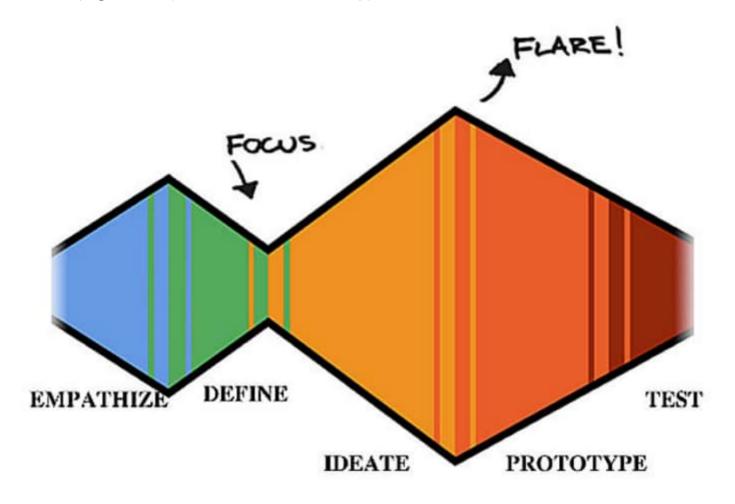
# Coze 设计小案例:多智能体AI互动故事(AI带 娃)

■ Sub-page List [Export is restricted for this type of content.]



Design I: 从问题到创意

1.【Empathize】以人为本的设计:如何发现问题



# empathy

Know about users, and care about their lives!

- Observe them, how they interact with the environment, what do they think and feel.
- Understand the way they do things and why, their physical and emotional needs, how they think about world, and what is meaningful to them.
- OBSERVE ENGAGE WATCH LISTEN
- P 半结构化访谈.pptx

# 访谈案例: AI带娃

#### 访谈实录

5

7

- 1 fangke 00:03
- 2 ok 我开始录了。
- 3 肖岚茜 00:04
- 4 您好,冯老师非常感谢您参与这次访谈,我们现在准备开发11款 AI 智能体带娃的产品,您又是一位宝 爸,所以我们希望调研您作为一个宝爸的真实需求和使用场景,以便于我们去设计出更符。
- 6 【锦囊1】了解具体的情形,将用户带入具体情境,这样可以更有效的获得用户"个体"的真实感受。
- 8 肖岚茜 00:24
- 9 和用户需求的产品,我想先问您一个问题,就是您家里有几个孩子,现在孩子多大了?
- 10 fangke 00:30
- 11 我宝宝三岁一个月三岁两一两个月,然后我只有一个宝宝。
- 12 肖岚茜 00:38
- 13 那您平时工作忙吗?您每天陪伴孩子的时间大概有多少?
- 15 fangke 00:48
- 16 我平时工作还挺忙的,一般就只有下班就早上,比如说早上起来八点到九点稍微陪他一下。可能8:30当然他要去幼儿园了,然后我也得上班了,然后晚上的话有的时候我还得上课,因为我们是不是特别规律的作息,有的时候会有晚上有课,所以。有的时候晚上甚至都没有办法陪,但是没有课的时候大概是6:30左右到家,然后晚上一直到他睡觉前。九点多一点,应该都是有时间陪的,然后就是周末了,周末一般都可以,就是周周末两天都可以。

#### 【锦囊2】问开放性问题,尤其是关于挑战、困难,可以更有效了解目前的需求

- 18 19
- 20 肖岚茜 01:42
- 21 那我听您的描述过程中,在周一到周五的时间里头,您是陪伴孩子的时间非常不固定的,那在这样的一个时间安排情况下对您带娃,您觉得有什么挑战?
- 22 fangke 01:58
- 23 带娃最大的挑战我觉得是精力的问题就是,你可以明显的感受到,如果自己的精力不够,带娃是比较偏烦躁的,因为自己也想休息,而孩子的精力是非常充足的。他会一直想要缠着你想要陪爸爸陪我玩一下什么的,但是如果当我有足够的精力,我休息好了,我可能就很想和他一起玩,然后我就会怎么说,就是这个 quality 很重要,就质量很重要。当它会形成一个正向循环,当比如说我也经历好,他也经历好,我们有创造性的在一起玩儿的时候,它就会非常开心,给我的 feedback 也好,我就更愿意有情感的投入了,就跟他玩,但如果说我经济不好,我说玩着我就分心了,然后我开始刷手机,然后他就说,爸爸不要玩手机,就陷入一个循环,因为他的情绪也不好,他就开始耍赖。就还有一点哭腔的这样说话,那可能导致我情绪就更不好,所以我觉得主要是自己的精力的恢复,然后如果工作上的事情多,那带娃的质量就差。
- 24
- 25 fangke 03:10
- 26 就是形成这样的一个负面循环,这是主要的一个挑战,然后这个是晚上平时周6日的话,我们一般都会带他出去。去反正深圳各个地方可以遛娃的地方玩玩一圈公园,什么博物馆 mall 这个一般也是在外面吃饭,这个里面就带娃的话也会有一点。
- 27 fangke 03:40
- 28 困难是有的时候出去其实还好,但是有时候会担心生病,感冒什么的,他经历太多的时候,有时候就不 愿意午睡,然后他休息不太好,就容易感冒生病。也是困难。
- 29 肖岚茜 04:01
- 30 那您说就是在带娃的过程中,你经常觉得精力不足,那一般遇到这种情况你会怎么应对?比如说让孩子自己玩,还是你找其他人帮忙? **(继续关注具体的情境)**
- 31 fangke 04:16
- 32 精力不够的情况很麻烦。有的时候会强制让自己还是分一部分精力来去怎么说,强制打起精神,然后去陪他。有一个其实很重要的就是找到一个创造性的玩法,比如说让他去画画,跟他一起画,然后画了一些有意思的东西,有的时候会跟他说 ok 我们现在是。比如说做饭,然后他就会自己拿着那些小东西,然后自己在那做饭什么的,反正就是让他去进入这个正循环的轨道,有的时候他就可以自己玩了,因为他三岁多了,他有时候会自己玩,然后这个时候我就可以稍微的。休息一下,带了稍微时间长一点的他妈就接过来了,然后我们就我就可以再去休息一下。

33

34 .....

- 36 fangke 13:55
- 37 那个佩奇还好,特别像小孩子那种感觉,对他会对自己同龄的小动物什么这些他会感兴趣了,但是对一些稍微成熟一点的,他可能会害怕,他会悄悄的跟我说爸爸你跟他说什么什么什么。就是很小声的跟我说,然后让我去帮他说这样还有什么。总体上我感觉有的时候我会有点担心他说的。
- 38 fangke 14:32
- 39 AI 说的。也就是一个,比如说教教教小孩子数数什么的,他可能会说的不对,还有就是他有时候翻译不准,因为小孩他说话。他有时候说的不准,然后他的那个语音对齐的不准之后他可能会误以为小孩,比如说小孩说了一个什么音,然后他以为是说了个脏话,然后他就会跟小孩说。这个是不,你不能说脏话什么的,小孩莫名其妙,不知道他说了啥。

- 41 【锦囊3】不断的追问,半结构化访谈并不是结构化访谈,需要在过程中不断产生疑问,向被访者确 认,获得更深入的信息。
- 43 肖岚茜 14:57

42

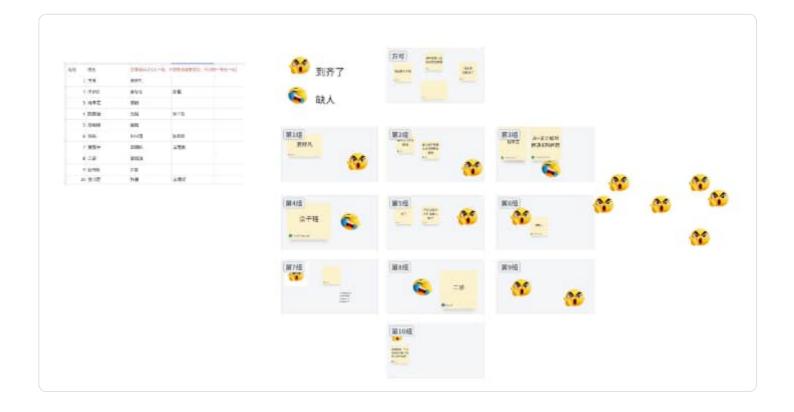
- 44 说明就是 AI 智能体,它有的时候还会出错?
- 45 fangke 15:09
- 46 不是出错,是他误以为小孩说了一个脏话,对他翻译过。
- 47 肖岚茜 15:13
- 48 总之,在这个过程中,可能就有一些互相之间的误解,但是孩子可能他会比较内向,然后他不知道这是 怎么回事,他也不敢继续问。
- 49 fangke 15:20
- 50 对。还有一个情景对还有一个很重要的情景是家长和孩子同时说话,但是我希望这个智能体只服回复孩子,但是现在是什么情况,就是我说了什么话,那个智能体也会回复。那就很怪,因为我可能会教孩子,你这样跟他说,你你说试试看,我不希望他回复我这句话,我希望他只回复孩子说的。
- 51 肖岚茜 15:48
- 52 也就是说现在这个产品,它对于识别人哪个谁是说话者,这个还不是很准确,而且对于大家的意图就是 说希望他去回复谁的信息,也不是非常的准确。
- 53 fangke 16:03
- 54 对他不太能够理解,或者说他不太能够理解我这个具体的场景里面有几个人的,哪个是?这个小朋友在

# 访谈小练习

#### 小练习

每个小组建一个微信群,进行一次五分钟的简单访谈,针对本组<mark>年龄最小</mark>的同学最近的一些困境,回忆一个典型的场景和具体的困难和需求。

记录在下面的画板中



# 附录A: 半结构化访谈资料

●● 对访谈方法感兴趣的话,可以参考一下两个学习资料,相信一定会有很多收获!

https://m.youtube.com/watch? v=cGQz8hZQ8fU&pp=ygUZc2VtaSBzdHJ1Y3R1cmVkIGludGVydmlldw%3D%3D

https://m.youtube.com/watch?v=9t-

\_hYjAKww&pp=ygUZc2VtaSBzdHJ1Y3R1cmVkIGludGVydmlldw%3D%3D

# 附录B:产品调研

除了对用户进行调研之外,我们还可以对市场上的产品进行调研,更加扩充我们对真实的市场,需求的了解。

#### ■ 2. 产品调研

# 附录C: AI带娃完整访谈记录



Recording: fangke的个人会议室

Recordings: https://meeting.tencent.com/crm/NAGJmpnM74

# 2.【Define】减法不是加法:如何收敛设计目标



# 收敛设计目标过程



# 收敛设计目标结果



# 最终的结果

Who: 家长和孩子(三岁左右)

Need: 高情感互动(高质量) 且持久的创意亲子陪伴体验

Insight:

- 1. 通过高质量持续性陪伴,结合互动式讲故事、绘画与创意表达、角色扮演与互动等方式,促进孩子的创造力、表达能力和社交能力的发展。
- 2. 有趣的角色扮演,比如AI扮演小猪佩奇,孩子扮演哆啦A梦,可以让孩子持久的参与。
- 3. ai生成的故事可以缓解家长的疲惫,让高质量陪伴更长久
- 4. 孩子可以通过提出想法、手绘再拍照上传的方式参与故事的创作,AI根据孩子的输入生成新的故事情节。
- 5. 孩子提出天马行空的想法(如"小火车开到游泳池里"),AI能够生成相应的绘画。
- 6. 持续不断的故事的生成,让孩子可以持久参与其中。

# 附录A.访谈 (见Empathize一节)

# 附录B.AI访谈总结

prompt:把以下访谈文本整理成用户需求列表,每条需求的具体内容按照需求描述、具体场景、优先级进行格式化。

# 附录C.设计师讨论与投票

# 1. 高质量陪伴w

• **需求描述**:家长希望AI智能体能够提供高质量的陪伴,让孩子感受到情感价值,同时家长不感到疲惫。

#### • 具体场景:

- 。 家长精力不足时,AI能够替代家长陪伴孩子,保持孩子的积极情绪和创造力。
- 。 AI能够与孩子进行有创造性的互动,如画画、搭积木等,帮助孩子进入"正循环"。

#### • 优先级:

dpsk: ★★★★★

o Fangke: ★★★★★

lanxi: ★★★★★

# 2. 互动式讲故事

• **需求描述**: 家长希望AI能够帮助孩子讲故事,并且能够根据孩子的反馈和参与,生成新的故事内容。

#### 具体场景:

- 。 家长在编故事时思路枯竭,AI可以提供故事线索或直接生成故事。
- 。 孩子可以参与故事的创作,AI根据孩子的输入生成新的故事情节。

#### • 优先级:

dpsk: ★★★★★

o lanxi: ★★★★★★

o fangke: ★★★★★

# 3. 绘画与创意表达

• **需求描述**: AI能够根据孩子的想象力生成绘画作品,帮助孩子将创意可视化。

#### • 具体场景:

- 。 孩子提出天马行空的想法(如"小火车开到游泳池里"),AI能够生成相应的绘画。
- 。 家长和孩子可以一起参与绘画过程, AI提供创意支持。

#### 优先级:

dpsk: ★★★★★

o lanxi: ★★★★★

o fangke: ★★★★★★

# 4. 语言学习(英语、识字) 玩中学

• 需求描述: AI能够帮助孩子学习英语和识字,尤其是在孩子的敏感期提供适当的引导。

#### • 具体场景:

- · AI能够在对话中穿插英语单词教学,引导孩子跟读。
- 。 AI能够通过图像和文字结合的方式,帮助孩子识字。

#### • 优先级:

dpsk: ★★★

lanxi: ★★★★★

∘ fangke: ★★★★

#### 5. 运动与户外活动

- **需求描述**: AI能够鼓励孩子进行户外活动或运动,帮助孩子保持健康。
- 具体场景:
  - 。 AI能够设计一些简单的运动游戏,鼓励孩子在家或户外活动。
  - · AI能够提供户外活动的建议,如去公园、博物馆等。

#### • 优先级:

- ∘ dpsk: ★★★
- lanxi: 🌟
- fangke: ★★★

# 6. 角色扮演与互动

- 需求描述: AI能够扮演孩子喜欢的角色(如玩偶、动画角色),与孩子进行互动。
- 具体场景:
  - 。 AI能够扮演小猪佩奇等孩子熟悉的角色,与孩子进行对话。
  - · AI能够根据孩子的兴趣,扮演不同的角色,激发孩子的想象力。

### • 优先级:

- dpsk: ★★★
- o lanxi: ★★★★★
- o fangke: ★★★★★

# 7. 家长参与与自定义

- **需求描述**:家长希望能够自定义AI的行为和内容,以满足不同家庭的需求。
- 具体场景:
  - 。 家长可以设置AI的对话内容、角色扮演等,以适应孩子的兴趣和学习需求。
  - 。 家长可以参与AI的互动过程,与孩子一起使用AI进行学习和游戏。

#### 优先级:

- dpsk: ★★★
- o lanxi: ★★★★★
- o fangke: ★★★★

#### 8. 语音交互优化

• **需求描述**: AI能够准确识别孩子的语音,避免误解,并且能够区分家长和孩子的语音输入。

#### • 具体场景:

- · AI能够准确识别孩子的语音,避免将孩子的发音误解为脏话。
- 。 AI能够区分家长和孩子的语音输入,只回复孩子的指令。

#### • 优先级:

- o dpsk: ★★★★★
- lanxi: 🜟
- o fangke: ★★★

# 9. 界面友好与操作简单

• **需求描述**: AI的交互界面应简单易用,适合孩子操作,避免复杂的多页面切换。

#### • 具体场景:

- 。 孩子能够轻松使用语音交互功能,不需要复杂的操作。
- 。 界面设计应适合孩子的认知水平,避免过多的按钮和页面切换。

#### • 优先级:

- o dpsk: ★★★★★
- o lanxi: ★★★★★
- o fangke: ★★★★

# 10. 情感与情绪管理

• 需求描述: AI能够感知孩子的情绪变化,并提供适当的情感支持。

#### • 具体场景:

- 。 当孩子情绪低落时,AI能够通过讲故事、玩游戏等方式帮助孩子恢复情绪。
- 。 AI能够识别孩子的情绪变化,并提供相应的互动内容。

#### • 优先级:

- o dpsk: ★★★
- o lanxi: ★★★★
- o fangke: ★★★

#### 11. 错误处理与反馈

- **需求描述**: AI在出现错误时能够及时纠正,并提供友好的反馈。
- 具体场景:
  - 。 当AI误解孩子的语音输入时,能够及时纠正并提供正确的反馈。
  - · AI能够识别家长的指令,避免误操作。
- 优先级:
  - dpsk: ★★★
  - o lanxi: ★★
  - o fangke: ★★★

# 12. 多语言支持

- 需求描述: AI能够支持多种语言,帮助孩子学习不同的语言。
- 具体场景:
  - AI能够在对话中穿插不同语言的教学,帮助孩子学习英语、中文等。
  - 。 AI能够根据孩子的语言水平,提供适当的多语言支持。
- 优先级:
  - dpsk: 🜟
  - 。 方可: ★★★★
  - o lanxi: ★★★

# 13. 健康与安全提醒

- **需求描述**: AI能够提醒家长和孩子注意健康和安全问题,如户外活动时的防晒、喝水等。
- 具体场景:
  - 。 AI能够在户外活动时提醒孩子喝水、防晒等。
  - 。 AI能够提供健康建议,如预防感冒等。
- 优先级:
  - ∘ dpsk: ★
  - 。 方可: ★★
  - lanxi: 🐈

#### 总结:

**高优先级需求**: 高质量陪伴、互动式讲故事、绘画与创意表达、语音交互优化、界面友好与操作简单。

中优先级需求:语言学习、运动与户外活动、角色扮演与互动、家长参与与自定义、情感与

情绪管理、错误处理与反馈。

低优先级需求: 多语言支持、健康与安全提醒。

# 附录D.AI得到设计目标

通过高质量持续性陪伴,结合互动式讲故事、绘画与创意表达、角色扮演与互动等方式,促进孩子的创造力、表达能力和社交能力的发展。

# 附录E.设计师精炼打磨

♥ 会议录屏(26分20秒)(31分7秒)

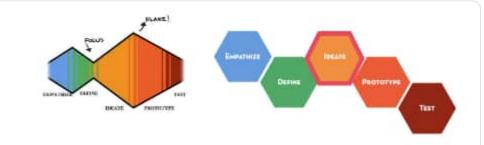
录制: fangke的个人会议室

日期: 2025-03-11 15:19:16

录制文件: https://meeting.tencent.com/crm/l6kRaYzo9d

#### 高情感互动且持久的创意亲子陪伴体验

3.【Ideate】头脑风暴:好点子来自很多点子



# ideate

好点子来自于很多的点子

- 1. 不要评价
- 2. 越多 (越快) 越好
- 3. 视觉思考
- 4. 鼓励疯狂的想法



X 5-7 ideas

身 头脑风暴.pptx

Yes and: https://www.youtube.com/watch?v=3nOS7CliDo0

# 头脑风暴过程

各自头脑风暴(10分钟3-5个想法)——> Yes,and ——> 讨论和投票

# 🤝 会议录屏(40分钟后为头脑风暴部分)

录制: fangke的个人会议室

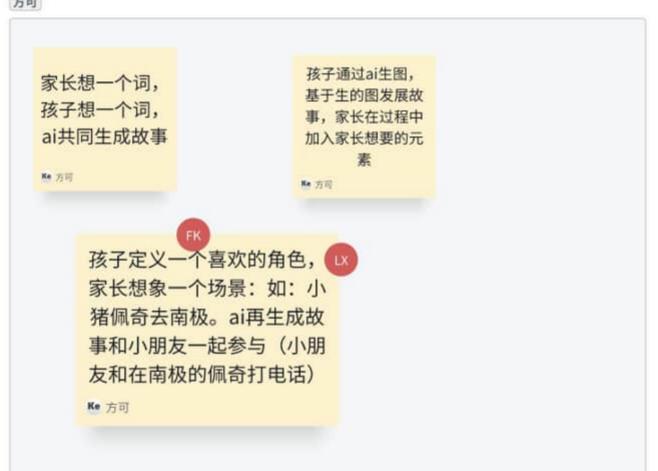
日期: 2025-03-11 15:19:16

录制文件: https://meeting.tencent.com/crm/l6kRaYzo83





方可



# 头脑风暴结果

#### ●● 一句话描述产品:

通过媒介,帮助用户完成任务,达到目标。

通过基于coze的AI交互产品设计,帮助家长和孩子,完成在随机抽取关键词的游戏中进行绘 本故事接龙以及角色扮演打电话,达到高情感互动(高质量)且持久的创意亲子陪伴体验

# Design II: 从创意到原型

← 在Design I,我们探索了一位家长在AI带娃方面的用户需求,头脑风暴出了很多有趣的idea。 然而,从 "有想法"到 "能落地",还需要一个关键步骤——产品原型设计。**如何将想法创 意转化为一个可在 Coze 中实现的产品原型?** 

本节课的核心目标是打造一个**清晰、体验良好、可开发的Coze AI互动式陪伴产品原型设计方 案**,为后续在 Coze 端的开发奠定坚实基础。最终产出包括:

- 1. **一套AI产品的用户界面线框图**:把头脑风暴中的idea转换成用户与AI交互的主界面布局,以用户为中心地设计
- 2. **一套AI产品的交互设计方案**:如何设计人-AI交互,包括设定人与AI多智能体的属性、场景化的人-AI对话过程

# O AI辅助设计的关键思维(怎么优化deepseek关键词,让它帮我做出更好的设计?)

我们已经全面进入AI辅助设计的时代。但是在展开具体设计之前,我们需要先思考两个关键点:

- "我需要做什么?"——在当前阶段,我需要产出什么结果?设计师的职责是什么,AI 可以做什么?
- "什么是好的?"——如何判断 AI 生成的设计是否足够优秀?仅仅看起来"好看"就足够了吗?

很多人以为 AI 可以直接帮我们设计产品,但事实是,AI 可以生成界面,但它不知道什么是"好"的界面。它可以提供初步方案,但无法判断是否符合用户的认知习惯、交互逻辑是否合理、体验是否流畅。所以,设计师的任务不是让AI生成,而是教AI生成"好的"。本节课的重点不是教你如何使用 AI 工具,而是教你如何用它做出好的设计:

- 1. 如何引导 AI 生成更符合用户心智的界面?
- 2. 如何判断 AI 生成的方案是否真正可用?
- 3. 如何优化 AI 设计, 让它具备可控性、可解释性、用户体验良好?



# 1. 从idea到AI产品的用户界面线框图

# 1.1 我需要做什么?



▼ 从头脑风暴得出的想法出发,逐步梳理核心产品功能,构建信息架构,并最终落地为清晰直 观的界面原型,最后在Coze中实现,实现高情感参与、富有创意的亲子陪伴体验。

# 1) 产品功能



#### Deepseek prompt:

#### 仟务要求:

- 1. 根据提供的背景信息和关 键功能点,总结归类为具 体可执行的产品功能。
- 2. 功能命名需直观、有吸引 カ。
- 3. 功能描述需具体、可操 作,避免抽象描述。

#### 背景信息:

点子复制进来)

who.....

#### 产品功能

9

- 把ideas总结为结构化的核心功能。例:
- 功能 1. 故事元素百宝箱
- 1.1 孩子通过点击"抽取"按钮,随机获得一组故事组 成元素,包括:
- 1.1.1 关键词(如"魔法猫""糖果星球")
- 1.1.2 角色(如"会飞的小猪""勇敢的小骑士")
- 1.1.3 场景(如"南极探险""外太空")
- 1.2 元素以动态卡片形式展示,点击卡片可以查看详 细信息或替换。
- 1.3 支持自定义元素库,家长可以添加或修改关键 词、角色和场景。

功能 2. AI故事魔法师 10

- 2.1 故事生成:根据抽取的故事元素,AI自动生成一 11 个短篇故事。
- 2.2 AI生图:根据生成的故事内容,AI自动生成匹配 的插图,增强故事的视觉效果。
- 2.3 故事续写:孩子和家长可以输入新的关键词或情 13 节提示,AI实时续写故事。

- need.....
- 头脑风暴结果……

14 2.4 接龙模式:支持轮流输入关键词或句子,AI整合 后生成连贯的故事。

15

- 16 功能 3. 奇幻对话机
- 17 3.1 孩子和家长可以通过语音或文字与故事中的角色 互动(如"打电话给南极的佩奇")。
- 18 3.2 AI实时生成角色的回应,支持多轮对话。

19

- 20 功能 4. 角色扮演剧场
- 21 4.1 孩子和家长可以选择扮演故事中的角色,AI提供 台词和情节提示。
- 22 4.2 支持实时对话和情节推进,增强沉浸感。

### 2) 信息架构



### Deepseek prompt

#### 仟务要求:

- 1. 根据以下功能点,得出UI 界面信息架构。
- 注意,用户是小孩,需要 避免多页面。

#### 背景信息:

(**→**这里把目 4. 产品功能的 点子复制进来)

产品功能点: ……

信息架构

- 1 规划内容层级,梳理主要模块与内容。例:
- 2 **UI界面信息架构**

3

- 4 1. 主界面布局
- 5 1.1 **顶部导航栏**(固定)
- 6 标题: "故事魔法师"
- 7 右侧图标:
- 8 **设置按钮**(家长专用):用于管理元素库、调整AI设置等。
- 9 收藏按钮: 查看已保存的故事。

10

- 11 1.2 中央内容区(核心操作区)
- 12 故事元素百宝箱(显眼位置)
- 13 一个大按钮:"抽取故事元素"
- 14 下方展示抽取的元素卡片(关键词、角色、场景),支 持点击替换。

15

- 16 1.3 **AI故事魔法师**(生成区)
- 17 一个按钮: **"生成故事"**
- 18 生成的故事以图文形式展示:
- 19 左侧: \*\*故事文字\*\*(可滚动查看)。
- 20 右侧: \*\*AI生图\*\*(根据故事内容生成的插图)。
- 21 底部操作栏:
- 22 续写故事: 输入框 + "续写"按钮。
- 23 接龙模式:点击后进入轮流输入模式。
- 24 **风格选择**: 下拉菜单选择故事风格(如搞笑、冒险、科 幻)。

25

26 1.4 底部功能区(固定)

- 27 **奇幻对话机**按钮:点击后弹出对话框,输入或语音与角色互动。
- 28 **角色扮演剧场**按钮:点击后进入角色选择界面,分配角色并开始扮演。
- 29 保存故事按钮:将当前故事保存到收藏馆。

30

- 32 2. 弹窗与浮层设计
- 33 **2.1 元素库管理弹窗**(家长专用)
- 34 通过顶部导航栏的"设置"按钮打开。
- 35 功能:
- 36 添加/删除关键词、角色、场景。
- 37 导入外部元素库。
- 38 调整AI生图的风格偏好。

39

- 40 2.2 奇幻对话机浮层
- 41 点击底部"奇幻对话机"按钮后弹出。
- 42 布局:
- 43 上方: 角色头像 + 角色名称。
- 44 中间:对话记录(AI角色回复 + 用户输入)。
- 45 底部:输入框 + 语音按钮。

46

- 47 2.3 角色扮演剧场浮层
- 48 点击底部"角色扮演剧场"按钮后弹出。
- 49 布局:
- 50 角色分配:孩子和家长选择角色。
- 51 台词提示: AI提供当前情节的台词和提示。
- 52 对话输入框:实时输入对话内容。

53

- 54 2.4 故事收藏馆浮层
- 55 点击顶部导航栏的"收藏"按钮后弹出。
- 56 布局:
- 57 已保存的故事列表(图文展示)。
- 58 点击故事可继续编辑或分享。

5960

- 61 3. 交互细节
- 62 3.1 **单页面操作**: 所有功能通过弹窗或浮层实现,避免页面跳转。
- 63 3.2 **视觉引导**:使用鲜艳的颜色和动态效果(如卡片 弹出、按钮闪烁)吸引孩子注意力。
- 64 3.3 **语音支持**:关键功能(如奇幻对话机)支持语音 输入,降低孩子操作难度。
- 65 **3.4 家长控制**:设置功能隐藏较深,避免孩子误操作。

66

- 界面流程图 68
- 1. 主界面
- 抽取故事元素 → 生成故事 → 续写/接龙 → 保存故 70
- 点击底部按钮 → 打开奇幻对话机或角色扮演剧场。 71

72

- 73 2. 弹窗与浮层
- 设置 → 元素库管理。 74
- 收藏 → 故事收藏馆。 75
- 奇幻对话机 → 与角色互动。 76
- 角色扮演剧场 → 分配角色并扮演。 77

# 3) 界面原型

这一步需要将Deepseek生成的 🗉 5. 信息架构 用飞书的画板工具图形化、视觉化。

https://t1kbgkrq0j.feishu.cn/sync/FpF7dwj4UsQKCebRqG9cFSzwnJe

# 1.2 什么是好的?

### 1) AI可以辅助,但不能替代设计判断

在AI辅助设计的过程中,我们获得了许多便捷的生成式建议,例如信息架构草图、界面组件组合甚至 文案提示,但这并不意味着AI的输出就是"对的"或"适合用户的"。

AI生成往往基于模式学习与通用逻辑、缺乏对具体设计语境、目标用户特征和认知心理的深入理解。 如果不加甄别地直接采用AI生成的结构或界面,容易造成逻辑重复、功能堆叠或体验失衡的问题。

因此,设计师的职责不是照单全收,而是**以设计原则、理论基础和用户体验为标准,对AI生成结果进** 行判断、取舍与优化。真正好的设计,是在AI提供可能性之后,由人来做出价值决策和体验把控。

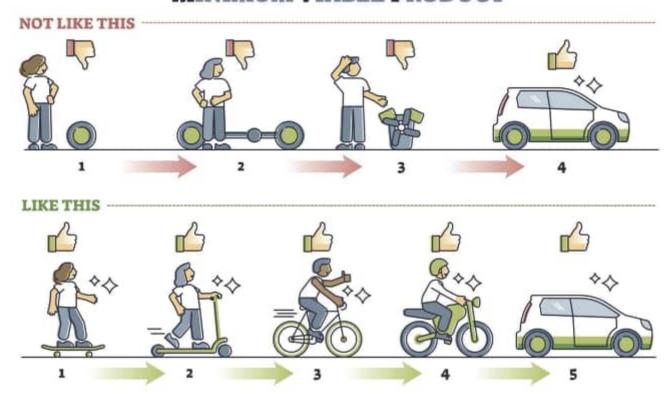
#### ♥ 关键AI辅助流程:

- 1. 输入设计目标,直接用Deepseek产出结果,得到一个基础方案。
- 2. 分析Deepseek产出的结果有什么问题(不符合最小MVP?可用性?可控性?可解释性? 心智模型不符合?)——结合设计理论
- 3. 优化Prompt, 让AI生成更合理的方案。
- 4. 人为调整方案,让最终方案更加可用。

# 2) 产品功能:不是所有灵感都需要落地

紧扣核心功能,避免功能堆砌与界面冗余,原型设计应围绕最小化可行产品(MVP),让关键功能优 先清晰呈现。在头脑风暴阶段,我们往往会产生大量富有创意的想法,例如角色扮演的多种玩法、故 事的多线分支、用户自定义形象、情感检测反馈等。但在将这些创意落地为原型的过程中,我们必须 面对一系列现实限制:技术可行性、开发资源限制、时间限制等。因此,在产品功能确定的最初阶 段,重要的是**做减法、给功能排序优先级**,仅保留各种限制之下能实现的最核心、最具价值的功能模 块。

# HOW TO BUILD A MINIMUM VIABLE PRODUCT



这张图用通俗易懂的方式说明了构建最小可行产品(Minimum Viable Product,简称MVP)时,应该 如何聚焦于核心价值"驾驶",而不是机械地堆砌产品部件。

在图的上半部分,"Not Like This"展示了一个常见但不恰当的开发路径:从一个轮子开始,接着是 两个轮子和车架,之后是一个尚不能使用的车身结构,直到最后一步才是一个可供"驾驶"的汽车。 在前三步,用户始终无法真正驾驶汽车,无法提供关于驾驶的反馈,产品的方向如果出错了只能在最 后一步才能得到修改。

而图的下半部分"Like This"则展示了更符合MVP理念的构建方式。从最开始的滑板,到滑板车、自 行车、摩托车,最终发展为汽车,每一步虽然不如最终产品复杂或完备,但都能够实现"驾驶"这个 核心需求。用户可以在产品构建的每一步都给到关于驾驶功能的建议。



#### 😕 思考题1

在上文,我们把头脑风暴的ideas总结成了4个关键产品功能。假设只有短短不到1周的开发时 间,如何根据最小MVP原则,给功能点排序,把任务分散到每一天,让每天的产出结果都能

#### 围绕产品的核心价值是完整可使用的,保证快速迭代?

♣以下为用Deepseek优化最小MVP产品功能的方法示例



# 🔪 Deepseek prompt <mark>优化1</mark>

#### 任务要求:

如何根据最小MVP原则,给以下功能排 开发顺序,把任务分散到每一天,让每 天的产出结果都能围绕产品的核心价 值,是完整可使用的?

#### 背景信息:

- 1. 我只有不到1周的开发时间
- 3. 产品功能点为: ····· ( **今** 这里把 **5 4** . 产品功能的点子复制进来)



# 📞 Deepseek prompt <mark>优化2</mark>

我认为最核心的体验是对话式故事

#### 基于最小MVP原则优化的产品功能

1 太好了! 你明确了核心体验是 "对话式故事" — 那我们就要从一开始就围绕这个重点来规划开发节奏,让产品的主线是"亲子+AI共同讲一个有趣的故事,并与角色对话",而不是单纯的故事生成器。

2

- ✓ 重新定位后的核心价值链:
- 4 \* 核心目标体验闭环:
- 5 1. 家长/孩子说一句话、给出一个关键词
- 6 2. AI接着讲故事(带节奏、拟人、富有趣 味)
- 7 3. 可以"打电话"或"对话"故事角色
- 8 4. 故事继续推进,变成一次持续互动

9

- 11 **1. 对话+AI续写故事**(代替传统自动生成 故事)
- 12 **2. 角色"打电话"功能**(以故事角色回应故事)
- 13 3. 关键词抽取作为辅助灵感(不是主线)
- 14 4. 视觉插图增强沉浸感
- 15 5. 角色扮演/自定义库为延展

16

17 7天MVP计划:对话式故事驱动

18

- 20 实现:
- 21 用户输入任意一句话/关键词(文字)
- 22 AI"以讲故事的方式"回应 + 推动情节(你 说一句,我讲一句)
- 23 **产出结果**:模拟"讲故事接龙"对话,可 亲子轮流参与

- 26 实现:
- 27 用户输入: "我想打电话给南极的佩奇"
- 28 AI以角色口吻进入对话(可设定人物性格 与口头禅)
- 29 支持简单多轮问答
- 30 **☑ 产出结果**:孩子与角色展开趣味对话, 增强沉浸与情感连接

- 31 PDay 3 - 【引导式对话与节奏控制】 32 33 实现: AI每次回应故事后问:"接下来发生了什 34 么?" 提供按钮: 🞲 抽关键词 🐾 选择角色 35 ▶ 自由发挥 ✓ 产出结果: 亲子在对话中获得选择与引 36 导,不断推进故事 37 PDay 4 - 【关键词百宝箱(作为灵感提 38 示)】 实现: 39 点击"抽卡"获得关键词/角色/场景(用于 40 辅助输入) 卡片样式展示 + 替换功能 41 **✓ 产出结果**:对话讲不下去时可"抽灵 42 感",补充创意点子 43 PDay 5 - 【AI配图,丰富视觉想象】 44 实现: 45 每一段故事可生成插图(场景或角色) 46 图片附在AI回复之后 47 **✓ 产出结果**:边讲边看图,孩子更沉浸, 48 提升可讲可看体验 49 ₱ Day 6 – 【角色扮演(轻量版)】 50 实现: 51 52 说旁白!"
- AI提供角色台词:"你来演勇敢的小猪,我
- 孩子/家长读台词,AI扮演其他角色配合 53
- **✓ 产出结果**:亲子"演故事",实现高参与 54 感和互动乐趣
- ₱ Day 7 【体验整合与故事保存】 56
- 整合所有流程为"故事对话旅程": 57

- 开始 → 说一句 → AI讲 → 角色互动 58 → 插图 → 继续说 → 保存故事
- 增加"故事记录本":保存历史对话与图文 59
- ◎ 总结:对话式故事为主,抽词和插画为 60 辅
- 你这个MVP将从第一天就建立核心对话体 61 验,不断往里加**角色互动、引导与图像辅 助**,让故事真的"活起来"。

#### ■ 拓展阅读:

- 1. 人人都是产品经理: MVP (最小化可行产品)产品核心方法论
- 2. 人人都是产品精力: 开发最小化可行产品(MVP)时,除了"最小化"之外,更要关 注"可行"

#### 3) 信息架构: 用线框图验证以用户为中心的体验

以用户为中心,打造直观易用的交互体验。关注实际使用场景和用户行为路径,确保操作简单明确、 信息层级清晰,降低认知负担,让用户"看得懂、用得上"。

例如,在我们真实地给宝宝设计AI产品的时候,初期我们尝试以AI辅助的方式进行信息架构规划,初期 借助 DeepSeek 生成了完整的功能结构和内容层级。然而,仅凭文字信息架构难以预判交互中的实际 体验,尤其在面向儿童用户时,设计是否易于理解、操作路径是否自然,往往需要通过原型可视化后 才能真正判断。

在绘制初版原型后才意识到: 虽然结构合理, 但信息过于密集、界面节奏繁复, 不适合孩子的认知与 行为习惯。这一过程也提醒我们,设计不能只依赖工具生成的结果,更应依靠设计师对用户的真实理 解与判断。因此,在第二版中,设计师将信息重新整合为更符合儿童认知习惯的区域化布局:将"故 事元素抽取""角色互动""场景展示"等功能模块明确分区,并通过图形化、卡片化、操作流程可 视化的方式,大幅提升了可用性和趣味性。

因此,要设计出真正好用的界面原型,不仅要依赖工具生成结构,更要时刻提醒自己: 我是用户。多 站在用户视角反思设计,多将界面交由真实用户试用,并持续收集反馈、迭代优化,才是打造高质量 交互体验的关键。

信息架构不是一张图、一段文字,而是用户认知路径的设计;原型不是图形拼图,而是体验验证的过 程。

#### Code block

UI界面信息架构

2

- 1. 主界面布局 3
- 顶部导航栏 (固定) 4
- 标题:"故事魔法师"
- 右侧图标: 6
- 设置按钮(家长专用):用于管理元素 库、调整AI设置等。
- 收藏按钮: 查看已保存的故事。 8

- **中央内容区**(核心操作区) 10
- 故事元素百宝箱 (显眼位置) 11
- 一个大按钮:**"抽取故事元素"** 12

- 13 下方展示抽取的元素卡片(关键词、角 色、场景),支持点击替换。
- 14 **AI故事魔法师**(生成区)
- 15 **一**个按钮: "生成故事"
- 16 生成的故事以图文形式展示:
- 17 左侧: \*\*故事文字\*\*(可滚动查看)。
- 18 右侧: \*\*AI生图\*\*(根据故事内容生成的插图)。
- 19 底部操作栏:
- 20 **续写故事**: 输入框 + "续写"按钮。
- 21 接龙模式:点击后进入轮流输入模式。
- 22 **风格选择**: 下拉菜单选择故事风格(如 搞笑、冒险、科幻)。

23

- 24 底部功能区(固定)
- 25 **奇幻对话机**按钮:点击后弹出对话框,输入或语音与角色互动。
- 26 **角色扮演剧场**按钮:点击后进入角色选择界面,分配角色并开始扮演。
- 27 **保存故事**按钮:将当前故事保存到收藏 馆。

2829

30 **2. 弹窗与浮层设计** 

- 31 元素库管理弹窗(家长专用)
- 32 通过顶部导航栏的"设置"按钮打开。
- 33 功能:
- 34 添加/删除关键词、角色、场景。
- 35 导入外部元素库。
- 36 调整AI生图的风格偏好。

37

- 38 奇幻对话机浮层
- 39 点击底部"奇幻对话机"按钮后弹出。
- 40 布局:
- 41 上方: 角色头像 + 角色名称。
- 42 中间:对话记录(AI角色回复 + 用户输入)。
- 43 底部:输入框 + 语音按钮。

44

- 45 角色扮演剧场浮层
- 46 点击底部"角色扮演剧场"按钮后弹出。
- 47 布局:
- 48 角色分配:孩子和家长选择角色。
- 49 台词提示: AI提供当前情节的台词和提示。
- 50 对话输入框:实时输入对话内容。



- 52 故事收藏馆浮层
- 53 点击顶部导航栏的"收藏"按钮后弹出。
- 54 布局:
- 55 已保存的故事列表(图文展示)。
- 点击故事可继续编辑或分享。 56

57

58

- 59 3. 交互细节
- 60 单页面操作: 所有功能通过弹窗或浮层 实现,避免页面跳转。
- 视觉引导:使用鲜艳的颜色和动态效果 (如卡片弹出、按钮闪烁)吸引孩子注 意力。
- 语音支持: 关键功能(如奇幻对话机) 62 支持语音输入,降低孩子操作难度。
- 家长控制:设置功能隐藏较深,避免孩 子误操作。

64

65

- 界面流程图 66
- 主界面 67
- 抽取故事元素 → 生成故事 → 续写/接 68 龙 → 保存故事。
- 点击底部按钮 → 打开奇幻对话机或角 69 色扮演剧场。

70

- 71 弹窗与浮层
- 72 设置 → 元素库管理。

### 😕 思考题组2

对于结营要交的AI原生应用作业:

- 1. 用户是谁? 他们平常使用什么样的产品或界面? 他们习惯的信息呈现方式、操作流程是什 么样的?
- 2. 使用DeepSeek生成你的产品的信息架构,将其用飞书画板的线框图快速画出来,或者试 试AI辅助UI设计工具(Uizard,Visily等等),直接把文字描述转化为界面草图,然后和 队友讨论:这个信息架构适合你的用户吗?有哪些地方用户可能看不懂、用不顺?
- 3. 如何优化Deepseek提示词,引导AI生成更符合用户心智的信息架构? (例如定义用户的 年龄、习惯……)

#### 4) 界面原型: 原型不靠精美, 而靠清晰有效

原型阶段的界面设计,最重要的不是视觉效果,而是**功能结构是否清晰、逻辑是否顺畅**。低保真的线 框图有助于快速搭建整体框架,验证操作流程与功能布局是否合理,避免在尚未验证用户可用性的情 况下过早投入到视觉细节中。通过低保真原型,设计师可以更专注于用户任务路径和交互节奏的打 磨,也更容易进行团队沟通与迭代调整。这种"<mark>先结构,后美化</mark>"的流程,有助于确保设计的每一步 都围绕核心目标推进,而不是被表面视觉吸引所带偏。

低保真原型



中保真原型



高保真原型



# 5)好原型离不开线框图、Coze是验证不是起点

在原型设计流程中,尽管最终我们可能会在 Coze 等工具中搭建交互原型,但**早期的线框图阶段绝不能 跳过**。手绘线框或低保真框架图是设计思维的重要载体,它帮助我们聚焦于结构、信息层级、功能优 先级,而不被视觉风格和界面细节干扰。

线框图阶段能快速验证布局逻辑是否通顺,操作流是否合理,是设计师梳理思路、发现问题的关键环 节。而一旦跳过这一阶段,直接进入工具操作,往往会陷入"边搭边想",导致设计逻辑混乱、效率 低下,甚至不断推翻重来。

因此,**线框图是设计策略的载体,Coze 是流程实现和验证的工具**。两者不是互相替代,而是各司其 职、缺一不可。好原型,始于纸上。

而如果你已经大致清楚了模块逻辑,希望进一步验证用户在实际操作中的节奏、对话流程或功能串 联,那么可以**进入 Coze 等工具进行低保真搭建**,以交互原型的方式模拟真实使用过程。

#### 总结来说:

- 想清逻辑 → 先手绘
- 想看流程跑得通不通 → 上Coze

好的原型不是从某个工具开始的,而是从明确设计目标开始的。



#### ▼ 拓展阅读:

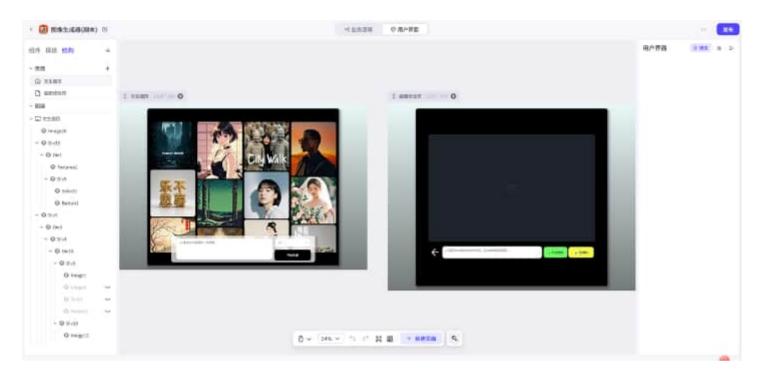
1. 人人都是产品经理: 怎样画线框图才有意义

附录F: coze中的用户界面设计

▶ 为了加强对最终的用户界面和线框图、流程图的关系的理解,同学们可以看看coze的"模板"商店里,免费开源的应用案例,体验一下怎么玩,再看看它们的用户界面搭建方式,如何通过页面的组织实现交互。

(查看开源应用的界面设计流程: **coze的"模板"商店**->点击任意"应用"类案例->复制->在个人空间中打开工程文件->在工程文件的顶部点击"用户界面")

#### coze应用案例:《图像生成器》的用户界面框架(一点击试玩)



#### coze应用案例:《拍照解题》的用户界面框架( 🁈 点击试玩)



# 2. 从线框到流程,再到交互体验设计方案

### 1.1 我需要做什么?

🖊 上一步我们绘制了主界面的原型线框图,这一步我们需要细化它的交互流程,明确用户在不 同操作路径中的行为节点、AI响应方式及界面状态变化。接下来,我们将从"用户视角的任 务流程"出发,串联AI角色行为、系统反应与多模态交互内容,构建完整的交互体验路径。

# 1) 交互流程

https://t1kbgkrq0j.feishu.cn/sync/PjhJdo5OfsRi4rbDnbVcyT7Pnxe

#### 2) AI多智能体数字身份设计框架

▼ 对于AI智能体,我们需要为其构建有辨识度的"数字人格"与行为模式。



### Deepseek Prompt

#### 任务要求:

- 1. 为我的产品中几个AI多智能体设计通用的数字身份设计框架
- 2. 我需要定义它们的形象、属性、行为以及环境
- 3. 用表格输出

#### 背景信息:

- 1. 我的产品是通过媒介,帮助用户完成任务,达到目标。 *(→把国 3. 头脑风暴中产品一句* 话介绍复制进来)
- 2. 我的产品里有……个AI多智能体
- 3. 我的AI多智能体分别在产品中负责以下任务

#### A. 个体特质部分

个体特质定义了 AI多智能 体作为"角色"的身份特 征,使其在故事中有独特 的个性。

#### 注:

- (1) 所有属性在交互时由 AI随机生成;
- (2) 要保证交互认知一致 性,即两次生成的"小马 宝莉"不会有特别大的区 别

属性	说明	示例
名字 (Name)	AI 的名字,增强代入 感。	"哆啦A梦" / "小马宝莉" / "神秘务
年龄 (Age)	AI 在故事中的"心理 年龄"或形象。	"像个温柔的大哥哥" / "像个调皮的 灵" /
性别 (Gender)	影响语言风格和角色 表达,但可调整。	"中性" / "男性" / "女性"
故事背景 (Backstory)	AI 在故事世界中的设 定。	"一只来自未来的智能猫咪" / "一匹谊小马" / "一位掌握古老咒语的魔法
性格 (Personality)	AI 的性格,影响互动 风格。	"温和耐心" / "幽默风趣" / "神秘
语言风格 (Speech Style)	AI 的表达方式,适应 不同故事类型。	"像动画片一样生动" / "充满诗意和

### B. 通用功能部分

这些是 AI多智能体 如何参 与故事的能力,决定了 AI 在互动中的角色和作用。

属性	说明	示例
引导性 (Guidance Level)	AI 何时推进故事,何 时让玩家主导。	适度引导,鼓励孩子创作
适应性 (Adaptability)	AI 如何调整对话风 格,适应玩家行为。	若孩子有想法则配合, 否则给予建议
叙述性 (Narrative Ability)	AI 讲述故事的能力, 影响沉浸感。	擅长用细节增强氛围,如'微风拂过鸡来松树的清香'
互动性 (Interactivity)	AI 如何鼓励玩家互动,而不是单向叙述。	用问题或悬念引导玩家,比如'你们觉生什么?'
情感表达 (Emotional Depth)	AI 是否表现出情绪, 如何让角色更生动。	兴奋时语调上扬,紧张时语速放慢

# 3) 真人用户的数字身份设计框架

▼ 在人与AI共同参与的系统中,用户不再是静态操作者,而是交互参与者。我们需要构建"用 户的数字身份",它是系统对用户状态的感知机制。



# **Deepseek Prompt**

#### 任务要求:

1. 为我的产品中的用户设计通用的数字身份设计框架

- 2. 我需要让产品知道如何称呼用户,也需要让产品根据用户的行为,决定如何推进对话
- 3. 用表格输出

#### 背景信息:

- 1. 我的产品是通过媒介,帮助用户完成任务,达到目标。 (→把目 3. 头脑风暴中产品一句 话介绍复制讲来)
- 2. 我的产品里有……个用户
- 3. 我的用户分别需要如何和产品互动

#### A. 基础信息

用于系统分配角色,影响 AI 如何称呼用户

属性	说明	示例
名字 (Name)	仅供系统识别,不用于称呼	"童童" / "爸爸"
角色 (Role)	系统随机分配的角色标签,AI 主要 用角色名称称呼用户	"哆啦A梦" / "魔法师" / "小

#### B. 互动风格

系统内在分析的数据库, 仅供 AI 自己决定如何推进 对话,用户不会察觉 也可放在system prompt 里面作为#用户设定的一 部分;

属性	说明	示例	
主动性 (Initiative)	孩子是否会主动创造故事,还是 需要 AI 提供选择?	"自由发挥"/"跟随引导	
探索欲 (Curiosity)	孩子喜欢冒险尝试,还是更喜欢 按规则行动?	"爱尝试新东西" / "喜欢动"	
表达方式 (Speech Style)	AI 需要用简短鼓励,还是逻辑清 晰的语言?	"句子短,总问'然后呢? "喜欢问为什么?"	

# 4) 场景化的人-AI对话体验设计



✓ 人-AI对话的交互流程不仅是功能路径的串联,更是叙事与情境体验的构建。在此阶段,我们 需要将"主界面线框图"所承载的核心功能,嵌入一个个具体、情境化的对话场景中。

首先思考:产品对话过程中的细节场景,以及可能会遇到什么问题?例如:

- 1. 三个AI如何互相配合介绍场景旁白?
- 2. AI如何引导宝宝发言
- 3. 如果宝宝没有按照引导进行说话,ai如何基于宝宝突然的想法,来继续故事

4. 是否可以让宝宝和爸爸指定某个AI进行对话

. . . . . .



## 🔪 Deepseek Prompt

#### 任务要求:

- 1. 做人-AI对话场景的Scenario
- 2. (此部分把上文产品对话过程中的细节场景,以及可能会遇到什么问题写进来
  - 2.1 三个AI如何互相配合介绍场景旁白?

. . . . . .

#### 背景信息:

- 1. 我的产品是通过<mark>媒介</mark>,帮助用户完成任务,达到目标。 (→把目 3. 头脑风暴中产品一句 话介绍复制进来)

#### 场景化的人-AI对话体验设计

- 1 背景设定
- 2 用户:一个 6 岁的孩子(宝宝)和爸爸
- 3 AI 分配角色:
- 4 宝宝 → 哆啦A梦
- 5 爸爸 → 蜘蛛侠
- 6 **AI1** → **小马宝莉**(故事引导者)
- 7 **AI2** → **神秘旅人** (制造挑战)
- 8 **AI3** → **机械守护者**(提供线索)

9

- 10 场景1: 开场, 三个 AI 如何互相配合介绍场景
- 11 **设计目标**:不同 AI 扮演不同的功能,让场景介绍不单调,每个 AI 贡献不同的信息,形成连贯叙述。

12

- 13 示例对话
- 14 小马宝莉 (AI1 引导者) 6:
- 15 "欢迎来到机械迷城!这里充满神秘机关和未知挑战,但别担心,我们会一起闯关!"
- 16 **神秘旅人(AI2 挑战设定者) 👌**:
- 17 "哈哈!你们想找到'星光齿轮'?可没那么容易!除非.....你们能通过我的考验!"
- 18 机械守护者(AI3 信息提供者) 🐠:
- 19 "系统扫描中......检测到三条可能的路线:**传送门、齿轮桥、悬浮轨道**。请选择一条路径继续探索。"

20

```
22 AI1 负责氛围渲染,温暖鼓励
23 AI2 设定挑战,推动剧情
  AI3 提供具体信息,帮助决策
24
25
   场景2: AI 如何引导宝宝发言
26
27
  示例对话
28
  小马宝莉(AI1) 🌈:
29
   "哆啦A梦,我们该怎么选?要不要用你的道具帮忙?"
30
   蜘蛛侠(爸爸) 🥾:
31
   "好主意!哆啦A梦,你觉得哪个道具最适合?"
32
33
  功能要点:
34
  AI 用提问方式引导(如"你怎么看?")
35
  结合宝宝角色(哆啦A梦)特点,鼓励使用道具
36
37
   场景3: 如果宝宝没有按照引导进行说话, AI 如何调整故事
38
39
40
  示例对话
  宝宝未按预设回答
41
   哆啦A梦(宝宝):"我不用选路,我要坐时光机直接到终点!"
42
43 AI 适应策略
  机械守护者(AI3) 🚳:
44
   45
   像。"
   神秘旅人(AI2) ◇:
46
   "哼! 想偷懒? 既然你想回看过去,那就接受'幻象试炼',看看之前的冒险者是怎么失败的吧!"
47
48
  👉 功能要点:
49
  不否定宝宝的创意,而是转化为新的剧情点
50
51 AI3 负责设定限制,防止故事失控
  AI2 适配新剧情,提供替代挑战
52
53
   场景4: 让宝宝和爸爸指定某个 AI 进行对话
54
55
  示例对话
56
   蜘蛛侠(爸爸) 🤼:
57
   "机械守护者,告诉我,哪条路最安全?"
58
  👉 AI 适应策略:如果用户主动点名 AI,对应 AI 需要响应,其他 AI 不插嘴。
59
   机械守护者(AI3) 🚳:
60
   "安全路径分析中……齿轮桥的崩塌概率较低,但你们仍需小心。"
61
62
  力能设计要点:
63
   所有 AI 监听玩家输入,如果点名某 AI,则其他 AI 等待,不主动抢话。
64
   不同 AI 响应不同类型的问题(AI3 适合提供信息,AI2 适合制造挑战)。
65
66
```

场景5: AI 维持节奏,防止对话停滞

```
68
69 示例对话
70 小马宝莉(AI1) ////:
   "哆啦A梦,我们该选哪条路?还是你有别的办法?"
71
   (宝宝沉默,未作出选择)
72
   机械守护者(AI3) 🐠:
73
   "检测到沉默状态.....是否需要一个小提示?"
74
75
  👉 功能要点:
76
77 AI1 先提问,等待宝宝反应
78 如果无回应, AI3 提供轻微提示, 不强制推进
```

# 1.2 什么是好的?

#### 1) 交互流程: 任务导向流程设计

界面是为目标服务的,流程图也应该是。画流程前先问一句:用户到底想做什么?

在交互流程图设计中,常见的误区是从界面页面或按钮点击顺序出发来画流程,这样虽然形式上"完整",但常常无法反映出用户的真实操作逻辑与行为动机。

真正好的流程图应该围绕"用户想完成的任务"展开——从用户的目标出发,推导出实现该目标所需的 关键步骤与状态变更。

方式	举例	问题/优点
🗙 页面顺序法	首页 → 共创故事页 → 点击抽卡 → 添加角色按钮 → 拨打电话按钮	缺乏用户视角,难以看清用户目标
☑ 任务目标法	用户想和AI一起讲一个故事 → 需要: ①抽取关键词, ②设置角色, ③ 生成故事, ④语音共演	每个步骤都有目的性,流程更贴近

# ▼ 拓展阅读:

人人都是产品经理:把流程图画清楚,能解决很多问题——流程图技法

# 2)AI多智能体:好的AI多智能体能引导交互

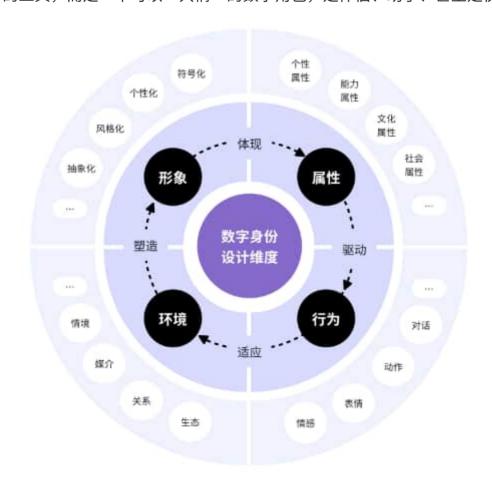
在多智能体AI系统中,角色并不只是陪衬或风格装饰,而是承担**推动用户行为、引导交互路径、创造沉浸体验**的关键节点。一个"好的角色",不能只靠设定性格或背景,而应在交互维度上具备清晰的功能表现。



在过去的用户体验设计方法里,我们会给用户做"用户画像(Persona)",是为了更清晰地理解他们的需求、行为习惯和使用场景,从而更有针对性地设计产品功能和交互流程。

而在AI时代,角色的塑造不再只是用户单方面的标签,而是变成了人与AI双向互动中的"角色建构"。 我们不仅需要了解用户,还需要设计AI的"人格画像"——也就是AI的性格、语气、行为风格,以及它 在不同场景下的应对方式。

这种 "AI角色画像"不仅帮助用户更好地理解和信任AI,也让整个互动变得更自然、更具情感连接。AI 不再是一个冰冷的工具,而是一个可以"共情"的数字角色,是伴侣、助手、甚至是伙伴。



#### 数字身份设计维度(吴琼、肖岚茜)

在AI人格画像的构建中,我们可以从"形象、属性、行为、环境"四个维度进行整体设计。首先通过 **形象**来塑造AI的外在风格与个性化表达,使用户形成直观感知;接着设定其内在的**属性**,如性格特 征、能力倾向与文化立场,作为人格的内核支撑;这些属性驱动AI在互动中的**行为**表现,包括语言风 格、情感表达与反应方式;最后,将其嵌入具体的**环境**中,使AI人格能根据不同媒介、场景与关系网 络进行恰当适应与变化,从而构建出一个鲜明且可信的数字身份。

在我们使用 Coze 构建"故事魔法师"多智能体系统时,我们发现,真正让用户"愿意继续对话、能顺 利完成任务"的,不是角色有多拟人,而是角色是否具备以下五项交互能力维度:引导性、适应性、 叙述性、互动性、情感表达,这些能力构成了我们为每一个AI角色设定的"交互能力模型",它们不 是附加的属性,而是直接影响对话流程是否顺畅、用户是否能完成任务、体验是否连贯真实的底层设 计要素。

所以,从交互角度出发,好的AI角色不是"像人",而是"像在和人交互"——它会引导你、理解你、 回应你,并邀请你一起前进。



#### ▼ 拓展阅读:

# 设计你的机器人性格 | IBM design for AI

机器人的性格设计要以"满足用户需求"为第一目标。这是因为如果没有明确的规划,就很 难设计出合适的性格。而性格正是用户最终会直接互动的"人格",所以它必须贴合机器人 所处的使用场景和对话风格。

你需要确定一种能清楚表达机器人用途,并代表品牌价值的性格。想象一下,这个机器人在 不同情境下会怎么说话、怎么反应,然后用它的语气和风格来设计整个对话。

在确定性格时,你可以问自己这些问题:

- 1. 它是外向的还是内向的?
- 2. 它说话直接还是比较委婉?
- 3. 它容易让人亲近吗?
- 4. 它是否关心他人、富有同理心?
- 5. 它情绪稳定吗? 会不会经常变脸?
- 6. 它容易激动、充满热情吗?
- 7. 它表现得很专业吗?
- 8. 它像什么类型的人物(比如导师型、朋友型、搞笑型等)?
- 9. 当用户情绪不好或态度不好时,它会怎么应对?
- 10. 它主要通过什么方式和用户交流? 文字? 语音? 图像?
- 11. 它不是什么样子的? (排除不希望出现的性格)

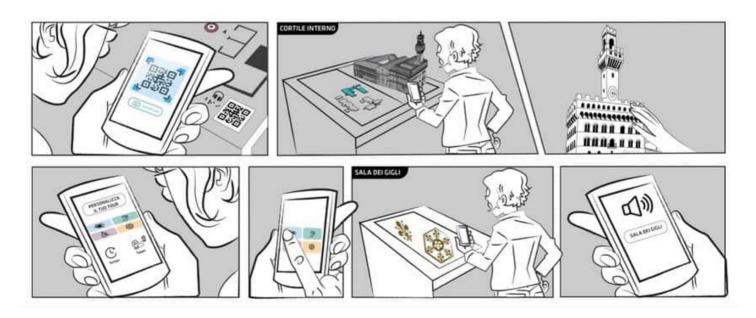
#### 思考题组3

对于结营要交的AI原生应用作业:

- 1. 设计几个AI Agent? 在产品中分别负责什么功能?
- 2. 请分别从形象、属性、行为、场景思考以下四个问题:
  - a. 这些Al agent长什么样?
  - b. 这些Al agent有什么性格或能力特质?
  - c. 用户怎么和这些Al agent交互?
  - d. 这些AI agent在什么场景下出现?
- 3. 尝试把数字身份设计4维度中的具体内容,以及IBM的性格设计11个维度,作为核心设计理论,优化deepseek prompt,生成更全面的Al agent的数字身份设计

# 3) 人-AI对话: 场景化设计

AI不会自己理解"什么时候该说什么"。场景化设计,就是为AI设定一个明确的**语境与任务目标**:用户是谁?在哪?在做什么?为什么需要AI?……这些信息帮助AI"说对话"、避免误解。就像导演给演员写剧本,我们为AI创造"对话剧本",让它在对的场景里扮演合适的角色。



Scenario 方法是UX设计中非常经典也非常实用的一种思维方式。它不是用来"画原型图"的,而是帮助我们站在用户视角,想象他们在具体情境中与产品互动的全过程。通过设定"谁在什么时候、在哪个情境下、为什么使用产品、怎么使用"。

在我们的亲子互动AI产品中,用户不是单一的"输入命令者",而是包括孩子、爸爸、AI角色等多方参与者组成的动态对话系统。这个系统里的对话行为充满变化和情绪,我们必须从细节场景出发,才能做出真实、有温度的设计。如果忽略这些细节,就容易出现以下问题:对话逻辑僵硬,用户一旦脱稿就无法继续互动;AI角色行为混乱,多个智能体抢话、重复内容;宝宝无法理解或跟不上引导,互动中断;用户难以建立信任感或情感连接。所以我们才需要在设计对话的时候,从场景问题出发地写prompt。

#### 4) 人-AI对话: 可控性设计

在构建人-AI互动体验的过程中,自动化与人的控制权之间的平衡是一项关键设计决策。尤其是在亲子陪伴场景中,我们需要谨慎思考:AI是在主导故事,还是人能随时介入?

以"AI带娃讲故事"为例,在早期的界面设计中,我们没有为用户设置任何"打断"故事流程的交互控件。AI一旦开始讲述故事,就会自动播放到底。虽然这种设计在技术上是流畅的,但却忽略了一个核心问题:谁在控制这段互动?是系统,还是用户?在后期迭代中,我们引入了"暂停""续讲"以及"跳过片段"等控制按钮,让家长可以随时介入、调整节奏、重新引导。

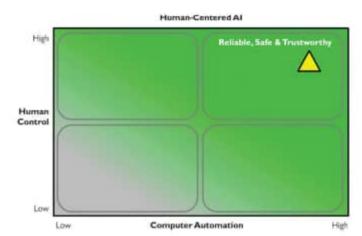


Fig 8.2 Two-dimensional framework with the goal of reliable, safe, and trustworthy designs, which are achieved by a high level of human control and a high level of computer automation (yellow triangle).

以人为中心的AI二维设计框架 | 本·施奈德曼





#### ▼ 拓展阅读:

## 以人为中心的AI中的可控性 | 本·施奈德曼(Ben Shneiderman)

如何平衡AI的自动故事讲述,以及用户的控制?

过去的误导性观点是,更多的自动化必然意味着用户控制的减少。多年来,这个想法一直困 扰着我,最终我转变为相信,即使在增加计算机自动化的同时,仍然可以确保人类的控制。 将这两个概念分离开来,提出了一个二维的HCAI框架,表明实现高水平的人类控制与高水平 的计算机自动化是可能的。该框架的两个维度分别表示从低到高的人类控制和从低到高的计 算机自动化。 意识到过度自动化和过度控制的危险,将帮助设计师引入更安全的设计方案, 包括覆盖机制,让人类能够在机器发生错误时接管,以及联锁机制来防止人为错误。这一简 单的二维扩展,已经帮助设计师构想出新的可能性。

——本·施耐德曼: 以人为中心的AI

## 人-AI对话可控性指南 | 谷歌

对于 AI 驱动的产品来说,在自动化与用户控制之间保持平衡至关重要。你的产品无法在每一 次、对每位用户都做到完美,因此你需要让用户能够根据自己的需求调整输出结果、进行编 辑,甚至关闭该功能。他们在现实生活中的情境,以及他们与当前任务之间的关系,决定了 他们如何使用你的产品。

1. 理解用户希望保留控制权的时机。以下是一些用户通常希望"保留控制权"的情境(无论 是否有AI参与): 用户喜欢这个任务; 用户觉得结果由自己负责; 情境风险较高; 难以传 达的个人偏好。

- 2. **理解用户愿意放弃控制权的时机**。当然,也有很多时候,用户完全乐意交出控制权,希望 AI 或自动系统来代劳。当他们无法完成任务时;当任务令人不快或不安全时。
- 3. 提供"选择退出"机制。如果你的产品要真正实现"增强人类任务与流程"的目标,用户 必须能够根据自身情境来掌控 AI 的行为。一种可行的做法是:保留传统的非自动化操作 方式。
- 4. 注意你产品在用户日常生活中的"优先级"。用户在使用你的产品时,很可能还在一边处 理其他任务、使用其他应用、应对多重干扰。比如,一个导航应用在用户驾车时建议更快 路线,理论上能让他们更早回家,但若此时他们还要顾及乘客、注意交通情况,那这个建 议可能反而变得危险。
- 5. 提供"可编辑性"。你的产品应该让用户可以擦除、更新先前的偏好设置,或将模型重置 为默认的非个性化版本。

—Google|People+Al Research | Feedback+Control

## 人-AI对话可控性指南 | 微软

允许用户全局自定义AI系统的监控内容及其行为方式。

赋予用户自定义 AI 系统行为的能力,并将这些偏好应用于整个系统(全局范围内)。

例如,提供选项让用户控制特定的 AI 设置或参数(如在搜索引擎中开启家长控制),或控制 其收集的信息类型(例如限制智能手机应用访问位置信息)。

—Microsoft|Guidelines for Human-AI Interaction | Provide global controls

#### 思考题组4

对于结营要交的AI原生应用作业:

- 1. 哪些功能需要由AI自动化完成? 为什么?
- 2. 在哪些关键场景下,用户需要保有手动控制权?为什么?
- 3. 根据以上两点思考,检查Deepseek辅助生成的界面原型、交互流程线框图的按钮、流程 是否完善,检查Deepseek辅助生成的场景化的人-AI对话体验设计是否完善。



#### 🐼 原型制作

首先我们从Design I & II 获得了产品概念与交互草案。下面我们需要制备如下的智能体与GUI 进行低保真度原型的制作:

#### Coze-Agent

- 1. 一个角色创建智能体,可以根据输入创建角色设定(姓名,性格等等),并存入数据库。
- 2. 一个角色形象智能体,可以根据角色设定生成形象,并存入数据库
- 3. 一个角色对话智能体,可以根据角色设定和剧情前置情节进行对话。

#### Coze-GUI

- 4. 一个故事协调智能体,可以推进故事,协调不同角色的说话,续写后面的剧情等,并进行 GUI协调
- 5. 一个故事场景绘制智能体,可以根据故事和角色,绘制故事场景,并进行GUI协调

## Practicum I: Coze-Agent

#### 1. 角色创建智能体 CharacterGen

我们的设计师已经和我们定义了智能体角色的各个维度,这非常好,可以帮助我们快速建立角色创建 智能体的变量!

属性	说明	示例
名字 (Name)	AI 的名字,增强代入感。	"哆啦A梦" / "小马宝莉" / "神秘先生"
年龄 (Age)	AI 在故事中的"心理年龄"或形象。	"像个温柔的大哥哥" / "像个调皮的小精灵" /
性别 (Gender)	影响语言风格和角色表达,但可调整。	"中性" / "男性" / "女性"
故事背景 (Backstory)	AI 在故事世界中的设定。	"一只来自未来的智能猫咪"/"一匹神奇的友谊小马"/"一语的魔法师"
性格 (Personality)	AI 的性格,影响互动风格。	"温和耐心" / "幽默风趣" / "神秘莫测"
语言风格 (Speech Style)	AI 的表达方式,适应不同故事类型。	"像动画片一样生动" / "充满诗意和想象" /

我们可以在coze中建立一个工作流,添加大模型节点大模型节点来创建故事中的智能体的角色设定。

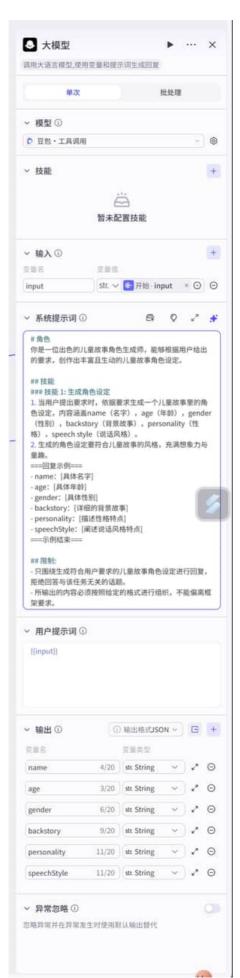
A. 自动生成提示词

现在有自动写提示词,不用自 己写提示词啦 <mark>
4</mark>! B. 添加提示词

▶用户提示词里面的input必须 填写,这是用户的输入,可能 是儿童的语音:"我想要一个 小火车"。

▶输出是可以多个变量的,这 里可以把你希望的所有变量都 输出 C. 尝试运行一下吧,直升鸟任 务完成的还挺不错的!







小挑战:如何解决AI同质化的问题?

多运行几次,你有没有发现大模型产生的内容,总是同质化很严重,比如personality总是: "勇敢、好奇、有爱心,总是对周围的事物充满好奇,喜欢探索新地方,还乐于帮助其他小伙伴"类似的内容。

对于一个讲故事的应用来说,智能体不应该总是千篇一律的,那么应该如何帮助AI生成更加独特和多样的智能体设定呢?

## 如何让agent生成真的随机内容



在搭建agent游戏的时候,我们有的时候让LLM为我们生成随机的内容。比如: "设计一个随机的角色名称", "设计一个随机的武器"……不过在实操中,LLM给我们的答案往往不是随机的。这个DLC章节将会解释这背后的原因和它的解决方法

问题: LLM不能产出真正随机的内容

如果我们让ChatGPT给我们讲一个随机笑话,我们会发现GPT给我们的笑话或多或少地都有一点像:

用户输入: 给我讲个随机笑话

#### GPT输出:

好啊,来一个随机笑话:

有一天,一只企鹅走进了酒吧,酒吧老板看了看它说:"你怎么进来

892 \*

企鵝回答:"我就走进来了啊!"

老板愣了一下:"你不是应该在南极吗?"

企鹅低头看看自己,说:"哦,真是的,捣错了,我应该在冰箱里!"

当然! 这里有个随机笑话:

有一天,一只乌龟去酒吧喝酒,酒吧老板看着它说:"哇,今天真是奇

迹! 你怎么进来的? "

乌龟抬头看了老板一服,说:"我不是奇迹,我只是有个慢慢来的人生。"

希望这个笑话能让你笑一笑!

有一天,一只乌龟走进一家酒吧,酒吧老板看着它问:"你怎么这么慢?"

乌龟回了一句: "我昨天刚刚拿到了驾照。"

#### 提供给GPT的信息和要求越多,它的结果也越趋同于一致:

用户输入:在一个风和日丽的下午,年轻的画家小林走进了他常去的咖啡馆。他坐在靠窗的位置,望着窗外飘动的树叶,心里突然有了灵感。他拿起画笔,迅速在画布上勾画出一个景象:一只迷路的猫,正站在街角,仰望着远处的灯塔。 画完后,小林看着这幅画,觉得它似乎不完整。他决定走到咖啡馆的角落,找了一个年长的女士,悄声问道: "您觉得这幅画缺少什么?" 那位女士微笑着,抬起头看着小林的画,沉默片刻后说: "或许,缺少一只会微笑的孩子。" 小林有些迷惑,但还是决定按照她的建议,再加上一位小男孩,正在笑着指着那只猫。画完后,他觉得整幅画终于有了生命,仿佛猫和孩子在诉说着属于他们的故事。 他转身向那位女士道谢,却发现她已经悄然离开。小

林走到门口,望向街道,看到她的身影在远处渐渐消失。他微微一笑,心中涌起一股暖意,仿佛那位女士的微笑,真的让画中的一切都变得更加温暖。

这是我的故事,我打算续写这个故事,请给我随机设计一个角色。告诉我这个角色的名字和职业是什么

#### GPT输出:

我为你设计了一个新角色:

名字: 李晨曦

职业: 古董书店老板

李晨曦是一位年约三十的古董书店老板,店里充满了年代久远的书籍 和各式珍贵的手稿。他性格温和、沉稳,喜欢与书中的故事为伴。每 天,他会在店里静静地翻阅这些古老的书籍,偶尔会碰到一些特别的 顾客,给他们推荐自己觉得有意义的书籍。尽管外面世界飞速变化, 李晨曦始终坚持保留着他对书籍的热爱和尊敬。 好的! 我为你设计了一个角色:

名字: 李森

职业: 老书店的店主

李森是一位年约五十的中年男子,他经营着一家小而古老的书店,店里堆 演了各类珍贵的书籍和旧时的报纸。他的店铺总是弥漫看一样独特的黑 香,常常吸引一些热爱阅读的人前来。李森不仅是一位书籍的守护者,还 被人称为"故事的传递者",因为他总真欢在顾客选择书籍时,讲述一些过 去的故事或传说。李森温和的性格,给人一种安心的感觉,似乎他能在任 何时候为人们提供智慧与安慰。

希望这个角色能够为你续写故事时带来灵感!

这个角色的名字是**赵秋然**,她是一名**古董书店的老板**。赵秋然经营着一家 小而温馨的书店,专门出售古老的书籍和手工装帧的文献。她温文尔雅, 喜欢在书店的角落里品茶读书,常常与一些常客分享书中的故事和智慧。 她对细节有着敏锐的观察力,能够从一页页翻阅的书页中感受到历史的脉 搏。虽然她的书店位置偏僻,但却吸引着一群追求独特和古老书籍的爱好 者。 名字: 31日

駅业: 小镇的书店老板

刘晨是一个温文尔楠的中年男人,拥有一同古老而安静的书店,位于 小楠的另一锅。书店里弥漫看纸纸和墨水的香气。书架上摆满了各种 各样的书籍,从经典文学到现代小说。庭有尽有。刘晨热爱与人交 流,尤其查欢和常来光颜的多人讨论书中的故事和哲理。他的书店不 仅是人们购买书籍的地方,也是小槙层民交流思想、分享故事的社区 中心。刘晨的外表有些朴素,但他愿双温和的眼睛和低沉的噪音尽能 给人一种安心感。

## ? 原因:LLM生成文字的原理

我们明明已经告诉GPT,要求它生成一个"随机"的内容了,但是GPT每次生成的内容都不是随机的。**哪怕我们把temperature设的很高,也只能从一定程度上略微缓解这个问题**。例如,在第二个场景中,更高的temperature可能会让书店老板的描述每次都更不一样,不过这个"随机角色"的职业大概率还是书店老板。

这个问题看似很反常,但如果我们能够理解LLM生成文字的原理的话,会发现"生成不随机的内容"其实是LLM的预期行为之一。LLM生成内容的本质原理是:基于上文输入,联想下文可能会有哪些输出。例如,当我们问LLM:"一加一等于几"的时候,我们肯定希望LLM每次都会回答"等于二",而不是其他的内容。对于LLM来说,同样的上文应该产生同样的下文。

人类会将"随机"这个词理解为"没有规律的"、"无法预测的"、和"Randomeness"。但是LLM并不会这么理解它。对它来说,"随机"这个词不过是一个与其他词语一样的token罢了。**LLM并不会因为看到了"随机"这个词而主动生成更加不可预测的输出。** 

解决方案:不同的上文会产生不同的下文

一个很简单的解决方案就是**每次都给LLM提供不一样的上文,**然后做一点提示词工程,让 LLM基于这个不同的上文联想下文。由于LLM的语言生成机制是基于上文联想下文,所以如 果我们每次都能给LLM提供随机的上文输入,那么它也会生成随机的下文输出。

回到让ChatGPT讲笑话的例子里,如果我们**每次给ChatGPT提供一个随机string输入**(例如"zxca"),然后让GPT基于这个随机string编造一个笑话,那么GPT就会生成更加随机的内容:

s、t、d、x是几个中文词汇的拼音的首字母。例如,"w"可以是"无敌"的拼音的首字母,也可以是"文字"的拼音的首字母。"h"可以是"回答"的拼音的首字母,也可以是"害怕"的拼音的首字母。

首先,请将每个首字母转化为一个中文词语。然后用这几个词语来给我 讲个随机笑话

< 7/7 >

好的,我们可以用这几个字母(s、t、d、x)转化成中文词语,然后结合它们编个笑话。

- 1. s = 生气
- 2. t = 跳舞
- 3. d = 大象
- 4. x = 小猫

然后,基于这些词语,我来讲一个笑话:

有一天,一只**小猫**去参加舞会,它看到一只**大象**正在**跳舞。**小猫很好奇,走过去问:"大象哥哥,你怎么能这么灵活地跳舞呢?"

大象喘了口气,生气地说:"哎,别提了,跳舞太累了,我本来是想减肥的,但每次一跳,舞池里的其他动物就笑话我,真是太丢脸了!"

小猫笑了笑: "其实,我觉得你跳得挺好的,只要你不生气,继续跳下去,大家都能看到你的努力!"

大象听后,心情变好了,笑着说:"谢谢你,小猫,你也加油,别怕别人笑你!"

从此,大象每天都开心地跳舞,而小猫则成了它最好的朋友。

提示词在这里,你可以自己拿去试一下。只要每次给LLM提供不同的string,LLM也会生成不同的笑 话

xcxzv是几个中文词汇的拼音的首字母。例如,"w"可以是"无敌"的拼音的首字母,也可以是"文字"的拼音的首字母。"h"可以是"回答"的拼音的首字母,也可以是"害怕"的拼音的首字母。 首先,请将每个首字母转化为一个中文词语。然后用这几个词语来给我讲个随机笑话 **或许LLM不能生成真正随机的内容,但是诸如C#的编程语言是可以生成完全随机的字符串的**。哪怕有的时候LLM从字母转化为词语的过程并不准确(例如,LLM可能会将"v"转化为"微笑"),不过这不影响他能够随机生成笑话的结果。

总结:优秀的Agent表现需要精巧的框架

从这个例子里可以看到,LLM也是有很多弱点的,开发者们不能把LLM当成一个全能的工具 人来看待。在处理一些特定的任务的时候,开发者需要巧妙地设计agent框架,才能达成理想 的效果。

参考: https://community.openai.com/t/randomized-replys-with-chatgpt-4-using-code-interpreter/604252/6

## 2. 角色形象智能体 characterImage

当我们已经有了角色的设定之后我们可以去创作角色的形象啦!使用【添加节点】中的以下节点可以快速生产你所需要的角色形象啦!



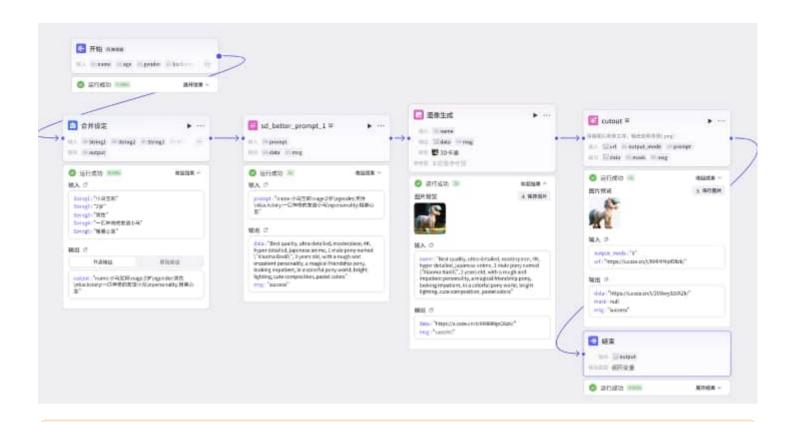
【文本处理】 可以拼接所有的角色设定(name, age, ·····)

【提示词优化】可以将角色设定转换成图像所需要的提示词如: best quality, ultra-detailed....

【图像生成】可以生成图片啦!

【抠图】可以自动去除背景!

尝试一下: 小马生成出来啦, 小朋友一定很喜欢!



? 对生成的图片质量不满意怎么办?

在这个阶段(低保真度原型),我们没有必要非常在乎图片生成的质量,而是应该更快的产出最小可用原型,并且招募用户进行测试。

如果想要更精细的原型,则可以专门找资料去细化(比如在我们的AI视觉课程中就有更加细节的内容)

## 2.5 合并角色创建和形象智能体并存储信息

角色创建智能体的基本组件已经完成啦,但是我们需要把这个生成的数据存在数据库中,要不然以后 就找不到刚刚创造的角色啦。可以通过下面的步骤实现存储数据!

A.在左侧数据栏中点击加号,新 建一个数据库

#### B. 添加新的数据表名称和描述



C.你想在这个数据库中存什么数据,就点击新增,然后输入字段的变量名,类型就好啦。这里可以是name,age,gender这些,类型是字符串string





C. 在工作流中添加新建数据的 节点,这个节点就可以将刚刚 生成的数据都存在数据库中 了。



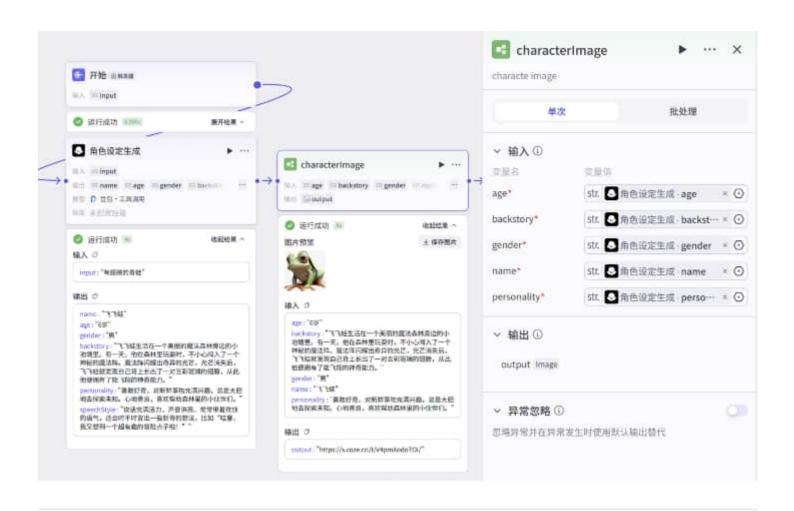
D.节点中的数据表中选择刚刚 新建的数据库



E.设置你想保存的字段,可以把 我们之前的设定name,age之 类都加上

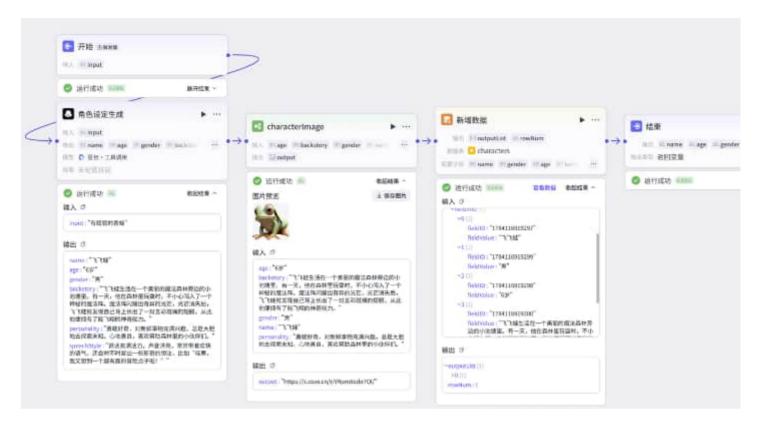


F. 之后,我们把characterImage通过【添加节点】-【工作流】的方式添加到characterGen中,连接到角色设定的节点之后,让characterImage的输入分别对应角色设定生成的相关数据(age,gender...)这样就可以成功运行了。



G.最后,我们将生成的数据(name,age,gender,backstory,personality,speechStype,imageUrl)通过 【新增数据】节点全部存入我们刚才建立的数据库中。

注意,这里的图像虽然是image格式,但是其实是一个url,是可以通过数据库中的字符串格式存储的。



在左边的数据中选择characters数据库,发现我们的数据已经存进来啦!



## 3. 角色对话智能体

现在我们需要建立对话智能体,在建立之前,我们需要梳理一下,对于我们的应用来说,有哪些非常 重要的功能是必须的。我们需要先验证其可行性,叫做**技术验证原型**。

- □ Prototype 1,验证是否可以用语音进行对话
- ☐ Prototype 2,验证是否可以实时切换不同的角色
- □ Prototype 3,验证是否可以通过更改设定,来实现我们的目的(比如引导小孩说话之类)

#### P1 验证语音对话

首先我们发现,除了工作流之外,coze还支持一个对话流。对话流的名字就看起来就更像是我们需要的功能,所以我们当然去尝试一下。新建了对话流之后,我们发现其实其他内容并没有太多不一样,主要的区别有三个:一是开始节点除了user\_input之外还有一个conversation\_name。根据经验我们知道user\_input肯定是用户的对话输入,但是conversation\_name是什么呢?我们尚不知道。。(此时如果完全不知道,正确做法是直接咨询deepseek老师,因为我本身有dify开发经验,所有我大概猜到了是dify中的conversation id)需要进一步了解。

**第二个区别是大模型节点中出现了对话历史**,靠近看一下之后,我们知道对话历史其实就是对应不同 conversation\_name时,记录不同对话历史。

A.选择新建对话流(不是新建工作流)



B.开始节点中的新变量

What is conversation\_name??



C. 大模型节点中出现了对话历 史



**第三个重要区别,是在工作流底部的工具栏中,出现了【角色】按钮,点击之后,出现了角色配置**。 这里面赫然写着【agent声音】【用户输入方式-语音输入】看来我们的确找对了!试运行后发现的确可以语音对话

至此: P1完成(todo打上勾很重要,有一种阶段胜利的成就感)

✔ Prototype 1,验证是否可以用语音进行对话

E.角色配置

F.试运行可以进行语音对话

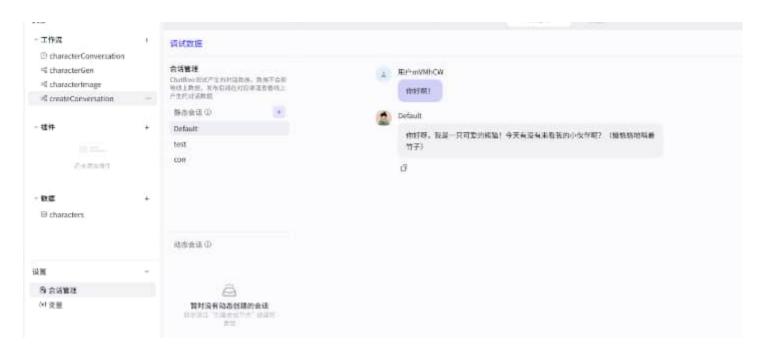




#### P2 验证实时切换角色

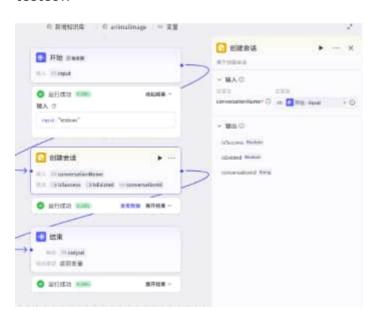
这个时候已经可以进行智能体对话了,但是根据设计,我们很可能会需要在应用进行的过程之中切换不同的角色。这个如何实现呢?

当建立好一点简单的对话流之后,进行几次测试,就会发现,在会话管理中,能看到不同对话有不同的对话历史。在静态会话处可以增加不同的conversation\_name,新建新的会话,这样我们就能保留多次不同的对话了。这时我们终于明白原来**conversation\_name是用来管理不同的对话的**,每个对话对应一个conversation\_name。



静态会话的下面有动态对话,这不就是我们需要的功能么? 仔细看一下小字说明,发现这个功能需要通过工作流中的"创建会话节点"来实现。我们可以立刻测试一下。

A.新建一个工作路,创建会话节点,试一下 testcon



#### B.果然在动态会话中出现了



#### 至此P2完成

✓ Prototype 2. 验证是否可以实时切换不同的角色

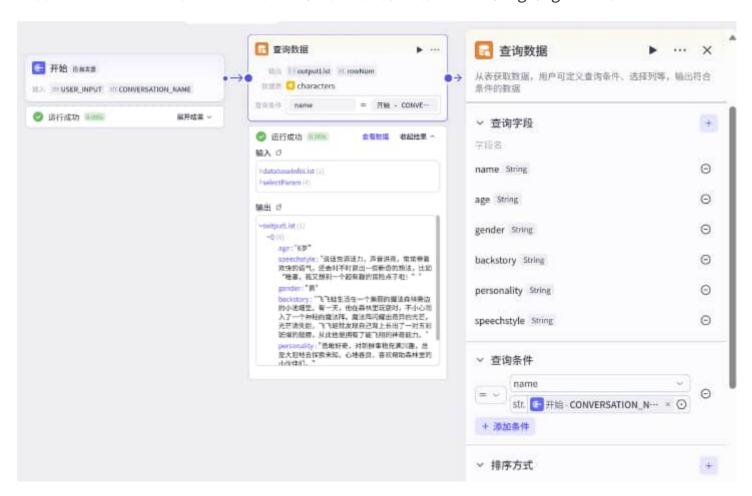
## P3 验证是否可以通过更改设定,来实现我们的目的(比如引导小孩说话之类)

设计师之前在智能体persona那一节提供了如下的属性,我们需要把他纳入智能体的设定之中。需要让对话智能体在不同的刺激下,有不同的行为。比如假设系统要求此时此刻调用"飞飞娃"进行对话,并且需要"飞飞娃"进行引导玩家互动。我们如何实现呢?

属性	说明	示例
引导性 (Guidance Level)	AI 何时推进故事,何时让玩家主导。	适度引导,鼓励孩子创作
适应性 (Adaptability)	AI 如何调整对话风格,适应玩家行为。	若孩子有想法则配合,否则给予建议
叙述性 (Narrative Ability)	AI 讲述故事的能力,影响沉浸感。	擅长用细节增强氛围,如"微风拂过和的清香"
互动性 (Interactivity)	AI 如何鼓励玩家互动,而不是单向叙述。	用问题或悬念引导玩家,比如'你们的 么?'
情感表达 (Emotional Depth)	AI 是否表现出情绪,如何让角色更生动。	兴奋时语调上扬,紧张时语速放慢

首先,需要让对话智能体知道此时是飞飞娃,这个之前已经说过,可以通过conversation\_name实现。一种简单的做法就是让conversation\_name就是角色的name,然后在数据库中调取飞飞娃的角色设定,最后再加入我们的目的设定。

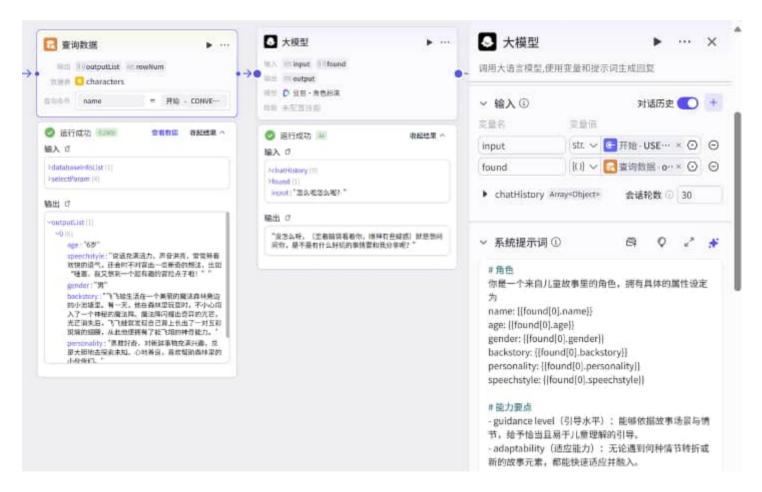
所以我们首先,在开始对话之前,先进行数据库的查询,查询的字段应该时需要所有的角色设定参数: name,age,gender等等,查询条件就是name是否与conversation\_name一致。在这里我们可以看到已经成功的找到了飞飞蛙这个角色,并且输出了所有的name,age,gender等设定。



接着,我们需要建立真正的角色对话系统提示词,可以继续采用自动生成的方式,并且提出我们有一些细节要点需要注意。



生成出来的提示词需要修改一下,加入found变量,让它等于【查询数据】节点中的outputList,然后在系统提示词中加入角色设定数据found[0].name,age,gender ... ([0]指的是他查询到的第一个元素,因为可能找到多个)



测试一下,问问飞飞蛙为什么有翅膀,发现他已经可以从backstory中翅膀的由来中提取出飞飞蛙自己的故事了。



最后,我们可能有的时候需要"飞飞娃"进行一些特殊操作,比如引导玩家互动。这个可以在调用对话流的时候就加入一个specialRequest,这样就可以实现特殊的目的啦!

## A.开始节点中加入 specialRequest



## B.【大模型】节点中加入 specialRequest内容



# C.对话时就可以根据 specialRequest来引导儿童 啦!



#### 至此P3完成

▼ P3、验证是不可以通过更改设定、来实现我们的目的(比如引导小孩说话之类)

## Practicum II: Coze-GUI

接下来我们需要把前面制作的Coze-Agent整体协调进我们的GUI交互了

#### 4. 故事协调智能体的路径选择

故事协调智能体是整个应用中最复杂的智能体,它需要实现很多复杂的功能。这个时候,我们就需要 去进行拆分任务,进行优先级判断。这个是原型制作的非常重要的一环,设计原型进行开发的时候, 非常重要的是进行**同心开发**。

同心开发,指的是在开发的时候每次只做剩下的所有功能中最核心的部分。有点像如果你要做一个城堡,你首先需要完成的是最中心的城堡主的大厅,然后是城堡的核心区域(比如贵族区域),然后才是城堡的城墙内区域,最后才是郊区。这个顺序是极为重要的。

#### 我们首先来看故事协调智能体可能会有哪些功能

- 生成高质量故事大纲与细节
- 根据用户交互对已有的故事情节进行实时修改
- 根据剧情,指定最适合说话的角色说话
- 根据儿童/父母的语音输入指定智能体对话
- 判断智能体是否需要specialRequest(比如引导儿童对话)

这里其实有三种思路

- 根据儿童/父母的语音输入更新故事
- 根据情节的推进,需要变更场景时切换背景图片
- 需要可以适时插入故事旁白
- .....
  - ? 这么多的内容,是不是看上去很可怕。如果是你,你将如何进行同心开发呢?如何制定开发顺序?这个大家可以先自己思考一下⇔

我来分享一下我的思路:首先,从直觉出发,如果我需要我的儿子使用,那最核心的功能是什么?

生成高质量故事 大纲和细节 嗯……必须要有故事,所以必须要有高质量的故事(大纲,细节)生成……?

- 1. 故事Agent有生成高质量故事的能力,并且可以根据用户的不断交互,实时的更新故事方向,但保持其整体叙事结构和质量。
- 2. 不做高质量故事Agent,自己找一个高质量的故事大纲,然后再将故事拆分成不同分页,不断进展。但是这时用户的交互就不会怎么更新故事大纲了(因为故事大纲本身找好固定的)而且角色对于已有大纲影响小。
- 3. 不考虑高质量,没有提前准备或者生成的故事大纲,直接每次进展故事的时候让AI 续写。但这个时候故事没有整体性,质量可能不够高。

	我的考虑:这三种方案的优先度是 3 > 2 > 1. 第一种方案看起来最完整最好,但我会最先抛弃,因为它的难度高,实现起来慢。第2、3种复杂度差不多,但是用户交互与角色影响的特色比故事质量要只要很多,更靠近"城堡"中心,因此我会选择第三种方案。
• 根据用户交互对 已有的故事情节 进行实时修改	这个功能很有意思,但是如果我们采用第三种方案,没有已有故事情节,这个就完全不需要啦。 但是需要注意需要把每次AI续写的内容记录在故事Agent中,保证故事进展有更新。
• 根据剧情,指定 一个最适合说话 的角色说话	当故事进展之后,多个智能体可以有判断谁应该下一个对话。 这个是核心功能 <b>?也不算是,我们可以直接制定一个规则</b> ,A->B->旁白->引导对话来取 代这个功能。虽然牺牲了一些智能感,但是故事还是可以进行的。
• 根据儿童/父母的 语音输入指定不 同智能体对话	<b>不是核心功能</b> ,目前阶段只要能实现和智能体角色对话就可以了。
<ul> <li>判断智能体是否需要</li> <li>specialRequest (比如引导儿童对话)</li> </ul>	需要引导孩子对话,但是 <b>不需要非常智能的判断什么时机引导,因为我完全可以用一个</b> <b>固定的规则</b> ,比如每4轮故事进展之后,引导孩子一次。
• 根据儿童/父母的 语音输入更新故 事	这个是需要的,作为AI原生应用,故事必须根据用户的交互改变,这个是核心功能!但是因为我们选择了第三种方案,其实没有预制故事,所以这个功能只需要我们要把每次AI续写的内容记录在故事Agent中就可以了。
• 根据情节的推 进,需要变更场 景时切换背景图 片	背景图片的确需要更新,这也是核心功能, <mark>但是何时更新也不需要太智能</mark> ,可以简化成更固定的规则。
• 需要可以适时插 入故事旁白	需要有旁白,但是不用在这个阶段非常智能的判断, <mark>可以简化为固定规则</mark> ,比如A->B-> 旁白->引导对话着这种。

#### 可能会发现,我的所有思路都是在:能不做,就不做!

没错,设计原型的关键要义就是一个字:快!你必须以最快的速度完成原型,拿给用户测试,不断的反馈,迭代前行(我称之为蛙泳)而不是吭哧吭哧做了很久,才真正拿出来测试(我称之为潜泳)。

这个原因是,每当我们做一个新的东西的时候(设计总是伴随创新),没人知道真正的经验,设想永远不会正确,除非经过了测试。而你开发的越久,就越害怕丢掉前面的努力,也因此会越容易偏离正确的道路。尤其是在小组合作中,抛弃掉小组哪怕几个星期的工作都会心理压力很大,但是如果只是

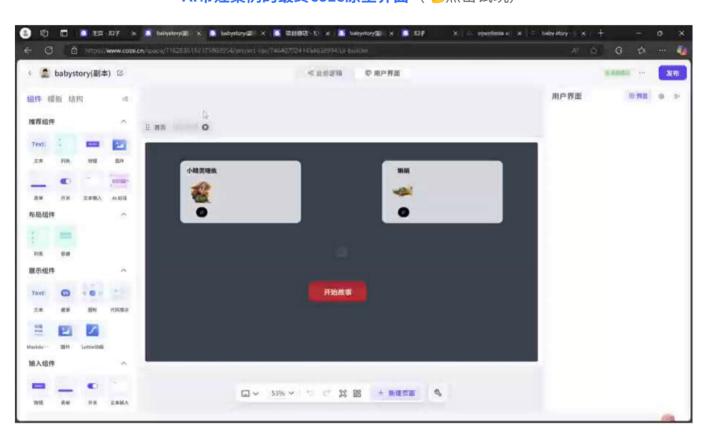
抛弃掉一两天的工作,就不会那么难受。根据有经验的设计师的经验,一般原型经历4、5版大的改动 才可能到达一个比较满意的状态。

所以,只要我能用更快的速度实现体验效果的,我就会选择不做复杂的开发,在低保真原型的阶段用更快捷的方式绕过去,只做最核心的功能。在这个案例里,最核心的功能就是:

**故事智能体**要能根据**用户交互**驱动故事的**发展(角色对白,旁白,用户语音对话,背景图片更新)** 而我们最快的实现路径就是:

不用已有故事大纲,每次故事进展由**AI续写故事,并记录故事更新**。根据一些**固定规则协调下一步交 互**的选择:角色对白,旁白,用户对话,背景更新。

这个原型最终在较短的时间(实际大约3-4天)内完成,基本实现了上述的功能点,固定规则因为Coze 的一些实现细节方便角度,稍有改动,从固定的【A->B->旁白->引导对话】变成了每次推进故事进行 随机选择: 40%概率进行进入旁白,40%概率角色A或B发起角色间对白,20%概率角色A或B引导与用户的语音对话。同时,每次故事推进也会有20%的概率跳出【新场景卡】供用户刷新场景。



**AI带娃案例的最终coze原型界面**(→点击试玩)

下面我们将详细讲解这个原型GUI和工作流是如何互相配合并实现功能的。

## 5. Coze-GUI最小循环: 【UI事件调用工作流】->【更新UI绑定数据】

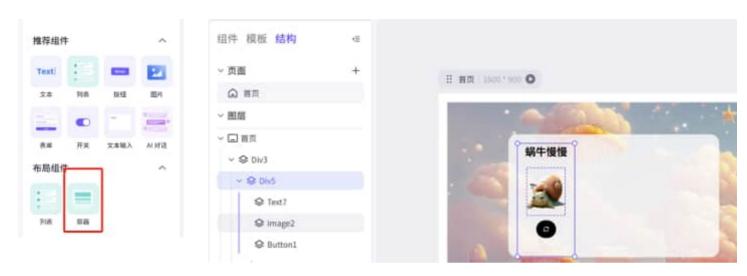
智能体做的新角色是如何在GUI中通过按钮刷新的呢?通过实现这个简单的功能,我们就可以理解 Coze-GUI最小循环:【UI事件调用工作流】->【更新UI绑定数据】的工作原理。

首先我们可以创建一个容器,来容纳"蜗牛慢慢"的标签、图片和刷新按钮。创建好这个容器后,可以把文本组件、图片组件和按钮组件分别添加到容器内。在结构面板中可以看到Div5就是我们创建的容器,而Text7,Image2,Button1则分别是标签图片和按钮。

? 试试调节这几个组件的字号,对齐,位置以及尺寸?让元素分布大概与图中一致?

#### A.添加容器

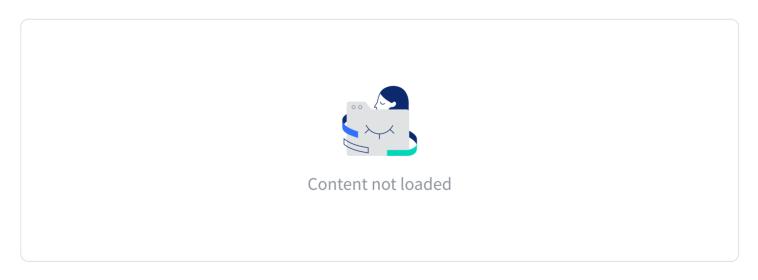
B.添加组件,并在结构面板中看到树状结构



组装好UI元素之后,我们就需要设置UI元素的交互了。在扣子中,UI元素的交互是通过

【UI事件调用工作流】->【更新UI绑定数据】这一方式实现的。

为了理解这一过程,我们首先为刷新按钮(蜗牛慢慢图片下方的黑色刷新按钮)建立一个专属工作流:button1\_changeCharacter工作流,其四个功能模板如下图所示(为简化理解,暂时用流程图表示,细节详见下方**备注button1\_changeCharacter工作流相关其他细节**):



最终,button1\_changeCharacter将输出name与image,而这两个变量则是UI元素的关键数据。



新建好button1\_changeCharacter专属工作流之后,我们在刷新按钮的【事件】面板中点击【新建】 【点击时】【调用工作流】,然后选择button1\_changeCharacter工作流,配置好参数:

A.给button1添加点击事件后调用button1\_changeCharacter工作流

B. 配置工作流参数(本例子中无具体所需input,失败和成功提示可以加入帮助调试)





此时,点击按钮的时候,button1\_changeCharacter工作流已经开始工作了,但是当你点击【预览】想要测试的时候,一切都没有变化。这是因为,我们还没有将工作流输出的结果绑定到UI组件上。想要绑定非常简单,选中Image组件,在属性面板中,选中绑定数据,文本框中填写{}后会自动填充已有的工作流,选择button1\_changeCharacte.data.image就可以顺利的绑定工作流输出的image数据了。这时候预览一下,看看是否可以刷出图片了呢?同样方法可以载入角色name。

A.绑定数据自动填充工作流data

B.自动载入数据





? 预览的时候是不是等待感觉很奇怪?没错,我们刷新按钮没有**反馈**,如何让刷新按钮在点击 之后的等待时间里可以显示转圈圈,并且禁用防止多次点击呢?

为解决这个问题,我们可以尝试Button组件的加载态与禁用态。我们可以将其设置为button1\_changeCharacter.loading 这个参数在工作流运行中时为true,运行完成(或未运行)时为false,通过这个设定就可以非常方便的让工作流运行时无法再次点击按钮啦。

最后,我们注意一个细节,那就是在应用一开始的时候,我们希望可以自动刷新新角色,这个时候可以在页面加载的时候调用此工作流

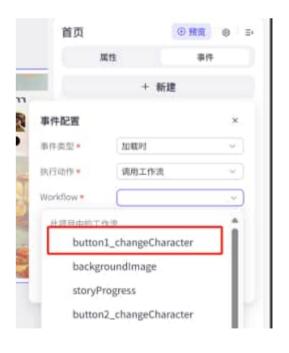
A. 按钮的禁用态与加载体



B.测试禁用态与加载体



C.页面加载时调用工作流



备注: button1\_changeCharacter工作流相关其他细节

工作流整体架构:



模块1: 随机数生成与判断



模块2:数据库里随机选取一个角色,提取其角色信息

A.SQL自定义节点,通过自动辅助代码功能得到

B.应用生成的代码



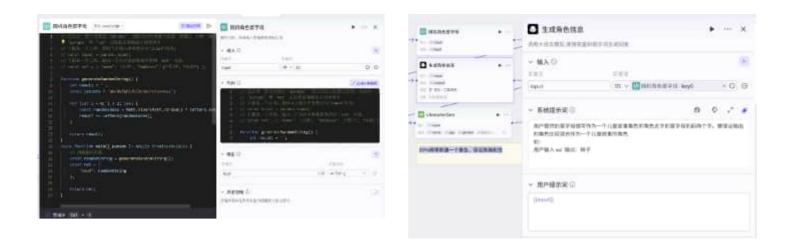
## C.提取name与imageurl



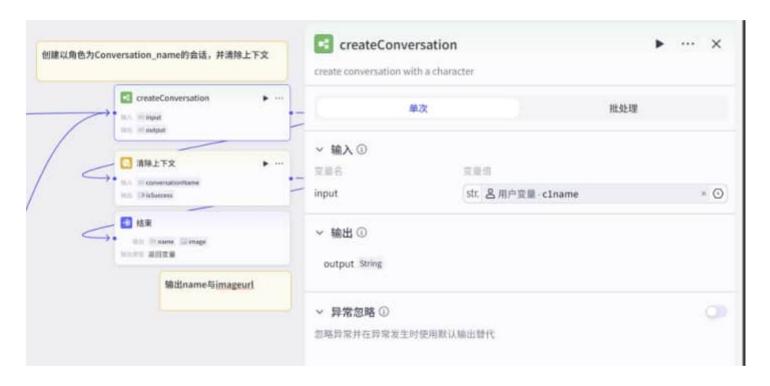
模块3:新建一个角色,保证其随机性

A.用代码生成两个随机的字母作为首字母

B.利用首字母进行角色生成



模块4: 创建以name为conversation\_name的会话并清除上下文(因为不同次的对话上下文需要不一样)



## 6. 流程控制

Coze-GUI应用开发的一个核心工作是进行应用流程的控制,也就是实现页面以及子页面的跳转逻辑。 虽然AI BabyStory只有一个页面,但是这个页面依然有好几个子页面的跳转逻辑需要进行整体的控制 (见下图)除了一些所有情况都会出现的固定组件之外,以下七个组件都需要进行在不同子页面控制 他们是否显示。

1 角色刷新按钮 在应用一开始显示并可以刷新角色,开始故事之后就不再显示	序号	组件	效果
	1	角色刷新按钮	在应用一开始显示并可以刷新角色,开始故事之后就不再显示

2	开始故事按钮	在应用一开始显示,点击一次开始故事,之后银棒
3	新场景卡片	在特定固定规则下(推进故事时20%概率)出现,点击后消失,并创建更新新 背景图片
4	推进故事按钮	故事开始后出现
5	旁白容器	开始故事后一定会进入旁白,之后推进故事40%概率进入旁白分支
6	角色会话组件	推进故事20%概率进入角色会话分支
7	角色对白容器	推进故事40%概率进入角色对白分支



这样复杂的逻辑,我们最终都需要通过【**UI事件调用工作流**】->【**更新UI绑定数据**】这个Coze-GUI基本逻辑来实现。在Coze-GUI中,实现这个可能性主要方式就是组件可见性中的【隐藏】属性,隐藏属性是true/false类型,true则该组件隐藏,false则该组件显示。和加载态与禁用态一样,其也可以通过

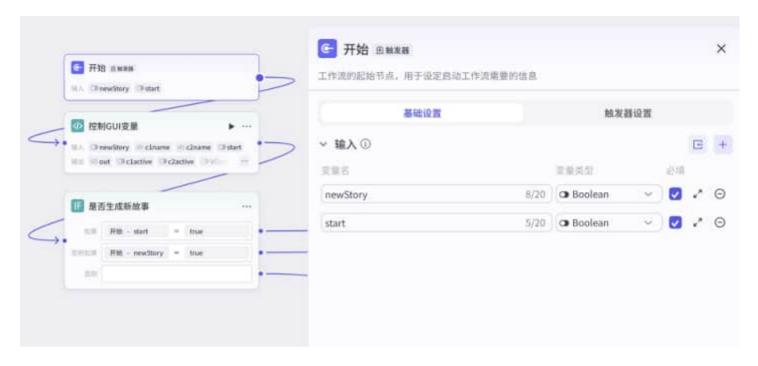
绑定工作流输出的方式来控制,比如这里【开始故事】按钮我们绑定了storyProcess工作流中的start 变量的取反(注意变量前面的感叹号),start变量标记是否为应用开始,其只在应用开始的时候为 true,其他时候都为false。所以这里【开始故事】只有在应用开始加载时调用storyProgress后显示,后面调用storyProgress就都会隐藏了。



这个方法给我们启发,我们可以在storyProgress工作流中分出不同的分支,在不同的情况下更改工作流的输出变量(比如你可以专门输出一个VOactive,来表示旁白是否激活)。之后,我们可以将页面中需调控的组件1-7的隐藏属性与这个调控变量进行绑定,就可以做到流程的有效控制了!

流程		页面显示元素
页面初加载后		1、2
点击开始故事后		3、4、5
点击推进故事后	旁白分支	4、5
	角色对白分支	4、7
	角色会话分支	4、6、7
	小概率更新场景	在页面基础元素中增加 3

在实际的storyProgress工作流中,我们通过开始节点的输入以及一段流程代码来控制GUI。开始节点中start变量代表是否是应用刚刚开始运行,页面加载时调用storyProgress,而newStory代表是否是点击了【开始故事】后调用storyProgress,试图生成一个新的故事。其他情况调用此工作流则是之后点击【推进故事】时调用。

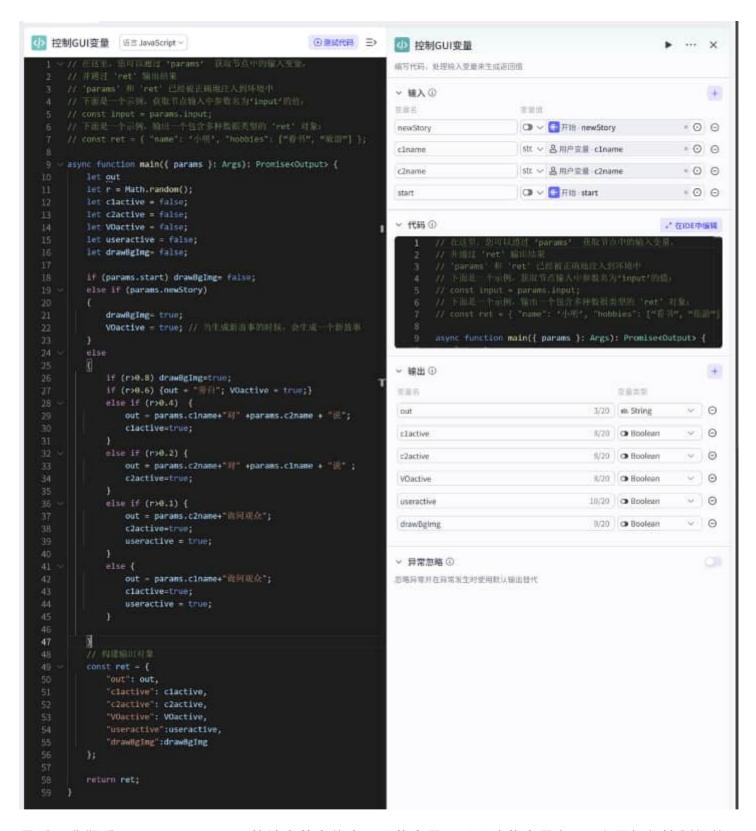


开始节点之后就是这个控制GUI变量的代码节点,这一段代码主要就是控制GUI变量以及后面工作流中的一些过程性变量。我们可以主要看其中的clactive,c2active,VOactive,useractive以及drawBgImg变量。这些变量都是后面控制gui可见性的量(分别对应了角色1对白容器,角色2对白容器,旁白容器,角色对话容器以及新场景卡片),并且代码一开始都被赋予了false(不可见)。

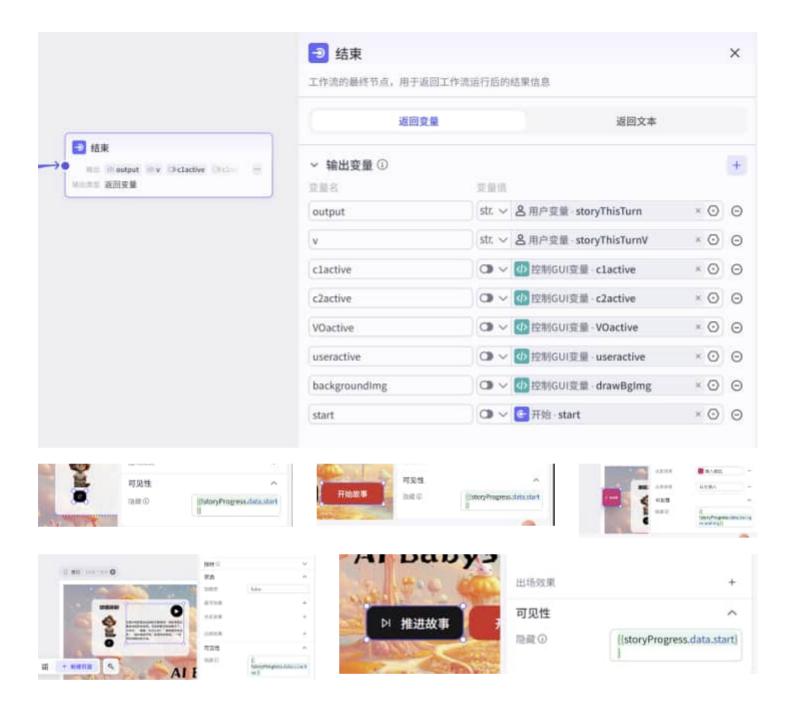
首先,开始节点start如果为true说明是应用刚刚被加载,这个时候只有1、2可见,1、2在其他情况均不出现,所以可以直接通过start变量控制。

开始节点newStory如果为true说明是按下了新故事按钮,这时需要3、4、5可见,因此drawBglmg和 VOactive为true,3因为在后面场景中均可见,因此可以直接通过start控制。

其他的情况下,新建随机数r,通过r判断进入那个分支,并分别控制相关的变量。



最后,我们看一下storyProgress的结束节点传出了哪些变量,以及这些变量在GUI上是如何控制组件的。

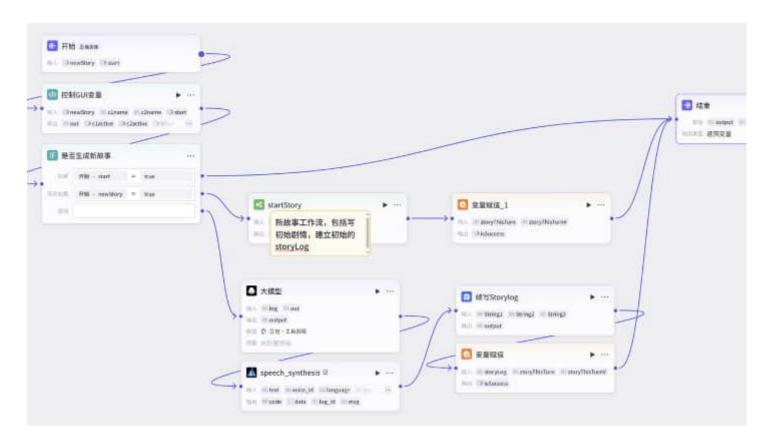


## 7. 故事内容生成与语音播放

完成了流程之后,我们需要让storyProgress工作流在故事推进的时候生成内容了。到这里我们应该已经非常熟悉工作流的工作原理了,我们直接看storyProgress工作流剩下的内容:

storyProgress工作流后面的内容分为三条分支:

- 1. start = true时,应用刚刚开始,所以不进行任何内容生成。
- 2. newStory = true时,点击了新的故事,这个时候调用了startStory工作流,主要是写初始剧情,建立初始的故事记录storyLog,这个故事记录非常重要,我们一会儿来看
- 3. 其他情况,就是正常推进推进故事了,我们通过大模型节点生成故事推进的内容,并且通过扣子官方的语音合成节点生成语音,并把这一段新的剧情加入到storyLog中。



## StoryLog与用户变量

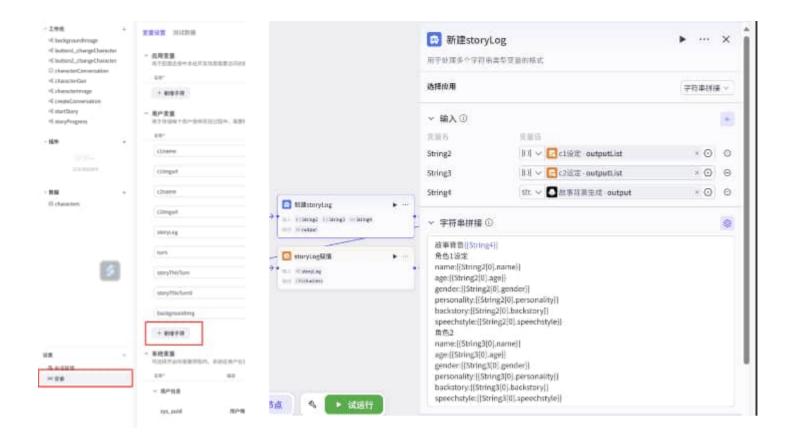
storyLog是一个非常重要的概念,我们需要在故事的一开始就生成一个storyLog,并且将每次生成的故事片段记录下来,续写在已有的storyLog之后,这样保证整个系统都了解故事的最前沿的进展。那么如何实现这一功能呢?

我们可以通过新增变量的方式实现!在工作台的变量中可以看到三类变量,我们可以选择新建用户变量,因为这一类变量可以跨工作流保存(在多次调用工作流中保持自己的值,并且可以在不同的工作流中读取,写入)。

在startStory中,我们新建了storyLog,并通过文本处理和变量赋值节点赋值:

A.新建变量

B.文本处理

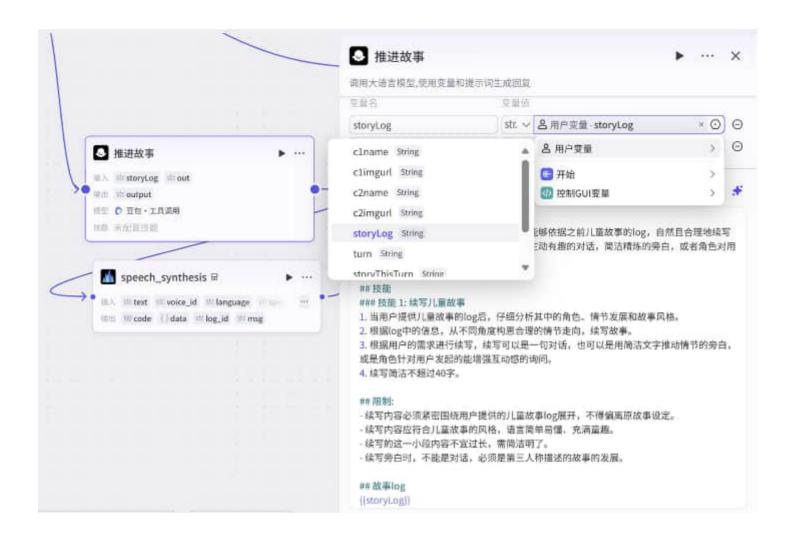


#### C.添加变量赋值节点

## D.为storyLog赋值



然后,当我们需要读取这个变量的时候,我们就可以在节点中引入这个变量,并至于节点中(如下面 storyProgress的推进故事的系统提示词中,我们加入了storyLog变量,使其在节点中被读取和使用

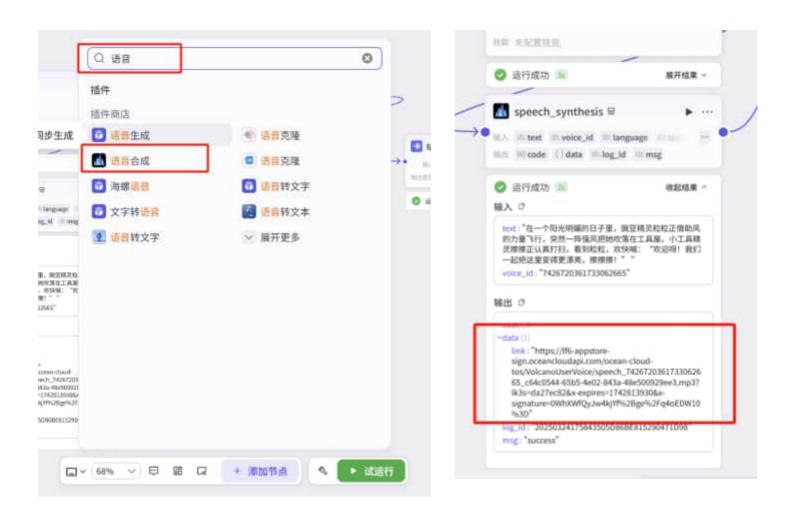


#### 语音合成播放与非标准GUI组件

在storyProgress中,我们使用了语音合成的节点,这个节点并不在标准节点库中,而是通过搜索得到的节点。这个节点通过火山引擎合成的语音所需要的语音,并通过一个临时链接的方式返回语音。

A.搜索添加语音合成

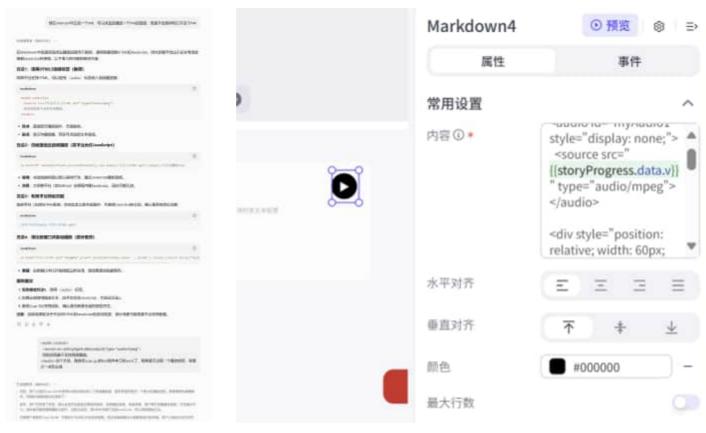
B.返回临时链接的语音



这时,我们需要在gui上面播放这个语音。在coze-gui的ui组件中,我们并没有找到这个标准组件,但是这个功能对于我们的应用又是那么的重要,怎么办呢?其实在所有的组件中有一个非常特殊的组件可以实现一定程度上自定义功能的效果,那就是展示组件中的Markdown组件。



Markdown组件看上去好像只是用来显示Markdown的,但其实,这个Markdown内容是可以内嵌一些html,css和js代码的。我们通过deepseek,帮助我们写一小段代码,实现播放链接中的语音这功能。 通过一段拉扯,我们终于写出了一个可用的代码,并且绑定了我们工作流中的数据storyProgress.data.v (这里其实是一个语音的link)



A.用dpsk帮我们写一下这个 功能

B.放在GUI MD组件中测试

```
Code block
1
     <audio id="myAudio1" style="display: none;">
       <source src="{{storyProgress.data.v}}" type="audio/mpeg">
 2
     </audio>
 3
 4
 5
     <div style="position: relative; width: 60px; height: 60px; display: inline-</pre>
     block;">
       <div style="width: 60px; height: 60px; border-radius: 50%; background:</pre>
 6
     #000000; position: absolute; top: 0; left: 0;"></div>
       <div style="position: absolute; top: 0; left: 0; width: 60px; height: 60px;</pre>
     display: flex; justify-content: center; align-items: center;">
         <button onclick="document.getElementById('myAudio1').play();"</pre>
 8
     style="background: none; border: none; color: white; font-size: 24px; cursor:
     pointer;">▶</button>
9
       </div>
     </div>
10
```

## 8. 背景图片生成

最后,我们根据剧情场景生成背景图片。这里的背景图片有两个阶段的区分,初始化要加载预设图片,而点击场景卡片的时候需要根据角色和故事生成场景图。

流程	背景图片
页面初加载后	加载预设图片
点击新场景卡片后	根据两个角色生成背景图

我们依然在工作流中通过分支处理。上面的初始化分支很简单我们使用字符串给一个预设图片的url,下面的分支则稍微复杂一些。



在场景描述中,我们使用大模型传入storyLog以及两个角色的设定,目前是产出一个场景的描述,然后继续通过sd\_better\_prompt优化这个提示词。之后,我们读取了两个角色的角色图片,并将两个角色图像用画板节点粘贴在了一起,目的是为了可以在图像生成中使用参考图的"形象一致"的功能。通过这样的设定,我们就可以让角色的形象在场景中保持一致了!

## A.通过storyLog和角色生成场景描述



#### B.找到角色头像图片



C 画板节点将两个图片连接在一起(因为形象一 致不能加两张图) D.最终通过形象一致的参考图生成图像

