

# 今天吃什么——作业报告

小组：汪汪队（赵天豪，张嘉诚，连昊）

## 一、程序功能介绍

### 1. “盲盒模式”

主界面的“盲盒模式”按钮，即可进入“盲盒模式”的界面。

“盲盒界面”提供了时段、食堂、菜品类别、主要成分、口味口感、忌口饮食限制六个类别的 tag，其中时段、食堂为单选，其余为不定向选择，当然某些类别没有想好也可以不选。由于菜品类别、主要成分、口味口感有很多不同的 tag，一次只呈现四个，因此提供了换一批的功能，点击即可换一批上述三方面的 tag，可以为用户提供更多的可能。

在选择好自己喜欢的 tag 之后，即可点击开始生成，机会为用户产生一张满足 tag 的食物的 detailed card，有关 detailed card 会在图鉴模式中详细介绍。

最后生成的 detailed card 界面也可以返回进行新一轮的生成。

请选择你喜欢的tag，点击开始生成，为你随机生成一份美味吧！（或者点击换一批来更换下面的选项）

时段			
<input type="checkbox"/> 早餐	<input checked="" type="checkbox"/> 午餐	<input type="checkbox"/> 晚餐	
食堂			
<input type="checkbox"/> 燕南	<input type="checkbox"/> 农园	<input type="checkbox"/> 学五	
<input checked="" type="checkbox"/> 家园	<input type="checkbox"/> 学一	<input type="checkbox"/> 其他	
菜品类别			
<input type="checkbox"/> 饮料	<input type="checkbox"/> 小吃	<input checked="" type="checkbox"/> 主菜	<input type="checkbox"/> 主食
主要成分			
<input type="checkbox"/> 鸡腿	<input type="checkbox"/> 土豆	<input type="checkbox"/> 肥肠	<input type="checkbox"/> 擂椒
口味、口感			
<input type="checkbox"/> 咸	<input type="checkbox"/> 嫩	<input type="checkbox"/> 干	<input type="checkbox"/> 酥
忌口、饮食限制			
<input type="checkbox"/> 全素	<input type="checkbox"/> 清真	<input checked="" type="checkbox"/> 无香菜	
<input type="checkbox"/> 低碳水	<input type="checkbox"/> 低糖	<input type="checkbox"/> 低油	
<input type="checkbox"/> 无麦麸	<input type="checkbox"/> 无坚果	<input type="checkbox"/> 无乳制品	

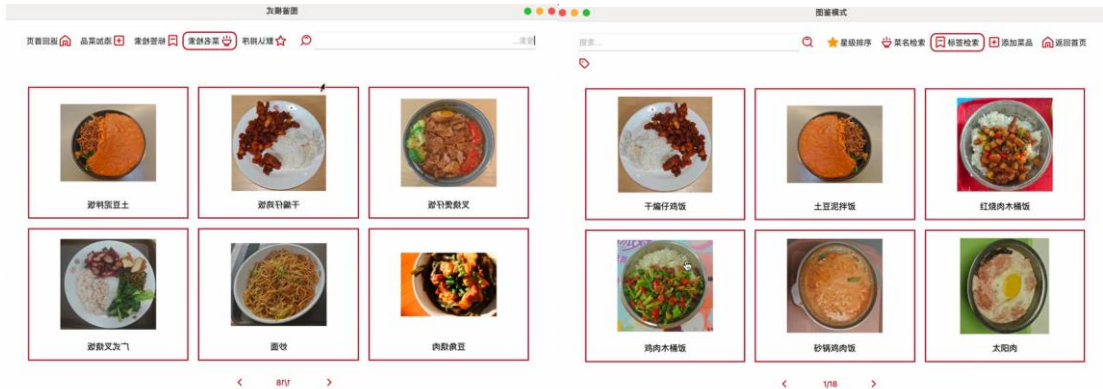
### 2. 图鉴模式

主界面的“图鉴模式”按钮，即可进入图鉴界面。

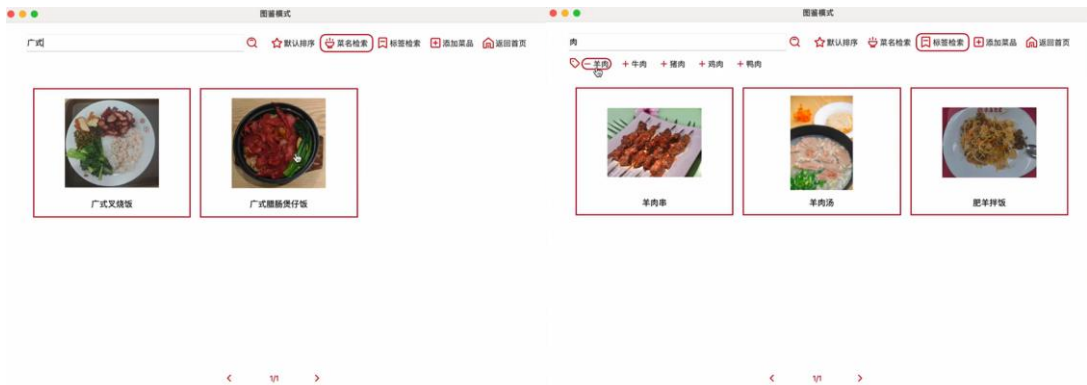
图鉴模式最初呈现为一页六张小的菜品 card，下方有页码并可以进行左右翻页的功能。

图鉴模式可以用来精准的搜索某一项菜品，也可以根据一些 tag 来呈现符合这些 tag 的所有菜品，同时这里可以进行翻页的功能。

排序方式：我们对于现在检索到的菜品有两种排序的方式，一种是“默认排序”，是根据加入菜品的顺序来排序这些菜品，另一种为“星级排序”，是根据检索到的菜品的星级由高到低的排序。



检索方法：我们主要提供两种检索的方法，一种是“菜名检索”：可以根据输入的菜名检索相关的菜品，并呈现他们的图片，此种检索方法适用于已经想好了希望使用什么菜品，来做提前的了解；一种是“标签检索”：我们对此菜品的分类方法主要是给菜品一定的 tag，在搜索框输入有关的字词，可以提供有之相关的一些 tag，用户可以选择自己喜欢的 tag，并且点亮它，缩小菜品的范围，再次搜索别的 tag 再次点亮，可以继续缩小菜品的范围，在这样的过程中，用户可以越来越接近自己喜欢的菜品。



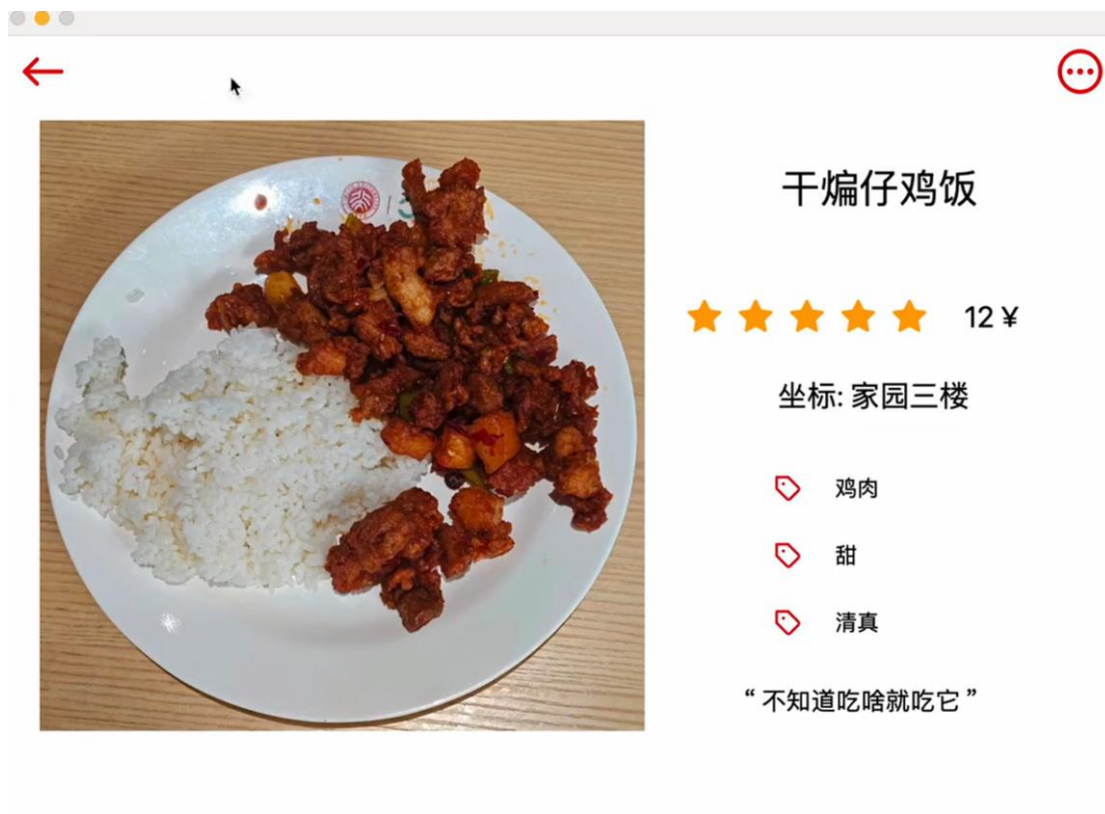
此外，在图鉴模式的界面，我们提供的六张小的菜品 card，点击之后即可进入一个菜品的 detailed card，接下来我会详细介绍 detailed card 中的一些功能以及内容。

最后图鉴模式界面会有一个添加菜品的功能，用户可以在本地添加菜品，并且上传图片，接下来会详细介绍。

### 3. detailed\_card

左上角的返回按键可以返回上一级的界面。

Card 中我们会提供菜品的图片，菜名，星级，价格，坐标（位置，在某食堂几楼），并且会提供三个点赞次数最多的 tag，和一个点赞次数最多的评论。



右上角是一个更多选项的按键，点击即可了解更多与这个菜品相关的信息。

更多界面的内容：在这里可以为菜品添加新的 tag，可以给菜品写新的评论，也可以为菜品重新进行评分，同时可以为已有的 tag，评论进行点赞和删除的操作，从而会影响 tag 和评论的 times 和存在性。

在更新 times 和存在性之后 detailed card 也会相应的更新其上显示的内容。

## 4. 添加菜品

后续如果需要添加更多的菜品便可以使用这个功能，点击添加菜品既可以开始使用。



首先是必填的选项：菜名，价格，位置，上传图片。

然后有一些选填的选项：时段、菜系、菜品类别、主要成分、烹饪方法、口味口感、忌口饮食限制、星级评分，其中我们为用户提供了一些已经有的选项 tag，同时用户也可以自行添加其他的 tag，以后这些 tag 也会加入我们的数

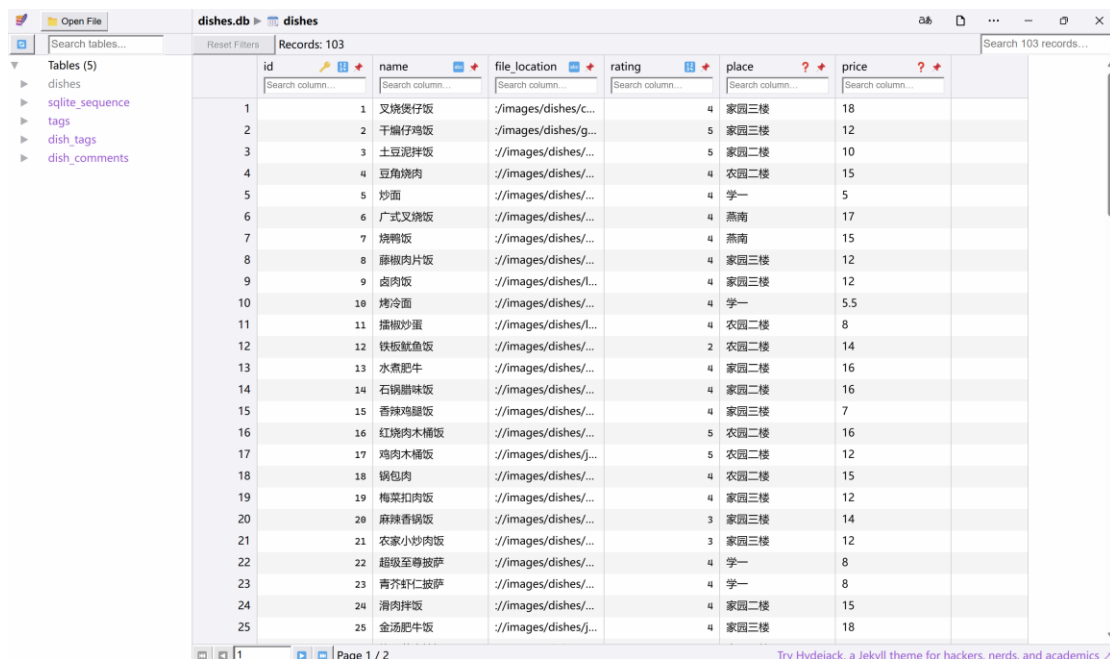
数据库中作为 tag 使用。  
最后点击 ok 就会将菜品加入我们的数据库了。

## 二、项目各模块与类设计细节

### 1.有关项目数据存储的数据库建立

我们使用了 sqlite3 的数据库，并且使用 qt 的方法去对数据库进行一系列的操作。

我们一共建立了四张表格 第一张是 dishes 表 其中存储了菜品的 id、name、file location、rating、place、price 六方面的信息；第二张是 tags 表，其中存储了 tag 的 id、name、type（这里 type 指的是这个 tag 是什么类型的，如 0 表示位置信息，1 表示时段信息等）的信息；第三章是 dish tags 表，其中存储了 dish id、tag id、times（点赞数）的信息，同时使用了键值对的方法联系了这张表中的 dish 和 tag 的 id 与前两张表中的 dish 和 tag 的 id 的关系，同时这张表内也有着 dish id 和 tag id 之间的键值对关系；第四张是 dish comments 表，这张表内存储了 id、dish id、comment、times 的信息，同时也有着这张表中的 dish id 与第一张表的 dish 的 id 的键值对关系。



id	name	file_location	rating	place	price
1	叉烧煲仔饭	:/images/dishes/c...	4	家园三楼	18
2	干扁仔鸡饭	:/images/dishes/g...	5	家园三楼	12
3	土豆泥拌饭	:/images/dishes/...	5	家园二楼	10
4	豆角烧肉	:/images/dishes/...	4	农园二楼	15
5	炒面	:/images/dishes/...	4	学一	5
6	广式叉烧饭	:/images/dishes/...	4	燕南	17
7	烧鸭饭	:/images/dishes/...	4	燕南	15
8	藤椒肉片饭	:/images/dishes/...	4	家园三楼	12
9	卤肉饭	:/images/dishes/l...	4	家园三楼	12
10	炖冷面	:/images/dishes/...	4	学一	5.5
11	擂椒炒蛋	:/images/dishes/l...	4	农园二楼	8
12	铁板鱿鱼饭	:/images/dishes/...	2	农园二楼	14
13	水煮肥牛	:/images/dishes/...	4	家园二楼	16
14	石锅腊味饭	:/images/dishes/...	4	家园二楼	16
15	香辣鸡腿饭	:/images/dishes/...	4	家园三楼	7
16	红烧肉木桶饭	:/images/dishes/...	5	农园二楼	16
17	鸡肉木桶饭	:/images/dishes/f...	5	农园二楼	12
18	锅包肉	:/images/dishes/...	4	农园二楼	15
19	梅菜扣肉饭	:/images/dishes/...	4	家园三楼	12
20	麻辣香锅饭	:/images/dishes/...	3	家园三楼	14
21	农家小炒肉饭	:/images/dishes/...	3	家园三楼	12
22	超级至尊披萨	:/images/dishes/...	4	学一	8
23	青芥虾仁披萨	:/images/dishes/...	4	学一	8
24	滑肉拌饭	:/images/dishes/...	4	家园二楼	15
25	金汤肥牛饭	:/images/dishes/f...	4	家园三楼	18

### 2. 主界面

主界面使用的是 Widght 类，并且进行一些更多的设计，背景图片是我们自己设计的符合北

大红的一张精美图片。

从主界面进入各个子界面是建立父子界面的关系，以方便进行各个界面之间的跳转。

```
class Widget : public QWidget
{
    Q_OBJECT

public:
    Widget(QWidget *parent = nullptr);
    ~Widget();

private slots:
    void turntomh();
    void turntotj();

private:
    Ui::Widget *ui;
};
```

### 3. “盲盒界面”

“盲盒界面”使用 UiMangheWindow 类，主要是以功能为主，画面上与整体风格接近。

使用了 getIdVector、random\_tags、check\_add、load\_dishes、showDetail、generate\_dish、home 这一系列函数，下面我们会介绍主要函数的功能。

getIdVector：获得数据库中的有关不同 tag 的信息，并且暂时存储在类中。

random\_tags：随机生成一批 tag，来为用户提供一些选择，点击换一批则会再次调用这个函数。

check\_add：用来检测某个 tag 是否被用户选中的函数。

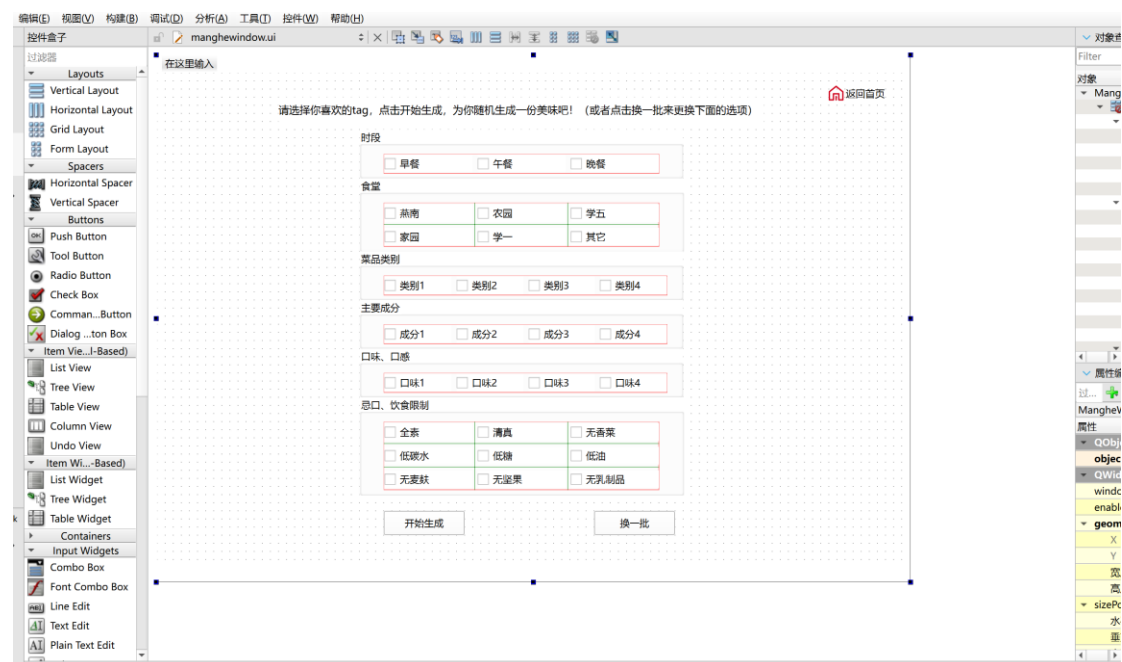
load\_dishes 根据用户选择的 tag 进行几次筛选工作并且最终导入这些菜品形成一个列表。

筛选的方式较为复杂：首先筛选出符合时段、食堂、忌口饮食限制所有信息的菜品，然后选择的不同菜品类别、主要成分、口味口感的菜品进行加入，此时由于某些 tag 拥有 times，我们会根据 times 来调整一个菜品的权重。总而言之，一个菜品被选入必须要符合时段、食堂、忌口饮食限制这三个方面的信息，同时其权重=含有的用户选择的 tag 的 times 之和。

generate\_dish：这里会根据上面加入的菜品及其权重真随机的产生一个菜品作为本次生成的结果。

showDetail：用于打开最后生成的菜品的 detailed\_card。

Home：用于返回主界面的函数。



### 3.图鉴界面

主要是设计了一个 TujianWindow 的类，其中主要设计了 selectTag、Sort、Model、resetCards 这些函数，我们还设计了 TagButton、PreviewCards、NotFoundWidget、MessageButton 这几个类来完成一些辅助性质的功能。

TujianWindow：这个类中主要设计与主界面的接口，以及设置和其余各种界面的关系。

selectTag：这个函数主要是用来根据已经选择的 tag 来选择出相应的菜品。



Sort：这个函数的主要作用是选择排序的方式，来改变下面的 card 的排序。

Model：这个函数的主要作用是用来改变检索菜品的方法，点击以改变是根据菜名还是根据 tag 来筛选菜品。

resetCards：这个是用来解决在每一次检索过后更新下方的 card。

TagButton：这个类是用来点击选择 tag 的类。

PreviewCards：这个图鉴界面用来呈现六个预览图片的类。

NotFoundWidget：这个是用来呈现检索信息过强而没有符合条件的菜品。

MessageButton：设计这个类是为了使得点击的过程中可以传递一个信息，而设计了一个 button 的类。

```
class TagButton:public QPushButton{
    Q_OBJECT
public:
    TagButton(const QString& text);
    ~TagButton();
    int tagId;
signals:
    void sendSelf(TagButton*);
public slots:
    void clickedMessage();
};

class SelectedTagButton:public QPushButton{
    Q_OBJECT
public:
    SelectedTagButton(const QString& text);
    ~SelectedTagButton();
    int index;
signals:
    void sendIndex(const int& num);
public slots:
    void clickedMessage();
};

class TujianWindow : public QMainWindow
{
    Q_OBJECT

public:
    explicit TujianWindow(QWidget *parent = nullptr);
    ~TujianWindow();
    int page;
    int totalPage;
    int searchMode;
    int sortflag=0;
    QVector<QString>selectedNames;
    QVector<int>selectedIds;
    Widget* wparent;

public slots:
    void addCards();
    void resetCards();
    void updateLabelSearch(const QString &text);
};
```

## 4.detailed\_card

这里我们设计了 DetailCard 这个类其中主要有 reset、moreThings、backward 这几个函数,同时设计了 ListItemWidget、ListItemWidget\_2、StarButton 这几个类来辅助完成功能。

两个 mode :一个是用于图鉴模式使用的 mode ,另一个是用于“盲盒模式”使用的 mode。

Reset : 这个类主要是在更新了一个菜品的相关信息之后会重新生成其 detailed\_card 来表现其新的信息。

moreThings : 这个是从 detailed\_card 进入其详情的界面,这里可以清楚的看见其所有相关 tag 的 times 和所有相关评论的 times。

Backward : 这个函数用于退回上一个界面。

ListItemWidget : 这个类是用来呈现 tag 的界面。有 increaseLikes、deleteSelf 这两个函数分别完成增加 times 和删除的功能。

ListItemWidget\_2 : 这个类则是为了呈现评论的界面。函数同上。

StarButton : 这个类是我们使用评分所用的星星,在呈现评分时是静态的,而在评分的过程中则是可以左右移动的鼠标来调整评分的星星数目。主要通过 leaveEvent、enterEvent 这两个函数来完成左右移动时点亮和暗灭的功能。



```

#ifndef DETAILCARD_H
#define DETAILCARD_H

#include <QWidget>
#include <QStandardItemModel>
#include "tujianwindow.h"

namespace Ui {
class DetailCard;
}

class DetailCard : public QWidget
{
    Q_OBJECT

public:
    explicit DetailCard(const int& _dish_id,int _mode,TujianWindow* tjparent=nullptr,QWidget *parent = nullptr);
    ~DetailCard();

public slots:
    void reset();

private slots:
    void moreThings();
    void backward();

private:
    TujianWindow* tjWindow;
    Ui::DetailCard *ui;
    QStandardItemModel* model;
    int mode;//mode为0是图鉴模式，mode为1是盲盒模式
    int dish_id;
};

#endif // DETAILCARD_H

```

## 5.add\_dialog

这里主要设计了 AddDialog 这个类 ,主要是使用了 selectFile、copyImageFile、addDish、addTag、addDishTagRelation、getIdByName、addAction、star 来实现其功能。

SelectFile、copyImageFile：这两个函数主要用来获取新增的图片以及新增图片的地址和图片的信息。

addDish：主要实现增加菜品的功能。

addTag：主要实现为菜品增加 tag 的功能。

addDishTagRelation：主要产生菜品和 tag 的关系。

getIdByName：根据一个 name 来获得其 id。

addAction：用来增加信息的总函数。

Star：用于评星级，及完善 StarButton 类的功能。

```

class AddDialog : public QDialog
{
    Q_OBJECT

public:
    explicit AddDialog(QWidget *parent = nullptr);
    ~AddDialog();
    int scores;

signals:
    void addDone();

public slots:
    void Selecteddisconnect();
private:
    Ui::AddDialog *ui;

private slots:
    void selectFile();
    void addActions();
    void starZero();
    void starOne();
    void starTwo();
    void starThree();
    void starFour();
    void starFive();
    //void on_chbxLunch_stateChanged(int arg1);
    //void on_chbkDinner_clicked(bool checked);
};

```

### 三、小组成员分工情况

赵天豪：负责“盲盒部分”的功能以及界面设计，相应部分的大部分代码，写初期报告和最终报告。

张嘉诚 负责图鉴模式的功能以及界面设计 相应部分的大部分代码 完成视频录制及上传，维护 github 上的工程，以及联系助教。

连昊：负责程序主界面设计，整体框架链接以及解决连接中出现的一些 bug，负责了大多数菜品图片及信息的采集工作，实现了 mac 和 windows 的兼容。

## 四、项目总结与反思

### 1. 总结

回顾整个项目的历程：从最初三人实现组队，并且决定叫汪汪队（是为了能够立大功）。再到第一次讨论时，考虑选题问题，在讨论后，决定从我们生活的实际角度出发，去做一个实用的程序来解决我们日常到了饭点却不知道应该吃什么的问题，并且我们思考出了盲盒模式这一较为有新意的功能，我们形成了最初的报告，同时也基本完成了我们的计划与分工。最初由连昊和张嘉诚两位同学一起设计了主界面，之后张嘉诚同学完成了图鉴模式的功能，建立了数据库，为这个项目打下了主要的基础；接着由赵天豪同学完成了“盲盒模式”的功能并且和之前的代码进行了兼容和连接的工作，正是在这个阶段我们发现了 mac 和 windows 系统不同版本的 qt 的兼容问题，连昊同学了解并且解决了这一问题，项目得以继续进展；之后大多数代码完成之后，连昊同学将我们收集的大量数据统一加入我们的数据库，并且完善了代码中的一些细节问题；中期展演，张嘉诚和赵天豪一起录制了视频，并且通过 B 站的方式交给了助教学长，虽然没有被选中展演，我们也做了充足的准备来迎接这一次的工作；中期过后我们主要是完成代码一些细节部分的维护和简化，同时对于产生的一些 bug 进行解决，这个过程中连昊同学做出了很大的贡献；最后赵天豪同学完成了这一次的报告，并将各种细节在报告中介绍，至此，这个程序的设计完整的结束。

关于小组中的个人的提升：

赵天豪：我在本次 qt 大作业中首先认识了又一位信科的同学（连昊同学），同时我们一起合作增进了友谊，我们在别的学科方面也能形成互相帮助。第一次接触 qt，我学会了很多 qt 的语法和 qt\_creator 工具的使用，同时对 sqlite 数据库的使用让我了解了更多工具的使用，还有在 GitHub 上的工程开发的流程。最重要的是我在本次工程中了解到了如何和别人一起开发，如何将自己的代码和别人的连接起来，如何合理分工并且协作完成一个任务。在此特

别感谢我的两个朋友，给了我很多帮助，让我领会团队完成一个项目的方法。

张嘉诚 :在编写程序图鉴模式的过程中 ,困难主要集中在为所设计的功能提供高效简洁的 UI 界面 ,尽可能为用户提供舒适的交互体验。而 QT 中的 widget 组件种类繁多 ,有时还需要进行重写来满足要求 ;组件之间的嵌套关系、交互顺序、signal-slot 连接 ,都需要进行深思熟虑。除此之外 ,对于菜品的标签 ,程序授予了使用者比较高的自由 ,使用者可以自行创建新的标签 ,也可以为菜品添加任意的标签 ,这就导致在一段时间的使用之后可能会发生 “同义标签过多” 、 “菜品标签不准确” 等问题 ,造成用户的检索困难。这就需要开发者定期对标签进行维护 ,耗时耗力。这一问题有待解决。

连昊 :这次大作业是我继黑白棋后第二次用 QT 进行项目开发 ,相比上次对功能的简单实现 ,在这次大作业的开发过程中 ,我更进一步认识了 C++ 面向对象的开发思想 ,同时也对协作开发有了更深的认识。此外 ,在开发过程中 ,我学习了数据库的用法以及跨系统兼容的优化 ,收获良多。

## 2.反思

在本次开发过程中我们也存在着一些问题 我们在项目的最初没有很明确的分工导致我们最初的进展显得有些缓慢 ,后期的规划和分工使得进展加快很多 ;之后的开发时 ,我们在有些时候没有做好代码之间的连结工作 ,使得阅读别人的代码成为一个很重大的问题 ,因此我们更加注重在代码中补充注释和对变量的合理命名 ,未来如果有机会应该制定一个统一的命名标准和写注释的标准 ;中期发现我们的项目在新意上也并不是很突出 ;同时开发过程中发现因为对 qt 的了解不足而使得效率较低 ,未来应该提早多多学习各种工具的使用以及更多的知识 ,提高自学的能力。