



INDUCTIVE AUTOMATION DESIGN SERIES

Design Like a Pro

Boost Your Skills in HMI / SCADA Project Development

Part 3: Designing HMI / SCADA Projects That Deliver Results

The end of a project can be the most rewarding and the most nerve-racking phase of developing an HMI / SCADA project.

All the work on your HMI / SCADA project has led to this – the start-up phase – the point at which you flip the switch and watch your project in action. A successful project can increase efficiency, productivity and profit; an unsuccessful project can cost time and money. With so much on the line you have to be sure that your project works, and the start-up phase is when you ensure that it's ready.

Every step of the start-up phase is about making sure the project works well for the operators who will use it. This paper is intended to give you tips to ensure you successfully complete the start-up phase so you can deliver an HMI / SCADA project that delivers results.

Design Like a Pro Series

At Inductive Automation we hear from professionals all the time about how they are using Ignition to create great HMI / SCADA projects. In addition to offering software solutions for the manufacturing industry, we are constantly striving to support the Ignition community with the training, consulting, and knowledge they need for success. As part of that ongoing effort we are offering this "Design Like a Pro" series of white papers and webinars to give you information that will help you design a successful HMI / SCADA project every time.

In this installment of the series, the start-up phase will be explored. After you've planned out and designed your project, the last step is to successfully launch it. This paper is focused on putting the finishing touches on your project to make sure it's a success from day one.



THE START-UP PHASE

When running a marathon the last six miles can be the most rewarding or disastrous leg of the race. For the experienced runner, the last six miles is what the marathon is all about. It's an opportunity to push himself as he moves toward a strong and exhilarating finish. For the inexperienced runner, the last six miles can be a torturous stretch of the race where every mile seems to get longer and the finish line farther away.

Finishing an HMI / SCADA project is much like finishing a marathon. If you have set a good pace for yourself by efficiently completing the planning and design phases, you will confidently move forward to a successful start-up of your project. However, just because the finish line is in sight, that doesn't mean the work is over. The start-up phase consists of the final crucial steps you must take to finish your project. By thoroughly testing, carefully securing and strategically launching your project you can ensure that you triumphantly cross the finish line.

Start-Up Phase Steps:

Step 1: Testing

Run your project through the paces to make sure it works.

Step 2: Security

Secure the access points and data integrity of your project.

Step 3: Launch

Go live with the project then follow up with operators to troubleshoot issues.

The most important benchmark of your project's success will be the approval of those who use it. A good project is one that informs its users and makes them more productive. Keep this in mind as you finish your project; the goal is to satisfy the needs of the project's users. If you use this goal as your finish line you will know when you've crossed it.

For a checklist to help you stay focused on the finish line as you progress through the start-up phase, see Appendix A of this document.

The Planning and Design Phases

The planning and design phases are critical to the creation of a successful HMI / SCADA project. These phases are covered in the previous installments of this series. For more information visit our website at www.inductiveautomation.com/resources to download the white papers and view the accompanying webinars for Design Like a Pro: Part 1 and Part 2.

STEP 1: TESTING

Testing your project is vital to its overall success. No matter how carefully a project has been put together, there is always a need to have a dedicated time at the end of the project for thorough testing.

Every project is different, which means that every project may require its own unique set of tests to ensure it operates correctly. Regardless of what kind of project it is, there are at least two specific tests you do with every HMI / SCADA project: an I/O check, and a click test.

I/O Check

The biggest thing you have to do when testing your project is a full I/O check. An I/O check consists of double checking all the I/O (input / output) points used in your project to ensure they match what is happening on the plant floor. There really is no way to do this without touring the facility with your project to make sure it accurately reflects reality. The best way to do this is to take your project around the facility and test it as you go.

For example, if you have a tank in your project you need to go to the actual tank when you test it. That way when you turn the tank on or off in your project you can verify that the tank is actually doing the action you want on the plant floor. Make your way around the facility and test the functions of your

project; if you find something that doesn't work, you will definitely want to fix it. The last thing you want your project to do is execute an unintended action.

Click Test

The click check is just what it sounds like. You need to go through your entire project and click every button, link and object in the project. You basically want to double check your project for every single action an operator could potentially do. Check every navigation link of every page, test every popup window, and try every button.

The basic rule is that you don't want to have anything in the project that you haven't tested yourself. This is especially true if you used time-saving features while developing your project, like templates or UDTs (user-defined types). Even if you made components automatically when developing the project, you should still test them manually.

It may take some time and effort, to test every part of your project but that is preferable to running into major problems after you launch it. A major problem after the project goes live could halt production, which could be very costly. Finish your testing and troubleshoot whatever problems arise; afterwards you will have peace of mind knowing that your project has been thoroughly tested.



STEP 2: SECURITY

Making sure that your project is properly secured is an important step before your project goes live. Leaving your project open runs the risk of putting potentially sensitive information and control functions into the hands of someone you don't want to have it. So how do you know what to secure? To determine the best security strategy you have to start asking some questions about your project. Here are four security questions that every project developer should ask about the security of his HMI / SCADA project.

Question #1: Where does the project need to be accessed?

Knowing where your project is being accessed is very important to its security. Determining where people will be using your project will inform you as to who could potentially see sensitive project information.

One major consideration is if anyone will need to use the project via the Internet. You must assume that any data transferred over the public Internet could be accessed by others and is therefore a security risk. To protect your project you need to put certain measures in place to safeguard your information.



With the rise in popularity of web-based SCADA programs, using standard web security protocols is more and more of a necessity. One security technology you should use is SSL (secure sockets layer). SSL is a secure communications protocol for the web. It's become quite common on the web and is used by most banking websites to protect their clients' information. SSL encrypts data communications that are transferred through the Internet, so even if the information is captured it will not be readable.

Another thing you need to consider is what kind of foot traffic will be around those who use your project. You need to ask yourself, if the user leaves his computer open will someone be able to access it and cause your company harm? If this is a concern, you may want to consider having the project screen shut down if a user has been inactive. By doing this you will lessen the chance of an unknown person accessing your project.

Question #2: Who needs access to the project?

Keeping access restricted is critical to your project's security. You need to know who is using your project to make sure that only the people who need access, get access. This is best handled through the use of a directory of unique user logins for your project. There are a couple of ways you can manage the user login directory for your HMI / SCADA project. You could create a new database to store the logins and update it whenever users need to be added or deleted. This method gives you total control over how you setup logins, but it can take some time and attention to manage it properly.

A better method is to piggyback on whatever login directory your business is already using. The most common directory service is Microsoft Active Directory™; many companies already use this service to manage the user logins for employees. The advantage of using Active Directory™ for your project is that you can keep all user login management in one place. So if an employee leaves the company and his user account is deleted from Active Directory™, his access to the HMI / SCADA project will be deleted as well. Keeping the management of your directory simple will decrease the potential for mistakes, which will increase the security of your project.

STEP 2: SECURITY

Question #3: What are users going to do with the project?

This question goes hand in hand with who will be using your project. For many projects – especially big ones – there are multiple functions that can be performed. You may not want everybody who can access the project to be able to perform the full range of project functions. A solution to this is limiting what users can do depending on their job role and where they access the project.

A line operator would most likely use the HMI / SCADA project much differently than a plant manager. You may not want line operators to access financial information in the project, and in the reverse, you may not want managers to affect control functions of the plant floor. Whatever the case, limiting the functionality of the project to certain users is a good idea because it reduces the chance of someone inadvertently doing something they shouldn't do.

The best way to do this is to define user roles in your project, and then assign users to the appropriate roles. The simplest breakdown of roles divides into two major roles: the read / write role and the read-only role.

The read / write role allows a user to do just about anything in the project because it allows him to write to tags which can control the individual functions of the project. The read-only role allows a user to read information from the project, but limits him from affecting anything. You can also combine the use of these roles to create specialized user roles that can write to some parts of the project but not to others. You can create as many roles as you would like, but keep in mind that in general it's better to keep it simple, that way it will be easier to manage.

One more thing to think about is what kind of access to give authorized users who are accessing the project remotely. It might make sense to give your plant manager full access to your control system both when he is logging into the project at your facility and from a remote location. However, you may not want to give a line operator the ability to control the line unless he is on-site.

It's possible to limit what certain user roles can and can't do depending upon what zone they are logging in from. For example you could setup one zone for your facility, and another zone for anything outside of the facility. By defining the read and write privileges for each zone, you can control what users are able to do and not do depending on which zone they're in.



Question #4: What have users been doing?

Even in the most secure systems, problems will still arise from time to time. You are bound to regret it if you don't give yourself good tools for resolving security problems.

Auditing is a powerful security tool that every project should have. Auditing is basically a tracking system that monitors when (or which) users log in and out of the project, what they do, and when they do it. This information can be invaluable to resolving security issues when they come up. If a security breach occurs on a system that has auditing in place, then the source and time of the security problem can be identified. This makes it a lot easier to troubleshoot the problem and put safeguards in place to ensure that it doesn't happen again.

Some SCADA systems have auditing built in, which means that getting auditing on your project is simply a matter of turning it on. However, many SCADA systems do not have auditing; in these cases you will need to set it up manually using databases.

STEP 3: LAUNCH

After successfully testing and securing your project it will be ready to go live. At this point it may be tempting to just launch it and walk away, but the professional HMI / SCADA project developer knows it is more prudent to move forward with caution. Launching a project at full scale too soon can be a costly mistake if you run into a significant problem. It is wise to plan a gradual roll out of your project in an effort to find and sort out any remaining issues. It is also essential to follow up with those who use the project to ensure that it functions well for them.

Rollout

A rollout plan is key to a project's successful launch. If the project is intended to run on multiple lines, at multiple facilities, then a possible rollout plan could be to start out on one line for a trial period. After that goes well the project can be extended to two lines, then four and eventually to another facility. At each point, run the project long enough that you feel confident it's working properly. If you troubleshoot problems as they arise, then each time the project's use is expanded, you can feel more and more confident in its stability.

Setup a solid timeline for the project's rollout, and let everyone know when they can expect to start using it. Keep in mind that there may be a learning curve, especially if your project is significantly different than what was being used before. It will take time for people to get accustomed to the project and how to use it. Give it time, be patient and don't get frustrated if the results you are looking for are not immediate. Every step of developing a new HMI / SCADA project is a process, not a destination, and rolling out your project is no exception.

When you start rolling your project out, make sure you have a complete backup of your project, and a solid backup plan. Backing up your project regularly ensures that you don't lose any of your work. Remember that you should never save the backup in the same place as the original. Using an external hard drive or cloud storage are both good options. Make sure the location you choose is secure, reliable and regularly maintained.



Follow-Up

The true value of your HMI / SCADA project will be determined by the people who use it daily. Your project can be seen as an essential tool for getting work done efficiently, or it can be seen as a hindrance; something that has to be worked around, instead of worked with. No one sets out to make a bad HMI / SCADA project, but they often end up that way because the end-user was not in the process of creating it.

If the project is designed to be a functional tool, then it is essential to get feedback from those who use it. After the project has been live for a short period of time it's critical to follow up on its effectiveness by talking to the end-users. Ask them what they like, what problems they have encountered, and how it can be improved so it is more useful to them.

Use their feedback to determine what follow-up project work you should do. The best projects are the ones that grow in proportion to and direction of their end-users' needs. Don't think of your project as a stagnant thing that is set in stone, but rather as a living and growing solution to ever-changing challenges. One of the biggest advantages of developing your own HMI / SCADA project is that it can be changed; this makes it infinitely more useful and adaptable to just about any situation. Don't look at doing follow-up work on your project as a failure; it's actually a step in the right direction because you are getting that much closer to achieving the results the project was designed for in the first place.

THE FINISH LINE



Each step of the start-up phase is about ensuring that your project can deliver the results expected of it. By taking time to thoroughly test your project you ensure that it works properly from the beginning. By answering the right questions about the security of your project you can deliver an HMI / SCADA project that keeps the company's information secure. Lastly, by gradually rolling out the project, and following up with end-users, you can make sure the project is an effective tool in the hands of operators.

Following the tips outlined in this paper, and in this series as a whole, will help you design an HMI / SCADA project that delivers results. The development process is just that, a process; one that can be improved and refined over time. The ultimate goal for any serious HMI / SCADA project developer should not be one great project, but rather to

create many great projects. The only way to do that is to improve your process and to look for the latest resources, training and software to aid you in creating projects that give your company a competitive edge. Keep working, keep developing great projects, and keep coming back to us for more tips on how to design HMI / SCADA projects like a pro.

Inductive Automation is committed to delivering meaningful knowledge, training and products that support the manufacturing and process control industry in HMI / SCADA development. For more design tips, check out the entire Design Like a Pro Series for a collection of white papers and webinars focused on development topics: www.inductiveautomation.com/resources

Start-up Phase Checklist

Appendix A

PROJECT DEVELOPMENT PHASE: START-UP

Making the last push at the end of a long HMI / SCADA project can be a challenge; it requires pacing and a focus on making it to the finish line. To help you finish strong, here is a checklist of the basic steps to take as you move through the start-up phase.

Every project is different, so one path won't work for every project, but this list is a good reference point and should help you stay on solid footing.

1. Testing

- | | | |
|--|---------|--------------|
| <input type="checkbox"/> a. I/O Check | due by: | assigned to: |
| <input type="checkbox"/> b. Click Test | due by: | assigned to: |

2. Security

- | | | |
|---|---------|--------------|
| <input type="checkbox"/> a. Answer Security Questions | due by: | assigned to: |
| <input type="checkbox"/> b. SSL in Place | due by: | assigned to: |
| <input type="checkbox"/> c. Directory Setup | due by: | assigned to: |
| <input type="checkbox"/> d. User Roles Built | due by: | assigned to: |
| <input type="checkbox"/> e. Auditing Setup | due by: | assigned to: |

3. Launch

- | | | |
|--|---------|--------------|
| <input type="checkbox"/> a. Rollout | due by: | assigned to: |
| <input type="checkbox"/> Rollout Plan in Place | due by: | assigned to: |
| <input type="checkbox"/> Project Backup Up-to-Date | due by: | assigned to: |
| <input type="checkbox"/> b. Follow-Up | due by: | assigned to: |
| <input type="checkbox"/> Feedback Collected | due by: | assigned to: |
| <input type="checkbox"/> Follow-Up Work Scheduled | due by: | assigned to: |