
50
50 TIPS
FOR SMARTQA
THIRUVENGADAM ASHOK

50 Tips to Smart QA:
A Transformative Guide for Agile & Insightful Software Testing

Step into the world of smarter, faster, and more innovative Quality Assurance (QA) practices with 50 Tips to Smart QA. This comprehensive guide is your ultimate toolkit to revolutionise how you approach software testing, offering practical, actionable strategies to streamline processes, maximise efficiency, and foster creative problem-solving.

Key Highlights:

Do Less, Achieve More: Discover minimalist testing strategies to focus on what truly matters without spreading thin.

Prevention Over Correction: Learn techniques to anticipate and prevent issues, saving time and effort.

Beauty in the Details: Elevate your QA deliverables with aesthetic, well-structured reports and plans that resonate.

Leverage Technology & Adaptability: Harness automation and tools to simplify workflows while remaining agile and ready to pivot.

Human-Centered Testing: Emphasise empathy and user-focused testing to ensure a seamless and enjoyable end-user experience.

Relentless Simplification: Break down complex problems and refine solutions for optimal clarity and impact.

This book goes beyond traditional QA principles, encouraging testers to think like scientists, act like engineers, and feel like artists. Each tip is designed to inspire innovation, foster continuous improvement, and balance the art and science of software testing.

Whether you're a seasoned QA professional or just starting in the field, 50 Tips to Smart QA is the guide you need to stay ahead in the fast-paced world of software development

About the author

Thiruvengadam Ashok is the CEO of STAG Software Private Limited & Co-Founder of Pivotrics, based in Bengaluru, India. Ashok has dedicated his career to the pursuit of quality assurance in software, continuously evolving his approaches to match the needs of modern systems. He is the creator of HyBIST, an innovative approach to SmartQA that has revolutionised how teams approach hypothesis-driven testing.

Ashok's professional life is deeply intertwined with his personal philosophy. A passionate ultra-distance runner and long-distance cyclist, he applies the principles of endurance and exploration to his work, constantly seeking out new ways to improve software quality. He is also an avid wordsmith, using his love of language to weave both poetry and technical innovation into his life's work.

He holds an M.S. in Computer Science from the Illinois Institute of Technology, a Bachelor's degree in Electronics and Communication Engineering from the College of Engineering, Guindy, and a Postgraduate Diploma in Environmental Law from the National Law School of India University, Bangalore. His life maxim—"Love what you do & Do only what you love"—is reflected in everything he undertakes, from professional projects to personal passions.

Disclaimer:
The information contained in this book is for educational and informational purposes only. While every effort has been made to ensure the accuracy of the content, the author and publisher make no representations or warranties regarding the completeness, accuracy, or applicability of the information provided. The strategies and methodologies described are for informational purposes and should be adapted to individual circumstances as necessary.

Trademarks*:*
All product names, logos, and brands mentioned in this book are the property of their respective owners. Use of these names, logos, and brands does not imply endorsement.

HyBIST is the intellectual property of STAG Software Private Limited.

Edition: First edition, 2025

1

"Don't do work. Prevent."

Smartness is about thinking well,
so as to not do. It is not avoiding it,
but about quashing the need for it.

With smarter analysis of change,
should I really regress this?

Can I inject code to self test,
so that I don't have to test it?

2

"Do less."

Smartness is about doing minimally.
Not due to of lack of time/ effort,
but being sharply focused
so as to not spread thin.

For example,
what may be minimal data sets
to ascertain correctness,
minimal environments to test on.

3

"Do just as much."

It is a kind of mix of (1) & (2).
How much to do, what can be avoided.
The continuous sharp sensitivity
to be just right enough.

Selecting the right set for scenarios from a larger set for (say) a given customer.

4

"Do anything beautifully."

Be it writing a document,
presenting a report,
or test data sets,
let it be beautiful.

Great aesthetics nourishes the soul
enabling us to do great work.

5

“Do quickly.”

Chunk tasks, get it done quickly.

Don’t stretch an activity,
strive to complete quickly
so that you can get feedback,
learn and refine, and of course
get work done faster!

6

“Detect at the earliest. Prevent if possible.”

Guess this is self explanatory!

7

"Exploit technology."

Automate maximally.

Use tools to
setup, execute, compare, manage,
check, monitor etc. to help you do,
to help you gain better insight.

8

"Adapt, adjust, adapt, adjust.."

Be like the water that flows.

Not fixated open,
to constantly adjust and refine
strategy, plan, scenarios, scripts,
tools, priorities, understanding.

9

"See the mirror constantly."

Setup measures for feedback,
to constantly analyse
and stay on course continually.

10

"See consciously, see unconsciously too."

Observation is a key skill for test folks, to judge, to see patterns, to connect the dots, to enhance test strategy and action.

11

“Add, delete, refine. Evolve.”

Utility of anything is never fixed,
everything looses shine with time.
Constantly egg yourself to evolve.

For example test cases over time
will stop finding issues,
some flows may never be done
requiring continuous evolution.

12

"Empty yourself periodically."

Make way for new thoughts/ideas
by discarding existing ones, periodically.

Consciously discharge to recharge.

13

“Focus on outcome, enjoy the journey.”

Repetitive testing can be boring.

Enjoy the journey
by observing little nuances
that is different every time.

14

"Doing is great, but value matters."

It is not just about doing activities, but producing great outcomes.

Doing excessive testing without demonstrating high utility to end customers is an exercise in futility.

15

“Be mindful, immerse yourself.”

Being in a flow allows you to be very sharply observant enabling you to deliver great work and making it enjoyable.

16

"Leverage other peoples' work."

Don't do what has been done earlier.

Leverage assets aggressively,
be it tools, frameworks, scripts,
scenarios/cases/data, strategy/plan.

Before embarking on an activity,
check if it has been done before.

17

"Blend left and right."

It is not just about using
the logical left brain or the creative right.

It is about a harmonious combination
of logical/scientific left brained thinking
with the creative right that makes testing
super effective and super efficient.

18

"Be rational, but trust your gut too."

Staying engaged, being immersive enables deep unconscious learning building gut feel.

Staying logical and rational is important, but don't underestimate the power of gut feel.

19

"Analyse situations logically, but act on the choices."

We analyse situations
(say) 'why did this occur'
and come up with a list of choices.

It is necessary to use, and on these choices to realise the full benefit of logical thinking.

20

"Stay focused and purposeful."

Purpose gives power to focus.

For example,
looking for specific issues
sharpens approach & test cases.

21

"Focus is great, but meander too."

Focus enables us to be purposeful,
but it is like a horse blinder.

Some bit of meandering,
observing the system at large
while performing a focused test
improves overall understanding,
enabling us to refine to do better.

22

"Look outside, learn from other disciplines."

Stick robot was inspired by insect and velcro by lizard feet.

Read, watch, experience things outside of one's discipline to innovate.

23

“Stay curious, question, explore.”

Testing is scientific exploration,
driven by curiosity and
fuelled by intense questioning.

24

"Decompose well, the problem solves itself!"

Well the problem may not solve itself completely, but good decomposition of a problem is very important to solving it.

Decomposing 'what-to-test' into various granular entities like screens, features, flows and 'what-to-test-for' into different types of issues and then test types enables clarity of problem & solution.

25

"Relentlessly simplify."

Albert Einstein said
"If you can't explain to a six year old,
you don't understand it".

Relentlessly simplify to sharpen clarity
and broaden understanding,
a key ingredient of great testing.

26

“Write less, accomplish more.”

Documentation is useful as history, but only when present is meaningful and valuable.

So think better, write just enough, focus on doing great now.

27

"Sift continuously to sharpen clarity."

It takes a lot of sifting
to separate gold from mud.

To understand a system,
play with it, read, explore, discuss,
repeat via varying,
discard what is not needed,
repeat until time runs out.

A deep understanding of
context, usage and system
is central to great testing.

28

"Think like a scientist, do like an engineer, feel like an artist."

Deep scientific thinking,
pragmatic implementation,
enjoying the aesthetics of doing & outcomes
is a brilliant combination that makes
activities enjoyable and outcomes valuable.

29

"Visualise in the mind's eye."

Seeing the system flows,
the perturbation of a modification,
structure of systems in one's mind eye clearly
is the ultimate result of great understanding
and makes probable issues stand out.

30

"Fly high to abstract, stoop low to see details continually."

See the forest for trees
to gain systemic understanding,
drill down to look at individual leaves
to understand the details.

Repeat these in an endless cycle to
understand system & context well.

32

"Keep your cup half empty."

"Exactly" said Master Ryutan. "You are like this cup; you are full of ideas. You come and ask for teaching, but your cup is full; I can't put anything in. Before I can teach you, you'll have to empty your cup."

Create space in your mind
to absorb, to learn, to understand,
and defer what is not needed now.

32

"Problem solving is a mix of techniques, principles, heuristics."

Apply techniques to solve problems, employ principles to chart direction and use heuristics to guide you.

There is no perfect formula, nor is solution only from prior experiences. It is a judicious combination of techniques, principles & heuristics(guidelines).

33

"Trust what you do, prove what you have done."

It is not just about "trust me",it is about demonstrating proof in what you have done.

Justifying test adequacy is one key application of this in our discipline.

34

"Understand the behavior outside, know how it is composed inside."

It is not 'black' or 'white'.

It is about knowing external behaviour and also internal structure in terms of architecture, data/control flows, interfaces and so on.

35

"Never forget that human uses the system."

Testing is not
a clinical examination of the system,
it is being empathetic so that a
human end user benefits from it.

36

"See the many dots, connect them continually."

Testing is not deterministic,
of following a simple set pattern.

It is observing,
experimenting,
seeing the dots,
constantly connecting them
to do better and better.

37

"Be open to different points of view without bias."

Testing requires a very open mind,
to see various points of view,
engage in argument &
disagreement without bias,
so that we may get ideas
to 'poke' the system and find issues.

38

“Form opinions based on facts.”

As much as it is important to be open,
formulate opinions based on pure facts
so that you can anchor and explore better.

39

“Relentlessly pursue, but know when to timeout.”

Sometimes bugs vanish down the rabbit hole, sometimes systems behaves weirdly.

These are not to be ignored, they are opportunities to pursue relentlessly, but do know when to timeout, be mindful of business & timelines.

40

"Constantly assess what you don't know, not gloat about what you know."

It is the gaps in what we know
that help us to become better,
forcing us to learn and refine,
not just the knowledge we possess.

41

"Do with pride, stay humble about outcomes."

Pride in work comes from the confidence we possess, very necessary for great testing.

When test artefacts are reviewed, demonstrating confidence is key. To be able to stay that way, it is important to be humble so that we don't become over confident!

42

"Prioritise continuously."

Testing is risk reduction.

Given challenges of time, cost & quality,
staying focused & moving forward despite
issues and challenges demands
we constantly re-prioritise and
focus on what is most relevant as of now.

43

"Stay balanced."

Extremes of too much testing or less testing,
too much reliance on tools,
or on only on facts etc.
may not be very productive.

It is a fine art of staying in balance,
that is key to great outcomes.

44

"Try to connect cause to effect constantly."

Attempting to figure out
potential causes from observed effects
enables to refine strategy & explore well.

45

“Pay attention to special cases, do not be satisfied with common causes.”

It is the interesting one-off situations that help us to understand things deeply rather that common occurrences of issues.

46

“Time is not a constraint – Focus on how much can be done, not on how much is needed."

We all know that the clock does not stop, we can only freeze what we can deliver. Given a time target, it is about how much I can accomplish, that matters in today’s world.

47

"Listen silently, talk excitedly!"

When we listen to some one's views silently without bias, new ideas emerge.

Similarly when we talk excitedly sharing views with the other person silently listening to us, ideas refine and sharpen!

48

"Take notes copiously."

While observing, listening, experimenting and exploring, take notes liberally.

To help you remember, and more importantly, come up with newer ideas when you re-read it.

49

"Stimulate all senses – write, draw, colours, direction, voice."

When you note down observations,
put together a plan,
jot down scenarios etc., mix it up.

Write words/sentences,
in various directions (up, down, angle…),
use colours liberally, draw, voice record too,
keep the right brain vibrant
stimulating the unconscious
to see the unknown.

50

"Code, design, build, troubleshoot, write, read."

It is not just testing that matters,
it takes well rounded skills related to full
software life cycle to deliver clean code.

Design and code,
build systems,
troubleshoot & support,
write documentation &
read other people's code
to become a brilliant software professional!

*"**We are SmartQA evangelists.** For over two decades we have transformed how individuals, teams and organisations have practised testing. We espouse methodology to test intelligently. Our mission - Elevate to high performance via SmartQA."*
www.stagsoftware.com

The HyBIST Approach to SmartQA - MASTERCLASS
Testing is deep probing to seek clarity and in the process uncover, preempt issues rapidly. The HyBIST approach enables designing smart probes and probing the system smartly.
https://smartqa.academy/courses/smartqa-using-hybist

doSmartQA - AI based Smart Probing Assistant to interrogate, hypothesise issues, design & evaluate user story or a set of stories in a sprint rapidly. An assistant for smart session-based testing based on HyBIST.
Download personal edition from **here**

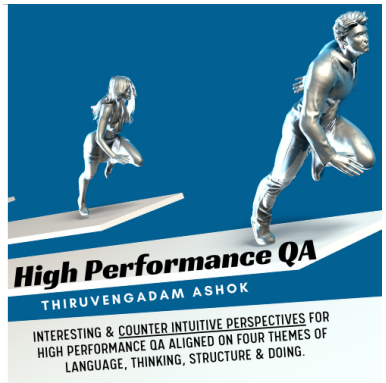
SmartQA Musings - A gentle flurry of interesting thoughts on smart assurance as a weekly webcast. A refreshing view of assurance to broaden & deepen thoughts/actions.
www.stagsoftware.com/subscribe

SmartQA Biweekly - Ignite your curiosity with fresh insights, thought-provoking ideas, and inspiring content delivered straight to your inbox every fortnight.
www.stagsoftware.com/subscribe

A rich collection of original content on smart assurance

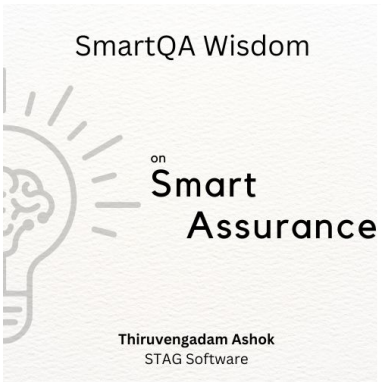
SmartQA eBooks - for Sr Engg/QA managers, Sr Test Practitioner & Young Test Practitioners too.

High Performance QA
Communicate Clearly
do SmartQA -The HyBIST Approach
HyBIST at a glance



Download from www.stagsoftware.com/smartqa-ebooks

SmartQA Wisdom - Profound nuggets of wisdom to think deeply, do rapidly & smartly for Test Practitioners.



on Smart Assurance
on Personal Growth
on Test Design
on Smart Understanding
on Mindset & Habits
on Problem Solving

Download from www.stagsoftware.com/smartqa-wisdom