

User Count / Customer Count

How many users / customers (broken out by roles / personas) are serviced by each system / application. Do new apps, features, or capabilities targeted at specific groups of users impact user count as desired / expected?

Team Health Score

A compound metric to assess if team members are engaged, able to work without impediments and enjoy what they are doing. Assess if activities to improve team health are providing desired benefits?

Validated Learnings / Learning Log

Data to confirm that teams are learning necessary skills to develop products, improve their architecture / technology, and enable outstanding customer experiences while building the product.

Team Member Churn / Team Tenure

Assess if team composition is stable – have team members been around for an extended period of time where they may challenge new ideas (expert bias) and/or is the team so new it needs coaching, guidance and management to stay aligned to goals & objectives.

Net Promoter Score of Product / App and/or Feature (rated by end-users)

Do our customers / end-users value the product and/or features being delivered to where they would recommend it to others – typically assessed on a 5-point or 10-point scale and included as a part of app store reviews, etc.

Frequency of Software / Capability Releases

How frequently are we able to release updates to our product, application and/or system to end-users – assess if the frequency of software release aligns to what is necessary for the business context / business goals? Are we able to release at a rate comparable to our competitors – do we need to for competitive advantage?

Release Time / Release Cost

How much time / cost is required for a software / capability release – this metric can assess if efforts to automate release activities provide intended benefits to reduce release time and/or cost.

New Capability / Feature Lead Time

What is the lead time to deliver a new capability / feature to end-users / customers from when it is first envisioned to when it is deployed in production – is this lead time predictable? Do efforts to reduce lead time to deploy new capabilities provide desired impacts?

Number of Social Media Impressions (market reach)

Assess the impact of marketing and/or new application / feature releases to drive impressions on social channels – increased impressions enable greater brand awareness & equity supportive of adoption, customer retainment, and new customer acquisition.

Development Cycle Time

The amount of time required for a team to write the code & tests to support a change request / story. Can be used to assess if the team has a consistent / predictable process, and also assess the impact of adopting standards, architecture patterns and/or tools to reduce time necessary for development and unit testing.

Build / Testing Cycle Time

How long does it take to build a new version of the app, application and/or system. Slow builds (longer than 10 minutes) increase the amount of time until a team can receive feedback to confirm that new changes have not broken prior functionality – slow builds may also be an indicator of not maintaining application architecture, tests or the build system.

Regression Test Cycle Time

How long does it take to run a regression test cycle to confirm that full application features are not impacted / broken by new changes. This metric can be used to demonstrate the impact of creating automated regression tests to enable faster testing to confirm proper application functionality.

Frequency of Regression Test Runs

How frequently are regression tests run to confirm that full application features are not impacted / broken by new changes. This metric can be used to demonstrate the impact of creating / adopting practices for automated regression testing that enable more frequent regression testing to increase quality / reliability.

Percent Complete & Accurate

Measuring the number of stories and/or change requests that are completed accurately without rework or error. This metric serves as a good demonstration to assess process control / process stability and the effectiveness of product management and testing activities to find issues / defects before release.

Cumulative Flow

Measuring the flow of change requests / stories through all steps in the development / testing process – enables assessment of work-in-progress at each step of the process. Assess if sufficient resources / capacity are allocated to prevent bottlenecks during development that could slow the delivery of new features. Use to calibrate work-in-progress limits to enable consistent flow.

Time Blocked Per Feature – Root Cause of Blockers (Blocker Clusters)

How much delay occurs during the development of a feature and more importantly what is the root cause of that delay. This metric is intended to guide identification of common sources of delay impacting multiple features / teams / projects so that common solutions can be enacted to reduce delay for all products / teams in the portfolio.

Escaped Defect Resolution Time

What is the resolution time (mean time to resolution – MTTR) to resolve feature / capability defects that are released to end users? This metric can be used to assess if resolution time is predictable and completed within SLAs and also to assess the impact of strategies to reduce defects such as reducing technical debt.

Planning to Implementation Ratio

What is the ratio of the time spent to prepare a feature / change request for implementation compared to the time required to develop and test it. Tracking this metric promotes measuring full lifecycle product development and can assess if sufficient (or too much) time is spent during ideation & design of the product vs. building the product / features for delivery.

Release Adoption / Install Rate

How many users consume updates and/or new features that are made available. This metric can assess if features and/or apps are aligning to customer / end-user needs (aka: are we building the right thing) and/or can be used to assess the effectiveness of marketing and/or social media to promote adoption of new features and/or apps.

Business Value Burn Up

An assessment of how much business value has been delivered to end-users. The goal of this metric is to determine when the majority of business value for an application or feature has been delivered signaling it is time to move on to another priority item and “cut the tail” of development to avoid focusing on low value features compared to higher value features already delivered.

Product Development Forecast

Target dates and/or investment levels for when specific features / capabilities will be available to end-users. Trend lines within a forecast (best case & worst case) enable business risk assessment and coordination of marketing, go-to-market, and/or product launch activities to promote adoption and/or sale of new app / products / features as they are delivered.

Emergent Usage Analytics

Usage analytics to assess if users are engaging with the application / system as intended. Review of analytics allows for confirmation of assumptions made within personas and/or target users regarding how the application will be used – assess if the application is being used as intended, or are users not able to engage as designed (and if so, why).

Observed Defect Density / Crash Density

How frequently does the application crash or experience an unintended error and what component / module of the application triggers such failures. Analysis of defect density & crash density data allows for identification of poor quality code and can identify gaps in testing where activities to improve the code & tests may reduce defects / crashes.

Code Churn Per Change Request / Feature

How many lines of code are changed for each feature / change request completed. This metric can contribute to technical risk assessment of features (is there more risk in features that impact more lines of code) and also to assess overall architecture, code quality and engineering disciplines as well maintained code should result in fewer lines of code change.

Application Code Complexity

A quantitative score based on the complexity of the application source code determined by algorithm. Code complexity is an indicator of technical risk and can also be used to assess practices supportive of reducing technical risk including: aligning to architecture standards or design patterns, and applying specific engineering disciplines (refactoring, etc).

Shared Code Ownership

How many team members contribute to or review the code/tests for change requests and/or new features. This metric enables assessment of if the team is working to achieve collective code ownership whereby all parts of the code & tests are understood and can be supported by multiple members of the team.

Test Code Complexity

A quantitative score based on the complexity the test code determined by algorithm. Test complexity is an indicator of technical risk and overall team / engineering disciplines – are teams engaging with appropriate rigor to maintain their test code so that automated tests provide reliable results in a timely manner.

Coding / Architecture Standards Adherence

Assessment / compliance score of an application aligning to required technical / architecture standards & checks. Tracking this metric enables confirmation that development aligns to required standards (evidenced by this metric) to enable technical improvement such as: reduced technical risk, fewer defects, improved performance, faster releases, etc.

Happiness Metric / Job Satisfaction

A qualitative assessment of if team members are happy in their roles working on the product / portfolio. Happy people provide better quality work / outcomes. Tracking this metric enables assessment of strategies / activities to improve team member morale.

Revenue per user / Revenue per session

What is the observed revenue (business value) received from each user of a system / application or within each session the application is used. Enable business / financial analysis to determine if new features / capabilities enable more revenue (business value) per user and/or per session.

Failed Build / Failed Test Response Time

How long is required for a team to investigate and fix failed builds (bad code, broken tests, etc). Failed build response time enables assessment of build / architecture quality (fewer failed builds can indicate better quality code) and if team has sufficient process, tools & maturity to enable timely response to support needs.

Team / Project Run Rate

The ongoing cost for a team to work on a product / project (weekly / monthly). Enable assessment to confirm that the team is delivering sufficient business value considering their cost. Additionally, enable assessment that strategies to improve efficiency / automation keep team run rate constant while increasing business value delivered.

Release Deviation Rate

The frequency that a software release does not go according to plan / process and intervention is needed. This metric can assess release process maturity / compliance, identify insufficient testing prior to release activities, and/or demonstrate the benefits of release automation so as to reduce incidence of deviations from expected process.

Organizational Health Score

A qualitative assessment to determine the health of teams within the portfolio inclusive of “smart” (goals, metrics, processes) and “healthy” (safety, engagement, feedback) criteria. Enable determination if strategies / practices to improve organizational health provide expected and desired improvements.

Staff Utilization

Allocation of time / capacity of staff across portfolio priorities, projects and/or products. Assess / compare if sufficient capacity has been allocated to support business goals and determine if staff is able to focus / engage in accordance with stated priorities.

Impact to Portfolio / Program / Product Overhead

The overhead (financial metric) necessary to sustain activities. Enable assessment of the impact that changes to process, people and/or platforms have to ongoing overhead expense.

Impact to Portfolio / Program / Product Profit / Net Contribution

The profit / net contribution (financial metric) generated by an application, system and/or features within an application. Enable assessment of if enabling a new application, system or features within an application has an impact on organization financials.

Impact to Portfolio / Program / Product Customer Acquisition Costs

The customer acquisition cost (financial metric) to increase the end-user / customer base for a system or application. Enable assessment of how new applications, features, marketing, and/or process changes have an impact on the cost to acquire a new user / customer of the system.

Impact to Business Inventory and/or Volume of Transactions

The transaction volume (business metric) of the business transactions that are enable / facilitated by the system, application and/or features within. Enable assessment of if new systems, applications and/or features have a significant impact on the number of transactions supported for an organization.

Incidence of Errors / Crashes During Performance / Stress Test Runs

Data to identify frequency and root cause of errors / crashes during performance testing. Enable assessment to determine if strategies to reduce errors / crashes when the system / application is under load provide desired benefits and if changes made during performance testing should be migrated to production environments.

Business Risk Score

A score (qualitative and/or quantitative) determined by inventorying and assessing all known business risks impacting the portfolio, project, product and/or system under examination. This metric guides prioritization of specific business risks to mitigate and enables confirmation that efforts to reduce business risk do not adversely impact other KPIs.

Technical Risk Score

A score (qualitative and/or quantitative) determined by inventorying and assessing all known technical risks / technical debt impacting the portfolio, project, product and/or system under examination. This metric guides prioritization of specific technical risks to mitigate and enables confirmation that efforts to reduce technical risk do not adversely impact other KPIs.

Business Impacts of Technical Debt and/or Legacy Platform Issues

Determination of estimated or actual business impacts experienced by operating in a legacy environment. Use this business impact data to assess / justify investments to improve the legacy environment and/or expedite efforts to migrate business activities to new systems and/or platforms to avoid adverse legacy platform issues.

Compute Cost for Capability / System / App

The ongoing operational cost for compute capacity (AWS, Azure, etc) to run the capability / system / app in all environments – could be computed from internal OpEx or provided by outside cloud provider invoice, ex: AWS. Assess if strategies, architecture changes, and/or pattern / tool adoption provide desired changes to ongoing compute cost.

Storage Cost for Capability / System / App

The ongoing operational cost for storage capacity (S3, etc) to run the capability / system / app in all environments – could be computed from internal OpEx or provided by outside cloud provider invoice, ex: S3. Assess if strategies, architecture changes, and/or pattern / tool adoption provide desired changes to ongoing storage cost.

Security / Vulnerability Score

A compound metric determined by security / vulnerability scan of application / system code. Assess if adoption of recommended architectures, security standards, and/or tools & processes provide improvement to application security & operational readiness.

Data Quality Score

A compound metric determined by assessment of the accuracy and completeness of the data necessary for an application or system to operate as intended. Enable assessment to determine if activities / strategies to improve data quality result in improved application stability, fewer crashes, and/or ability to service more customers / more business transactions without error.

Blocker Resolution Time

Once a blocker (or type of blocker) is identified, how long does it take to resolve. This metric enables assessment of team discipline & process to prioritize blockers as they arise and assessment of management support for team activities – determine if portfolio governance / management processes engage the right people quickly to minimize time blocked.

Business Stakeholder Engagement Rate & Reviews

The frequency that business stakeholder feedback is received and what rating / feedback do business stakeholders provide. Enable proactive management of business / stakeholder risk by reminding to solicit frequent stakeholder feedback and take action to sustain favorable rating / perception of services provided.

End-User / Customer Engagement Rate & Reviews

The frequency and ratings that end-users / customers provide in response to services provided by systems, applications, apps (app store ratings). Frequent ratings demonstrate an active customer community and high rankings demonstrate that the product / experience being provided aligns to customer / end-user needs.

User Analytics / Build the Right Thing Metric

Analytics data obtained by instrumenting applications to confirm applications / features provide desired end-user outcomes / experiences. Assessment of analytics data is intended to reveal insights to increase customer delight and reduce friction within the end-user / customer experience.

Business Scenario Monitoring

Determining business impacts (lost revenue, sales, etc) that result from technical outages and/or poor system performance (dropping transactions, etc). Compute business losses as a result of poor system performance / outages to be able to demonstrate the return on investment of improving system / platform architecture that reduce impacts to critical business scenarios.

Use of non-standard / non-approved tools / platforms

Tracking (via automated checks) systems / applications / apps that have dependencies on non-approved tools / libraries and/or modules. Enable identification of applications that may require additional scanning and/or auditing due to non-compliance, and also assess effectiveness of activities to guide adoption / alignment to standard tools & platforms.

Portfolio / Program / Product revenue

Tracking of revenue generated by the components of the portfolio. Enable assessment of if new features and/or applications bring in desired revenue to the portfolio / business (true business value) – enable comparison against portfolio costs to determine if portfolio is generating desired business results.

Operational Expense

Tracking the operational expenses necessary to sustain the portfolio and/or projects & products within the portfolio.

Used to assess if activities to improve processes, streamline activities, simplify infrastructure and/or adopt automation provide impact to operational expense.

Capital Expense

Tracking the capital expenses necessary to sustain the portfolio and/or the business activities supported by the portfolio. Can be used to demonstrate the financial impact of implementation and/or architecture changes to shift to cloud or a managed services provider.

Number of change requests received / Backlog intake

The number of changes requests / work items received in a period of time. Assess how frequently work is submitted to the portfolio and if there is sufficient capacity for execution within the portfolio to keep up with work intake – do business factors exist to justify observed work intake rate.

Distribution of work items / change requests received

What is the distribution of work items received across the categories of work supported by the portfolio. Assess if portfolio capacity aligns to observed work intake – determine if there are business trends and/or operational activities to support observed work item distribution.

Business Impact (Business Value) of Non Automated Tasks

What is the business impact (measured in revenue, cost, etc) for portfolio activities that require manual support (releases, testing, etc). This metric is used to assess business / operational risk of how much the business could adversely impacted by human error, and can be used to quantify the impact of adopting automation to reduce operational / business risks.

Change request flow rate (push/pull)

Comparison of change request completion rate to change request entry rate. This metric is used to assess if the capacity of the portfolio / service desk can keep up with the rate at which change requests are received. Assess if process changes and/or automation improve the flow rate / capacity.

Lead time to deploy new infrastructure / platform capabilities

How long do development / engineering teams need to wait until they are provided with environments needed for development or testing. This metric can be used to demonstrate the business impact and increased capacity achieved by automated provisioning of infrastructure to engineering and development groups – ex: self-service portals.

Resolution time of key system outages

How frequently do critical system outages occur (SEV1s) and when an outage does occur, what is the time necessary to recover (Mean Time To Resolution – MTTR). This metric is commonly used to demonstrate the business impact of addressing technical debt to improve reliability of code, or adopting automated testing to reduce critical defects.

Root Cause Analysis Time

When a critical system outage occurs, how long does it take to determine the root cause? This metric can be used to assess process health and overall shared understanding of systems architecture and capabilities across the teams & engineers that support them. Assess if activities for knowledge sharing to understand legacy code & platforms result in faster response time to service outages.

Green To Red Day Ratio

The ratio between Green Days (no critical events) and Red Days (critical events occur). Can be used as a high level assessment / trend to determine if activities to improve architecture, code quality, support processes and/or operational readiness are providing desired benefits / impacts.

Duration of downtime / maintenance windows

How long must the system / platform be taken down for upgrades? This metric can be used to demonstrate the impact of technology, process and/or platform changes to enable continuous operations, Blue/Green deployments and/or other strategies to reduce planned downtime.

Frequency of downtime / maintenance windows

How frequently must the system / platform be taken down for upgrades? This metric can be used to demonstrate the impact of technology, process and/or platform changes (aka. Serverless) to reduce the frequency of when system maintenance interrupts operations.

Incidence of & type distribution of escaped defects

What types of escaped defects are seen most frequently – defects are mapped by functionality, architecture, APIs, etc. This metric is used to assess if activities to reduce technical debt, increase automated testing, and/or adopting technical standards reduces the frequency of specific types of defects.

Customer satisfaction (Net Promoter Score) of service provided

Would customers recommend the service / product based on the experience they receive? This customer satisfaction can be applied to products within the portfolio (would end-users recommend them) and/or to the level of service / experience provided from IT operations & support desk activities.

Customer satisfaction (Net Promoter Score) of service available

Would customers recommend the service / product based on the options available (without necessarily consuming them)? This metric is used to assess if the product and/or IT operations are offering the right options / features to incent customers to engage – understand if the product / service offerings align to what customers actually need / want.

Flow efficiency to complete work items / change requests

The ratio of time spent actively working on an item (value-add) vs. the total cycle time necessary to complete a work item. This metric is used to assess if activities to increase efficiencies, automation, and/or reduce blockers provide intended benefits to deliver value faster – also can assess if there is too much work-in-progress.

Cost to complete a work item / change request

What is the average cost to complete a work item. Assess if the cost to complete a work item is aligned to the business value it provides when completed, and/or how many business transactions are necessary to offset the cost to complete the item / change request.

Service automation / orchestration cost

What is the cost (estimated / actual) to automate processes / routines. Used to assess the cost savings (business value) of automating repeated manual processes and demonstrate return on investment from automation activities.

Adoption Rate

How many users consume each available product, feature and/or service. Enable assessment of activities to promote adoption and also to determine return on investment by adoption of products, features and/or services against the business goals they enable.

Cost savings / cost avoidance by self service

Cost savings achieved by transitioning manual tasks to automated tasks. Identify ways to reduce cost or fund new investment / innovation by automating low value repeated tasks with self service options.

License utilization rate

How many licenses / instances available have been allocated. This metric enables assessment of adoption and/or standards alignment activities – determine if sufficient licenses exist for expected capacity – determine if license pool can be adjusted to optimize license cost.

Incidence of job / role overloading

How frequently do team members step outside of their assigned roles to complete the work at hand? Assess the collaborative culture of the portfolio and if team members are able and desire to do what it takes to enable successful delivery by working together.

Incidence of SLA / SLE non-compliance

How frequently do products in the portfolio violate SLAs / SLEs? Assess if activities to improve SLA / SLE compliance provide intended benefits. Enable internal calculation / forecasting of penalties due as a result of non-compliance. Determine if SLAs / SLEs are sustainable given observed activities or if calibration is necessary.

Business Impact of SLA / SLE non-compliance

Determine the business impact (cost per minute, etc) of SLA / SLE non-compliance. This metric is used to assess if investments / efforts made to improve SLA / SLE compliance are justified given the expected penalties of non-compliance.

Operational Automation Ratio

The ratio of automated vs. manual activities (runbook activities) necessary to operate the systems in the portfolio. Used to assess operational risk and readiness from being too dependent on human tasks for deployments, operations & testing. Supports the demonstration of trends to increase automation over time.

Cost / time of duplicated efforts

How much time is spent by multiple groups / teams doing the same job / task. Understand the cost / waste of duplicate efforts and then demonstrate the impact of aligning to common (ideally automated) processes and/or common architectures and tools.

Percentage of deployments to managed platforms

How many systems / applications in the portfolio are deployed to managed / standard platforms. Used to assess alignment / adoption to portfolio standards that seek to improve efficiency and also improve operational readiness.

System / Application performance analytics

What are the observed utilization statistics / analytics (compute, network, storage) for each system in the portfolio. This metric enables understanding of resource consumption by each system to enable capacity planning.

Frequency / pattern of performance hot spots

What patterns cause system utilization statistics / analytics (compute, network, storage) to spike. Enable identification of expected / normal spikes vs. unexpected spikes which could be leading indicators of performance issues, lost transactions, and/or system outages.

Number of managed platforms available

The number of managed platforms / configurations available in the portfolio to which capabilities can be deployed. Assess & monitor if the portfolio provides sufficient deployment platforms / architectures given the business needs, portfolio size and portfolio complexity. Guard against excessive technical complexity (too many platforms / tools) within the portfolio.

Number of non-complaint environments in production

The count of environments / applications in production that do not comply with architecture, platform or security standards. Assess alignment to required standards, establish accountability for application owners (architects, etc) to align to standards – evaluate effectiveness of programs to guide adoption (training, etc).

Number of automated processes within the portfolio

The count of jobs / configurations within the portfolio that are automated. Used to measure the growth of automation in portfolio operations which in turn should enable business benefits such as: increased flow, increased efficiency, higher quality and improved operational readiness.

Project / Product Infrastructure & Automation Maturity

A score determined at the product or team level within the portfolio by adoption of desired standards inclusive of processes, architecture & automation. Enable transparency of adoption to desired standards, and provide guidance to teams / engineers regarding “what good looks like”.

Completion rate of documented improvements

A count of how many improvements have been completed by members of the portfolio. This metric encourages a culture of accountability that focuses on continuous improvement by all teams – enable recognition for activities that improve products within the portfolio and the technologies and processes that support the portfolio.

Test coverage – all levels of the stack

The percentage of portfolio functionality that is supported by automated testing – automated testing is reported inclusive of unit, integration, regression, and infrastructure testing. Enable assessment of efforts to increase test coverage to decrease escaped defects, improve operational readiness, etc.

Operational Readiness Score / Risk Score

A compound metric to determine overall operational risk to portfolio systems / products by assessment. Demonstrate the impact of activities to increase testing and/or automation to reduce operational risk.

Failed login attempts

The number of times users cannot log into systems within the portfolio. Sounds simple but if users cannot log in (especially new users) then they will be unable to use features to maximize business value. Assess the impact of strategies / design to simplify login activities.

Help / Support Calls

How frequently do users have to call / request support. Assess the usability (UX) & design applications and/or efforts to improve the usability (UX) & design of applications. Assess if activities improve design (UX) reduce support calls which in turn results in higher application / system rating by users.

Revenue / Profit Per Customer

How much revenue (or profit) does each end-user / customer generate in a period of time. Assess if features and/or campaigns to bring in more revenue from customers (or groups of customers) have desired outcomes and/or determine why campaigns do not generate desired results. Determine if revenue / profit per customer justifies level of ongoing investment given business goals.

Cost of Spares / Infrastructure on Hand

What is the holding cost (inventory) of spare infrastructure necessary to sustain operations. Assess the financial impact of transitioning from on-premise operations to managed services and/or cloud providers for which on-premise inventory / spares is not required.

Sales Pipeline – potential new sales / new customers

How many new customers are being targeting for sales? Do updates to systems / products within the portfolio have expected impacts on pipeline – ex: growing pipeline or closing new customers in the pipeline. Are there key dates that new features need to be available to close new customers in the pipeline?

Portfolio Investment Ratio

How much of the total portfolio funding is committed to new development & operations vs. what is in reserve. Assess if the portfolio has sufficient funding in reserve for surge capacity and/or strategic projects necessary to mitigate business risk and/or explore new opportunities.

Application response time / page load

How long does it take a page (or functionality) to load within an application. Assess if applications in the portfolio are aligning to usability standards to provide the best end-user experience. Determine if sufficient infrastructure capacity is available to sustain page load / performance and also to assess if efforts to improve architecture and/or performance provide expected benefits.

Stakeholder Participation Rate

The number of stakeholders necessary to sign off on a portfolio work item (feature). Used to assess if there is enough stakeholder collaboration based on overall business goals & business risks, and guard against potential groupthink scenarios where there are too many stakeholders involved (which can slow progress) given business goals / risks.

Backlog Dwell Time

The average time that a portfolio work item (feature) spends in the portfolio backlog. This metric is used to assess if portfolio planning practices avoid excessive up-front planning (maintaining a backlog that is too long) – excessive planning increases rework as emergent business needs result in stale work items.

Support Call Response Time

The time necessary to resolve support issues. Support response time is a contributor to overall service / application rating (nobody likes to be on hold) and can be used to assess the effectiveness of design / UX activities that are intended to reduce incidence / time required for specific types of support calls.