SCENARIO EXERCISE

How to develop your own ethical chatbot

www.aileaders-project.eu

Co-funded by the European Union

CONTENTS

Co-funded by the European Union

Co-funded by the European Union. Views and opinions expressed are however those of the author or authors only and do not necessarily reflect those of the European Union or the Foundation for the Development of the Education System. Neither the European Union nor the entity providing the grant can be held responsible for them.

Abstract

Type of OER:

- Scenario Exercise

Goal/Purpose:

- Provide a hands-on, low-cost *prototype* showing how to build and deploy an **ethical chatbot** in Google Colab.

Expected Learning Outcomes:

- The student will be able to create a functional chatbot using beginner-friendly tools, incorporating ethical design principles.

Suggested Methodological Approach (Case-Based Learning, Problem-Based Learning...):

- Problem-based learning

Keywords:

- Ethical AI, Chatbot Prototype, Google Colab, Hugging Face Transformers, DialoGPT, Gradio UI, Content Moderation, Disclaimers.

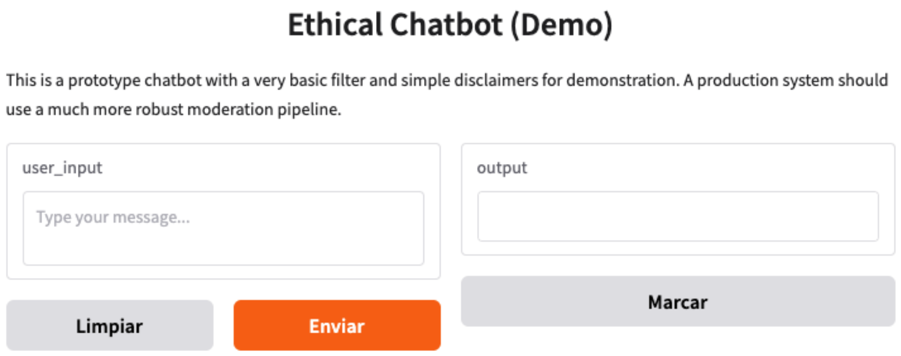
Introduction

Introduction

Generative chatbots are moving from research labs into everyday customer-service desks, classrooms, and personal devices, but their rapid adoption has outpaced equally rapid work on **Ethical AI** safeguards.

Developers who want to ship useful assistants face a key problem: *how do you add even minimal content-moderation and transparency layers without expensive infrastructure or proprietary APIs?*

In the next sections we will provide some guidelines to know the requirements that an ethical chatbot should meet, as well as a hand—on exercise to develop your own chatbot.



Ethical Chatbot (Demo)

This is a prototype chatbot with a very basic filter and simple disclaimers for demonstration. A production system should use a much more robust moderation pipeline.

Guidelines on how to develop an ethical chatbot

Requirements for an ethical chatbot (1)

1. DEFINE THE CHATBOT'S PURPOSE AND SCOPE

Identify the use case:

- Determine what your chatbot should do (e.g., answer customer service questions, facilitate mental health support, or provide general Q&A).

Clarify limitations:

- Be explicit about what your chatbot cannot do (e.g., provide medical advice, handle financial transactions).

2. ESTABLISH ETHICAL AND BEHAVIORAL GUIDELINES

Develop an "Ethical Charter" or policy:

- Outline the principles your chatbot should adhere to (e.g., avoiding harmful content, recognizing user privacy).
- Include fairness, safety, transparency, and respect for diversity.

Create a "Code of Conduct" for your chatbot:

- How should it respond to hateful or harassing content?
- What stance does it take on misinformation or disinformation?
- How does it handle requests for personal or sensitive data?

Set up fallback mechanisms: in scenarios where the chatbot is asked illegal or harmful instructions, design it to respond with a refusal or safe alternative (such as pointing the user to qualified professionals).

3. MODEL TRAINING

Source ethical training data:

- Ensure data does not infringe on privacy or copyrights.
- Strive for representative data that includes diverse languages, regions, and cultures to minimize bias.
- Filter and preprocess:
- Remove or label potentially harmful or biased content.
- Consider using off-the-shelf tools or in-house filters to identify hate speech, explicit content, or personal data.

Model selection:

- Balance model complexity with interpretability and resource usage.
- Reinforcement Learning from Human Feedback (RLHF) or similar approaches can help refine model behavior post-training.

Requirements for an ethical chatbot (2)

4. PROVIDE TRANSPARENCY TO USERS

Disclose your chatbot's nature:

- Clearly inform users they are interacting with an AI-driven system.
- Provide disclaimers about any limitations and possible inaccuracies.

Explain data usage:

- Inform users how their data is being stored, processed, and used for improving the system.
- Adhere to data protection regulations (e.g., GDPR, CCPA) and make the privacy policy easily accessible.

Offer an easy "exit":

- Let users know they can halt or opt out of data collection at any time.
- Provide a mechanism for deleting or anonymizing their data when requested.

5. HANDLE EDGE CASES AND SENSITIVE TOPICS

Refusal or safe-complete:

- For illegal, harmful, or highly sensitive requests (e.g., medical, legal, financial advice), the chatbot should either:
- Provide a disclaimer and partial information with references, or
- Direct the user to professional help.

Mental health or self-harm context:

- Prepare safe responses: express empathy, encourage the user to seek professional help, and share contact details for support services (where possible).

Emergency scenarios:

- Make it clear that the chatbot is not equipped to handle emergencies.
- Provide instructions or contact details for emergency services if recognized signals of danger appear.

6. COMPLY WITH LEGAL AND ETHICAL REQUIREMENTS

Regulatory compliance:

- Stay updated on legislation regarding AI, data privacy, and digital services in regions where your chatbot will operate.
- Implement robust data protection and consent mechanisms.

Liability considerations:

- Clearly define who is responsible when the bot provides incorrect or harmful information.
- Publish disclaimers detailing the chatbot's scope (e.g., "not a licensed medical professional").

Maintain a Clear Ethical Vision

- Maintaining an ethical chatbot is an ongoing process. Periodically revisit your principles and ensure they align with emerging regulations, societal values, and community feedback. Consistency and transparency about updates help build trust. Track performance metrics and incorporate user feedback loops to achieve continuous improvement and governance.

Tools presentation

Tools used in this scenario exercise

- **Python**: the most used programming language in Data Science
- **A python editor**: Google Colab or Jupyter Notebook
- **Libraries and Frameworks**
  - Transformers (Hugging Face): load and run the pre-trained conversational model (DialoGPT-medium)
  - PyTorch (torch): deep-learning backend for model inference
  - Gradio: zero-code web UI to deploy and share the chatbot
- **Techniques Applied**
  - Data & code setup in a free, GPU-enabled Colab notebook
  - Loading a pre-trained dialogue model and maintaining conversation state
  - Content Moderation: keyword blacklist (or pipeline classifier) to block toxic inputs
  - Sensitive-Topic Detection: keyword scan → automatic disclaimers (“consult a professional”)
  - Chatbot Response Generation: sampling from DialoGPT with history context
  - Rapid Deployment: single-cell Gradio interface that outputs a public link for testing

Hands-on activities

Solution Description (follow the provided notebook)

This exercise will guide you on how to develop a chatbot prototype built entirely with free tools: a Google Colab notebook that loads a Hugging Face Transformers model (e.g., DialoGPT), wraps it with lightweight content-moderation rules and disclaimers for sensitive topics, and serves the result through a no-code Gradio UI. In under an hour, you'll see how to go from concept to a shareable link, proving that rapid deployment and responsible design don't have to be mutually exclusive.

We provide a ready-to-run **Google Colab notebook** that walks you line-by-line through the implementation of an ethical chatbot. The notebook is structured in clearly labelled sections so you can understand and modify each layer of the solution.

Call to Action

Open the *Ethical Chatbot Demo* notebook, execute each section in order, and within minutes you'll have a live chatbot complete with basic ethical safeguards, ready to demo to teammates or iterate for production use.

Conclusion

Conclusion

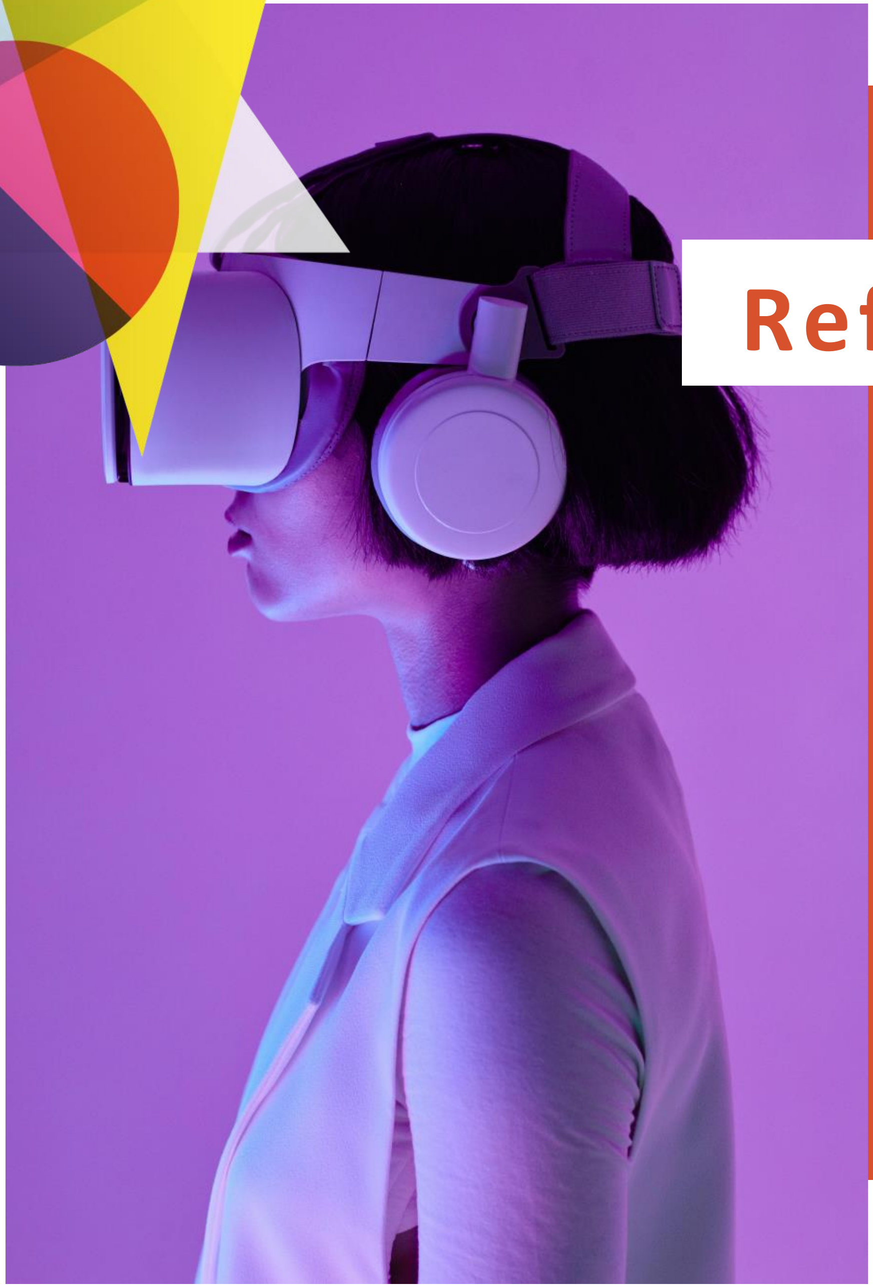
Key Insights

- Clearly defining what the chatbot should and should *not* do is the single strongest guard-rail against future misuse.
- Biases in training data propagate directly into model behavior; diverse, privacy-respecting datasets are non-negotiable.
- A real-time moderation filter catch different failure modes.
- Transparency is a must. Users who know they're talking to an AI, understand the limits, control their data and are more forgiving of occasional mistakes.
- Governance must be continuous, to ensure not only a good performance, but also the accomplishment of legal and ethical requirements.

Key Lessons

1. It's easier to soften overly strict safeguards than to bolt on moderation after harmful outputs have reached users.
2. Track harmful-output rates, demographic fairness, and user satisfaction—then tie model updates to concrete targets.
3. Smaller, interpretable models with clear guard-rails often out-perform larger black-box models on safety and compliance.
4. Tools and policies fail without a team mindset that prioritizes user well-being over speed or novelty
5. Embed ethics into every sprint.

References



References

- **European Commission.** (2024). *EU Artificial Intelligence Act:* *https://digital-strategy.ec.europa.eu/en/policies/regulatory-framework-ai*
- **Hugging Face Transformers Documentation:** https://huggingface.co/docs/transformers/en/index
- **Gradio documentation**: https://www.gradio.app/docs

Complementary material

Click to type…

Jupyter Notebook (IPYNB) File

- A detailed Python Notebook with commented code that walks through each step of the process is included.
- The notebook contains:

#	Notebook section	What it does
1	**Environment Setup**	Installs transformers, torch, and gradio; checks for GPU.
2	**Model Loading**	Pulls the open-source conversational model DialoGPT-medium (or any model you specify) from Hugging Face and initialises a conversation buffer.
3	**Basic Moderation Filter**	Implements a simple keyword blacklist *and* shows how to swap in a Hugging Face toxicity pipeline for stronger filtering.
4	**Sensitive-Topic Guardrails**	Scans user input for medical, legal, or financial keywords and injects an automatic *“Please consult a professional”* disclaimer.
5	**Response Generation Logic**	Encodes the user prompt, appends chat history, and samples the model to produce a context-aware reply.
6	**Gradio UI Deployment**	Wraps the reply function in a no-code web interface; running the cell returns a public link you can share instantly.
7	**Customization Tips**	Inline comments show where to plug in a different model, expand the filter list, or add analytics.

> As a technologist, I see how AI and the fourth industrial revolution will impact every aspect of people's lives.

Fei Fei Li

leaders

Follow our journey

www.aileaders-project.eu

Co-funded by the European Union

Co-funded by the European Union. Views and opinions expressed are however those of the author or authors only and do not necessarily reflect those of the European Union or the Foundation for the Development of the Education System. Neither the European Union nor the entity providing the grant can be held responsible for them.