship sessions.
- **Way Forward:**
 - Track my contributions and impact regularly.
 - Seek feedback from senior engineers and managers.
 - Set quarterly growth checkpoints to assess progress toward senior-level competencies.

2. Lead a Team or Project

- **Goal:**
Take ownership of a project or lead a small team, managing timelines, responsibilities, and outcomes effectively.
- **Reality:**
I've contributed to multiple projects but haven't yet had the opportunity to lead one end-to-end.
- **Options:**
 - Volunteer to lead small features or internal initiatives.
 - Support project planning and stakeholder communication.
 - Learn from current project leads and understand their workflows.
- **Way Forward:**
 - Communicate my interest in leadership to my manager.
 - Build trust by consistently delivering high-quality work.
 - Ask for mentorship or shadowing opportunities with current leads.

3. Mentor Junior Software Engineers

- **Goal:**
Support and mentor junior engineers by sharing knowledge, offering guidance, and helping them grow.
- **Reality:**
I have valuable experience and insights but haven't formally mentored anyone yet.

- **Options:**
 - Offer help during onboarding or code reviews.
 - Initiate knowledge-sharing sessions or pair programming.
 - Document learnings and best practices for others to use.
- **Way Forward:**
 - Look for opportunities to guide newer team members.
 - Be proactive in offering support and feedback.
 - Track mentoring interactions and reflect on their impact.

Process Maps: Please list the names of all process maps in the iGrafx platform that relate to your specific role. If applicable, please list the names of process maps that relate to your specific role which have yet to be created, so that you get credit for these as well.

Manager Evaluation

Response:

Employee Evaluation

Response: N/A. I don't want have access to IGrafx.

Additional Feedback

Additional Feedback: Please feel free to tell us what else you would like us to know about your role in the company, or anything else on your mind.

Manager Evaluation

Response:

Employee Evaluation

Response: As a Software Engineer on the Data team, I've made a significant impact by refining and maintaining scalable data models that bridge technical precision with business needs. I've supported cross-functional teams by resolving data-related issues, ensuring seamless system integration and consistency. Through the creation of clear Entity-Relationship Diagrams (ERDs), I've helped both technical and non-technical stakeholders understand complex data structures. My work in data transformation, automation of repetitive tasks, and comprehensive documentation has streamlined workflows, improved data accuracy, and enhanced team efficiency.