# POSE Application (Assembled)

## Project Summary (with Keywords)

Overview

PolicyEngine proposes to establish a sustainable and robust open‑source ecosystem (OSE) that supports community‑driven, distributed development and deployment of transparent tax‑and‑benefit microsimulation. We will transition an already deployed open‑source product into a well‑governed ecosystem with security, privacy, and contributor pathways that enable adoption across academia, government, and civil society.

Intellectual Merit

The project advances the state of open policy microsimulation by combining validated rules (federal/state tax and major benefits), enhanced microdata (imputation via quantile regression forests), and gradient‑descent reweighting (MicroCalibrate, optionally with L0 sparsity). We formalize governance, quality controls, and supply‑chain security to enable reproducible, high‑fidelity analysis and faster policy learning cycles.

Broader Impacts

Broader impacts include democratizing access to high‑quality policy analysis; improving transparency and reproducibility in applied economics; accelerating evidence‑based policymaking; and training contributors in secure, open development practices. We emphasize broad participation (onboarding, mentorship, documentation) and responsible data use through privacy‑preserving workflows.

Keywords: CISE; Public Finance; Social Policy; Open Source; Reproducibility

## Context of OSE

### Vision and Need

Our long-term vision is a secure, community-governed open-source ecosystem for transparent analysis of tax-and-transfer policies that supports timely, reproducible insights for decision-makers and the public. The societal need is clear: fiscal and social policy debates frequently rely on proprietary models that limit scrutiny, reproducibility, and open access. An open, well-governed OSE lowers barriers to participation and accelerates evidence-based policymaking.

### Product and Maturity

We are transitioning an existing open-source microsimulation product with an active user base into a sustainable OSE. The product supports configurable tax/benefit rules, dataset adapters, and reporting pipelines. It has external usage in research, journalism, and policy analysis. Current development follows open practices (issues/PRs, CI, releases), and we will formalize these into a durable ecosystem.

### Guiding Principles

We ground the OSE in four practical commitments. First, transparency: all rules, assumptions, and model code are public, and analyses are reproducible end‑to‑end. Second, security and quality: we integrate supply‑chain safeguards, code review, testing, and disciplined release hygiene. Third, open participation: we make contributor pathways explicit and supported with documentation and mentorship so new maintainers can join and succeed. Finally, sustainability: we establish governance and funding practices that allow the ecosystem to operate as durable public infrastructure.

### Anticipated Broader Impacts

Broader impacts include democratized access to policy analysis, improved reproducibility in applied economics, accelerated policy learning cycles, and contributor upskilling in secure open-source practices. Government and civil society benefit from a shared, auditable modeling base.

### Pointer to Open-Source Product

Per solicitation, we will include a formal citation to the public repository in References Cited and avoid URLs in the Project Description body. The citation will identify the repository, versioned releases, and key artifacts.

### Phase I Context and I‑Corps Plan

We received a Phase I award in August 2025. Because that was on the registration deadline for the August cohort, we will participate in the January I‑Corps for POSE cohort. Team: PI, co‑PI, and an external OSE mentor (named in the personnel list). Commitment: ~41 hours per team member plus 100 interviews (mentor not required to interview beyond kickoff). Outputs will directly feed finalization of governance policies, contributor onboarding, and the security plan.

### Modular Architecture and Current Components

To support sustained community development, we have separated core capabilities into focused packages with independent lifecycles. L0 provides PyTorch L0/L0+L2 regularization and gating primitives for differentiable sparsity and feature selection in calibration workflows. MicroImpute offers quantile regression forest imputation alongside baseline methods with benchmarking based on quantile‑loss. MicroCalibrate implements gradient‑descent reweighting to match administrative targets for state and congressional districts; it integrates L0 sparsity via Hard Concrete gates, supports hyperparameter tuning and holdout robustness checks, and supersedes our earlier prototype. These methods underpin our enhanced microdata and calibration work (Ghenis & Woodruff 2024). Each package is documented, tested, and versioned so contributors can work in well‑scoped areas that roll up to the broader OSE.

### Solicitation Alignment (Phase II Requirements Coverage)

This narrative covers all Phase II elements. The Context of OSE establishes vision, need, current status, novelty, and team qualifications and points to public repositories via References Cited. Ecosystem Establishment/Growth describes discovery, engagement of users and contributors, partner recruitment, milestones, and metrics. Organization and Governance explains licensing, decision‑making, working groups, and accountability. Continuous Development Model details open, distributed development with CI/testing, releases, and backports. The Risk Analysis/Security Plan addresses access control, secure SDLC, SBOM and signed releases, incident response, provenance, and privacy. Community Building outlines onboarding, mentorship, support, and recognition. Sustainability covers both financial and operational sustainment. Evaluation Plan and Metrics specifies adoption, quality, security, and calibration metrics and how we will measure them. Required attachments will be provided per PAPPG: letters of collaboration from current users and contributors, the Project Personnel/Collaborators list, the budget and justification, Facilities/Other Resources, a two‑page Data Management & Sharing Plan, SciENcv biosketches for senior personnel, Current and Pending (Other) Support, and any required documentation for equipment, consultants, and subawards in Phase II.

## Ecosystem Establishment/Growth

We will grow external contributors and maintainers, increase deployments across academic, civic, and policy organizations, and improve issue and review throughput as well as release cadence. The January I‑Corps for POSE cohort anchors our discovery work: we will conduct roughly one hundred structured interviews across policy analysts, researchers, journalists, civic‑tech developers, and state and local staff, synthesize the adoption blockers we hear, and translate those findings into a public roadmap that unblocks real users. We include a design‑freeze milestone in the first quarter to incorporate I‑Corps outcomes before the broader rollout. In parallel, we will cultivate pilot adopters and contributor pipelines with research partners such as NBER‑affiliated teams and universities including Georgetown and USC, along with think tanks and civic‑tech groups.

Contributor onboarding will feel purposeful and supportive. We will publish starter issues and step‑by‑step contributor guides, pair new contributors with maintainers, and hold regular office hours. Quarterly workshops and hackathons will bring users and maintainers together, and we will add fellowship opportunities where appropriate. For users, we will ship complete examples and datasets and provide “one‑command” analyses so the first run succeeds; releases will be accompanied by migration notes and clear change summaries.

Where it helps adoption, we will engage industry and international collaborators for use‑cases and contributions under compatible licenses, making our contributor and licensing guidance explicit from the outset. Over the first year, we expect to complete interviews and publish the adoption‑blockers report and onboarding materials; establish baseline security practices including SBOM and signed releases; ship high‑value data adapters; and complete at least two pilots. By the end of the first year we intend to have community maintainers operating in at least two domains and several case studies documenting impact and performance improvements.

### Commercial Adoption Channels

PolicyEngine’s OSE is strengthened by commercial users whose day‑to‑day needs closely overlap with public‑interest analysis. Tax accountants and tax‑software vendors can use the rules engine to test scenarios and ensure compliance across federal and state regimes; when they request a new provision or fix, our contribution playbook makes it easy to upstream the change—either by submitting a pull request themselves or by funding us or a maintainer to implement it with tests and documentation so it benefits the wider policy community. Market‑research organizations and retailers can use our calibrated local microdata (with forthcoming consumption integration) to understand county‑level demand, affordability, and incidence under policy changes; their improvements to data adapters, validation, or pipelines are likewise upstreamed and versioned. Financial institutions can apply the same local datasets and rules for economic projections and stress‑testing; where they underwrite performance improvements, we turn them into documented, reproducible releases. In each case, a permissive license, clear governance, and a straightforward CLA/DCO path align commercial incentives with public benefit.

Two concrete integrations illustrate immediate momentum. First, the UK Cabinet Office engagement is already informing government‑grade workflows; we will capture the lessons into governance and adoption playbooks and seek a formal letter of collaboration. Second, we will develop calibrated weights for states and congressional districts and package them as reusable data artifacts with documented pipelines, supported by Arnold Ventures in the U.S. following earlier UK work funded by the Nuffield Foundation. Alongside these, we will publish stable releases of L0, MicroImpute, and MicroCalibrate with quickstarts and examples and invite maintainers to steward each module.

We will track progress using a small set of meaningful indicators: non‑team pull requests and issues per month, active maintainers and time‑to‑review, release frequency and the number of named deployments, and the reproducibility of published analyses. These measures are simple to compute and make adoption and contributor health visible to reviewers and partners.

### Current Users and Collaborators

PolicyEngine already supports real policy workflows. The U.S. Joint Economic Committee has used the platform to examine distributional and budget questions. New York State Senator Andrew Gounardes’s office and D.C. Councilmember Zachary Parker’s office have both used the rules engine to simulate tax‑credit reform proposals, iterating quickly on design details and distributional effects. The Niskanen Center uses PolicyEngine for public‑interest policy analysis and communication. The New York Times also employed the model during its coverage of the 2025 reconciliation bill (the “One Big Beautiful Bill Act”), and although that analysis was not published due to the bill passing more quickly than expected, that work hardened our release and validation processes. With support from the Nuffield Foundation, we enlisted Citizen Codex to conduct a UX research study; the findings are informing contributor onboarding, documentation structure, and the layout of example‑driven “first‑run” experiences.

Since Phase I, collaboration with the National Bureau of Economic Research (NBER) has deepened under a formal memorandum of understanding. Working with Dan Feenberg, we are expanding an automated cross‑model validation harness that compares calculations across hundreds of thousands of scenarios and leverages historical tax‑form archives to ensure comprehensive coverage. Feenberg is also advising our I‑Corps for POSE participation as an external mentor, helping translate decades of TAXSIM operational lessons into our OSE practices.

Philanthropic and research partners are reinforcing adoption. Arnold Ventures has supported capabilities including customizable behavioral responses and state‑specific analyses; their SALT‑AMT interaction work has been used to inform congressional staff, with real‑time iterations replacing static presentations. Open libraries produced through this work have already been adopted by peer groups such as the Yale Budget Lab. Internationally, the UK Cabinet Office engagement is documented in HM Treasury’s Algorithmic Transparency Record, and major UK policy institutions—including the Institute for Fiscal Studies, Resolution Foundation, and Centre for Policy Studies—have begun using PolicyEngine to validate analyses. In parallel, academic users at USC are applying PolicyEngine to study marginal tax rates and benefit cliffs and are exploring bridges to Census workflows (e.g., Supplemental Poverty Measure, distributional national accounts). These relationships illustrate the network effects the OSE is designed to cultivate across academia, government, and civil society.

Complementary research collaborations include work to enhance the Survey of Consumer Finances (SCF) with imputed state identifiers for state‑tax analysis and early exploration of Social Security reform scenarios, coordinated with researchers at Brookings and contacts at federal agencies to document transparency benefits relative to closed models.

## Organization and Governance

### Governance Model

The OSE is run by a small steering group consisting of the PI, co‑PI, and community maintainers. This group sets the roadmap and release train and convenes a public RFC process whenever changes have broad impact. Work happens in lightweight working groups—modeling, data, infrastructure, and documentation—each with a short charter and rotating facilitation so leadership and institutional knowledge grow beyond the core team.

### Licensing

We will use an open‑source license chosen to maximize downstream adoption and remove friction for partners. Policies for data adapters and third‑party dependencies are documented and consistently applied. Depending on the contribution pathway, we will use a Developer Certificate of Origin or contributor agreement that is simple, well understood, and friendly to institutional contributors.

### Decision‑making and Accountability

Roadmaps, issues, design notes, and decisions are public by default, with review thresholds stated in advance so contributors know how to propose and land changes. A clear code of conduct and conflict‑of‑interest policy set expectations for collaboration. Security reports flow through a dedicated private channel with responsible‑disclosure timelines and a security advisory process tied to signed releases.

### Sustainability Structures

For financial operations, we will work through a fiscal sponsor or foundation partner so we can transparently receive funds, publish budget priorities, and report on spending and results. These structures turn good intentions into routines that persist across staff changes and funding cycles.

## Continuous Development Model

### Methodology and Infrastructure

We will practice trunk‑based development with protected branches and continuous integration that runs unit and integration tests, produces deterministic builds, and stores build artifacts for traceability. Reproducible analysis pipelines use pinned environments and explicit provenance so a reviewer or partner can rerun a result and get the same numbers.

### Quality, Security, and Privacy

Quality gates combine coverage targets, static analysis, and dependency scanning. Every release is signed and accompanied by a software bill of materials and checksums for verification. Sensitive changes receive two‑person review and critical paths have named codeowners. No sensitive data is stored in the repository; adapters are designed to run locally or in secure environments, and we document anonymization and pseudonymization patterns for teams that need them.

### Release and Backports

We will ship on a predictable cadence and maintain long‑term‑support branches for adopters who need stability. Security fixes are backported promptly and communicated with clear advisories so downstream systems can patch confidently.

### Modular Packages and Documentation

The modular packages make these practices concrete. L0 delivers layers and penalties with examples and property tests that replace one‑off sparsity code in downstream repositories. MicroImpute provides QRF‑based imputation with benchmarking and a typed API. MicroCalibrate exposes reweighting with loss categories and administrative target loading and integrates L0 sparsity with tuning and holdout robustness. All three come with narrative documentation and examples that shorten the path from first import to first contribution.

### Validation and Interoperability

To help teams migrate with confidence, we will publish a transparent validation harness that compares PolicyEngine to TAXSIM (Feenberg & Coutts 1993) and to peer tools such as PSL Tax‑Calculator, the Atlanta Fed Policy Rules Database scenarios, Yale Budget Lab tools, and BenefitKitchen, as well as to published scores from ITEP, the Tax Foundation, CBO, and JCT. Under a formal collaboration with NBER, we are expanding this harness with large‑scale, versioned scenario packs (hundreds of thousands of cases) and historical tax‑form archives curated by Dan Feenberg so coverage is comprehensive and reproducible. The harness runs in CI and emits signed scorecards and dashboards that track deltas by provision and geography over time. Internationally, we will add scenarios for the UK (IPPR, HMT, OBR) and Canada (SPSDM) to help researchers and agencies cross‑walk results. An independent evaluator will audit assumptions and methods and publish an assessment. These activities build trust and reduce the switching cost for teams migrating from closed or bespoke models, without using Phase II funds for net‑new feature development.

## Risk Analysis / Security Plan

### Threat Model and Risks

We take a conservative view of the supply chain and assume adversaries may attempt dependency compromise, typosquatting, or malicious contributions. We also plan for data‑handling mistakes—such as leakage via logs or build artifacts—and for integrity risks like reproducibility drift or insecure release artifacts. The plan below addresses those risks with concrete, auditable practices.

### Security Plan

Identity and access management is based on least‑privilege roles, two‑factor authentication, scoped secrets, and periodic access reviews. The secure development lifecycle includes code review, static and dynamic analysis, and dependency pinning and scanning. We publish a SBOM for each release and sign artifacts. Incident response uses a private reporting channel, simple SLAs for triage and fixes, and a disclosure workflow aligned with CISA/NSA guidance (CISA & NSA 2022) and OpenSSF best practices.

### Quality, Provenance, and Privacy

Every dataset and ruleset includes provenance, and build artifacts are checksummed so releases are traceable and reproducible. Policy rule changes leave an audit trail that reviewers can follow. We keep code and sensitive data separate by design, offer practical guidance on lawful and ethical data use, and consult IRBs where appropriate when human data is involved.

## Community Building

### Community Strategy

Our community plan is deliberately simple. Discovery happens through the I‑Corps interviews and ongoing user research, and the results are reflected in a public roadmap that is kept current. In a recent UX study conducted by Citizen Codex (funded by the Nuffield Foundation) (Citizen Codex 2025), we combined stakeholder interviews, usability testing and observations, an audit of the current app, affinity mapping, and archetype analysis across policy analysts, economists, researchers, journalists, and novice users. The study surfaced strengths—open code, easy sharing, a non‑programmer UI, and rapid hypothesis testing—as well as pain points around navigation, policy organization and naming, and feedback loops. Those findings are driving concrete changes to information architecture, contributor onboarding, and example‑driven “first‑run” experiences.

Onboarding meets people where they are, with first‑timer issues, clear contributor guides, a mentorship program, office hours, and starter projects that can be completed in a weekend. Recognition is built into releases and docs, with credits and lightweight badges, and we reserve small micro‑grants for documentation and adapters. Support lives on an open forum and chat, with quick responses to issues and pull requests, a monthly community call to keep momentum, and quarterly hackathons to welcome new contributors.

### Design Commitments (from UX Research)

We are turning research into visible changes. We will simplify navigation and policy organization so that common tasks (select a policy, set a timeframe, view distributional and budget results) are one or two clicks away. Policy naming will follow a consistent style guide and include search‑forward aliases (e.g., common nicknames for credits) to reduce confusion. The first‑run experience will offer example‑driven walkthroughs for analysts and non‑programmers, with copy‑to‑notebook snippets for those who prefer code. Feedback loops become explicit: the app will include a “suggest a fix” entry point that opens a pre‑filled issue with context, and the public roadmap will mark items that came from user studies so contributors see how their input moves work forward. To support international adopters (e.g., Canadian government teams), we will prioritize basic localization (bilingual documentation and UI strings) so contributors and users can work in their preferred language.

## Sustainability

### Financial Model

We will blend philanthropic grants, public‑sector contracts that fund well‑scoped enhancements, and community sponsorships. The core will remain open without paywalls; for adopters who prefer a supported option, we may offer hosted services that simplify operations while keeping all code open and releases public. This keeps incentives aligned: the more the ecosystem grows, the more institutions can justify underwriting improvements that benefit everyone.

### Operational Sustainability

Sustainability is primarily an organizational practice, not a funding event. We distribute responsibility across working groups and community maintainers so no single person becomes a bottleneck. We lower onboarding costs through repeatable processes and documentation, and we set concrete targets for new maintainers and external pull requests so we can see progress. Each quarter we allocate time for “keep‑the‑lights‑on” work—security triage, dependency management, and build/release care—so reliability does not depend on heroics.

### Runway and Partners

Our Phase I award (August 2025) funds OSE scoping and participation in I‑Corps for POSE. Arnold Ventures is supporting development of state and congressional district weights in the U.S., building on earlier work in the UK funded by the Nuffield Foundation. Their prior investments also enabled user‑customizable behavioral responses and SALT‑AMT interaction analysis using PolicyEngine’s state encodings—tools used to brief congressional staff—and helped seed open libraries that groups such as the Yale Budget Lab have adopted. The UK Cabinet Office engagement, documented in HM Treasury’s Algorithmic Transparency Record, underscores government appetite for integrating PolicyEngine into policy workflows and provides a concrete model for U.S. public‑sector adoption. Together, these relationships create a practical runway for the OSE to grow while maintaining a disciplined, open process.

### Industry Participation

Commercial adopters help sustain the ecosystem with both engineering and funding. When a tax firm or software vendor needs a new provision or a state‑level nuance, the change is specified, implemented with tests and documentation, and merged upstream so the policy community gains the capability as well. When a market‑research firm or financial institution funds performance or data‑pipeline improvements, we ship them as signed, reproducible releases. We do not grant exclusivity; instead, sponsors receive recognition, predictable delivery, and the benefits of an expanding contributor base. POSE funds remain focused on OSE enablement and operations (governance, security, release engineering, onboarding, evaluation); firm‑specific features are supported by external funds and incorporated when they generalize.

## Evaluation Plan and Metrics

### Outcomes and Metrics

We will measure what matters to an ecosystem: adoption, contribution, quality and security, and calibration performance. Adoption is tracked through named deployments, monthly active users, and simple retention measures that show whether organizations keep the system in use. Contribution is captured by external pull requests and issues, the time it takes to review and merge changes, and the number of active community maintainers. Quality and security are monitored by test coverage and reliability checks, the time to remediate vulnerabilities, and the share of releases that are signed and include a SBOM; reproducibility audits make sure published numbers can be rerun. For calibration work, we report the share of targets within tolerance—for example within ten percent—the loss trajectory over epochs, stability across holdout splits, and, when we exclude targets, why we chose to do so.

### Measurement Plan

We will baseline these metrics at the outset and present them on a public dashboard. Each quarter the steering group reviews the indicators and the evidence from I‑Corps interviews and observed usage, then adjusts the roadmap accordingly. The validation harness runs in CI and publishes signed scorecards; for state and congressional district weighting runs, we retain epoch‑level logs and share summaries of loss reduction and target coverage so partners can judge progress at a glance. The guiding idea is to make our own learning loop visible: when adoption falters, we see it; when reviews slow, we see that too; when calibration and cross‑model agreement improve, the charts make the story legible to reviewers and collaborators.

## Broader Impacts (PAPPG required section)

### Broader Impacts

This project broadens access to evidence‑based policymaking by delivering an auditable, open‑source platform for distributional and budget analysis that researchers, policymakers, journalists, and the public can use and improve. In practice, this means that a legislative office can test a change to a credit and see its incidence; a newsroom can reproduce the analysis behind a headline; and an academic seminar can dig into the assumptions and extend the work. By reducing dependence on proprietary models, we make debates easier to adjudicate on the merits and help good ideas travel further, faster.

Our broader‑impacts activities focus on practical steps that make participation and use easier. Contributor pathways come with mentorship, first‑timer issues, and clear documentation so new maintainers can confidently make their first change. Training and outreach run through workshops, case studies, and partner pilots in academia and government, so the tooling is taught where it will be used. Privacy‑preserving workflows and guidance keep sensitive microdata out of repositories and show analysts how to work safely and lawfully with local systems.

Over the award we expect to see wider adoption of reproducible policy analysis, wider access to high‑quality tools, a steady increase in external contributions and maintainers, and clearer transparency around modeling assumptions and results. We will track adoption, contribution, and quality/security alongside calibration metrics, and we will adjust activities based on evidence from I‑Corps interviews and observed usage, making our learning process as open as our code.

### Illustrative Use Cases

Consider three common settings. A state revenue office is evaluating a refundable credit redesign; an analyst loads a canonical scenario, toggles eligibility parameters, and immediately sees distributional tables by decile, geography, and family type, with footnotes that cite assumptions and links to reproducible notebooks. A newsroom needs to sanity‑check a claim about the budget effect of a late‑breaking amendment; an editor opens a saved scenario, updates the effective date and income ranges, and exports ready‑to‑publish charts, with the methodology available in the references. A benefit navigator wants to estimate the take‑up and cash‑flow effect of a policy change for their clients; a program lead imports a county‑level scenario, runs head‑to‑head comparisons against baseline and published scores, and shares results with a case‑worker team. In each case, the path from question to reviewed answer is short, reproducible, and public—exactly the behavior an OSE should enable.

## Conclusion and Roadmap

Conclusion and Roadmap

PolicyEngine is already used to answer real questions about tax and benefit policy. Phase II funding will not change what the code does so much as how the ecosystem around it works. Our plan focuses on the practices that make an open‑source ecosystem durable: governance that anyone can understand and rely on; release engineering that institutions can trust; contributor pathways that turn users into maintainers; and evaluation that shows, in public, whether adoption and quality are improving.

The immediate steps are straightforward. We complete I‑Corps for POSE in January and publish the adoption‑blockers report and onboarding materials. We formalize governance and security—CODEOWNERS, RFCs, signed releases, and SBOM—so partners have clear assurances. We release stable versions of L0, MicroImpute, and MicroCalibrate with examples that shorten time‑to‑first‑contribution, and we document the calibration pipeline so state and congressional district runs are reproducible. With partners, we run at least two pilots that demonstrate the end‑to‑end path from a policy question to a reviewed, reproducible answer.

From there the work becomes a cadence. We ship on a predictable schedule; we answer issues and reviews quickly; we recruit and support new maintainers; and each quarter we look at the dashboard and change course if the evidence says we should. The product of this process is not just a model but a public, verifiable way of working that others can adopt and extend. That is the point of an ecosystem: it outlasts any single team.

Timing and Decisions

Based on recent POSE cycles, we expect roughly six to eight months from the September deadline to an award start. In practice, that means decisions in February to April 2026 and a start window in March to June 2026. Our January I‑Corps for POSE work feeds directly into an early design‑freeze milestone before the broad rollout. Phase II dollars are strictly for ecosystem enablement and operations; Phase I covers scoping, and we will not duplicate that scope in Phase II.

# Summary

* Sections complete: 0/11

## Sections needing attention

## References Cited

References Cited (PAPPG format to be finalized)

* Louizos, C., Welling, M., & Kingma, D. P. (2018). Learning Sparse Neural Networks through L0 Regularization. International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1712.01312
* Meinshausen, N. (2006). Quantile Regression Forests. Journal of Machine Learning Research, 7, 983–999. http://jmlr.org/papers/v7/meinshausen06a.html
* OpenSSF. Open Source Security Foundation Best Practices Criteria. https://www.bestpractices.dev/en/criteria
* CISA & NSA. (2022). Securing the Software Supply Chain: Recommended Practices for Developers. https://media.defense.gov/2022/Sep/01/2003068942/-1/-1/0/ESF\_SECURING\_THE\_SOFTWARE\_SUPPLY\_CHAIN\_DEVELOPERS.PDF
* Feenberg, D., & Coutts, E. (1993). An Introduction to the TAXSIM Model. Journal of Policy Analysis and Management, 12(1), 189–194. (NBER TAXSIM). https://www.nber.org/research/data/taxsim
* PolicyEngine MicroImpute (Python Package). Quantile regression forest imputation with benchmarking. https://pypi.org/project/microimpute/
* PolicyEngine MicroCalibrate (Python Package). Gradient‑descent reweighting with optional L0 sparsity; tuning and robustness tools. https://pypi.org/project/microcalibrate/
* PolicyEngine L0 (Python Package). PyTorch layers and penalties implementing Hard Concrete gates. https://pypi.org/project/L0-python/
* PolicyEngine US Enhanced CPS Methodology (documentation and validation dashboards). https://policyengine.github.io/policyengine-us-data/

Phase I References (selected)

* Ahmadi, V. (2024). How PolicyEngine UK models behavioural responses. https://policyengine.org/uk/research/behavioural-responses
* Feenberg, D. & Coutts, E. (1993). An Introduction to the TAXSIM Model. JPAM. https://www.jstor.org/stable/3325474
* Ghenis, M. (2024). NEO Philanthropy Awards $200,000 Grant to PolicyEngine. https://policyengine.org/us/research/neo-philanthropy
* Ghenis, M. (2024). Nuffield Foundation awards £251,296 to enhance PolicyEngine UK. https://policyengine.org/us/research/uk-nuffield-grant
* Ghenis, M. & Gross, B. (2024). PolicyEngine Powers Niskanen Center's Analysis. https://policyengine.org/us/research/niskanen-center-analysis
* Ghenis, M., Woodruff, N., DeBacker, J., & Ahmadi, V. (2024). PolicyEngine US Launches Behavioral Responses. https://policyengine.org/us/research/us-behavioral-responses
* Henson, E. (2024). MyFriendBen Receives Grant to Expand Access to Public Benefits. https://www.myfriendben.org/myfriendben-receives-grant-to-expand-access-to-public-benefits/
* Imagine LA (2022). Imagine LA Receives $1M Grant to Develop Social Benefit Navigator. https://www.prnewswire.com/news-releases/imagine-la-receives-1-million-grant-from-conrad-n-hilton-foundation-to-develop-social-benefit-navigator-301674511.html
* Volk, A. (2024). New AI Feature Helps People Understand Their Taxes and Benefits. https://policyengine.org/us/research/us-household-ai
* Woodruff, N. (2022). How machine learning tools make PolicyEngine more accurate. https://policyengine.org/us/research/how-machine-learning-tools-make-policyengine-more-accurate
* Woodruff, N. (2022). How developers can explore the PolicyEngine API. https://policyengine.org/us/research/how-developers-can-explore-the-policyengine-api
* Woodruff, N. (2022). From OpenFisca to PolicyEngine. https://policyengine.org/us/research/from-openfisca-to-policyengine
* Woodruff, N. (2023). Digital Public Goods Alliance recognizes PolicyEngine as a DPG. https://policyengine.org/us/research/digital-public-goods-alliance-recognizes-policyengine-as-a-digital-public-good

Citizen Codex UX Study

* Citizen Codex (2025). Re‑imagining PolicyEngine, v2. Internal UX study (PPTX). materials/phase2/ux/citizen\_codex\_ux.pptx

Validation and Peer Models (for References Cited)

* NBER TAXSIM: https://www.nber.org/research/data/taxsim
* PSL Tax‑Calculator: https://github.com/PSLmodels/Tax-Calculator
* Atlanta Fed Policy Rules Database (PRD): https://www.atlantafed.org/economic-mobility-andresilience/advancing-careers/policy-rules-database
* Yale Budget Lab: https://budgetlab.yale.edu/
* BenefitKitchen: https://www.benefitkitchen.com/
* Institute on Taxation and Economic Policy (ITEP): https://itep.org/
* Tax Foundation Models: https://taxfoundation.org/
* Congressional Budget Office (CBO): https://www.cbo.gov/
* Joint Committee on Taxation (JCT): https://www.jct.gov/
* IPPR (UK): https://www.ippr.org/
* Institute for Fiscal Studies (UK IFS): https://ifs.org.uk/
* Resolution Foundation (UK): https://www.resolutionfoundation.org/
* Centre for Policy Studies (UK): https://cps.org.uk/
* HM Treasury (UK): https://www.gov.uk/government/organisations/hm-treasury
* HM Treasury (2024). HMT Modelling - Policy Engine: Algorithmic Transparency Record. https://www.gov.uk/algorithmic-transparency-records/hmt-modelling-policy-engine
* Nuffield Foundation (2024). Enhancing, localising and democratising tax-benefit policy analysis. https://www.nuffieldfoundation.org/project/enhancing-localising-and-democratising-tax-benefit-policy-analysis
* Office for Budget Responsibility (UK OBR): https://obr.uk/
* SPSDM (Statistics Canada): https://www.statcan.gc.ca/eng/microsimulation/spsdm/spsdm

Letters of Support from PBIF Application (Referenced for OSE Stakeholder Validation)

* Better Government Lab (BGL), Georgetown University. Letter of Collaboration for PolicyEngine Policy Library. August 15, 2025.
* National Bureau of Economic Research (NBER). Collaboration Letter supporting PolicyEngine initiatives. August 15, 2025. Commitment to contribute Dan Feenberg's historical tax form archive.
* Policy Simulation Library (PSL). Letter of Support for PolicyEngine Policy Library. August 16, 2025.
* Prenatal-to-3 Policy Impact Center, Vanderbilt University. Letter of Support for PolicyEngine Policy Library. August 17, 2025.
* University of Southern California (USC). Letter of Support for PolicyEngine PBIF Application. August 2025.
* Urban Institute. Letter of Support for PolicyEngine Policy Library. August 14, 2025. Exploration of subgrant arrangement for policy document contributions.
* Benefit Kitchen. Letter of Support for PolicyEngine PBIF Application. August 15, 2025. Relies on authoritative policy documents for benefit eligibility nationwide.

Note: Per solicitation, URLs appear in References Cited and not in the Project Description body.