Postlight

IC Track   **Management Track**

	Associate	Mid	Senior	Lead (IC)	Principal	Lead (Mgmt)	Associate Director	Director
Strategy Ability to take multiple inputs and identify the right problems to solve.								
Systems Thinking Ability to envision, evaluate, and build complex platforms.								
Time Management Ability to estimate work and manage your own time.								
Communication Ability to deliver the right message with the right level of detail to the right audience.								
Collaboration Ability to properly empower and leverage the right people to move a piece of work forward.								
Growth Mindset Ability to understand and own your development, along with helping others do the same.								

Postlight – Engineering Growth Framework

	Associate Engineer	Engineer	Senior Engineer	Lead Engineer	Principal Engineer
	IC (Individual Contributor) and Management			**IC**	
Technical Ability Ability to write, rewrite, and study code.	**[Programming]** Follows Postlight and client code conventions while solving clearly-delineated problems. Gets up to speed on projects with guidance from others. Assists in delivering code to clients.	**[Programming]** Follows Postlight and client code conventions while solving ambiguous problems. Gets up to speed on existing projects with minimal oversight. Assists in delivering code to clients.	**[Programming]** Helps determine Postlight and client code conventions while solving large and ambiguous problems. Gets up to speed on projects with no oversight and guides others in doing so. Coordinates delivering code to clients.	**[Programming]** Determines Postlight and client code conventions while solving complex, opaque problems. Quickly gets up to speed on projects with no oversight and helps codify doing so for others. Coordinates delivering code and infrastructure to clients.	**[Programming]** Determines Postlight and client code conventions while quickly solving complex, opaque problems. Translates client needs into clearly-defined platforms, builds those platforms, and sets up others to do the same. Comfortable managing the entire technical lifecycle of a project, from kickoff to delivery.
	[Refactoring] Helps identify technical debt. Reorganizes code at the component level. Undertakes riskier refactors with assistance.	**[Refactoring]** Identifies technical debt. Reorganizes code involving several components. Evaluates refactor risk and seeks assistance when risk is high.	**[Refactoring]** Identifies and proposes fixes for technical debt. Reorganizes code at the application level. Evaluates refactor risk on behalf of the team, and seeks to understand when to undertake refactors.	**[Refactoring]** Identifies and proposes fixes for technical debt. Reorganizes code at the project level. Guides larger refactors undertaken by team members, and advises the client and team when risk is too high.	**[Refactoring]** Identifies, prioritizes, and proposes fixes for technical debt. Reorganizes code at the organization level. Guides larger refactors undertaken by team members, and advises the client and team when risk is too high. Introduces new methodologies for code management to Postlight.
	[Breadth of Knowledge] Learns new technologies with a guide. Seeks to build familiarity with technologies that are commonly used at Postlight.	**[Breadth of Knowledge]** Learns new technologies after some exposure. Familiar with technologies commonly used at Postlight. Shares that knowledge with others.	**[Breadth of Knowledge]** Learns new technologies quickly. Familiar with common Postlight technologies and with some specialized languages or frameworks. Able to introduce new technologies to Postlight. Regularly shares knowledge with others.	**[Breadth of Knowledge]** Becomes an authority on new technologies quickly. Familiar with a broad array of technologies, including those used at Postlight, or highly specialized in Postlight's high-need technologies. Introduces new technologies to Postlight. Mentors others.	**[Breadth of Knowledge]** Becomes an authority on new technologies quickly and teaches them to others. Combines deep expertise in Postlight's high-need technologies with a nuanced understanding of the bleeding edge across the industry. Introduces new classes of technologies to Postlight. Mentors others.

	Associate Engineer	Engineer	Senior Engineer	Lead Engineer	Principal Engineer
	IC (Individual Contributor) and Management			**IC**	
Systems Thinking Ability to envision, evaluate, and build complex platforms	**[Architecture]** Adds new components to existing projects. Understands architectural diagrams and describes the broad strokes to others. Learns why certain technologies are chosen and gives feedback on using them.	**[Architecture]** Adds new libraries to existing projects; starts small projects with familiar tools. Assists in writing and maintaining architectural diagrams. Understands why technologies are chosen and gives feedback on using them.	**[Architecture]** Adds to or starts large projects with appropriate tools. Writes and maintains architectural diagrams. Chooses technologies for projects based on experience and team feedback.	**[Architecture]** Adds to or starts large projects that may require many different coordinated toolsets. Plans, writes, and maintains these architectures to minimize surprises. Chooses technologies for projects based on experience, team feedback, and a thorough evaluation of the problem space.	**[Arcitecture]** Adds to or starts projects at scales new to Postlight. Plans, writes, and maintains these architectures to eliminate surprises. Chooses technologies for projects based on deep experience, team feedback, and a thorough evaluation of the problem space. Introduces new architectural approaches to Postlight.
	[Evaluation] Can "gut check" code for strengths and weaknesses. Solicits feedback on technical decisions and seeks to improve implementation planning. Gives some targeted impressions on how to incrementally improve existing systems.	**[Evaluation]** Evaluates single-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for the duration of the project. Advises on how to incrementally improve existing systems.	**[Evaluation]** Evaluates large or multi-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for years after project handoff. Advises on how to substantially overhaul existing systems.	**[Evaluation]** Evaluates large, theoretical, and/or multi-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for a product's major-version lifetime. Advises on how to substantially overhaul existing systems while keeping what works.	**[Evaluation]** Evaluates very large, theoretical, and/or multi-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for a product's lifetime. Advises on how to substantially overhaul existing systems while keeping what works. Able to make evaluations based on incomplete information, technical interviews with clients, or limited access to code.

	Associate Engineer	Engineer	Senior Engineer	Lead Engineer	Principal Engineer
	IC (Individual Contributor) and Management			**IC**	
Time Management Time Management Ability to estimate work and manage your own time.	**[Estimation & Prioritization]** Estimates their own velocity with a large margin of error. Seeks assistance in prioritizing work.	**[Estimation & Prioritization]** Estimates their own velocity with a small margin of error. Prioritizes work with minimal assistance; separates and orders code changes such that wasted code is minimized.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error. Prioritizes work without assistance; separates and orders code changes such that wasted code is minimized, and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a very small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same. Comfortable providing timelines for a project's overall engineering effort and assisting Strat in a project's pre-development process.
	[Self-Direction] Prefers daily check-ins to stay on task. Relies on the PM to drive the project forward. Excels when assigned a single task at a time.	**[Self-Direction]** Stays on task if check-ins are missed or happen asynchronously. Works with the PM to drive the project forward. Switches between tasks when needed at the cost of focus and velocity.	**[Self-Direction]** Stays on task even if alone on a project. Able to drive a project forward when a PM is absent for short periods of time. Switches between tasks or projects when needed at a small cost of focus and velocity.	**[Self-Direction]** Stays on task when alone on a project and helps others stay on task when on a team. Able to sub for a PM as needed. Switches between tasks or projects when needed with minimal impact on focus and velocity.	**[Self-Direction]** Stays on task when alone on a project for indefinite amounts of time and helps others stay on task when on a team. Able to sub for a PM as needed. Switches between tasks or projects when needed without impact on focus and velocity.

	Associate Engineer	Engineer	Senior Engineer	Lead Engineer	Principal Engineer
	IC (Individual Contributor) and Management			**IC**	
Comm. Ability to deliver the right message with the right level of detail to the right audience.	**[Listening & Responding]** Answers well-scoped technical questions clearly and in terms the PM or client can understand. Raises own blockers to the team with minimal time lost being blocked. Stays aware of others' day-to-day work.	**[Listening & Responding]** Answers most technical questions clearly and in terms the PM or client can understand. Raises own blockers to the team quickly. Stays aware of others' work enough to sub in mid-ticket if necessary.	**[Listening & Responding]** Answers technical questions clearly and in terms the PM or client can understand. Raises own blockers to the team quickly, and directly addresses others' blockers during standups. Tracks the team's progress, checking in on team members that may be blocked or sidetracked.	**[Listening & Responding]** Answers technical questions clearly and in the PM's or client's own words. Anticipates future technical questions. Raises own and addresses others' blockers quickly. Tracks the team's progress, checking in on team members that may be blocked or sidetracked.	**[Listening & Responding]** Answers technical questions clearly and in the PM's or client's own words. Anticipates future technical questions. Raises own and addresses others' blockers quickly. Tracks the team's progress, checking in on team members that may be blocked or sidetracked. Comfortable providing precise technical questions for clients during project planning and discovery. Translates client needs into technical requirements as needed.
	[Technical Communication] Adds comments on complex or confusing pieces of code. Writes ready-for-work tickets with the PM's assistance. Opens pull requests that require minimal questions from reviewers to be traceable.	**[Technical Communication]** Adds enough comments to make code readable throughout an application. Writes ready-for-work tickets with occasional PM input. Opens pull requests with enough description and comments to be traceable.	**[Technical Communication]** Documents code thoroughly with comments, generated docs, and READMEs. Writes ready-for-work tickets and helps others refine tickets. Opens pull requests with description and comments that preempt reviewer confusion. Presents technical concepts to internal and external audiences.	**[Technical Communication]** Documents code with comments, generated docs, and READMEs such that it can be handed off at any time. Provides the PM with architecture diagrams. Writes ready-for-work tickets and helps the PM and others refine tickets. Opens pull requests with description and comments that aim to teach the reviewer why the approach was taken. Presents technical concepts to internal or external audiences in high-stakes settings.	**[Technical Communication]** Documents code with comments, generated docs, and READMEs such that it can be handed off at any time. Provides the PM with architecture diagrams. Writes ready-for-work tickets and helps the PM and others refine tickets. Opens pull requests with description and comments that aim to teach the reviewer why the approach was taken. Comfortable providing technical documentation to assist Strat in a project's pre-development process. Presents technical concepts to internal or external audiences in high-stakes settings. Represents engineering at Postlight in the industry.

	Associate Engineer	Engineer	Senior Engineer	Lead Engineer	Principal Engineer
	IC (Individual Contributor) and Management			**IC**	
Collab. Ability to properly empower and leverage the right people to move a piece of work forward.	**[Enablement]** Helps create an environment where the team is productive. Brings Engineering's perspective to the conversation and seeks out feedback from strat, product, design, and engineering to improve how the team works together.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Uses the feedback to improve the team's dynamic.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Consistently improves the team's dynamics based on the feedback and considers ways to improve Postlight overall.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Consistently improves the team's dynamics based on the feedback and brings best practices to improve Postlight overall.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Consistently improves the team's dynamics based on the feedback, brings best practices, and implements systems of measurements that improve Postlight overall.
	[Design & Product Fluency] Relies on expertise of design and product. Demonstrates ability to learn about design and product concepts.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to gut check decisions.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to appropriately challenge decisions and propose alternative ideas.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to appropriately challenge decisions and propose alternative ideas.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to appropriately challenge decisions and propose alternative ideas.
Growth Mindset Ability to understand and own your development, along with helping others do the same.	**[Awareness]** Actively solicits feedback from team, always assuming positive intent. Receives guidance on growth opportunities.	**[Awareness]** Actively solicits feedback from team, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities.	**[Awareness]** Actively solicits feedback from team and leadership, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities as well as those of people they directly interact with.	**[Awareness]** Actively solicits feedback from team and leadership, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities. Able to identify strengths, weaknesses, and growth opportunities that span the department.	**[Awareness]** Actively solicits feedback from team and leadership, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities. Able to identify strengths, weaknesses, and growth opportunities that span the department.
	[Development] Leverages professional development and other resources to improve.	**[Development]** Leverages professional development and other resources to improve. Shares individual learnings with the broader team.	**[Development]** Leverages professional development and other resources to improve beyond current responsibilities. Improves the dynamics and overall expertise of the team by mentoring, influencing, and leading where appropriate.	**[Development]** Leverages professional development and other resources to improve beyond current responsibilities. Actively manages the dynamics and overall expertise of the team by mentoring, influencing, and leading where appropriate.	**[Development]** Leverages professional development and other resources to improve beyond current responsibilities. Actively manages the dynamics and overall expertise of the team by mentoring, influencing, and leading where appropriate.

	Engineer	Senior Engineer	Lead Engineer	Associate Director, Engineering	Director Engineering
	IC (Individual Contributor) and Management			**IC**	
Technical Ability Ability to write, rewrite, and study code.	**[Programming]** Follows Postlight and client code conventions while solving ambiguous problems. Gets up to speed on existing projects with minimal oversight. Assists in delivering code to clients.	**[Programming]** Helps determine Postlight and client code conventions while solving large and ambiguous problems. Gets up to speed on projects with no oversight and guides others in doing so. Coordinates delivering code to clients.	**[Programming]** Determines Postlight and client code conventions while solving complex, opaque problems. Quickly gets up to speed on projects with no oversight and helps codify doing so for others. Coordinates delivering code and infrastructure to clients.	**[Programming]** Solves complex, opaque problems; quickly gets up to speed on projects with no oversight; helps others do so as well. Coordinates delivering code and infrastructure to clients. Considers the overall programming trends of the organization against the industry and helps guide those trends accordingly.	**[Programming]** Solves complex, opaque problems; quickly gets up to speed on projects with no oversight; helps others do so as well. Coordinates delivering code and infrastructure to clients. Closely tracks the overall programming trends of the organization against the industry and guides those trends accordingly.
	[Refactoring] Identifies technical debt. Reorganizes code involving several components. Evaluates refactor risk and seeks assistance when risk is high.	**[Refactoring]** Identifies and proposes fixes for technical debt. Reorganizes code at the application level. Evaluates refactor risk on behalf of the team and seeks to understand when to undertake refactors.	**[Refactoring]** Identifies and proposes fixes for technical debt. Reorganizes code at the project level. Guides larger refactors undertaken by team members and advises the client and team when risk is too high.	**[Refactoring]** Identifies, prioritizes, and proposes fixes for technical debt. Reorganizes code at the organization level. Guides larger refactors undertaken by team members and advises the client and team when risk is too high.	**[Refactoring]** Identifies, prioritizes, and proposes fixes for technical debt. Reorganizes code at the organization level. Guides larger refactors undertaken by team members and advises the client and team when risk is too high.
	[Breadth of Knowledge] Learns new technologies after some exposure. Familiar with technologies commonly used at Postlight. Shares that knowledge with others.	**[Breadth of Knowledge]** Learns new technologies quickly. Familiar with common Postlight technologies and with some specialized languages or frameworks. Able to introduce new technologies to Postlight. Regularly shares knowledge with others.	**[Breadth of Knowledge]** Becomes an authority on new technologies quickly. Familiar with a broad array of technologies, including those used at Postlight, or highly specialized in Postlight's high-need technologies. Introduces new technologies to Postlight. Mentors others.	**[Breadth of Knowledge]** Becomes an authority on new technologies quickly and teaches them to others. Combines deep expertise in Postlight's high-need technologies with a nuanced understanding of the bleeding edge across the industry. Looks for opportunities to expand others' expertise. Helps develop and maintain systems to encourage knowledge-sharing within the organization.	**[Breadth of Knowledge]** Becomes an authority on new technologies quickly and teaches them to others. Combines deep expertise in Postlight's high-need technologies with a nuanced understanding of the bleeding edge across the industry. Expands others' expertise. Develops and maintains systems to encourage knowledge-sharing within the organization.

	Engineer	Senior Engineer	Lead Engineer	Associate Director, Engineering	Director Engineering
	IC (Individual Contributor) and Management			**IC**	
Systems Thinking Ability to envision, evaluate, and build complex platforms	**[Architecture]** Adds new libraries to existing projects; starts small projects with familiar tools. Assists in writing and maintaining architectural diagrams. Understands why technologies are chosen and gives feedback on using them.	**[Architecture]** Adds to or starts large projects with appropriate tools. Writes and maintains architectural diagrams. Chooses technologies for projects based on experience and team feedback.	**[Architecture]** Adds to or starts large projects that may require many different coordinated toolsets. Plans, writes, and maintains these architectures to minimize surprises. Chooses technologies for projects based on experience, team feedback, and a thorough evaluation of the problem space.	**[Architecture]** Adds to or starts large projects that may require many different coordinated toolsets. Plans, writes, and maintains these architectures to minimize surprises. Chooses technologies for projects based on experience, team feedback, and a thorough evaluation of the problem space.	**[Architecture]** Adds to or starts very large-scale projects. Plans, writes, and maintains these architectures to eliminate surprises. Chooses technologies for projects based on deep experience, team feedback, and a thorough evaluation of the problem space. Vets new architectural approaches and centralizes techniques across the organization.
	[Evaluation] Evaluates single-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for the duration of the project. Advises on how to incrementally improve existing systems.	**[Evaluation]** Evaluates large or multi-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for years after project handoff. Advises on how to substantially overhaul existing systems.	**[Evaluation]** Evaluates large, theoretical, and/or multi-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for a product's major-version lifetime. Advises on how to substantially overhaul existing systems while keeping what works.	**[Evaluation]** Evaluates large, theoretical, and/or multi-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for a product's major-version lifetime. Advises on how to substantially overhaul existing systems while keeping what works.	**[Evaluation]** Evaluates very large, theoretical, and/or multi-application codebases for strengths and weaknesses. Makes technical decisions that remain reasonably correct for a product's lifetime. Advises on how to substantially overhaul existing systems while keeping what works. Able to make evaluations based on incomplete information, technical interviews with clients, or limited access to code.
	[Architecture] Adds new libraries to existing projects; starts small projects with familiar tools. Assists in writing and maintaining architectural diagrams. Understands why technologies are chosen and gives feedback on using them.	**[Architecture]** Adds to or starts large projects with appropriate tools. Writes and maintains architectural diagrams. Chooses technologies for projects based on experience and team feedback.	**[Architecture]** Adds to or starts large projects that may require many different coordinated toolsets. Plans, writes, and maintains these architectures to minimize surprises. Chooses technologies for projects based on experience, team feedback, and a thorough evaluation of the problem space.	**[Architecture]** Adds to or starts large projects that may require many different coordinated toolsets. Plans, writes, and maintains these architectures to minimize surprises. Chooses technologies for projects based on experience, team feedback, and a thorough evaluation of the problem space.	**[Architecture]** Adds to or starts very large-scale projects. Plans, writes, and maintains these architectures to eliminate surprises. Chooses technologies for projects based on deep experience, team feedback, and a thorough evaluation of the problem space. Vets new architectural approaches and centralizes techniques across the organization.

	Engineer	Senior Engineer	Lead Engineer	Associate Director, Engineering	Director Engineering
	IC (Individual Contributor) and Management			**IC**	
Time Management Time Management Ability to estimate work and manage your own time.	**[Estimation & Prioritization]** Estimates their own velocity with a small margin of error. Prioritizes work with minimal assistance; separates and orders code changes such that wasted code is minimized.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error. Prioritizes work without assistance; separates and orders code changes such that wasted code is minimized, and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a very small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same. Comfortable providing timelines for a project's overall engineering effort and assisting Strat in a project's pre-development process.
	[Self-Direction] Stays on task if check-ins are missed or happen asynchronously. Works with the PM to drive the project forward. Switches between tasks when needed at the cost of focus and velocity.	**[Self-Direction]** Stays on task even if alone on a project. Able to drive a project forward when a PM is absent for short periods of time. Switches between tasks or projects when needed at a small cost of focus and velocity.	**[Self-Direction]** Stays on task when alone on a project and helps others stay on task when on a team. Able to sub for a PM as needed. Switches between tasks or projects when needed with minimal impact on focus and velocity.	**[Self-Direction]** Stays on task when alone on a project and helps others stay on task, whether as teammates or direct reports. Able to sub for a PM as needed. Switches between tasks or projects when needed with minimal impact on focus and velocity.	**[Self-Direction]** Stays on task when alone on a project and helps others stay on task, whether as teammates, direct reports, or the department. Able to sub for a PM as needed. Switches between tasks or projects when needed with minimal impact on focus and velocity.
	[Estimation & Prioritization] Estimates their own velocity with a small margin of error. Prioritizes work with minimal assistance; separates and orders code changes such that wasted code is minimized.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error. Prioritizes work without assistance; separates and orders code changes such that wasted code is minimized, and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same.	**[Estimation & Prioritization]** Estimates their own and their team's velocity with a very small margin of error; able to prioritize and distribute work to others based on these estimates. Juggles in-flight refactors without wasting code and helps others do the same. Comfortable providing timelines for a project's overall engineering effort and assisting Strat in a project's pre-development process.

	Engineer	Senior Engineer	Lead Engineer	Associate Director, Engineering	Director Engineering
	IC (Individual Contributor) and Management			**IC**	
Comm. Ability to deliver the right message with the right level of detail to the right audience.	**[Listening & Responding]** Answers most technical questions clearly and in terms the PM or client can understand. Raises own blockers to the team quickly. Stays aware of others' work enough to sub in mid-ticket if necessary.	**[Listening & Responding]** Answers technical questions clearly and in terms the PM or client can understand. Raises own blockers to the team quickly and directly addresses others' blockers during standups. Tracks the team's progress, checking in on team members that may be blocked or sidetracked.	**[Listening & Responding]** Answers technical questions clearly and in the PM's or client's own words. Anticipates future technical questions. Raises own and addresses others' blockers quickly. Tracks the team's progress, checking in on team members that may be blocked or sidetracked.	**[Listening & Responding]** Answers technical questions clearly and in the PM's or client's own words. Anticipates future technical questions. Raises own and addresses others' blockers quickly. Tracks the team's progress, checking in on team members and direct reports that may be blocked or sidetracked.	**[Listening & Responding]** Answers technical questions clearly and in the PM's or client's own words. Anticipates future technical questions. Raises own and addresses others' blockers quickly. Tracks the team's progress, checking in on team members and direct reports that may be blocked or sidetracked.
	[Technical Communication] Adds enough comments to make code readable throughout an application. Writes ready-for-work tickets with occasional PM input. Opens pull requests with enough description and comments to be traceable.	**[Technical Communication]** Documents code thoroughly with comments, generated docs, and READMEs. Writes ready-for-work tickets and helps others refine tickets. Opens pull requests with description and comments that preempt reviewer confusion. Presents technical concepts to internal and external audiences.	**[Technical Communication]** Documents code with comments, generated docs, and READMEs such that it can be handed off at any time. Provides the PM with architecture diagrams. Writes ready-for-work tickets and helps the PM and others refine tickets. Opens pull requests with description and comments that aim to teach the reviewer why the approach was taken. Presents technical concepts to internal or external audiences in high-stakes settings.	**[Technical Communication]** Documents code with comments, generated docs, and READMEs such that it can be handed off at any time. Provides the PM with architecture diagrams. Writes ready-for-work tickets and helps the PM and others refine tickets. Opens pull requests with description and comments that aim to teach the reviewer why the approach was taken. Comfortable providing technical documentation to assist Strat in a project's pre-development process. Teaches technical concepts internally and encourages direct reports to do the same.	**[Technical Communication]** Documents code with comments, generated docs, and READMEs such that it can be handed off at any time. Provides the PM with architecture diagrams. Writes ready-for-work tickets and helps the PM and others refine tickets. Opens pull requests with description and comments that aim to teach the reviewer why the approach was taken. Comfortable providing technical documentation to assist Strat in a project's pre-development process. Teaches technical concepts internally and encourages direct reports to do the same.

	Engineer	Senior Engineer	Lead Engineer	Associate Director, Engineering	Director Engineering
	IC (Individual Contributor) and Management			**IC**	
Collab. Ability to properly empower and leverage the right people to move a piece of work forward.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Uses the feedback to improve the team's dynamic.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Consistently improves the team's dynamics based on the feedback and considers ways to improve Postlight overall.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Consistently improves the team's dynamics based on the feedback and brings best practices to improve Postlight overall.	**[Enablement]** Creates an environment where the team is productive. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Consistently improves the team's dynamics based on the feedback, brings best practices, and implements systems of measurements that improve Postlight overall.	**[Enablement]** Empowers team to do their best work by driving ownership deep into the organization. Brings Engineering's perspective to the conversation and creates a feedback loop between strat, product, design, and engineering. Consistently improves the team's dynamics based on the feedback, brings best practices, and implements systems of measurements that improve Postlight overall.
	[Design & Product Fluency] Relies on expertise of design and product, shares opinions on design and product topics, and is able to gut check decisions.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to appropriately challenge decisions and propose alternative ideas.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to appropriately challenge decisions and propose alternative ideas.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to appropriately challenge decisions and propose alternative ideas.	**[Design & Product Fluency]** Relies on expertise of design and product, shares opinions on design and product topics, and is able to appropriately challenge decisions and propose alternative ideas.

	Engineer	Senior Engineer	Lead Engineer	Associate Director, Engineering	Director Engineering
	IC (Individual Contributor) and Management			**IC**	
Growth Mindset Ability to understand and own your development, along with helping others do the same.	**[Awareness]** Actively solicits feedback from team, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities.	**[Awareness]** Actively solicits feedback from team and leadership, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities as well as those of people they directly interact with.	**[Awareness]** Actively solicits feedback from team and leadership, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities. Able to identify strengths, weaknesses, and growth opportunities that span the department.	**[Awareness]** Actively solicits feedback from team and leadership, always assuming positive intent. Understands own strengths, weaknesses, and growth opportunities. Consistently identifies strengths, weaknesses, and growth opportunities that span the department.	**[Awareness]** Actively solicits feedback from team and leadership, always assuming positive intent. Understands own strength, weaknesses, and growth opportunities. Consistently identifies strengths, weaknesses, and growth opportunities that span the organization.
	[Development] Leverages professional development and other resources to improve. Shares individual learnings with the broader team.	**[Development]** Leverages professional development and other resources to improve beyond current responsibilities. Improves the dynamics and overall expertise of the team by mentoring, influencing, and leading where appropriate.	**[Development]** Leverages professional development and other resources to improve beyond current responsibilities. Actively manages the dynamics and overall expertise of the team by mentoring, influencing, and leading where appropriate.	**[Development]** Leverages professional development and other resources to improve beyond current responsibilities. Actively manages the dynamics and overall expertise of the department by developing and managing frameworks, processes, and programs.	**[Depelopment]** Leverages professional development and other resources to improve understanding of the business beyond core areas of expertise. Actively manages the dynamics and overall expertise of the department by developing and managing frameworks, processes, and programs.
				[Talent Management] Manages a team of ICs and/or managers and holds them accountable to performance standards. Builds a high-performing department by executing on hiring and staffing plans, actively engaging in recruiting and hiring exceptional talent, fostering a culture of diversity, equity, and inclusion, fostering opportunities for high performers, celebrating success, and guiding managers through tough decisions on exiting employees.	**[Talent Management]** Manages a team of senior ICs and/or managers and holds them accountable to performance standards. Has a proven track record of building a high performing department by creating hiring and staffing plans that scale, cultivating and growing new managers, attracting and hiring exceptional talent, building a culture of diversity, equity, and inclusion, building opportunities for high performers, celebrating success, and guiding managers through tough decisions on exiting employees.

R.E. Factor
Title: Engineer

Technical Ability	Systems Thinking	Time Management	Communication	Collaboration	Growth Mindset
EFFECTIVE	**SURPASSING**	**EFFECTIVE**	**LEARNING**	**EFFECTIVE**	**EFFECTIVE**
R is great at spinning up on new projects, learning the tech required, and making code contributions in the first couple of weeks. They did solid work combining some tricky components on project ABC, and even built a state machine despite being new to the concept.	R really shines when they get to think about systems as a whole. We know that when R is staffed on a project, they're able to see how it all fits together and thus make efficient changes where immediately needed. They excel at adding new parts to a project's architecture and are a trusted voice on team ABC on when to make substantial changes, and how.	R's estimates have been fairly accurate on their last three teams, with occasional time losses that happen to everyone. When asked to pitch in on a blog post, they did have trouble context switching, but were able to get quickly back up to speed after it published.	Perhaps because R is so eager to make changes, they tend to open underdocumented pull requests that cost the rest of the team extra time spent trying to make sense of what's going on. They tend to lean on the PM of team ABC for details of technical tickets that could be inferred. However, they're great at talking to clients at the right level of detail, and are aware that commenting and PR documentation are an area worth growing.	R consistently refers back to the PM and designers on their team during implementation. They're passionate about typography in particular, and are able to catch instances in the design where it was incorrectly applied and flag them back to the designer.	R always asks for feedback in checkins, and has spent a good portion of their PD budget on courses. They're aware of their weakneses and starting to develop a feel for understanding the strengths and weaknesses of others as well.