# Meeting Transcription Service Manual

## 1. Introduction

This manual describes setup and usage for the Meeting Transcription Service, which uses Azure Speech Services, Flask-SocketIO, and SQLite to provide real-time speech-to-text transcription and persistent storage of transcripts.

## 2. Features

- Continuous speech recognition via Azure Cognitive Services  
- Real-time client updates with Flask-SocketIO  
- Persistent SQLite storage of all transcripts  
- Rotating file logging for monitoring and debugging

## 3. Tech Stack

- Python 3.9+  
- Flask  
- Flask-SocketIO  
- Flask-SQLAlchemy  
- Azure Cognitive Services Speech SDK  
- SQLite  
- python-dotenv

## 4. Installation

1. Clone the repository:  
 git clone https://github.com/your-username/meeting-transcription.git  
 cd meeting-transcription  
  
2. Create and activate a virtual environment:  
 python3 -m venv venv  
 source venv/bin/activate  
  
3. Install dependencies:  
 pip install -r requirements.txt

## 5. Configuration

Create a `.env` file in the project root with the following variables:  
  
SPEECH\_KEY=  
SPEECH\_REGION="southeastasia"  
OPENAI\_API\_KEY=  
BLOB\_SAS\_TOKEN=  
BLOB\_SAS\_URL=  
SQLALCHEMY\_DATABASE\_URI="sqlite:///transcripts.db"  
SECRET\_KEY="secret-key"  
  
Ensure `config.py` reads these values correctly.

## 6. Usage

1. Run the application:  
 python app.py  
  
2. Open your browser at http://localhost:5000 to connect a client.  
3. Speak into your default microphone — recognized text appears on-screen and is saved to `transcripts.db`.  
4. Inspect stored transcripts manually:  
 sqlite3 transcripts.db  
 .tables  
 SELECT \* FROM transcript LIMIT 5;

## 7. Project Structure

```  
meeting-transcription/  
├── app.py # Main application logic  
├── config.py # Configuration settings  
├── requirements.txt # Dependencies  
├── transcripts.db # SQLite database file (auto-generated)  
├── .env.example # Example env vars  
└── templates/  
 └── index.html # Client UI template  
```

## 8. Logging

Logs are written to `app.log` using a rotating file handler (max 10MB per file, 5 backups). Check this file for INFO, WARNING, and ERROR entries.

## 9. License

This project is licensed under the MIT License.