Eric Roberts

Summer 2021: Bi-Annual Check-in Self Evaluation and Managerial Feedback

TABLE OF CONTENTS

Summary remarks

Akshata Naikar
Engineering Manager II

Eric, in the past six months you've been developing the skills and experience to grow into a strong individual contributor on the team. You have done a solid job on the PAC team working on building out Grassroots functionality and implementing the PAC settings form, among other features. I also wanted to commend you for building the UserValidationError system. Thank you for writing this and improving the lives of other developers too!
Thank you Eric, for your hard work and all your contributions over the past six months.

Manager review

Akshata Naikar
Engineering Manager II

1. Key Accomplishments

In this section, please highlight the team member's most impactful contributions

- For team members with Career Growth Frameworks:
  Based upon your selections and comments in the Skills Matrix, provide a summary of the team member's performance for the evaluation period. Goal length: 4 - 8 sentences.
- For team members without Career Growth Frameworks: Based upon your comments in the Key Accomplishments and Areas for Continuous Improvement section below, provide a summary of the team member's performance for the evaluation period. Goal length: 4 - 8 sentences.

Example:
Key Accomplishments:
In the last six months, Jasmine rolled out Quorum Odyssey 3.0 to all Quorum clients. The publication of a Business Case helped multiple stakeholders to have a better understanding of the product positioning and how to answer Frequently Asked Questions (FAQs). Jasmine did a strong job of involving necessary decision-makers (e.g., Abu for Business Development Team) and training Customer Support Specialists to support the launch of upcoming functionality. Furthermore, Jasmine took the lead on outlining clear business objectives (e.g., Earn $4,500 MRR within 30 days of launch) that will enable us to determine whether or not the Research and Development investment in Quorum Odyssey 3.0 was worthwhile and validates a real market opportunity. Jasmine took care to not only establish the business objectives but also put together documentation with an exhaustive list of common business objectives for product development that can be repurposed to save time for future product and feature launches.

In the last six months, Eric has done a solid job on the PAC team working on building out Grassroots functionality and implementing the PAC settings form, among other features. He continues to be an independent and capable engineer that the team can rely on. He can take requirements and convert them into a technical implementation with minimal support. His code is generally high quality and when there are bugs, he quickly addresses them. I also wanted to commend Eric for building the UserValidationError system. He identified a common problem - sending error information to the frontend and displaying it to the user. Rather the reimplementing the boilerplate error handling logic, he worked on a general purpose solution that can be dropped where needed. Thank you for writing this and improving the lives of other developers too!

2. Areas of Improvements

In this section, please select three or fewer skills that the team member should spend the most time working to improve over the next six months.

- For team members with Career Growth Frameworks:
  Select two attributes from the Skills Matrix that the individual should focus on for improvement in the six-month term ahead. When possible, provide relevant examples of instances from the past six months in which the individual could have been more successful in achieving a particular outcome. Goal length: 8 - 16 sentences.
- For team members without Career Growth Frameworks:
  Select two to four skills and/or behaviors that the individual should focus on for improvement in the six-month term ahead. When possible, provide relevant examples of instances from the past six months in which the individual could have been more successful in achieving a particular outcome. Goal length: 16 - 24 sentences.

Example:
Areas for Continuous Improvement:
Project Management:
Jasmine regularly completes projects from beginning to end and identifies all necessary stakeholders. Jasmine also proactively creates communications plans, particularly for those with high influence. In the past six months, she has made considerable improvements in delivering results against target due dates. In order to continue to improve at Project Management, Jasmine needs to explicitly define project quality standards (e.g., in our finished product, team members will be able to access Client Health Notes via Salesforce in 20 seconds or less) at the beginning of a project and identify how she wants to measure whether or not she has achieved the quality standards. For example, in order to determine whether or not the Quorum Odyssey 3.0 launch was successful, she will need baseline information about current utilization (0% DAU/MAU, 0% MAU/YAU), identify the goal level of utilization (e.g., 1% DAU/MAU, 10% MAU/YAU), and measure actual utilization at the end of the project in order to determine whether she can close out the project or if she needs to cycle back around to improve the quality.

Code Quality

- Keep documentation and comment clarity in mind for future maintainability
- Look to keep things DRY and composable
- Make it a habit to scan your code once you're done with the core implementation. You should be looking for missing docstrings, missing tests, unclear code/comments, etc. Before I create a PR, I always look over my diff with these things in mind so I make sure to get ahead of some of the reviewer comments.
- Continue working on your code review rigor. As you advance in your career, you will spend a larger proportion of your time giving feedback to others. See these videos I created for some of my thoughts on code review and scan through some of the code review Matt has given other team members on PAC for more inspiration.

3. Performance Summary

In this section, please offer a summary of this team member's performance.
-For team members with Career Growth Frameworks:
Provide a summary of the team member's performance for the evaluation period. Goal length: 4 - 8 sentences.
-For team members without Career Growth Frameworks: Based upon your comments in the Key Accomplishments and Areas for Continuous Improvement section above, provide a summary of the team member's performance for the evaluation period. Goal length: 4 - 8 sentences.
The Performance Summary section should be able to double as the verbal summary that you present during department calibration sessions.
Example:
Performance Summary:
Overall, Jasmine is a strong individual contributor who has developed the skills and experience to have an impact on the team beyond herself. She has an advanced level of understanding of all Quorum products and is capable of producing high-quality individual work. She very consistently epitomizes the Customer Success team's purpose and guiding principles and provides a model for other team members to aspire to. Jasmine's greatest areas for improvement in her current role include planning, scoping, and executing on projects, specifically defining and measuring quality standards and avoiding scope creep; retaining clients at or above retention goals; and taking ownership of cross-functional projects that are outside of her comfort zone to expand the breadth of her corporate knowledge.

Eric, in the past six months you've been developing the skills and experience to grow into a strong individual contributor on the team. You have done a solid job on the PAC team working on building out Grassroots functionality and implementing the PAC settings form, among other features. I also wanted to commend you for building the UserValidationError system. Thank you for writing this and improving the lives of other developers too!

Thank you Eric, for your hard work and all your contributions over the past six months. I'm looking forward to our Bi-Annual Check-In conversation next week. Before that, please take a look at the review packet and **note specific sections in the review (with page numbers)** that you'd like to discuss. Also, **note specific competency areas where you'd like to focus on** with respect to growth and professional development. Please **note 3-5 goals for yourself**, which we can discuss and consolidate with my goals for you.I've booked an hour slot and we will follow the below format.

1. First, I'll give a brief overview of my observations.
2. Next, I would like to hear from you about how the past six months were for you, what are the things that you are most proud of and what are the more challenging/biggest areas for growth.
3. We will then move into discussing any of the sections in my review of you that you'd like more context on or want to talk about further, with you leading this portion of the conversation.
4. We will then ensure we are on the same page about path forward and goals for the next 6 months.
5. Lastly, we will discuss your concerns noted in your self review and discuss your ideas/potential solutions for the same.

Additionally, if you feel comfortable, please feel free to provide feedback to me and areas where I need to grow in order to better support you as a manager. If not, please feel free to reach out to Jonathan Marks for any specific concerns/issues you may have with me.

4. Leadership - Mastery

Developing: "- When engaged by other engineers, provides personal, professional, and technical support. - Often seeks out mentorship and feedback with an eye towards self-improvement. - Is developing and practicing leadership traits, like professionalism, work ethic, honest, productivity, and empathy."

Proficient: "- Often proactively engages other engineers to offer high quality personal, professional, and technical support. - Helps create a safe environment for others to learn & grow. - Takes on small roles that involve leadership (e.g. being an onboarding buddy, a mentor) - Often offers time to coach teammates and facilitate growth in an open, respectful, flexible, and empathetic manner. - Consistently demonstrates strong leadership traits. - Viewed by other peers as a strong engineer and teammate."

Advanced: "- Consistently seeks opportunities to and regularly engages with other engineers to provide high quality personal, professional, and technical support. - Helps create a safe environment for others to learn & grow - Often takes on roles that involve leadership (being a tech lead, onboarding buddy, mentor, etc.) - Almost always demonstrates strong leadership traits. - Is a role model for other engineers. - Participates in evaluating candidates and conducting interviews"

Excellent: "- Raises the bar for leadership across the department. - Consistently creates opportunities to engage with other engineers to provide exceptional personal, professional, and technical support. - Consistently takes on roles that involve leadership (being a tech lead, onboarding buddy, mentor, etc.) - Mentors across teams in an open, respectful, and empathetic manner. Improves the entire organization by teaching others and sharing knowledge. - Fosters a culture of mentorship by creating opportunities for others to showcase and develop their skills. - Is a role model for team members company-wide. - Helps set hiring direction through a combination of candidate evaluation, interviews, and proactive recruiting."

5. Leadership - General Comments

In the past six months, you have been offering support and guidance over multiple areas like test coverage and Docker. You were able to add Docker set-up on Windows and local elastic search sections to respective guides. You've onboarded Akshat and helped him troubleshoot various issues both on Mac and Windows OS. You also contributed to the hiring process by conducting interviews. Going forward, as you familiarize yourself with our technical systems, you should continue to take ownership on some of the features and eventually be a reference for other members in the future.

6. Problem Solving (SE) - Mastery

Developing: "- Demonstrates ability to debug and solve familiar code and systems and a growing knowledge of those systems. - Approaches each task as an opportunity to learn and continually applies learning to subsequent challenges. - Seeks to solve problems by themselves and is sometimes able to solve them. Appropriately reaches out to other engineers for help and guidance when encountering a problem beyond their knowledge or experience. - Participates in technical design conversations and occasionally offers suggestions or advice."

Proficient: "- Demonstrates strong debugging skills and often solves problems in both familiar and unfamiliar systems. - Consistently recognizes own mistakes and uses them as learning and teaching opportunities. Iteratively adjusts problem solving approach to minimize the risk of repeating mistakes. - Often follows ""Fix It the First Time"" where appropriate to avoid recurrence of problems. - Plays a large role in technical decisions involving projects assigned to them. Considers the impact of technical decisions on both the project at hand as well as the system overall. - Proactively anticipates challenges to completing both functional and non-functional requirements. Demonstrates attention to detail and care about all parts of the system in question."

Advanced: "- Demonstrates strong debugging skills and consistently solves problems in both familiar and unfamiliar systems. - Anticipates problems before they occur and often builds systems that successfully mitigate risks and exploit opportunities. - Consistently balances between helping others solve problems and simply solving the problem themselves. - Plays a large role in technical decisions for both their own projects and consults on design for projects that impact the team and system as a whole. - Almost always follows ""Fix It the First Time"" where appropriate to avoid recurrence of problems. - Proactively anticipates and designs solutions for problems that might impact both a particular project and the system overall. - Demonstrates significant understanding of both systems they own and the system as a whole and identifies additional requirements and challenges that would be otherwise overlooked."

Excellent: "- Consistently anticipates and solves problems in both familiar and unfamiliar systems. - Consistently leads design on large scale projects critical to business and builds systems that successfully mitigate risks and exploit opportunities. - Almost always chooses the right balance of solving problems and empowering others to solve problems. - Consistently designs critical architecture that is successful across multiple dimensions, including performance, scalability, robustness, and maintainability. - Almost always designs architecture that is robust against single points of failure, both in terms of systems and people. - Identifies barriers that are slowing down the team and initiatives at Quorum and creates practical technical solutions to increase efficiency. - Demonstrates significant understanding of both systems they own and the system as a whole, and consistently identifies additional requirements and challenges that would be otherwise overlooked."

7. Problem Solving - General Comments

You have done a great job in fixing bugs over various parts of the product. Going forward, you should continue to increase your ability to satisfy the functional and non-functional requirements of a project and also have an increased voice in technical design decisions.

8. Process Adherence and Improvement - Mastery

Developing: "- Understands and consistently follows the team's practices and processes (e.g. proper pull request workflow, checking for locks before applying migrations, organizing tickets in Jira, etc.) - Adapts to new processes & systems, thinks critically about their impact, and communicates about them with other members of the team. - Is learning to measure and optimize their individual workflow. - Is learning and occasionally suggests improvements to processes that can benefit the team."

Proficient: "- Consistently follows Quorum's established processes. - Sometimes identifies new opportunities for the team by investigating new technologies and processes where appropriate. With support from other team members and following approval, executes projects to implement new systems and make the rest of the team better. - Often identifies improvements to their individual workflows (e.g., I currently manage my individual To Do List in X way. If I switched to Y way, I think I would increase my turnaround time on tasks.) in addition to improvements in team-wide process."

Advanced: "- Almost always follows Quorum's established processes. - Consistently identifies new opportunities for the team by investigating new technologies and processes. With support from other team members and following approval, executes projects to implement new systems and make the rest of the team better. - Consistently looks for ways to improve the team where appropriate, piloting new systems and pushing them forward to impact the rest of the team. - Consistently identifies and implements improvements to their individual workflows (e.g., I currently manage my individual To Do List in X way. If I switched to Y way, I think I would increase my turnaround time on tasks.) in addition to improvements in team-wide process. - Supports and mentors other members of the team to help them execute processes."

Excellent: "- Almost always follows Quorum's established processes. - Consistently spearheads and delivers on process improvements at the individual, team, department, and company-wide levels (e.g. cross-department incident response practices). - Exhibits strong decision making in choosing the most critical process challenges and prioritizes them based on maximizing business impact. - Has a track record of successfully changing how Quorum engineering functions to meet multiple business critical needs. - Fosters a culture of within the team of continuous improvement on process and consistently supports, mentors, and coaches other members of the team on process improvement and execution, resulting in increased buy-in and team-wide optimization."

9. Process Adherence and Improvement - General Comments

Thank you on your work on UserValidationError system and making lives of other developers easier! Moving forward, you should continue to think about how the team can modify existing or introduce new processes to improve the effectiveness of the team.

10. Execution of Epics - Mastery

Developing	**Proficient**	Advanced	Excellent
Developing: "- Attempts to complete epic work on time and with high quality and is able to incorporate feedback and support from other engineers to do so. - Understands what ""high quality"" means and is able to demonstrate that knowledge in their code, though they may require prompting from other engineers. - Understands and acts according to epic task prioritization. - Once feedback has been given about a project, is able to learn from and apply that feedback to the rest of the project."	**Proficient: "- Often completes epic work on time. - Consistently completes epic work with high quality according to business and technical requirements. - Often contributes to epic task prioritization to maximize the probability of success and escalates concerns to product or technical stakeholders. - Gives and applies prioritization feedback on epic work. - Often meets inter-project goals and supports relevant project stakeholders (such as product testing sessions)."**	Advanced: "- Consistently executes epic work on time without shortcuts. - Almost always completes epic work with high quality according to business and technical requirements. - Consistently plays an integral role in prioritization, project organization, scoping and roadmapping. - Often provides and relays feedback to other engineers within an epic, breaking down knowledge silows and fostering collaboration. - Consistently meets intra-project goals and supports relevant project stakeholders (such as product testing sessions)."	Excellent: "- Almost always executes projects with a stellar degree of quality, timeliness, and without shortcuts. - Almost always completes epic work with high quality according to business and technical requirements. - Independently scopes and prioritizes both high level and low-level tasks, and consistently anticipates and works to resolve problems in project roadmapping. - Provides consistent support to ensure any engineers working on a given project also execute with high quality and timeliness, using tactics such as code and documentation review as well as in-person feedback. - Almost always meets intra-project goals and supports relevant project stakeholders (such as product testing sessions)."

11. Execution of Epics - General Comments

In the past six months, you have consistently delivered on all your assigned tasks. For PAC epic, you have stepped up to ensure you have clear understanding of requirements by collaborating with other team members, given the fact that requirements by product was unclear. I am glad that you are not discouraged by scope creep and always ensure you continue to implement additional changes as required. Going forward, I encourage you to continue to alert your lead/manager on scope creep and reset the expectations on the same.

12. Maintenance / Support - Mastery

Developing: "- Is responsive to Opsgenie reporting systems and reactive JIRA tickets and occasionally requires support from other engineers to determine if a report is a real issue or to resolve a particular problem. - Is occasionally proactive in solving problems by preventing the recurrence of many similar issues, and is learning how to spot opportunities to fix issues catagorically. - Is still learning about Quorum's infrastructure how to debug and solve common types of errors across the stack."

Proficient: "- Responds to issues in Opsgenie reporting systems and reactive JIRA tickets without prompting or oversight. - Often demonstrates strong decision making in solving a problem themselves or escalating. - Demonstrates a mix of proactive and reactive solutions to issues and thinks about how to proactively avoid problems in systems they maintain. - Is able to solve most common problems in our infrastructure themselves. Demonstrates strong debugging & problem-finding skills to resolve many uncommon issues."

Advanced: "- Consistently proactively anticipates problems with systems they maintain. - Consistently demonstrates strong decision making in solving a problem themselves or escalating. - Is consistently responsive to bugs and small feature requests and appropriately prioritizes them based on business need. Communicates clearly to relevant stakeholders across teams and departments. - Consistently resolves issues with both their own systems and the system as a whole, and often identifies root causes where possible. - Supports other engineers when problems occur and helps them identify ways to avoid issues in others' systems."

Excellent: "- Almost always proactively anticipates problems with systems they maintain and develops processes and systems to avoid, measure, or anticipate problems. - Almost always demonstrates strong decision making in solving a problem themselves or escalating. - Is exceedingly responsive to bugs and small feature requests and appropriately prioritizes them based on business need. Communicates clearly to relevant stakeholders across teams and departments. - Almost always resolves issues with both their own systems and the system as a whole, and consistently identifies root causes where possible. - Builds systems after problems occur to identify, avoid, and resolve future problems. - Almost always supports other engineers when problems occur and helps them identify ways to avoid issues in others' systems."

13. Maintenance / Support - General Comments

Over the last 6 months, you have become more comfortable being on-call and have been responding to genie alerts. You are able to fix issues in a timely manner on the areas of product you are familiar with or able to escalate as needed. Moving forward, you should continue to familiarize yourself with Quorum's

infrastructure and monitoring tools and help support the team. You should continue to adopt "fix it the first time" mentality and build systems to prevent issues from occurring again in the future.

14. Code Quality - Mastery

Developing: "- Often writes code that meets standards for style and functionality. - Is learning to understand and write topical tests for the full impact of code they write. - Occasionally builds some systems that are abstract, legible, or easy to use by other engineers. - Is learning to write clear and useful documentation and improves them when prompted by others. - Occasionally considers observability and monitoring when writing code."

Proficient: "- Consistently writes code that meets or exceeds standards for style and functionality. - Consistently understands and tests for the full impact of code they write. - Often designs code for robustness and extensibility. - Occasionally builds abstract systems that are easy to interpret and be used by other engineers. - Often writes documentation that is thorough and clear, and keeps it up to date if prompted. - Often considers observability and monitoring."

Advanced: "- Almost always writes high quality code that is legible, performant, robust, and maintainable. - Consistently understands the full impact of code they write. Designs tests that are non-brittle, scale well, are maintainable, and avoid practically all avoidable issues. - Consistently designs code for robustness and extensibility. - Often builds abstract systems that are easy to interpret and be used by other engineers with an eye towards future business needs. Drives adoption of these systems. - Consistently writes and updates documentation that is thorough, clear, and used by others to help themselves. - Consistently considers observability and monitoring."

Excellent:"- Sets the standard for code quality among features, languages, and systems they build and maintain. - Almost always writes high quality code that is legible, performant, robust, and maintainable by themselves and others. - Almost always understands the full impact of code they write. Designs tests that are non-brittle, scale well, are maintainable, and avoid practically all avoidable issues. - Consistently builds and drives adoption of abstract systems that easy to use by other engineers and add critical business value. - Leads initiatives to improve testing infrastructure and considers stability, maintainability, and performance in running testing in both development and production. - Almost always writes and updates documentation that is thorough, clear, and used by others to help themselves. Keeps it up to date without prompting. - Almost always considers observability and monitoring in production and builds legible monitoring tools useful to the entire team."

15. Code Quality - General Comments

Feedback from Mares
Eric should continue working on his software engineering skills, such as code organization, architecture, and readability. See this PR on pledge amounts where Matt provided feedback on structure and implementation details. Also see this PR, where you were challenged to use some of the ledger functionality to check supporter donor eligibility. Here are some primary takeaways from that PR:

- Sometimes, we should yell loudly and raise an error rather than handle it. Resist the urge to always be "gentle" and make the code not error out. For example, in this case, handling a DoesNotExist doesn't make sense because it is a precondition of the function that the account exists. You should keep in mind a function's preconditions/postconditions when thinking about error handling so that you prevent giving too much responsibility to the function.
- Keep documentation and comment clarity in mind for future maintainability
- Look to keep things DRY and composable (see this comment)

Some generalized suggestions for Eric:

- Make it a habit to scan your code once you're done with the core implementation. You should be looking for missing docstrings, missing tests, unclear code/comments, etc. Before I create a PR, I always look over my diff with these things in mind so I make sure to get ahead of some of the reviewer comments.
- Continue working on your code review rigor. As you advance in your career, you will spend a larger proportion of your time giving feedback to others. See these videos I created for some of my thoughts on code review and scan through some of the code review Matt has given other team members on PAC for more inspiration.

16. Domain Expertise - Mastery

Developing	**Proficient**	Advanced	Excellent
Developing: "- Demonstrates growing knowledge of key Quorum concepts and technical knowledge. - Seeks out opportunities to improve base skills and demonstrates ability to pick up new technical skills. - Focuses on growing their knowledge of Quorum's systems and technical skill set broadly. - Seeks support from other engineers to learn more."	**Proficient: "- Demonstrates strong knowledge of core Quorum concepts and technical knowledge with developing mastery in several areas. - Has learned how to gain additional knowledge about Quorum products and external technical skills. - Often demonstrates strong product knowledge in decision making. - Continues to develop their skills intentionally. - Has in-depth knowledge of immediate systems they've worked on and some knowledge of adjacent systems. "**	Advanced: "- Demonstrates high levels of Quorum product and technical knowledge with mastery in many areas. - Has strong, well-founded opinions about how to build software in their respective domain, but is adaptable and open to new ideas. - Continues to develop skills over time, focusing on both breadth and depth of technical knowledge. - Consistently demonstrates strong product knowledge in decision making. - Understands what tools, best practices and relevant industry trends are available and effectively applies them to projects when useful and appropriate. - Teaches teammates that may know less about their domain in a way that is understandable and not condescending. - May occasionally contribute to the technology ecosystem at large."	Excellent: "- Demonstrates high levels of Quorum product and technical knowledge with mastery in most areas - Consistently shares knowledge and seeks opportunities to teach others about their particular domain. - Thoughtfully and practically introduces concepts and technologies from their domain to solve important problems, and empowers teammates to learn and improve on these concepts. - Work based on their expertise is consistently proven lasting and successful. - Almost always demonstrates strong product knowledge in decision making. - Uses expertise to improve Quorum's capabilities in their domain. - Often contributes to the domain ecosystem at large. "

17. Domain Expertise - General Comments

In the past six months, you have explored and fixed bugs within various product features. You have better understanding of core features like Outbox, Bulk updates and Profiles. In the next six months, you should continue to work on expanding your understanding/knowledge of Grassroots and PAC, and aim to become the code maintainer. You should continue to work diligently on the issues/bugs assigned to you, continue to find the root cause of the issue along with the systems it impacts. You should continue to explore Quorum's complex systems and leverage the knowledge of other engineers to build your expertise.

18. Self-Organization - Mastery

Developing: "- Often completes small projects with some support or oversight. - Spends time researching a problem before looking to others for support. - Pays attention to how they spend their time, and seeks feedback from others on how to spend time well. - Occasionally goes down ""rabbit holes"" but attempts to identify cases where that occurs to avoid future recurrence."

Proficient: "- Often completes medium sized projects with little support or oversight. - Demonstrates willingness and ability to research and solve problems independently. - Spends time well and balances between major projects, maintenance, team tasks, and other responsibilities. - Thinks proactively about how to spend their time well, and requests support when necessary."

Advanced: "- Consistently completes large projects with little support and oversight - Can take a complex problem, break it down into tasks, and complete those tasks with relative ease - Ensures commitments are realistic, understands priority and urgency of tasks, and delivers on them accordingly - Consistently allocates their time effectively and efficiently - Proactively plans their time in well advance and correctly predicts time expenditure "

Excellent: "- Requires little oversight beyond high-level direction in all projects - Leverages their understanding of business value to spend time where valuable - Almost always allocates their time effectively and efficiently - Consistently plans their time well in advance and correctly predicts time expenditure - Goes beyond knocking tasks off a list by identifing and suggesting areas of future work or systems to save their and their team's time in the future."

19. Self-Organization - General Comments

In the past six months, you were able to manage your time well between PAC tasks, on call support, P1/P2 issues and onboarding duties. Going forward, I encourage you to continue to plan your tasks ahead of time and provide adequate estimates on tasks.

20. Communication - Mastery

Developing: "- Demonstrates desire and ability to communicate candidly, accurately, concisely, and regularly - Understands when to keep investigating and when to escalate problems when blocked - Asks for more context when appropriate - Occasionally considers relevant stakeholders and asks for guidance on how to engage them."

Proficient: "- Consistently communicates candidly, accurately, concisely, and regularly. - Collaborates well with team members as both a mentor and a mentee. - Takes in vague requirements and asks the right questions to ensure they are clarified. - Often escalates problems appropriately and considers relevant stakeholders. - Actively listens and seeks input from others to fully understand a problem and give thoughtful responses. - Understands when and how to appropriately and effectively offer feedback. - Seeks out and receives constructive criticism well. "

Advanced: "- Almost always communicates candidly, accurately, concisely, and regularly. - Consistently escalates problems quickly and keeps all stakeholders in the loop. - Facilitates discussion within their team, ensuring everyone has an opportunity to share their opinions and be heard, and that discussion outcomes and expectations tie to business goals. - Quickly extracts core issues from discussions and meetings to make them more productive. - Collaborates effectively with teammates and others outside them team. - Clearly and effectively gives and receives constructive feedback. - Actively and empathetically listens to ensure everyone gets a chance to share their thoughts and feels that they're heard."

Excellent: "- Almost always communicates candidly, accurately, concisely, and regularly. - Almost always escalates problems quickly and keeps all stakeholders in the loop - Facilitates conversations with disparate groups of people to help them collaborate, identify common goals, and reach consensus. Ensures everyone has an opportunity to share their opinions and be heard, and that discussion outcomes tie to business goals. Guides discussion towards decisions and gets buy-in. - Adapts their language to meet the needs of various levels of technical and non-technical audiences. - Clearly and effectively gives and receives constructive feedback. - Has others seek them out for advice on communication and for help giving difficult feedback. - Fosters a culture of effective communication among their team and the company at large."

21. Communication - General Comments

You have effective communication skills and you communicate clearly and concisely. You proactively reach out to Product and tech lead to get clarity on the requirements. You solicit feedback and act on the same. You explain your thought process and ideas on your code reviews. Going forward, you should

continue to share your thoughts and inputs on broader platform like dept-wide retrospectives.

22. Open to Individual Growth - Mastery

Developing: "Occasionally demonstrates an interest in improving skills relevant to their job, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Occasionally acts on feedback. Does not regularly seek out feedback. Occasionally seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job. Occasionally embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Has not begun building a network of external advisors and mentors."

Proficient: "Somewhat regularly demonstrates an interest in improving skills relevant to their job, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Somewhat regularly seeks out feedback and somewhat regularly acts on that feedback. Somewhat regularly seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job. Somewhat regularly embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Is working on building a network of external advisors and mentors."

Advanced: "Regularly demonstrates a personal commitment to improving skills relevant to their job and to their future career growth at Quorum, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Regularly seeks out feedback, including from other team members, and regularly acts on that feedback. Regularly proactively seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job and somewhat regularly uses them to suggest or implement improvements at Quorum. Regularly embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Is working on building or has built a network of external advisors and mentors."

Excellent: "Almost always demonstrates a strong, personal commitment to improving skills relevant to their job and to their future career growth at Quorum, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Almost always seeks out feedback, including from other team members, and regularly acts on that feedback. Almost always proactively seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job and regularly uses them to suggest or implement improvements at Quorum. Almost always embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Has built and continues to cultivate a network of external advisors and mentors and uses them as resources."

23. Open to Individual Growth - General Comments

You've been intentional about your personal growth at Quorum. You often reach out to team members and via slack channels on things you are stuck on. You are constantly working towards increasing your knowledge on tools and tech stack. You regularly invest your time on reading on various blogs. As you grow in your role, I encourage you to continue to build professional network of peers/mentors to further your software development career. You should continue to take ownership over your individual growth and exemplify Quorum's company values through your work.

24. Supports Quorum's Growth - Mastery

Developing	**Proficient**	Advanced	Excellent
Developing: "Occasionally demonstrates an interest in building trust with individuals across Quorum, such as by offering to support other team members. Occasionally executes on quarterly OKRs. Occasionally participates in department- and team-wide conversations."	**Proficient: "Somewhat regularly demonstrates an interest in building trust with individuals across Quorum, such as by offering to support other team members. Somewhat regularly executes on quarterly OKRs in a timely manner. Somewhat regularly participates in department- and team-wide conversations. Somewhat regularly sets an example for continuous improvement. Occasionaly demonstrates an interest in growing the Quorum team by conducting interviews, creating a high-quality candidate experience, and submitting scorecards in a timely manner. "**	Advanced: "Regularly demonstrates a personal investment in building trust with individuals across Quorum, such as by offering to support other team members. Regularly executes on quarterly OKRs in a high-quality and timely manner. Regularly participates in department- and team-wide conversations and somewhat regularly initiates important conversations. Regularly sets an example for continuous improvement that colleagues seek to emulate. Somewhat regularly takes the lead on projects of importance that benefit Quorum, such as company-wide OKRs and other priorities. Regularly demonstrates an interest in growing the Quorum team by conducting interviews, creating a high-quality candidate experience, and submitting scorecards in a timely manner. "	Excellent: "Almost always demonstrates a personal investment in building trust with individuals across Quorum, such as by offering to support other team members, including outside their own department. Almost always executes on quarterly OKRs in a high-quality and timely manner. Almost always participates in department- and team-wide conversations and regularly initiates important conversations. Almost always sets an example for continuous improvement that colleagues seek to emulate. Regularly takes the lead and owns the execution on projects of importance that benefit Quorum, such as company-wide OKRs and other priorities. Almost always demonstrates an interest in growing the Quorum team by conducting interviews, creating a high-quality candidate experience, and submitting scorecards in a timely manner. "

25. Supports Quorum's Growth - General Comments

In the past six months, you have been intentional about contributing to team growth and success by

- Onboarding Akshat, our summer intern

- Sharing your thoughts and feedback on process changes
- Conducting interviews
- Offering support to other team members and performing code reviews

Moving forward, I would encourage you to build rapport with individuals across different teams within Quorum. Along with understanding our development team dynamics and objectives, it is essential to have an insight into cross-departmental functions and goals. I would also enciurage you to invest time in exploring tools/processes/best practices and suggest improvements to the engineering team.

Manager review

Akshata Naikar
Engineering Manager II

1. Overall Performance Score

- ○ Needs Improvement
- ○ Meets expectations
- ✓ **Exceeds expectations**
- ○ Strongly Exceed Expectations
- ○ Truly outstanding
- ○ Ramping (New Hires Only)

2. Overall Performance Stage

- ○ Early
- ✓ **Mid**
- ○ Late

Self review

ER
Eric Roberts
Software Engineer II

1. Leadership - Mastery

Developing: "- When engaged by other engineers, provides personal, professional, and technical support. - Often seeks out mentorship and feedback with an eye towards self-improvement. - Is developing and practicing leadership traits, like professionalism, work ethic, honest, productivity, and empathy."

Proficient: "- Often proactively engages other engineers to offer high quality personal, professional, and technical support. - Helps create a safe environment for others to learn & grow. - Takes on small roles that involve leadership (e.g. being an onboarding buddy, a mentor) - Often offers time to coach teammates and facilitate growth in an open, respectful, flexible, and empathetic manner. - Consistently demonstrates strong leadership traits. - Viewed by other peers as a strong engineer and teammate."

Advanced: "- Consistently seeks opportunities to and regularly engages with other engineers to provide high quality personal, professional, and technical support. - Helps create a safe environment for others to learn & grow - Often takes on roles that involve leadership (being a tech lead, onboarding buddy, mentor, etc.) - Almost always demonstrates strong leadership traits. - Is a role model for other engineers. - Participates in evaluating candidates and conducting interviews"

Excellent: "- Raises the bar for leadership across the department. - Consistently creates opportunities to engage with other engineers to provide exceptional personal, professional, and technical support. - Consistently takes on roles that involve leadership (being a tech lead, onboarding buddy, mentor, etc.) - Mentors across teams in an open, respectful, and empathetic manner. Improves the entire organization by teaching others and sharing knowledge. - Fosters a culture of mentorship by creating opportunities for others to showcase and develop their skills. - Is a role model for team members company-wide. - Helps set hiring direction through a combination of candidate evaluation, interviews, and proactive recruiting."

2. Leadership - General Comments

Over the last six months I've been active in slack channels like #dev-testing to provide assistance to engineers when I can. In particular using my experience using docker to help with Docker problems. I've also proactively written solutions to technical problems I've had that I've included in documentation and shared, such as adding instructions for local elasticsearch.

Since Aksaht joined I've helped him get comfortable, set up his environment and helped him go through the onboarding exercises. I've been proactive in seeing if he needs help and assiting him when he needs it, as well as encouraging Akshat to go to social events.

I regularly participated in first round interviews during the last hiring process.

3. Problem Solving (SE) - Mastery

Developing: "- Demonstrates ability to debug and solve familiar code and systems and a growing knowledge of those systems. - Approaches each task as an opportunity to learn and continually applies learning to subsequent challenges. - Seeks to solve problems by themselves and is sometimes able to solve them. Appropriately reaches out to other engineers for help and guidance when encountering a problem beyond their knowledge or experience. - Participates in technical design conversations and occasionally offers suggestions or advice."

Proficient: "- Demonstrates strong debugging skills and often solves problems in both familiar and unfamiliar systems. - Consistently recognizes own mistakes and uses them as learning and teaching opportunities. Iteratively adjusts problem solving approach to minimize the risk of repeating mistakes. - Often follows ""Fix It the First Time"" where appropriate to avoid recurrence of problems. - Plays a large role in technical decisions involving projects assigned to them. Considers the impact of technical decisions on both the project at hand as well as the system overall. - Proactively anticipates challenges to completing both functional and non-functional requirements. Demonstrates attention to detail and care about all parts of the system in question."

Advanced: "- Demonstrates strong debugging skills and consistently solves problems in both familiar and unfamiliar systems. - Anticipates problems before they occur and often builds systems that successfully mitigate risks and exploit opportunities. - Consistently balances between helping others solve problems and simply solving the problem themselves. - Plays a large role in technical decisions for both their own projects and consults on design for projects that impact the team and system as a whole. - Almost always follows ""Fix It the First Time"" where appropriate to avoid recurrence of problems. - Proactively anticipates and designs solutions for problems that might impact both a particular project and the system overall. - Demonstrates significant understanding of both systems they own and the system as a whole and identifies additional requirements and challenges that would be otherwise overlooked."

Excellent: "- Consistently anticipates and solves problems in both familiar and unfamiliar systems. - Consistently leads design on large scale projects critical to business and builds systems that successfully mitigate risks and exploit opportunities. - Almost always chooses the right balance of solving problems and empowering others to solve problems. - Consistently designs critical architecture that is successful across multiple dimensions, including performance, scalability, robustness, and maintainability. - Almost always designs architecture that is robust against single points of failure, both in terms of systems and people. - Identifies barriers that are slowing down the team and initiatives at Quorum and creates practical technical solutions to increase efficiency. - Demonstrates significant understanding of both systems they own and the system as a whole, and consistently identifies additional requirements and challenges that would be otherwise overlooked."

4. Problem Solving - General Comments

Over the past 6 months I've solved numerous bugs in various parts of the system and have used the oppertunities to learn the different technical pieces of Quorum so I can more quickly identify bugs in the future. For instance, fixing bugs in the address parsing logic, in blanked email and template detection for outbox, and improving my own understanding of Facebook's Graph API when solving problems with my own lead ads system, and working in search profiles when necessary.

I've been proactive in figuring finding and repoting bugs I've come accross, understanding the cause and opening tickets with enough information for myself or others to fix them when they can be prioritized. For instance, tracking a sheets regression reported to #sentry-new while on call.

When my own PRs cause regressions or have to be reverted, beyond understanding the technical cause I identify issues in our testing infastructure that could prevent these issues from happening again. For instance, working around issues in cypress that prevented us from testing anonymous GR users to prevent a regression for that scenerio.

5. Process Adherence and Improvement - Mastery

Developing: "- Understands and consistently follows the team's practices and processes (e.g. proper pull request workflow, checking for locks before applying migrations, organizing tickets in Jira, etc.) - Adapts to new processes & systems, thinks critically about their impact, and communicates about them with other members of the team. - Is learning to measure and optimize their individual workflow. - Is learning and occasionally suggests improvements to processes that can benefit the team."

Proficient: "- Consistently follows Quorum's established processes. - Sometimes identifies new opportunities for the team by investigating new technologies and processes where appropriate. With support from other team members and following approval, executes projects to implement new systems and make the rest of the team better. - Often identifies improvements to their individual workflows (e.g., I currently manage my individual To Do List in X way. If I switched to Y way, I think I would increase my turnaround time on tasks.) in addition to improvements in team-wide process."

Advanced: "- Almost always follows Quorum's established processes. - Consistently identifies new opportunities for the team by investigating new technologies and processes. With support from other team members and following approval, executes projects to implement new systems and make the rest of the team better. - Consistently looks for ways to improve the team where appropriate, piloting new systems and pushing them forward to impact the rest of the team. - Consistently identifies and implements improvements to their individual workflows (e.g., I currently manage my individual To Do List in X way. If I switched to Y way, I think I would increase my turnaround time on tasks.) in addition to improvements in team-wide process. - Supports and mentors other members of the team to help them execute processes."

Excellent: "- Almost always follows Quorum's established processes. - Consistently spearheads and delivers on process improvements at the individual, team, department, and company-wide levels (e.g. cross-department incident response practices). - Exhibits strong decision making in choosing the most critical process challenges and prioritizes them based on maximizing business impact. - Has a track record of successfully changing how Quorum engineering functions to meet multiple business critical needs. - Fosters a culture of within the team of continuous improvement on process and consistently supports, mentors, and coaches other members of the team on process improvement and execution, resulting in increased buy-in and team-wide optimization."

6. Process Adherence and Improvement - General Comments

I strive to make my workflow more efficient and share the results with the team. For instance, I keep directory of tools including shell scripts and aliases that I use regularly and that would cost me more time developing if they were not there. Expamples are finding the latest hotfix branch on origin, logging jira smart commits without affecting your current branch (so it behaves similar to the cli) and checking

changed files between the latest hotfix and merge-base.

I've been proactive in sharing these files with the team, and they are all currently in the "development/dotfiles/eric" directory of the repository. They also include notes on environment setup so I can easily share, for instance, lists of macport dependencies if necessary.

Additionally I actively add features to the codebase in a general way when I believe they can be useful to the rest of the team. For instance, introducing a UserValidationError class which automatically gets converted into 400 errors that will integrate with BACKENDERROR to create swals, and redux forms. Additionally adding and modifying test helpers for what would otherwise be done in less efficient ways, such as providing ability to add eligible ledger settings to supporter on creation or functions like assertDictShallowSubset.

7. Execution of Epics - Mastery

Developing: "- Attempts to complete epic work on time and with high quality and is able to incorporate feedback and support from other engineers to do so. - Understands what ""high quality"" means and is able to demonstrate that knowledge in their code, though they may require prompting from other engineers. - Understands and acts according to epic task prioritization. - Once feedback has been given about a project, is able to learn from and apply that feedback to the rest of the project."

Proficient: "- Often completes epic work on time. - Consistently completes epic work with high quality according to business and technical requirements. - Often contributes to epic task prioritization to maximize the probability of success and escalates concerns to product or technical stakeholders. - Gives and applies prioritization feedback on epic work. - Often meets inter-project goals and supports relevant project stakeholders (such as product testing sessions)."

Advanced: "- Consistently executes epic work on time without shortcuts. - Almost always completes epic work with high quality according to business and technical requirements. - Consistently plays an integral role in prioritization, project organization, scoping and roadmapping. - Often provides and relays feedback to other engineers within an epic, breaking down knowledge silows and fostering collaboration. - Consistently meets intra-project goals and supports relevant project stakeholders (such as product testing sessions)."

Excellent: "- Almost always executes projects with a stellar degree of quality, timeliness, and without shortcuts. - Almost always completes epic work with high quality according to business and technical requirements. - Independently scopes and prioritizes both high level and low-level tasks, and consistently anticipates and works to resolve problems in project roadmapping. - Provides consistent support to ensure any engineers working on a given project also execute with high quality and timeliness, using tactics such as code and documentation review as well as in-person feedback. - Almost always meets intra-project goals and supports relevant project stakeholders (such as product testing sessions)."

8. Execution of Epics - General Comments

I've primarily worked on PAC this review cycle, which has been a large Epic with many challenges and roadblocks. This has required me to better learn working with others in organizing dependent issues, meeting deadlines, finding the correct people to look over work or who have special knowledge and working around when requirements are unclear or don't exist.

For instance, one project required building a settings page to collect information for the FEC Form 1 statement of organization, and provide client side validation to meet our backend requriements and ensure the client is following FEC requirements. This involved reaching out to more FEC-knowledgable like

Matt and Camille and doing my own reserch reading FEC requirements.

In another instance working on a system that heavily depended on Payroll dates and creating a proper system for calculating payroll dates based on precise requirements, while trying to fill in the blanks of vague requirements while communiciating what I'm doing and confirming I'm headed in the right direction, or changing if not. This also involved expanding the scope of other tickets when realizing certain payroll configurations will have different number of payroll cycles depending on the current year.

I've taken and given input on ticket prioritization, for instance stopping work on review tickets to help with Home Depot requirements and switching the order I do tickets to prevent blocking others.

9. Maintenance / Support - Mastery

Developing: "- Is responsive to Opsgenie reporting systems and reactive JIRA tickets and occasionally requires support from other engineers to determine if a report is a real issue or to resolve a particular problem. - Is occasionally proactive in solving problems by preventing the recurrence of many similar issues, and is learning how to spot opportunities to fix issues catagorically. - Is still learning about Quorum's infrastructure how to debug and solve common types of errors across the stack."

Proficient: "- Responds to issues in Opsgenie reporting systems and reactive JIRA tickets without prompting or oversight. - Often demonstrates strong decision making in solving a problem themselves or escalating. - Demonstrates a mix of proactive and reactive solutions to issues and thinks about how to proactively avoid problems in systems they maintain. - Is able to solve most common problems in our infrastructure themselves. Demonstrates strong debugging & problem-finding skills to resolve many uncommon issues."

Advanced: "- Consistently proactively anticipates problems with systems they maintain. - Consistently demonstrates strong decision making in solving a problem themselves or escalating. - Is consistently responsive to bugs and small feature requests and appropriately prioritizes them based on business need. Communicates clearly to relevant stakeholders across teams and departments. - Consistently resolves issues with both their own systems and the system as a whole, and often identifies root causes where possible. - Supports other engineers when problems occur and helps them identify ways to avoid issues in others' systems."

Excellent: "- Almost always proactively anticipates problems with systems they maintain and develops processes and systems to avoid, measure, or anticipate problems. - Almost always demonstrates strong decision making in solving a problem themselves or escalating. - Is exceedingly responsive to bugs and small feature requests and appropriately prioritizes them based on business need. Communicates clearly to relevant stakeholders across teams and departments. - Almost always resolves issues with both their own systems and the system as a whole, and consistently identifies root causes where possible. - Builds systems after problems occur to identify, avoid, and resolve future problems. - Almost always supports other engineers when problems occur and helps them identify ways to avoid issues in others' systems."

10. Maintenance / Support - General Comments

During my numerous on calls in the last review cycle I've responded to ticket in numerous parts of the system, identified and reported problems. When necessary I find the correct maintainer to notify of the issue or escalate to a more knowledgable set of engineers if a serious issue occurs beyond my

knowledge.

11. Code Quality - Mastery

Developing: "- Often writes code that meets standards for style and functionality. - Is learning to understand and write topical tests for the full impact of code they write. - Occasionally builds some systems that are abstract, legible, or easy to use by other engineers. - Is learning to write clear and useful documentation and improves them when prompted by others. - Occasionally considers observability and monitoring when writing code."

Proficient: "- Consistently writes code that meets or exceeds standards for style and functionality. - Consistently understands and tests for the full impact of code they write. - Often designs code for robustness and extensibility. - Occasionally builds abstract systems that are easy to interpret and be used by other engineers. - Often writes documentation that is thorough and clear, and keeps it up to date if prompted. - Often considers observability and monitoring."

Advanced: "- Almost always writes high quality code that is legible, performant, robust, and maintainable. - Consistently understands the full impact of code they write. Designs tests that are non-brittle, scale well, are maintainable, and avoid practically all avoidable issues. - Consistently designs code for robustness and extensibility. - Often builds abstract systems that are easy to interpret and be used by other engineers with an eye towards future business needs. Drives adoption of these systems. - Consistently writes and updates documentation that is thorough, clear, and used by others to help themselves. - Consistently considers observability and monitoring."

Excellent:"- Sets the standard for code quality among features, languages, and systems they build and maintain. - Almost always writes high quality code that is legible, performant, robust, and maintainable by themselves and others. - Almost always understands the full impact of code they write. Designs tests that are non-brittle, scale well, are maintainable, and avoid practically all avoidable issues. - Consistently builds and drives adoption of abstract systems that easy to use by other engineers and add critical business value. - Leads initiatives to improve testing infrastructure and considers stability, maintainability, and performance in running testing in both development and production. - Almost always writes and updates documentation that is thorough, clear, and used by others to help themselves. Keeps it up to date without prompting. - Almost always considers observability and monitoring in production and builds legible monitoring tools useful to the entire team."

12. Code Quality - General Comments

I strive to write clean and well-documented code, following patterns in the codebase and improving other code when necessary. I make an effort to document functions and classes and be clear on the necessity as well as why of those functions. I try to make code future-proof so it can evolve with changing requirements while staying focused on the problem at-hand and not add excess or unused code. I always write my functions to increase testability, and test functionality, or to catch future problems, such as ensuring a test will fail if a function can't handle a new enumItem.

For instance, when writing logic to validate donation amounts don't exceed the FEC's contribution limits invovled many well-tested submethods.

13. Domain Expertise - Mastery

Developing: "- Demonstrates growing knowledge of key Quorum concepts and technical knowledge. - Seeks out opportunities to improve base skills and demonstrates ability to pick up new technical skills. - Focuses on growing their knowledge of Quorum's systems and technical skill set broadly. - Seeks support from other engineers to learn more."

Proficient: "- Demonstrates strong knowledge of core Quorum concepts and technical knowledge with developing mastery in several areas. - Has learned how to gain additional knowledge about Quorum products and external technical skills. - Often demonstrates strong product knowledge in decision making. - Continues to develop their skills intentionally. - Has in-depth knowledge of immediate systems they've worked on and some knowledge of adjacent systems. "

Advanced: "- Demonstrates high levels of Quorum product and technical knowledge with mastery in many areas. - Has strong, well-founded opinions about how to build software in their respective domain, but is adaptable and open to new ideas. - Continues to develop skills over time, focusing on both breadth and depth of technical knowledge. - Consistently demonstrates strong product knowledge in decision making. - Understands what tools, best practices and relevant industry trends are available and effectively applies them to projects when useful and appropriate. - Teaches teammates that may know less about their domain in a way that is understandable and not condescending. - May occasionally contribute to the technology ecosystem at large."

Excellent: "- Demonstrates high levels of Quorum product and technical knowledge with mastery in most areas - Consistently shares knowledge and seeks opportunities to teach others about their particular domain. - Thoughtfully and practically introduces concepts and technologies from their domain to solve important problems, and empowers teammates to learn and improve on these concepts. - Work based on their expertise is consistently proven lasting and successful. - Almost always demonstrates strong product knowledge in decision making. - Uses expertise to improve Quorum's capabilities in their domain. - Often contributes to the domain ecosystem at large. "

14. Domain Expertise - General Comments

I've taken an effort to gain experience in as many parts of Quorum as I can, including features like Outbox, Bulk Upload or Profiles that may not be my primary focus to work on. Through PAC I've touched many of these systems, including modifying contact profiles and altering bulk upload functionality. During on calls this wider knowledge has let me more quickly make decisions on whether something is a bug or where bugs are coming from, for instance knowing if a complicated search query is returning expected results.

15. Self-Organization - Mastery

Developing: "- Often completes small projects with some support or oversight. - Spends time researching a problem before looking to others for support. - Pays attention to how they spend their time, and seeks feedback from others on how to spend time well. - Occasionally goes down ""rabbit holes"" but attempts to identify cases where that occurs to avoid future recurrence."

Proficient: "- Often completes medium sized projects with little support or oversight. - Demonstrates willingness and ability to research and solve problems independently. - Spends time well and balances between major projects, maintenance, team tasks, and other responsibilities. - Thinks proactively about how to spend their time well, and requests support when necessary."

Advanced: "- Consistently completes large projects with little support and oversight - Can take a complex problem, break it down into tasks, and complete those tasks with relative ease - Ensures commitments are realistic, understands priority and urgency of tasks, and delivers on them accordingly - Consistently allocates their time effectively and efficiently - Proactively plans their time in well advance and correctly predicts time expenditure "

Excellent: "- Requires little oversight beyond high-level direction in all projects - Leverages their understanding of business value to spend time where valuable - Almost always allocates their time effectively and efficiently - Consistently plans their time well in advance and correctly predicts time expenditure - Goes beyond knocking tasks off a list by identifing and suggesting areas of future work or systems to save their and their team's time in the future."

16. Self-Organization - General Comments

When deciding how to spend my time I use indivisual organization tools, indivisual instinct and the input of my team during standups and check-ins. For PAC, I've used our tri-weekly checkins to communicate what I plan to work on and seek feedback if it's not clear which is a priority or what may be blocking other's so I can do those first. I've also had to balance my Jira work with helping Akshat on technical, knowing when to stop my work when Akshat needs immediate help.

When given new problems I research the problem independently before asking for help, but will not waste time if I know someone else is able to help. I regularly break complex tasks in Jira, linking dependent tickets to keep track of indivisual parts.

I use a personal notebook to keep my own notes on what I'm working on, who I've promised tasks to and due dates. I keep this updated in the morning and regularly jot things down during meetings.

17. Communication - Mastery

Developing: "- Demonstrates desire and ability to communicate candidly, accurately, concisely, and regularly - Understands when to keep investigating and when to escalate problems when blocked - Asks for more context when appropriate - Occasionally considers relevant stakeholders and asks for guidance on how to engage them."

Proficient: "- Consistently communicates candidly, accurately, concisely, and regularly. - Collaborates well with team members as both a mentor and a mentee. - Takes in vague requirements and asks the right questions to ensure they are clarified. - Often escalates problems appropriately and considers relevant stakeholders. - Actively listens and seeks input from others to fully understand a problem and give thoughtful responses. - Understands when and how to appropriately and effectively offer feedback. - Seeks out and receives constructive criticism well. "

Advanced: "- Almost always communicates candidly, accurately, concisely, and regularly. - Consistently escalates problems quickly and keeps all stakeholders in the loop. - Facilitates discussion within their team, ensuring everyone has an opportunity to share their opinions and be heard, and that discussion outcomes and expectations tie to business goals. - Quickly extracts core issues from discussions and meetings to make them more productive. - Collaborates effectively with teammates and others outside them team. - Clearly and effectively gives and receives constructive feedback. - Actively and empathetically listens to ensure everyone gets a chance to share their thoughts and feels that they're heard."

Excellent: "- Almost always communicates candidly, accurately, concisely, and regularly. - Almost always escalates problems quickly and keeps all stakeholders in the loop - Facilitates conversations with disparate groups of people to help them collaborate, identify common goals, and reach consensus. Ensures everyone has an opportunity to share their opinions and be heard, and that discussion outcomes tie to business goals. Guides discussion towards decisions and gets buy-in. - Adapts their language to meet the needs of various levels of technical and non-technical audiences. - Clearly and effectively gives and receives constructive feedback. - Has others seek them out for advice on communication and for help giving difficult feedback. - Fosters a culture of effective communication among their team and the company at large."

18. Communication - General Comments

I regularly communicate with other team members about tickets, deadlines, work blockers and technical problems. I strive to give productive feedback, and communicate where I disagree while being open to changing my own opinion, and without getting too invested in my disagreement acknowledging when

things need to move forward.

For instance, when reviewing other's code I think critically and communicate problems that may occur, while being open to discussion on other solutions.

I take criticism seriously and use the feedback to improve my work and how I communicate with others. For instance, taking private Lattice feedback and doing more of the positive and less of the negative aspects given.

19. Open to Individual Growth - Mastery

Developing: "Occasionally demonstrates an interest in improving skills relevant to their job, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Occasionally acts on feedback. Does not regularly seek out feedback. Occasionally seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job. Occasionally embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Has not begun building a network of external advisors and mentors."

Proficient: "Somewhat regularly demonstrates an interest in improving skills relevant to their job, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Somewhat regularly seeks out feedback and somewhat regularly acts on that feedback. Somewhat regularly seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job. Somewhat regularly embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Is working on building a network of external advisors and mentors."

Advanced: "Regularly demonstrates a personal commitment to improving skills relevant to their job and to their future career growth at Quorum, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Regularly seeks out feedback, including from other team members, and regularly acts on that feedback. Regularly proactively seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job and somewhat regularly uses them to suggest or implement improvements at Quorum. Regularly embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Is working on building or has built a network of external advisors and mentors."

Excellent: "Almost always demonstrates a strong, personal commitment to improving skills relevant to their job and to their future career growth at Quorum, which may be identified in Bi-Annual Check-Ins, in Individual Development Plans, and through coaching. Almost always seeks out feedback, including from other team members, and regularly acts on that feedback. Almost always proactively seeks outside resources, such as books, blogs, newsletters, events, and conferences, that expand their knowledge and skills relevant to their job and regularly uses them to suggest or implement improvements at Quorum. Almost always embodies and acts on Quorum's Company Values (Invest in People, Take the Lead, Own the Execution, Embrace the Rumble, and Build Something You're Proud Of) Has built and continues to cultivate a network of external advisors and mentors and uses them as resources."

20. Open to Individual Growth - General Comments

I always look to improve myself professionally on technical and personal goals. I try to keep up to date on technologies, even if they are not directly to my work by, for instance, following and reading the LWN.net blog, and following tens of other programming blogs, like Netflix's and Instagram's Medium blogs. In personal time I host and play with Linux servers, and do my own programming. I try to bring this experience back to work, for instance my development files would not make me as productive if I didn't have some personal experience writing shell scripts.

21. Supports Quorum's Growth - Mastery

Developing: "Occasionally demonstrates an interest in building trust with individuals across Quorum, such as by offering to support other team members. Occasionally executes on quarterly OKRs. Occasionally participates in department- and team-wide conversations."

Proficient: "Somewhat regularly demonstrates an interest in building trust with individuals across Quorum, such as by offering to support other team members. Somewhat regularly executes on quarterly OKRs in a timely manner. Somewhat regularly participates in department- and team-wide conversations. Somewhat regularly sets an example for continuous improvement. Occasionaly demonstrates an interest in growing the Quorum team by conducting interviews, creating a high-quality candidate experience, and submitting scorecards in a timely manner. "

Advanced: "Regularly demonstrates a personal investment in building trust with individuals across Quorum, such as by offering to support other team members. Regularly executes on quarterly OKRs in a high-quality and timely manner. Regularly participates in department- and team-wide conversations and somewhat regularly initiates important conversations. Regularly sets an example for continuous improvement that colleagues seek to emulate. Somewhat regularly takes the lead on projects of importance that benefit Quorum, such as company-wide OKRs and other priorities. Regularly demonstrates an interest in growing the Quorum team by conducting interviews, creating a high-quality candidate experience, and submitting scorecards in a timely manner. "

Excellent: "Almost always demonstrates a personal investment in building trust with individuals across Quorum, such as by offering to support other team members, including outside their own department. Almost always executes on quarterly OKRs in a high-quality and timely manner. Almost always participates in department- and team-wide conversations and regularly initiates important conversations. Almost always sets an example for continuous improvement that colleagues seek to emulate. Regularly takes the lead and owns the execution on projects of importance that benefit Quorum, such as company-wide OKRs and other priorities. Almost always demonstrates an interest in growing the Quorum team by conducting interviews, creating a high-quality candidate experience, and submitting scorecards in a timely manner. "

22. Supports Quorum's Growth - General Comments

I support Quorum's growth with my project and bug fixing work and regularly attend company-wide events and talk to people inside and outside the development team. I communicate with CSMs and Sales about the impact of bugs. I look to people accross the company as examples and try to be an example for others.

23. During this Biannual Check-In Cycle, would you like to be considered for a promotion in your current role?

For this next section, review the career ladder to your role or any other role you would like to consider in your career progression. We are asking team members to self-nominate to begin the dialogue with their managers. This will help guide the conversation to ensure we offer appropriate feedback and resources.
Similar to the Skill Matrix, please review to see if you meet all the qualifications for the role you want to consider. And take the opportunity to offer your reasoning using the comment section below.

MULTIPLE CHOICE

- A Yes, I am ready to be considered for a promotion in my current role
- B No, I am still adequately challenged and enjoying my development in my current role
- ✓ **Maybe, I'd like to talk about what my options are moving forward**

24. What do you want your next position at Quorum to be? How would your responsibilities change?

I aim to move into a SWE II role when ready for my next promotion. I want to be recognized for more technical knowledge and skill by being trusted with bigger and more complicated projects. I want to execute on large project, and be someone others can go for help with their own decision-making.

25. What type of career growth is most important to you? (additional responsibility, leading a team, promotion, etc)

I'm primarily looking for additional responsibility, and would like to be put in positions involving where I can be more impactiful on the team. I would like to be given responsibilities to put me on track for a SWE2.

26. I foresee myself working here one year from now.

Please rate the statement above.

1	2	3	4	5
Strongly Disagree	Disagree	Neutral	Agree	**Strong Agree**

No written response

27. I believe I can achieve my full potential here at Quorum.

Please rate the statement above.

1	2	3	4	5
Strongly Disagree	Disagree	Neutral	Agree	**Strong Agree**

No written response

28. I am excited by the work and opportunities available at Quorum.

Please rate the statement above.

1	2	3	4	5
Strongly Disagree	Disagree	Neutral	Agree	**Strong Agree**

No written response

29. What most excites you about being at Quorum moving forward?

I'm excited by the oppertunity to learn and expand my skillset as Quorum grows into new ane exciting projects.

30. Do you have any concerns moving forward?

We seem to lack a lot of more senior engineers that can provide the knowledge to build Quorum and the learning and growth of others. Since our most senior engineers have gained so much of that experience at Quorum we might be missing out on better ways to do things. Seeing a lot really great people leave or are about to leave exacerbates this, as our "go-to person" for certain questions disappears. It would be meaningful to see some highly experienced technical non-manager roles open up, beyond replacing team members with those of the same level.