TESTCo
Outsourced Software Testing
www.testco.com

Website Testing Cheat Sheet

Some Really Useful Things to Think About Before You Start Testing Your Website

IN THIS REPORT

Problem: Lack of Goal, Strategy and Plan for website testing projects can lead to missed deadlines, frustrated clients and wasted time and money.

Solution: A simple template for building an effective Goal, Strategy and Plan for website testing projects.

Benefits: Goals achieved, Risks reduced/avoided, Frustration and Worry eliminated.

Introduction

Testing your website can be daunting. You have a ton of pages and lots of really important forms to capture information from your prospects and guests. Your audience has one of over 250 different browser/operating system/devices. You need it to work right and look good!

How can get you get the website testing and confidence you need without spending days or weeks doing the testing?

At TESTCo, we've tested a lot of websites and we've built a "cheat sheet" to help you think through the problem and get your "biggest bang for the buck".

We break our thinking down into three different phases – Goals, Strategy and Plans.

Each phase has several valuable critical thinking questions to help you understand the phase and begin using this guide quickly.

Goals

Goals help us understand **why** we're doing the project and **what** outputs/results we need to produce to create the value our client needs.

SMART Goals are a great place to start. This isn't a Goal Setting Cheat Sheet though.

Nevertheless, setting and achieving project goals is a critical activity on TESTCo projects because it forms the strongest foundation for delivering real value.

Value, like beauty, is in the eye of the beholder – in this case, our clients. So, we want to make sure we know **exactly** what they want us to produce.

Critical Thinking Questions for Goals

- Who wants this outcome?
- What outcome do I want to achieve?
- Why is that important?
- When is it needed?

Strategy

Strategy has almost become a trite word so we've learned to get very specific with it for our clients.

We use "Strategy" to describe **how** a combination of Assets, Advantages, Actions, and Constraints that can be used to achieve the Goal. BTW, you can't really have a Strategy without a Goal.

For example, with a Goal of "Test and confirm on these four Browser/Device Combinations within 1 business day", we could pull together a couple of different strategies.

- Example 1 – Scripted – The QA Team collaborates to build a small set of User Scripts/Stories and then run each script/story on a variety of browsers/devices. This strategy gives the most website coverage but may result in uneven device coverage if constrained within a short timeframe. This approach is usually best on newer sites that have not been subjected to detailed website testing previously.
- Example 2 – Exploratory – The QA Team collects all necessary browser/device combinations and then run detailed exploratory testing strategies. This strategy ensures that all devices are covered but may result in uneven functional coverage if constrained within a short timeframe. This approach is usually best on mature sites where new devices may pose more risk than functional changes in your website software.
- Example 3 – Regression – The QA Team collects requirements and user stories to build test cases/test plans and then proceeds with orderly execution of the test plan. This strategy ensures that here is a measurable method of testing all aspects of your website and it provides a solid foundation for subsequent rounds of testing – usually used for

critical information websites. This strategy is expensive and time consuming but worth it if you will need to fix and regression test and if the cost of a defect is high.

Critical Thinking Questions for Strategy

- What things/people/technology are available to help achieve the goal?
- What advantages do each of them have that makes achieving the goal easier/faster/better?
- Can any of them be combined to make it even easier to achieve the goal?
- What limitations/constraints do each of them have? How much does each cost?
- What risks/conflicts/issues/problems need to be specifically avoided?
- Which combination of things/people/technology produce the fastest/bestest/cheapest with the lowest risk and cost?
- The best combination is your best strategy!

Plan

Plans tell us **who** and **when**. You've selected your best strategy and the plan arranges all your assets with advantages into actions on a timeline.

At TESTCo, we've learned that the real value comes from the "planning", and not the "plan". Plans can't control or predict – just suggest and report. Planning – the critical thinking needed to find, select and manage the assets and advantages needed to produce the goal – is actually where the magic happens!

You already know how to build a plan.

Here's how to Think about Planning:

Critical Thinking Questions for Planning

- What is the current Critical Path to the Goal?
- Are all of the Assets taking appropriate Action using their Advantages?
- Is every Activity within two degrees of separation from the Goal?
- What Constraints are missing from the plan?
- Are the Top-Three Risks addressed in this plan?

- Is the Goal achievable as of today?

Summary

We use this Goal/Strategy/Plan (GSP) approach on every project at TESTCo – even a short one- or two-day website testing project. It's a lot to cover and a short report doesn't really do the subject justice. I've included an appendix with some more Critical Thinking Questions that are more specific.

What Do I Do Next?

Thanks for reading this report. I hope you've found it useful and valuable.

We're passionate about software quality engineering and helping people fix, finish and launch better software and website projects. We'd love to talk with you about it and share what we've learned.

You can implement these ideas yourself! In fact, we recommend you give it a try.

If you have questions or get stuck, just call or email us and ask!

No Charge! Free! Gratis!

We enjoy sharing what we've learned with other people who share our passion for creating and delivering value.

On the other hand, if you find that you want these benefits but don't want to spend time figuring it out for yourself, we're happy to do the work for you.

If you need help today, then by all means, please call my cell phone at 512-970-7283. If I don't answer, please leave a voicemail letting me know that you need help urgently. I'll call you back within one business day. You should also send an email to jeff@testco.com so I'll have two opportunities to see that I need to get in touch with you quickly.

If you don't need help today but think you might need help in the near future, please send an email to jeff@testco.com and we'll arrange a time to talk on the phone and answer any questions you have.

Here's what one of our On-Demand Software Testing clients said recently about a website testing project –

"TESTCo was crucial to meeting our testing goals and deadlines. Even though I was too busy to provide proper direction to the testing team, they were able to get started on their own, create test plans based on what I wanted and deliver results from day one. As the testing needs of the project grew, TESTCo was easily able to grow with us. They continued to build requirements themselves and they were able to manage the testing process in a way that worked for us, not against us. Bringing in TESTCo is a huge win for any software engineering or website testing project." - Apar

I wish you the best of luck and success with your next project.

Jeff Hotz, President/Founder, TESTCo

About TESTCo
TESTCo provides On Demand Software Testing Services so that you can avoid nasty surprises when you release your software.

Appendix

Additional critical thinking questions for your website testing project.

- What do I want to accomplish in this project?
- What budget do I have?
- What devices do I have (and need) to test with?
- Where do I get the devices I need but don't have?
- Who has time/knowledge/expertise to do the testing?

- How much time will be needed for each test scenario? Will it vary by device or environment?
- Who is best suited to test each website area or device/environment?
- What needs to be tested? Do we need test cases or test scenarios?
- What features/functionality need testing?
- Is negative testing needed?
- How will we measure test and test case coverage?
- How will we report and track defects and test verification?
- How will we avoid reporting duplicate defects?
- How will we format and report defects so they are useful to the developers?
- How and when will defects be fixed and then re-tested?
- Who needs to be informed on a daily basis? Weekly? Monthly? What information do they need to be confident in the project?
- How often to verify progress or re-plan the project?
- How much budget tracking is needed and how often?
- How to make sure the testing team is most productive without down time?