

**Team Members:** Katya Tipps, Evan Mullins, Isabella Bodea, Rhyan Parrish

**Statement of Intent:**

Whether due to changes in location or unexpected complications concerning multiple specialties, patients often move between primary care providers. This process of remembering, repeating, and coordinating medical history between providers leads to fragmented care, wasted time, and emotional fatigue.

For providers, a great deal of patient information is lost in this “telephone” game from provider to patient to provider, often roughly piecing together the patient's medical history through word of mouth and cumbersome medical transcriptions. Oftentimes, these conversations over hazy backgrounds are further worsened by providers splitting time between transcribing information and engaging with the patient. Ultimately, this divided experience results in patients leaving their provider's office feeling disconnected, frustrated, and unfulfilled, leaving roughly 30% of patients switching providers<sup>1</sup>.

**Target Users:**

- Healthcare Professionals
- Patients needing repetitive/multipurpose care (elderly, chronic illnesses, extraneous circumstances)

**Solution: Medlog (AI-Powered Clinical Summarization Portal)**

A web-based service that accumulates medical information across providers and visits that generates a clear and structured timeline of a patient's care to improve understanding of a patient's medical history. Included in the provider's timeline is an ability to record, transcribe, and summarize clinical conversations to be added to the patient's record.

**Technology Used:**

- Frontend: Javascript, HTML + CSS
- Speech-to-Text: Python
- LLM Summarization: Gemini API (temporary solution)

**How *Medlog* Runs:**

*Medlog*'s home screen opens with a secure login featuring two-factor authentication, allowing for users to clarify if they are a patient or a provider. Their chosen view will bring them to their respective page. For patients, they will be able to set up their patient profile, receive invites from their providers, and view summaries given by each respective provider. For providers, they will be able to invite new patients or click on a patient to view a summarized timeline-view of the patient's medical record.

The *Medlog* transcription technology currently operates off a blend of python and Google's Gemini architecture. For version 1.0 of the product, the transcriber operates locally off the system and requires

---

<sup>1</sup> <https://etactics.com/blog/patient-retention-statistics>

that a user input their own Gemini API key for operation. As it runs, python records the conversation, outputs the local .wav file to an encrypted Gemini window that transcribes then summarizes the conversation into a markdown window looking specifically for medical terminology. After this summary is outputted, the .wav file is deleted for security purposes.

### **Business Model:**

*Medlog* has been designed for a seamless integration into existing healthcare systems. In its infancy, this summarization and modeling technology would have to be deployed within a local scale at select clinics with patients willing to experiment with this new technology.

With *Medlog* iteration over time, an ideal adoption would occur by companies such as EPIC, who already manage systems such as MyChart. Large-scale adoption of the *Medlog* offers existing ecosystems a drastic increase in quality of life to patients and providers. While this adoption means sacrificing flexibility from being disconnected from a larger ecosystem, *Medlog* would benefit greatly from being able to utilize existing secure and compliant data structures in its deployment. In more fiscal terms, data-centers are expensive while *Medlog* seeks to operate with zero economic profits.

### **Important Notice on HIPAA Compliance and Data Security:**

HIPAA compliance requires multiple considerations regarding the backend and frontend of the *Medlog* software, beyond the patient consent within the clinical environment. First, *Medlog* needs to adopt secure 2FA log-in systems to validate the identity of those who log in. Second, *Medlog* needs to incorporate an auto-log out system to manage these identities over time. Third, *Medlog* servers should provide an audit log of when a patient's records are being accessed to maximize security in the patient ecosystem. Fourth, *Medlog* should ideally adopt a cloud-based ground-up architecture that recreates the Gemini Ecosystem without having to record/send clinical conversations to outsourced companies. This rebuilding of an LLM could take time, but hospitals such as UF Shands' GatorTron provide proof of concept. Finally, to ensure a robust backend that supports many users while maintaining HIPAA compliance, a third-party cloud environment appears ideal (such as AWS or Azure). In this cloud-storage, a business negotiation such as a BAA (Business Associate Agreement) would be required to minimize liabilities while maximizing security and compliance between all parties involved.

### **Future Development Roadmap:**

Because the DTE Hackathon constricted the team to a 24 hour period, a period which did not accommodate for a general lack of back-end development skills and funding for cloud servers, a list of future developments has been created.

- Ability to record transcript and have it save to the timeline view (see comments about HIPAA).
- Account links/invitation between provider and patient information
- Provider can see a list of every provider a patient has along with contact information
- Patient has a medical history and medical timeline option in their menu
- When viewing patients, information defaults to a timeline of summaries, with the option to view full medical history
- Messaging feature between providers regarding a patient

- Creation of a more robust backend / API for upscaling and security
- Development of secured LLM to cut out middle-man in data sending
- Cloud Data Storage (patient-to-provider encryption)
- Secure Server Hosting