NRM Path Analysis

Evaluation Process: What to look for and red flags

U.S. Department of Agriculture, Forest Service

January 2, 2020

Background

The USDA Forest Service is about to embark on an important procurement to find a vendor team to assist them in building open source software that will enable members of the public to purchase firewood permits online.

Purpose of this guide

This document is intended to support the evaluation team and help them identify indicators of a strong vendor proposal. We've included three common technical evaluation criteria and both red and green flags that you might come across while reading proposals. **Note**: This is not intended to be an exhaustive list — every procurement is different.

Evaluation Factors

Technical Approach

Sample evaluation language:

> In evaluating an Offeror’s technical approach, the USFS will consider (a) the quality of the Offeror’s plans to provide the open source, agile development services required, including user research and design, (b) the extent of the Offeror’s understanding of the details of the project requirements, and (c) the extent to which the Offeror has identified potential obstacles to efficient development, and has proposed realistic approaches to overcome those potential obstacles.

What to look for:

- Competency. They should propose using the right tools in the right way and can defend their recommendations.
- A lack of novelty. The best approaches will recommend time-tested software and infrastructure, employing design patterns that are known to work.
- A lack of certainty. Maybe the vendor’s idea is a good one, maybe it’s a bad one — they can’t really know yet, and they need to know that. This should include highlighting weak points or areas of uncertainty in their technical approach.
- A vision. The vendor needs to *see* this, in a way that can act as a catalyst to the USFS' vision.
- A clear grasp on what the USFS is doing here. There should be no serious misunderstandings of information that was described clearly in the RFQ.
- Experience developing open source software.
- Collaboration. They expect the USFS' product owner to be a valuable, active member of the team.
- Constant usability testing. They plan to test the software with end-users throughout the entire development process.

Red flags:

- Misidentifying the name of technologies in a way that indicates a lack of experience communicating about them (e.g. “we’ll index records with an

Elasticsearch," instead of "with Elasticsearch," or "we recommend using JAVA," instead of "Java").

- Excessive complexity.
- Shirking page-limit rules (with tiny fonts, reduced leading, etc.) because they believe their technical approach to be so brilliant that it can't possibly fit within the prescribed limit.
- Basing their solution on fundamental misunderstandings of the USFS' needs that they should have understood.
- Proposing the use of arcane platforms and technologies, especially when those arcane platforms and technologies are house specialties of the vendor.
- They never once mention accessibility of the software to the 1 in 5 Americans with a disability.
- They don't consider, explicitly or implicitly, that user research will ultimately determine the approach, which in turn will dictate the technical approach.
- They're proposing to outsource what should be core competencies, e.g. DevOps, or Javascript.
- They propose a process that includes working for long stretches of time without interacting with the Forest Service and/or users of what's being built.
- They propose relying on focus groups, instead of structured, one-on-one usability testing sessions.

Staffing Approach

Sample evaluation language:

> In evaluating an Offeror's staffing approach, the USFS will consider (a) the skills and experience of the Key Personnel and other individuals that the Offeror plans to use to provide the required services, (b) the mix of labor categories that will comprise the Offeror's proposed development team, and (c) the Offeror's proposed number of hours of services to be provided by each member of the Offeror's proposed development team.

What to look for:

- A small number of team members, each providing a clear value. There is a purpose for everybody proposed.

- Familiarity with and use of modern software languages (e.g. Python, Ruby, PHP, C#, Javascript)
- Familiarity and use of web-based application programming interfaces (APIs), especially REST and GraphQL.
- Experience using Git for software version control.
- The lead developer’s skill mix and experience covers a substantial portion of the likely work needed to be done on the USFS' project.
- If the developers have presences on social coding platforms (e.g., GitHub), how does their work look? What expertises are evident there? Do they have expertise that don’t appear on their resume, but their work shows that they possess?
- Staff resumes support their claimed expertise. (For example, does the Content Designer have any actual content design experience, or are they a project manager in sheep’s clothing?)

Red flags:

- Overstaffing the bid. A team that consists of people with fancy titles, or far more experience than is necessary means that the vendor either doesn’t understand this way of working or is trying to squeeze every dime they can out of the engagement.
- Poorly-designed websites for the company, proposed subcontractor, or proposed staff resumes.
- Proposing antiquated software technologies that don’t have an active developer community (e.g. Cold Fusion, ASP, FoxPro)
- Lack of experience with test automation, aka DevOps, aka test-driven development (TDD).
- The lead developer isn’t impressive.
- The team has no apparent experience with usability research.
- The flashiest member of the team is proposed to spend a tiny amount of time on the project.
- Key skills don’t appear in any resumes, such as:
    - Automation, continuous deployment/delivery, DevOps
    - Migrating from one platform to another
    - Rapid prototyping
- No actual technical staff, but has “access to a database of resumes.”
- Proposed staff don’t currently work for the vendor, and there is no letter of intent from the proposed staff.
- Proposed staff resumes are copied from the internet, in large part or, more rarely, in whole.

Similar Experience

Sample evaluation language:

> In evaluating an Offeror’s similar experience, the USFS will consider the extent to which the Offeror has recently provided software development services for projects that are similar in size, scope, and complexity to the project described in this RFQ, and the quality of those services. In evaluating the quality of those services, the USFS will consider, among other things, the revision history for all files in the source code samples provided. The USFS will also consider the user research and design-related artifacts that were associated with the source code samples provided or submitted separately. In considering an Offeror’s similar experience, the USFS may also consider information from any other source, including Offeror’s prior customers, and public websites.

Caveat: Don’t spend time reviewing their Git repositories if you aren’t familiar with Git. This is something that a 18F tech lead and product owner will do and provide their thoughts and recommendations back to the team as technical advisors. A 18F tech lead and product owner will be looking for things like:

- Proper use of Git, commit changes with personal accounts (not organizational), use of a branching / merging strategy, informative comments, evidence of code reviews, and use of a CI/CD pipeline.
- Code that conforms well to our RFQ’s QASP.
- Git collaboration. Work was performed in reasonable ~~an implausibly tiny~~ number of GitHub comments.
- Substantial projects. The projects were not created just to have something to point to for this RFQ.
- How they incorporate user feedback into their development process.
- That their tests are written well, and cover the supermajority of the code.
- Consistent, enforced code style.

What you should look for:

- Work that is conceptually similar to the USFS' needs, i.e. a system in which a record is submitted, and then moves along an assembly line with different controls and modifications to the record along the way.
- Work that was centered on user needs, as opposed to leading with solutionism.
- Work that was completed by a team of a size that's similar to the size of the team that they're proposing.
- Design artifacts that show continuous and ongoing usability testing that show a user-centered approach to iterative design and development.

Red flags:

- The cited projects bear little evidence of having been created by the vendor.
- The projects are trivial.
- There's a finished product, but no code, or vice-versa.
- The projects do not include good design artifacts and research plans.