## 《饭否》软件需求调查分析

### 用户领域的组织结构

本软件为类B2B模式，用户群主要分为三类：餐厅、顾客和骑手。餐厅为卖家，负责提供产品。顾客为买家，可以在平台上进行点餐。骑手则负责在餐厅地址和顾客地址之间往返来送餐。

### 功能型需求

本软件需要实现的功能有以下几种：

账户功能。根据用户的要求有注册、登录、根据账户组来决定用户的权限。共需要设置管理员、餐厅、骑手、顾客这四类账户组。

根据不同的账户组，各自拥有以下功能：

管理员：拥有管理账户的权限、以及其他账户的所有权限。

餐厅：拥有添加、修改、删除菜品属性的功能，接单功能。

骑手：拥有接单功能，确认送达功能。

顾客：拥有点单功能，评论功能和举报功能。

### 非功能型需求

本软件需要保证以下非功能性需求：

1. 在IOS和Android上均可运行。
2. 在不同手机、不同版本的安卓手机上均能适配。
3. 系统需稳定、可靠、安全、可更新。
4. 满足多种排序方式，使用户体验更人性化。
5. 增加记忆功能，顾客点单时可回忆上次点单内容（除非商家已做产品信息变更）。
6. 自动为骑手推荐离自己更近、点单等待时间更长的单。

### 设计约束

1. 减少跨系统的交互，一个系统尽量只能CURD自己业务域的数据库，不要跨域去操作其他应用的数据。
2. 尽量减少IO以及网络的访问，将多次的调用整合在一次操作中完成，尽量减少IO资源的浪费。
3. 禁止在操作数据库或者外部接口时候放在循环里面，尽量做成批量接口调用。
4. 系统间调用最好是只读，系统间的修改用事件或者消息来实现比较靠谱。
5. 尽可能在数据模型上控制业务对象的约束关系；如果通过程序逻辑去保证完整性与一致性，会存在一定的风险。

参考文献：

1. “需求分析”百度百科：

<https://baike.baidu.com/item/%E9%9C%80%E6%B1%82%E5%88%86%E6%9E%90>

1. “需求调研”百度百科：

<https://baike.baidu.com/item/%E9%9C%80%E6%B1%82%E8%B0%83%E7%A0%94/9331616>

1. 《常见的系统设计规范（约束）》——CSDN

<https://blog.csdn.net/xuxian6823091/article/details/80474029>