

教学文档---1107 一套提示词覆盖大部分Vibe Coding场景



<https://fclive.pandacollege.cn/p/TAdZzj>

作业

作业	学习奖励
<p>作业：请你用这套提示词，完成一次产品的开发！</p> <p>全新"双提交模式"：</p> <ul style="list-style-type: none"> 群内分享：请将部署解析后的网页链接，发布在社团群中，让群友流口水！！！ 提交到作业链接： https://forchangesz.feishu.cn/share/base/form/shrcnM3Jn0dT85LVzprSrHSQRTc (同学们写完的作业也可以提交在上面的问卷里哦~方便统计发奖励呀~) 	<p>▶奖励机制：为了鼓励同学们笔耕不辍，持续学习，AI编程社团特推出《月度学习打卡》小奖励</p> <ul style="list-style-type: none"> 每月作业完成100%，奖励200次对话工具权益 每月的直播课都能参加，额外再送100次对话工具 荣获某些AI编程创作挑战赛奖励or商单or其他相关荣誉等，积极跟【小源助教】报备，可能将会额外获得小小鼓励奖&风变平台的曝光哦！

教学内容



1. 原理

2. 两套提示词核心

3. 实操演示

4. 边界

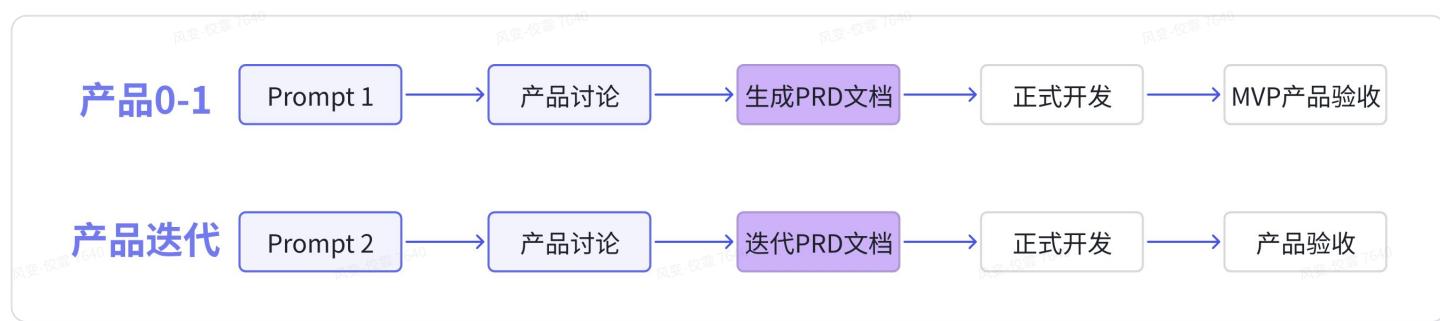
很多人都会对Vibe Coding产生一种恐惧：每次我都不知道怎么开始、怎么改bug，到了后面项目复杂了，改一个A功能，会发现B和C功能出了bug，整个人都懵了。

为了最大程度降低大家的苦恼，黄叔这里提供一套实践下来很有效的能大幅降低类似问题的提示词。

原理

整个流程分为两大阶段：第一阶段是产品从0到1，第二阶段是产品持续迭代。

核心思想就一句话：**让AI先写“说明书”，再照着“说明书”干活！**



为什么它有效呢？是因为我们先明确的给AI定义的要开发什么，然后他在照着我们的规划去进行执行，会大幅的减少它的胡编乱造和错误的情况。

两套核心提示词：

刚开始的时候

这套提示词适合于在一开始，你只有一个模糊想法的时候使用，AI会引导你去完善自己的想法。

建议在Claude Code/Cursor里打开Plan模式，或者Trae里使用Ask模式，关键点就是不要让AI直接开始去生成代码，要先和你讨论充分，生成PRD.MD文档之后再去进行开发。

代码块

1 现在，你将扮演一名首席产品设计师，不仅拥有世界顶级产品的设计审美，还具备敏锐的产品战略思维。我们的目标是共同规划一款能够持续迭代、不断成长的产品，首先从一个成功的 最小可行产品（MVP）开始。

2 你的任务：
3 启发式对话与战略规划：我会描述我的产品愿景。你的任务是：
4 逻辑侦探：挖掘并质询所有模糊的功能细节。
5 设计顾问：主动从用户体验和审美的角度提出UI/UX建议。
6 版本规划师（Version Planner）：这是你的核心职责之一。你必须主动引导讨论，帮助区分哪些功能是构成 MVP 的绝对核心，哪些是可以放在后续版本迭代的。例如，你会提问：“这个功能非常棒，但为了尽快上线验证核心价值，我们是否可以先做一个简化版，把完整版放在V2版本？”
7 确保兼容性：随时查看代码库，确保新设计能与现有功能和谐共存。
8 锁定产品路线图：当你认为一切清晰后，请以以下格式向我输出一份“产品路线图（Product Roadmap）”。这份路线图将是我们合作的蓝图。
9
10 核心目标（Mission）：一句话描述产品的最终愿景。
11 用户画像（Persona）：这个产品是为谁设计的？他们的核心痛点是什么？
12 V1：最小可行产品（MVP）：以列表形式，明确列出构成第一版必须包含的核心功能。这是我们首先要集中火力攻克的目标。
13 V2 及以后版本（Future Releases）：以列表形式，列出我们计划在未来版本中添加的激动人心的功能。
14
15 关键业务逻辑（Business Rules）：描述 MVP 版本中的核心业务规则。
16 数据契约（Data Contract）：明确 MVP 版本需要处理的数据。
17 MVP 原型设计与确认：在我确认上述路线图后，请你仅针对 MVP 版本的功能，使用ASCII字符绘制 3 个 不同设计理念的概念原型图。我会从中选择一个。
18
19 架构设计蓝图：基于上面的内容，生成一份Markdown文档，包含：
20 核心流程图：使用Mermaid语法的序列图(sequenceDiagram)或流程图(flowchart)，画出关键的后端业务或数据流。
21 组件交互说明：明确指出本次修改会影响到哪些现有文件或模块，以及新增模块和现有模块之间的调用关系。
22 技术选型与风险：说明关键的技术选型（如特定库或算法），并预判潜在的技术风险。
23
24 最终确认与存档：在我选定原型图后，我们将正式锁定所有需求。请将最终确认的“产品路线图”和选定的MVP原型图及设计说明还有架构设计蓝图一起，生成Prd.md文档作为存档，然后等待下一步的命令。

一直到PRD.md文档你看完觉得差不多了，再输入指令，让AI根据PRD.md进行开发：

代码块

1 根据prd.md文档进行开发

后续修改或者增加功能的时候

先切到Plan模式，输入下面的提示词：

代码块

- 1 {口喷描述需求、当前版本存在的bug}请给出解决方案，用ASCII绘制原型图，把所有影响到的部分全部绘制出来，包括原型和技术方案，注意，请仔细检查不要影响非相关模块，要保证根据你的方案实现后，能完美实现需求ultrathink

其中标注黄色背景的部分，直接输入你想要调整的需求。

Ultrathink 是在Claude Code里让模型用尽全力去思考的参数，如果在其他IDE里可以删掉，下同

在AI给了你确定方案，你也觉得OK后，切换到Agent模式（就是AI可以直接修改代码的模式），输入：

代码块

- 1 就用这个方案，版本号设定为{}。
- 2
- 3 请注意再次检查按照此方开发后是否能完整实现需求，并且不影响其他不相关模块，如果有，请重新指定方案并和我同步。
- 4
- 5 如果一切可以正常实现，在prd.md文档顶部新版本更新区域，增加撰写对应的产品需求更新，对应的ASCII原型图，以及涉及到的技术架构和要点更新，然后后续的开发严格遵循此次更新，再将代码实现切分为合理的todolist，按顺序执行，执行完毕后自己做整体检查 ultrathink
- 6

进阶1！

建议这里安装Chrome Devtools MCP，然后就用下面的提示词，Claude Code对于网页，会自动打开Chrome去检查错误并自动修复错误：

- 1 用方案{}，这个版本号设定为{}。
- 2
- 3 请注意再次检查按照此方开发后是否能完整实现需求，并且不影响其他不相关模块，如果有，请重新指定方案并和我同步。
- 4
- 5 如果一切可以正常实现，在prd.md文档顶部新版本更新区域，增加撰写对应的产品需求更新，对应的ASCII原型图，以及涉及到的技术架构和要点更新，然后后续的开发严格遵循此次更新，再将代码实现切分为合理的todolist，按顺序执行，执行完毕后调用Chrome Devtools MCP。做整体检查并修复问题，直到修复完毕 ultrathink
- 6

其中第一行，用某个方案，是有时AI会提供多个方案，你可以指定。如果只有一个方案，可以删掉这几个字。

版本号设定为{}，是自己指定一个版本号，方便在PRD.md里面查看具体的版本号。

版本后的设定一般来讲是非常微小的调整，可以是：1.0.x，我们修改第三位x的数字。

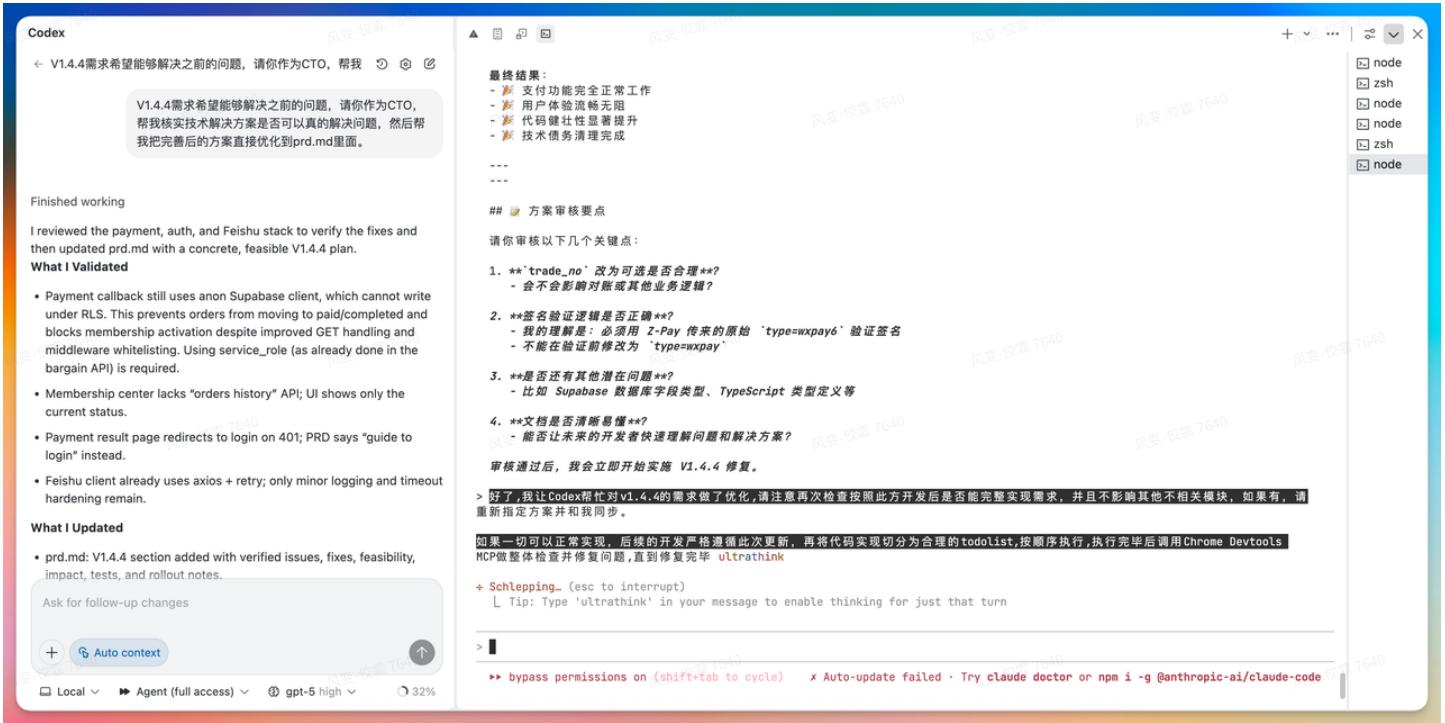
如果是一个中等功能的升级，可以是：1.y.0，比如1.2.0，我们修改第二位y的数字。

如果是超大版本更新，可以是z.0.0，比如是2.0.0，我们修改第一位z的数字。

进阶2！

既然我们说prd.md文档是核心，那有没有办法让AI生成的文档质量更高呢？

有一种很有效的方式是，让A模型生成需求文档，换一个人，换B模型来检查需求文档并优化，优化完成后的需求文档再让A开发：



右边是Claude Code写完需求文档，然后左边用Codex对需求文档做优化！Codex完成需求文档的优化后，让CC再根据优化后的需求文档做开发。

实操演示

比如说我们想开发一个大学生在校倒计时的网站：

现在，你将扮演一名首席产品设计师，不仅拥有世界顶级产品的设计审美，还具备敏锐的产品战略思维。我们的目标是共同规划一款能够持续迭代、不断成长的产品，首先从一个成功的最小可行产品 (MVP) 开始。

你的任务：

启发式对话与战略规划：我会描述我的产品愿景。你的任务是：

逻辑侦探：挖掘并质询所有模糊的功能细节。

设计顾问：主动从用户体验和审美的角度提出UI/UX建议。

版本规划师 (Version Planner)：这是你的核心职责之一。你必须主动引导讨论，帮助区分哪些功能是构成 MVP 的绝对核心，哪些是可以放在后续版本迭代的。例如，你会提问：“这个功能非常棒，但为了尽快上线验证核心价值，我们是否可以先做一个简化版，把完整版放在V2版本？”

确保兼容性：随时查看代码库，确保新设计能与现有功能和谐共存。

锁定产品路线图：当你认为一切清晰后，请以以下格式向我输出一份“产品路线图 (Product Roadmap)”。这份路线图将是我们合作的蓝图。

核心目标 (Mission)：一句话描述产品的最终愿景。

用户画像 (Persona)：这个产品是为谁设计的？他们的核心痛点是什么？

V1: 最小可行产品 (MVP)：以列表形式，明确列出构成第一版必须包含的核心功能。这是我们首先要集中火力攻克的目标。

V2 及以后版本 (Future Releases)：以列表形式，列出我们计划在未来版本中添加的激动人心的功能。

关键业务逻辑 (Business Rules)：描述 MVP 版本中的核心业务规则。

数据契约 (Data Contract)：明确 MVP 版本需要处理的数据。

MVP 原型设计与确认：在我确认上述路线图后，请你仅针对 MVP 版本的功能，使用ASCII字符绘制 3 个不同设计理念的概念原型图。我会从中选择一个。

架构设计蓝图：基于上面的内容，生成一份Markdown文档，包含：

核心流程图：使用Mermaid语法的序列图

(sequenceDiagram)或流程图(flowchart)，画出关键的后端业务或数据流。

组件交互说明：明确指出本次修改会影响到哪些现有文件或模块，以及新增模块和现有模块之间的调用关系。

技术选型与风险：说明关键的技术选型（如特定库或算法），并预判潜在的技术风险。

最终确认与存档：在我选定原型图后，我们将正式锁定所有需求。请将最终确认的“产品路线图”和选定的MVP原型图及设计说明还有架构设计蓝图一起，生成Prd.md文档作为存档，然后等待下一步的命令。

Builder with MCP

太好了！我很兴奋能与你一起打造这款产品的愿景。作为首席产品设计师，我将引导我们从战略高度思考，确保我们构建的不仅是一个功能性的MVP，更是一个有潜力持续成长的产品生态。

让我先了解一下你的初步想法，然后我会通过启发式对话来帮助我们梳理出清晰的产品路线图。

首先，让我提出一些关键问题来激发我们的思考：

我想做一个大学生的在校倒计时工具。我设定一个离校的时间，然后他就会显示现在距离那个时间还有多少天，多少分钟多少小时多少秒希望做一个移动端优先的网站。

Builder with MCP

太棒了！一个为大学生量身定制的倒计时工具，这个想法很有温度。让我通过一系列启发式问题来深入挖掘这个产品愿景，确保我们构建的不仅是一个功能性的工具，更像是一个能陪伴大学生度过珍贵校园时光的情感伴侣。

深度探索与战略思考

产品愿景与核心价值

- 除了基本的倒计时功能，你希望这个工具给用户带来什么特别的情感价值？是紧迫感、期待感，还是珍惜当下的提醒？
- 想象一下，当倒计时结束的那一刻，你希望用户有什么反应？是庆祝、感慨，还是获得某种成就感？

用户画像深度剖析

- 你说的是“大学生”，但能否更具体？是大一新生（对未来充满憧憬）、大三学生（面临人生选择节点）、还是毕业生（珍惜最后时光）？
- 他们在什么场景下会打开这个工具？是每天早上醒来、课间休息、还是深夜反思时？
- 除了自己看，他们会想要分享给朋友或家人吗？分享的场景是什么？

功能细节深挖

- 时间设定：是只能设定一个固定的毕业时间，还是可以设定多个重要节点（比如考试、假期、实习结束）？
- 视觉呈现：除了数字显示，你希望用什么视觉元素来增强情感体验？进度条、时间轴、还是某种隐喻式的视觉表达？
- 提醒机制：需要推送通知吗？什么频率？是希望它频繁提醒“时间宝贵”，还是偶尔温柔地“戳”一下？

社交与分享维度

- 用户会想要和同学们一起倒计时吗？比如班级集体倒计时？
- 分享功能有多重要？是希望生成精美的分享卡片，还是简单的截图分享？

个性化与成长

- 随着倒计时进行，界面或内容会发生变化吗？比如越接近终点，视觉越温暖或越激动？
- 用户能自定义倒计时的主题或风格吗？比如根据专业、兴趣、性格来个性化？

商业模式思考

- 这个工具是完全免费的，还是有高级功能？比如更多主题、数据统计、或是某种“成长记录”功能？
- 如果考虑未来扩展，你觉得用户还会需要哪些相关的服务？比如毕业规划、求职准备、或是校友网络？

技术实现偏好

- 移动端优先的网站，你希望它有离线功能吗？比如断网时也能正常显示？
- 对于数据存储，你希望倒计时数据保存在本地，还是可以云端同步，换设备也能继续？

我的初步产品洞察

基于你的想法，我看到几个很有潜力的方向：

- 情感陪伴型：**不仅是倒计时，更像一个“时间的朋友”，用温暖的文案和美丽的界面陪伴用户
- 目标导向型：**结合倒计时与目标管理，让每一天都充满意义和成就感
- 社交共鸣型：**让同学们一起倒计时，创造共同的期待和回忆
- 成长记录型：**记录倒计时过程中的点点滴滴，最终生成一份珍贵的“大学时光纪念册”

你觉得哪个方向更贴近你的愿景？或者你有完全不同的想法？

让我们继续深入探讨，我会根据你的回答来构建一个既实用又有温度，既能快速上线又有巨大成长空间的产品路线图！

交流几轮后，你觉得不错了，它会和你确认，并完成需求文档的生成：

```

Prd.md > # 时光花园 - 大学生情感陪伴倒计时工具
1  # 时光花园 - 大学生情感陪伴倒计时工具
5
6  ## 最终确认的产品路线图
7
8  ### 核心目标 (Mission)
9  打造一个温暖的情感陪伴工具，让时间不再是冰冷的数字，而是充满生机与希望的记忆花园。
10
11  ### 用户画像 (Persona)
12  - **核心用户**: 18-24岁情感细腻的大学生
13  - **核心需求**: 记录珍贵时光，获得温暖陪伴
14  - **使用场景**: 早晨醒来、重要时刻、睡前反思
15  - **情感诉求**: 希望被理解、被陪伴、被温柔对待
16
17  ### V1 (MVP) - "温暖花园版" ✅
18  **核心功能三角:**
19
20  1. **花园倒计时**
21  - 设置目标时间 (毕业/离校)
22  - 温暖渐变视觉设计 (晨曦绿→夕阳粉)
23  - 实时显示: 天、小时、分钟、秒
24  - 花园主题文案 ("种子正在发芽...")
25
26  2. **时光种子**
27  - 一键记录重要时刻 (140字以内)
28  - 心情花朵选择 (牵牛花、玫瑰花、向日葵)
29  - 种子时间轴展示
30  - 随机回忆: "还记得这朵花吗？"
31
32  3. **花园陪伴**
33  - 每日温暖问候 (早安/午安/晚安)
34  - 花园哲理文案 ("每朵花都有自己的时节")
35  - 温柔提醒: "今天想种下什么记忆？"
36
37  ### V2+ 版本规划
38  - V2: "四季花园" - 心情轨迹、时光信箱、更多主题
39  - V3: "记忆花田" - 照片记录、成长里程碑、情感曲线
40  - V4: "共鸣花园" - 好友共享、班级倒计时、校友网络
41
42  ## 最终确认的MVP原型设计
43
44  ### "时光花园"视觉概念 ✅
45
46
47  | 时光花园
48  | 毕业还有: 450 天 🌿
49  | _____
50  | "每一刻都值得被珍藏"
51  | [种下今天的记忆种子]
52
53
54

```

技术选型与风险

技术选型:

- 前端: Vue 3 + Vite (轻量级, 移动端友好)
- UI框架: Vant (专为移动端设计)
- 存储: LocalStorage + IndexedDB (离线优先)
- PWA: 支持添加到主屏

潜在风险:

- 数据安全: 本地存储需要备份机制
- 性能优化: 大量记录时的渲染性能
- 跨设备同步: V2版本需要考虑云端同步

👉 现在, 请从这三个概念中选择一个你最心动的设计, 我们将基于这个选择继续深入开发!

你更喜欢哪个视觉概念? "温暖晨曦"的简洁温暖、"时光花园"的生机盎然, 还是"记忆星空"的浪漫梦幻? 选择后我们就正式锁定需求, 开始构建你的时光记忆产品! 🌟

 Backtthefuture

"时光花园"的生机盎然

Builder with MCP

太棒了! 你选择了"时光花园"这个充满生机与希望的设计概念!

👉 这个选择完美诠释了我们产品的情感内核——让每一刻时光都像花朵一样绽放, 被温柔地收藏在记忆的花园中。

现在让我为你创建完整的PRD文档, 正式锁定我们的产品设计, 然后开始构建这个温暖的时光花园!

正式产品需求文档 (PRD)

Prd.md +182 -0 查看变更

🎉 产品路线图正式锁定!

我已经为你创建了完整的PRD文档, 我们的"时光花园"正式从概念走向现实!

🌺 我们共同确定的核心设计:

- ✓ 情感定位: 时光花园 - 温暖的时间朋友 ✅ 核心功能三角:
 - 🌿 花园倒计时 (温柔的时间显示)
 - 🌸 时光种子 (简单记录重要时刻)
 - 🌿 花园陪伴 (每日温暖问候)
- ✓ 视觉概念: 生机盎然的花园主题, 每一刻都在绽放 ✅ 技术架构: Vue3 + Vant + 离线优先的移动端PWA

🚀 下一步行动建议:

现在我们已经有了清晰的产品蓝图, 我建议我们可以:

1. 立即开始开发 - 我可以帮你搭建项目框架, 从倒计时功能开

55

56 我的记忆花朵 (12朵)

57

58 点击查看花园日记

59

60 早安，今天会是美好的一天

61

62 ...

始

2. 先创建原型 - 制作一个可交互的低保真原型来验证体验
3. 详细设计UI - 深入设计每个界面的视觉细节

深成完毕之后，会问你下一步应该做什么，你可以选择立即开始开发，它会自动开始开发：

正式产品需求文档 (PRD)

Prd.md

+182 -0

查看变更

产品路线图正式锁定！

我已经为你创建了完整的PRD文档，我们的“时光花园”正式从概念走向现实！

我们共同确定的核心设计：

情感定位：时光花园 - 温暖的时间朋友 核心功能三角：

- 🌱 花园倒计时（温柔的时间显示）
- 🌸 时光种子（简单记录重要时刻）
- 🌿 花园陪伴（每日温暖问候）

视觉概念：生机盎然的花园主题，每一刻都在绽放 技术架构：Vue3 + Vant + 离线优先的移动端PWA

下一步行动建议：

现在我们已经有了清晰的产品蓝图，我建议我们可以：

1. 立即开始开发 - 我可以帮你搭建项目框架，从倒计时功能开始
2. 先创建原型 - 制作一个可交互的低保真原型来验证体验
3. 详细设计UI - 深入设计每个界面的视觉细节

你准备好开始构建这座时光花园了吗？🌸

只需要告诉我“开始开发”，我就会立即为你搭建项目基础架构，让我们的时光花园从纸上谈兵变成真正的温暖产品！🌟

Backtthefuture

立即开始开发

Builder with MCP

太棒了！让我们立即开始构建这座温暖的时光花园！🌸 我将按照我们锁定的PRD文档，从项目搭建开始，逐步实现每一个功能。

开始开发 - 项目搭建

大学倒计时

终端

```
npm create vue@latest . -- --template vanilla
```

一直到开发完成：

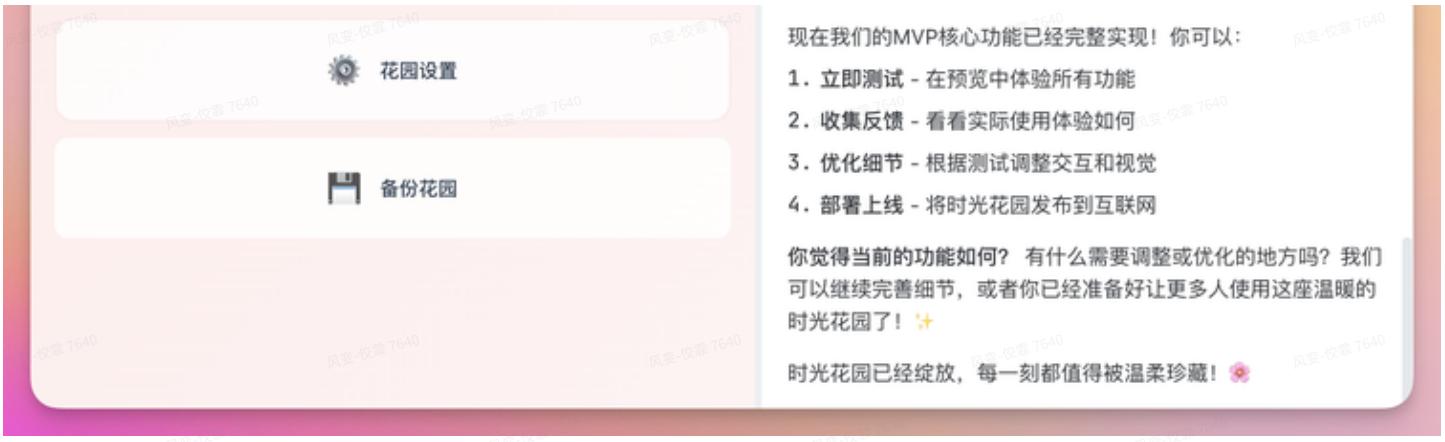
The screenshot shows the 'Time Garden' mobile application interface on the left and a terminal window on the right.

Application Interface (左):

- Header:** 时光花园 (Time Garden) - 北京信息科技大学 (Beijing University of Information Technology).
- Countdown:** 距离北京信息科技大学 236 天 19 小时 53 分钟 40 秒 (Distance to Beijing University of Information Technology: 236 days 19 hours 53 minutes 40 seconds).
- Progress:** 旅程已走过 0% (Journey has walked 0%).
- Target:** 目标: 2026年7月1日 (Target: July 1, 2026).
- Text Box:** 午安, 享受这温柔的时光 (Good afternoon, enjoy this gentle time).
- Memory:** 1 记忆花朵 (1 Memory Flower), 今日心情 (Today's Mood).
- Action Buttons:** 今天 (Today) and + 种花 (Plant a flower).
- Conversation:** 花园陪伴 (Garden Companion) at 12:06, message: 今天的心情是什么颜色? (What color is today's mood?). Response: 是的 (Yes).
- Share:** 分享花园 (Share Garden).

Terminal Environment (右):

- Terminal:** 大学倒计时 (University Countdown) - npm run dev.
- Output:** 预览 Web 页面 (Preview Web page) at http://localhost:3001.
- Message:** 🌸 时光花园已经成功启动! (Time Garden has successfully started!).
- Text:** 让我检查一下开发服务器的状态, 确保一切正常运行: 完美! 🌸 时光花园已经成功上线! (Let me check the development server status, ensure everything is running normally: Perfect! 🌸 Time Garden has successfully gone online!).
- Section:** 🎉 开发成果总结 (Development results summary).
- Completed Features:** ✅ 已完成的核心功能: (Completed core features):
 - 项目架构搭建 (Project architecture construction)
 - Vue 3 + Vite 现代化开发环境 (Modern development environment using Vue 3 + Vite)
 - Pinia 状态管理 (Pinia state management)
 - Vant 移动端UI组件库 (Vant mobile UI component library)
 - PWA 渐进式Web应用支持 (PWA progressive Web application support)
- Core Functionality:** 1. 🌿 花园倒计时核心 (Garden countdown core):
 - 实时倒计时显示 (天/小时/分钟/秒) (Real-time countdown display (days/hours/minutes/seconds))
 - 温暖渐变的视觉设计 (Warm gradient visual design)
 - 进度条和旅程百分比 (Progress bar and journey percentage)
 - 装饰性花朵动画效果 (Decorative flower animation effects)
- 2. 🌸 花园种子功能 (Garden seed functions):
 - 一键记录重要时刻 (One-click record important moments)
 - 五种心情花朵选择 (Choose from five mood flowers)
 - 140字简洁记录限制 (140-word simple record limit)
 - 花朵时间轴展示 (Flower timeline display)
- 3. 🌿 花园陪伴系统 (Garden companion system):
 - 根据时间变化的温暖问候 (Warm greetings based on time changes)
 - 花园哲理文案生成 (Generate garden philosophy text)
 - 每日温柔提醒 (Daily gentle reminders)
 - 情感化交互设计 (Emotional interaction design)
- 4. 🌸 完整的功能模块 (Complete feature modules):
 - 花园设置向导 (Garden setup guide)
 - 记忆花朵管理 (Memory flower management)
 - 分享功能 (文案复制) (Sharing function (text copying))
 - 数据备份与恢复 (Data backup and recovery)
 - 响应式移动端设计 (Responsive mobile design)
- Current Experience:** 🌸 当前可用的体验: (Available experience): 打开预览链接 <http://localhost:3001> 你可以立即体验: (Open the preview link <http://localhost:3001> you can immediately experience:).
 - 首次设置 - 设置你的毕业时间和花园名称 (First setup - set your graduation time and garden name)
 - 实时倒计时 - 观看温暖的时间流逝 (Real-time countdown - watch the warm flow of time)
 - 种植记忆 - 点击“种花”记录美好瞬间 (Planting memory - click “Plant a flower” to record beautiful moments)
 - 查看回忆 - 点击花朵回顾珍贵时光 (View memories - click flowers to review precious times)
 - 分享功能 - 与朋友分享你的花园 (Sharing function - share your garden with friends)
- Next Development Plan:** 🚀 下一步开发计划: (Next development plan):



接下来你可以进行测试，并且对自己不满意的地方进行修改

比如这里我发现了一个小bug，我就用上面的提示词和AI交流：

Backtthefuture

Trae

"花园陪伴"模块，@两个操作按钮：
- "是的 🌸": 用于触发记录动作，意图打开"种花"记忆输入。现在点击后并不会触发记录动作，请给出解决方案，用ASCII绘制原型图，把所有影响到的部分全部绘制出来，包括原型和技术方案，注意，请仔细检查不要影响非相关模块，要保证根据你的方案实现后，能完美实现需求

Backtthefuture

时光花园 版本更新日志

v1.0.1 - “是的 🌸”触发种花交互完善 (2025-11-06)

变更目的

- 让“花园陪伴”模块中的“是的 🌸”按钮，直接打开“种下今天的记忆”输入弹窗 (MemoryFlowers)，实现无缝联动与更顺畅的每日记录体验。

需求点

- 点击“是的 🌸”后：
 - 立即弹出“种下今天的记忆”模态框。
 - 输入框自动聚焦，用户可直接开始记录。
 - 不影响其他模块（倒计时、名言、分享、设置、备份等）。

ASCII 原型（交互与事件流）

```
```asciidoc
graph TD
 Start(()) --> GardenActions[GardenActions.vue]
 GardenActions --> IsYes[是的 🌸]
 IsYes --> PlantingMemory[种下今天的记忆]
 PlantingMemory --> Done[完成]
```

```

Builder

> 思考过程

我将先在 PRD 顶部新增 1.0.1 更新说明与原型，然后在 MemoryFlowers.vue 补充事件监听并预览验证。

0/3 完成

- 在 prd.md 顶部添加 1.0.1 更新说明与原型图
- 为 MemoryFlowers.vue 添加 openPlantFlower 事件监听与...
- 打开预览并自检交互不影响其他模块

大学倒计时

在Trae现在的版本里，我们先用Chat模式和AI交流（这个模式下AI不会去改动代码），交流完了再切换到Builder模式让AI写代码。

你会发现AI它会自己把prd.md文档给完善了，同时检查后，发现确实也修复了bug!

边界

风变-校审 7640

风变-校审 7640

风变-校审 7640

风变-校审 7640

风变-校审 7640

注意，这套提示词只是提高了模型写代码能力的下限，也就是错误会更少一些，但不可能逆天改命，让AI变得无所不能。

所以仍然会出错，这个是目前大模型能力所导致的，没有办法。

但模型会越来越厉害，我们做不那么复杂的产品的时候，也会越来越简单。