European Journal of AI, Computing & Informatics

Article

AI-Assisted Decision Support in Housing Policy Implementation: Distinguishing Deterministic Rules, Manual Review, and Heuristic Suggestion

Jingyuan Huang [1,*]

[1] Independent Researcher, New York, United States
* Correspondence: Jingyuan Huang, Independent Researcher, New York, United States

Abstract: U.S. housing policy directly shapes the housing options available to homeowners and families, but easing zoning rules and broadening legal permission do not, on their own, translate into completed homes. Implementation difficulties surface at two distinct stages: many eligible homeowners never formally apply once they encounter the cost, complexity, and uncertainty of pre-application analysis, and even among those who do apply, a substantial share of permitted projects never reach completion. The principal site of friction is the implementation phase, not the legalization phase—and addressing it requires mechanisms that translate policy rules into parcel-level feasibility judgments that property owners, designers, and local governments can rely on. This article examines how artificial intelligence and related computational methods should be applied to residential buildability assessment. The central question is where AI should sit—and where it should not—within decisions that involve regulatory compliance. The article proposes a three-layer architecture distinguishing rule sets, deterministic calculation, and heuristic suggestion. Matters involving regulatory boundaries—setbacks, lot coverage, building separation—should be handled through deterministic rule sets and calculation; AI may legitimately assist in building and maintaining such systems, but should not perform runtime compliance adjudication. Site conditions that cannot be reliably confirmed from public data should enter manual review. Heuristic methods are appropriate for ranking, placement suggestion, and scenario comparison within the feasible space defined by deterministic rules. Accessory dwelling units (ADUs) serve as the principal scenario for discussion and validation, but the architecture is not specific to ADUs—it applies more broadly to small-scale residential development, adaptive reuse, accessory-structure approval, and land-use change review. The article is presented as a normative contribution to the literatures on AI in public-rule systems, housing policy implementation, and digital governance of the built environment.

Keywords: Housing policy implementation; Accessory dwelling unit; Parcel-level feasibility; Decision support framework; Deterministic rules; Heuristic suggestion; Built environment; Digital governance

Received: 20 April 2026
Revised: 31 May 2026
Accepted: 11 June 2026
Published: 18 June 2026

1. Introduction

Housing policy in the United States faces challenges related to insufficient supply, complex land-use regulation, and uneven local implementation. Although many states have adopted zoning reform and ADU policies to promote supply, easing the rules does not automatically translate into completed homes. Implementation difficulties surface at two distinct stages: many homeowners with eligible parcels never formally apply once they encounter the cost, complexity, and uncertainty of the pre-application process, and even among those who do apply, a substantial share never reach completion. Statewide data from California illustrate the second-stage attrition: between 2018 and 2019, ADU permits issued rose from roughly 6,000 to more than 15,000, but completions over the same period rose only from about 2,000 to nearly 7,000 [1]—meaning fewer than half of permitted projects reached the built stage during this window. Earlier scholarly work on backyard housing identified the same pattern earlier: regulatory complexity, site-specific feasibility uncertainty, and homeowner navigation costs prevent latent demand from converting into built units, even where local policy formally permits them [2]. The conclusion across these studies is consistent: legal authorization is not equivalent to practical execution. The implementation phase—not the legalization phase—is where most of the loss occurs.

For homeowners and small-scale developers, the practical difficulty is often not whether a city allows ADUs in general, but whether a particular parcel has the conditions required for implementation. Whether a parcel has development potential depends on lot area, existing structures, setback requirements, lot coverage, height limits, alley relationships, environmental critical areas, historic overlays, and permit records—information distributed across GIS datasets, zoning ordinances, building codes, and permit databases.

The combination of GIS, open government data, and recent advances in artificial intelligence has created a foundation for digitizing this kind of analysis. The central argument of this article is that the key obstacle in housing policy implementation is the absence of a mechanism that translates policy rules, parcel data, and permitting constraints into explainable, reviewable, parcel-level feasibility judgments. The value of AI here lies not in maximizing automation, but in establishing appropriate computational boundaries according to task type. As Veale and Brass argue, deploying algorithmic systems in public administration raises distinct concerns about accountability, contestability, and administrative discretion—concerns that apply directly to housing implementation [3]. The specific question this article develops—how AI should be applied within such decision-support infrastructure—is approached through the three-layer architecture distinguishing rule sets, deterministic calculation, and heuristic suggestion (Section 3.3). Earlier work by the author has examined ADU implementation barriers and decision-support frameworks for housing supply more broadly; the present article builds on that background but does not retread it [4, 5].

ADUs serve as the principal scenario examined throughout this article because they exhibit several properties well-suited to systematic validation: rules that are largely computable, parcel-level analysis units that are clearly defined, and regulatory frameworks that vary substantially across jurisdictions. The framework itself, however, is not specific to ADUs. The same architecture applies to other built-environment regulatory tasks where law must be translated into parcel-level judgment—small-scale residential development, adaptive reuse, accessory-structure approval, historic-building modification, and land-use change review. What changes across these scenarios is the content of the rule set and the boundary of manual review; what stays the same is the architectural separation between deterministic computation, heuristic suggestion, and human judgment.

The remainder of the article is organized as follows. Section 2 examines the parcel-level translation problem. Section 3 develops the phased implementation framework, with the three-layer architecture as its core. Section 4 discusses representative application scenarios. Section 5 examines governance boundaries and limitations. Section 6 concludes.

2. The Parcel-Level Translation Problem in Housing Policy Implementation

2.1. The Translation Problem from a System-Architecture Perspective

The fundamental challenge of housing policy implementation at the parcel level—that legal authorization does not equate to practical buildability—has been examined from a policy-execution perspective in Huang and from a decision-support perspective in Huang [4, 5]. The present article does not re-establish these arguments; it takes them as foundational and instead focuses on the question they leave open: given that an intermediate translation layer is required, how should that layer be computationally architected, and where in that architecture is artificial intelligence appropriately deployed?

This question is theoretically substantive rather than merely technical. Different categories of regulatory tasks carry different epistemic structures: some involve deterministic rule application against quantifiable parcel attributes; others involve site facts that public data cannot reliably confirm; still others involve open-ended ranking among candidate options where reasonable people can differ. A computational architecture that treats these task categories identically will fail in characteristic ways. The architecture this article proposes is designed to match computational method to epistemic task type.

Moudon and Hubner, in their foundational work on parcel-level GIS for land supply monitoring, identified the parcel as the appropriate unit of analysis. However, their framework predates AI-assisted analysis [6]. The question of *how* AI should fit within parcel-level decision-support systems remains under-developed in the recent computational urban-systems literature, which has focused on inferring zoning categories from incomplete data or modeling land-cover change rather than on the design principles for tools supporting implementation decisions [7, 8].

2.2. Public Built-Environment Data as a Technical Basis for Intelligent Assistance

Housing policy implementation in the United States rests on a relatively rich public-data foundation. Parcel boundaries, zoning layers, building footprints, cadastral records, permit records, environmental constraints, and public-service distribution are available in many cities through open-data platforms or administrative databases. These data provide the foundation for AI-assisted parcel-level analysis.

However, the existence of data does not mean that data directly support decision-making. Sources vary in format, update cycle, spatial alignment, and interpretive standard. A usable digital tool must clean, match, and standardize these scattered data, and convert them into computable variables. AI-assisted housing policy implementation is therefore not detached technology, but a process of rule integration and decision support built on existing public data systems.

2.3. Data-Source Hierarchy and Rule-Source Binding

A parcel-level housing feasibility tool should establish a data-source hierarchy: official primary data first, followed by authoritative aggregated data, then open data and geometric inference, and finally user-confirmed information. The hierarchy requires the system to indicate the source and confidence level of every output. Tree locations, accessory structures, pools, informal access routes, and other site conditions that may not be stably available from public data should enter a review process rather than be inferred.

Rule parameters, in turn, should be bound to their regulatory sources. Setback distances, lot-coverage limits, floor-area caps, height limits, parking exemptions, and ADU counts should not enter the system as isolated numbers; they should be associated with specific code provisions, conditions of applicability, and update dates. Rule-source binding improves traceability and reduces the risk of rule misuse or outdated interpretation—addressing one of the central concerns Veale and Brass raise about algorithmic systems in public administration: the difficulty of contesting decisions whose basis is opaque [3].

3. A Phased Implementation Framework

3.1. Multi-Source Parcel Data Integration

The first phase of digital-tool intervention in housing policy implementation is multi-source data integration. In parcel-level analysis, the parcel should be treated as the minimum unit of analysis. The system must integrate parcel boundaries, lot area, existing building footprints, zoning classification, road and alley relationships, environmental constraints, permit records, and public-service conditions around each parcel.

Technically, a unified data foundation should first be established to spatially match GIS layers, cadastral information, building footprints, zoning data, environmental constraints, and permit records. Data fields must be standardized so that they can enter rule calculation and feasibility scoring. A data-source hierarchy should be established: official data first, authoritative aggregated data second, and open data, geometric inference, or user confirmation as supplements.

The key to multi-source data integration is not the simple accumulation of data, but the creation of an input structure for policy implementation. Only when parcel data, rule data, and site uncertainty are processed separately can the system output have explanatory value.

3.2. Parcel-Level Feasibility Screening and the Scoring Layer

Once data integration is in place, the system can perform parcel-level feasibility screening: structured aggregation of multiple regulatory variables against a parcel record, producing a ranked or categorized output. Three properties of this task shape the design.

First, the variables involved—lot area, zoning, building footprints, setbacks, lot coverage, environmental constraints, permit records—are heterogeneous in their epistemic status: some are deterministically extractable from official data, others inferable from imagery with characteristic error rates, still others obtainable only from owner input. A system treating all as equally reliable will produce confidence intervals it cannot defend.

Second, the output is decision-support information, not regulatory determination. The design objective is not maximum predictive accuracy but maximum interpretability of the screening rationale. A multi-indicator scoring approach—with each indicator's contribution exposed to the user—satisfies this objective; black-box machine learning does not. This approach has methodological lineage in GIS-based multi-criteria decision analysis [9, 10]. The weighted linear combination (WLC) form expresses the comprehensive feasibility score for parcel i as:

$$S_i = \Sigma_{j=1}^{n} w_j \cdot x_{ij}$$

where S_i is the feasibility score of parcel i, $x_{(ij)}$ is its standardized value on indicator j, w_j is the weight of indicator j, and n is the number of indicators. The auditability of WLC—each indicator's contribution is mathematically explicit—satisfies the requirement that screening rationale remain interpretable to users, reviewers, and downstream professionals.

Third, screening runs at scale: a system applied to a city's residential parcel inventory must produce outputs for thousands of parcels. This constrains the design toward deterministic computation while simultaneously requiring the system to flag parcels that warrant exit into manual review. Outputs such as "preliminarily feasible," "requires further review," or "obvious constraints present" should be understood as pre-application decision support, not final permitting conclusions.

3.3. The Three-Layer Architecture: Rule Sets, Deterministic Calculation, and Heuristic Suggestion

The central theoretical contribution of this article is a three-layer architecture distinguishing rule sets, deterministic calculation, and heuristic suggestion. These three layers correspond to three different epistemic categories of decision support, each with different computational requirements and different governance implications.

The rule set is the foundation. It is formed by structuring zoning ordinances, building controls, ADU rules, setback requirements, lot coverage, floor-area limits, building height, building separation, and special-district restrictions. Its function is to convert regulatory text into computable parameters and constraints. The rule set does not produce judgments; it provides the structured representation of law against which judgments can be made.

On top of the rule set, the system executes deterministic calculation. Any matter that can be calculated from official data, parcel geometry, and clear rules should be handled through deterministic logic, producing the same output for the same input. Setback distances, building separation, lot coverage, floor-area controls, and review triggers should be jointly determined by the rule set and parcel data. Items that cannot be directly determined from reliable data—alley accessibility, tree locations, accessory structures, informal access routes—should be marked for manual review rather than guessed.

A critical clarification: *deterministic* does not mean *AI-free*. This layer can and often does involve substantial AI assistance—AI may help translate ordinance text into rule modules, generate test cases, structure the data-source hierarchy, or even author the computational logic itself. What makes the layer deterministic is not the absence of AI in its construction, but the *property of its output*: given the same inputs, the system produces the same outputs, and every step is traceable to its rule source. AI is appropriate as an authoring tool for building and maintaining deterministic systems; what is inappropriate is using AI as a runtime adjudicator that replaces deterministic computation with real-time probabilistic inference for compliance-relevant outputs.

A useful way to frame this distinction is in the language of contemporary AI system design, which separates *workflows*—deterministic pipelines where every step is explicitly defined and the execution path is predictable—from *AI agents*, which receive a high-level goal and decide autonomously which tools to invoke and in what order. Agents offer flexibility on open-ended tasks, but the same goal can produce different execution patterns in different sessions. Compliance-relevant judgments cannot tolerate that flexibility: what homeowners and reviewers need is the assurance that the same parcel, evaluated against the same rules, returns the same answer regardless of who asks or when. The three-layer architecture proposed here therefore operates the compliance layer in workflow form (AI may participate in its construction, but the runtime is deterministic); agentic or heuristic flexibility is permitted only after compliance boundaries have been established by deterministic computation.

Heuristic methods follow once deterministic boundaries are established. After the system has identified candidate spaces that satisfy setbacks, building separation, and lot coverage, heuristic indicators can rank candidate locations or provide preliminary placement suggestions—access convenience, spatial continuity, relationship to the existing dwelling, impact on rear-yard use, construction accessibility, privacy. These outputs are framed as suggestions, not compliance judgments.

3.3.1. Deterministic and Non-Deterministic Heuristics

A practical distinction often overlooked: heuristic methods can be either deterministic or non-deterministic, and the difference matters more than whether the system is labeled "AI." A *deterministic heuristic* uses fixed rules, priorities, and scoring logic; given the same inputs, it produces the same output every time. The author's own ADU setback rule set—comprising more than fifty conditional rules covering interior, corner, and through lots with various exception conditions—is an example: heuristic in that it encodes designed conditional logic rather than continuous optimization, but also deterministic, because identical inputs yield identical outputs.

A *non-deterministic heuristic*, by contrast, contains randomness, dynamic adaptation, free generative behavior, or context-sensitive interpretation; the output may vary across runs even with identical inputs. AI systems that freely generate placement suggestions or interpret user prompts differently each session fall into this category. In residential design and permitting, non-deterministic behavior in compliance-relevant decisions undermines user trust—homeowners rightly ask why the same lot returns different feasibility outputs on different days.

The choice between them per task can be expressed as:

layer(t) = {
rule set / deterministic, if t ∈ T_(compliance)
deterministic heuristic, if t ∈ T_(ranking)
non-deterministic AI, if t ∈ T_(explanation)
manual review, if t ∈ T_(unverifiable)
}

where T_(compliance) contains tasks with regulatory consequences, T_(ranking) contains tasks of ordering compliant options by user preference, T_(explanation) contains open-ended natural-language tasks, and T_(unverifiable) contains tasks whose underlying facts cannot be confirmed from public data. The principle: compliance judgments must be deterministic—whether direct rules or deterministic heuristics—while explanation and summarization can tolerate non-deterministic AI. This distinction also maps onto Veale and Brass's macro/meso/street-level framework: the architecture operates at the street level while preserving the meso-level discretion that compliance review requires [3].

3.3.2. Three-Layer Integration in Practice

The architecture's practical instantiation is a three-layer integration: Layer 1 is deterministic rule check (zoning, setback, lot-type classification, pass/fail/warning), with stable, auditable, reproducible output. Layer 2 is AI-assisted recommendation within the feasible space defined by Layer 1: AI ranks candidate options, explains trade-offs in natural language, and personalizes against user preferences—but does not alter the feasible space. Layer 3 is human review and user preference: the user selects what they value (sunlight, main-house relationship, noise minimization, usable yard preservation), and the system adjusts Layer 2 outputs accordingly. Final compliance and design decisions remain with qualified professionals and the user.

The principle is concise: the system does not use AI to replace deterministic zoning checks. Deterministic rule sets define the feasible design space; AI-assisted heuristics help rank, explain, and personalize feasible options.

Responsible AI-assisted housing policy tools should therefore maximize the fit between task and method rather than maximize automation for its own sake. This principle allows AI to contribute efficiency in complex information processing and user interaction, while avoiding false confidence in public-rule systems.

3.4. Buildable-Envelope and Spatial-Constraint Analysis

After data integration and rule calculation, the system can generate a buildable envelope—a preliminary spatial boundary formed by parcel boundaries, setbacks, existing building locations, building separation, lot coverage, height limits, and environmental constraints. In ADU scenarios, the system generates an inset area from parcel boundaries according to setback requirements, then subtracts the existing building and its required separation to form a possible placement area.

Buildable-envelope analysis should be based on actual parcel polygon boundaries rather than rectangular bounding boxes. For irregular, rotated, or flag-shaped parcels, bounding-box simplification may produce setback errors. The buildable envelope is not a building design or permit drawing—it is a visualization of regulatory constraints and parcel conditions, helping users understand why some areas have development potential while others are restricted.

3.5. Periodic Regulatory Monitoring and Rule-Set Maintenance

A frequently overlooked but operationally critical question is how a deployed digital tool maintains alignment with evolving regulations over time. Zoning ordinances, building codes, and ADU rules are not static. Seattle Ordinance 127376, effective January 21, 2026, illustrates that even well-established frameworks continue to evolve through periodic amendment. A digital tool that was correct at deployment can silently become incorrect through legislative change.

This challenge admits an AI-appropriate solution consistent with the three-layer architecture. Periodic regulatory monitoring—a recognized practice in compliance-focused AI applications—can be applied as a scheduled background task. On a defined cadence (quarterly for stable jurisdictions; monthly during legislative cycles), the system can automatically:

- Crawl official municipal and state legislative repositories for changes to relevant code sections;
- Compare current text against the version locked at the previous rule encoding;
- Generate a structured change report identifying affected rule modules;
- Alert maintainers to provisions requiring re-locking or human review.

This monitoring task is well-suited to AI assistance because it is descriptive and informational rather than adjudicative. The AI does not interpret the legal change or modify the rule set autonomously; it identifies that a change has occurred and surfaces it for human attention. The actual re-encoding remains with human judgment—preserving the accountability principle of Section 3.3. For multi-jurisdiction deployment, AI-assisted monitoring scales across jurisdictions in a way that manual tracking does not.

4. Representative Application Scenarios

4.1. ADU Parcel-Level Feasibility Screening

ADU policies provide a representative application scenario. Such systems integrate parcel boundaries, zoning classifications, existing building footprints, alley relationships, setback requirements, lot coverage, and permit records to conduct preliminary feasibility screening. The system does not decide whether a project will ultimately be approved—it identifies whether a basis exists for further analysis, outputting parcel lists, map markers, buildable envelopes, rule-constraint notes, and items requiring manual review.

The screening is particularly consequential for detached ADUs (DADUs), where a new structure must fit alongside the existing dwelling within the parcel boundary. Setback offsets, building separation, lot-coverage caps, and tree-protection constraints jointly define a buildable envelope; the deterministic-rule layer computes this envelope, while the heuristic-recommendation layer ranks candidate placements within it. For attached or interior-conversion ADUs, the same architectural pattern applies, but the buildable-envelope computation is replaced by interior-feasibility checks against unit-size minimums and means-of-egress requirements.

At the placement-suggestion level, the system may provide heuristic ranking within compliant candidate spaces—comparing locations by sunlight exposure, relationship to the main dwelling (privacy versus circulation convenience), noise exposure, and preservation of usable yard area. This ranking is framed as preliminary design suggestion, not regulatory conclusion: the rule set determines boundaries, deterministic calculation identifies the feasible range, and the AI-assisted recommendation layer helps users navigate trade-offs among feasible options according to stated preferences. Per the architecture in Section 3.3, the ranking layer should be implemented as a deterministic heuristic—same inputs, same outputs—even when AI is involved in user-preference elicitation or natural-language explanation. The literature on ADU implementation has documented that the predevelopment feasibility-analysis phase is among the most-frequently-cited points of friction for homeowners [11, 12].

4.2. Project Execution Tracking: Hard and Soft Dependencies

The three-layer architecture extends naturally beyond initial feasibility screening into project execution. ADU implementation involves a coordination problem with both hard and soft dependencies among permitting, design, financing, transport, installation, inspection, and utility connection.

Hard dependencies are deterministic: a permit must be approved before formal construction begins; a foundation must be complete before a prefabricated structure is delivered; a utility connection must be confirmed before certain inspections; a passing inspection must precede the next construction phase. These constraints encode as deterministic rules: if A is incomplete, B cannot begin; if A is delayed by Δt, downstream tasks B, C, and D inherit a delay of at least Δt when they sit on the critical path. The total project duration is bounded below by the longest chain of hard dependencies—the standard critical-path result:

$$T_{project} \geq \max_{p \in P} \sum_{i \in p} t_i$$

where P is the set of all paths through the dependency graph and t_i is the duration of task i.

Soft dependencies are coordination patterns that are not strictly enforced by sequencing rules but that materially affect efficiency. Transport scheduling, crane booking, installation crew availability, and inspector booking form a coordination cluster: even though no single deterministic rule requires them to move together, delays in one substantially increase the cost of the others if they have already been scheduled. AI-assisted heuristics can identify such clusters from project task descriptions and email correspondence, suggest parallelization opportunities (site preparation can proceed during permit review), and flag risk concentrations (a delivery, a crane booking, and a crew mobilization scheduled the same week should be treated as a single coordination unit).

The architectural principle is the same as in Section 3.3: deterministic rules define hard dependencies; AI-assisted heuristics identify soft dependencies, coordination risks, and parallelization opportunities; humans make the final scheduling and resource decisions. AI's value is not in deciding the project schedule, but in identifying relationships among tasks, surfacing risks, explaining downstream impacts of delays, and proposing more efficient coordination patterns.

4.3. Process Feedback and Bottleneck Analysis

At a broader level, the framework can support feedback management of the implementation process itself. Housing policy execution involves permit application volume, review cycles, correction rounds, rule conflicts, and review-resource allocation. Process management can integrate parcel screening, application submission, review status, and approval time into a feedback system. The purpose is not to replace permitting agencies, but to identify bottlenecks—if many applications are repeatedly delayed by the same type of rule interpretation or missing documentation, policy departments can adjust guidance, improve procedures, or strengthen pre-application support.

The load of a permitting system can be analyzed using established results from queuing theory: average system load is the product of average arrival rate and average system residence time. If the application arrival rate increases while average time in the system also lengthens, pending cases accumulate. AI-assisted systems can help classify application status and sources of delay, enabling earlier identification of implementation bottlenecks.

4.4. Framework Portability Beyond ADUs

The three-layer architecture does not depend on ADUs as the specific subject matter. It depends on a more general judgment: whether a task requires deterministic output, whether its underlying data can be reliably confirmed from public sources, and whether it engages compliance boundaries. This judgment transfers to other built-environment regulatory contexts:

Small-scale residential development. Lot-level zoning compliance, setback verification, density caps, and pre-application risk identification map directly onto the framework. The rule-set content changes; the architecture does not.

Adaptive reuse of existing structures. Conversion of garages, basements, or accessory structures into habitable space involves rules that are largely computable (egress requirements, ceiling-height minimums, square-footage standards), combined with site conditions that often require manual review (existing structural condition, prior code-compliance history).

Historic-building modification. Quantifiable constraints (façade alteration percentages, height envelopes within historic districts) belong in the deterministic layer; judgment about "impact on historic character" remains with manual professional review.

Land-use change review. Triggering conditions for use variances, neighborhood notification requirements, and compatibility-of-use checks decompose along the same lines—deterministic where rules are clear, manual review where discretionary judgment is required.

Accessory-structure approval (garages, sheds, fences). Area, setback, and height calculations are deterministic; site conditions enter manual review.

In each case, what must be redone is the rule-set content (translating the relevant body of law into structured constraints) and the boundary of manual review (identifying what cannot be reliably confirmed from public data). What carries over is the architecture: the principled separation between rule sets, deterministic computation, manual review, and heuristic suggestion. The framework offers a methodology for systematizing law for digital implementation, not a domain-specific solution for ADUs.

5. Governance Boundaries, Responsibility Allocation, and Limitations

AI-assisted housing policy implementation must define governance boundaries clearly. Residential development involves property rights, public safety, building codes, environmental protection, and administrative approval. No digital framework should replace formal permitting procedures or the professional judgment of architects, engineers, surveyors, attorneys, or local government reviewers.

System outputs should be defined as pre-application assistance rather than final legal conclusions. Rule calculations should preserve sources, assumptions, and conditions of applicability so users can understand the basis of each result. For incomplete data or conditions that cannot be reliably determined, the system should mark uncertainty rather than produce a seemingly certain answer. Frameworks must also include version control and rule-update mechanisms to prevent outdated rules from continuing to shape decisions—a function the periodic monitoring discussed in Section 3.5 directly supports.

These governance principles connect to broader debates in algorithmic regulation. Veale and Brass argue that algorithmic systems in public administration raise distinct concerns about accountability, contestability, and the preservation of administrative discretion [3]. The three-layer architecture proposed in this article addresses those concerns: rule sets make the regulatory basis of decisions auditable; deterministic calculation makes the same input produce the same output (a foundational requirement of contestability); and the explicit separation of heuristic suggestion from deterministic boundary preserves the meso-level discretion that algorithmic systems should not preempt.

The framework has two principal limitations. First, this article establishes an architectural framework rather than reporting deployment outcomes. Its practical effectiveness should be further evaluated through real-world implementation, user studies, and cross-jurisdictional testing. Second, the framework focuses on parcel-level feasibility and implementation support rather than the broader political economy of housing production. Questions of affordability, financing, construction cost, land-market behavior, and distributional equity remain essential to housing outcomes but fall outside the scope of this article.

6. Conclusion

The value of artificial intelligence in housing policy implementation lies not in replacing professional design or permitting authority, but in constructing transparent, traceable, parcel-level decision-support infrastructure. The article's central contribution is a three-layer architecture distinguishing rule sets, deterministic calculation, and heuristic suggestion—each layer matched to a category of regulatory task by the test of whether deterministic or non-deterministic behavior is appropriate. A critical clarification is that *deterministic* refers to runtime output behavior, not to the absence of AI in system construction; AI may legitimately assist in building and maintaining deterministic compliance systems even when it should not adjudicate compliance at runtime.

Future research should examine cross-jurisdictional rule differences, manual-review mechanisms, and user-responsibility boundaries. Empirical validation of the framework through deployed systems will be necessary to evaluate practical effectiveness across jurisdictional contexts. With clear governance boundaries and careful empirical evaluation, artificial intelligence can move from a narrow design-automation tool toward digital infrastructure that supports housing policy implementation and built-environment governance.

References

1. K. Chapple, A. Lieberworth, D. Ganetsos, E. Valchuis, A. Kwang, and R. Schten, *ADUs in California: A Revolution in Progress*. Berkeley, CA: Terner Center for Housing Innovation and Center for Community Innovation, University of California, Berkeley, 2020.
2. J. Wegmann and K. Chapple, "Hidden density in single-family neighborhoods: Backyard cottages as an equitable smart growth strategy," *Journal of Urbanism: International Research on Placemaking and Urban Sustainability*, vol. 7, no. 3, pp. 307–329, 2014.
3. M. Veale and I. Brass, "Administration by algorithm? Public management meets public sector machine learning," in *Algorithmic Regulation*, K. Yeung and M. Lodge, Eds. Oxford, UK: Oxford University Press, 2019, pp. 121–149.
4. J. Huang, "Accessory dwelling units as a policy execution challenge: Feasibility, regulatory risk, and early-stage decision modeling," *Global Journal of Science & Innovation*, Vol. 3, No. 1, pp. 1–8, 2026.
5. J. Huang, "From policy authorization to practical execution: A decision-support framework for implementing housing supply strategies in the United States," *Strategic Management Insights*, vol. 3, no. 1, pp. 24–31, 2026.
6. A. V. Moudon and M. Hubner, Eds., *Monitoring Land Supply with Geographic Information Systems: Theory, Practice, and Parcel-Based Approaches*. New York, NY: John Wiley & Sons, 2000.
7. M. A. Lawrimore, G. M. Sanchez, C. Cothron, M. G. Tulbure, T. K. BenDor, and R. K. Meentemeyer, "Creating spatially complete zoning maps using machine learning," *Computers, Environment and Urban Systems*, vol. 112, art. no. 102157, 2024.
8. Y. Yao, Y. Jiang, Z. Sun, et al., "Applicability and sensitivity analysis of vector cellular automata model for land cover change," *Computers, Environment and Urban Systems*, vol. 109, art. no. 102090, 2024.
9. C.-L. Hwang and K. Yoon, *Multiple Attribute Decision Making: Methods and Applications—A State-of-the-Art Survey*, Lecture Notes in Economics and Mathematical Systems, vol. 186. Berlin, Germany: Springer-Verlag, 1981.
10. J. Malczewski, "GIS-based multicriteria decision analysis: A survey of the literature," *International Journal of Geographical Information Science*, vol. 20, no. 7, pp. 703–726, 2006.
11. J. Greenberg, H. Phalen, K. Chapple, D. Garcia, and M. Alameldin, *ADUs for All: Breaking Down Barriers to Racial and Economic Equity in Accessory Dwelling Unit Construction. Berkeley*, CA: Terner Center for Housing Innovation and Center for Community Innovation, University of California, Berkeley, 2022.
12. K. Chapple, D. Ganetsos, and E. Lopez, *Implementing the Backyard Revolution: Perspectives of California's ADU Owners*. Berkeley, CA: Center for Community Innovation, University of California, Berkeley, 2021.