ITAViP Toolkit: Technology

Technology

This section of the toolkit is focused on activities that will help you with establishing, stabilizing, and evolving sustainable technology for your program. It is designed to help you structure discussions and gain consensus about next steps for sustainable technology for your OSS program. It is not a toolkit for day-to-day software development.

Definition: The core of each program supported by It Takes a Village is open-source software or systems serving cultural and scientific heritage organizations. There are parallels with proprietary software development processes, but working within the open source world brings its own challenges around community, resources, and governance that affect the software development process.

Instructions

Brand new to It Takes a Village in Practice? Check out our Getting Started Resources before jumping in!

Review the activities and goals below to select ones you want to use for sustainability planning. These activities are designed to help you move from one phase to the next and can be used in any order. You can use any or all of them.

These activities are currently available for beta testing. As you use them, the ITAViP Co-Directors are eager for feedback via ITAV@lyrasis.org

<table>
<thead>
<tr>
<th>Phase 1: Laying the Groundwork</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Phase 1 Objectives</strong> include: Understand Core Community Needs, Continue to Gather Data, and Communicate Progress and Process with Stakeholders</td>
</tr>
<tr>
<td><strong>1. Who Are Your Technology Stakeholders?</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>2. Technical Skills Inventory (Part One)</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>3. Personas and Pathways</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>4. Landscape Analysis</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>5. Buy A Feature Game</strong></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td><strong>6. Community QA Testing</strong></td>
</tr>
<tr>
<td></td>
</tr>
</tbody>
</table>
7. Documentation
Friction Logging
Goal: Understand how usable your current documentation is via friction logging with documentation users.
Related Objectives: Continue to Gather Data

Phase 2: Expanding and Integrating

Phase 2 Objectives include: Engage the Community, Grow Thoughtfully, and Consider Integration over New Development

8. Shadow Observations
Goal: Gain insight about users’ jobs, pains, and gains; learn about users’ day-to-day work in a real-life setting; and incorporate end-users’ experiences into program planning.
Related Objectives: Engage the Community

9. Recognition and Contributions
Goals: Understand how the program currently recognizes contributions, understand the kind of recognition that is most appreciated/motivating to each kind of contributor, and discover how the program can improve in this area.
Related Objectives: Engage the Community

10. Value Propositions for Job Descriptions
Goals: Create a value proposition for a new program position (e.g. community manager, technical lead) to help articulate significance of allocating resources to program leadership, and optionally, create a job description for a new program position.
Related Objectives: Engage the Community

11. Building Welcoming Communities
Goals: Identify what documentation your program has that supports growing a successful community, and prioritize missing documentation and develop a plan for its development.
Related Objectives: Grow Thoughtfully

12. Not Invented Here: Assessing Integration Options
Goals: Identify functional requirements or feature requests that may be solved by integration, and identify and assess potential integration solutions to satisfy functional requirements or feature requests.
Related Objectives: Consider Integration over New Development

Phase 3: Preparing for Change

Phase 3 Objectives include: Reassess Community Needs, Plan for Evolution, and Document an Exit Strategy

14. Technology Stakeholders: Does the Shoe Still Fit?
Goal: Re-identify and re-prioritize the community’s technology stakeholders, create a high-level map of the technology stakeholders’ characteristics, and support program team’s work developing strategies to advance technology sustainability.
Related Objectives: Reassess Community Needs

15. Catastrophizing - Tech Edition (Part One)
Goals: Understand how the program’s current technical staff, stack, and roadmap work with unexpected issues (catastrophes), and help programs identify how resilient their staff, stack, and long-range technical strategy are.
Related Objectives: Reassess Community Needs

16. List of Dreams
Goals: Articulate what future directions your core application could take, assess skills and technology requirements for future directions, and plan for and prioritize professional development opportunities for program stakeholders (staff, contributors, trainers, etc.).
Related Objectives: Reassess Community Needs

Goal: Turn the issues identified in Part 1 of Catastrophizing into actionable plans.
Related Objectives: Plan for Evolution

18. Technical Skills Inventory (Part Two)
Goals: Update your program’s inventory of what technical skills are needed for technical staff, contributors, and users to develop, support, and maintain the platform; and identify new skills that your program will need to acquire or old skills that can be phased out as the program advances.
Related Objectives: Plan for Evolution

19. Developing an End-of-Life Communication Plan
Goal: Develop plan to communicate end-of-life decisions with community.
Related Objectives: Document an Exit Strategy

20. Winding Down an Open Source Software Program
Goal: Decide when a project or program is no longer useful, understand how to disengage from a project or program, and determine what to do about code, repositories, websites, wikis, and other project or program assets.
Related Objectives: Document an Exit Strategy