

ZERO TO HUNDRED DEPLOYS A DAY

# MILESTONES CHECKLISTS



# Phase 1 – Deploy Once a Week

	Measurements	Development Practices	Communications	Culture
<b>Milestones</b>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Track Deploy Frequency</li> <li><input type="checkbox"/> Track Change Lead Time</li> <li><input type="checkbox"/> Drill down on activities that drive Change Lead Time</li> <li><input type="checkbox"/> Achieve Deploy Frequency of 1x week</li> <li><input type="checkbox"/> Achieve Change Lead Time of &lt; 5 days</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Use developer branches</li> <li><input type="checkbox"/> Have an always deployable release branch</li> <li><input type="checkbox"/> Have a process to accept or reject changes to release branch</li> <li><input type="checkbox"/> Use pull request for new changes</li> <li><input type="checkbox"/> Have a process for merging new changes</li> <li><input type="checkbox"/> Achieve 50% unit test coverage</li> <li><input type="checkbox"/> Unit tests are at most 20% flakey</li> <li><input type="checkbox"/> Unit tests run on every commit</li> <li><input type="checkbox"/> Have a mostly automated CD pipeline that can build a release from release branch</li> <li><input type="checkbox"/> Have a well-defined production environment</li> <li><input type="checkbox"/> Have basic observability system</li> <li><input type="checkbox"/> Designate deploy owner. Recommendation for this phase: a SRE or infra-minded developer</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Have a well-defined merge window</li> <li><input type="checkbox"/> Have a well-defined production deploy window</li> <li><input type="checkbox"/> Designate a deploy reviewer</li> <li><input type="checkbox"/> Have a process for communicating releases to developers, managers, PMs</li> <li><input type="checkbox"/> Track high-level work going into a deploy in issue tracker</li> <li><input type="checkbox"/> Have a real-time chat tool for developers and those who own the deploy</li> <li><input type="checkbox"/> Have a way, even if cumbersome, to know what's been deployed</li> <li><input type="checkbox"/> Have documentation on the deployment process</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Have a weekly or bi-weekly sprint to agree on task size and tasks for this window</li> <li><input type="checkbox"/> Have an automated CI against the release branch, with the results visible to the team</li> <li><input type="checkbox"/> Designate a disturbed role</li> <li><input type="checkbox"/> Include Product Managers in sprint planning</li> <li><input type="checkbox"/> Implement feature flags</li> <li><input type="checkbox"/> Have a recurring planning meetings, code reviews, and weekly demos</li> <li><input type="checkbox"/> Budget engineering time to keep the release branch builds passing</li> <li><input type="checkbox"/> Make DORA / Accelerate metrics data available, with context, to execs</li> <li><input type="checkbox"/> Report on uptime or equivalent metric for applications</li> <li><input type="checkbox"/> Share high-level information about large chunks of functionality that are shipping</li> </ul>
<b>Tooling to support</b>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Set up tooling to:             <ul style="list-style-type: none"> <li>▪ DORA metrics tracking</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Set up tooling for:             <ul style="list-style-type: none"> <li>• Source control</li> <li>• Automated testing / CI</li> <li>• Deployment / CD</li> <li>• Observability</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Set up tooling for:             <ul style="list-style-type: none"> <li>• Issue tracking</li> <li>• Chat</li> <li>• Deployment tracking</li> <li>• Knowledge management</li> </ul> </li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Set up tooling for:             <ul style="list-style-type: none"> <li>• Feature flagging</li> </ul> </li> </ul>

# Phase 2 – Deploy Once a Day

	Measurements	Development Practices	Communications	Culture
<b>Milestones</b>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Track Deploy Frequency and Change Lead Time</li> <li><input type="checkbox"/> Define Change Failure rate per your needs</li> <li><input type="checkbox"/> Track Change Failure Rate by codebase and flag</li> <li><input type="checkbox"/> Track MTTR</li> <li><input type="checkbox"/> Have ability to quickly see outliers on your Failure Rate and MTTR</li> <li><input type="checkbox"/> Achieve Deploy Frequency of 1x day</li> <li><input type="checkbox"/> Achieve Change Lead Time of &lt; 2 days</li> <li><input type="checkbox"/> Achieve Change Failure Rate of &lt; 15%</li> <li><input type="checkbox"/> Achieve MTTR of &lt; 1 day</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Use pull request for changes</li> <li><input type="checkbox"/> Have a process like PRGB for code reviews</li> <li><input type="checkbox"/> 75+ % unit test coverage</li> <li><input type="checkbox"/> Unit tests are &lt;10% flakey</li> <li><input type="checkbox"/> Unit tests gate deploys</li> <li><input type="checkbox"/> Have end-to-end tests for critical paths</li> <li><input type="checkbox"/> End-to-end tests run against a production-like environment</li> <li><input type="checkbox"/> Have a well-defined pre-production environment, with enforced and automated soak time</li> <li><input type="checkbox"/> Set up automated process where deploys are triggered and executed in a CD pipeline and take &lt; 2 hours</li> <li><input type="checkbox"/> Have advanced observability with automated alerts</li> <li><input type="checkbox"/> Well-defined on-call roster with clear escalation chain</li> <li><input type="checkbox"/> Deployers have access to observability metrics and can diagnose production systems</li> <li><input type="checkbox"/> Track errors in logs or via tool</li> <li><input type="checkbox"/> Use feature flag</li> <li><input type="checkbox"/> Use configuration-as-code tool</li> <li><input type="checkbox"/> Designate developers as deploy owners</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> All developers know how to trigger deployment process</li> <li><input type="checkbox"/> Have a communication channel to share when a deploy is happening</li> <li><input type="checkbox"/> Everyone has access to see the important health metrics</li> <li><input type="checkbox"/> Have a well-defined and automated escalation policy</li> <li><input type="checkbox"/> Have a well-defined cadence for daily releases</li> <li><input type="checkbox"/> Have a way to share progress toward a larger goal with PMs and Managers, and high-level goals with stakeholders</li> <li><input type="checkbox"/> Have a communication channel between developers and infrastructure engineers</li> <li><input type="checkbox"/> Inform Customer Support when bug fixes or support issues have been deployed</li> <li><input type="checkbox"/> Break work into small batches roughly equal to one deploy</li> <li><input type="checkbox"/> Trace deploys to issues</li> <li><input type="checkbox"/> Have a way to spin up new rooms for incident response</li> <li><input type="checkbox"/> Have central docs for common scenario troubleshooting</li> <li><input type="checkbox"/> Have well-defined conditions for triggering rollback decision</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Deploy every pull request one at a time, so you can review batch size while doing code review</li> <li><input type="checkbox"/> Developers are empowered to deploy their own changes to all the environments</li> <li><input type="checkbox"/> Developers have enough metrics and an understanding of their norms to verify deploys</li> <li><input type="checkbox"/> Developers understand how to escalate and perform rollback</li> <li><input type="checkbox"/> Enable “continuous demo” by looping in PM or Design when work has shipped</li> <li><input type="checkbox"/> Adopt six-week cycles to get Devs, PM, and Design to think in smaller, shippable units</li> <li><input type="checkbox"/> Managers continuously measure the team’s Accelerate metrics</li> <li><input type="checkbox"/> Managers invest in tooling that allows identification of bottlenecks and prioritize removing them</li> <li><input type="checkbox"/> Work with Marketing to coordinate feature flag launches with marketing launches</li> <li><input type="checkbox"/> Have a consistent way to disseminate product changes info to Sales and Support</li> </ul>

# Phase 3 – Deploy One Hundred Times a Day

	Measurements	Development Practices	Communications	Culture
<b>Milestones</b>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Track DORA metrics by team (e.g., programming language, front-end vs. back-end, etc.)</li> <li><input type="checkbox"/> Track org-wide anomalies and report on org-wide trends and performance</li> <li><input type="checkbox"/> Achieve Elite level performer per DORA:</li> <li><input type="checkbox"/> Deploy Frequency of 10+ per day</li> <li><input type="checkbox"/> Change Lead Time of &lt; 1 days</li> <li><input type="checkbox"/> Change Failure Rate of &lt; 15% of changes</li> <li><input type="checkbox"/> MTTR of &lt; 1 hour</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Implement release trains</li> <li><input type="checkbox"/> Implement canary deploys</li> <li><input type="checkbox"/> Implement pre-production environments with approvals</li> <li><input type="checkbox"/> Implement smoke tests</li> <li><input type="checkbox"/> Implement automated rollbacks</li> <li><input type="checkbox"/> Implement drift detection</li> <li><input type="checkbox"/> Implement centralized tracking “single pane of glass”</li> <li><input type="checkbox"/> Developers have access to custom deploy workflows engine that can automate integrations with other important systems</li> <li><input type="checkbox"/> Developers own their deploys</li> <li><input type="checkbox"/> Have a dedicated engineering team owning pipelines, tooling, and platform.</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Have a fully automated high-level deploy notifications tailored for all functions: Support, PM, Sales, etc. and executives.</li> <li><input type="checkbox"/> Have a fully automated and self-explanatory process for how and when to deploy and rollback</li> <li><input type="checkbox"/> Have notifications that alert teams when drift has breached a set level</li> <li><input type="checkbox"/> Have a well-defined and published “train schedules”</li> <li><input type="checkbox"/> Have a way to tell a developer where their changes are in a deployment pipeline.</li> </ul>	<ul style="list-style-type: none"> <li><input type="checkbox"/> Developers believe in the institutional safety net: bad changes are highly unlikely to make it to production, and if they do the blast radius is small</li> <li><input type="checkbox"/> Developers are confident they can make a mistake and ship a fix in no time</li> <li><input type="checkbox"/> Developers feel deploys are a non-event</li> <li><input type="checkbox"/> PMs and Designers work in smaller increments just like Devs</li> <li><input type="checkbox"/> Managers feel the team is punching above its weight</li> <li><input type="checkbox"/> Management focuses on scaling issues vs. ability to ship.</li> </ul>