{mthree}

Observability

Site Reliability Engineering

Objectives

In this module, we will provide an overview of the concept of observability.

Learning Objectives

- Describe the parts of observability
- Discuss the concept of application performance monitoring
- Explain the relationship between service level indicators (SLIs), service level objects (SLOs) and service level agreements (SLAs)

{mthree}

Observability

- More than monitoring
- Three parts
  - >>> Logs
  - >>> Metrics
  - >>> Traces
- Using data from a complex system to infer its state
  - >>> Capacity & performance
  - >>> Satisfaction
  - >>> Expectations

{mthree}

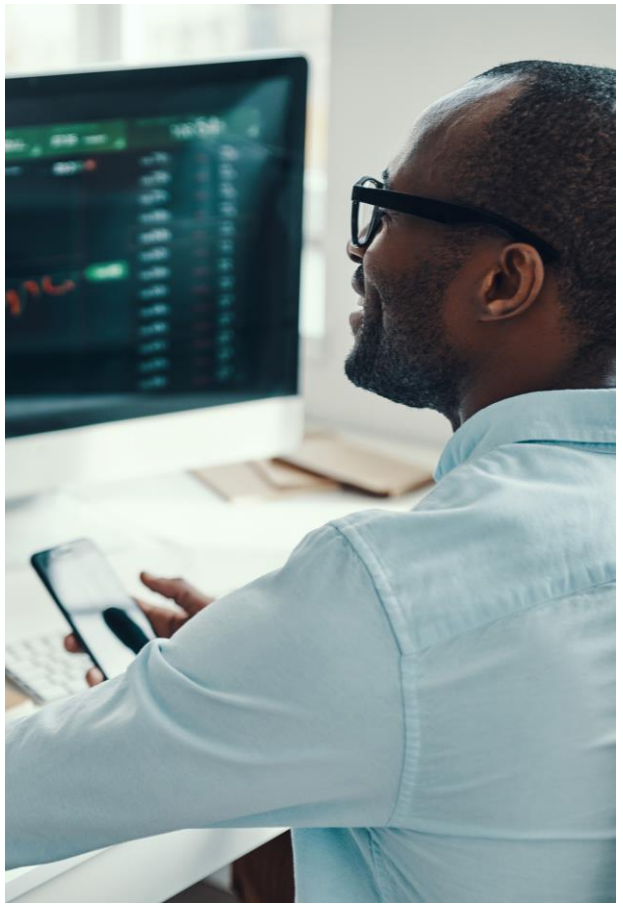
Three Parts of Observability

Logs	Record of past events
Metrics	Current data about the system components
Traces	Capture activity for a business transaction Shows interactivity in complex systems

{mthree}

Observability Data

- Combining four parts to answer questions
  - >>> Why performance is degrading
  - >>> What dependency behaviors have changed
  - >>> Why this application is failing
  - >>> Where to look for a fix

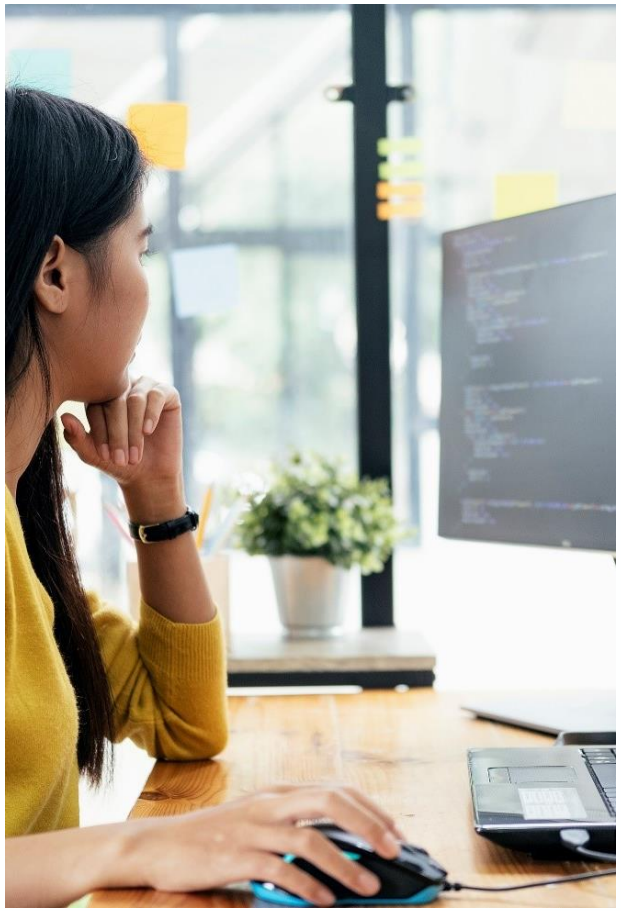


{mthree}

Expectations

- Focus on user expectations
  - >>> Delivering more than expected doesn't gain much
  - >>> Failing to deliver will lose a lot

- Observability should provide at a glance:
  - >>> Are we out of SLO?
  - >>> Should SLO be adjusted?
  - >>> Where is the problem?



{mthree}

Observing Toil

- How do we measure toil success?
- Where do we get the metrics?

{mthree}

Q&A

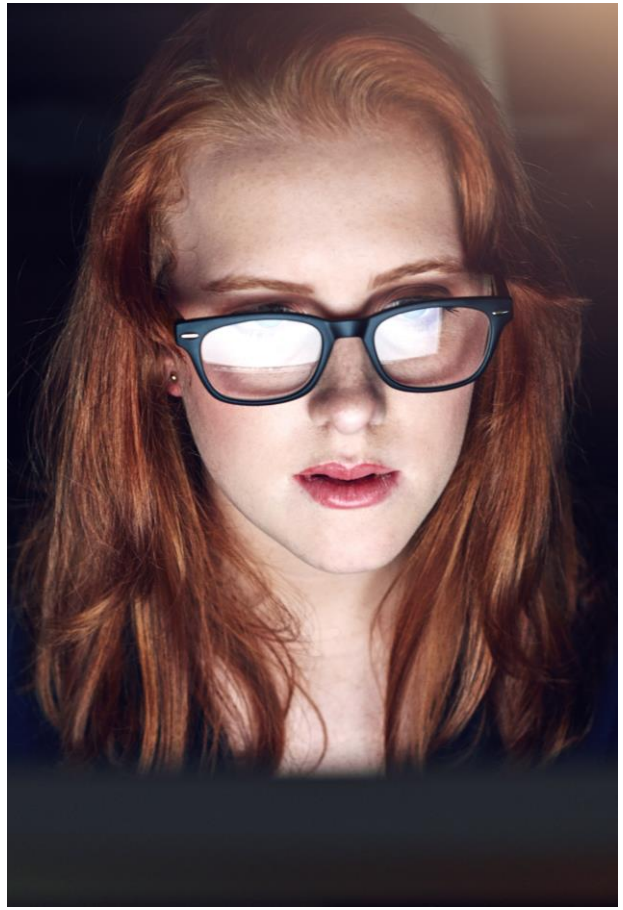
{mthree}

SLAs vs SLOs vs SLIs

{mthree}

Observing Toil

- Toil backlog
  - >>> Should be reducing
  - >>> Should not see the same toil recurring in the backlog
  - >>> Error budget not decreasing as fast = more reliable system
- Reduced fatigue in the team
- Shorter MTTR
- Toil metrics come from
  - >>> Ticketing and job systems such as Jira and ServiceNow
  - >>> Other systems that gather information about time at work



{mthree}