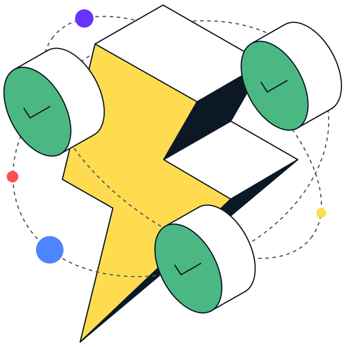
Original Software

The Business Case for UAT

originalsoftware.com

The importance of UAT

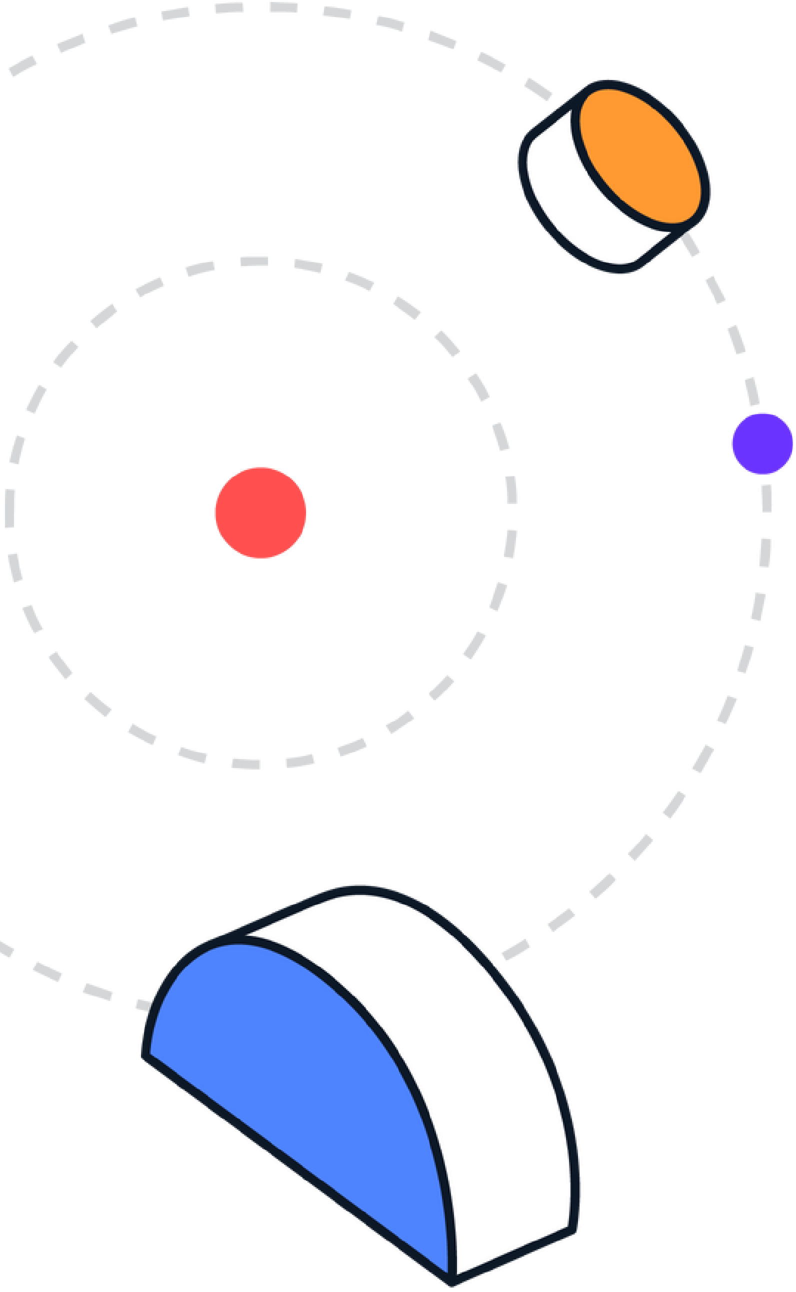


User Acceptance Testing is a key part of the software delivery life cycle and adoption process.

It serves multiple purposes, typically:

- A final quality check that there are no glaring errors.
- A quality check against less common scenarios that perhaps only the users know about.
- A usability check, assuring that the job can still get done, hopefully, more efficiently than before.
- Familiarization, enabling changes to documentation operating procedures to be implemented.
- Training. Understanding process changes and informing the rest of the user community need about the changes.
- Buy-in. Helping establish user commitment to the change.
- Acceptance. Sign off that we are good to go.

Original Software has created innovative software solutions to make the process easier, less time consuming, more effective, more collaborative and more enjoyable for both users and the others engaged in UAT.



UAT challenges and solutions

Time & Cost

- Manual testing is time-consuming. Users are taken from their day jobs, so who does their jobs while they are testing? Does the work get delayed? Do you employ temporary staff or pay overtime? Reducing the amount of time, UAT takes by making it easier for users and hence more efficient will reduce costs.

- It takes a lot of time to properly document and describe how to recreate an issue in testing, but this can be done automatically with virtually zero effort. This creates a significant time saving, not just for the tester or user but for anyone needing to look at the issue report.

- If the information needs to be passed to a third party, there can begin a game of Q&A tennis. Very often, it seems there is always some other piece of information and evidence required. When it happens, this process takes a lot of time and involves a lot of frustration. Having all the evidence, in detail, from the start is like serving an ace; the vendor immediately has all they need. Resolution is faster and less painful.

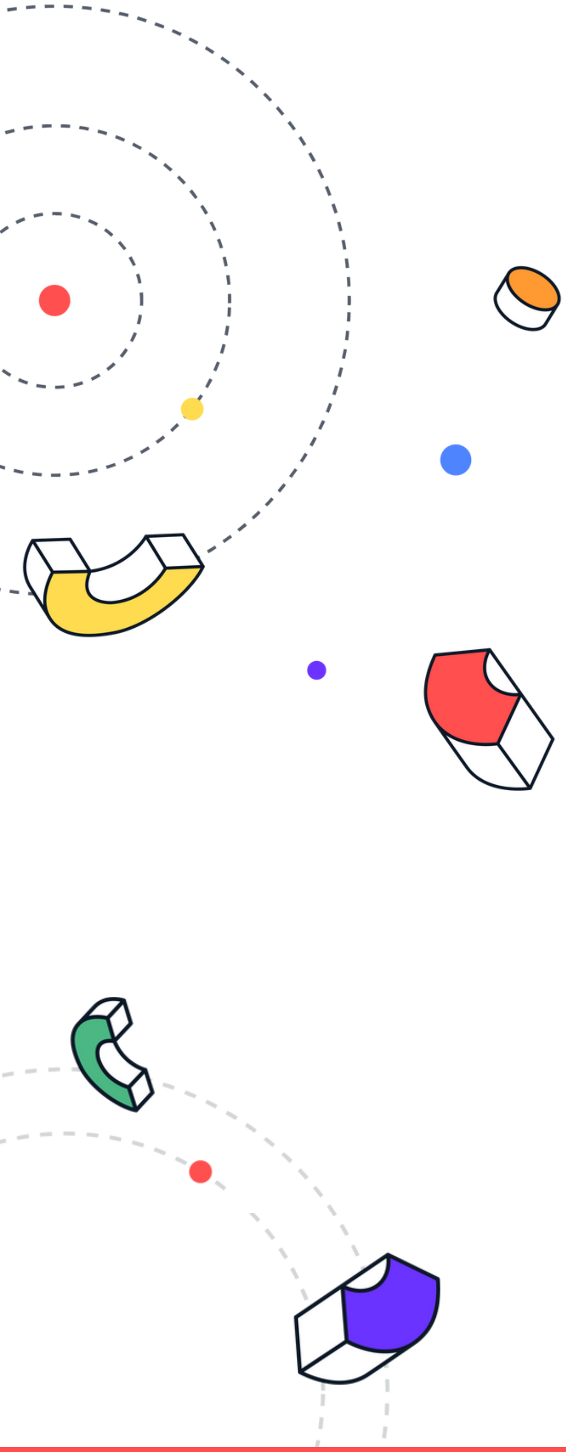
UAT challenges and solutions

Quality

- Users are not testers; they miss things, they may get bored, and issues may slip through. Therefore, it is good to reduce the load on them by using automation of normal UAT cases in advance of manual UAT and to help them in execution with tools to make it more interesting and less time-consuming.

- They get more bored and less engaged the second and the third time, so testing becomes weaker the closer you get to deployment. This increases risk. By optimizing the testing before and during UAT, more errors will be found earlier and will mean fewer phases of UAT, keeping the quality of testing higher.

- The issues that escape to UAT and, worse to production have to be found, documented, analysed, recreated, understood, fixed, tested, and implemented. Meanwhile, the business has to cope, perhaps with a workaround or avoid this area. The people involved are not doing the useful things they could be doing. Issues slow down the business and IT's delivery of new features. Issues are expensive, and investing in avoiding them saves money.



UAT challenges and solutions

Quality

- Issues cause disruption to the business and to customers. Especially if they affect customer facing services. This impacts reputation and revenue.

- Quality governance. Users are not usually trained testers, and they may not understand the best way to perform thorough testing. They may not even understand the importance, and hence being able to oversee in detail what testing was actually done can enable UAT management to redirect effort to be more effective and more complete.

- Quality in its broadest sense, has many facets beyond "Does it work as intended". Ideally, any change delivered should make a process easier, better, faster or some achieve some other positive parameter. This includes aspects affecting the User Experience (UX) and hence being able to review response times, the path through the application, the mouse movements, mouse dwell times and keyboard sequences helps assess the usability design.

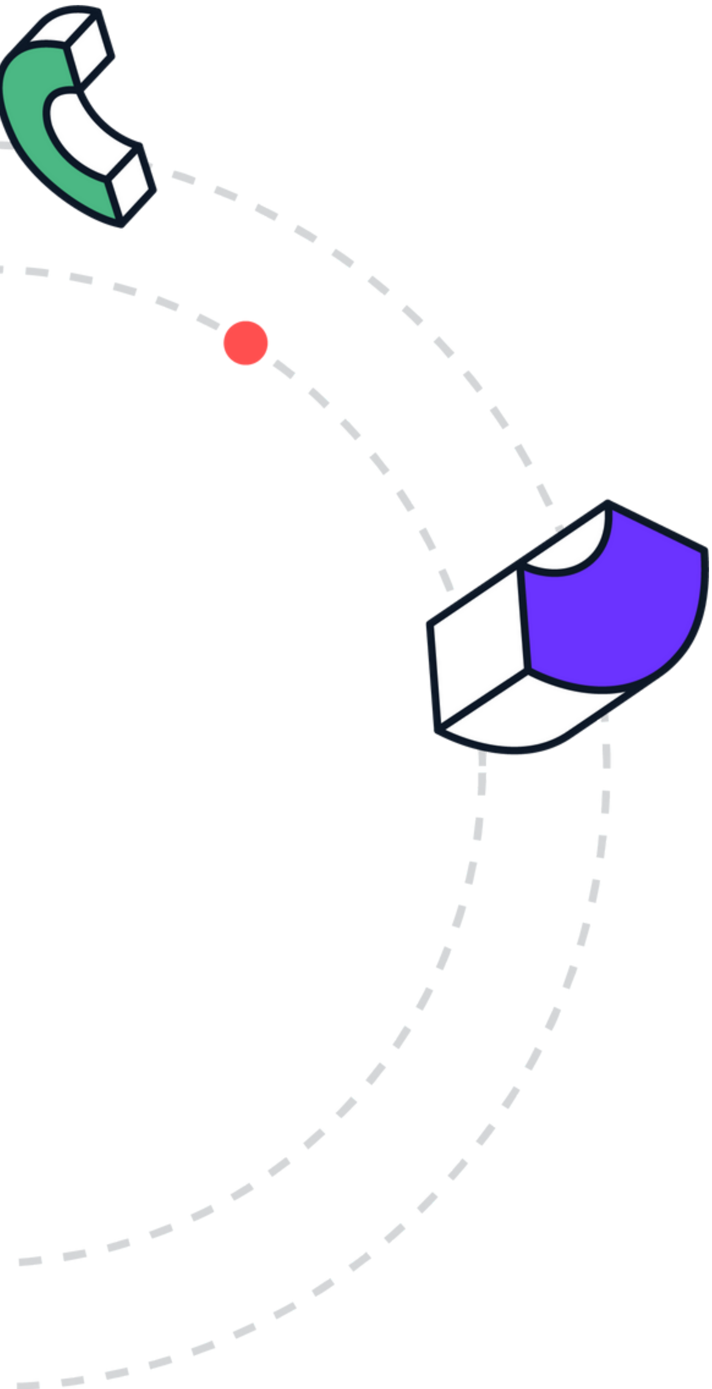
UAT challenges and solutions

Velocity

- By being more efficient in testing, taking less time, but testing in more depth, software delivery is sped up. IT becomes more productive. Collaboration is enhanced.

- More projects are delivered earlier, and the benefits and ROI they offer are achieved sooner.

Collaboration from QA and delivery teams with people doing UAT mean better engagement and attention to the task. Testing can be properly planned and allocated based on resource availability.



UAT challenges and solutions

Going Live and Success

Having the application working well is clearly a significant advantage at the time of rollout. But there are many other factors which lead to a successful project.

Users need to know how to use the changed application and will need materials and perhaps workshops to help them gain this understanding.

The output from TestAssist includes Word documents, PDF docs, Videos and 'Animations' all of which can show how the application is used to achieve the business objective, automatically captured during a test or business process capture session. It is much less effort to create than a manual documentation activity, plus the animations, in particular, are easily enhanced and annotated to help users to follow the steps.

During the project, communication and collaboration are essential. Giving business users clear advance notice of resource needs, test content, progress and timescales helps ensure the project flows smoothly.

UAT challenges and solutions

Make good testers great. Using TestAssist helps make good testers great. It provides instant feedback for agile mock-ups prototyping as well as enabling the production of training guides and process documentation for free.

Incorporating Qualify allows every member of the quality process, from management through development and QA to end users, to share in the quality process with real-time decision-making, driving efficiency.

This all makes for a compelling case to continue manual testing, albeit with the right tools, alongside tests that can be automated, saving money and allowing for the reallocation of those resources to the areas where manual work is required.

Let your team focus on delivery rather than infrastructure and tools.

This guide has been brought to you by Original Software. With the happiest customers in software testing, we've been helping businesses meet their objectives and deliver quality software through our range of innovative code-free testing solutions for over 25 years.

originalsoftware.com

Original Software