Tech Communication / Knowledge Management Space

1. **Goals**
   - **a.** Improve consistency & quality of reference and workflow docs
   - **b.** Define standards
   - **c.** Curate Structure
   - **d.** Organize expert contributions
2. **Obstacles**
3. **Ideal Candidates**

Goals

1. **Improve consistency & quality of reference and workflow docs**
   - **a.** Docs as Code (DaC) (or hybrid of)
     - Keeps documents in sync with code, accurate, and prioritized
     - Writers/KMs should see themselves as value-adds to the team/ department
       - Agile stories, Epics, OKRs
       - Part of stand up, Sprint planning, and Roadmaps
     - Writers and Developers are immersed into the same spaces and collaborate on documents
       - Writers, if given the space to attend meetings, safely ask questions, and sandbox-type permissions, produce a more comprehensive document
   - **b.** Tech writing best practices
     - Follow document lifecycle / request process
     - Adhere to a template & naming convention
     - Understand 5 components of technical writing – Clear, Concise, Consistent, Comprehensive, Compliant
     - Create a style guide for consistency and expectations

2. **Define standards**
   - **a.** Maintain document consistency through use of templates, style guide, and best practices, etc.
   - **b.** Focus on "What" documents first to define standards, set clear expectations, align goals, timelines, & talking points
     - What do we do?
     - What do we support?
     - What are our goals?
     - What are our SLAs?
     - What are our policies?
     - What are our products/tools/services?
     - What are the benefits of using said products/tools/ services?
     - What are the regulations we must follow?
     - What are our markets?
     - What is the definition of important ?
     - What has changed?
     - What are the terms and conditions?
       - E.g., Marketing and Sales docs, Company Policies, Release Notes, Product Guides
   - **c.** After you have established the What, focus on the "How"
     - How do we do what we say we do?
     - How do the processes flow?
     - How do we install/ implement/test?
     - How do we troubleshoot?
     - How do we reach our markets?
     - How do we remain complaint?
     - How do services & dependencies talk to each other?
     - How do we accomplish our goals or meet our SLAs?
     - How do we satisfy customers/ support consumers?
       - E.g., How to guides & Job Aids, Business Continuity Plans, API/SDK docs, Swim lanes
   - **d.** Once the *What* and *How* documents have been created, then focus on the "Ed"
     - E.g., Training manuals, tutorial videos, web-based walk-thrus, Knowledgebase documents, User support manuals, FAQs

3. **Curate Structure**
   - **a.** Establish a single source of truth with a CMS
   - **b.** Understand asset management best practices – version control, document sharing, naming conventions, archiving
   - **c.** Enforce processes and template adherence
   - **d.** Launch a change management / process adoption campaign with benefits & incentives, timelines, and clear expectations (buy-in is important)
   - **e.** Evangelize and over-communicate dates and instructions, etc. through specific communication channels dedicated to the topic.
     - Share goals, accomplishments, and impactful product updates & new features in creative ways to keep users engaged and returning. This transparency builds trust and rapport for change management/ process adoption, audience participation in testing & feedback, and identifies FAQs and gaps in knowledge (All beneficial to document improvement)

   f. Have a clear path for document requests and prioritization, support vs ownership

4. **Organize expert contributions**
   a. This will come naturally with established processes, templates, structure, and rapport built through 1.a
   b. Ask open ended questions to fill in process gaps and eliminate knowledge silos
   c. Don’t make assumptions about personas and skill levels
   d. Shadow, interview, test with experts to become an SME as well
   e. Organize around What/ How / Ed may should help

Obstacles

1. User buy-in (what’s in it for me?)
2. Agreeing on the CMS / Single source of truth
3. Adhering to established templates, norms, & processes
4. Prioritizing documents alongside code
5. Large inventory of inconsistent, outdated documents scattered about multiple platforms (migration plan)
6. Unclear asset governance and retention policies
7. Delays inherent to document publishing process – Because our publishing service is still an MVP, there is some manual work. So, my pain point is that I cannot make live updates on the fly. I like to be able to fix a typo, a broken link, or make small tweaks as I receive feedback or discover. Consider options that allow the writer to make edits independent of developer intervention.
8. Considering multiple audiences with different permission levels – internal vs company vs external (each may need a different template / style to clearly distinguish between to avoid sharing sensitive information)
9. Mandates? Currently the struggle I am working through - Establish a centrally located document library for a platform of 11 teams. For our service to publish the document, it must be in .md and in a particular GitHub space; however, teams keep creating new documents in Confluence. The company does not want to remove access to Confluence, nor do they want to enforce GitHub which leaves the migration plan without a clear roadmap, milestones, and end date.

Ideal Candidate

- Technical writing, knowledge management, or communication background
- Min 5 yrs. experience (based on our conversation)
- Writing experience over techy savvy
- Project management skills or experience is a +
  - There is a great deal of iteration, nudging, and time management required to receive proper feedback from experts
- Asset management foundations
- Change management / process adoption understanding
- Process improvement background is a +
  - Not afraid to ask questions, dig deep, find underlying causes
- Ability to context switch / be flexible
- Can handle ad hoc tasks and uncertainty
- Figma is a ++ (or some other design related skills) – Infographics are crucial
  - Technical communication is an assortment of means to communicate technical information.
  - Visuals are key to explaining complex information/ processes, especially for English as a second language (ESL) colleagues and helps with accessibility.
  - Feedback from devs over the years – they don’t like to read, give them info in the quickest way possible.
- Insightful, creative, curious thinkers
- Asks for user feedback and engages with audiences
- Capacity to lead with confidence & hold their own among developers and influencers
- Understands how their role and documents benefit the team/ department/ company
- Knows Markdown, GitHub and / or the particular platform you wish to work from e.g. Confluence

Thoughts post discussion

- Tagging, smart search functionality, parent/ child relationships – I think of that as the information architecture of the CMS or document repo.