Global Governance Frameworks - framework development guide

Welcome to the Global Governance Frameworks frameworks development! This guide will help you create comprehensive, actionable governance frameworks that enable different systems to interoperate and evolve together.

🎯 What are governance frameworks?

Governance frameworks are structured approaches to decision-making, coordination, and accountability that can be adapted across different contexts and scales. Unlike rigid prescriptive models, our frameworks provide flexible tools and patterns that enable various governance systems to communicate, collaborate, and learn from each other.

Our frameworks address real-world challenges through practical tools that organizations, communities, and governments can implement immediately while contributing to larger systemic transformation. Each framework represents both a standalone solution and a component in a broader ecosystem of governance innovation.

📋 Getting started

Before beginning framework development, familiarize yourself with our existing frameworks to understand scope, structure, and approach. Review the Style Guide for formatting requirements and the Treaty for Our Only Home as an example of comprehensive framework development.

Consider what unique expertise and perspective you bring to governance challenges. Are you experienced in education, healthcare, environmental policy, or technology governance? Do you work with local communities, regional organizations, or international institutions? Your background will inform both the framework's focus and its practical applications.

Join our Discord community to connect with other framework developers and explore collaboration opportunities. Many successful frameworks emerge from partnerships between domain experts, governance theorists, and implementation practitioners.

🔧 The framework authoring workflow

Creating effective governance frameworks requires systematic development that moves from conceptual clarity through detailed content creation to practical implementation tools.

Phase 1: outline perfection

A comprehensive, logical outline forms the foundation of every successful framework. This phase requires the most careful attention because structural problems become exponentially harder to fix once content creation begins.

Initial brainstorming starts with your expertise, research, and understanding of specific governance challenges. What problems are you trying to solve? Who are your primary audiences? What existing approaches have succeeded or failed? What unique insights can you contribute to the field?

AI-assisted iteration can help structure and refine your thinking. Use powerful models like Claude to explore different organizational approaches, identify gaps in your reasoning, and suggest additional considerations. The goal is not to let AI create your framework but to use it as a sophisticated thinking partner that helps you develop your ideas more thoroughly.

Achieving perfection means your outline is comprehensive, logical, and complete before any drafting begins. Every major section should have clear purpose and relationship to other sections. The flow should guide readers naturally from understanding problems through exploring solutions to implementing practical changes. Test your outline by explaining it to colleagues or presenting it to potential users for feedback.

This is the most critical step in framework development. A perfect outline almost guarantees a coherent final document, while a flawed outline creates problems that persist through every subsequent phase.

Phase 2: content drafting and asset creation

With a solid outline established, you can begin creating the detailed content that transforms conceptual frameworks into practical tools.

Creating the index.md file provides your framework's central landing page and table of contents. This file should clearly explain what the framework addresses, who should use it, and how different sections connect to create a comprehensive approach. Follow the patterns established in existing frameworks while adapting to your specific content and audience.

Drafting with AI oversight involves using multiple powerful models to ensure both quality prose and robust analysis. Use Claude to draft content for each section based on your perfected outline, ensuring consistency with

our style guide and maintaining focus on practical implementation. Use Gemini or another model for oversight, reviewing drafted text for accuracy, clarity, alternative perspectives, and deeper insights that might strengthen the framework.

This two-AI system helps balance efficiency with quality while maintaining your voice and expertise as the primary author. The AI tools accelerate content creation but your judgment determines which suggestions to accept, modify, or reject.

Structuring sections with front-matter requires organizing content into separate files for each major section, ensuring each follows our technical specifications for metadata, formatting, and integration with the website's navigation system. Consult the Style Guide for specific requirements and examples from existing frameworks.

Phase 3: tools, visuals, and resources

Governance frameworks become most useful when they include visual explanations, practical tools, and resources that support implementation efforts.

Creating visuals (SVGs) helps make complex governance concepts accessible and memorable. Use Claude to generate initial code for SVG visuals based on your descriptions of processes, relationships, or organizational structures. These AI-generated starting points typically require manual revision and perfection using vector editors like Inkscape to achieve professional quality and visual clarity.

Focus on visuals that genuinely illuminate your framework's key concepts rather than decorative elements. Process diagrams, organizational charts, decision trees, and relationship maps often prove most valuable for governance frameworks.

Preparing downloadable tools (PDFs) extends your framework's utility through standalone resources that organizations can use in meetings, training sessions, or planning processes. These might include worksheets, assessment tools, reference cards, or implementation checklists that translate framework concepts into actionable steps.

To add PDF tools to the website, you'll need to update the build script located at `/scripts/pdf-generator.js`. Use Claude to help generate the required JavaScript snippets that add your new files to the automated PDF generation process.

Developing implementation resources might include templates for organizational assessments, facilitation guides for workshops, presentation materials for introducing frameworks to stakeholders, or case study templates for documenting implementation experiences.

Phase 4: review and refinement

Quality frameworks emerge from iterative improvement based on feedback from multiple perspectives and expertise areas.

Self-review involves systematically checking your framework against our quality standards. Does every section serve a clear purpose? Are recommendations practical and actionable? Does the flow guide readers logically from understanding to implementation? Are examples relevant and diverse? Do visuals genuinely clarify rather than complicate concepts?

Community review provides external perspectives that improve both content quality and usability. Share drafts in Discord channels for feedback from other framework developers, subject matter experts, and potential users. Request specific feedback on areas where you feel uncertain or want additional validation.

Expert consultation may be valuable for frameworks addressing highly technical or specialized domains. Reach out to practitioners, academics, or policy experts who can validate your approaches and suggest improvements based on their experience.

User testing with representatives of your target audience reveals gaps between your intentions and reader understanding. Can they follow your reasoning? Do they find recommendations practical? What additional support would help them implement your framework?

Phase 5: submission and integration

Once content and assets are complete, the framework enters the formal review process for inclusion in our ecosystem.

Pull request submission involves opening a GitHub pull request with all framework files, following our contribution guidelines for branch naming, commit messages, and documentation. Include a clear description of the framework's purpose, target audience, and relationship to existing frameworks in the ecosystem.

Review process includes evaluation by core team members and community reviewers who assess content quality, alignment with our principles, practical utility, and technical correctness. Be prepared to address feedback and make revisions based on reviewer suggestions.

Integration planning coordinates with website developers who will create the technical infrastructure needed to make your framework accessible through our site. This includes page creation, navigation updates, and integration with our download and tool systems.

📝 Content guidelines and standards

Creating frameworks that work effectively across different contexts requires attention to both substantive quality and technical consistency.

Writing quality standards

Clarity and accessibility ensure frameworks remain useful for diverse audiences with varying levels of governance expertise. Write in active voice when possible, define technical terms clearly, use concrete examples to illustrate abstract concepts, and structure information hierarchically to support different reading approaches.

Practical focus distinguishes our frameworks from purely theoretical treatments. Every recommendation should include specific implementation guidance, examples should come from real-world applications, and abstract principles should connect clearly to actionable steps.

Cultural sensitivity recognizes that governance approaches must adapt to different cultural contexts while maintaining core principles. Avoid assumptions about organizational structures, decision-making processes, or authority relationships that may not translate across cultures.

Evidence basis grounds recommendations in research, case studies, and proven practices while acknowledging limitations and areas requiring further development. Cite sources appropriately and distinguish between established knowledge and innovative proposals requiring testing.

Technical formatting requirements

Markdown structure follows our established patterns for headers, sections, cross-references, and metadata. Consult the Style Guide for specific formatting requirements and examples from existing frameworks.

Front-matter specifications include required metadata fields that enable proper site integration, navigation, and search functionality. Each section file needs appropriate front-matter following our established templates.

Cross-reference consistency ensures links between framework sections and to other site content work correctly and support reader navigation through related materials.

Asset integration requires proper file naming, placement, and referencing for images, PDFs, and other resources that accompany framework content.

🛠️ Tools and development resources

Framework development benefits from using appropriate tools that support both creative thinking and technical precision.

Writing and collaboration tools

AI assistance can accelerate content creation while maintaining quality through strategic use of models like Claude for drafting and Gemini for oversight and review. Use AI as a sophisticated thinking partner rather than a replacement for your expertise and judgment.

Version control through Git enables collaboration, tracks changes, and maintains backup copies of your work throughout the development process. Learn basic Git workflows if you're not already familiar with them.

Markdown editors like Typora, Mark Text, or VS Code with markdown extensions provide better writing environments than basic text editors, including live preview, table editing, and formatting assistance.

Visual design tools

Vector graphics creation requires tools like Inkscape (free) or Adobe Illustrator for creating and editing SVG files that integrate well with web content and maintain quality at different sizes.

Diagramming tools like Draw.io, Lucidchart, or Figma can help conceptualize visual representations before creating final SVG versions.

Color and design consistency should follow our established visual identity guidelines while allowing frameworks to develop distinct visual personalities that support recognition and navigation.

Testing and feedback platforms

Community channels provide spaces for sharing drafts, requesting feedback, and coordinating with other developers through Discord conversations and GitHub discussions.

User testing can be facilitated through video calls, surveys, or facilitated workshops where potential users interact with draft frameworks and provide structured feedback.

Expert review networks help connect framework developers with domain experts who can validate technical content and suggest improvements based on field experience.

🤝 Collaboration and coordination

Framework development often benefits from collaboration between complementary expertise areas and perspectives.

Finding collaboration partners

Skill complementarity works well when domain experts partner with writing specialists, governance theorists collaborate with implementation practitioners, or technical experts work with community organizers who understand user needs.

Geographic diversity strengthens frameworks by incorporating perspectives from different regions, governance traditions, and cultural contexts that reveal assumptions and expand applicability.

Sectoral expertise combinations create more robust frameworks when contributors understand different organizational contexts like government agencies, nonprofits, businesses, and community groups.

Coordination mechanisms

Project planning tools help collaborative teams track progress, assign responsibilities, and maintain momentum through framework development phases. Use Discord channels, GitHub projects, or external tools like Trello depending on team preferences.

Regular check-ins maintain alignment and address challenges before they become serious problems. Weekly or bi-weekly video calls often work well for active development phases.

Version control coordination requires clear branching strategies and merge procedures when multiple contributors are editing framework content simultaneously.

Role clarity prevents conflicts and ensures comprehensive coverage by establishing who takes primary responsibility for different framework sections, visual design, tool development, and coordination tasks.

📊 Quality assurance and testing

Effective frameworks undergo systematic testing and refinement before publication to ensure they work as intended for target audiences.

Content validation approaches

Logic and consistency checking involves systematic review of arguments, recommendations, and examples to ensure internal coherence and alignment with stated principles.

Completeness assessment evaluates whether frameworks address their stated scope adequately and provide sufficient guidance for implementation without overwhelming users with unnecessary detail.

Practical utility testing involves presenting frameworks to potential users and evaluating whether they can successfully apply recommendations to their specific contexts and challenges.

Cultural sensitivity review examines assumptions, examples, and language choices that might create barriers for users from different cultural backgrounds or organizational contexts.

User testing methodologies

Structured feedback sessions with representative users provide systematic evaluation of framework usability, clarity, and practical value through guided discussion of specific sections and tools.

Implementation pilots involve working with organizations willing to test framework applications in real contexts, providing valuable feedback about what works, what doesn't, and what additional support would be helpful.

Expert review panels bring together specialists in relevant domains to evaluate technical accuracy, alignment with best practices, and contribution to existing knowledge in the field.

Iterative improvement incorporates feedback systematically while maintaining framework coherence and avoiding scope creep that dilutes focus or usability.

🌍 Impact and implementation support

Creating frameworks represents only the beginning of their potential contribution to improved governance practices.

Supporting framework adoption

Implementation guidance helps organizations translate framework concepts into specific actions appropriate to their contexts, constraints, and capabilities.

Training materials enable framework developers to support adoption through workshops, presentations, and educational content that builds user capacity and confidence.

Community building around specific frameworks creates networks of practitioners who can share experiences, solve implementation challenges collaboratively, and contribute to framework evolution based on real-world application.

Case study development documents framework implementations to provide examples for future users and insights for framework improvement based on practical experience.

Measuring framework impact

Adoption tracking monitors how many organizations are using frameworks and in what contexts, providing insights about reach and areas for additional development.

Implementation outcomes evaluate whether framework adoption leads to improved governance practices, better decision-making, or other intended results in user organizations.

Ecosystem contributions assess how individual frameworks contribute to the broader goal of enabling governance system interoperability and coordination.

Continuous improvement incorporates learning from implementation experience into framework updates that increase effectiveness and usability over time.

🚀 Getting started with your first framework

Beginning framework development can feel overwhelming, but systematic approaches make the process manageable and rewarding.

Choosing your focus area

Start with expertise areas where you have substantial knowledge, experience, or passion that can contribute unique insights to governance challenges.

Identify gaps in existing resources where new frameworks could provide valuable tools for practitioners, organizations, or communities facing specific governance challenges.

Consider scope carefully to ensure your framework addresses a meaningful problem without becoming unwieldy or trying to solve everything simultaneously.

Assess feasibility based on your available time, access to collaboration partners, and resources needed for comprehensive framework development.

Development timeline planning

Phase planning breaks framework development into manageable stages with clear milestones and deliverables that maintain momentum while ensuring quality.

Time allocation recognizes that outline development typically requires more time than initially expected, while content creation can proceed quickly with good structural foundation.

Collaboration coordination builds in time for review cycles, feedback incorporation, and revision that improve framework quality through multiple perspectives.

Buffer planning accounts for unexpected challenges, scope adjustments, and the iterative nature of creating tools that work effectively in practice.

First steps for new developers

Join community channels to connect with experienced framework developers who can provide guidance, answer questions, and suggest collaboration opportunities.

Study existing frameworks to understand our approach, quality standards, and technical requirements while identifying opportunities for new contributions.

Start small with focused frameworks that address specific challenges rather than attempting comprehensive solutions to broad governance problems.

Plan for feedback by identifying potential reviewers and user testers early in the development process rather than waiting until content is complete.

Framework development represents one of the most impactful ways to contribute to improved governance practices worldwide. Your expertise, perspective, and creativity can create tools that enable better coordination, more

effective decision-making, and stronger democratic practices across diverse contexts.

The governance challenges facing humanity require collaborative innovation that combines deep expertise with practical wisdom. By contributing frameworks, you join a global community working to build the infrastructure for more cooperative, resilient, and just governance systems.

Ready to start? Join our Discord community in the **#framework-development** channel and connect with other developers who can support your framework creation journey.