

## **Arnulfo Gonzalez**

## **Emalie Webb**

## **Kyle Frese**

## **James Wilson**

## **Danny Newton**

## **Proof of Concept**

### **Solution Overview**

#### **Core Functionality**

CasaLingua simplifies complex housing-related documents into accessible language across multiple languages and literacy levels, supporting users in understanding their rights and responsibilities. We want to make the documentation digestible for the applicant, but also provide a way for the jurisdictional staff to communicate to the applicant with all the respect they deserve. The language is more accessible and layman, but maintains a respectful tone - and importantly for the staff it remains legally compliant and policy prescriptive.

#### **Unique Technical Approach**

CasaLingua's technical foundation rests on a self-contained simplification AI using local LLaMA2/Mistral models to convert legal housing documents into simplified yet legally compliant language. This simplification engine connects to comprehensive rules and policy databases through components including SimplificationLLM, RuleEngine, LegalCheck, TermDB, and PolicyDB. The system's multilingual capabilities are powered by NLLB or M2M100 models with dedicated cultural adaptation and quality checking systems to ensure accurate, contextually appropriate translations. Furthermore, Casalingua will leverage fine-tuned simplification models (mT5/BART) and voice processing (Whisper) in a modular, scalable architecture.

The technical architecture prioritizes human oversight through a robust review and approval workflow with specialized interfaces for document comparison and collaborative editing. Once approved, documents flow through content formatting and version control systems before being published via API and delivered through a CDN. The platform incorporates extensive analytics tracking user behavior and system performance, while continuous improvement is facilitated through a sophisticated training pipeline with model registry and specialized knowledge bases. This architecture ensures both technical sophistication and human-centered design in delivering accessible housing documentation.

#### **User Interaction Flow**

1. User uploads or pastes a housing document.
2. AI classifies and segments content.
3. Simplified output is generated and displayed.
4. Optional voice readout or translation is offered.
5. User provides feedback to improve model accuracy.

### **Technical Specifications**

#### **Key Algorithmic Innovations**

* Fine-tuned multilingual text simplification models
* Context-aware segmentation of legal documents
* Adaptive reading level generation
* Multimodal input/output pipeline [ Handles various input formats ]

#### **Technology Stack**

See **Technical Stack Summary** section above.

#### **Implementation Approach**

* Build modular FastAPI backend to support inference and user sessions.
* Use LLaMA2/Mistral via llama.cpp for efficient on-device inference.
* Frontend integrates React and Tailwind with secure API calls.
* Store anonymized user interactions in Postgres/Redis and MinIO for auditability and performance.

### **Feasibility Assessment**

#### **Technical Viability**

We as Casalingua will use our model and its ability to learn general language representations from massive multilingual datasets. This pre-trained knowledge can then be effectively transferred and fine-tuned for specific downstream tasks with significantly less language-specific data.In doing so, saving significant time and computational resources that would otherwise be required to train such large models from scratch.This being done, leading to continuous improvements in model architectures, training techniques, and evaluation methodologies. This ensures ongoing advancements and readily available knowledge. Transformer models are generally scalable and can be adapted to handle different languages and data volumes, providing flexibility for future expansion of the project. Ollama (for local model running), Whisper (for multilingual speech-to-text), and NLLB (for multilingual translation) in significantly reducing infrastructure costs by providing readily available and often free alternatives to proprietary solutions.

Explicitly state that your architecture will utilize modular APIs to ensure seamless and scalable integration with other platforms like Bloom, allowing for future collaboration and expansion of capabilities without significant architectural overhauls.

* Proven transformer models exist for multilingual NLP.
* Open-source tools (Ollama, Whisper, NLLB) reduce infrastructure cost.
* Modular APIs ensure scalable integration with Bloom and other platforms.

#### **Potential Challenges**

Despite the power of multilingual models, Casalingua anticipates several potential challenges. Firstly, **simplifying complex legal text without altering its precise legal meaning** requires careful consideration and specialized techniques beyond standard NLP. Secondly, **ensuring our language models remain unbiased across diverse dialects and cultural contexts** is paramount, especially given the inherent nuances and potential for misinterpretation in legal language across different cultures. Finally, achieving **real-time performance under resource constraints** will be a significant hurdle, particularly with computationally intensive transformer models processing complex legal documents. To mitigate these challenges, Casalingua will employ effective fine-tuning with high-quality, relevant multilingual legal data, develop specific strategies to address linguistic and cultural biases, and explore optimization techniques to ensure efficient real-time processing within our operational limitations.

* Simplifying legal text without altering legal meaning.
* Ensuring language models remain unbiased across dialects and cultures.
* Real-time performance under resource constraints.

#### **Mitigation Strategies**

To proactively address the potential challenges identified, Casalingua will implement a multi-faceted approach centered around robust mitigation strategies. Firstly, to ensure the accurate simplification of legal text without altering its core meaning, we will **integrate legal advisors directly into our model validation loop**. These experts will review the model's output, providing critical feedback on the preservation of legal intent and identifying any potential misinterpretations or inaccuracies. Secondly, to guarantee our language models remain unbiased and culturally sensitive, we will **build a continuous feedback system involving a diverse group of test users** representing various linguistic backgrounds and cultural perspectives. Their ongoing input will be crucial in identifying and rectifying any unintended biases or culturally inappropriate outputs. Finally, to overcome the challenge of achieving real-time performance under resource constraints, we will **optimize our models through techniques such as quantization to reduce model size and computational demands, and leverage GPU acceleration for faster processing**, ensuring efficient and timely delivery of simplified legal information.

* Include legal advisors in model validation loop.
* Build continuous feedback system with diverse test users.
* Optimize models using quantization and GPU acceleration.
* Handling specific CPU/GPU and cost resources for local fine-tuning.

Casalingua will handle preprocessing to the model offline and will measure the impact of the model running, if there is a target computational budget we will work with you to meet your compliance needs and cost. API computational cost will be the main concern to address, as the training aspect will not be a computational impact to the cost.

CasaLingua pipeline using **TinyLlama (1.1B Chat)** + Transformers **can run entirely offline** — but the **performance and feasibility depend on your system setup**.

**Model: TinyLlama/TinyLlama-1.1B-Chat-v1.0**

**Model Size**

• ~1.1B parameters (≅ 2.5–4 GB in RAM depending on precision)

• ~2.3 GB download via Hugging Face in float16

• 4-bit quantized versions require ~1.2 GB VRAM

**Recommended System Requirements**

**Component**

**Minimum (for CPU)**

**Recommended (for GPU)**

**CPU**

4-core (Intel i5/Ryzen 3)

8-core+ (M1/M2/Intel i7/Ryzen 7)

**RAM**

8 GB (16 GB better)

16 GB (for background multitasking)

**GPU**

Not required

NVIDIA 6 GB VRAM+ or M1/M2/M3 Mac

**Disk**

10 GB free (models, logs, cache)

SSD preferred

**Inference Speed**

~30–90 sec / run on CPU

~2–5 sec / run on GPU