Boston University
Electrical & Computer
Engineering

Boston University
Electrical & Computer Engineering
EC 463 Senior Design Project

First Prototype Test Report

BUtLAR

By
Team 12
Digital Human - Yobe

Team Members

Noa Margolin noam@bu.edu
Suhani Mitra suhanim@bu.edu
Jackie Salamy jesalamy@bu.edu
Andrew Sasamori sasamori@bu.edu

Required Materials:

Hardware:

- Raspberry Pi V5
- Two Røde Microphones
- LCD Screen (PHO 113 computer lab monitor)

Software:

- Shell Script
    - arecord (Linux binary/command)
    - g++
    - Python Virtual Environment
- Yobe SDK (GrandE)
- Audio Generation
    - IDListener_demo.cpp
    - normalize_wav.cpp
- Google Speech-To-Text API
    - googleTabulate.py
- OpenAI API
    - OpenAItesting.py

Prototype/MVP Goal: has backend pipeline integration of functioning noise immunity, personalized voice recognition, and gives relevant/ accurate responses

Setup:

Our system setup begins with the hardware components: a Raspberry Pi connected via Ethernet to host the software on a Linux machine and two Rode Microphones for capturing audio input. The microphones are set at a standard of 8 inches apart, facing upwards. The pipeline is driven by a Bash script that automates the processes of audio capture, processing, and response generation. As depicted in Figure 1, the backend workflow captures audio, normalizes it, and processes it using Yobe's SDK. The pipeline then performs speech-to-text transcription, saving the resulting text to a .txt file. Subsequently, the OpenAI-powered LLM generates a response based on the public information for general questions and utilizes our prompt engineering document for use-case-specific instances. For this test, we utilize a BU-specific database with information about certain professors' classes taught. Finally, the LLM-generated response is conveyed through a digital human, enabling seamless and interactive user interface (UI) engagement.

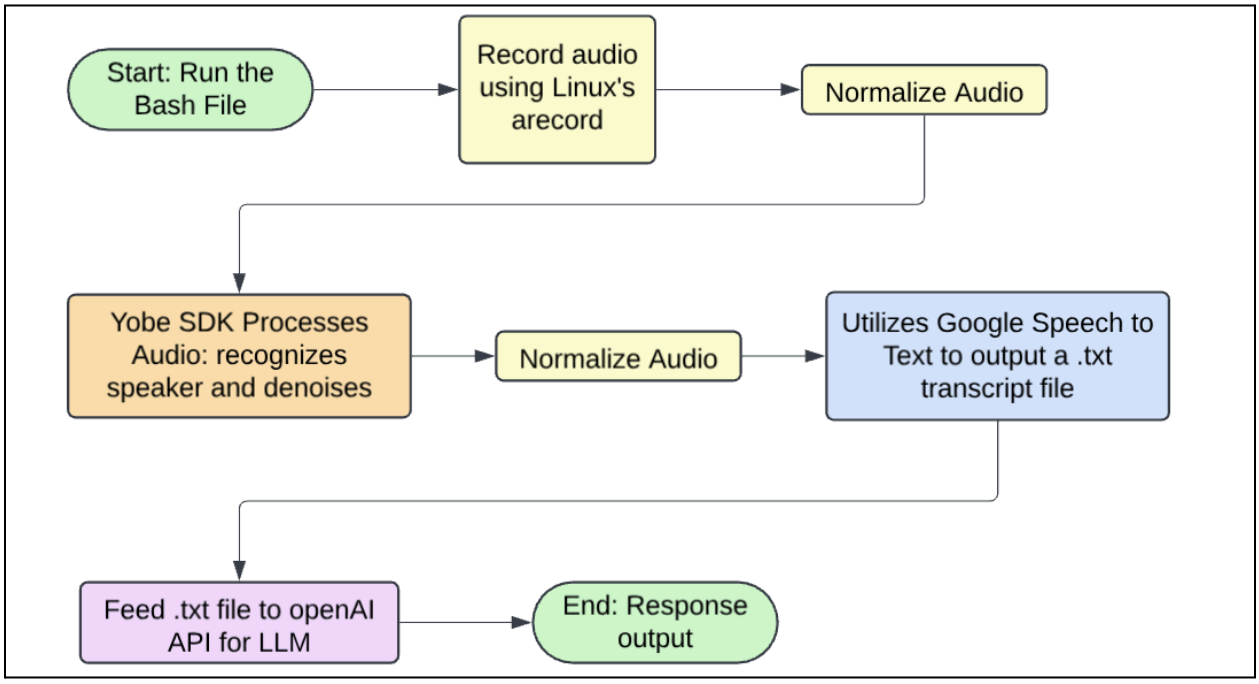

Figure 1: Illustration of Backend System Integration

Pre-Testing Setup Procedure:

Raspberry Pi Connection:

1. 2 AI-Micro Rode Dual Speakers are connected to Raspberry Pi.
2. Raspberry Pi is connected to the network via Ethernet.
3. Run the Bash script generate_wav.sh on the Raspberry Pi.

Server-Side Connection:

1. Establish SSH connectivity with the Raspberry Pi (remote access) using the following command: **ssh yobe@128.197.180.176**
2. Navigate to the appropriate directory:
    a. cd BUtLAR_Voice-Powered-Digital_Human_Assistant/Audio

Running Bash Script

1. Execute permissions to the script generate_wav.sh: **chmod +x generate_wav.sh**.
2. Execute the script to begin recording: **./generate_wav.sh**.

Testing Procedure:

There are 8 specific tests that must be evaluated as either "Pass" or "Fail." To achieve a "Pass," each test must meet its unique criteria, ensure a latency of less than 12 seconds from the end of audio recording to transcript generation, and produce a transcript that accurately conveys the intended message.

1. Personalized voice recognition
    a. Personalized Recognition Test 1: Andrew main speaker, Noa background
    b. Personalized Recognition Test 2: Suhani main speaker, Jackie background
    c. Play the post-processed audio file to verify this
2. Works with noise
    a. Noise Environment Test 1: Crowd chatter at 3 volumes and play the output
    b. Noise Environment Test 2: Beeping at 3 volumes and play the output
3. Relevant/accurate responses to questions that can be answered from the internet or specific BU questions i.e. our BU-specific database with classes taught.
    a. First, ask general internet question:
        i. Relevance Response General Test 1: About Boston
    b. BU-specific question – has to be information accessible in the database
        i. Relevance Response Use-Case Test 1: Classes taught
        ii. Relevance Response Use-Case Test 2: Find me the office hours
4. Can connect to UI
    a. Plays on D-iD, digital human accurately speaks LLM-generated response.

Measurable Criteria:

Specific Test Case Requirements:

I. **Personalized voice recognition:**
   A. Two speakers, one authorized through the template file and one not.
   B. Results will be verified by replaying the audio to confirm correctness. Processed audio must only contain the authorized speaker's voice.

II. **Works with noise:**
   A. The transcript after the speech-to-text conversion will be checked for conveying the correct message. The transcript must accurately process the spoken message and output the corresponding text.

III. **Relevant/accurate responses** to questions that can be answered from the internet or specific BU questions (i.e. small database of classes taught) and **can connect to digital human UI:**
   A. General and case-specific questions will be asked.
   B. The LLM response will be checked for conveying an accurate message answer to the speaker's query.
   C. D-iD video speaks out the LLM response.

General Requirements:
In addition to satisfying the criteria above, the system must meet the following overarching requirements for every test case:

- **Latency:** The time from the end of the audio recording to the generation of the LLM-generated response must be less than 12 seconds.
- **Message Accuracy:** The speech-to-text transcript must accurately convey the intended message query.

Score Sheet:

Requirement	Transcript is correct (Y/N)	Latency (<12s)	Pass/Fail
Personalized Recognition Test 1	N/A	N/A	Pass
Personalized Recognition Test 2	N/A	N/A	Pass
Noise Environment Test 1	Y	8.436254567 seconds	Pass
Noise Environment Test 2	Y	8.138132386 seconds	Pass
Relevance Response General Test 1	Y	8.781687853 seconds	Pass
Relevance Response Use-Case Test 1	Y	8.950091169 seconds	Pass
Relevance Response Use Case Test 2	Y	9.257524406 seconds	Pass
Can connect to UI (D-ID)	Y	N/A	Pass
Result →		8 /8	

Test Data Conclusions

Following the testing of BUtLAR, we established that for a test to be deemed successful, the transcript must be accurate, and the latency must be under 12 seconds. In Personalized Recognition Test 1, where Andrew was the primary speaker and Noa was in the background, the audio was successfully processed, isolating only Andrew's voice while effectively denoising Noa's, resulting in a pass. Similarly, in Personalized Recognition Test 2, where Suhani was the primary speaker and Jackie the background voice, only Suhani's voice appeared in the output, also resulting in a pass.

For the Noise Environment Test 1, which included crowd chatter at 3 volumes, the transcript was accurate, outputting: "That's great! Staying hydrated is important for your overall health. If you have any specific questions or topics in mind related to health, hydration, or

anything else, feel free to ask!" with a latency of 8.44 seconds, thus passing successfully. Likewise, for Noise Environment Test 2 involving a beeping sound at 3 volumes, it produced the correct transcript, "That's great! Food can be a source of joy and comfort. Do you have any favorite dishes or types of cuisine?" with a latency of 8.14 seconds, therefore also passing.

In Relevance Response General Test 1 for asking a general question about Boston, the transcript was correct, yielding: "Boston is one of the oldest cities in the United States, founded in 1630…," with a latency of 8.78 seconds, confirming a pass. Similarly, in Relevance Response Use-Case Test 1 where the user asked which courses Professor Goyal teaches, the correct output was: "Yes, Vivek Goyal teaches the following courses: EC516 Digital Signal Processing…," with a latency of 8.95 seconds, also passing. Relevance Response Use-Case Test 2 produced a correct transcript: "Yes, Kyle Best's office hours are on Tuesdays from 11 am to 12 pm…" with a latency of 9.26 seconds, resulting in another pass.

The final test, if the retrieval-augmented-generated (RAG) text response can connect to UI (D-ID), correctly outputs the transcript: "Yes, Alan Pisano teaches the following courses: EC402 Feedback Control Systems…" confirming successful integration with the UI as a digital human appeared with correct audio.

While all tests passed, we concluded that the project's next phase must focus on optimizing latency to enable faster response times and improve the flow of natural conversation in machine-human interaction. The current latency disrupts conversational fluidit Potential solutions to address this include replacing Google ASR speech-to-text with WhisperAI, considering a different or lower version of the large language model (OpenAI), and exploring Yobe latency optimizations. Additional challenges identified during testing include transcribing complicated names, reducing latency (since parallel processing is not achievable), evaluating the costs and potential latency issues with D-ID as the UI platform, and investigating minor recording bugs with Yobe (i.e. imperfections in recognizing authorized user's voice). Our current design requires a full audio file query in order to begin the pipeline—each step of our integration is performed sequentially. To address this, we aim to implement real-time query input processing, handling spoken words incrementally to minimize latency. This requires enabling audio buffer supplementation at each step of the pipeline while ensuring sufficient context is available for the LLM to generate knowledgeable responses. These optimizations and refinements will be the focus as we continue to enhance our prototype in the next stages of development.