JUPITER PRIME
Consulting • Training • Advisory
جوبيتر برايم للاستشارات والتدريب

COURSE BROCHURE

# Site Reliability Engineering (SRE) Training Course

11 – 15 January 2027
Milan

**jupiterprime.uk**
info@jupiterprime.uk
Registered in the United Kingdom and Saudi Arabia

# Site Reliability Engineering (SRE) Training Course

**Ref:** 1319_49472 **Date:** 11 – 15 January 2027 **Location:** Milan **Fees:** 7200 GBP

Site Reliability Engineering SRE is a 1-week training course for DevOps engineers, platform engineers and IT operations leads, in which each participant drafts a Production Service Reliability Plan for a live service they run. Customers now expect digital services to stay available around the clock, and outages reach social media within minutes, so reliability has become a product decision rather than an afterthought. Jupiter Prime Training delivers this SRE course.

## Why Site Reliability Engineering SRE Matters Now

Three pressures shape SRE today. Release frequency keeps rising, so the gap between shipping features and keeping services stable must be negotiated with numbers rather than opinion. Cloud and microservice architectures multiply failure paths that no single operator can hold in mind. And on-call fatigue drives engineers out of operations roles when repetitive toil is left unchecked. Teams without measured targets swing between release freezes and firefighting. The course rests on the service level indicator, objective and agreement model SLI, SLO, SLA and blameless postmortem practice.

## Who Gains Most from Site Reliability Engineering SRE

- Engineers who own deployment pipelines, rollbacks and release safety for production services
- Platform specialists responsible for shared infrastructure, clusters and observability stacks
- Operations leads who run on-call rotations, escalation paths and incident command
- Application engineers who carry pager duty for the services they build
- Service owners who negotiate availability targets with product and business stakeholders

A product squad sending developers, platform engineers and its on-call lead together gains most, since error budget policy needs all three.

## Site Reliability Engineering SRE Capabilities

Participants turn user journeys into measurable reliability targets and use those targets to steer release and staffing choices.

- Specify SLIs and SLOs from critical user journeys — product owners setting availability targets
- Calculate error budgets and burn rates — release go or freeze decisions
- Design symptom-based alerting and dashboards — on-call engineers and escalation leads
- Apply incident command roles during outages — customer communication and recovery speed
-

Build blameless postmortems with tracked action items — engineering leadership and risk owners
- Assess toil and automate repetitive tasks — capacity planning for operations staff

## Key Site Reliability Engineering SRE Terms

- Service Level Indicator SLI — A measured ratio of good events to total events, such as successful requests or fast page loads.
- Service Level Objective SLO — The target value for an SLI over a rolling window, agreed between engineering and product.
- Error Budget — The amount of unreliability an SLO allows, spent on releases, experiments and unplanned failures.
- Burn Rate — How fast a service consumes its error budget relative to the rate that would exhaust it exactly at window end.
- Toil — Manual, repetitive, automatable operational tasks that grow with service size and add no lasting improvement.
- Blameless Postmortem — A written incident review that examines system and process causes instead of assigning personal fault.

## Site Reliability Engineering SRE Day by Day

## Day 1: Reliability as a Product Feature

- SRE Principles and the DevOps Relationship — Who Owns Production Stability
- Critical User Journey Mapping — Choosing What Reliability Means to Customers
- Availability, Latency and Durability Targets — Balancing Cost Against Expectation
- Dependency and Failure Domain Diagrams — Spotting Shared Single Points of Failure
- Reliability Maturity Self-Assessment — Setting the Starting Point for Change

## Day 2: SLIs, SLOs and Error Budgets

- SLI Specification and Measurement Points — Load Balancer, Client or Synthetic Probes
- SLO Window and Target Selection — Rolling Versus Calendar Compliance Periods
- Error Budget Policy Writing — Release Freeze and Escalation Triggers
- Multi-Window Burn Rate Alerts — Paging Only on Meaningful Budget Loss
- SLA Alignment With Internal SLOs — Contract Penalties and Safety Margins

## Day 3: Observability and Incident Response

- Metrics, Logs and Traces — Choosing Signals for Fast Fault Isolation
- Golden Signals Dashboards — Latency, Traffic, Errors and Saturation at a Glance
- On-Call Rotation and Runbook Design — Sustainable Paging Load per Engineer
- Incident Command System Roles — Commander, Communications and Operations Lead
- Status Page and Stakeholder Updates — Customer Trust During Outages

## Day 4: Learning From Failure and Removing Toil

- Blameless Postmortem Templates — Contributing Factors and Action Item Ownership
- Toil Inventory and Measurement — Engineering Time Caps for Operations Work
- Automation Candidates and Self-Healing — Restart, Scale and Failover Scripts
- Progressive Delivery With Canary and Feature Flags — Limiting Release Blast Radius
- Chaos Experiments and Game Days — Validating Recovery Before Real Incidents

## Day 5: Practising the Production Service Reliability Plan

- SLO Set for a Case Service — Applying Day 2 Specification Methods
- Alert and Dashboard Review — Applying Day 3 Signal Choices
- Toil Reduction Backlog — Applying Day 4 Automation Priorities
- Capacity Forecast and Load Testing Plan — Headroom for Peak Demand
- Completed Production Service Reliability Plan — Agreement With Product and Operations

## Applying Site Reliability Engineering SRE After the Course

- Publish two SLOs and an error budget policy for one customer-facing service
- Run a blameless postmortem on the most recent outage and track its action items to closure

## Site Reliability Engineering SRE Questions FAQ

### What is site reliability engineering SRE, and why does it matter now?

SRE applies software engineering to operations, using measured reliability targets to guide releases. It matters now because faster deployment and complex cloud systems make outages more frequent and more visible, in sectors from online banking to e-commerce.

### Which roles gain most from site reliability engineering SRE?

DevOps engineers, platform engineers and IT operations leads gain most. Developers who share on-call duty also benefit.

### How does site reliability engineering SRE differ from an IT service management course?

SRE focuses on engineering practice: SLOs, error budgets, automation and observability. An IT service management course concentrates on process frameworks for requests, changes and service desks rather than code-driven reliability work.

### What do you bring back from site reliability engineering SRE training?

You bring back a reliability plan for one of your own production services, with SLOs, an error budget policy, alert rules and a toil reduction backlog.

## Site Reliability Engineering SRE: The Way Forward

The Production Service Reliability Plan first informs whether the next release goes ahead or the service pauses to recover its error budget. Service owners, on-call engineers and product managers all use it. After the first full SLO window, revisit the targets themselves: an SLO that is never at risk is probably too loose, and one always breached is too tight.