# 11\_10 中午演讲

## 请简单介绍你的传智健康项目

传智健康管理系统是一款应用于健康管理机构的业务系统，实现健康管理机构工作内容可视化、患者管理专业化、健康评估数字化、健康干预流程化、知识库集成化，从而提高健康管理师的工作效率，加强与患者间的互动，增强管理者对健康管理机构运营情况的了解。

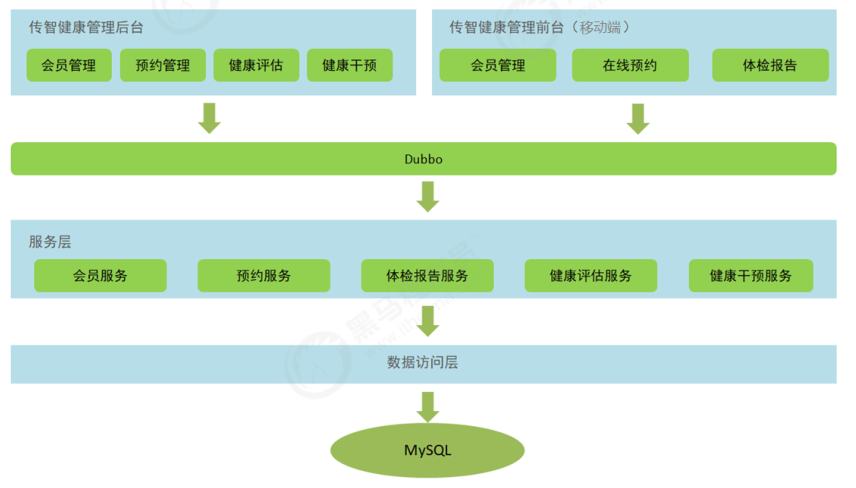
系统分为传智健康后台管理系统和移动端应用两部分。其中后台系统提供给健康管理机构内部人员（包括系统管理员、健康管理师等）使用，微信端应用提供给健康管理机构的用户（体检用户）使用。

后台系统和移动端应用都会通过Dubbo调用服务层发布的服务来完成具体的操作。本项目属于典型的SOA架构形式。

**技术架构:**

****

**主要功能:**



## vue有哪些生命周期函数？分别在什么时候执行？

beforeCreate（创建前） 在数据观测和初始化事件还未开始  
created（创建后） 完成数据观测，属性和方法的运算，初始化事件，$el属性还没有显示出来

beforeMount（载入前） 在挂载开始之前被调用，相关的render函数首次被调用。实例已完成以下的配置：编译模板，把data里面的数据和模板生成html。注意此时还没有挂载html到页面上。  
mounted（载入后） 在el 被新创建的 vm.$el 替换，并挂载到实例上去之后调用。实例已完成以下的配置：用上面编译好的html内容替换el属性指向的DOM对象。完成模板中的html渲染到html页面中。此过程中进行ajax交互。

beforeUpdate（更新前） 在数据更新之前调用，发生在虚拟DOM重新渲染和打补丁之前。可以在该钩子中进一步地更改状态，不会触发附加的重渲染过程。  
updated（更新后） 在由于数据更改导致的虚拟DOM重新渲染和打补丁之后调用。调用时，组件DOM已经更新，所以可以执行依赖于DOM的操作。然而在大多数情况下，应该避免在此期间更改状态，因为这可能会导致更新无限循环。该钩子在服务器端渲染期间不被调用。

beforeDestroy（销毁前） 在实例销毁之前调用。实例仍然完全可用。  
destroyed（销毁后） 在实例销毁之后调用。调用后，所有的事件监听器会被移除，所有的子实例也会被销毁。该钩子在服务器端渲染期间不被调用。

1.什么是vue生命周期？  
 答： Vue 实例从创建到销毁的过程，就是生命周期。从开始创建、初始化数据、编译模板、挂载Dom→渲染、更新→渲染、销毁等一系列过程，称之为 Vue 的生命周期。

2.vue生命周期的作用是什么？  
 答：它的生命周期中有多个事件钩子，让我们在控制整个Vue实例的过程时更容易形成好的逻辑。

3.vue生命周期总共有几个阶段？  
 答：它可以总共分为8个阶段：

创建前/后,

载入Dom(页面渲染)前/后,

更新前/后,

销毁前/销毁后。

4.第一次页面加载会触发哪几个钩子？  
 答：会触发 下面这几个beforeCreate, created, beforeMount, mounted 。

5.DOM 渲染在 哪个周期中就已经完成？  
 答：DOM 渲染在 mounted 中就已经完成了。

## 请解释vue中v-model的含义

数据双向绑定数据(v-model)

**概述：**

Vue的双向绑定可以实现: 数据变化的时候, 页面会自动刷新, 页面变化的时候，数据也会自动变化.

**注意：**

1. 双向绑定, 只能绑定"文本框,单选按钮,复选框,文本域,下拉列表"等
2. 文本框(value)/单选按钮/textarea, 绑定的数据是字符串类型
3. 单个复选框, 绑定的是boolean类型
4. 多个复选框, 绑定的是数组
5. select单选对应字符串，多选对应也是数组

.sync 属性 和 v-model

.sync 属性双向数据绑定,将属性的值和数据进行双向绑定,

V-model 将标签的value 与数据进行绑定

## 简单说明vue的响应数据格式

响应内容是json字符串

{

 // `data` 由服务器响应的数据

 data: {

Message:xxx,

Flag:xxx,

Data:xxx

...

},

 // `status` 来自服务器响应的 HTTP 状态码

 status: 200,

 // `statusText` 来自服务器响应的 HTTP 状态信息

 statusText: 'OK',

 // `headers` 服务器响应的头

 headers: {},

 // `config` 是为请求提供的配置信息

 config: {}

}

使用 then 时，你将接收下面这样的响应：

axios.get('/user/12345')

.then(function(response) {

  console.log(response.data);

  console.log(response.status);

  console.log(response.statusText);

  console.log(response.headers);

  console.log(response.config);

});

在使用 catch 时，或传递 rejection callback 作为 then 的第二个参数时，响应可以通过 error 对象可被使用。