Refactor the chatbot in the Speak Activity to use gen-AI

Basic Information

Full Name: Yedian Cheng

Email: chengyedian@gmail.com

GitHub Username: YedianCheng

First Language: Chinese

Location and Timezone: Vancouver, Canada (UTC-7)

Previous Open Source Work:

I have set up a complete Sugar development environment with the Speak Activity running in a Fedora-based virtual machine, allowing me to test and modify the codebase effectively. As part of my GSoC 2025 preparation, I submitted my first pull request to Sugar Labs: feat: add mock LLM chatbot response function, along with a short demonstration video: https://www.youtube.com/watch?v=3mubSEPcv1k.

This pull request introduces a mock implementation of the get_llm_response() function to simulate chatbot replies. It establishes the foundation for integrating large language models (LLMs) into the activity, enabling future development of dynamic persona responses and AI-driven interaction.

Why I am a Good Fit:

I believe I am a strong candidate for this project because of my experience in both AI and user interface design:

I recently completed a project titled **"Building LLM Interactive Personas"**, where I developed a dynamic chatbot system using LangChain, RAG, FastAPI, and React. This system supported customizable personas and achieved a **95% satisfaction rate** in user studies.

As a former designer, I am proficient in **Photoshop and Illustrator**, and I look forward to contributing creative and expressive character illustrations (SVGs) to enhance the child-friendly feel of Speak.

As a parent of two six-year-old daughters, I truly value focused, distraction-free learning environments like Sugar. This gives me a unique perspective on how children interact with educational tools, and motivates me to build features that are both fun and meaningful. I have strong time management skills and a solid support system, ensuring I can fully commit to the GSoC timeline.

I am excited about the opportunity to contribute to Sugar Labs and grow as both a developer and a designer in the open-source community.

Project Details

What am I making?

I am developing an enhanced version of the **Speak Activity** that introduces the following key improvements:

- **Invented spelling correction**: This feature will assist early readers by interpreting and correcting misspelled words, enabling more accurate text-to-speech output and chatbot understanding.
- **Modern LLM chatbot integration**: A large language model (LLM) will be integrated to provide natural, contextually appropriate responses that complement—rather than replace—the existing AIML-based chatbot system.
- **Customizable visual personas**: Learners will be able to select from multiple persona styles (e.g., soft-spoken, sporty), each offering a distinct tone, vocabulary, and facial expression style.
- **Emotion-based facial expressions**: The chatbot's facial expressions, built using SVG graphics, will change dynamically based on the sentiment of the response to enhance emotional engagement.
- **User interface enhancements**: Key areas of the interface will be redesigned to support the new functionality while maintaining the activity's original simplicity.

Specific improvements will be integrated into existing parts of the UI:

- In the **ABC tab** (text-to-speech), invented spelling correction will enhance how input is spoken aloud.

- In the **chat tab**, the LLM-based chatbot will offer supportive, age-appropriate prompts such as "Have you tried sounding this word out?" or "That's a fun word." The character's eye and mouth shapes will update in real-time according to the chatbot's emotional tone.

- Upon first launch, the learner will be invited to select a preferred persona, which will be remembered in future sessions. A settings panel will allow persona switching and feature customization.

All new features will be optional and non-intrusive. If the chatbot backend is unavailable, the activity will still function normally using the existing AIML engine and offline text-to-speech.

How will it impact Sugar Labs?

This project brings meaningful enhancements to one of the most widely used activities in the Sugar ecosystem. By making the activity more expressive, intelligent, and accessible to early learners, this work will:

- Improve the educational value of the activity by supporting self-expression and communication.

- Demonstrate how modern AI technologies such as LLMs and intelligent spell correction can be responsibly integrated into child-friendly learning tools.

- Encourage contributors and educators to explore new pedagogical opportunities with Sugar.

- Serve as a model for incorporating AI features in lightweight, offline-compatible educational environments.

Technologies

The following technologies will be used:

- **Python 3** – for activity logic and backend functionality.

- **Sugar Toolkit (GTK+3, sugar3)** – for UI and activity integration.

- **GStreamer** – for text-to-speech output.
- **LangChain and FastAPI** – for prototyping server-side LLM response logic, if needed.
- **SymSpell or Hunspell** – for efficient spelling correction.
- **SVG (via Cairo graphics)** – for rendering and animating facial expressions.
- **Git** – for version control and collaboration.

Deliverables

Required:

- Functional integration of an LLM-based chatbot that complements the existing AIML system.
- A working invented spelling correction module integrated into the ABC (text-to-speech) interface.
- At least two configurable chatbot personas, each with a unique visual and conversational style.
- Emotion-based facial expression system with corresponding SVG assets.
- A configuration panel for enabling or disabling features and switching personas.
- Documentation for users, developers, and future contributors.

Optional:

- Speech-to-text input for more natural interaction.
- Audio-based feedback that matches the emotional tone of text.
- Multi-language support to expand accessibility.
-

Timeline

Weekly Commitment:

- Estimated: **20–25 hours/week**
- Vacation adjusted buffer during Week 2

Progress Reporting:

- Weekly check-ins with mentor via GitHub issues and Zulip.
- Use demo videos, logs, and screenshots to share milestones.
- Midterm and Final evaluations submitted through GSoC platform.

Post-GSoC Plans:

- Continue contributing to Sugar Labs by:
    - Supporting future contributors in the Speak Activity.
    - Proposing educational persona expansions and UX enhancements.
    - Exploring audio/speech improvements for low-resource setups.

Week	Dates	Milestones & Tasks
Pre-GSoC	May 8 – June 1	- Familiarize with the Speak Activity codebase and Sugar Toolkit. - Review past related implementations (e.g., AIML logic, SVG emotion rendering). - Gather user feedback and mentor advice on current Speak Activity features.

		- Design technical plan for modular LLM integration and persona support. - Initial UX sketches and persona concepts.
Week 1	June 2 – June 8	*Already completed* - Scaffolding mock LLM integration. - Ensured compatibility with existing AIML engine. - Confirmed local activity build and modification pipeline.
Week 2	June 9 – June 15	- Implement persona configuration structure. - Design UI for persona selection on first launch. - Integrate SVG variations for 2 base personas. **Vacation notice**: June 12–17 (limited availability)
Week 3	June 16 – June 22	- Introduce sentiment tagging into mock LLM replies. - Link sentiments to corresponding SVG facial expressions. - Begin prototype of spelling correction (e.g., SymSpell).
Week 4	June 23 – June 29	- Refine spelling correction and show suggestions in ABC tab. - Simulate invented spelling tests and evaluate suggestions.
Week 5	June 30 – July 6	- Refactor LLM and spelling logic into independent modules. - Add Settings toggle to enable/disable LLM. - Mid-point testing, feature freeze, and UI review.

Week 6	July 7 – July 13	- Begin prototyping actual LLM backend (e.g., Ollama, GPT4All). - Explore local/offline-friendly LLM models. - Prepare demo and documentation for midterm evaluation.
Week 7	July 14 – July 20	- Integrate LLM backend into Speak Activity. - Implement fallback mechanism to AIML if LLM fails. **Midterm Evaluation**
Week 8	July 21 – July 27	- Improve chatbot response quality based on spelling confidence. - Test LLM personas with different tone settings.
Week 9	July 28 – August 3	- Expand persona UI and allow users to switch persona styles. - Add at least one new persona (e.g., sporty, cheerful).
Week 10	August 4 – August 10	- Finalize all UI features, including persona customization and SVG expression integration. - Perform detailed system-wide testing.
Week 11	August 11 – August 17	- Write final technical documentation and user instructions. - Record final demo video and take screenshots.
Week 12	August 18 – August 25	- Polish codebase and verify tests pass. - **Final Submission** to Sugar Labs and GSoC portal.