

# DOs and DON'Ts

## Spring Cleaning

### Activate Your Competitive Advantage

Leverage our 20+ years testing software.

PQA Testing is the leading Canadian solutions provider of independent, objective and unbiased software testing services.

[Invite us to help](#) make your software testing the best it can be.

1-855-455-7725  
[sales@pqatesting.com](mailto:sales@pqatesting.com)  
<http://www.pqatesting.com>



/PQATesting



/PQATesting



/company/PQA

# Defects Spring Clean-Up

## *DOs and DON'Ts*

### DOs

- ✓ Do set time aside at least once a year for this task. If you don't set time aside and ignore it for too long, it will become a daunting task.
- ✓ Do look at the lowest priority issues first. If it was marked trivial three years ago, it's likely to be even less relevant today.
- ✓ Do a high-level re-test of issues that are stale. They may have actually already been fixed in recent releases or feature changes may have made their priority higher or lower.
- ✓ Do consider gathering a group of stakeholders together and step through a large chunk of issues. It can be the fastest way to resolve them.
- ✓ Do set criteria for keeping and closing issues before starting the spring clean-up. It's easier to make decisions when everyone agrees on the methodology.
- ✓ Do advocate for an un-loved defect to get put into an active backlog if it is reasonable to do so.
- ✓ Do re-assign stale defects to the appropriate person and set a timeline for corrective actions to occur.

# DOs and DON'Ts

## Spring Cleaning

### Activate Your Competitive Advantage

Leverage our 20+ years testing software.

PQA Testing is the leading Canadian solutions provider of independent, objective and unbiased software testing services.

[Invite us to help](#) make your software testing the best it can be.

1-855-455-7725  
[sales@pqatesting.com](mailto:sales@pqatesting.com)  
<http://www.pqatesting.com>



/PQATesting



/PQATesting



/company/PQA

# Defects Spring Clean-Up

## *DOs and DON'Ts*

### DON'Ts

- ✘ Don't delete something simply because it's trivial or old. There are many factors that might still make an older issue relevant.
- ✘ Don't keep something just because it's someone's pet bug. Hold the constant criteria assessment for keeping or closing all bugs and tests.
- ✘ Don't delete issues, close them. This makes them still available to learn from and search back to in the future.
- ✘ Don't exclude closed items from searches. If the bug is found again, you want your duplicate search to bring back the closed item.
- ✘ Don't advocate to fix old issues without finding out if the work-arounds that have been produced are now core features in the application.
- ✘ Don't renumber defects.
- ✘ Don't keep things open just because they are a valid bug. If it isn't a high enough priority to get fixed, it's enough that it's in the system for reproducibility, even if closed. Utilize Deferred or Backlog statuses when they are available.