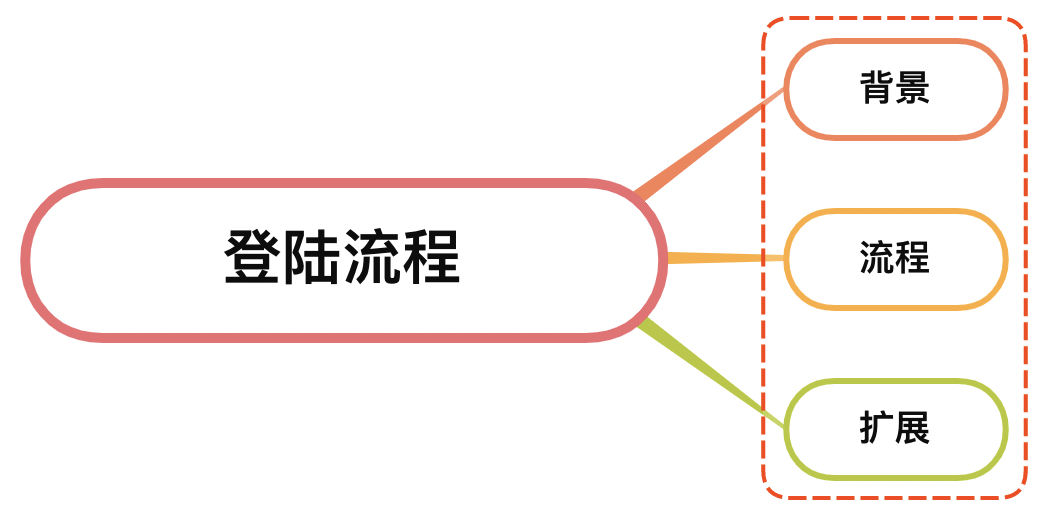
# 面试官：说说微信小程序的登录流程？



## 一、背景

传统的web开发实现登陆功能，一般的做法是输入账号密码、或者输入手机号及短信验证码进行登录

服务端校验用户信息通过之后，下发一个代表登录态的 token 给客户端，以便进行后续的交互,每当token过期，用户都需要重新登录

而在微信小程序中，可以通过微信官方提供的登录能力方便地获取微信提供的用户身份标识，快速建立小程序内的用户体系，从而实现登陆功能

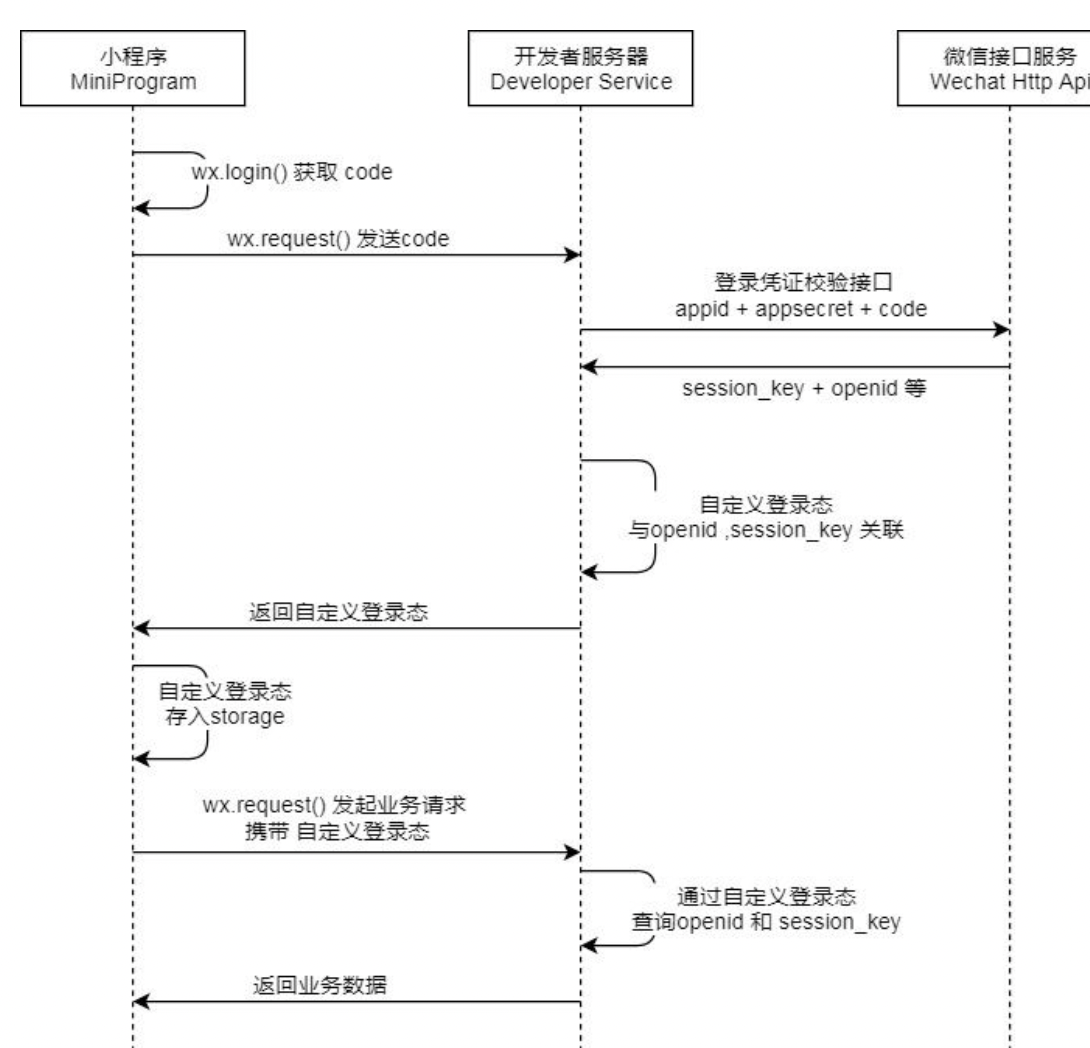
实现小程序用户体系主要涉及到openid和code的概念：

* 调用wx.login()方法会生成code，将code作为参数传递给微信服务器指定接口，就可以获取用户的openid

对于每个小程序，微信都会将用户的微信ID映射出一个小程序 openid，作为这个用户在这个小程序的唯一标识

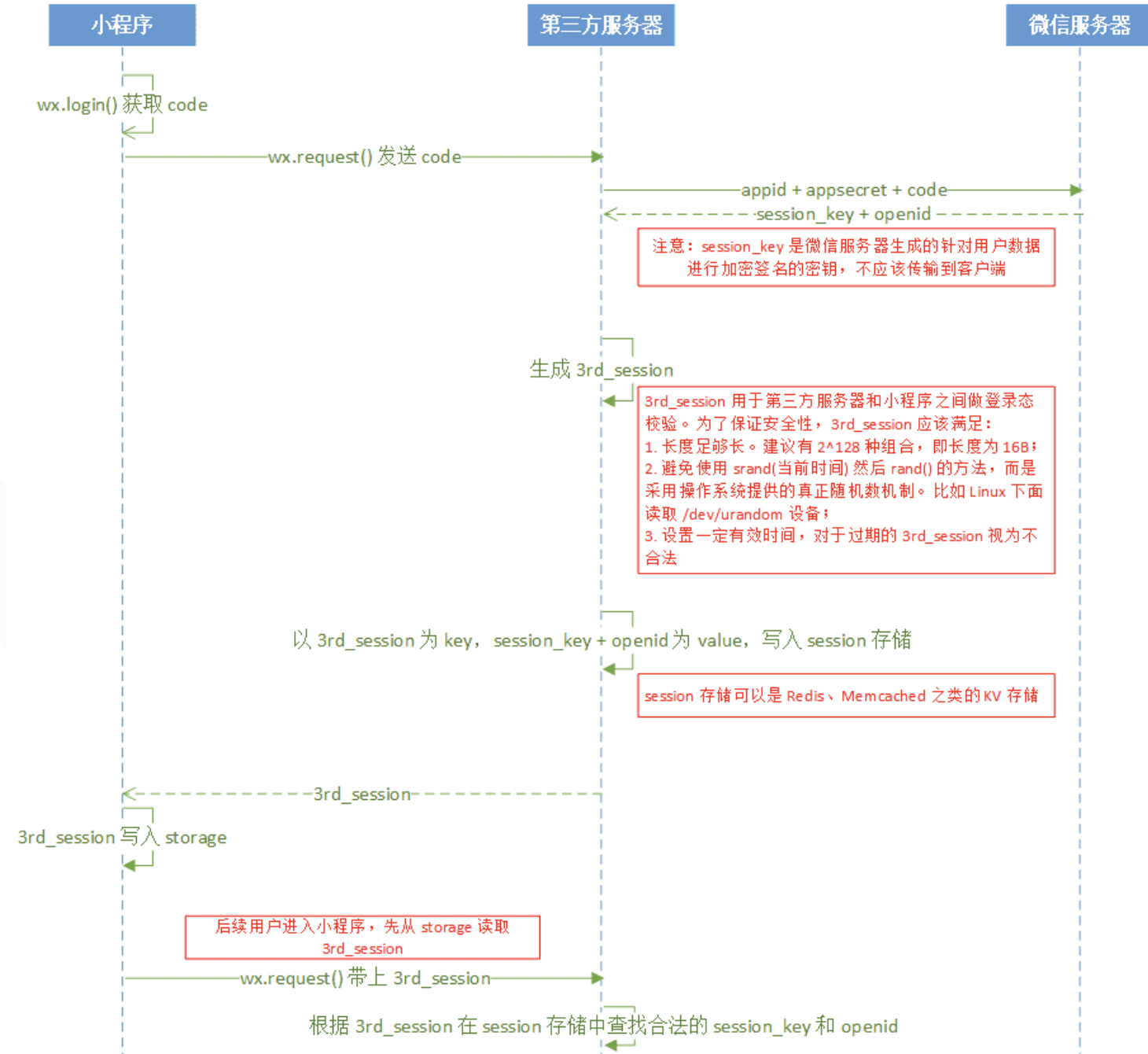
## 二、流程

微信小程序登陆具体实现的逻辑如下图所示：



* 通过 wx.login() 获取到用户的code判断用户是否授权读取用户信息，调用wx.getUserInfo 读取用户数据
* 由于小程序后台授权域名无法授权微信的域名，所以需要自身后端调用微信服务器获取用户信息
* 通过 wx.request() 方法请求业务方服务器，后端把 appid , appsecret 和 code 一起发送到微信服务器。 appid 和 appsecret 都是微信提供的，可以在管理员后台找到
* 微信服务器返回了 openid 及本次登录的会话密钥 session\_key
* 后端从数据库中查找 openid ，如果没有查到记录，说明该用户没有注册，如果有记录，则继续往下走
* session\_key 是对用户数据进行加密签名的密钥。为了自身应用安全，session\_key 不应该在网络上传输
* 然后生成 session并返回给小程序
* 小程序把 session 存到 storage 里面
* 下次请求时，先从 storage 里面读取，然后带给服务端
* 服务端对比 session 对应的记录，然后校验有效期

更加详细的功能图如下所示：



## 三、扩展

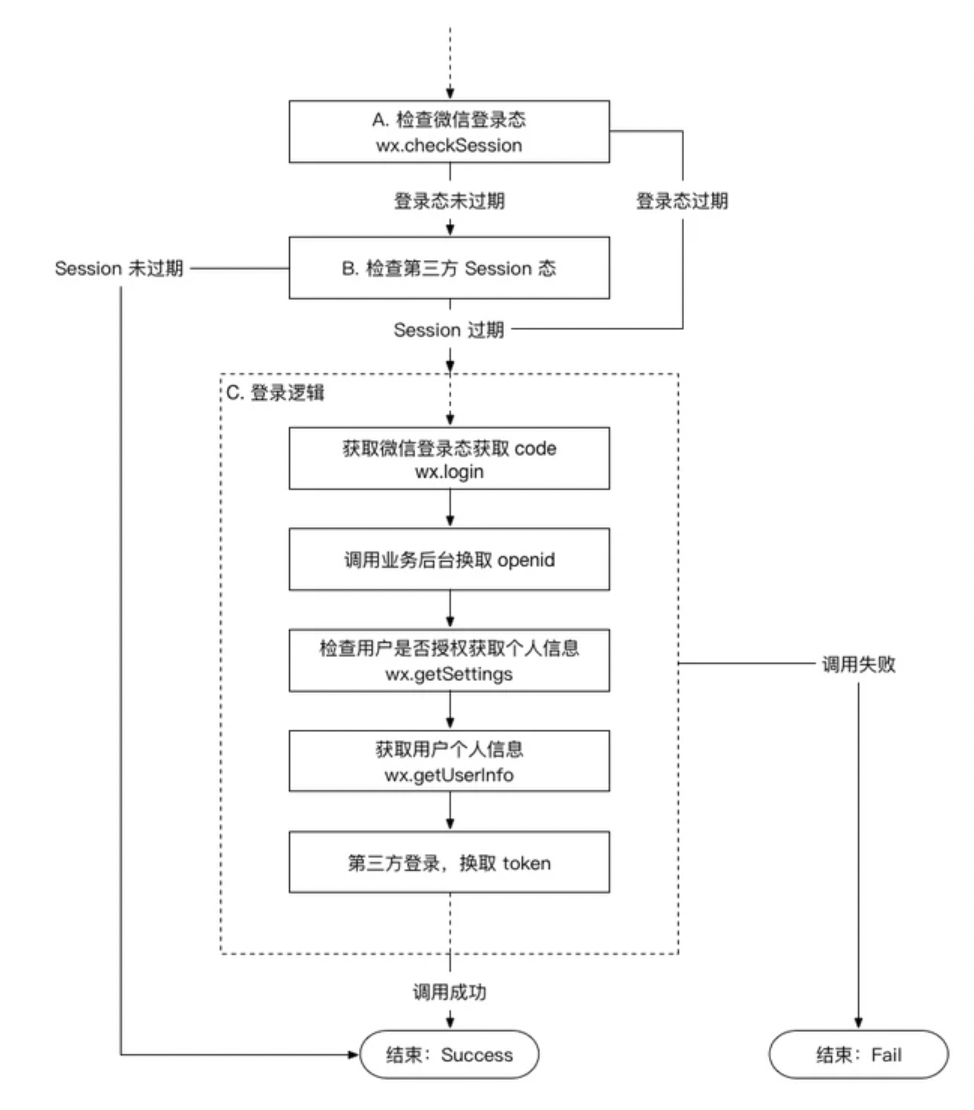
实际业务中，我们还需要登录态是否过期，通常的做法是在登录态（临时令牌）中保存有效期数据，该有效期数据应该在服务端校验登录态时和约定的时间（如服务端本地的系统时间或时间服务器上的标准时间）做对比

这种方法需要将本地存储的登录态发送到小程序的服务端，服务端判断为无效登录态时再返回需重新执行登录过程的消息给小程

另一种方式可以通过调用wx.checkSession检查微信登陆态是否过期：

* 如果过期，则发起完整的登录流程
* 如果不过期，则继续使用本地保存的自定义登录态

这种方式的好处是不需要小程序服务端来参与校验，而是在小程序端调用AP，流程如下所示：



## 参考文献

* https://segmentfault.com/a/1190000016750340
* https://juejin.cn/post/6955754095860776973
* https://www.cnblogs.com/zwh0910/p/13977278.html