Do Now

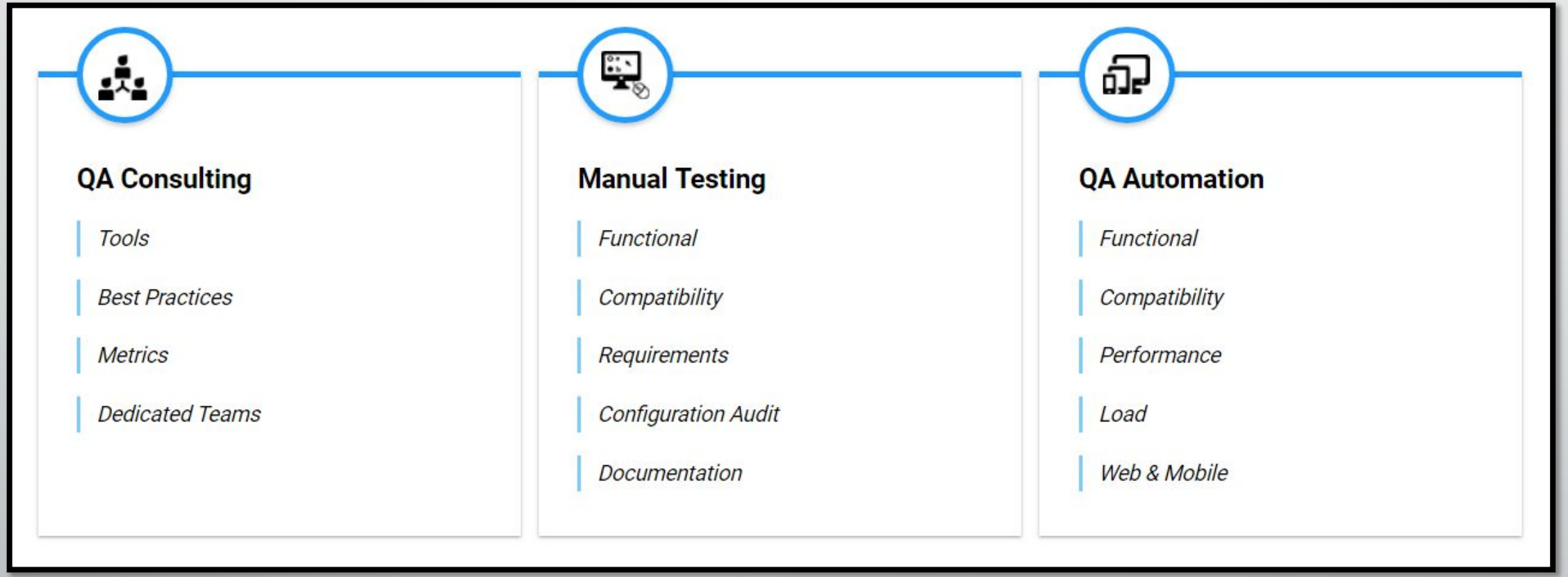
Open your project:

- Specification sheet
- Design document
- All programming files
- Run your program to be user ready

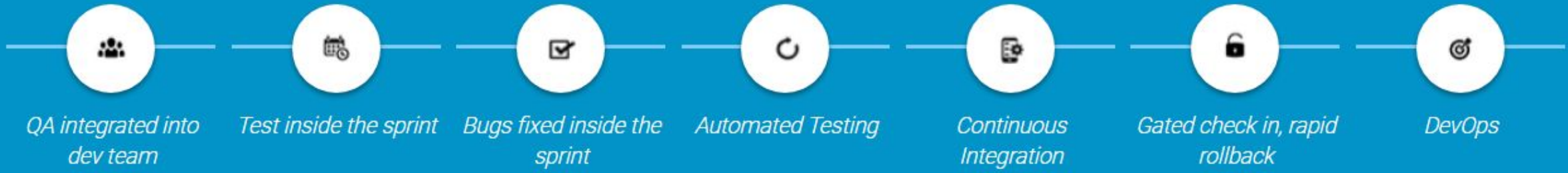
Welcome!

Quality Assurance Team

Testing quality of applications so they function, perform, and scale properly



Key components of Coherent's Agile testing include:



Test Planning

- Test Strategy – a good strategy is product specific, risk focused, diversified & practical
- Test Plan - strategy + logistics
- Test Organization – roles and responsibilities

Test Design

- Analyze requirements
- Determine coverage – risk based & practical
- Determine testing procedures: automated, manual, exploratory, load
- Requirements traceability matrix (RTM)

Test Build

- Create manual test cases
- Create automated test cases
- Define test data
- RTM

Test Execution

- Execute manual tests
- Execute automated tests
- Record defects
- Verify RTM

Agile

- a method of project management, used especially for software development, that is characterized by the division of tasks into short phases of work and frequent reassessment and adaptation of plans.

Agile software development

- **Agile software development** is an approach to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-functional teams and their customer(s)/end user(s). It advocates adaptive planning, evolutionary development, empirical knowledge, and continual improvement, and it encourages rapid and flexible response to change.

Agile software development principles

The *Manifesto for Agile Software Development* is based on twelve principles:

- Customer satisfaction by early and continuous delivery of valuable software.
- Welcome changing requirements, even in late development.
- Deliver working software frequently (weeks rather than months)
- Close, daily cooperation between business people and developers
- Projects are built around motivated individuals, who should be trusted
- Face-to-face conversation is the best form of communication (co-location)
- Working software is the primary measure of progress
- Sustainable development, able to maintain a constant pace
- Continuous attention to technical excellence and good design
- Simplicity—the art of maximizing the amount of work not done—is essential
- Best architectures, requirements, and designs emerge from self-organizing teams
- Regularly, the team reflects on how to become more effective, and adjusts accordingly