GreatDemo!

Peter E. Cohan
The Second Derivative
1532 Scenic Drive
Trinidad, CA 95570
T: +1 650 631 3694
E: PCohan@SecondDerivative.com
W: https://GreatDemo.com

A Perfect Demo Environment…

Have you ever heard (or said):

> "Sorry, I don't have an example like that in our demo system…"
>
> "No, we can't show that workflow – this is a demo database…"
>
> "Our demo environment isn't up-to-date – that's why this feature isn't working…"
>
> "No, I don't have good data to show in this report – no data, in fact – sorry…!"
>
> "That report hasn't been built for our demo system – but trust me, it's terrific!"

In your demos, how often do you find you are apologizing for an inability to demonstrate capabilities, complete workflows, or present compelling results and reports? Far too often, in many cases!

A perfect demo environment enables you to show what you need to show, clearly and convincingly. No apologies, no excuses. Accordingly, here are some recommendations for demo environments to make your demos as crisp, compelling and successful as possible.

Don't Call It "Demo"

One of our objectives in a software demo is to "suspend disbelief". Anything that looks real helps our cause; anything that appears fake is going to hurt us.

In spite of the fact that we teach "demo" skills in our Great Demo! Workshops, and in spite of the fact that the customer has *asked* for a "demo", I recommend avoiding the use of the word "demo" in, well, demos… And, in particular, in demo environments.

Why? Because customers unconsciously translate the word "demo" to mean "fake". If it is a *demo* environment, then it is not real, from customers' perspective. Anything done in a such an environment is suspect, accordingly!

So, don't call it your "demo" system or "demo" environment or similar. Instead, give it the name of a fictional but plausible customer or other realistic-sounding option. ("Apex" is an example we used long ago…)

And Don't Label Things "Demo"

Similarly, anything labeled "demo" in your (ahem) environment causes the same result.

File names, module labels, login names, forms, dashboards – I've seen all of these emblazoned with "demo" in one form or another. All of these subtly scream "fake!" to our customers.

Again, use a neutral name or naming convention that looks like a real customer. Much better!

No Obviously Fictional Names

Similarly, when a customer sees the names of famous characters in your demo environment, this also says, "This is fake!" to customers.

I recently saw a vendor that used the actors' names from the full Oceans XX series of movies – including George Clooney, Brad Pitt and Julia Roberts (the original cast) plus Sandra Bullock, Cate Blanchett and Anne Hathaway – all of these names were in the vendor demo database. I've also seen sports figures, important scientists, authors, and more.

Perhaps it's easy to remember the names, but they communicate "This is fake!" to customers.

Instead, use realistic names, addresses and other data. One simple solution is to use the real names from your organization – that's an improvement.

A great source for convincing, dummy data is to use the databases and data sources created and used by your own organization's QC team. After all, your QC folks need realistic data sufficient to explore a broad range of testing scenarios (hmmm – potentially very similar scenarios to what you need to demonstrate!). A few example sources are DTM Data Generator, SQL Data Generator and Mockaroo.

Include Problems, Exceptions, and Opportunities

Status quo is boring.

I've seen countless demos that present a dashboard or report that shows everything "looking good" – everything is "in the green", without any problems or concerns. This is pointless. You need to show that your tools can *find* the customer's problems and that your software can directly address those problems or enable solutions.

The most compelling demos show examples of how customers can solve their business problems, address outliers and exceptions, and surface and exploit opportunities. This means that your demo environment needs to include good instances of these with good example data.

Problems should be uncovered and presented clearly – and provide opportunities for alerts to be generated and sent from your system (just like a customer would use it).

Exceptions should be similarly discoverable – with the appropriate data to explore and determine how to address (just as a customer would investigate).

Opportunities should be readily surfaced and exploitable (just as a customer would pursue).

The more realistic, the better…!

U.S. vs. International

When I was banished overseas (to Switzerland – tough duty), I realized that our demo environment was entirely U.S.-focused. This can be a big "fail" for non-U.S. customers.

U.S.-based companies often use demo environments that show U.S. maps, list U.S. cities and addresses, U.S. telephone numbers, U.S. currency, U.S. ZIP codes (even the use of the term "ZIP Code" is an example!), U.S. product names and more. This can be semi-insulting to non-U.S. audiences – and it won't help your cause.

If you want to sell in Europe, make sure you have Euros, European addresses and postal codes, telephone country codes, and all of the other units and measures that are used in your target countries and markets. (Furlongs per fortnight is one of my favorite arcane units…)

Want to sell into the UK or other English-speaking (but non-U.S.) counties? You may also need to use their local version of English spellings and vocabulary. Now *that* box is ticked…

Launching into Asia-Pacific? Same guidelines.

You don't support local languages in these regions? You may want to rethink your strategy…!

Very simply, we need to have region-specific data (as appropriate) if we want to build a vision for customers of using our tools in the region(s) in which *they* operate. [And, to be fair, if you are a non-U.S. software company seeking to enter the U.S. market, the same guidelines apply to you as well…!]

Market Alignment

Have you ever seen a demo presented to a large construction firm that used example data from the banking industry? (I did – it failed…)

Best case: your data, vocabulary, use-cases and demo examples match the customer's specific market or vertical.

Next best: your data etc. are *close enough* to the customer's specific market that the customer can make the connection. Note: it is the *customer* who decides if your data is "close enough…"

Generally insufficient: your data etc. doesn't bear any resemblance to your customer's market – you are asking them to make too large a leap. (Disbelief is no longer suspended…)

Really bad: data that is obviously fake *and* has nothing to do with the target market…!

One successful approach I've seen is to have a fairly neutral set of data, but *also* have the ability to configure or customize the screen labels, dashboards, forms and field names to map to specific markets. This may require some level of configuration or programming to achieve, but the payoff can be high.

An additional note to consider: Customer Success guidelines suggest that the further your prospective customers are from your current, successful target markets, the worse the fit – and likely represent the highest risk of failed or unhappy implementations.

Market Maturity

Interestingly, customers at different stages of the Technology Adoption Curve will likely react very differently to demo data. "Technology Adopters" and "Early Adopters" are often very forgiving of the data that is used. "Early Majority" customers may be reasonably forgiving, but the further you move to the right, towards and into the "Late Majority" the more they need to see their own data (or what appears to be their own data).

Who Am I?

Logging-in as "Admin" is another approach that yell's "Fake!" to your audience…

The only people who login as "Admin" in real life are, well, the system administrators…! This is even more of an issue when vendors start talking about how "the system can be configured by role for end-users, managers, executives" etc.

Along similar lines, try to avoid obviously fake names, such as:

- Stanley Staffer
- Mary Manager
- Ernie Executive
- Adeline Admin

(Note: I've seen all four in demos…!) Do these look real? Nope – suspend disbelief!

Be a Customer

Seriously, in a sense. Some of the best demo environments are “customer” instances, created and maintained *exactly* the same way as real customers’ environments.

For SaaS organizations, this can be hugely advantageous – your demo “customer” is treated the same as all the other customers, getting updates, improvements, etc. exactly on the same timeline.

You’ll know your demo environment will be up-to-date and consistent with what all the other customers are using. No surprises…! (Or fewer surprises, at least…)

Keep It Current

I recently saw a workflow “alert” that showed a missed task that was *5 years old*! Holy old cow! Some manager is going to be *very* angry, if this was real life.

Workflow dates need to appear to be current (or reasonably so). The older the information, the less believable it is.

In many systems, this presents some challenges and suggests that there needs to be a way to “refresh” demo environments.

Refresh or Reset

Perhaps one of the hardest tasks for those who maintain demo environments is to keep them current and up-to-date (to avoid embarrassing problems like the above). There are several strategies that can be employed:

- Review and refresh the data: this relies on somebody periodically working through the relevant data/workflows/alerts/etc. to cleanse outdated bits and update as needed. It is work – but it needs to be done to keep the demo environment as realistic and believable as possible.

- Reset: wouldn’t it be great to simply hit a button and have everything returned to a pristine “zero” state? There are a number of vendors who, in theory, provide these capabilities – Cloudshare, Skytap, Quali, Qloudable, VMware Workstation, Microsoft Dynamics’ Test Drive, and other tools may provide the capabilities to enable this. Check them out if you are unfamiliar…!

- Alternatively, you may want to create scripts in your own environment to accomplish the same goal.

Availability

In some organizations, demo environments need to be reserved ahead of time (for a range of reasons). In some cases, these reservations may need to be done a day or more in advance. This can make things challenging for any demos that need to be done in a shorter timeframe or on an *ad hoc* basis. I've heard these environments referred to as having been implemented by the "Sales Prevention Team…!"

In other cases, it may take time to "spin-up" a demo environment – this may be unavoidable, but similarly troublesome – and you'll need to plan ahead accordingly.

And, of course, have you ever tried to access your demo environment only to find it "down" or simply unavailable? Not good, particularly if you flew across the country (or ocean) for the most important demo of the year…!

In a demo that I saw recently, the software announced, "Not Enabled for This Demo Account" when the presenter tried to show a specific use-case. Even worse, the use-case was introduced by the *vendor* as "here's something really cool…", not in response to a customer question…!

Fundamentally, your demo environment needs to consistently available – you need to be able to count on it.

Performance

"…The demo system is slow today because…" Have you ever heard this in a demo meeting? Have you ever said it yourself?

Customers will only remember one word: "slow."

Customers assume that whatever environment you are using for your demo is going to be better than theirs – often *much* better. I've never heard a customer say, "Our network is blisteringly fast…!"

In some cases, you may be able to pre-fetch, cache or otherwise out-flank areas of potential performance problems. You'll want to fully characterize and be closely familiar with your demo environment to know how to prepare.

Best case? Your demo environment is truly "blisteringly fast" and has no visible performance issues.

Surprise – No Internet!

Have you ever found yourself in a situation where you needed access to the web to run your software – and found zero connectivity?

Or have you ever been asked to do a demo at a government organization that didn't allow access through their network?

[This reminds me of the Monty Python Cheese Shop sketch – "Sorry, fresh out…"]

You may need to have backup plans in place for these situations, that might include:

1. The ability to run your full software from your laptop, without any internet connectivity.
2. Or, the ability to use your own "myfi" or "mifi" environment, if allowed.
3. Or, (least attractive, but still a bit of a self-rescue) you've captured a healthy set of the key screens from your software into PowerPoint – that you can share without any network connection to give the customer a sense of what your system can do…

"Trust Me…"

Far too often, I see demos that show reports with sketchy data – or no data at all…! This is even worse when the reports are the final screens for important use cases… (Fail.)

Workflows need to work; reports need to be reasonably (or fully) populated; graphs need to display properly; alerts need to be driven by realistic actions; and so on. Your software needs to appear to operate as it would for the customer, as much as possible.

It is unacceptable to hear the vendor say, "Oh trust me, these reports are terrific…!"

Projector Woes

Have you ever connected to projector (or "beamer") in a meeting room and been surprised at what happened to your screen? Many conference rooms have old, lower resolution projectors that are simply insufficient for demos from high resolution laptops, resulting in invisible buttons, lost portions of your screen and unexpected scroll bars.

At minimum, characterize the projector you must use so that you can reduce the surprises and be prepared – practice key portions of your demo *in-situ*, before the meeting starts.

Even better, considering purchasing and carrying with you a projector that enables your software to be shown in the best possible light (no charge for this pun). An investment of a few hundred dollars or euros could save thousands in otherwise wasted time and travel expenses…!

Mini Mouse

Finally, consider your mouse cursor – after all, your audience will likely be trying to watch its movements for an hour or longer. How visible is it?

You may want to consider increasing its size and/or fill. Most of the default mouse settings on Windows and Macintosh laptops are *too small* for demos. Explore the options and see what looks best with your software and typical demo scenarios.

Infrastructure Checklist

This a long article – a lot of information. We've explored some aspects of your hardware and software demo environments, but every individual's situation is unique.

In Great Demo! Workshops we suggest making (and using) an Infrastructure Checklist to reduce the risk of bad things happening to you in your demos: "The same bad thing should never happen to you more than once, if it is under your control…!"

A Perfect Demo Environment

A truly perfect demo environment probably doesn't (yet) exist, but by pursuing the guidelines above, you will certainly improve your probability of success with your demos. A few changes may yield the difference between obviously "fake" and successfully suspending disbelief.

For more articles on demonstration effectiveness skills and methods, visit our website at https://greatdemo.com/. For demo tips, best practices, tools and techniques, join the Great Demo! LinkedIn Group or explore our blog at https://greatdemo.com/blog/.