

Levels and checkpoints

The levels for *Automation standards* are typified as follows:

- **Forming:** *Test automation code is centrally managed*
- **Norming:** *Test automation code is configurable and reviewed*
- **Performing:** *Test automation adheres to de facto standards*

Please find the checkpoints below.

Forming

1. Information on how to use the automation is available and accurate.
2. Test automation code is centrally maintained in a repository with the use of a distributed version control system.
3. Test automation code is debugged and tested.

Norming

1. Automation does not contain “hard-coded” values referring to environmental parameters, such as absolute magic numbers, file paths, URLs, etc.
2. A consistent coding style for automation is used, focused on readability and reusability. Style guides and programming standards support modular and standardized development of test scripts.
3. Test automation code is reviewed and/or programmed in pairs.
4. Rubber ducking is used to ensure simplicity and knowledge transfer.

Performing

1. Automation is designed according to de facto design patterns (behavioral, creational, etc.).
2. Existing (working) code is refactored into smaller components when necessary.
3. Continuous Integration (including but not limited to unit tests, integration tests, automated user interface tests) is practiced.

[Terug naar TI4Automation](#)