coveros

# THE MODERN SDLC CHECKLIST

**An advanced and efficient software development lifecycle (SDLC) requires the following critical attributes. Use this checklist to help track your organization's SDLC maturity.**

## AGILE DEVELOPMENT

**Work incrementally as a team to define, design, develop, and test software, breaking down the barriers between the users, business, and development pieces of an organization.**

- ☐ Product owner defines and prioritizes user stories with teams.
- ☐ Product owner works in the best interests of the business and users.
- ☐ Cross-functional teams estimate their own work and balance new features with maintenance and technical debt efforts.
- ☐ Retrospectives are used to continuously improve the development process. Teams openly assess what works and what doesn't work and experiment to make the process better able to deliver business value.
- ☐ Production quality, releasable software is the definition of done for user stories, iterations, and defined releases.
- ☐ A modular, well-structured, and decomposed approach to development is used to separate concerns and provide a way to independently and rapidly deploy components.
- ☐ The organization takes advantage of cloud native development through capabilities such as Serverless, Scaling on Demand, and Infrastructure/Platforms/Software as a Service.

### AGILE DEVELOPMENT RESOURCES

**Eliminating silos in scrum: How to avoid a major agile failure**

coveros

ELIMINATING SILOS IN SCRUM:

How to Avoid a Major Agile Failure

Jeffery Payne | Chief Executive Officer | jeff.payne@coveros.com

**Effective agile leadership starts with self-care** READ BLOG ▶

## DEVOPS

**Align development and operations activities to create an automated approach to continuously and rapidly deliver capabilities to business customers.**

- □ DevOps processes have been adopted that result in continuous coordination between development and operations (and everyone in the end-to-end product lifecycle).
- □ Automated pipelines exist for pushing code through testing platforms and into production.
- □ Value stream mapping is used to identify and drive improvements to the overall DevOps process.
- □ Automated testing is integrated into both Continuous Integration (CI) and Continuous Development (CD) at each stage in the process with a focus on shifting these activities left as much as possible.
- □ Ability to do continuous delivery and continuous deployments of changes. This supports both the need to build confidence in the software and the ability for urgent deployments to be done as quickly as possible.
- □ Using platform engineering to create and maintain standard Continuous Delivery (CD) practices within an organization.

### DEVOPS RESOURCES

**No more excuses: Overcoming key challenges to DevOps test automation**

**Value Stream Mapping gone wrong** READ BLOG ▶

## CONTINUOUS TESTING

**Leverage agile testing and automation to continuously test applications during development and delivery.**

- □ Test automation is a key QA driver that is primarily done below the User Interface (UI) leveraging the Agile Test Automation Pyramid approach.
- □ Test staff is fully integrated into cross-functional teams that plan, estimate, develop, and test together day-to-day.
- □ Developers are responsible for testing their own code at the unit level as well as participating in whole-team quality practices.
- □ Testers pair with developers explicitly on user stories and work together day-to-day.
- □ Automated quality gates are established with business input to provide continuous feedback on the alignment of application capabilities and quality with business requirements.

### CONTINUOUS TESTING RESOURCES

**David Bernstein: 'All the great virtues of software come with testability'**

READ BLOG ▶

**3 big things organizations get wrong about software testing**

READ BLOG ▶

## APPLICATION SECURITY

**Ensure the security of software applications, platforms, and delivery pipelines.**

- □ Architectural risk analysis and threat modeling are used to build security in and design downstream security testing.
- □ Defense in depth is integrated into the process by utilizing code analysis, dynamic testing, validating the security of open source components, managing secrets, and security of platforms on-premise or in the cloud.
- □ Automated security tools are shifted left and are part of the continuous development, test, and delivery process.
- □ Security education for business, development, and test team members is continuous to help the team understand and integrate security.

### APPLICATION SECURITY RESOURCES

**Don't overlook insider threats - and more cybersecurity lessons**

READ BLOG ▶

**The key to AppSec is early detection. Are you ready for a security checkup?**

READ BLOG ▶

**Contact Coveros** coveros.com | training.coveros.com ✉ info@coveros.com ☎ 929.341.0139