

How UX Becomes the Biggest Obstacle in Gen AI Projects

Author: Eric Bréhault

Date: October 7, 2024

Tags: AI | Developers | Front-end

The imitation game

The problem with intelligence is that it is very difficult to prove. On this regard, we (humans) are trained since our childhood to apply the “duck test” heuristic (“if it looks like a duck, swims like a duck, and quacks like a duck, then it probably is a duck”) to intelligence: we call intelligent something/someone which looks intelligent.

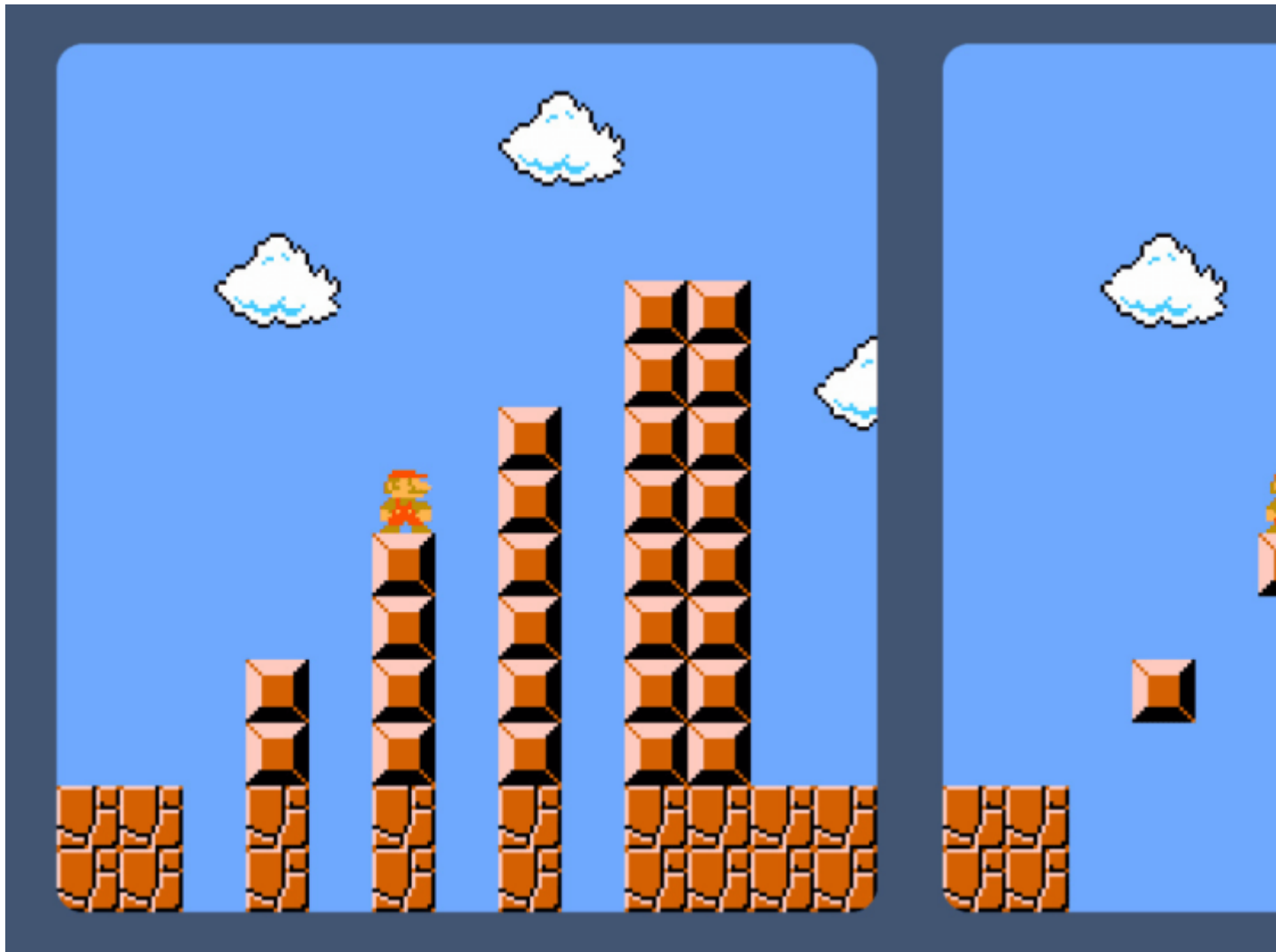
It’s not like it is a bad approach, that’s actually the only one. And as a matter of fact, that’s the very principle of the Turing test: if a machine can fool a human into thinking it is a human, then it is intelligent.

Hence AI is mostly about imitating intelligence.

UX is the way to make the duck quack like a duck.

That’s where UX comes into play. UX influences the perception the user has of any AI application. An API endpoint does not look intelligent to most people, but a chatbot interface might, even though both are essentially the same thing. As brilliantly

illustrated with these 2 Mario Bros screenshots, UX is precisely about leveraging the difference of perception between things that are essentially the same:



So let's put our UX glasses on and check what makes a chatbot look intelligent and let's forget a moment about the quality of the answers (here I need to quote Alan Turing: "The test results would not depend on the machine's ability to give correct answers to questions, only on how closely its answers resembled those a human would give.", and yes, humans do give wrong answers quite often).

The first thing that comes to mind is the pace of the answer: it does not reply in one go. It pauses between the part of the answer. Like it is hesitating, trying to find the best way to say something. Like us!!!

We also pauses, right? Okay, the LLM pauses not because it hesitates, but simply because it takes some CPU time to generate a complete sentence by calculating the most likely word one after the other.

Right, but what's the difference with our brains? We feel like these pauses are our way to connect to our inner self in order to deliver a valid idea, but unconsciously, **isn't our brain simply calculating the most likely word one after the other?** Who knows... No matter what, what is important is that these pauses make the chatbot look intelligent :).

The second thing is the way the chatbot rephrases. Instead of just returning the piece of information the way it was in the original data, it produces a new sentence. Ok, if it rephrases, it means it understand it, right?

Well, actually no, here again, the reason it rephrases is because **LLMs could be considered as compression algorithms**, they compress text in a smaller version, but with loss, so they are not able to restore the initial wording. But our natural tendency for anthropomorphism makes us think that if it rephrases, it understands.

And there are tons of other details that make the chatbot look intelligent: the way it says "Hi" (did you notice none of the most used SQL databases never say "Hi"?), the way it handles emojis, the way it handles the user's mood, etc.

We see the global user experience with generative AI is pretty good at building the illusion of intelligence, strengthening the trust the user has in the AI.

But that's a double-edged sword.

Stupid is as stupid does

You should keep in mind that intelligence and stupidity are the two sides of the same coin.

Stupidity only apply to things that are supposed to be intelligent. If a rock doesn't understand a question, it's not stupid, it's just a rock. If a cat asks for going outside, you open the door, but then the cat doesn't go outside, you might be more inclined to call the cat stupid (specially if he wakes you up at 3am for that), because you have a certain expectation of intelligence from a cat.

It's all about your expectation. The highest the expectation, the more likely you are to be disappointed.

We just said that the UX around generative AI tends to develop very high expectations from the user (partly because of user's anthropomorphism and common misunderstandings about AI, and partly because, well, your UX team is good at its job). So the risk of disappointment is high.

And it will come, inevitably. Because AI is not intelligent. It's just quacking like a duck.

Let's say you forgot to mention the current date in your system prompt. The user is having great time chatting with your AI, and then asks "What else happened since yesterday?". Here, it is very likely that the AI will answer wrongly, because it does not know the current date, plus it is not aware of it doesn't know, so it will just make up something, and look stupid.

 **When did Hedy Lamarr die?**

Hedy Lamarr died on January 19, 2000.

 **What did she invent last year?**

Hedy Lamarr invented a secret communication system that could guide military weaponry using radio frequency hopping technology is the basis for WiFi, GPS, and Bluetooth.

The disappointment will be huge, because the user had a strong feeling they were chatting with something very close to a human even though they know it is not, which is even worse, because all users know that any random stupid software they use on a daily basis is able to filter by date and get results corresponding to yesterday (the most

common SQL databases can do that). So, what they see is a human-like system not able to understand a simple question but also a software not able to do a simple filtering task.

That's where the UX challenge is.

How to manage the user's expectations

When building the UX around a generative AI, you need to keep a fragile balance between making your app look trustworthy and not overpromising.

The first thing to do is to make sure the user understands the limitations of the system. You can do that by making the AI say "I don't know" when it doesn't know, or by making it say "I'm not sure, but I think..." when it is not sure.

Then, you should try to identify wrong answers. It can be done by monitoring the chat answers and use an evaluation model, like **REMi**. You can also try to detect wrong answer in real time, and then provide alternative solutions to the user. For example, you could point the terms of the question which seem to be confusing for the LLM, or, in case of RAG, **explain the retrieved context seems not relevant enough**, possibly propose other questions that would be more consistent with the scope of the current knowledge field, etc.

What is critical is to be very transparent about the fact that the system is not magic, it is a very clever assistant, but still an assistant, implying the user is the one leading the process.

You want to avoid a situation where the user is blindly following a foolish GPS that would drive them in a dead-end.

Conclusion

As you can see, the user experience in a generative AI project will be based on both AI techniques (mostly the prompt, but also evaluation frameworks, etc.) and UI interactions.

It is a very challenging task as it implies a mix of skills and cross-disciplinary collaboration, but it is also a very rewarding one, as it is the key to make your AI project successful.