

My Project

1.0

Создано системой Doxygen 1.9.6

1 Иерархический список классов	1
1.1 Иерархия классов	1
2 Алфавитный указатель классов	3
2.1 Классы	3
3 Список файлов	5
3.1 Файлы	5
4 Классы	7
4.1 Класс Dish_change_form	7
4.1.1 Подробное описание	7
4.1.2 Конструктор(ы)	8
4.1.2.1 Dish_change_form() [1/3]	8
4.1.2.2 Dish_change_form() [2/3]	8
4.1.2.3 Dish_change_form() [3/3]	8
4.1.3 Методы	8
4.1.3.1 make_order	9
4.2 Класс Extra_ingredient	9
4.2.1 Подробное описание	10
4.2.2 Конструктор(ы)	10
4.2.2.1 Extra_ingredient() [1/2]	10
4.2.2.2 Extra_ingredient() [2/2]	10
4.3 Класс Log_in_accant	10
4.3.1 Подробное описание	11
4.3.2 Конструктор(ы)	11
4.3.2.1 Log_in_accant()	11
4.3.3 Методы	12
4.3.3.1 successful_login	12
4.4 Структура Order	12
4.4.1 Подробное описание	13
4.5 Класс OrderWindow	13
4.5.1 Подробное описание	13
4.5.2 Конструктор(ы)	13
4.5.2.1 OrderWindow()	13
5 Файлы	15
5.1 Файл C:/Coursework_3/field_kitchen/dish_change_form.cpp	15
5.1.1 Подробное описание	15
5.2 Файл C:/Coursework_3/field_kitchen/dish_change_form.h	15
5.2.1 Подробное описание	16
5.3 dish_change_form.h	16
5.4 Файл C:/Coursework_3/field_kitchen/log_in_accant.cpp	17
5.4.1 Подробное описание	17
5.5 Файл C:/Coursework_3/field_kitchen/log_in_accant.h	18

5.5.1 Подробное описание	18
5.6 log_in_accaunt.h	18
5.7 Файл C:/Coursework_3/field_kitchen/main.cpp	19
5.7.1 Подробное описание	19
5.8 Файл C:/Coursework_3/field_kitchen/orderwindow.h	19
5.8.1 Подробное описание	20
5.9 orderwindow.h	20
Предметный указатель	23

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

Order	12
QDialog	
Dish_change_form	7
Log_in_accant	10
QMainWindow	
OrderWindow	13
QWidget	
Extra_ingredient	9

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

Dish_change_form	Класс, реализующий диалоговое окно, для изменения состава блюда	7
Extra_ingredient	Класс виджета для получения информации о дополнительных ингредиентах . . .	9
Log_in_accaunt	Класс диалогового окна для входа в аккаунт	10
Order	Структура, для хранения информации о заказанном блюде	12
OrderWindow	Класс, главную форму в которой происходит заказ блюд, просмотр корзины, просмотр истории заказов с аккаунта	13

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

C:/Coursework_3/field_kitchen/dish_change_form.cpp	
Файл с определениями классов <code>Dish_change_form</code> и <code>Extra_ingredient</code>	15
C:/Coursework_3/field_kitchen/dish_change_form.h	
Файл с объявлениями классов <code>Dish_change_form</code> и <code>Extra_ingredient</code> и структурой <code>Order</code>	15
C:/Coursework_3/field_kitchen/log_in_accaunt.cpp	
Файл с определением класса <code>Log_in_accaunt</code>	17
C:/Coursework_3/field_kitchen/log_in_accaunt.h	
Файл с объявлением класса <code>Log_in_accaunt</code>	18
C:/Coursework_3/field_kitchen/main.cpp	
Файл с инициализацией приложения	19
C:/Coursework_3/field_kitchen/orderwindow.h	
Файл определяет класс <code>OrderWindow</code>	19

Глава 4

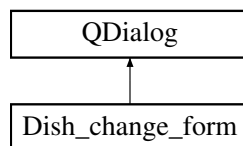
Классы

4.1 Класс Dish_change_form

Класс, реализующий диалоговое окно, для изменения состава блюда

```
#include <dish_change_form.h>
```

Граф наследования:Dish_change_form:



Сигналы

- void `make_order` (`Order` ord)
Сигнал испускаемый при успешном завершении диалогового окна

Открытые члены

- `Dish_change_form` (`QWidget` *parent=nullptr)
Конструктор по умолчанию.
- `Dish_change_form` (`QString` d_name, `QWidget` *parent=nullptr)
Конструктор по названию блюда.
- `Dish_change_form` (`Order` ord, `QWidget` *parent=nullptr)
Конструктор по готовому заказу. Используется для редактирования сделанных заказов из корзины
- `~Dish_change_form` ()
Деструктор.

4.1.1 Подробное описание

Класс, реализующий диалоговое окно, для изменения состава блюда

4.1.2 Конструктор(ы)

4.1.2.1 Dish_change_form() [1/3]

```
Dish_change_form::Dish_change_form (
    QWidget * parent = nullptr ) [explicit]
```

Конструктор по умолчанию.

Аргументы

parent	указатель на родительский виджет.
--------	-----------------------------------

4.1.2.2 Dish_change_form() [2/3]

```
Dish_change_form::Dish_change_form (
    QString d_name,
    QWidget * parent = nullptr )
```

Конструктор по названию блюда.

Аргументы

d_name	Название выбранного блюда.
--------	----------------------------

4.1.2.3 Dish_change_form() [3/3]

```
Dish_change_form::Dish_change_form (
    Order ord,
    QWidget * parent = nullptr )
```

Конструктор по готовому заказу. Используется для редактирования сделанных заказов из корзины

Аргументы

ord	Struct Order по данным из которой формируется форма.
-----	--

4.1.3 Методы

4.1.3.1 make_order

```
void Dish_change_form::make_order (
    Order ord ) [signal]
```

Сигнал испускаемый при успешном завершении диалогового окна

Аргументы

ord	Заказ передающийся в главную форму.
-----	-------------------------------------

Объявления и описания членов классов находятся в файлах:

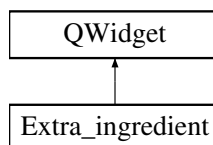
- C:/Coursework_3/field_kitchen/dish_change_form.h
- C:/Coursework_3/field_kitchen/dish_change_form.cpp

4.2 Класс Extra_ingredient

Класс виджета для получения информации о дополнительных ингредиентах.

```
#include <dish_change_form.h>
```

Граф наследования:Extra_ingredient:



Открытые члены

- [Extra_ingredient](#) (QWidget *parent=nullptr)
Конструктор по умолчанию.
- [Extra_ingredient](#) (QString ing_name, QString ing_price, int max_count, QWidget *parent=nullptr)
Конструктор по входным данным.
- ~Extra_ingredient ()
Деструктор.

Открытые атрибуты

- QLabel * name_lable
Указатель на надпись с названием.
- QLabel * price_lable
Указатель на надпись с ценой порции.
- QSpinBox * spinbox
Указатель на Spinbox с помощью которого задается количество заказанных порций ингредиента.
- QHBoxLayout * h_lay
Указатель на слой виджета.

4.2.1 Подробное описание

Класс виджета для получения информации о дополнительных ингредиентах.

4.2.2 Конструктор(ы)

4.2.2.1 Extra_ingredient() [1/2]

```
Extra_ingredient::Extra_ingredient (
    QWidget * parent = nullptr ) [explicit]
```

Конструктор по умолчанию.

Аргументы

parent	Указатель на родительский виджет.
--------	-----------------------------------

4.2.2.2 Extra_ingredient() [2/2]

```
Extra_ingredient::Extra_ingredient (
    QString ing_name,
    QString ing_price,
    int max_count,
    QWidget * parent = nullptr )
```

Конструктор по входным данным.

Аргументы

ing_name	Название ингредиента.
ing_price	Цена порции ингредиента.
max_count	Максимальное число порций ингредиента доступное для этого блюда.
parent	Указатель на родительский виджет.

Объявления и описания членов классов находятся в файлах:

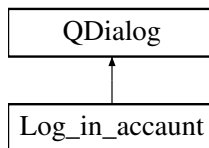
- C:/Coursework_3/field_kitchen/[dish_change_form.h](#)
- C:/Coursework_3/field_kitchen/[dish_change_form.cpp](#)

4.3 Класс Log_in_accaunt

Класс диалогового окна для входа в аккаунт.

```
#include <log_in_accaunt.h>
```

Граф наследования:Log_in_accaunt:



Сигналы

- void [successful_login](#) (QString client_id)
Сигнал успешной регистрации.

Открытые члены

- [Log_in_accaunt](#) (QWidget *parent=nullptr)
Конструктор по умолчанию.
- ~Log_in_accaunt ()
Деструктор.

Защищенные члены

- void keyPressEvent (QKeyEvent *event)
Обработка нажатия на клавиши во время выполнения формы.

4.3.1 Подробное описание

Класс диалогового окна для входа в аккаунт.

4.3.2 Конструктор(ы)

4.3.2.1 Log_in_accaunt()

```
Log_in_accaunt::Log_in_accaunt (
    QWidget * parent = nullptr ) [explicit]
```

Конструктор по умолчанию.

Аргументы

parent	указатель на родительский виджет.
--------	-----------------------------------

4.3.3 Методы

4.3.3.1 successful_login

```
void Log_in_accaunt::successful_login (
    QString client_id ) [signal]
```

Сигнал успешной регистрации.

Аргументы

client← _id	id аккаунта с которого осуществлен вход.
----------------	--

Объявления и описания членов классов находятся в файлах:

- C:/Coursework_3/field_kitchen/log_in_accaunt.h
- C:/Coursework_3/field_kitchen/log_in_accaunt.cpp

4.4 Структура Order

Структура, для хранения информации о заказанном блюде.

```
#include <dish_change_form.h>
```

Открытые члены

- Order ()=default
Конструктор по умолчанию.
- ~Order ()=default
Деструктор по умолчанию.

Открытые атрибуты

- QString dish_name
Имя блюда в заказе.
- QList< QString > deleted_ingredients
Список удаленных ингредиентов.
- QList< QPair< QString, int > > extra_ingredients
Список дополнительных ингредиентов ингредиентов.
- double price

4.4.1 Подробное описание

Структура, для хранения информации о заказанном блюде.

Объявления и описания членов структуры находятся в файле:

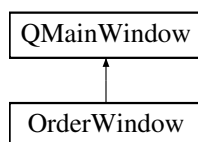
- C:/Coursework_3/field_kitchen/[dish_change_form.h](#)

4.5 Класс OrderWindow

Класс, главную форму в которой происходит заказ блюд, просмотр корзины, просмотр истории заказов с аккаунта.

```
#include <orderwindow.h>
```

Граф наследования:OrderWindow:



Открытые члены

- [OrderWindow](#) (QWidget *parent=nullptr)
Конструктор по умолчанию.
- ~OrderWindow ()
Деструктор.

4.5.1 Подробное описание

Класс, главную форму в которой происходит заказ блюд, просмотр корзины, просмотр истории заказов с аккаунта.

4.5.2 Конструктор(ы)

4.5.2.1 OrderWindow()

```
OrderWindow::OrderWindow (  
    QWidget * parent = nullptr )
```

Конструктор по умолчанию.

Аргументы

parent	указатель на родительский виджет.
--------	-----------------------------------

Объявления и описания членов классов находятся в файлах:

- C:/Coursework_3/field_kitchen/[orderwindow.h](#)
- C:/Coursework_3/field_kitchen/orderwindow.cpp

Глава 5

Файлы

5.1 Файл C:/Coursework_3/field_kitchen/dish_change_form.cpp

Файл с определениями классов [Dish_change_form](#) и [Extra_ingredient](#).

```
#include "dish_change_form.h"  
#include "ui_dish_change_form.h"
```

5.1.1 Подробное описание

Файл с определениями классов [Dish_change_form](#) и [Extra_ingredient](#).

Авторы

Алексеев А.В.

Заметки

Ответственный: Полевой Д.В.

5.2 Файл C:/Coursework_3/field_kitchen/dish_change_form.h

Файл с объявлениями классов [Dish_change_form](#) и [Extra_ingredient](#) и структурой [Order](#).

```
#include <QDialog>  
#include <QSqlDatabase>  
#include <QSqlQuery>  
#include <QDebug>  
#include <QLabel>  
#include <QCheckBox>  
#include <QSpinBox>  
#include <QSqlError>  
#include <QHBoxLayout>  
#include <QString>  
#include <QList>
```

Классы

- struct [Order](#)
Структура, для хранения информации о заказанном блюде.
- class [Extra_ingredient](#)
Класс виджета для получения информации о дополнительных ингредиентах.
- class [Dish_change_form](#)
Класс, реализующий диалоговое окно, для изменения состава блюда

5.2.1 Подробное описание

Файл с объявлениями классов [Dish_change_form](#) и [Extra_ingredient](#) и структурой [Order](#).

Авторы

Алексеев А.В.

Заметки

Ответственный: Полевой Д.В.

5.3 dish_change_form.h

[См. документацию.](#)

```
00001
00005
00006 #ifndef DISH_CHANGE_FORM_H
00007 #define DISH_CHANGE_FORM_H
00008
00009 #include <QDialog>
00010 #include <QSqlDatabase>
00011 #include <QSqlQuery>
00012 #include <QDebug>
00013 #include <QLabel>
00014 #include <QCheckBox>
00015 #include <QSpinBox>
00016 #include <QSqlError>
00017 #include <QHBoxLayout>
00018 #include <QString>
00019 #include <QList>
00020
00021 namespace Ui {
00022 class Dish_change_form;
00023 class Extra_ingredient;
00024 }
00025
00027 struct Order {
00029     QString dish_name;
00031     QList<QString> deleted_ingredients;
00033     QList<QPair<QString, int>> extra_ingredients;
00034     double price;
00036     Order() = default;
00038     ~Order() = default;
00039 };
00040
00042 class Extra_ingredient : public QWidget
00043 {
00044     Q_OBJECT
00045 public:
00048     explicit Extra_ingredient(QWidget *parent = nullptr);
00054     Extra_ingredient(QString ing_name, QString ing_price, int max_count, QWidget *parent = nullptr);
00056     ~Extra_ingredient();
00058     QLabel* name_label;
00060     QLabel* price_label;
00063     QSpinBox* spinbox;
00065     QHBoxLayout* h_lay;
00066 };
```

```

00067
00069 class Dish_change_form : public QDialog
00070 {
00071     Q_OBJECT
00072
00073 public:
00074     explicit Dish_change_form(QWidget *parent = nullptr);
00075     Dish_change_form(QString d_name, QWidget *parent = nullptr);
00083     Dish_change_form(Order ord, QWidget *parent = nullptr);
00085     ~Dish_change_form();
00086
00087
00088 private slots:
00089     //Нажатия на кнопки
00091     void on_select_Button_clicked();
00093     void on_back_Button_clicked();
00094     // Закрытия окна
00096     void on_Dish_change_form_accepted();
00098     void on_Dish_change_form_rejected();
00099     //Изменение цены
00101     void change_price();
00102
00103
00104 private:
00106     Ui::Dish_change_form *ui;
00107
00109     double base_price;
00111     QString dish_name;
00113     QSqlDatabase db;
00115     QList<QLabel*> ingredients;
00117     QList<QCheckBox*> delete_ingredients;
00120     QList<Extra_ingredient*> extra_ingredients;
00122     QVBoxLayout *optional_lay;
00124     QVBoxLayout *ingredient_lay;
00126     QVBoxLayout *extra_lay;
00129     void dish_start();
00130
00131 signals:
00134     void make_order(Order ord);
00135 };
00136
00137
00138 #endif // DISH_CHANGE_FORM_H

```

5.4 Файл C:/Coursework_3/field_kitchen/log_in_accaunt.cpp

Файл с определением класса [Log_in_accaunt](#).

```

#include "log_in_accaunt.h"
#include "ui_log_in_accaunt.h"

```

5.4.1 Подробное описание

Файл с определением класса [Log_in_accaunt](#).

Авторы

Алексеев А.В.

Заметки

Ответственный: Полевой Д.В.

5.5 Файл C:/Coursework_3/field_kitchen/log_in_accaunt.h

Файл с объявлением класса [Log_in_accaunt](#).

```
#include <QDialog>
#include <QSqlDatabase>
#include <QSqlQuery>
#include <QKeyEvent>
#include <QRegularExpression>
#include <QRegularExpressionValidator>
```

Классы

- class [Log_in_accaunt](#)
Класс диалогового окна для входа в аккаунт.

5.5.1 Подробное описание

Файл с объявлением класса [Log_in_accaunt](#).

Авторы

Алексеев А.В.

Заметки

Ответственный: Полевой Д.В.

5.6 log_in_accaunt.h

[См. документацию.](#)

```
00001
00005
00006 #ifndef LOG_IN_ACCAUNT_H
00007 #define LOG_IN_ACCAUNT_H
00008
00009 #include <QDialog>
00010 #include <QSqlDatabase>
00011 #include <QSqlQuery>
00012 #include <QKeyEvent>
00013 #include <QRegularExpression>
00014 #include <QRegularExpressionValidator>
00015
00016 namespace Ui {
00017 class Log_in_accaunt;
00018 }
00019
00021 class Log_in_accaunt : public QDialog
00022 {
00023     Q_OBJECT
00024
00025 public:
00028     explicit Log_in_accaunt(QWidget *parent = nullptr);
00030     ~Log_in_accaunt();
00031
00032 private slots:
00034     void on_Accept_Button_clicked();
00035
00037     void on_Log_in_accaunt_rejected();
00038
```

```

00040 void on_back_Button_clicked();
00041
00043 void on_registration_Button_clicked();
00044
00045 private:
00047 Ui::Log_in_accaunt *ui;
00049 QSqlDatabase db;
00051 QSqlQuery* query;
00053 bool is_registration = false;
00054
00055 protected:
00057 void keyPressEvent(QKeyEvent *event);
00058
00059 signals:
00062 void successful_login(QString client_id);
00063 };
00064
00065 #endif // LOG_IN_ACCAUNT_H

```

5.7 Файл C:/Coursework_3/field_kitchen/main.cpp

Файл с инициализацией приложения

```

#include "orderwindow.h"
#include <QApplication>

```

Функции

- int main (int argc, char *argv[])

5.7.1 Подробное описание

Файл с инициализацией приложения

Авторы

Алексеев А.В.

Заметки

Ответственный: Полевой Д.В.

5.8 Файл C:/Coursework_3/field_kitchen/orderwindow.h

Файл определяет класс [OrderWindow](#).

```

#include <QMainWindow>
#include <QSqlDatabase>
#include <QSqlQuery>
#include <QDebug>
#include <QLabel>
#include <QHBoxLayout>
#include "dish_change_form.h"
#include <QFont>
#include <QMessageBox>

```

Классы

- class [OrderWindow](#)

Класс, главную форму в которой происходит заказ блюд, просмотр корзины, просмотр истории заказов с аккаунта.

5.8.1 Подробное описание

Файл определяет класс [OrderWindow](#).

Файл объявляет класс [OrderWindow](#).

Авторы

Алексеев А.В.

Заметки

Ответственный: Полевой Д.В.

5.9 orderwindow.h

[См. документацию.](#)

```
00001
00005
00006 #ifndef ORDERWINDOW_H
00007 #define ORDERWINDOW_H
00008
00009 #include <QMainWindow>
00010 #include <QSqlDatabase>
00011 #include <QSqlQuery>
00012 #include <QDebug>
00013 #include <QLabel>
00014 #include <QHBoxLayout>
00015 #include "dish_change_form.h"
00016 #include <QFont>
00017 #include <QMessageBox>
00018
00019 QT_BEGIN_NAMESPACE
00020 namespace Ui { class OrderWindow; }
00021 QT_END_NAMESPACE
00022
00025 class OrderWindow : public QMainWindow
00026 {
00027     Q_OBJECT
00028
00029 public:
00032     OrderWindow(QWidget *parent = nullptr);
00034     ~OrderWindow();
00035
00036 private:
00038     void categories_show();
00040     void dishes_show();
00042     void cart_show();
00044     void history_show();
00046     void history_dishes_show();
00047
00049     void database_connect();
00050
00051 private slots:
00052     // Обработка нажатий на кнопки
00054     void on_select_Button_clicked();
00056     void on_back_Button_clicked();
00058     void on_history_Button_clicked();
00060     void on_account_Button_clicked();
00062     void on_cart_finish_Button_clicked();
00063
00064     // Обработка сигналов из сторонних форм
00066     void cart_update(Order const& new_order);
```



```
00068 void cart_add(Order ord);
00070 void del_order_cart();
00072 void successful_login(QString client_id);
00073
00076 void on_listWidget_doubleClicked(const QModelIndex &index);
00077
00078 private:
00080 Ui::OrderWindow *ui;
00082 QSqlDatabase db;
00084 QSqlQuery* query;
00085
00087 bool is_cart = false;
00089 bool is_history = false;
00091 QString history_order = "";
00093 QString select_category = "";
00094
00096 Dish_change_form *dish_select;
00098 QList<Order> cart;
00100 QString client = "NULL";
00101 };
00102 #endif // ORDERWINDOW_H
```


Предметный указатель

C:/Coursework_3/field_kitchen/dish_change_form.cpp,
[15](#)

C:/Coursework_3/field_kitchen/dish_change_form.h,
[15](#), [16](#)

C:/Coursework_3/field_kitchen/log_in_accaunt.cpp,
[17](#)

C:/Coursework_3/field_kitchen/log_in_accaunt.h,
[18](#)

C:/Coursework_3/field_kitchen/main.cpp, [19](#)

C:/Coursework_3/field_kitchen/orderwindow.h,
[19](#), [20](#)

Dish_change_form, [7](#)
 Dish_change_form, [8](#)
 make_order, [8](#)

Extra_ingredient, [9](#)
 Extra_ingredient, [10](#)

Log_in_accaunt, [10](#)
 Log_in_accaunt, [11](#)
 successful_login, [12](#)

make_order
 Dish_change_form, [8](#)

Order, [12](#)
OrderWindow, [13](#)
 OrderWindow, [13](#)

successful_login
 Log_in_accaunt, [12](#)