

Structuring an Optimal Site Reliability Engineering Team



Maria Wilvenna Añora
Co-founder and Strategy Head, AtoANI

@wilvieanora



Overview



- SRE Team Structures
 - Types
 - Comparison
- Anti-pattern Traps
 - Definition
 - Types to Avoid
- Bootstrapping in SRE
 - Definition
 - Methods
 - Risks
 - Mitigation



Overview



- **Site Reliability Engineer**
 - Role Definition
 - Required Skills
- **Designing a Hiring Pipeline**
- **Team Career Progression**
 - Definition
 - Plan Creation
- **Use Case**
 - Structuring an optimal Site Reliability Engineering function



Site Reliability Engineering Team Structures



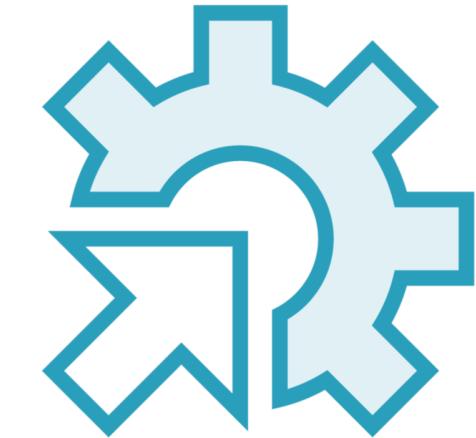
Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



A photograph of two women in profile, looking at a computer monitor. The woman on the left wears glasses and a pink sweater, while the woman on the right has red hair and a plaid shirt. The monitor displays a terminal window with multiple lines of code.

Kitchen Sink

Kitchen Sink

- First and only SRE team in existence in the organization
- Everything SRE Team
- Scope of services or workflows
 - Unbounded
 - Constantly changing



A photograph of two women looking at a computer monitor. The woman on the left is wearing glasses and a pink sweater, while the woman on the right has red hair and is wearing a plaid shirt. They are both looking intently at the screen, which displays several lines of code.

Kitchen Sink

Advantages

- No coverage gaps between SRE teams
- Easy to spot patterns and similarities
 - Services and projects
- Tends to act as glue between Dev teams
 - Creating solutions: software





Disadvantages

- Lack of an SRE team charter
 - Scope of task team covers
- Suffer from lack of clarity of team goals
 - Can lead to operational overload
- Company and system complexity grows
 - From: Having deep positive impact
 - To: Making shallow contributions
- Team issues negatively impact business





Recommended Implementation

- Company that want to start SRE journey
- Company with limited scope
 - Fewer applications
 - Result: Fewer user journeys
- Increase in adoption of SRE practices
 - Demand for dedicated SRE team
- Scope remains small enough
 - Cannot justify multiple SRE teams



Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



Infrastructure



Infrastructure

- **Core Focus:**
 - Improving reliability of supporting infrastructure that runs the service
 - Behind-the-scenes tasks
 - Help development team get job done faster and easier
- Less focused: Customer-facing code written in-house
- Maintain: Shared services, Components
 - Related to stabilizing infrastructure





Advantages

- Allows use of DevOps practices
 - Maintain user-facing products
 - Without divergence in practice across the business
- Focus on providing highly reliable infrastructure
 - Define production standards as code
 - Simplify things for developers running own services





Disadvantages

- Team issues negatively impact business
 - Depending on infrastructure scope
- Lack of direct customer contact
 - Focus on improvements not tied to customer experience
- May be required to split infrastructure teams as complexity grows





Recommended Implementation

- Several development teams
 - Varying: scope, complexities, infrastructure requirements
 - Need: common standards and practices
- Large companies
 - Infrastructure DevOps team
 - Write software for applications team
 - SRE team
 - Focus on reliability



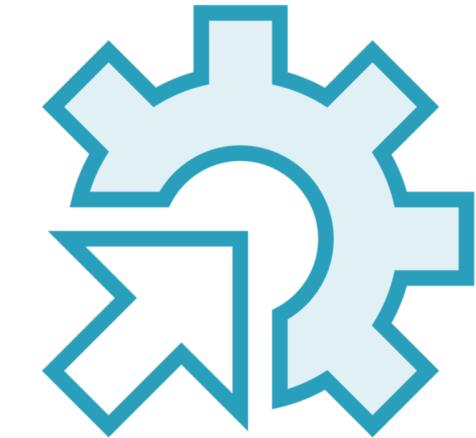
Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



A photograph showing a person's hands typing on a black keyboard. In the background, a computer monitor displays a dark-themed code editor with multiple tabs open, showing snippets of code. A silver laptop is also visible on the desk to the left.

Tools-only

Tools-only

- **Core Focus:**
 - Improving reliability of individual services
 - Support and planning systems with reliability-oriented feature set
- **Aim:**
 - Build highly specialized reliability-related tooling
 - Ultimately define production standards



A photograph of a person's hands typing on a black keyboard. In the background, a computer monitor displays a dark-themed code editor with multiple tabs open, showing lines of programming code. A silver laptop is partially visible on the left. The scene is set on a light-colored wooden desk.

Tools-only

Advantages

- Help developer counterparts
 - Measure
 - Maintain
 - Improve
 - System reliability
 - Capacity planning



Disadvantages

- Runs risk of solving wrong problems
 - Needs to work hard to stay aware of practical problems
 - Teams tackling front-line reliability
- High risk of increase
 - Toil
 - Overall workload



Tools-only



Recommended Implementation

- Company that need highly specialized reliability-related tooling



Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



Application

Application

- Team works to improve reliability
 - Critical application or business area
- Reliability of ancillary services
 - Batch processors
 - Sole responsibility of different team
 - Developers: Dev, Ops





Advantages

- Provides clear focus on team's effort
- Clear link on business priorities
 - Team effort





Disadvantages

- Potential duplication of base infrastructure
- Divergence of practices between teams
- Limits knowledge sharing and mobility





Recommended Implementation

- For second (and up) team for companies
 - Started with kitchen sink, infrastructure, tools-only
 - Have key user-facing application
 - High reliability needs
 - Justify relatively large expense
 - Dedicated set of SREs



Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



Embedded



Embedded

- Teams have site reliability engineers embedded with developer counterparts
- Work relationships tend to be project-bounded
- SREs usually very hands-on





Advantages

- Enables focused SRE expertise
 - Directed to specific problems or teams
- Allow side-by-side demonstration
 - SRE practices





Disadvantages

- May result to lack of standardization
 - Between teams
 - Divergence in practice
- May not have chance to spend time with peers for mentorship





Recommended Implementation

- Works well to either:
 - Start an SRE function
 - Scale implementation
- Project or team needs SRE for a period of time
- Augmenting impact
 - Tools-only or infrastructure team



Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



Consulting

- Similar to embedded team structure
- Difference: Teams tend to avoid changing customer code and configuration
 - Services in scope
- Write code and configuration in order
 - Build and maintain tools



Advantages

- Help with further scaling an existing SRE function's positive impact
 - Being decoupled
 - Directly changing
 - Code
 - Configuration





Disadvantages

- May lack sufficient context to offer useful advice
- Risk of being perceived as hands-off
 - Capable of indirect technical impact
 - Don't change
 - Code
 - Configuration





Recommended Implementation

- Company is large
- Demands have outgrown what can be supported by existing SRE team



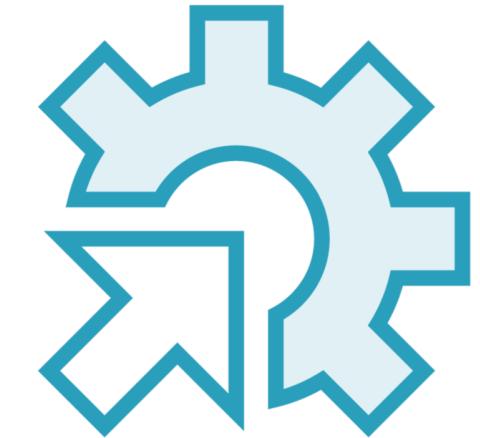
Site Reliability Engineering Team Structures



Kitchen Sink



Infrastructure



Tools-only



Application



Embedded



Consulting



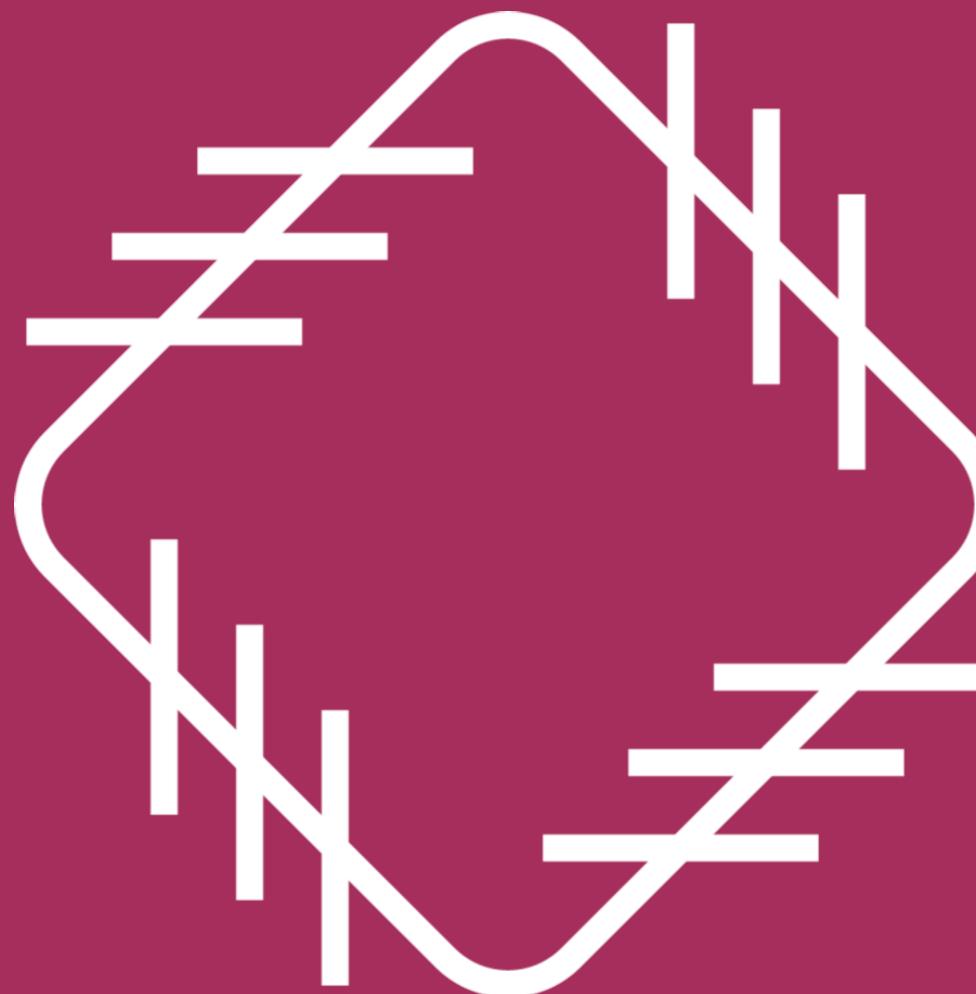
Anti-pattern Definition and Traps to Avoid



Anti-pattern

A common response to a recurring problem that is usually ineffective and risks being highly counterproductive.





Insights concerning anti-pattern

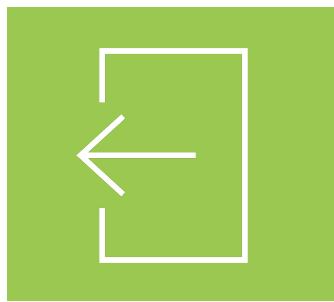
- **Users shouldn't notice an outage before the SRE team does**
- **Solution should eliminate errors**
 - Not provide point fixes
- **Failure is an opportunity to improve**
- **Human toil should not be inputted to a machine or software**



Antipattern Traps to Avoid in Team Development



Creating experts whose knowledge is compartmentalized



Treating outages as secrets to avoid blame



Giving all new project work to the most senior engineers



Not considering user expectations and planning for capacity



Antipattern Traps to Avoid in Team Development



Settling for point fixes rather than long-term fixes



Treating errors that occur as human error



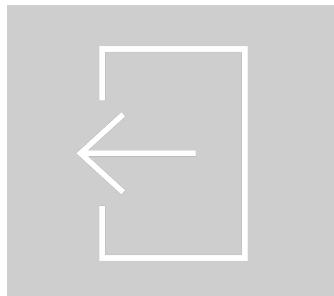
Not empowering teams to solve problems



Antipattern Traps to Avoid in Team Development



Creating experts whose knowledge is compartmentalized



Treating outages as secrets to avoid blame



Giving all new project work to the most senior engineers



Not considering user expectations and planning for capacity





Experts with compartmentalized knowledge

Creating experts on teams

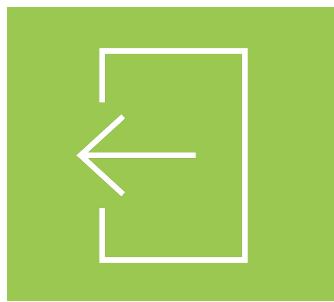
- Compartmentalized
 - Knowledge, Skills
- Risk of not having right knowledge and skills when an issue arise
 - Hinder productivity
 - Result to error
- What needs to be done?
 - Hire engineers with the right skills set
 - Train engineers to address issues:
 - Individual, Group



Antipattern Traps to Avoid in Team Development



Creating experts whose knowledge is compartmentalized



Treating outages as secrets to avoid blame



Giving all new project work to the most senior engineers



Not considering user expectations and planning for capacity



Treating outages as secrets



Treating outages as secrets to avoid blame

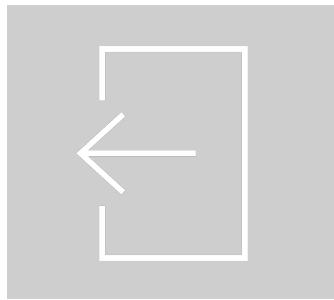
- Typical: If blamed for mistakes
 - Encourages hiding mistakes
- What needs to be done?
 - Have a blameless culture
 - Embrace and acknowledge failure
 - Learn from failure
 - Address cause of failure
 - Improve handling of incident
 - Results to more robust systems



Antipattern Traps to Avoid in Team Development



Creating experts whose knowledge is compartmentalized



Treating outages as secrets to avoid blame



Giving all new project work to the most senior engineers



Not considering user expectations and planning for capacity



Giving new project to senior engineers



Giving all new project to senior engineers

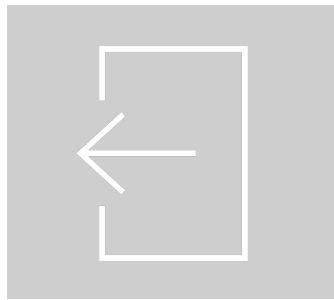
- Risk of not preparing new engineers
 - Handle SRE tasks on their own
- Work overload to senior engineers
- Limits opportunity for new engineers
 - Learning new skills
 - Career progression
- What needs to be done?
 - Let new engineers handle project work
 - With assistance, partial ownership
 - Then full ownership once ready



Antipattern Traps to Avoid in Team Development



Creating experts whose knowledge is compartmentalized



Treating outages as secrets to avoid blame



Giving all new project work to the most senior engineers



Not considering user expectations and planning for capacity



User expectations and capacity planning



Not considering user expectations and planning for capacity

- Not meeting user expectations
 - Lead to dissatisfied customers
 - Bad reputation: team, organization
- Not planning for capacity
 - Overloading with work to meet service level objectives (SLOs)
 - Leads to burnout
- What needs to be done?
 - Align set SLOs with user expectations
 - Plan for capacity aligned to SLOs



Antipattern Traps to Avoid in Team Development



Settling for point fixes rather than long-term fixes



Treating errors that occur as human error



Not empowering teams to solve problems



Settling for point fixes

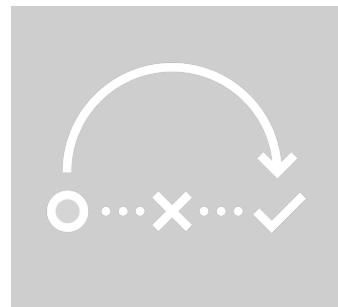


Settling for point fixes

- Being contented with short-term fixes
 - Problem root cause not addressed
 - Possible problem reoccurrence
- What needs to be done?
 - Assess problem impact then mitigate
 - Identify root cause
 - Implement a long-term fix or solution
 - Prevent the problem from reoccurring



Antipattern Traps to Avoid in Team Development



Settling for point fixes rather than long-term fixes



Treating errors that occur as human error



Not empowering teams to solve problems





Treating errors that occur as human error

Treating errors that occur as human error

- Presents risk of not solving the error
- In SRE, problems are not human errors
 - They are system problems
- What needs to be done?
 - Acknowledge that the errors that occur are due to system problems
 - Identify system-related solutions
 - Implement solutions that make systems foolproof
 - Features, Functions



Antipattern Traps to Avoid in Team Development



Settling for point fixes rather than long-term fixes



Treating errors that occur as human error



Not empowering teams to solve problems



Not empowering teams to solve problems



Not empowering teams to solve problems

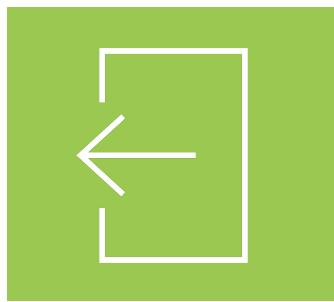
- Risk of always relying on outside help
- SRE team incapable of solving problems
 - Individualized, inefficient, unproductive
- Can demotivate engineers
- What needs to be done
 - Encourage SRE teams to solve problems
 - Together as a team
 - Creatively, eliminate judgement
 - Provide a workplace environment
 - Facilitates team problem solving



Antipattern Traps to Avoid in Team Development



Creating experts whose knowledge is compartmentalized



Treating outages as secrets to avoid blame



Giving all new project work to the most senior engineers



Not considering user expectations and planning for capacity



Antipattern Traps to Avoid in Team Development



Settling for point fixes rather than long-term fixes



Treating errors that occur as human error



Not empowering teams to solve problems



Bootstrapping Methods in SRE



Bootstrapping

A self-starting process that is supposed to continue or grow without outside assistance or external output like creating a self-sustaining system using existing resources.



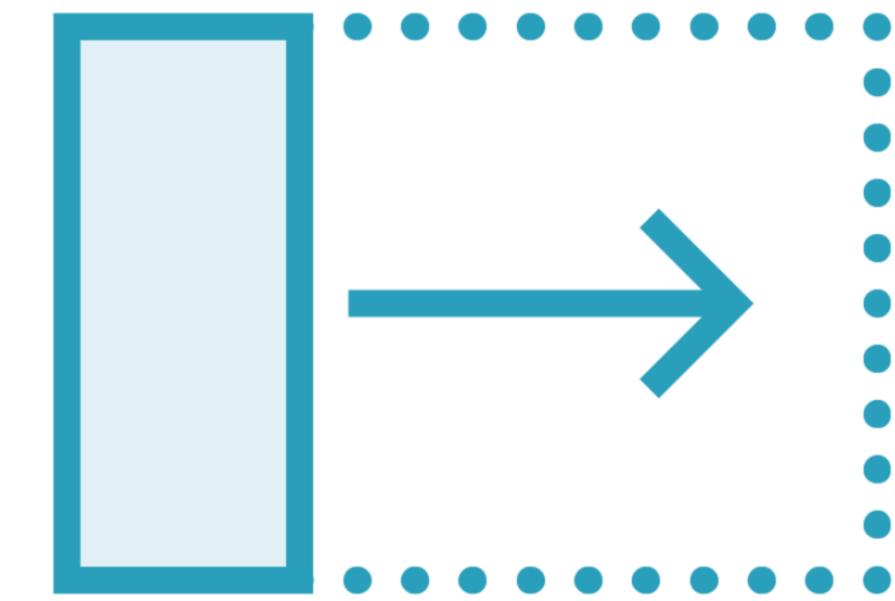
Bootstrapping Methods



**Creating a new
team as part of a
major project**



**Assembling a
horizontal SRE team**



**Converting an
existing team
to an SRE team**



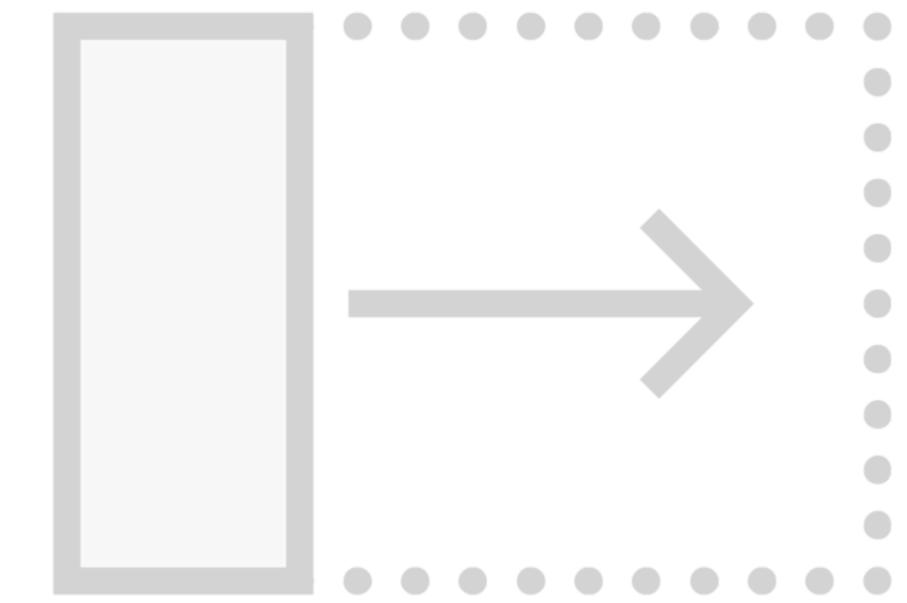
Bootstrapping Methods



**Creating a new
team as part of a
major project**



**Assembling a
horizontal SRE team**



**Converting an
existing team
to an SRE team**





Creating a new team as part of a project

Creating a new team as part of a project

- Major project
 - Large enough to justify new headcount
 - Identified project risks
 - Reliability, operational capability





Creating a new team as part of a project

Risks

- Team taking too many services at once
 - No time to address risk: permanent way
- Team underdelivers
 - Becomes too contemplative
 - Neglecting current service needs
- Team doesn't examine work thoroughly
- Abandon SRE principles and practices
 - To meet milestone requirements
- Conflict with existing teams
- Does not have necessary skills
 - Delivers only partial improvement





Creating a new team as part of a project

Mitigations

- **Engage initially on a single service**
 - **With significant impact**
- **Engage as early as possible**
 - **Have input to the design**
 - **Focus on defining service level objectives (SLOs)**
 - **Analyzing reliability risks**
- **Clear agreement on service conditions**
- **Keep number of new hires to <1/3 of team**
 - **Training effort doesn't overwhelm existing team members**



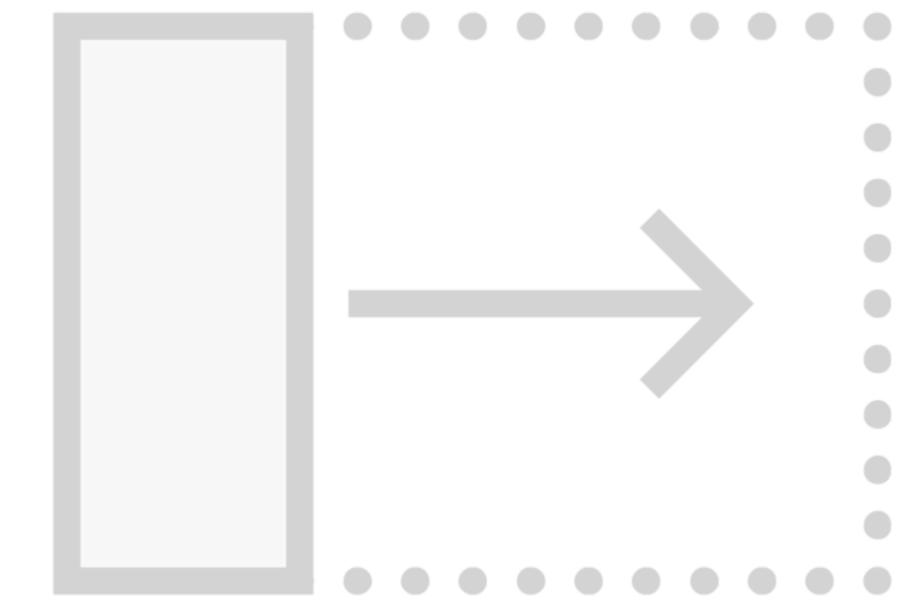
Bootstrapping Methods



**Creating a new
team as part of a
major project**



**Assembling a
horizontal SRE team**



**Converting an
existing team
to an SRE team**





Assembling a horizontal SRE team

Assembling a horizontal SRE team

- Small team of site reliability engineers
 - Consults across several teams
- Establish best practices and tools
 - Configuration management
 - Monitoring
 - Alerting





Assembling a horizontal SRE team

Risks

- Team perceived as a new organization
 - Does no real work
 - Adds no real value





Assembling a horizontal SRE team

Mitigation

- Team should include respected engineers
 - Subject matter experts
- Team undertakes project work
 - Focuses on delivering tools
 - Monitoring, alerting, rollouts, checklists, best practices
 - With clear short-term beneficial impact
 - To at least two teams
 - Communicate successes and benefits
 - Focus on solutions
 - Enablers vs. Gatekeepers



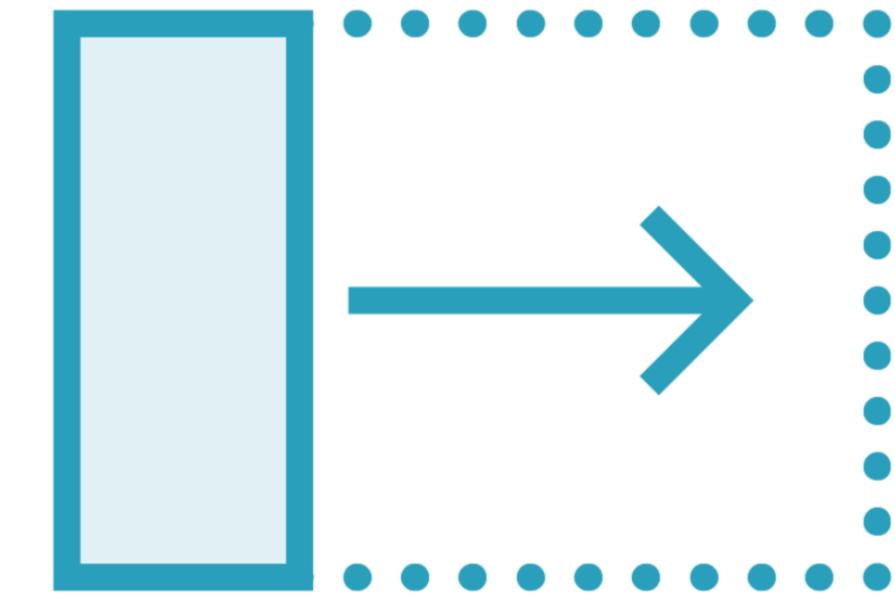
Bootstrapping Methods



**Creating a new
team as part of a
major project**



**Assembling a
horizontal SRE team**



**Converting an
existing team
to an SRE team**





Converting an existing team to an SRE team

Converting an existing team to an SRE team

- Existing team
 - Not a product development team
- Typical team candidates
 - Operations team
 - Responsible for managing a system
 - Organization use heavily
- Avoid renaming a team
 - “Operations” to “SRE”
 - Without applying SRE
 - Practices, principles



Converting an existing team to an SRE team



Risks

- **Perceives team conversion as start**
 - **Job losses due to automation**
- **Team doesn't support conversion**
- **Have no extra capacity**
 - **Change day-to-day activities**
- **Sees no benefit to day-to-day routine**
- **Works with systems that do not support**
 - **Scripting, automation**
- **No skills to automate current workload**
- **No skills to evolve toward an SRE team**
- **No interest to acquire skills**





Converting an existing team to an SRE team

Mitigation

- Secure senior management support
- Renegotiate responsibilities
 - Create extra capacity to effect change
- Manage communication of change
- Have access to robust support
 - Technical, personal
- Deal with concern about job losses
- Provide training to acquire skills SRE need
- Changes aligned: SLOs, SRE practices
- Add experienced engineer or developer
- Regular progress review
 - Internal, stakeholders



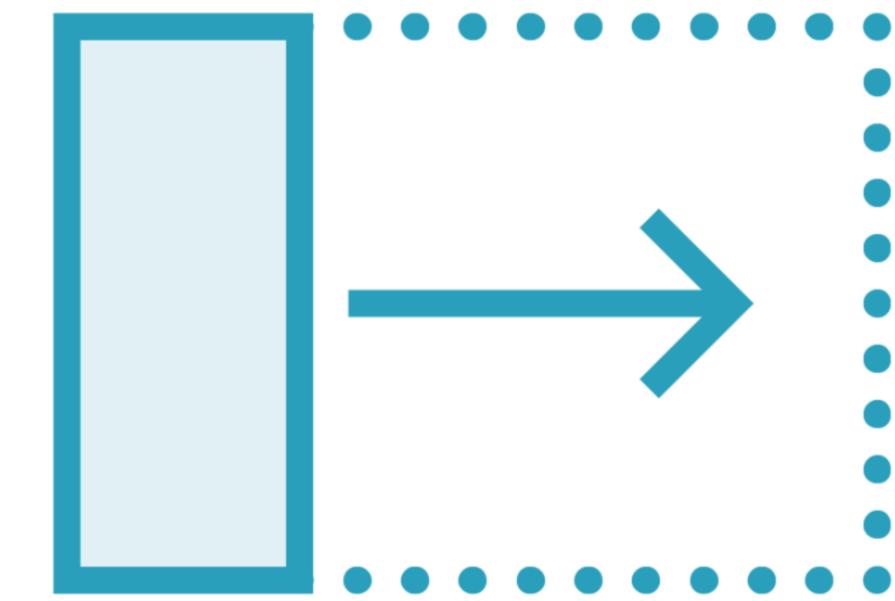
Bootstrapping Methods



**Creating a new
team as part of a
major project**



**Assembling a
horizontal SRE team**



**Converting an
existing team
to an SRE team**



Site Reliability Engineer Technical and Soft Skills



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**





Technical Skills Requirement

Software Development

- Experience and knowledge of major programming languages
 - Python
 - C++
 - Java
- Able to create tools for managing and automating infrastructure



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**





Technical Skills Requirement

Continuous Integration Knowledge

- Comprehensive knowledge
 - Continuous integration
 - Delivery
 - Deployment pipeline
 - Tools
 - GitLab
 - Jenkins
 - SonarQube



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**





Technical Skills Requirement

Major Operating Systems Competency

- Major operating systems
 - Windows
 - Linux
 - Unix
 - macOS
- Experience in OS administration



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**



Technical Skills Requirement



Experience in Networks

- Network visualization
- Network monitoring tools
 - Splunk
 - Nagios
 - Grafana



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

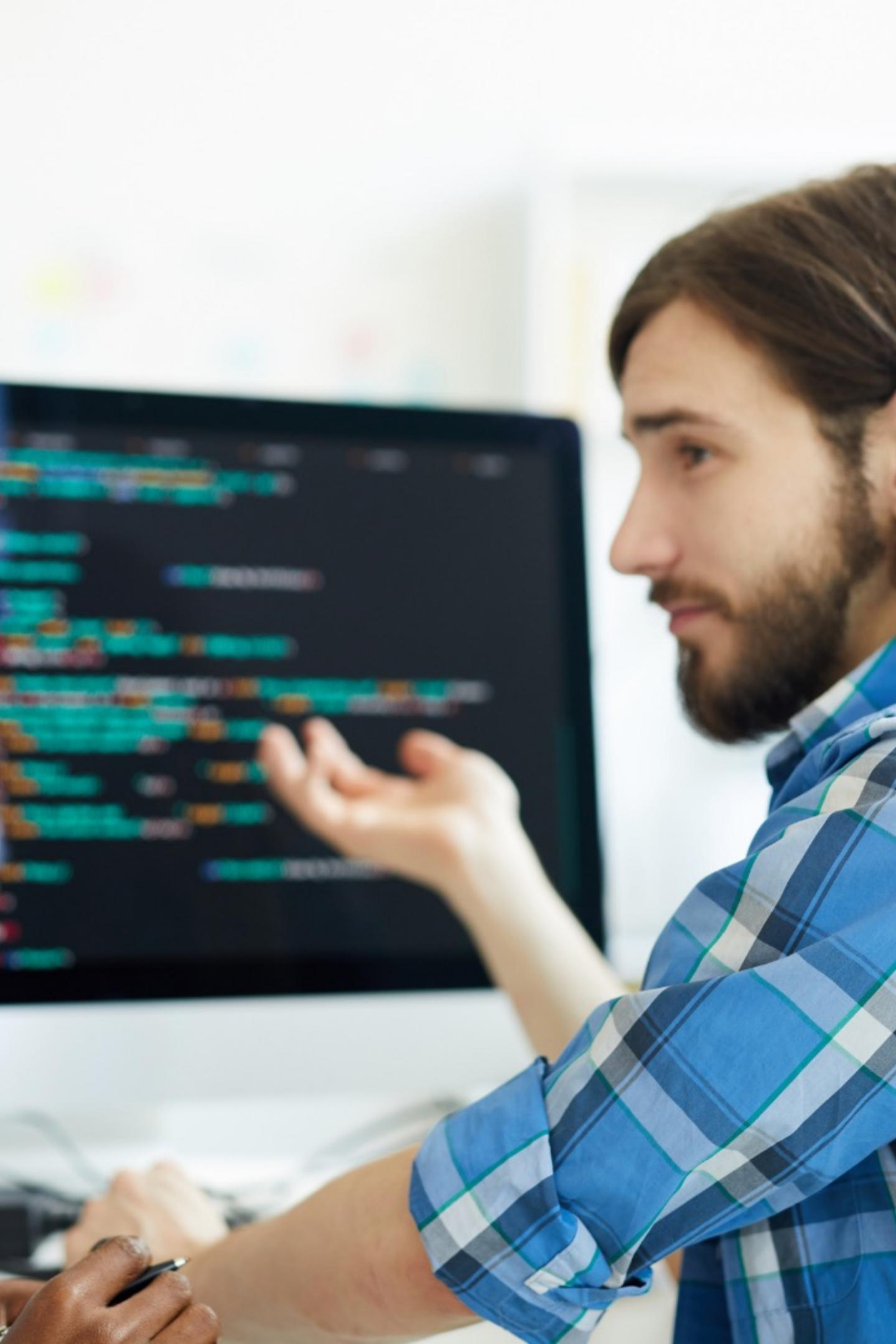
**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**





Technical Skills Requirement

Knowledge of Protocols and Cloud Hosting

- Protocols
 - TCP/IP
 - HTTPS
 - SMTP
 - DNS
- Cloud Hosting Services
 - Microsoft Azure
 - Hostinger
 - A2 Hosting
 - HostGator



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**





Technical Skills Requirement

IT Troubleshooting and Root Cause Analysis

- Expertise
 - IT Troubleshooting
 - Root cause analysis (RCA)
- Mitigate service downtimes



Soft Skills Requirement

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools



Soft Skills Requirement

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools



Soft Skills Requirement



Bigger Picture Analysis

- Understand how SRE team's work
 - Drive overall business
 - Affect a system or team
- Thinks of the bigger picture
 - Outside of the day-to-day activities



Soft Skills Requirement

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools



Soft Skills Requirement



Pragmatic and Forward-thinking Analysis

- Approach: realistic, rational
- Consider how their work affect
 - System or team
 - Future



Soft Skills Requirement

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools





Soft Skills Requirement

Embrace every Opportunity to Automate

- Successfully increase reliability
 - Without slowing software releases
- Proactive about finding new ways
 - Address manual tasks, bugs
 - Automate process or fix
 - Obsessive focus on automation



Soft Skills Requirement

**Bigger Picture
Analysis**

**Pragmatic and
Forward-Thinking
Analysis**

**Embrace every
Opportunity
to Automate**

**Ability to Persuade
Organizations**

**Problem Solving
Ability**

**Existing Skill Set
Expansion to
include New Tools**



Soft Skills Requirement



Ability to Persuade Organizations

- Confidence to advocate
 - Automation
 - SRE initiative
- Ability to convince people
 - Do things not done before
 - Do things few wants to do
- Effective salespeople
 - Sell long-term benefits of automation
 - Even with near-term sacrifices involved



Soft Skills Requirement

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools



Soft Skills Requirement



Problem Solving Ability

- **Detail-oriented**
 - **Define the problem**
 - **Detect root cause**
 - **Generate solution**
 - **Select best solution**



Soft Skills Requirement

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools



Soft Skills Requirement



Existing Skill Set Expansion

- Willing to adapt to work requirements
 - Flexibility
 - Motivation to learn something new
- Develop new skills
 - Out of comfort zone



Technical Skills Requirement

**Software
Development**

**Continuous
Integration
Knowledge**

**Major Operating
Systems
Competency**

**Experience in
Networks**

**Knowledge of
Protocols and
Cloud Hosting**

**IT Troubleshooting
and Root Cause
Analysis Expertise**



Soft Skills Requirement

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools



Designing a Hiring Pipeline for Site Reliability Engineering Function



Steps for Designing a Hiring Pipeline for SRE



Identify required technical and soft skills



Define role of the Site Reliability Engineer



Set-up comprehensive interview process



Set-up the right compensation and incentive system



Steps for Designing a Hiring Pipeline for SRE



Set-up the right organizational structure and support system



Conduct hiring activity



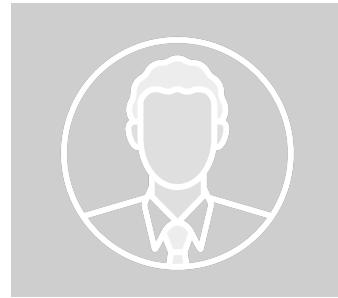
Evaluate and monitor hiring process



Steps for Designing a Hiring Pipeline for SRE



Identify required technical and soft skills



Define role of the Site Reliability Engineer



Set-up comprehensive interview process



Set-up the right comprehension and incentive system





Identify required technical and soft skills

Identify required technical and soft skills

- Technical skills
 - Indicate in “Job Description”

Software Development

Continuous Integration Knowledge

Major Operating Systems Competency

Experience in Networks

Knowledge of Protocols and Cloud Hosting

IT Troubleshooting and Root Cause Analysis Expertise





Identify required technical and soft skills

Identify required technical and soft skills

- Soft skills
 - Indicate in “Job Description”

Bigger Picture Analysis

Pragmatic and Forward-Thinking Analysis

Embrace every Opportunity to Automate

Ability to Persuade Organizations

Problem Solving Ability

Existing Skill Set Expansion to include New Tools



Steps for Designing a Hiring Pipeline for SRE



Identify required technical and soft skills



Define role of the Site Reliability Engineer



Set-up comprehensive interview process



Set-up the right comprehension and incentive system



A photograph showing a person's hands working on a white laptop. A smartphone is connected to the laptop via a USB cable. In the background, there is a white coffee cup with a black sleeve and lid. The laptop screen displays an Android Studio interface with code and project files visible.

Define role of the Site Reliability Engineer

Define role of Site Reliability Engineer

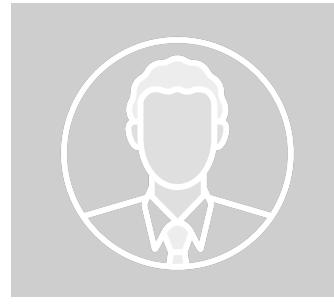
- As required by organization
- Engineering experience
- Improve: systems, processes
- Develop solutions to evolving challenges
 - Substitute automation for human labor
- Handle operations or on-call duties
 - Develop: systems, software
 - Increase site reliability and performance
- Indicate in job description



Steps for Designing a Hiring Pipeline for SRE



Identify required technical and soft skills



Define role of the Site Reliability Engineer



Set-up comprehensive interview process



Set-up the right comprehension and incentive system





Set-up comprehensive interview process

Set-up comprehensive interview process

- Depend on SRE talent level requirement
 - High-scale project, large organization
 - Small-scale project, small organization





Set-up comprehensive interview process

Sample comprehensive interview process

- Short pre-screening
 - Elementary knowledge of SRE role
- Technical pre-interview
 - Using video conferencing app with screen sharing function
 - Tool: Zoom, Microsoft Teams
 - Assess code knowledge
 - Solve: basic problems, daily SRE tasks
 - Tool: Coderpad
 - Check understanding
 - How system works





Set-up comprehensive interview process

Sample comprehensive interview process

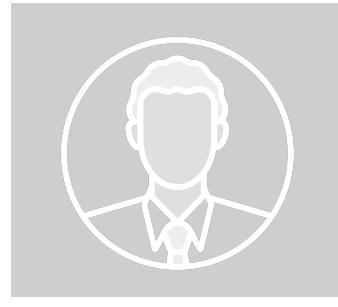
- Main interview
 - Start: involve many specialists
 - Dividing interviews into stages
 - Separate interviewer per topic
 - Goal: discover limit of candidate's knowledge on SRE



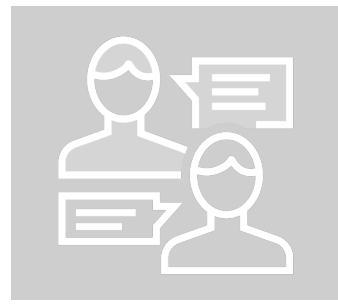
Steps for Designing a Hiring Pipeline for SRE



Identify required technical and soft skills



Define role of the Site Reliability Engineer



Set-up comprehensive interview process



Set-up the right compensation and incentive system





Set-up compensation and incentive system

Set-up compensation and incentive system

- Company resources are limited
- Align what the company can offer vs. what the employees desire
- Determine what motivates employees
 - Monetary
 - Compensation, Bonuses
 - Non-monetary
 - Incentive, Recognition, Benefits



Steps for Designing a Hiring Pipeline for SRE



Set-up the right organizational structure and support system



Conduct hiring activity



Evaluate hiring process





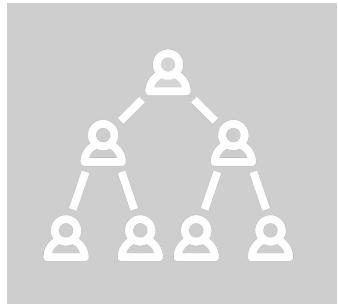
Set-up organization and support system

Set-up organization and support system

- **Determine organizational structure**
 - Required and relevant for the team
 - Identify hiring needs
 - Identify needed support system
- **Set-up organizational structure**
- **Provide required hires for different roles**
 - Leadership, SRE, Support



Steps for Designing a Hiring Pipeline for SRE



Set-up the right organizational structure and support system



Conduct hiring activity



Evaluate hiring process



A photograph of a young man with dark hair and a beard, wearing round glasses and a brown turtleneck sweater, smiling and shaking hands with another person whose back is to the camera. They are seated at a light-colored wooden desk with a white mug, a pen, and some papers. The background is a bright, modern office space.

Conduct hiring activity

Conduct hiring activity

- Advertise open positions for SRE team
- Review applications
 - Required skills: technical, soft
- Select applicants for interview
- Implement comprehensive interview
- Assess applicants
- Conduct background check
- Decide which applicants to hire
- Provide job offer
- Hiring formalities, onboarding



Steps for Designing a Hiring Pipeline for SRE



Set-up the right organizational structure and support system



Conduct hiring activity



Evaluate hiring process





Evaluate and monitor hiring process

Evaluate hiring process

- Evaluate hiring process
 - Effectiveness
 - Efficiency
 - Relevance
 - Appropriateness
- Evaluation methods
 - Satisfaction survey
 - Hiring manager, candidate or applicant
 - Key Performance Indicators
 - Time to hire, offer acceptance rate



Steps for Designing a Hiring Pipeline for SRE



Identify required technical and soft skills



Define role of the Site Reliability Engineer



Set-up comprehensive interview process



Set-up the right compensation and incentive system



Steps for Designing a Hiring Pipeline for SRE



Set-up the right organizational structure and support system



Conduct hiring activity



Evaluate hiring process



Creating a Plan for Career Progression in a Site Reliability Engineering Team



Career Progression

Process of climbing the ladder during an individual's working life – moving forward, being promoted, finding new challenges, and getting the most out of one's career.

Citation: PUSHfar, Career Progression, <https://www.pushfar.com/article/career-progression/#:~:text=Career%20progression%2C%20quite%20simply%20put,most%20out%20of%20your%20career>.



Career Progression Plan

Roadmap detailing how employees on a particular role can advance through the company or organization.



Career Progression Plan



Career Progression Plan

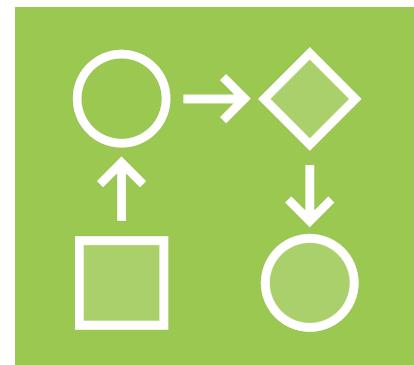
- Questions addressed
 - What path (or paths) can be followed?
 - What steps should be taken?
 - Timeline for required milestones?
 - What resources are available to use?



Steps in Creating a Career Progression Plan



Evaluate needs of the company, team, and employees



Create a career path for members of an SRE team



Define metrics required to reach defined career levels



Steps in Creating a Career Progression Plan



Review the career path and metrics with team members



Identify company resources and determine compensation plan



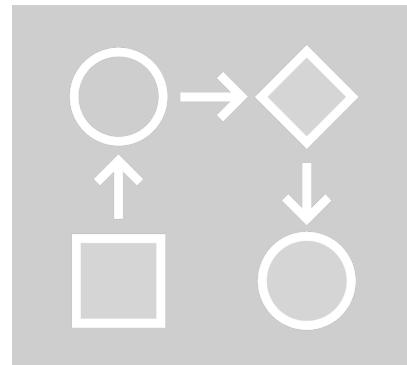
Document detailed career progression plan



Steps in Creating a Career Progression Plan



Evaluate needs of the company, team, and employees



Create a career path for members of an SRE team



Define metrics required to reach defined career levels



Evaluate needs of company, team, employees



Evaluate needs of company, team, employees

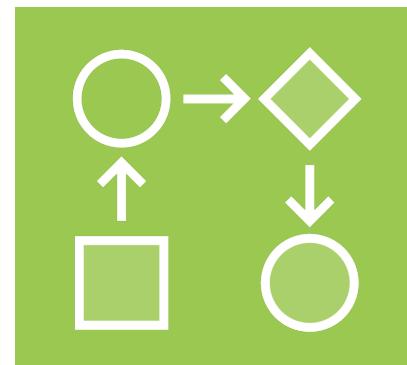
- **Needs of the company**
 - Operational and project requirements
- **Needs of the team**
 - SRE team organizational structure
 - Manpower to fill-out roles
 - Justified by additional value provided
- **Needs of employees**
 - Aspirations for career progression
 - Tasks that increase motivation and engagement



Steps in Creating a Career Progression Plan



Evaluate needs of the company, team, and employees



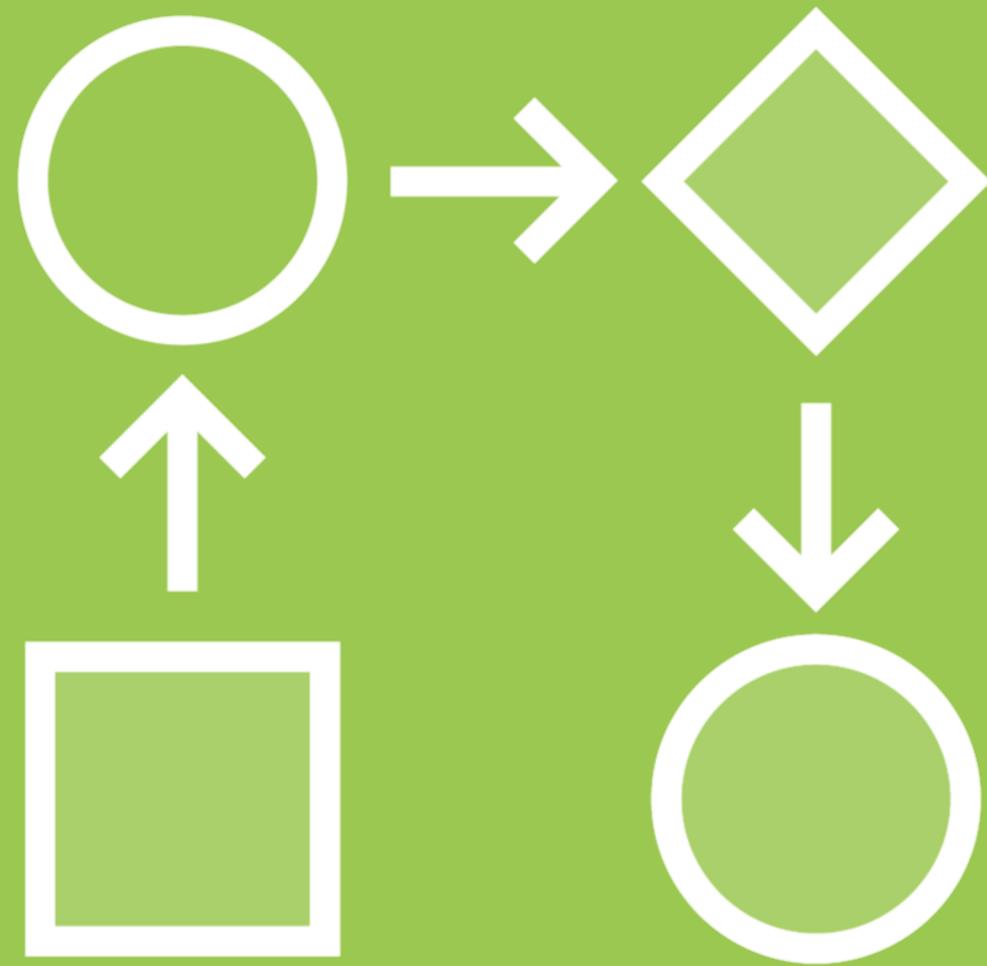
Create a career path for members of an SRE team



Define metrics required to reach defined career levels



Create a career path for members



Create a career path for members

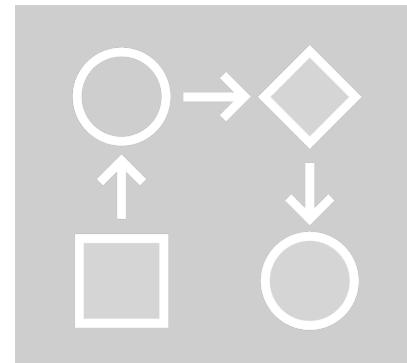
- Indicate required roles
 - Based on organizational structure
- Determine career track or path per role
- Indicate career levels
 - Starting, Mid-level
 - Senior, Leadership



Steps in Creating a Career Progression Plan



Evaluate needs of the company, team, and employees



Create a career path for members of an SRE team



Define metrics required to reach defined career levels



Define metrics required for career levels



Define metrics required for career levels

- Indicate metrics and milestones
 - Required per role and career level
- Required metrics and milestones
 - Performance evaluation score
 - Years of relevant experience
 - Scope of task or projects handled
 - Ranking: role, career level



Steps in Creating a Career Progression Plan



Review the career path and metrics with team members



Identify company resources and determine compensation plan



Document detailed career progression plan



Review career path and metrics with the team



Review career path and metrics with the team

- Discuss with SRE team
 - Career progression plan per role
 - Required metrics and milestones
- Align and obtain buy-in
- Increases accountability
 - Motivates employees to take action
- Increases transparency
 - Requirements to be promoted



Steps in Creating a Career Progression Plan



Review the career path and metrics with team members



Identify company resources and determine compensation plan



Document detailed career progression plan



Identify resources and compensation plan



Identify resources and compensation plan

- **Assess available company resources**
- **Generate expected compensation**
 - **Every role, every career level**
 - **Align with available company resources**
 - **Align with employee motivation**
- **Indicate other resources that can support career progression**
 - **Course and training opportunities**
 - **Mentorship Program**
 - **Conferences, seminars, learning events**



Steps in Creating a Career Progression Plan



Review the career path and metrics with team members



Identify company resources and determine compensation plan



Document detailed career progression plan



Document detailed career progression plan

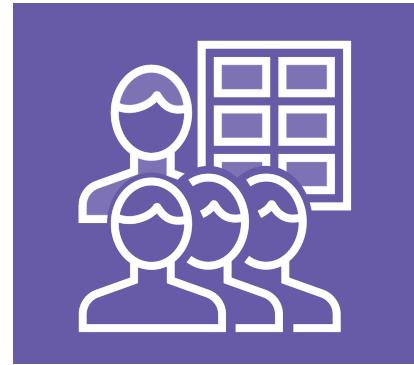


Document detailed career progression plan

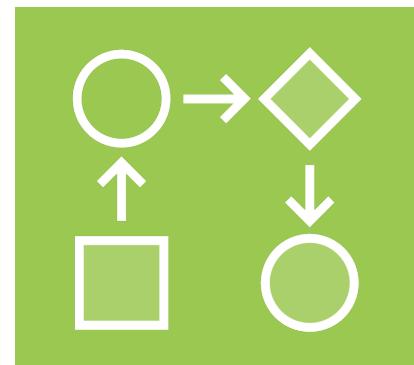
- Incorporate review session feedback
- Document career progression plan
- Submit to HR for review
- Finalize career progression plan based on HR feedback
- Resubmit to HR
- HR to keep record in file



Steps in Creating a Career Progression Plan



Evaluate needs of the company, team, and employees



Create a career path for members of an SRE team



Define metrics required to reach defined career levels



Steps in Creating a Career Progression Plan



Review the career path and metrics with team members



Identify company resources and determine compensation plan



Document detailed career progression plan



Make adjustments to the career progression plan as the employees and company evolve.



Use Case: Structuring an Optimal Site Reliability Engineering Function





Globomantics

(Manufacturing Company)





Scenario

Globomantics, a multinational manufacturing company, has recently initiated a strategy to move towards digitalization to improve operational efficiency in its different functional areas.

A Site Reliability Engineering (SRE) team is recently established to manage its relevant IT operations. Being newly established, several team structures were implemented throughout its early phases.





What to do

Identify what team structure is implemented and if an intervention with regards to structuring a Site Reliability Engineering (SRE) function is needed or not in each scenario

- **If intervention is not needed:** Indicate which concept is implemented
- **If intervention is needed:** Indicate which concept needs to be implemented and expound on how it should be implemented
- **Concept options**
 - Designing a hiring pipeline
 - Creating a plan for career progression







New SRE Team

- Started operating 6 months ago
- First and only Site Reliability Engineering team currently in existence at Globomantics
- Need additional Site Reliability Engineers (SREs) to current organizational structure
 - Handle new scope of services and work

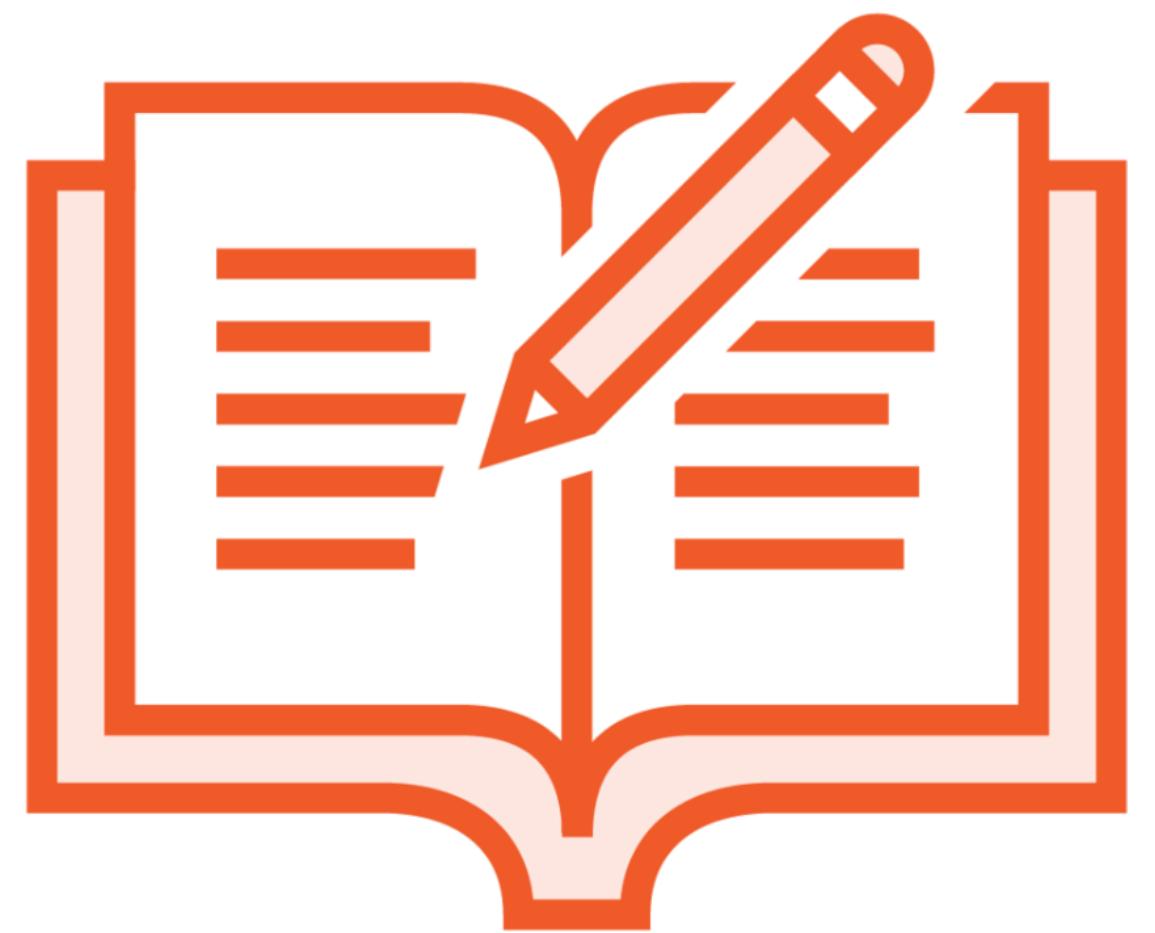




New SRE Team

- Initiated the following hiring or recruitment process steps to fill-out new roles
 - Identified technical and soft skills for the role
 - Defined the role of SREs to hire
 - Set-up comprehensive interview process
 - Set-up right compensation and incentive system
 - Set-up right organizational structure and support system
 - Conducted hiring activity
 - Evaluated and monitored overall hiring process





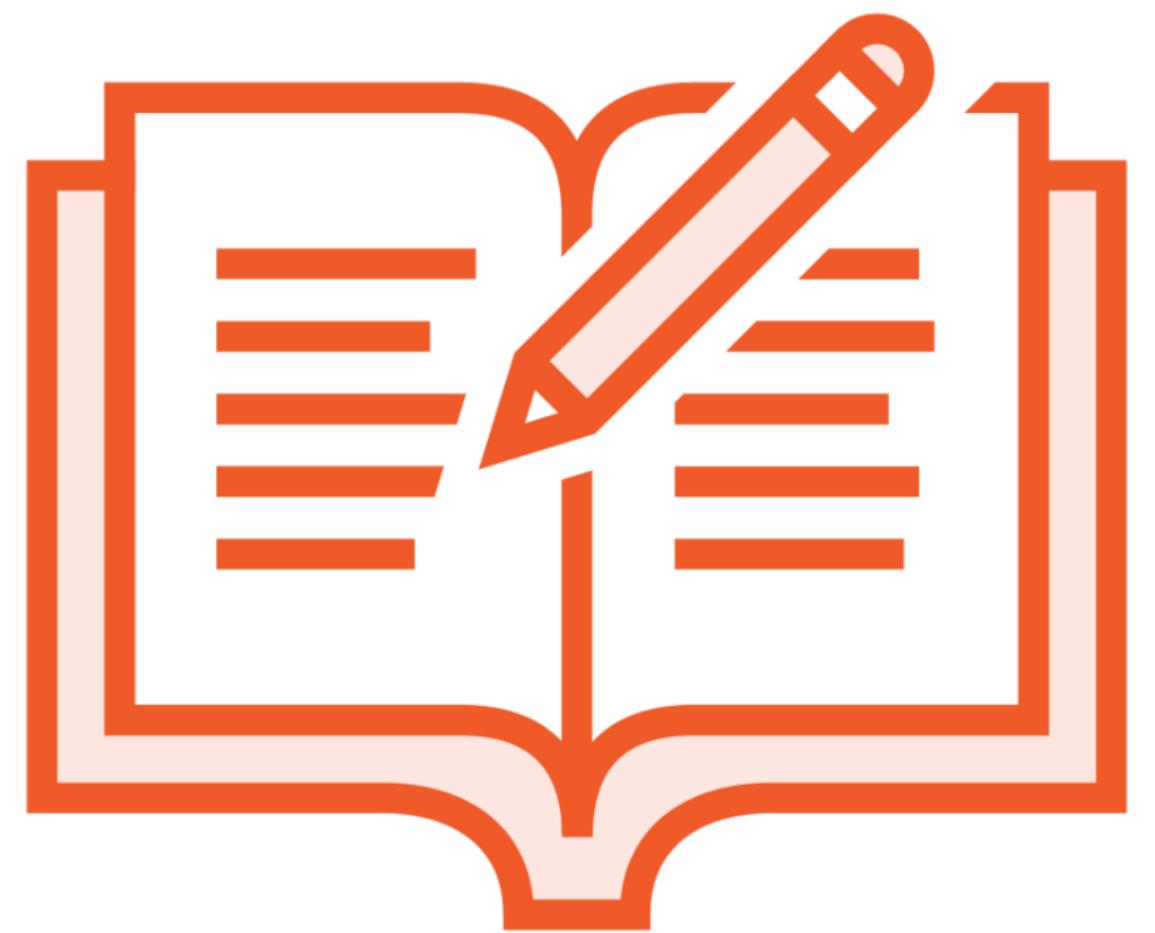
A photograph of two people, a man and a woman, working together on a computer. The man, wearing glasses and a beard, is focused on the screen, while the woman, also wearing glasses, points at the screen, likely discussing the code. The background shows multiple computer monitors displaying lines of code.

Case Scenario 2

2-year-old SRE Team

- Started operating 2 years ago
- 2nd Site Reliability Engineering team currently in existence at Globomantics
- Need for dedicated Site Reliability Engineers (SREs)
 - Expand scope with high reliability needs
- Have key user-facing applications created
- Team works to improve reliability
 - Critical business unit
- Structure limits knowledge sharing among SREs
 - Limited mobility to handle different projects
 - Lack of defined career path for SREs
 - Demotivated most SREs as opportunities for career growth is limited





Answer: Case Scenario 1



Kitchen Sink

- Team Structure

Designing a hiring pipeline

- Concept implemented
- Intervention is not needed



A photograph of two people working on a computer. A woman with red hair and glasses is pointing at the screen, while a man with a beard and glasses is typing on the keyboard. The screen displays code or data. The scene is lit with warm, orange light.

Answer: Case Scenario 2

Application

- Team Structure

Creating a plan for career progression

- Concept that needs to be implemented
- Intervention is needed



A photograph showing a woman with glasses and a man with a beard and glasses working together at a computer. The woman is pointing at the screen, and they appear to be discussing code or data. The screen shows multiple windows of text.

Answer: Case Scenario 2

Steps in creating a plan for career progression

- Evaluate needs of the company, team, and employees
- Create a career path for members of an SRE team
- Define metrics required to reach defined career levels
- Review the career path and metrics with team members
- Identify company resources and determine compensation plan
- Document detailed career progression plan



Summary



- You should be able to:
 - Determine what SRE team structure fits the current needs of your organization
 - Avoid anti-pattern traps in responding to problems faced by your organization
 - Implement the right bootstrapping method for structuring your SRE team
 - Determine required soft and technical skills for a Site Reliability Engineer
 - Incorporate it in designing a hiring pipeline for your organization
 - Set-up an effective team career progression plan for your SRE team

