

Mind the Sensors: Interactive Consent for Smart Environments

Presenter: Junyu Liu, Maoqi Xu, Jacob Yang
Mentor: Brian Wang
Professor: Prof. Mani Srivastava

Motivation and Objectives

Objectives:

- Let people walking into a smart room easily say “yes/no” to what sensors can collect about them.
- Make simple, clear consent screens or prompts that work with what’s available: phone, camera, or voice.
- Record what they chose so the space remembers and doesn’t ask twice.

Stakeholders:

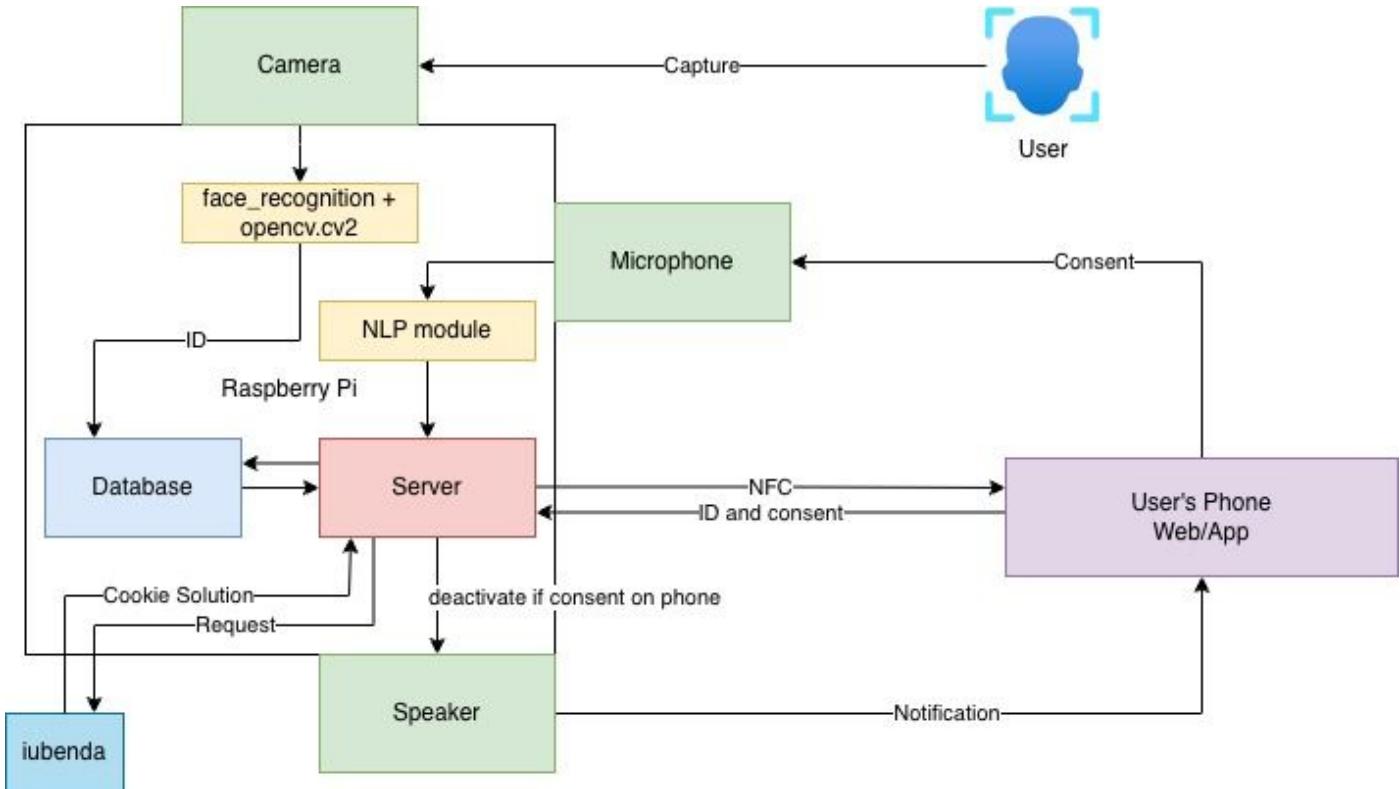
- Visitors feel informed and in control instead of watched.
- Building owners get a cleaner, more trusted way to run smart spaces that respects privacy rules.
- Designers and developers gain reusable patterns to add consent to real spaces, not just websites.
- Visitors with disability or without an electronic device at hand

Deliverables:

Scenarios: 4 concrete entry setups (camera-web based, camera-voice based, voice-only based, and voice-website based), each with clear assumptions.

Prototypes: Working demos

Methods - Facial Recognition based



- Server - Flask app + iubenda
- Database - SQLite
- Embedded platform - Raspberry Pi
- Facial Recognition - Face_recognition + opencv.cv2

Methods - Server



Flask Application (Python):



API Startpoints: Polling the database to ask if a face is detected.



Logic Layer: Processes incoming sensor data (Face ID) and queries the database to determine if the user has already provided consent.



Web Serving: Renders the 'Welcome' and 'Consent' pages for users accessing the system via web pages



Lubenda Integration (US State Law):



Compliance: Embeds a standardized cookie and privacy policy solution directly into the Flask templates.



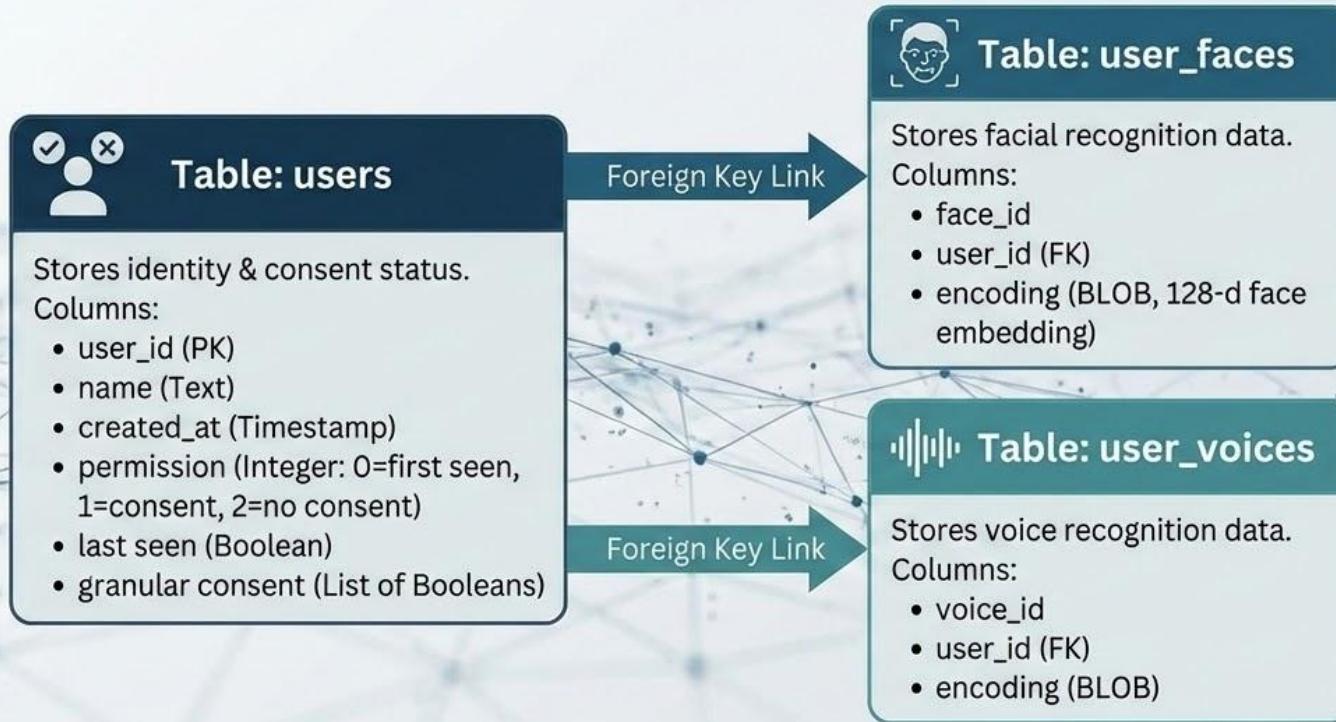
Cookie Management: Automatically handles the 'Essential' vs. 'Analytics' cookie consent banners required for privacy compliance



Granular Choices Display: Uses a callback function to retrieve return user's granular cookie options and display on the welcome page

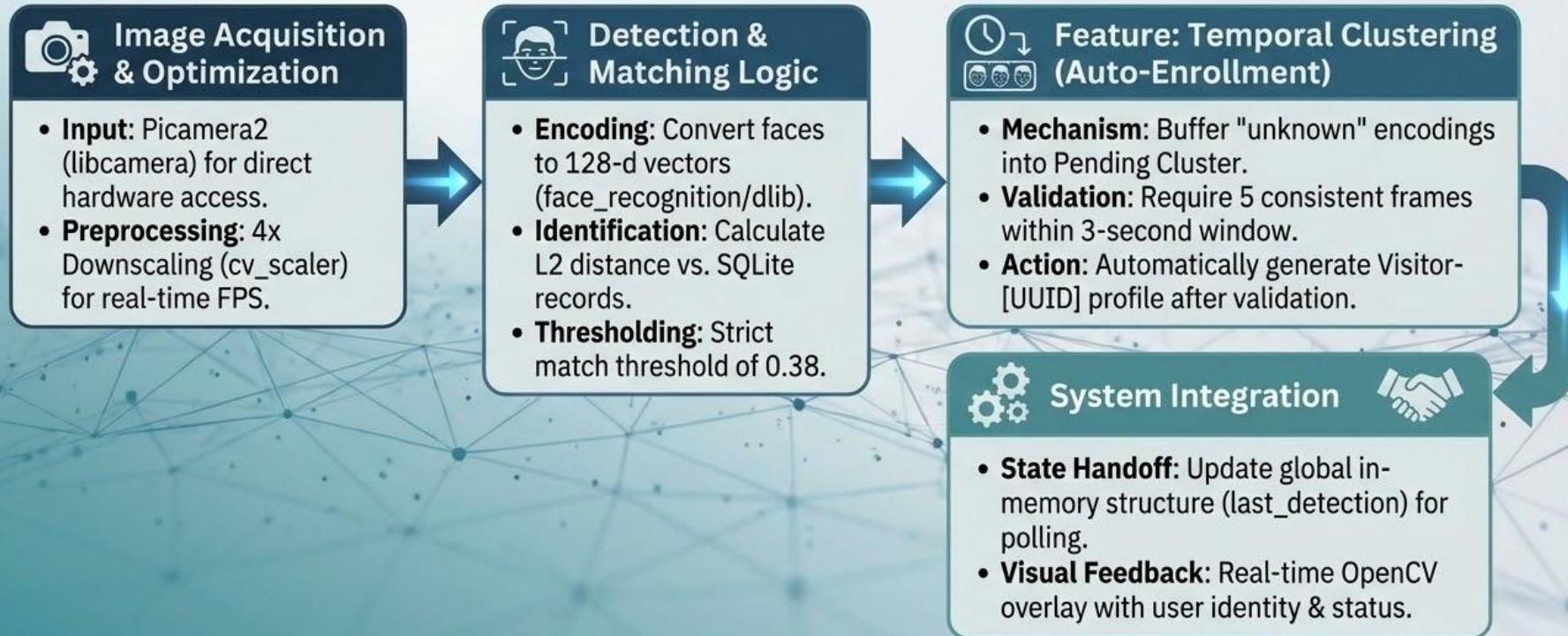
Methods: Database

Technology: SQLite (Lightweight, serverless, ideal for Raspberry Pi)



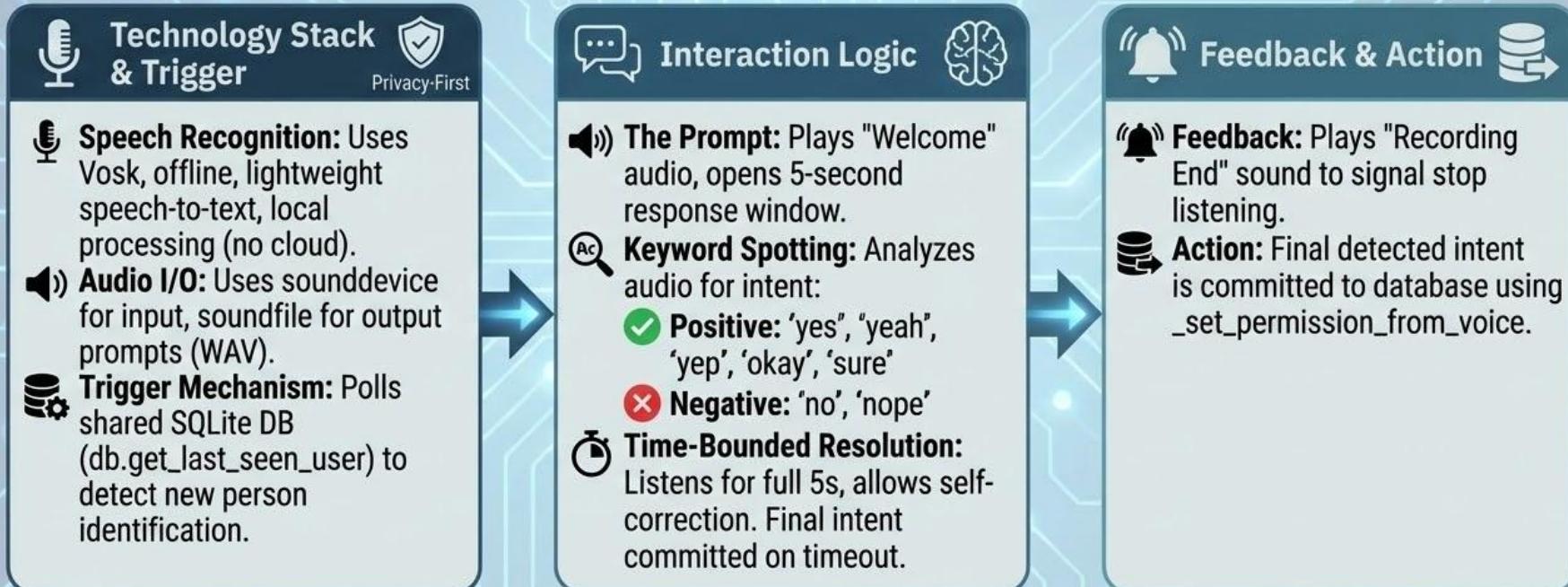
Methods: Facial Recognition

Facial Recognition Pipeline

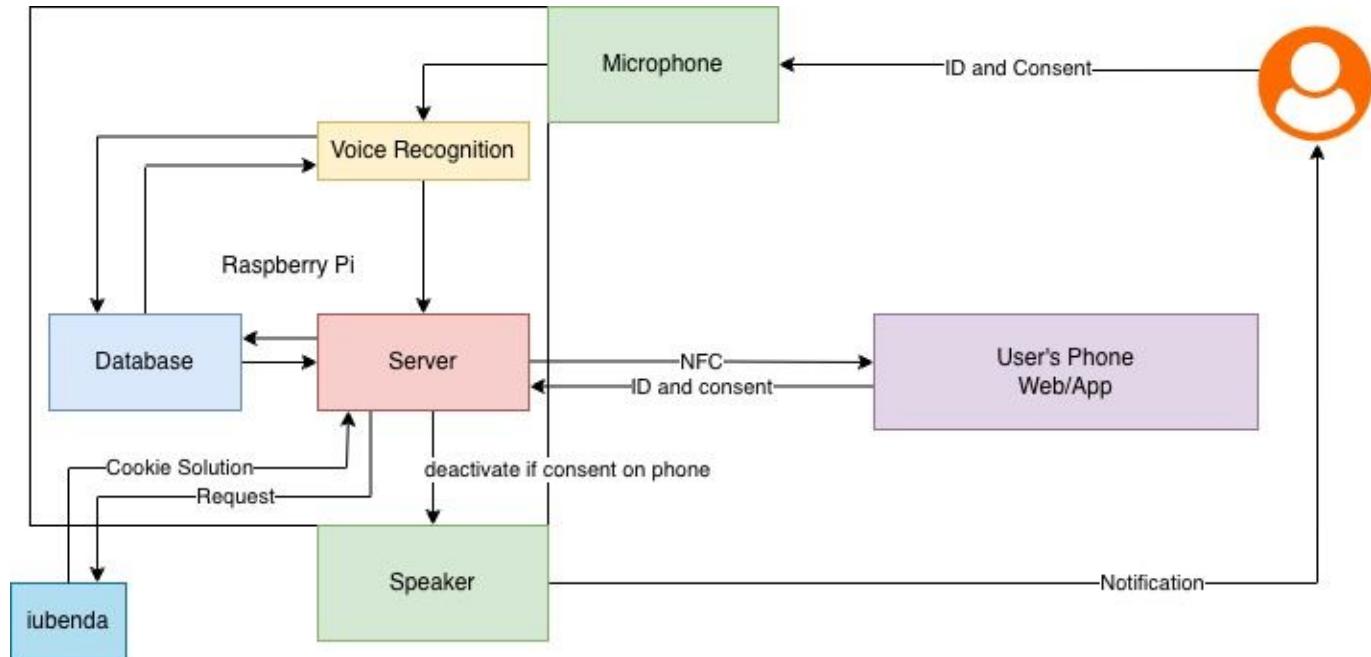


Methods: NLP Module

Natural Language Processing & Voice Interaction



Methods - Voice Recognition Based



- Server - Flask app + iubenda
- Database - SQLite
- Embedded platform - Raspberry Pi
- Voice Recognition - Resemblyzer

Methods - Voice Recognition

Audio Acquisition & Wake Word:

- **Input:** Uses sounddevice for raw audio stream capture (16kHz sample rate) to ensure low-latency processing.
- **Activation:** Implements an "Always-Listening" loop using **Vosk (Offline ASR)** to detect the specific wake phrase "**Apple**" before triggering the recording pipeline.
- **Privacy:** All speech processing happens locally on the Raspberry Pi; no audio is sent to external clouds.

Speaker Identification (Biometrics):

- **Core Engine:** Utilizes Resemblyzer, a deep learning library derived from Google's GE2E (Generalized End-to-End Loss) model, to extract voiceprints.
- **Vectorization:** Converts a 3-second audio clip into a **256-dimensional floating-point embedding** that uniquely represents the speaker's vocal characteristics.
- **Matching Metric:** Uses **Cosine Similarity** (ranges 0.0 to 1.0) rather than Euclidean distance.
- **Threshold:** Sets a strict similarity threshold of **0.75** to differentiate between distinct speakers..

Evaluation of usability - Camera Based

Voice	Clarity	Ease of use	Trust	Success Rate
User_1	4	3	5	3/3
User_2	4	2	5	2/3
User_3	3	3	5	2/3

Website	Clarity	Ease of use	Trust	Success Rate
User_1	4	3	5	3/3
User_2	5	3	5	3/3
User_3	5	4	5	2/3

Evaluation of usability - Voice Based

Voice	Clarity	Ease of use	Trust	Success Rate
User_1	5	4	4	3/3
User_2	4	4	5	3/3
User_3	5	4	5	3/3

Website	Clarity	Ease of use	Trust	Success Rate
User_1	5	5	4	3/3
User_2	5	5	5	3/3
User_3	5	4	4	2/3

Evaluation

Facial Recognition:

- False Match Rate (FMR) = around 6/20 with threshold already lower to 0.38
- False Non-match Rate (FNMR) = 0 but the official rate for dlib's Resnet is ~ 0.2%

Voice Recognition:

- Keyword Spotting Accuracy = ~100% as we didn't see a mismatch for yes/no
- False Match Rate (FMR) = 3/20

DEMO