

THE M365 INTRANET “PUNCH LIST”



Sean P. McDonough
Senior Solution Architect and Consultant
Akumina, Inc.

Nashville Microsoft Community Days



Special Thank You to Our Statewide Sponsors



Special Thank You to Our Event Sponsors




A “PUNCH LIST?”



WIKIPEDIA
The Free Encyclopedia

Q ∞ Donate Create account Log in ...

 Punch list  1 language ▾

Article Talk Read Edit View history Tools ▾

From Wikipedia, the free encyclopedia

A **punch list** is a document prepared during key milestones or near the end of a construction project listing works that do not conform to contract drawings and specifications that the general contractor must correct prior to final payment.^[1] The work may include incomplete or incorrect installations or incidental damage to existing finishes, material, and structures. The list is usually made by the owner, architect or designer, or general contractor while they tour and visually inspect the project.^[2]

In the United States construction industry, contract agreements are usually written to allow the owner to withhold (retain) the final payment to the general contractor as "retainage".^[3] The contractor is bound by the contract to complete a list of contract items, called a punch list, in order to receive final payment from the owner. The designer (typically a licensed professional [architect](#) or [engineer](#)) is usually also incorporated into the contract as the owner's design representative and [agent](#), to verify that completed contract work has complied with the design.^[citation needed]

In most contracts, the general conditions of the contract for construction require the

A COMMON INTRANET PROJECT

How many of you have lived through this cycle in some form?

- Get a mandate (and some budget) to build an intranet.
- Assemble your intranet implementation team.
- Put together a project plan.
- Analyze the existing intranet (if one exists).
 - Make a determination of what you'll keep, discard, or relocate.
 - Document new features/capabilities that must go into the release.
- Put together the wireframes (and *maybe* some functional specs).
- Build and/or implement the technological pieces.
 - Encounter the inevitable delays due to technology and other factors.
 - Just prior to release, let users play with what's being built.
- Turn it on, declare it's live, and then begin the "real fun" ...



INITIAL PLANNING CONSIDERATIONS



**BULLETS TO REVIEW IN
DETAIL LATER (HOMEWORK!)**

- Intended intranet purpose – why is it being constructed?
- How is your intranet currently used by your end-users?
- What functionality and/or content exists currently that must be included in the new intranet?
- How often is your intranet content updated? By how many people?
- What content functionality and capabilities will your intranet offer?
- Do you want to integrate your intranet with other business systems?
- What are you seeking to offer from a personalization perspective?
- Is there a desire to centralize one or more processes within the intranet?
- What do you want the intranet to look like?
- What sort of user experience and functionality will you include?
 - *Note: This is not the same as what your intranet will look like.*

ARE YOU CLASSIC OR MODERN?

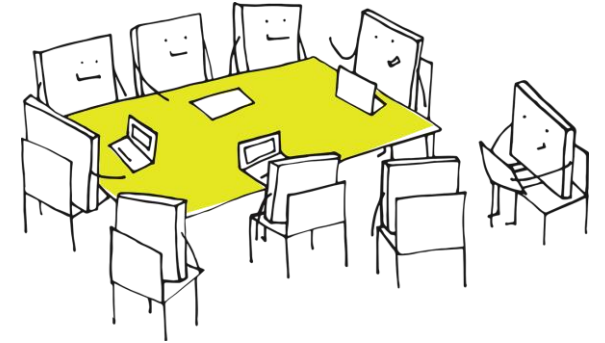
By that question, I mean “how do you view corporate intranets?”

- Do you view them as a “nice to have” place for news and announcements? A place for (largely static) policies and procedures? A primarily read-only environment used by corporate communications? This is the traditional view of an intranet.
- Do you view intranets as places where people spend most of their time working, collaborating, and communicating? As digital workplaces? Work hubs used by everyone – not just corporate communications? This is a mindset that has grown in mindshare particularly among younger employees and digital natives.

Know who you are building an intranet for!



ROLES: BUILDING THE OPTIMAL TEAM



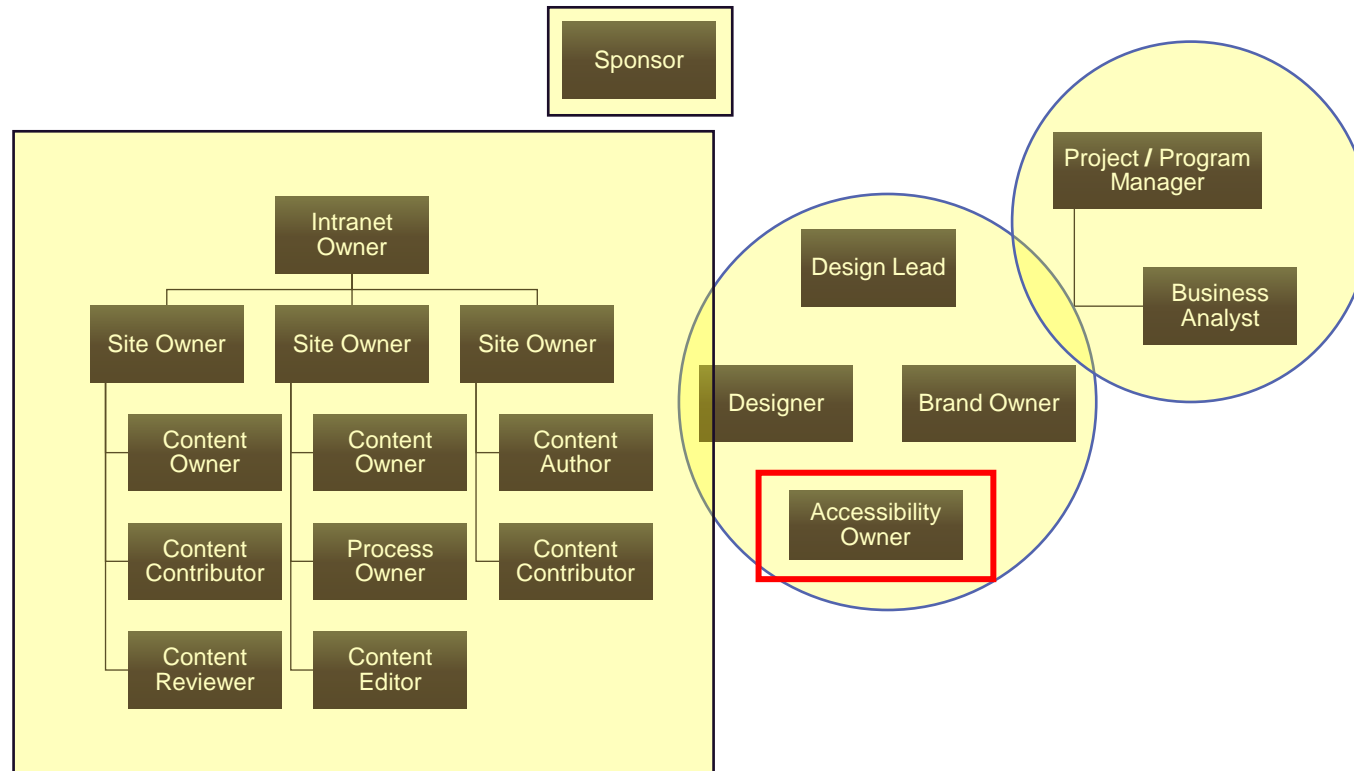
Team composition depends on many factors.

- The size of your organization.
- Your familiarity with the intranet platform.
- Whether you're building, implementing an intranet platform, or both.
- Whether you're working with outside consultants.
- The scope of requirements for the new intranet.
- The skill sets of those involved.

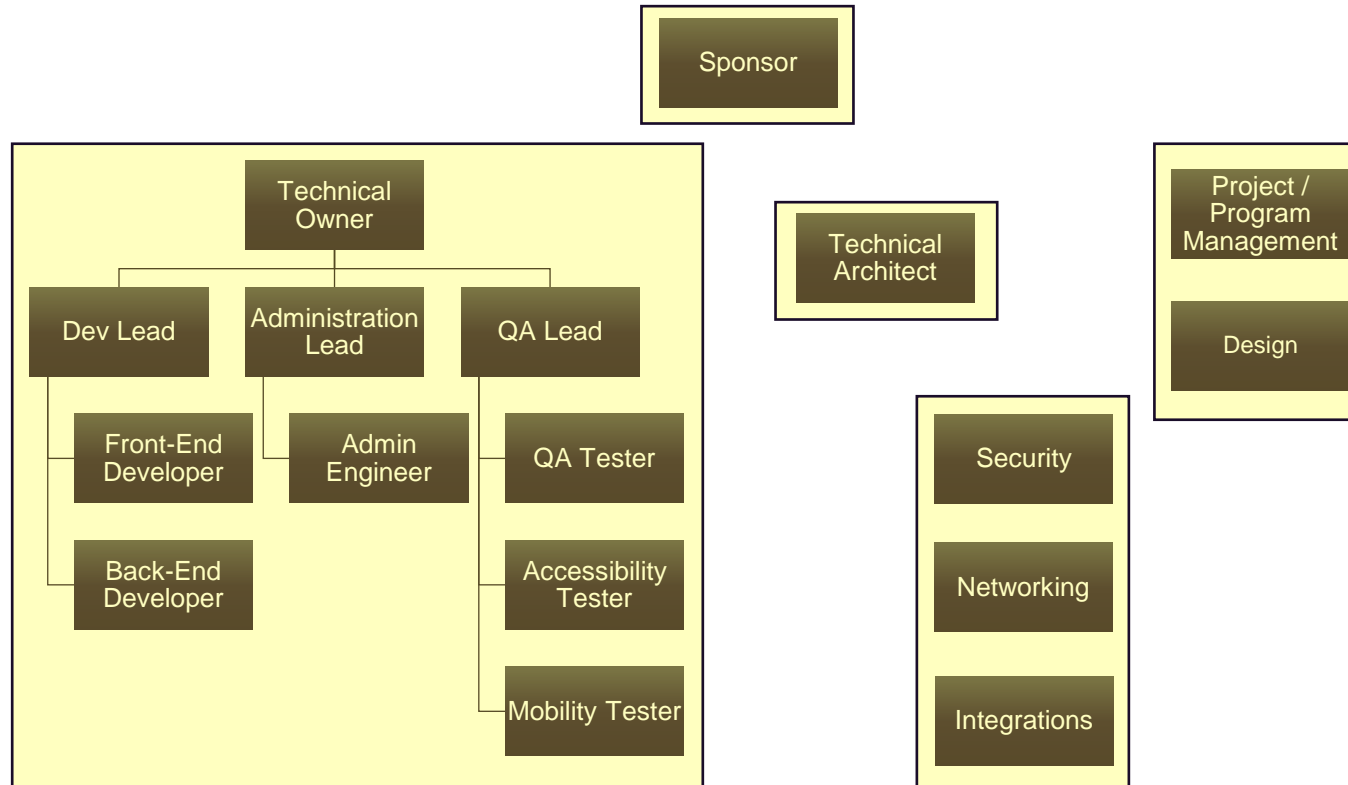
I tend to look at team composition through the following lens:

- Who are the content people?
- Who are the technical / platform people?
- Who are the “cat herders?”
- One or more people can be classified in multiple categories.

COMMON CONTENT-CENTERED ROLES



COMMON TECHNICAL ROLES



STEERING THE SHIP



When choosing your team and how to run your project, experience has shown:

- Project velocity is inversely proportional to the number of people “issuing” requirements.
- The “design by committee” approach that some larger organizations are fond of becomes a very real source of contention, delay, inefficiency, and rework.
- The most efficient and effective projects I’ve worked on have been those where the requirements (content) and implementation (tech) sides have had singular champions empowered to make binding decisions, give direction, and build.

BEFORE STARTING THE PROJECT



- How will you run your project? Hopefully with Agile ...
- How many environments will you need and have?
- Has the subject of migration been discussed (for intranet replacement projects)?
- Have DevOps been considered and addressed?
- Have you started thinking through a governance plan?
- Have you started planning for security?
- Do you have what you need to integrate additional systems?
- Do you have user stories and functional specifications?
- What are your plans for usage testing and acceptance?
- Have you started thinking about how you'll release?

**BULLETS TO REVIEW IN
DETAIL LATER (HOMEWORK!)**

THE RIGHT MINDSET

Recognize and acknowledge that building an intranet is *not* a prescriptive process.

Every intranet project I've worked on previously had started out with a set of target deliverables that underwent revision due to costs, complexities, and other factors that weren't known (or were in flux) at project inception.

At this stage of the game, flexibility and adaptability are highly desired team traits. This aligns with an Agile approach to delivery.



A SPECIAL WORD ABOUT IA

If you aren't familiar with the elements of SharePoint information architecture (IA), you absolutely must get acquainted during the planning phase (before implementation).

- Site Columns.
- Content Types.
- Managed Metadata and Term Sets.

What does IA afford you?

- Dramatically improved Search experience.
- Processes and workflows can easily target documents of specific categories.

Visit the magnificent [Sue Hanley](#) and her blog for more info.

It is exceptionally hard to “go back” and retrofit an IA structure onto a SharePoint site without impacting use.



STARTING OFF SMOOTHLY

Everyone wants to start off right. Some advice:

- As mentioned, assume one or more requirements will change once development is underway.
- Use an agile methodology to effectively manage priorities.
- Have admins, developers, architect(s), QA resources, BA resources, etc., fully opt into an agile approach.

PMs: support your team and don't "throw them to the wolves" by caving-in to new features and changing targets.

- Violations of the "Space-Time Continuum" commonly happen with less experienced scrum masters and project managers.
- If a feature gets added, one of equal weight or development time needs to be moved to the backlog.
- It's tempting to promise just "one additional feature." Don't.
- Your development team will thank you. 😊



WHAT ABOUT GOVERNANCE?

Make sure you start thinking and planning for how you'll manage your site once it's built.

- It may be tempting to view governance as something you'll do later, but you really need to start planning prior to release.
- Once you release, you've opened Pandora's box.



You'll need a strategy for many scenarios. A few of the more common ones:

- How will new sites get created? Who can request them?
- Who needs to approve permissions changes? Who carries out those changes?
- What is your plan for old and expired content? Archive it? Delete it? When?

Third Party tools can help, but only so far. Effective governance requires more than a just a software tool!

AND WHILE WE'RE ON THE TOPIC ...



Security, like governance, needs to be anticipated and addressed. Start the conversation and planning early.

- Will you use SharePoint groups? (Azure) Active Directory groups? Both?
- Will you need more than the three built-in modern site SharePoint groups?
- How granularly do you want to allow and control permissions?
- Will you enable newer content sharing mechanisms?
- Will you need custom permission levels for some content?
- Could a third-party tool help? (the answer is “yes”)

INTEGRATION FRUSTRATION

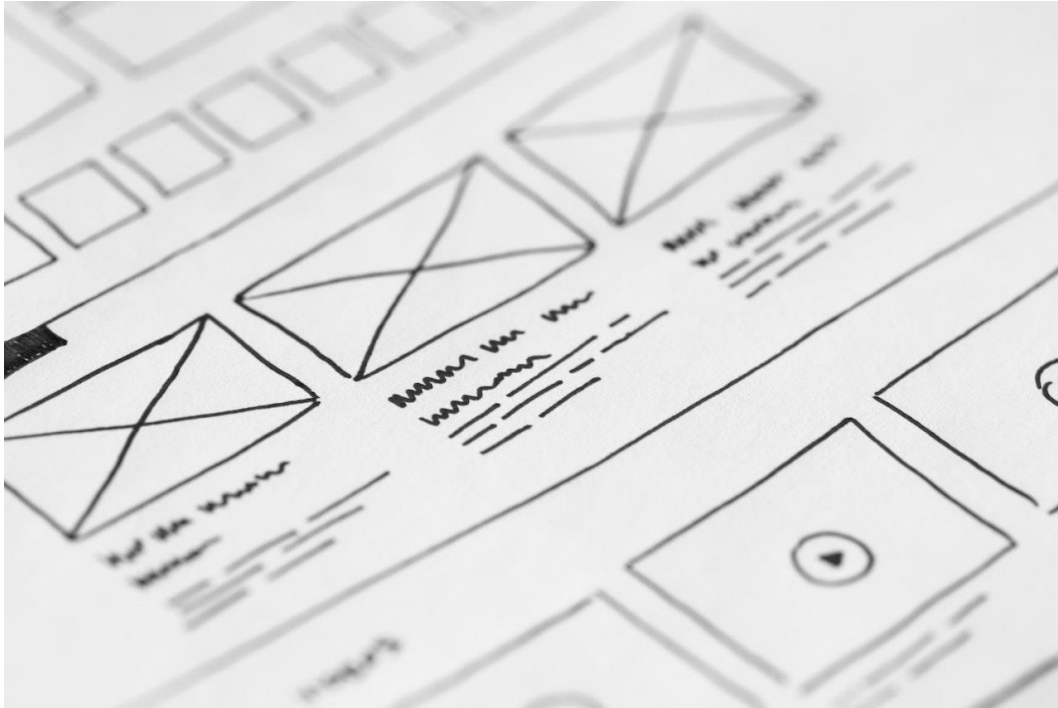
Integrating external services or line-of-business systems requires careful thought and planning:

- Are you using an intranet framework? Just SharePoint?
- Does the target system supply web services or integration points that can be consumed?
- Does the team possess the technical capability and knowledge to integrate?
- Do you have the potential for XSS and/or CORS issues?

Advice: limit your initial scope of integrations. They can become a bottomless pit and time sink. And make sure the requirements have been written in detail.



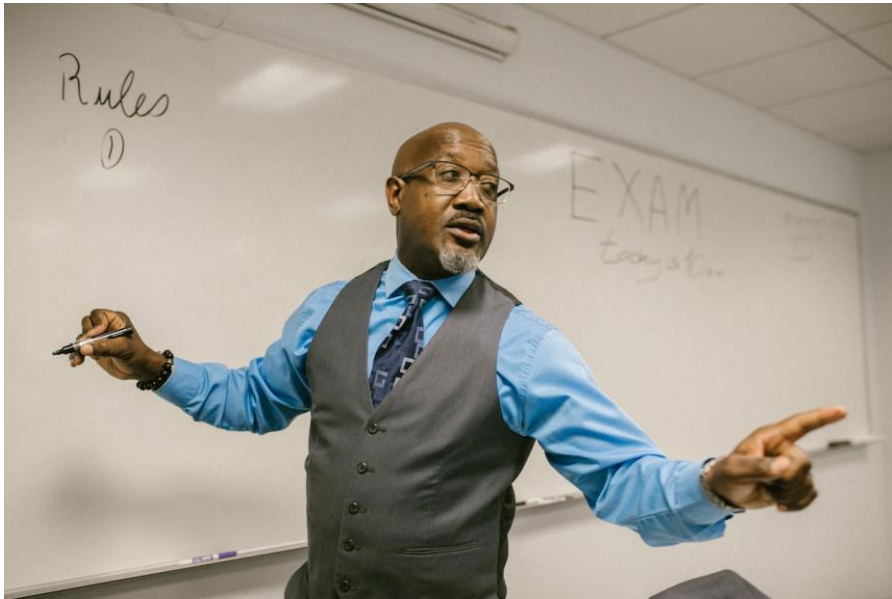
REQUIREMENTS



A wireframe **IS NOT** A user story

(and it is not a functional specification, either)

TESTING



If you don't let your user base see your work until you're done, you're missing the opportunity to adjust what you're building to meet their expectations.

- Incorporate testing early and often.
- Lean towards data-driven design and ask users for feedback at every opportunity.
- SMEs play a role, but don't undervalue feedback supplied by average users.
- This includes technical/feature testing and user acceptance testing.
- Please staff a QA team!

RELEASE TIME

The day you've been working towards all along – the moment when you can “flip the switch” and dazzle users!

- “Big bang” releases aren't recommended for cloud services like SharePoint Online
- They don't take advantage of the cloud properties like elasticity.
- Use Microsoft's [SharePoint Portal Launch scheduler](#) to stage the release out so that your SPO tenant can respond and scale seamlessly.
- See my [blog post](#) on the topic.



RELEASE TIME, PART TWO

Intranets seldom follow the plotline of “Field of Dreams.”

- Just because you build it doesn’t mean they’ll come.
- A plan to build user interest is recommended.

User adoption can be its own challenge, so take some tips from those who have scaled that mountain.

- Teasers and progressive reveals.
- Intranet scavenger hunts.
- A contest to name the new intranet.
- Featured employee spots.
- Intranet usage tips and tricks.



LATER STEPS AND CONSIDERATIONS

The period after release often entails transition.

- The team may transform with support and new personnel
- Project velocity commonly slows to varying degrees

About your support model

- Who's doing support?
- What are your SLAs?
- How many levels of support are there?
- How do you prioritize what gets worked on?

A support model and framework is something you will want to have in-place ahead of initial release.



OTHER THOUGHTS

Some (definitely not *all*) concerns now that you're "live"

- What are your plans for when the site goes down? Because it will.
- If you have only one environment, you might want to revisit that decision.
- What have you set up to gather user feedback, input, and suggestions?
- How will you handle team members leaving? There will undoubtedly be knowledge they have that you won't want to lose, especially for those that will have to fill their roles.
- Site performance is a common next area of focus once the site is in-use. Expect to have questions about optimizing and improving the user experience.
- Will you need to upgrade at some point? What does that look like?
- Do you have a plan to address the future and future needs in general?

**BULLETS TO REVIEW IN
DETAIL LATER (HOMEWORK!)**

Thank you

Final
Questions
?





Sean P. McDonough

sean@SharePointInterface.com

sean.mcdonough@akumina.com

LinkedIn: <https://www.linkedin.com/in/smcdonough/>

Blog: <https://SharePointInterface.com/>

About: <https://spmcdonough.com/>

REFERENCES

1. Linear toughened silicone grafted polymers
<https://patents.google.com/patent/US5919879>
2. Akumina | Employee Experience and Modern Intranet Software
<https://www.akumina.com/>
3. Understanding WSS 3 and MOSS 2007
<https://www.mssqltips.com/sqlservertip/3773/understanding-wss-3-and-moss-2007/>
4. Microsoft Content Management Server 2002 Service Pack 2 Resources
<https://www.andrewconnell.com/blog/microsoft-content-management-server-2002-service-pack-2-resources/>
5. SharePoint Online Public Websites to be discontinued
<https://learn.microsoft.com/en-us/sharepoint/troubleshoot/sites/public-websites-be-discontinued>
6. SharePoint, Team Collaboration Software Tools
<https://www.microsoft.com/en-us/microsoft-365/sharepoint/collaboration>

REFERENCES

7. Susan Hanley's Blog
<https://www.susanhanley.com/>
8. Project Management Institute | Agile Practices
<https://www.pmi.org/learning/featured-topics/agile>
9. ShareGate Migration
<https://sharegate.com/microsoft-migration>
10. Five reasons to avoid 'Full Trust' solutions in SharePoint
<https://rencore.com/blog/5-reasons-to-avoid-full-trust-solutions-in-sharepoint/#:~:text=Full%20Trust%20Farm%20Solutions%20allow,create%20custom%20functionality%20for%20SharePoint.>
11. SharePoint Add-ins
<https://learn.microsoft.com/en-us/sharepoint/dev/sp-add-ins/sharepoint-add-ins>

REFERENCES

12. GitLab | What is DevOps?

<https://about.gitlab.com/topics/devops/>

13. git – local-branching-on-the-cheap

<https://git-scm.com/>

14. Azure DevOps

<https://azure.microsoft.com/en-us/products/devops>

15. SharePoint governance overview

<https://learn.microsoft.com/en-us/sharepoint/governance-overview>

16. Dictionary.com | Pandora's Box

<https://learn.microsoft.com/en-us/sharepoint/governance-overview>

REFERENCES

17. Managing SharePoint Online Security: A Team Effort
<https://learn.microsoft.com/en-us/microsoft-365/community/sharepoint-security-a-team-effort>
18. OWASP | Cross Site Scripting (XSS)
<https://owasp.org/www-community/attacks/xss/>
19. MDN | Cross-Origin Resource Sharing (CORS)
<https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS>
20. IDF | Wireframing
<https://www.interaction-design.org/literature/topics/wireframing>
21. Atlassian | User Stories with examples and a template
<https://www.atlassian.com/agile/project-management/user-stories>

REFERENCES

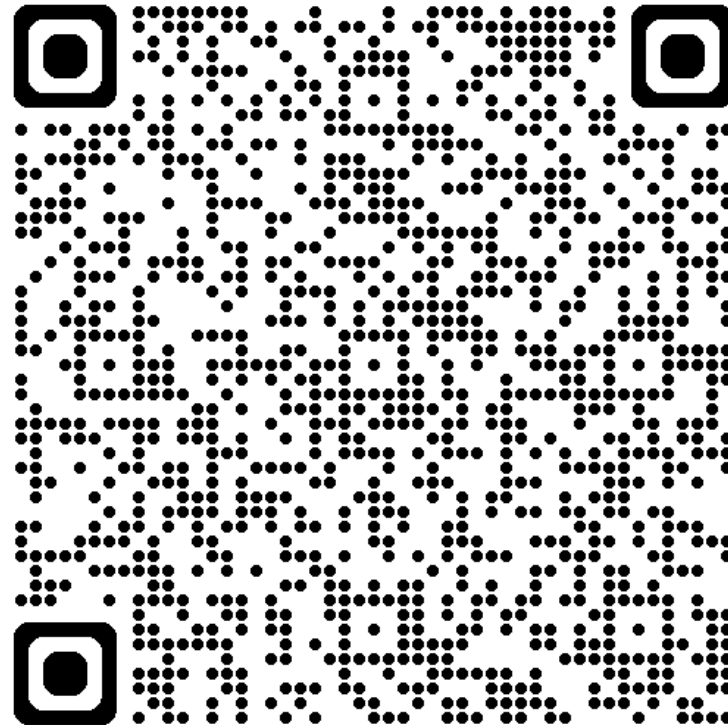
22. Launch your portal using the SharePoint Portal launch scheduler
<https://learn.microsoft.com/en-us/microsoft-365/enterprise/portallaunchscheduler?view=o365-worldwide>
23. Launching Your SPO Site or Portal
<https://sharepointinterface.com/2021/09/15/launching-your-spo-site-or-portal/>
24. Wikipedia | Field of Dreams
<https://learn.microsoft.com/en-us/microsoft-365/community/sharepoint-security-a-team-effort>
25. November 2014 MNSPUG – SharePoint User Adoption
<https://www.slideshare.net/mnspug/november-2014-mnspug-sharepoint-user-adoption>

Nashville Microsoft Community Days



FEEDBACK

**Fill out the
feedback form
to be eligible
for prizes!**



TNDays.com/NashSurvey

**Thank You
Valued Contributors!**

