

# LocalTranscript (VoiceType)

macOS menu bar app for Vietnamese/English speech-to-text dictation. Hold a hotkey, speak, release — text appears at your cursor. Runs 100% offline using WhisperKit on Apple Silicon.

## Features

- **Hold-to-talk:** Hold Option+Space to record, release to transcribe
- **Toggle mode:** Press to start recording, press again to stop
- **Universal text insertion:** Works in any app via Accessibility API
- **100% offline:** All processing on-device with CoreML-optimized Whisper models
- **Vietnamese + English:** Supports both languages with automatic detection
- **Menu bar app:** Minimal UI, always accessible from menu bar
- **Floating indicator:** Visual feedback during recording/transcription
- **Transcription history:** Review, copy, or delete past transcriptions
- **Customizable:** Choose hotkey, model size, language mode

## Requirements

- macOS 14.0 (Sonoma) or later
- Apple Silicon Mac (M1/M2/M3)
- ~1-4 GB disk space for Whisper models (varies by model size)

## Installation

### Download Release

Download the latest [.dmg](#) from [Releases](#) and drag to Applications.

### Build from Source

#### Requirements:

- Xcode 16+ with Swift 6.0 (recommended)
- Xcode 15.3+ with Swift 5.10 (requires compatibility fixes below)

```
# Clone the repository
git clone https://github.com/quanvan1999/voicetype.git
cd local-transcript

# Open in Xcode
open LocalTranscript/LocalTranscript.xcodeproj

# Or build from command line
xcodebuild -project LocalTranscript/LocalTranscript.xcodeproj \
-scheme LocalTranscript \
-configuration Release \
build
```

## Swift 5.10 Compatibility (Xcode 15.x)

If building with Xcode 15.x (Swift 5.10), you may encounter dependency issues. Fix by pinning `swift-async-queue` to version 0.5.2:

1. Open `LocalTranscript.xcodeproj/project.pbxproj`
2. Find the `swift-async-queue` package reference and change:

```
kind = upToNextMajorVersion;
minimumVersion = 1.0.0;
```

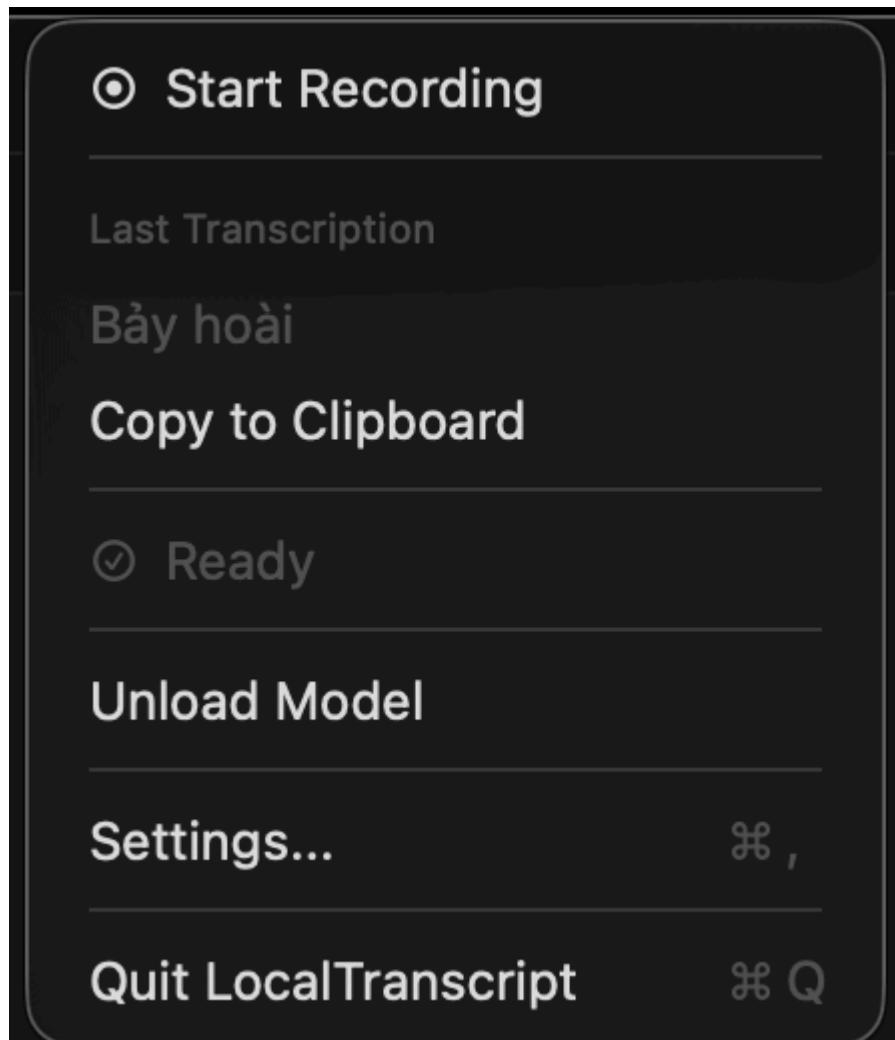
to:

```
kind = exactVersion;
version = 0.5.2;
```

3. Update `Services/TranscriptionQueue.swift` - replace  
`Task(on: fifoQueue)` with  
`fifoQueue.enqueueAndWait()`

## Usage

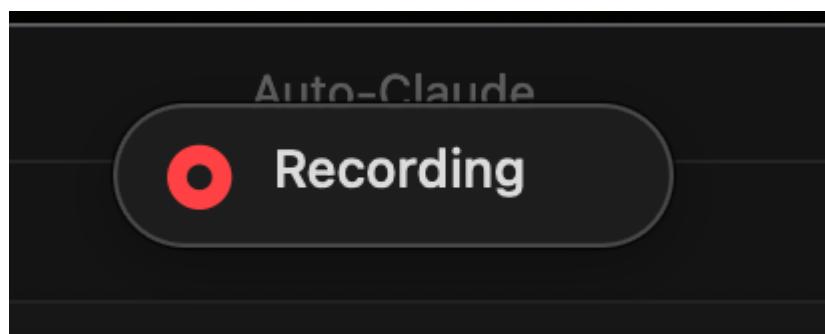
1. **Launch the app** — appears in menu bar with microphone icon



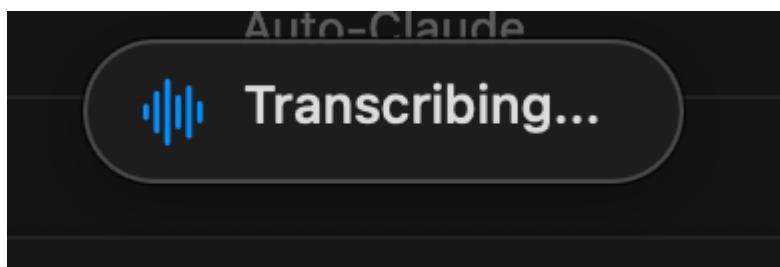
2. **Grant permissions** when prompted:

- Microphone access (for recording)
- Accessibility access (for text insertion)

3. **Hold Option+Space** and speak



4. **Release** — text is inserted at cursor



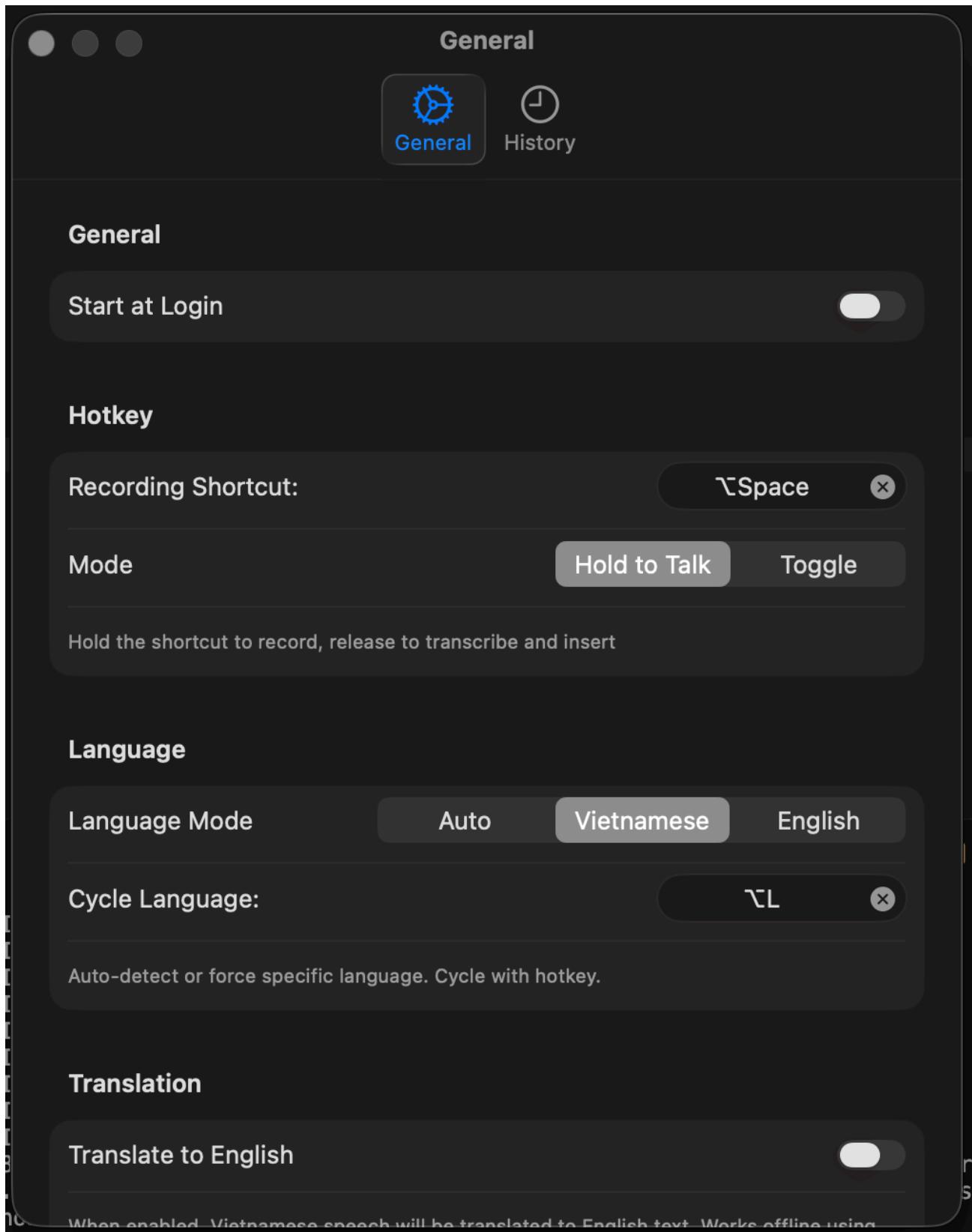
## Hotkeys

Hotkey	Action
Option+Space	Hold to record, release to transcribe
Option+L	Cycle language mode (Auto → Vietnamese → English)

## Settings

Click the menu bar icon → Settings to configure:

- **Hotkey:** Change the recording hotkey
- **Mode:** Hold-to-talk or toggle on/off
- **Model:** Choose Whisper model size (tiny/base/small/medium/large)
- **Language:** Auto-detect, Vietnamese only, or English only
- **Launch at login:** Start automatically on login



## Transcription History

View and manage your past transcriptions:

History

General History

I need to speak Vietnamese.  
7 hrs, 50 min - Vietnamese - 4.3s EN

The main features of the current LeBow  
7 hrs, 51 min - Vietnamese - 6.3s EN

Any updates in the next 2 days?  
8 hrs, 13 min - Vietnamese - 15.4s EN

Facebook.com/blogmodel  
8 hrs, 20 min - English - 20.0s

Okay, it's working.  
8 hrs, 21 min - English - 8.1s

Xin chào  
1 day, 4 hrs - Vietnamese - 3.7s

ừ ừ à à à  
1 day, 4 hrs - Vietnamese - 14.4s

ừ ừ  
1 day, 4 hrs - Vietnamese - 1.7s

ừ ừ  
1 day, 4 hrs - Vietnamese - 1.6s

Hey hey  
1 day, 4 hrs - Vietnamese - 6.9s

 Clear All

## Architecture

```

LocalTranscript/
├── LocalTranscriptApp.swift      # App entry, MenuBarExtra +
Settings
└── Models/
    ├── AppState.swift          # Central state management
    ├── LanguageMode.swift      # Auto/Vietnamese/English
    └── StatusIndicatorState.swift
└── Services/
    ├── TranscriptionService.swift # Recording → transcription →
insertion
    ├── AudioRecorder.swift      # AVAudioEngine, 16kHz capture
    ├── HotkeyService.swift      # Global hotkey handling
    └── TextInsertionService.swift # AXUIElement + clipboard
fallback
    ├── ModelManager.swift       # WhisperKit model loading
    └── HistoryManager.swift     # SwiftData persistence
└── Views/
    ├── MenuBarView.swift        # Menu bar dropdown
    ├── SettingsView.swift       # Settings window
    ├── StatusIndicatorPanel.swift # Floating recording overlay
    └── HistoryView.swift        # Transcription history

```

## Tech Stack

- **SwiftUI** with MenuBarExtra for menu bar integration
- **WhisperKit** for CoreML-optimized speech recognition
- **KeyboardShortcuts** for global hotkey handling
- **SwiftData** for transcription history
- **Accessibility API** for text insertion into any app

## Permissions

The app requires these permissions:

Permission	Why
Microphone	Record speech for transcription
Accessibility	Insert text into other applications

Input Monitoring

Detect global hotkey presses

The app runs **non-sandboxed** because Accessibility API and global hotkeys require it.

## Privacy

- All audio processing happens locally on your Mac
- No audio data is ever sent to any server
- Transcription history stored locally in SwiftData
- No analytics or telemetry

## Uninstall

1. Quit LocalTranscript from the menu bar
2. Delete the app: drag `/Applications/LocalTranscript.app` to Trash
3. Remove support files (optional):

```
# Application data and settings
rm -rf ~/Library/Application\ Support/LocalTranscript/

# Cache files
rm -rf ~/Library/Caches/LocalTranscript/

# WhisperKit models (shared with other WhisperKit apps)
rm -rf ~/Library/Caches/huggingface/
```

## Troubleshooting

### Text not inserting in some apps (Electron apps like VS Code, Slack)

These apps don't fully support Accessibility API. The app automatically falls back to clipboard paste (Cmd+V). Your clipboard is restored after insertion.

### Model download stuck

WhisperKit downloads models on first use. Check internet connection and try a smaller model first (tiny/base).

## **Hotkey not working**

Go to System Settings → Privacy & Security → Input Monitoring and ensure LocalTranscript is enabled.

## **Contributing**

Contributions welcome. Please open an issue first to discuss proposed changes.

## **License**

MIT License - see [LICENSE](#) for details.