# 技术方案

## 1，基本架构

项目目前采用典型三层结构：



1. 手机APP实现用户功能，管理页面（CMS）用于内部管理。
2. 应用服务器实现手机应用和管理页面需要的业务逻辑。
3. 数据库存储结构化、需要动态生成的数据。

实现三层结构的软件架构如下：



1. NGINX作为应用服务器反向代理，并提供静态数据访问。
2. TOMCAT作为jsp/servlet引擎。TOMCAT可以有多台做负载均衡，NGINX分配不同JSP请求给不同TOMCAT。更高级的负载均衡以后可以通过RESIN的应用服务器集群实现。
3. MySQL采用Master – Slave结构的异步复制机制，主数据库执行所有前端请求操作，从数据库可以承担一些后台读操作或数据分析等不需更新数据的管理操作，此外从数据库在本地做增备。

## 2，主要技术问题解决方案

### 手机APP语音编解码互操作

手机APP需要支持IOS和ANDROID两个平台，为了实现这两个平台上语音能够正确编解码，需要采用统一编码格式。

编码选择Opus，编解码器使用WebRTC的实现，IOS参考Novocaine实现。

### 应用服务器结构

NGINX与TOMCAT连接使用proxy模块，通过URL路径区分提交TOMCAT的请求。

### 数据库方案

主数据库与从数据库配成主备复制，从数据库定期做增备。主数据库故障可以临时切到从数据库，从数据库也可Point-in-Time恢复到特定时间点。

### 应用内支付方案

目前支付宝不支持应用内担保交易，只能实现应用内即时到帐。可以使用无线快捷支付或手机网站支付。支付确认机制，手机端支付成功后点击返回给手机应用一个返回值，应用再把返回值发回给server；另外支付宝服务器会异步回调server，通过url参数将支付结果告诉server，server可以根据两个途径的返回值判断支付是否成功。

### 消息推送方案

IOS使用APNS (Apple Push Notification Server)；ANDROID使用GCM (Google Cloud Messaging)。