## 总体描述

## 2.1产品前景

## 大学里有来自天南地北的学生，每个人口味不一样。学生不知道什么菜品符合自己的口味，吃到不合适的菜品时也没有方便的反馈渠道。食堂工作人员不知道菜品是否符合学生的口味，没有方便的渠道获知学生们对菜品的感想。此系统目的在于帮助学生自主选择套餐，针对学生的个人口味和身体情况，制定合适的餐饮计划，并提供方便快捷的渠道供学生和食堂工作人员沟通，使食堂工作人员能针对学生的评论改进菜品。预期系统投入使用一个月后，使用此系统的学生对食堂菜品的满意度提高20%，在食堂就餐的人数增加20%。

## 2.2产品功能

产品的主要功能有：

**FID1**：学生能[对菜品进行评价（选择菜品口味标签、打分、文字评价）](#_Toc466745738)

**FID2：**学生能输入个人信息（输入身体指标、输入每日餐饮总金额）

**FID3：**学生可以[查看推荐食品](#_Toc466745742)

**FID4：**学生和食堂工作人员能[查看菜品评价](#_Toc466745744)

**FID5：**学生和食堂工作人员能[回复他人评价](#_Toc466745745)

**FID6：**食堂工作人员能[输入/修改菜品信息](#_Toc466745746)

**FID7：**食堂工作人员能[删除菜品](#_Toc466745747)

**2.3用户特征**

此系统有两类用户，一类是在食堂吃饭的学生，另一类是在食堂做饭菜的食堂工作人员。学生来自天南地北，每个人口味相差很大，由于食堂菜品种类繁多，往往不知道什么菜符合自己的口味，在选择吃什么菜时有选择困难症。学生每天使用此系统2-3次。食堂工作人员负责做饭菜，但是往往无法获知学生对菜品的评价，难以改进菜品，食堂工作人员每天使用此系统多次.

**2.4约束：**

此产品的约束如下：

**CON1：**产品应能在ios端和android端运行

**CON2：**数据库使用MySQL数据库

**CON3：**系统占用内存不得超过100M

**CON4：**系统应该符合大学的规章制度  
**CON5：**在推荐食品时应该考虑食客的风俗习惯，如民族，宗教的因素。

**2.5假设和依赖**

此产品的假设和依赖如下：

**Suppose1:**食堂工作人员能熟练使用手机

**Suppose2:**系统收录了所有的菜品信息

**Suppose3:**食堂经理支持此系统

**Suppose4:**食堂工作人员愿意使用此系统

**Suppose5:**食堂工作人员能及时更新菜品信息

**Suppose6:**学生会通过锁定推荐套餐来标记吃过的套餐

**Suppose7:**本系统只在南京大学内使用

## 3.2功能需求

### 3.2.7 查看菜品信息

#### 3.2.7.1 特性描述

学生和食堂工作人员能查看菜品的具体信息

优先级 = 高。

#### 3.2.7.2 刺激/响应序列

刺激：用户选定一道菜品

响应：系统显示菜品具体信息

刺激：用户选定菜品的一个评论

响应：系统显示一个用于输入评论的文本框

刺激：用户输入评论并保存

响应：系统提示保存成功并刷新页面，显示用户输入的评论

注：菜品的具体信息包括菜品简介、菜品的评价统计（标签统计和分数统计）、用户对菜品的评价

#### 3.2.7.3 相关功能需求

|  |  |
| --- | --- |
| **编号** | **需求描述** |
| **Dishes.Evaluate.heat** | 菜品的热度，数值等于对该道菜品评价的人数，参见《需求列表》FR25 |
| **Dishes.Evaluate.label** | 系统应当显示菜品的标签统计数据， 参见《需求列表》FR26，参见**Dishes.Evaluate.labelFilt、**  **Dishes.Evaluate.label.sort** |
| **Dishes.Evaluate.label.filt** | 系统应当过滤掉一些评价数量相对少的标签，参见《需求列表》FR27 |
| **Dishes.Evaluate.label.sort** | 系统显示的标签应按倒序排序，参见《需求列表》FR28 |
| **Dishes.Evaluate.totalscore** | 系统应显示菜品的总评分，参见《需求列表》FR29 |
| **Dishes.Evaluate.scoreDistribute** | 系统应显示菜品的评分分布，参见《需求列表》FR30 |
| **Dishes.Evaluate.userEvaluate** | 系统应显示用户对菜品的评价，参见《需求列表》FR31;  参见UserEvaluate |
| **UserEvaluate.sort** | 用户的评价应按时间顺序排列，参见《需求列表》FR32 |
| **UserEvaluate.score** | 系统应显示每个评论的分数 |
| **UserEvaluate.label** | 系统应显示评论所贴的标签 |
| **UserEvaluate.text** | 系统应显示文字评论 |
| **UserEvaluate.time** | 系统应显示评论的时间 |
| **UserEvaluate.userID** | 系统应显示评论人 |
| **Dishes.introduction** | 系统应显示菜品的简介 |

### 注：评论内容见3.2.8回复评价

### 3.2.8 回复评价

#### 3.2.7.1 特性描述

学生和食堂工作人员能在菜品具体信息或者我的消息列表里选定一条评论对其进行回复

优先级 = 高

#### 3.2.7.2 刺激/响应序列

刺激：用户选定一条评论

响应：系统显示用户输入评论的文字框

刺激：用户输入评论并提交

响应：系统保存用户的评论、刷新数据库和当前页面，显示更新后的页面，并将用户输入的评论放置在最显眼的位置

#### 3.2.7.3 相关功能需求

|  |  |
| --- | --- |
| **编号** | **需求描述** |
| Reply.reply | 系统应允许用户对评论进行回复 |
| Reply.check | 系统应对用户提交的回复进行检查，参见Reply.check.null，Reply.check.wordNumber |
| Reply.replyOthersReply | 系统应允许用户对回复进行回复，参见《需求列表》FR35 |
| Reply.inform | 当有用户对自己的评论或回复进行回复后，系统应提示被回复人查看回复 |
| Reply.info | 系统应保存回复的相关具体信息，参见Reply.info.content，Reply.info.time，Reply.info.userID |
| Reply.info.content | 系统应保存回复的文字内容 |
| Reply.info.time | 系统应标注回复的时间 |
| Reply.info.userID | 系统应标注回复者ID，参见《需求列表》FR37 |
| Reply.cancel | 系统应允许用户在编辑回复过程中放弃回复 |
| Reply.content.Del | 系统应允许用户对输入尚未提交的文字进行删除 |
| Reply.check.null | 系统不允许可以提交内容为null的回复 |
| Reply.check.wordNumber | 用户输入的评论字数不得超过140字，参见《需求列表》FR34 |
| Reply.refresh | 用户在提交回复后，系统应刷新页面，并显示回复 |