

爆款视频仿制 AI workflow — 技术方案 V2.8

****本次升级****：商用化准备 — 用户认证 + 积分会员 + API KEY 管理 + 用户分级分权

****最后更新****：2026-07-06

1. 版本演进

版本	日期	核心变更
:-----	:----:	:-----
V2.2	2026-05	Wan2.2 I2V 统一架构，AutoDL GPU
V2.7	2026-06	Ark Seedance 2.0 视频生成，Pipeline 优化
V2.8	**2026-06**	**登录/注册 · 积分会员 · API KEY配置 · 用户管理**

2. 新增功能

2.1 用户认证

- 邮箱注册（密码 SHA256 哈希）
- Flask Session 登录/退出

- 默认管理员 `admin@qianshi.cool`
- 页面路由: `/` 始终显示登录页, 登录后跳 `/workflow/`

2.2 积分系统

项	值
---	---
注册赠送	30 积分
视频复刻	-10 积分
分镜分析	-3 积分

2.3 API KEY 管理

- 所有 KEY 存 SQLite, Web 界面 (`/settings`) 增删改
- 模型: Ark Seedance / DeepSeek / 智谱 / TOS
- 启动优先级: 数据库 > .env

2.4 用户分级分权

- ****普通用户****: 查看自己的积分/记录
- ****管理员****: `/admin` 查看所有用户, 增删改查, 设置管理员

3. 技术架构

组件	版本	用途
:-----	:-----:	:-----
Python	3.12	服务端

Flask	3.1	Web 框架
SQLite	3	data/app.db (用户+密钥+日志)
Gunicorn	26	WSGI (2 workers)
Nginx	1.24	反向代理 + SSL
FFmpeg	6.1	视频处理

4. 数据库 (SQLite, data/app.db)

users — id, email, password_hash, is_admin, points, membership_type, membership_expire, monthly_used, created_at

api_keys — id, key_name, key_value, description, updated_at

usage_logs — id, user_id, action, points_change, detail, created_at

5. 路由表

路由	认证	说明
:-----	:-----:	:-----
`/`	否	始终显示登录页
`/login`, `/register`	否	登录/注册页
`/workflow/`	是	工作台主页
`/profile`	是	个人中心 (积分/使用记录)
`/settings`	是+管理员	API KEY 在线管理
`/admin`	是+管理员	用户管理 (增删改查)

`/api/auth/*`	否/是	注册/登录/登出/用户信息
`/api/keys`	是+管理员	KEY CRUD
`/api/admin/users`	是+管理员	用户管理 API
`/api/user/*`	是	积分扣减/使用记录
`/api/start`, `/api/state` 等	是	视频管线

6. Nginx 代理规则（关键修复）

```
qianshi.cool/          → 静态 HTML（千视科技首页）
qianshi.cool/login     → proxy → Flask /login
qianshi.cool/register  → proxy → Flask /register
qianshi.cool/workflow/ → proxy → Flask /workflow/ ← 必须代理到同名路径，否则 session 丢失
qianshi.cool/admin     → proxy → Flask /admin
qianshi.cool/profile   → proxy → Flask /profile
qianshi.cool/settings  → proxy → Flask /settings
qianshi.cool/api/      → proxy → Flask /api/
qianshi.cool/download/ → proxy → Flask /download/
qianshi.cool/preview/  → proxy → Flask /preview/
```

7. 视频管线（pipeline_v27.py）

6 个步骤

步骤	内容	依赖
:----:	:-----	:-----
1	分镜分析 + 提示词生成	Zhipu GLM-4.6V / DeepSeek

2	提示词拆分	DeepSeek（多段时）
3	TOS 上传参考图	火山引擎 TOS
4	Seedance 视频生成	Ark API
5	TTS 配音 + 字幕	Edge-TTS / 原音轨
6	拼接合成	FFmpeg

分段策略

- 视频 > 15 秒：按 15 秒一段拆分，多段并行生成
- 视频 ≤ 15 秒：`actual\_seg = min(15, video\_duration)`，整段生成
- 输出时长下拉：固定 5 秒选项 + 分段时长选项

8. Ark Seedance API 限制（2026-07-02 测试结论）

场景	结果
---	---
AI 写实人物图	× PrivacyInformation 被拒
真实视频截图（含人脸）	× PrivacyInformation 被拒
服装模特图	× 检测到真人
只传产品/衣服图（无人脸）	☑稳定通过
纯产品演示视频	☑稳定通过

「只换衣服」可行性

不上传人物图 + 衣服图放产品区 + 分镜提示词描述原视频人物特征 + 强约束「人物必须穿着产品图中的服装」= 稳定跑通。

9. 文件管理

```
每次运行: work_v27_/      ← 独立目录, 互不干扰
每次上传: input///        ← session 隔离
每次输出: output/final_.mp4 ← 带时间戳, 不覆盖
源视频:    input/product_001//
人物照:    persons/product_001//
产品照:    products/product_001//
```

10. 部署路径

路径	内容
:-----	:-----
`/home/ubuntu/viral_video_workflow/webui/`	Flask + 模板 + DB
`/home/ubuntu/viral_video_workflow/scripts/`	视频管线
`/home/ubuntu/viral_video_workflow/data/app.db`	SQLite
`/var/www/qianshi/index.html`	官网首页
`/etc/nginx/sites-enabled/qianshi.cool`	Nginx 配置
`/etc/systemd/system/videoforge.service`	系统服务

11. 页面导航

qianshi.cool (首页)

└─ 视频爆款复刻 → /login → 登录页

└─ 注册 → /register

└─ 登录成功 → /workflow/ (工作台)

└─ 视频复刻

└─  个人中心

└─  用户管理 (管理员)

└─ ⚙ 设置 (API KEY)

└─  回到首页

12. 多语言支持 (2026-07-06)

- ASR 自动检测语言 (whisper 不再硬编码 `language="zh"`)
- 英文保留空格, 中文自动去空格 (CJK 字符占比 >30% 判定)
- 英文 TTS: edge-tts `en-US-EricNeural` 男声
- 字幕支持中英文混排, 时间戳词级对齐

13. 积分定价规则 (2026-07-07)

换算: 1 元 = 10 积分, 1 积分 = 0.1 元

扣分规则 (按分辨率)

		系数	5 秒	10 秒	30 秒		成本	
--	--	----	-----	------	------	--	----	--

分辨率	每秒成本					10 秒收入		毛利率
---	---	---	---	---	---	---	---	---
480p	0.59 元	×10	50 分	100 分	300 分	10 元	5.9 元	41%
720p	1.33 元	×25	125 分	250 分	750 分	25 元	13.3 元	47%
1080p	3.30 元	×65	325 分	650 分	1,950 分	65 元	33 元	49%

公式： 扣分 = 时长(秒) × 系数 。分镜分析 10 分，注册赠送 150 分

充值赠分规则

充值	得分	赠送	每 100 分价格	成本	毛利	毛利率
---	---	---	---	---	---	---
50 元	500 分	—	10.0 元	5.9 元	4.1 元	41%
100 元	1,050 分	+5%	9.5 元	5.9 元	3.6 元	38%
200 元	2,200 分	+10%	9.1 元	5.9 元	3.2 元	35%
500 元	6,000 分	+20%	8.3 元	5.9 元	2.4 元	29%
1,000 元	12,500 分	+25%	8.0 元	5.9 元	2.1 元	26%

成本基准：480p 10 秒 = 5.9 元（含 30% 重试）。毛利率最低 26%。

14. 支付系统（2026-07-09 更新）

14.1 软件架构

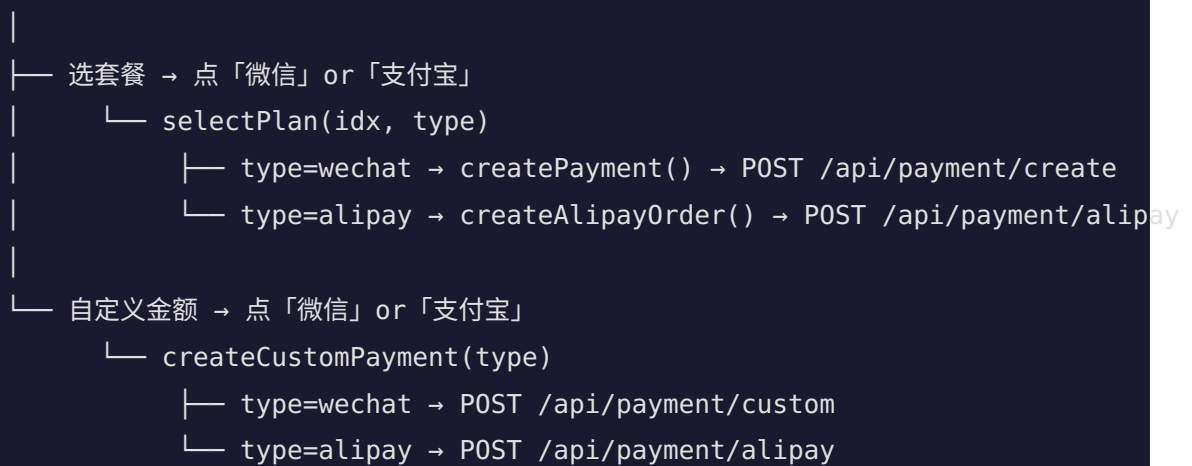
```
webui/payment.py      # 核心模块：微信+支付宝下单、回调、积分
├─ _get_client()       微信客户端 (wechatpayv3 SDK)
├─ create_order()      微信套餐下单
├─ create_custom_order() 微信自定义金额下单
├─ verify_and_fulfill() 微信回调验签+积分
├─ _get_alipay()       支付宝客户端 (python-alipay-sdk)
├─ create_alipay_order() 支付宝下单
├─ verify_alipay_notify() 支付宝回调验签+积分
├─ _save_order()       订单入库 (pay_orders 表)
└─ _get_order()        订单查询

webui/app.py          # 路由层
├─ GET /recharge       充值页面
├─ POST /api/payment/create 微信套餐下单
├─ POST /api/payment/custom 微信自定义金额下单
├─ POST /api/payment/alipay 支付宝下单
├─ POST /api/payment/alipay_notify 支付宝回调
└─ POST /api/payment/wx_notify 微信回调

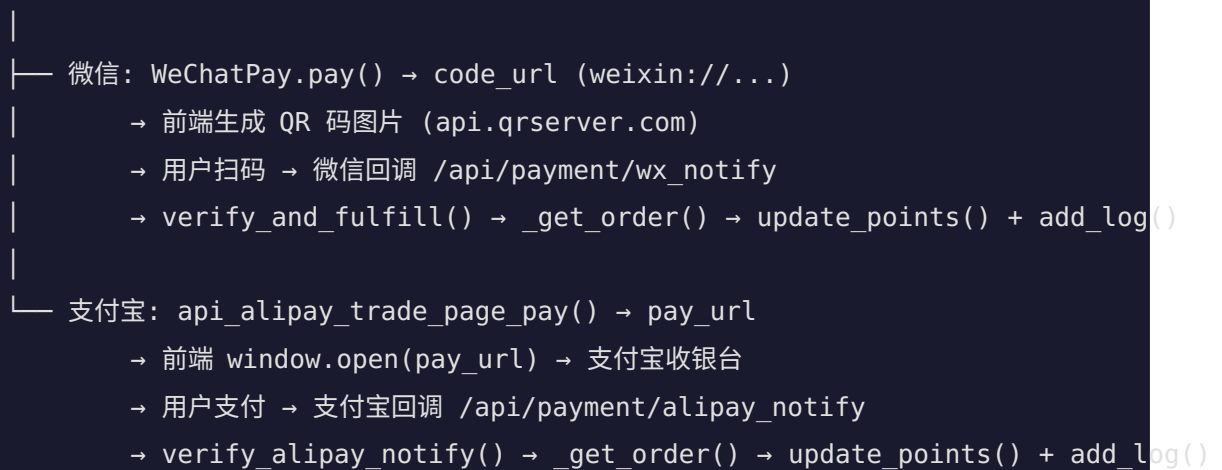
webui/template.html   # 前端充值页面
├─ selectPlan()        套餐选择
├─ createPayment()     微信支付
├─ createAlipayOrder() 支付宝支付
└─ createCustomPayment() 自定义金额支付
```

14.2 支付流程

用户 → /recharge 充值页面



后端 payment.py



14.3 二维码生成

微信：后端返回 `code_url` (`weixin://wypay/bizpayurl?pr=xxx`)，前端通过 Google QR API 生成图片：

```
let qrUrl = 'https://api.qrserver.com/v1/create-qr-code/?size=200x200&data='
    + encodeURIComponent(j.qr_url);
document.getElementById('qrCode').innerHTML = '';
```

支付宝：后端返回 `pay_url`（支付宝收银台 URL），前端直接新窗口打开：

```
window.open(j.pay_url);
```

14.4 安装部署

组件	安装命令
---	---
微信 SDK	<code>`pip3 install --break-system-packages wechatpayv3`</code>
支付宝 SDK	<code>`pip3 install --break-system-packages python-alipay-sdk`</code>
证书	<code>`certs/1115016766/apiclient_key.pem`</code> 等
支付宝密钥	<code>`zhifubao/new_20260709_1456.md`</code>

callback_host: `https://qianshi.cool`（代码中 `CALLBACK_HOST` 变量）

14.5 测试

```
# 登录获取 session
curl -X POST http://127.0.0.1:8899/api/auth/login \
  -H 'Content-Type: application/json' \
  -d '{"email":"admin@qianshi.cool","password":"admin123"}' -c /tmp/t.txt

# 微信 1元测试
curl -X POST http://127.0.0.1:8899/api/payment/custom \
  -b /tmp/t.txt -H 'Content-Type: application/json' -d '{"amount":1}'
# → {"ok":true, "qr_url":"weixin://..."}
```

```
# 支付宝 1元测试
curl -X POST http://127.0.0.1:8899/api/payment/alipay \
  -b /tmp/t.txt -H 'Content-Type: application/json' -d '{"amount":1}'
# → {"ok":true, "pay_url":"https://openapi.alipay.com/..."}
```

注意：支付回调必须走云端`https://qianshi.cool`，本地开发服务器无法接收回调。
测试充值请在云端进行。

14.6 当前状态

支付	状态
---	---
微信支付	☒ 全部完成 (AppID: wx0dbbb415f5f5527d, 商户号: 1115016766)
支付宝	☒ 应用审核中 (APPID: 2021006171615599, 待上线)

14.7 充值规则

- 固定套餐：10元/100分 ~ 1000元/12500分(+25%)
- 自定义金额：1元起，1元=10积分
- 支付后自动回调加积分 + 记录 usage_logs
- 订单持久化：`pay\_orders` 表（防重启丢失）

15. 用户认证（2026-07-07）

- 手机号注册（11位，唯一），登录支持手机号或邮箱
- 忘记密码 → 验证码 → 重置（开发模式验证码显示在页面）
- 注册赠送 50 积分（够 1 条 480p 5s）

- 前端页面 `/forgot`、API `/api/auth/send\_code`、`/api/auth/reset\_password`

Web UI 改动 (2026-07-07)

- **分辨率选择器**：480p/720p/1080p，关联扣分系数
- **预估积分显示**：选择分辨率时实时显示消耗预估
- **充值页面** `/recharge`：套餐选择 + 自定义金额 + 微信扫码支付
- **导航栏**：新增「☒充值」按钮，移除会员徽章
- **扣分公式**：`时长(秒) × 分辨率系数`
- **注册积分**：150 → 50 分
- **db.py**：新增 `calc\_video\_cost()`、`RESOLUTION\_COEFF` 常量
- 验证码开发模式显示在页面

16. 已知遗留

项目	状态
---	---
语音转文字加入提示词	☒ 2026-07-03
字幕独立控制	☒ 2026-07-03
字幕时间对齐	☒ 2026-07-03
多语言支持	☒ 2026-07-06
手机号注册+密码重置	☒ 2026-07-07
积分定价	☒ 2026-07-07
分辨率选择	☒ 2026-07-07
微信支付	☒ 已上线 (v3 加密回调解密 + 自动积分)
支付宝	☒ 已上线 (RSA2 验签 + 自动积分)
API KEY 未全覆盖	待修

SMS 短信平台	待接入
开源模型接入	后续版本
GPU 轨道	已废弃

17. 运维

```
# 服务管理
sudo systemctl restart videoforge
sudo systemctl status videoforge

# 数据库查询
cd /home/ubuntu/viral_video_workflow
python3 -c "from webui.db import get_all_users; [print(dict(u)) for u in get_all_users()]"

# 手动加积分
python3 -c "from webui.db import update_points; update_points(USER_ID, 1000)"
```