到云产品概要设计

姓名：郑泽鸿、李志坤、吉鹏云、

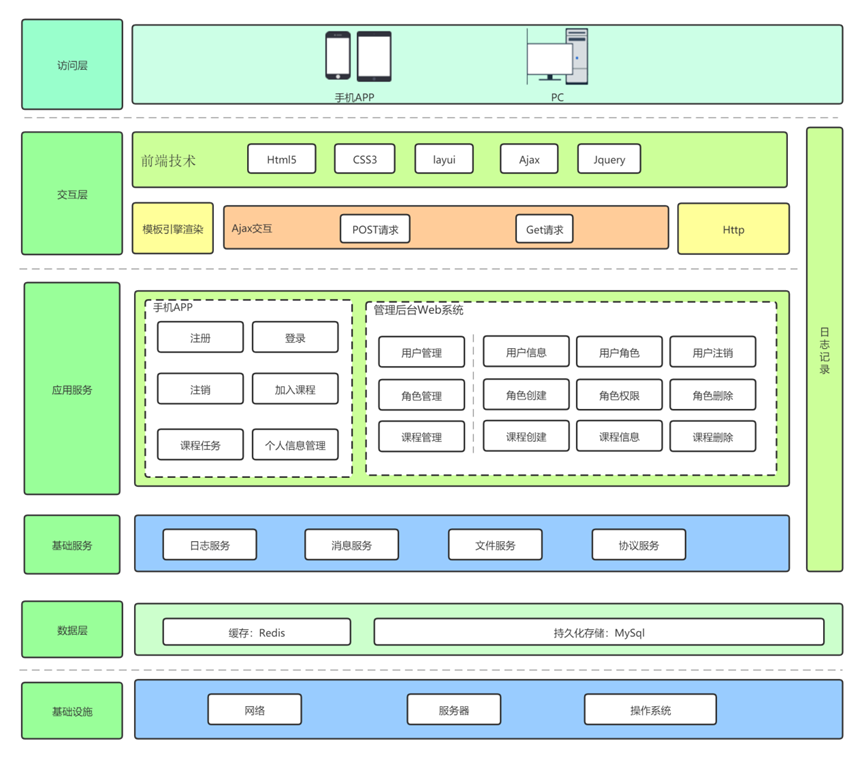
魏明辉、叶似锦

学号：190327103、190327051、190327039、

190327078、190327090

组号：第23组

# 到云App程序架构图



# 到云App开发技术说明

到云App的开发使用原生Android开发技术，Android是世界上市场占有率最高的开源移动操作系统，运行着Android系统的移动设备数不胜数。Android系统大致分为四层架构：Linux内核层为Android设备的硬件提供底层的驱动；系统运行库层通过一些C/C++库为Android系统提供了主要的特性支持；应用框架层为应用程序的开发提供了各种API；应用层是所有Android App运行的层。

原生Android的开发中有四大组件：活动、服务、广播接收器和内容提供器，四大组件为到云 App的功能的实现提供了基础的支持。

此外原生Android提供了丰富的控件、轻量级关系型数据库SQLite、强大的多媒体功能、方便的地理位置定位等等，为到云App的开发提供了极大地方便。

原生Android的开发使用Java语言，Java语言面向对象，一次编译到处运行，同时具有安全性和健壮性，原生Android也为Java语言提供了较多的类库，方便了Java语言的开发使用。

原生Android开发的界面UI开发使用XML语言，编辑器也能够很好地根据编辑的UI生成XML文件，并且能够使得UI的设计跟代码分开，同时XML使用广泛，结构明确清晰。

# WEB技术选择

前端技术：

1. 技术介绍：

(1) MVC：Model(模型)+View(视图)+controller(控制器)，主要是基于分层的目的，让彼此的职责分开。View通过Controller来和Model联系，Controller是View和Model的协调者，View和Model不直接联系，基本联系都是单向的。用户User通过控制器Controller来操作模板Model从而达到视图View的变化。

(2) MVP：是从MVC模式演变而来的，都是通过Controller/Presenter负责逻辑的处理+Model提供数据+View负责显示。在MVP中，Presenter完全把View和Model进行了分离，主要的程序逻辑在Presenter里实现。并且，Presenter和View是没有直接关联的，是通过定义好的接口进行交互，从而使得在变更View的时候可以保持Presenter不变。

(3) MVVM：MVVM是把MVC里的Controller和MVP里的Presenter改成了ViewModel。Model+View+ViewModel。View的变化会自动更新到ViewModel,ViewModel的变化也会自动同步到View上显示。这种自动同步是因为ViewModel中的属性实现了Observer，当属性变更时都能触发对应的操作。

2. 技术选择：

(1)UI框架：Ant Design

Ant Design。旨在统一中台项目的前端 UI 设计，屏蔽不必要的设计差异和实现成本，解放设计和前端的研发资源。蚂蚁金服对Ant Design的定位不仅仅是一套UI组件库，而是一套前端设计语言，提供了大量的设计规范和示例。基本上想到的页面交互都可以用AntDesign来实现，大大降低了页面开发成本。

(2)前端框架：Vue.js

Vue.js是属于MVVM模式的框架。MVVM是把MVC里的Controller和MVP里的Presenter改成了ViewModel。Model+View+ViewModel。Vue.js通过简单的API提供高效的数据绑定和灵活的组件系统。Vue.js的特性如下：

(1) 轻量级的框架

(2) 双向数据绑定

(3) 指令

(4) 插件化

Web架构设计图：

Web端的架构设计主要包括以下六个部分：前端UI、展示层、业务层、数据层、数据库和运行环境。前端我们采用一套渐进式JavaScript框架Vue.js，其自底向上逐层应用，方便与第三方库或既有项目整合。UI的设计我们采用Ant Design,其屏蔽不必要的设计差异和实现成本，解放设计和前端的研发资源。而展示层，我们利用到Ajax实现前后端数据交互。业务层分为六个部分：班课管理、菜单管理、角色管理、用户管理、首页和登录。数据层包含存储过程、数据缓存等多个数据处理操作。数据库是基于当前主流的MySQL,其体积小、速度快、总体拥有成本低，尤其是开放源码这一特点，是该项目使用其的主要原因。服务器我们使用的是面向单机应用场景的新一代计算服务,提供应用一键部署、一站式域名解析、网站发布、安全、运维、应用管理等服务的阿里云服务器。

