**Министерство образования, науки и молодежной политики Нижегородской области**

**Государственное бюджетное профессиональное образовательное учреждение**

**«Нижегородский Губернский колледж»**

Методическая комиссия «Информатика и вычислительная техника»

Допущен к защите:

преподаватель

\_\_\_\_\_\_\_\_\_\_\_\_ В.А. Авакян,

\_\_\_\_\_\_\_\_\_\_\_\_ В.В. Циркова

«21» марта 2022 г.

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

ПМ.05 РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ КОМПЬЮТЕРНЫХ СЕТЕЙ

Студент: Горшенин А.А.

Специальность, группа: 09.02.07, 44П

Нижний Новгород

2022 г.

**СОДЕРЖАНИЕ**

[**ВВЕДЕНИЕ** 3](#_Toc98766125)

[**Область применения программы** 3](#_Toc98766126)

[**Цели и задачи профессионального модуля – требования к результатам освоения профессионального модуля** 3](#_Toc98766127)

[**ГЛАВА 1 ПМ 05.01 «РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ»** 6](#_Toc98766128)

[**1.1 Проектирование системы и работа с базой данных** 6](#_Toc98766129)

[**1.2 Оценка качества программных средств** 10](#_Toc98766130)

[**ГЛАВА 2 ПМ 05.02 «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»** 12](#_Toc98766131)

[**2.1 Анализ бизнес-процессов предметной области проекта. Создание структурной схемы проекта. Создание диаграммы модулей. Составление списка артефактов проекта.** 12](#_Toc98766132)

[**2.2 Разработка интегрированной информационной системы** 30](#_Toc98766133)

[**2.3 Отладка и тестирование** 55](#_Toc98766134)

[**ГЛАВА 3 МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ** 60](#_Toc98766135)

[**3.1 Процесс изучения постановки задачи** 60](#_Toc98766136)

[**3.1.1 Постановка задачи** 60](#_Toc98766137)

[Построить математическую модель в общем и частном виде данного класса задач. 60](#_Toc98766138)

[Решить построенную математическую модель методом потенциалов. 60](#_Toc98766139)

[Составить программу реализации метода решения модели. 60](#_Toc98766140)

[Найти оптимальное распределение трех видов механизмов между тремя участками работ. Данные об эффективности использования механизмов конкретного типа на участке работ сведены в таблицу. 60](#_Toc98766141)

[**3.1.2 Входные и выходные показатели задачи** 60](#_Toc98766142)

[**3.2 Составление математической модели задачи в общем и частном виде** 61](#_Toc98766143)

[**3.2.1 Определение типа математической модели** 61](#_Toc98766144)

[**3.2.2 Описание процесса построения математической модели задач линейного программирования в общем виде** 61](#_Toc98766145)

[**3.3 Процесс решения математической модели заданным методом** 64](#_Toc98766146)

[**3.4 Составление программы реализации метода решения ММ** 65](#_Toc98766147)

[**ЗАКЛЮЧЕНИЕ** 67](#_Toc98766148)

[**ПРИЛОЖЕНИЯ** 68](#_Toc98766149)

# **ВВЕДЕНИЕ**

## **Область применения программы**

Программа профессионального модуля является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности 09.02.07 «Информационные системы и программирования».

В части освоения основного вида деятельности «Осуществление интеграции программных модулей» и соответствующих профессиональных компетенций (ПК):

ПК 5.1. Собирать исходные данные для разработки проектной документации на информационную систему;

ПК 5.2. Разрабатывать проектную документацию на разработку информационной системы в соответствии с требованиями заказчика;

ПК 5.3. Разрабатывать подсистемы безопасности информационной системы в соответствии с техническим заданием;

ПК 5.4. Производить разработку модулей информационной системы в соответствии с техническим заданием;

ПК 5.5. Осуществлять тестирование информационной системы на этапе опытной эксплуатации с фиксацией выявленных ошибок кодирования в разрабатываемых модулях информационной системы;

ПК 5.6. Разрабатывать техническую документацию на эксплуатацию информационной системы;

ПК 5.7. Производить оценку информационной системы для выявления возможности ее модернизации.

## **Цели и задачи профессионального модуля – требования к результатам освоения профессионального модуля**

С целью овладения указанным видом профессиональной деятельности и соответствующим профессиональными компетенциями обучающийся в ходе освоения профессионального модуля должен:

Уметь:

* Разрабатывать модули программного обеспечения для мобильных платформ;
* Осуществлять инсталляцию, настройку и обслуживание программного обеспечения компьютерных систем;

Знать:

* Базовые и прикладные информационные технологии;
* Инструментальные средства информационных технологий;
* обеспечение непротиворечивости и целостности данных информационных технологий;
* принципы работы с информационными технологиями.

Первой целью учебной практики является разработка программного модуля для автоматизации части деятельности фирмы, занимающейся продажей масок.

Основой данного программного модуля должна быть реляционная база данных, которая должна хранить следующие сведения:

* Сведения о продуктах;
* Сведения о типах продукта;
* Сведения о истории изменений цены;
* Сведения о стоимости продукта;
* Сведения о распространителях;
* Сведения о истории изменения приоритета распространителя;
* Сведения о типе распространителя;
* Сведения о конечных пунктах сбыта;
* Сведения о поставщиках;
* Сведения о материалах поставщиков;
* Сведения о материалах;
* Сведения о типах материала;
* Сведения о истории количества материала;
* Сведения о материалах продукта.

Должно быть разработано клиентское приложение, обеспечивающее связь между пользователем и БД, а также реализующее следующие функции:

* Добавление, редактирование, удаление и хранение данных;
* Поиск информации, ее фильтрация и сортировка.

Второй целью учебной практики является разработка мобильного приложения для магазина «Намордник», занимающегося продажей масок в условиях пандемии.

Должно быть разработано мобильное приложение, реализующее следующие функции:

* Просмотр данных;
* Поиск, сортировка и фильтрация информации.

# **ГЛАВА 1 ПМ 05.01 «РАЗРАБОТКА ПРОГРАММНЫХ МОДУЛЕЙ»**

## **1.1 Проектирование системы и работа с базой данных**

**Провести анализ предметной области в соответствии с индивидуальным заданием**

Магазин «Намордник» занимается продажей масок, поставляемых поставщиками. Маски распространяются по различным торговым точкам города, которые реализуют товар конечному потребителю.

1. Анализ требований и информационных потребностей

Анализ требований:

База данных должна содержать:

* Сведения о продуктах;
* Сведения о типах продукта;
* Сведения о истории изменений цены;
* Сведения о продажах продукта;
* Сведения о распространителях;
* Сведения о истории изменения приоритета распространителя;
* Сведения о типе распространителя;
* Сведения о конечных пунктах сбыта;
* Сведения о поставщиках;
* Сведения о материалах поставщиков;
* Сведения о материалах;
* Сведения о типах материала;
* Сведения о истории количества материала;
* Сведения о материалах продукта.

1. Выявление информационных объектов и связей между ними.

База данных включает в себя следующие данные:

Распространители:

* ID Распространителя;
* Наименование распространителя;
* Тип распространителя;
* Адрес;
* ИНН;
* КПП;
* Имя директора;
* Телефон;
* Email;
* Логотип;
* Приоритет.

История изменения приоритета распространителя:

* ID Приоритета;
* ID Распространителя;
* Дата изменения;
* Приоритет.

Тип распространителя:

* ID Типа;
* Наименование;
* Изображение.

Материалы:

* ID Материала;
* Наименование;
* Количество в упаковке;
* Количество единиц;
* Количество на складе;
* Минимальное количество;
* Описание;
* Стоимость;
* Изображение;
* ID Типа материала.

История количества материала:

* ID Изменения;
* ID Материала;
* Дата изменения;
* Количество.

Поставляемые материалы:

* ID Материала;
* ID Поставщика.

Тип материала:

* ID Типа;
* Наименование;
* Процент дефектов.

Продукты:

* ID Продукта;
* Наименование;
* ID Типа продукта;
* Артикул;
* Описание;
* Изображение;
* Стоимость;
* Номер цеха производства.
* Минимальная цена распространения.

История изменения стоимости продукта:

* ID Изменения;
* ID Продукта;
* Дата изменения;
* Стоимость.

Материалы продукта:

* ID Продукта;
* ID Материала;
* Количество.

Продажи продукта:

* ID Продажи;
* ID Распространителя;
* ID Продукта;
* Дата продажи;
* Количество.

Тип продукта:

* ID Типа;
* Наименование;
* Процент дефекта.

Конечные пункты сбыта:

* ID Пункта сбыта;
* Наименование;
* Адрес;
* ID Распространителя.

Поставщики:

* ID Поставщика;
* Наименование;
* ИНН;
* Дата начала поставок;
* Рейтинг качества;
* Тип поставщика.

**Создать макеты экранных форм и отчётов**

Макеты экранных форм находятся в «Приложении А».

## **1.2 Оценка качества программных средств**

Тест-план

Нужно протестировать следующие компоненты информационной системы:

* Формы GUI
* База данных

1. Критерии качества
   1. Надежность
   2. Функциональность
   3. Удобство использования
   4. Портативность
   5. Эффективность
2. Документация
   1. Анализ предметной области;
   2. Техническое задание;
   3. Тестовые сценарии.
3. Стратегия тестирования

Тестирование проводится вручную, каждый объект конфигурации будет тестироваться с разными входными данным на предмет проявления различных ошибок. При возникновении ошибок необходимо их задокументировать, а также выписать их решение.

1. Ресурсы

Минимальные системные требования для работы программного продукта должны быть следующими:

* Операционная система Microsoft Windows 7 (SP1);
* Двухъядерный процессор с тактовой частотой-1,8 ГГц;
* Объем оперативной памяти 2ГБ;
* Объем свободного дискового пространства 30ГБ;
* Разрешение монитора 1 280 х 720;
* Видеоадаптер с минимальным разрешением1 280 х 720.

1. Метрики
   1. Надежность – 2

* Отказоустойчивость – 1
* Восстанавливаемость – 1
  1. Функциональность – 3
* Исправность – 1
* Точность – 1
* Безопасность – 1
  1. Удобство использования – 0,5
* Понятность – 0,2
* Простота – 0,3
  1. Портативность – 1
* Удобство установки – 1
  1. Эффективность – 1
* Эффективность использования – 1

# **ГЛАВА 2 ПМ 05.02 «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»**

## **2.1 Анализ предметной области проекта.**

Важной задачей, которую необходимо решить фирме «Намордник» - является предоставление информации о ее продукции конечным потребителям. С целью ее решения было создано мобильное приложение, позволяющее пользователям искать, фильтровать и сортировать товар, а также просматривать о них информацию. заключение договоров на поставку компьютеров различной комплектации, ведение складского учета.

Описание предметной области: Исходя из справочной информации о товаре покупатель имеет возможность предварительно выбрать товар, подходящий ему.

Готовая система должна решать такие задачи как просмотр информации о товарах, его поиск, фильтрация и сортировка.

Разрабатываемое программное обеспечение должно иметь:

* Активность со списком товаров и возможностью поиска, фильтрации и сортировки;
* Активности с подробным описанием товаров.

Система «Намордник» предназначена для просмотра конечным потребителем информации о товаре, возможности его поиска, фильтрации и сортировки.

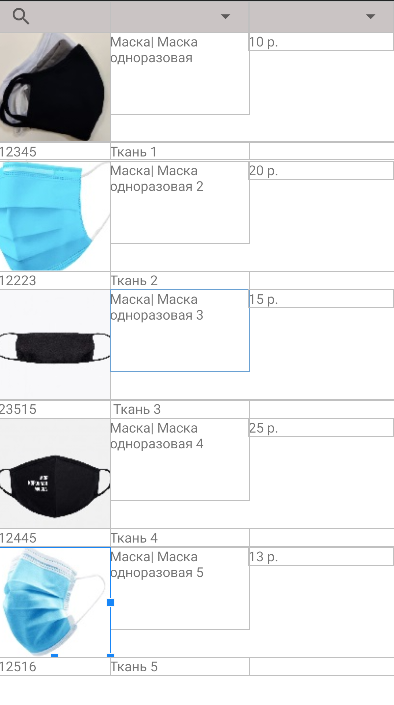
Прототип интерфейсных решений

Рисунок 1 – Активность «Главный экран»

На главном экране помимо кнопок-картинок и краткой информации о товарах находятся управляющие элементы: строка поиска и спиннеры фильтрации и сортировки.

Рисунок 2 – Активность «Страница товара»

На страницах товаров находится подробная информация о них и кнопка возврата на главный экран.

Исходный код проекта

Проект разрабатывался в таких средах как: Android Studio при использовании языка Java.

Количество файлов проекта

В проекте должны содержаться следующие файлы:

* Проект в Android Studio.

Руководство администратора системы

В программе реализованы 6 активностей: «Главный экран» и 5 активностей с информацией о товарах.

На всех активностях реализованы методы для перехода на другие активности:

@Override

public void onClick(View view) {

switch (view.getId()) {

case R.id.imageButton4:

Intent intent2 = new Intent(this, page2.class);

startActivity(intent2);

break;

}

}

На Главном экране реализованы методы для фильтрации и сортировки:

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

//создание строковых массивов, которые будут использоваться спиннерами

String[] data1 = {"Сортировка", "По алфавиту", "Тип продукта", "Артикул", "Цена"};

String[] data2 = {"Материал1", "Материал2", "Материал3", "Материал4", "Материал5"};

/\*\*

\* Called when the activity is first created.

\*/

ImageButton imageButton4;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

imageButton4 = findViewById(R.id.imageButton4);

imageButton4.setOnClickListener(this);

// адаптер

//присвоение спиннерам соответствующих строковых массивов

ArrayAdapter<String> adapter = new ArrayAdapter<String>(this, android.R.layout.simple\_spinner\_item, data2);

adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

ArrayAdapter<String> adapter2 = new ArrayAdapter<String>(this, android.R.layout.simple\_spinner\_item, data1);

adapter2.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

Spinner spinner5 = (Spinner) findViewById(R.id.spinner5);

Spinner spinner6 = (Spinner) findViewById(R.id.spinner6);

spinner5.setAdapter(adapter);

spinner6.setAdapter(adapter2);

// заголовок

spinner5.setPrompt("Title");

spinner6.setPrompt("Title");

// выделяем элемент

spinner5.setSelection(2);

spinner6.setSelection(2);

// устанавливаем обработчик нажатия

spinner5.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view,

int position, long id) {

// показываем позицию нажатого элемента

Toast.makeText(getBaseContext(), "Position = " + position, Toast.LENGTH\_SHORT).show();

}

@Override

public void onNothingSelected(AdapterView<?> arg0) {

}

});

spinner6.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view,

int position, long id) {

// показываем позицию нажатого элемента

Toast.makeText(getBaseContext(), "Position = " + position, Toast.LENGTH\_SHORT).show();

}

@Override

public void onNothingSelected(AdapterView<?> arg0) {

}

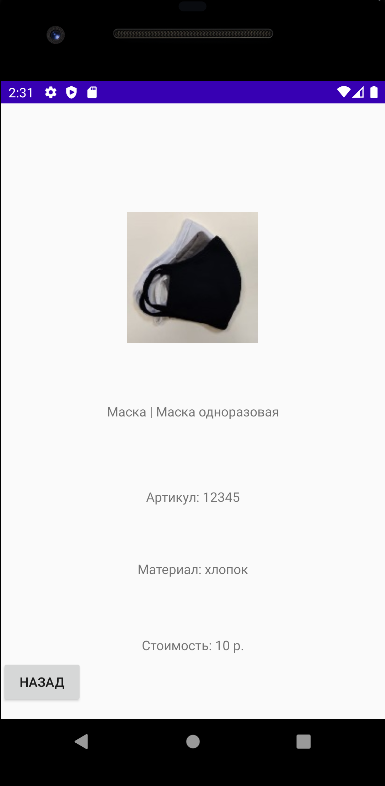
});

}

## **2.2 Разработка интегрированной информационной системы**

Интерфейс

Рисунок 3 – Главный экран приложения на устройстве

Рисунок 4 – Пример страницы товара на устройстве

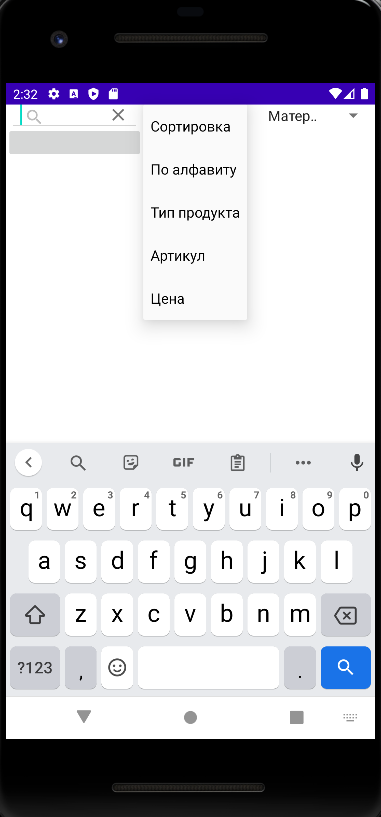
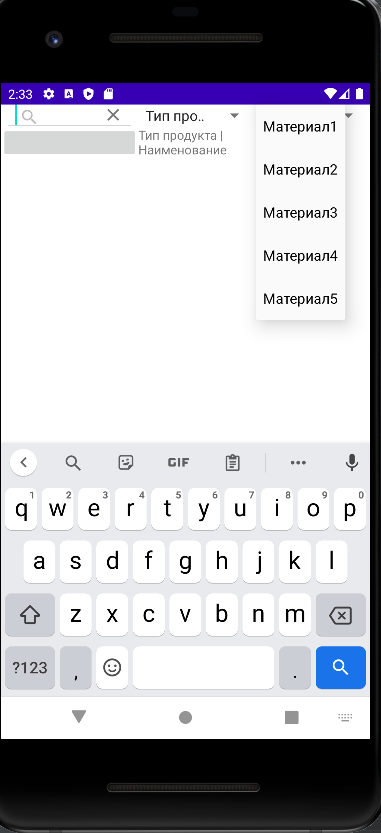


Рисунок 5 – Работа сортировки

Рисунок 6 – Работа фильтрации

Код Активности «Главный экран»:

package com.example.chirc;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.ImageButton;

import android.widget.Spinner;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity implements View.OnClickListener {

//создание строковых массивов, которые будут использоваться спиннерами

String[] data1 = {"Сортировка", "По алфавиту", "Тип продукта", "Артикул", "Цена"};

String[] data2 = {"Материал1", "Материал2", "Материал3", "Материал4", "Материал5"};

/\*\*

\* Called when the activity is first created.

\*/

ImageButton imageButton4;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

imageButton4 = findViewById(R.id.imageButton4);

imageButton4.setOnClickListener(this);

// адаптер

//присвоение спиннерам соответствующих строковых массивов

ArrayAdapter<String> adapter = new ArrayAdapter<String>(this, android.R.layout.simple\_spinner\_item, data2);

adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

ArrayAdapter<String> adapter2 = new ArrayAdapter<String>(this, android.R.layout.simple\_spinner\_item, data1);

adapter2.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

Spinner spinner5 = (Spinner) findViewById(R.id.spinner5);

Spinner spinner6 = (Spinner) findViewById(R.id.spinner6);

spinner5.setAdapter(adapter);

spinner6.setAdapter(adapter2);

// заголовок

spinner5.setPrompt("Title");

spinner6.setPrompt("Title");

// выделяем элемент

spinner5.setSelection(2);

spinner6.setSelection(2);

// устанавливаем обработчик нажатия

spinner5.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view,

int position, long id) {

// показываем позицию нажатого элемента

Toast.makeText(getBaseContext(), "Position = " + position, Toast.LENGTH\_SHORT).show();

}

@Override

public void onNothingSelected(AdapterView<?> arg0) {

}

});

spinner6.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {

@Override

public void onItemSelected(AdapterView<?> parent, View view,

int position, long id) {

// показываем позицию нажатого элемента

Toast.makeText(getBaseContext(), "Position = " + position, Toast.LENGTH\_SHORT).show();

}

@Override

public void onNothingSelected(AdapterView<?> arg0) {

}

});

}

//переход по экранам по кнопке

@Override

public void onClick(View view) {

switch (view.getId()) {

case R.id.imageButton4:

Intent intent2 = new Intent(this, page2.class);

startActivity(intent2);

break;

}

}

}

# **ЗАКЛЮЧЕНИЕ**

В ходе прохождения учебной практики, мной были освоены части основного вида деятельности «Осуществление интеграции программных модулей» и соответствующих профессиональных компетенций (ПК):

ПК 2.1. Разрабатывать требования к программным модулям на основе анализа проектной и технической документации на предмет взаимодействий компонентов;

ПК 2.2. Выполнять интеграцию модулей в программное обеспечение;

ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств;

ПК 2.4. Осуществлять разработку текстовых наборов и тестовых сценариев для программного обеспечения;

ПК 2.5. Производить инспектирование компонент программного обеспечения на предмет соответствия стандартам кодирования.

Были выполнены следующие цели по овладения указанным видом профессиональной деятельности и соответствующим профессиональными компетенциями:

Иметь практический опыт:

* В интеграции модулей в программное обеспечение;
* В отладке программных модулей.

Уметь:

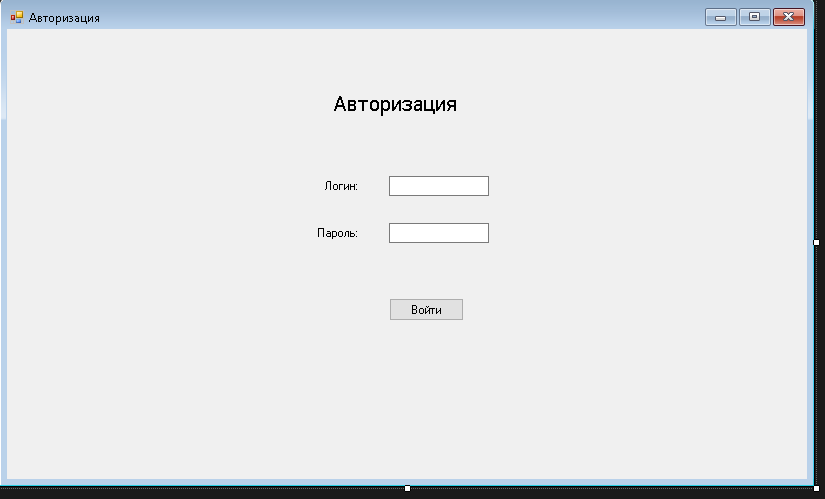
* Использовать выбранную систему контроля версий;
* Использовать методы для получения

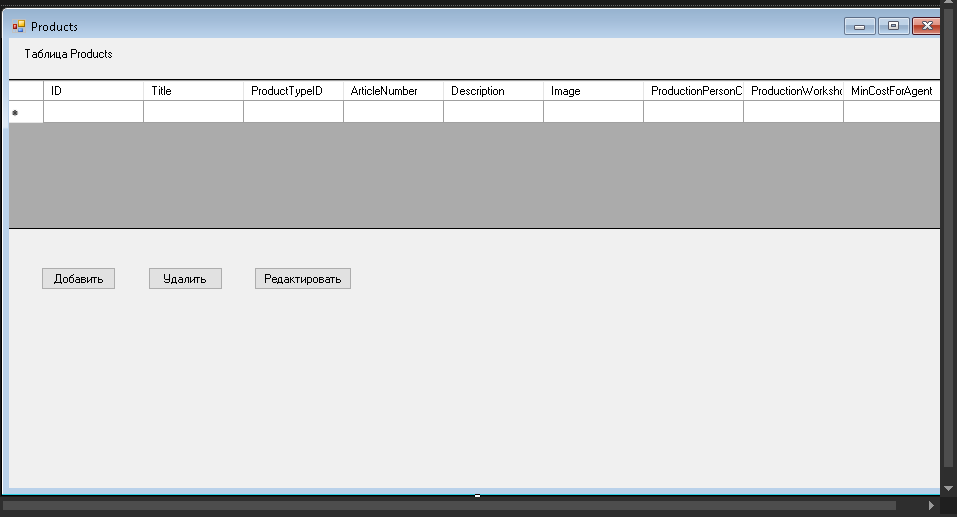
Знать:

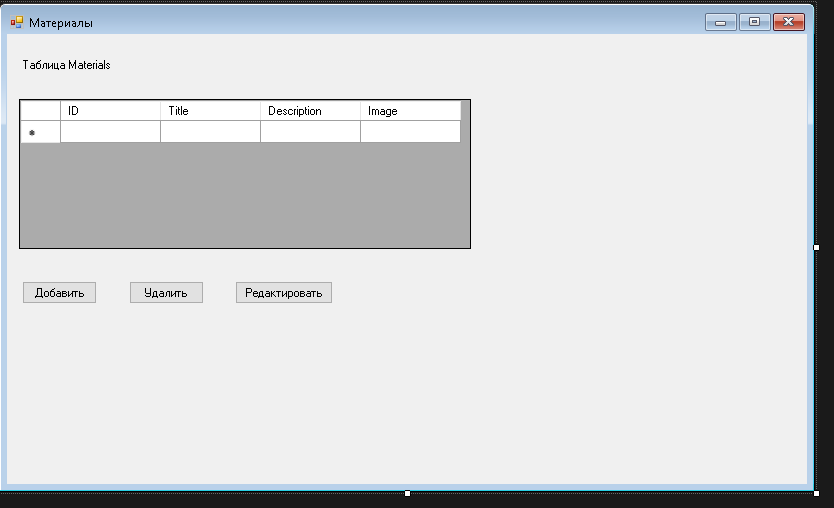
* Модели процесса разработки программного обеспечения;
* Основные принципы процесса разработки программного обеспечения;
* Основные подходы к интегрированию программных модулей;
* Основы верификации и аттестации программного обеспечения.

# **ПРИЛОЖЕНИЯ**

**Приложение А**

**Форма «Авторизация»** 

**Форма «Продукты»** 

**Форма «Материалы»** 

**Форма «Просмотр продуктов»**