|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_\_Робототехники и комплексной автоматизации\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_\_\_\_\_\_\_Системы автоматизированного проектирования\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

***К КУРСОВОМУ ПРОЕКТУ***

***НА ТЕМУ:***

***\_\_\_\_\_\_Разработка информационной системы\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_«Склад»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

***\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

Студент \_\_\_\_РК6-54Б\_\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_Магомедов З.А.\_\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель курсовой работы **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_Пивоварова Н.В.\_\_\_**

(Подпись, дата) (И.О.Фамилия)

*2021 г.*

# **Аннотация**

Курсовой проект включает в себя реализацию параметризованных запросов через пользовательский интерфейс, авторизацию пользователей, проект и реализацию основного бизнес-процесса, а также работу с хранимыми процедурами.

Для каждого пользователя информационной системы составлены варианты ее использования. При этом для каждого из них составлено описание, которое включает в себя сценарий, BPMN-диаграмму, требования к шаблонам, программную архитектуру реализации варианта использования. Всего предусмотрено 3 варианта использования:

1. Работа с запросами
2. Авторизация
3. Основной бизнес-процесс

Оглавление

[**Аннотация** 2](#_Toc90396953)

[**Задание. Описание предметной области** 4](#_Toc90396954)

[**Модель базы данных** 5](#_Toc90396955)

[**Определение конечных пользователей** 6](#_Toc90396956)

[**Разработка UML-диаграммы вариантов использования** 6](#_Toc90396957)

[**Вариант использования Главное меню** 6](#_Toc90396958)

[**Вариант использования Работа с запросами** 8](#_Toc90396959)

[**Вариант использования Авторизация** 11](#_Toc90396960)

[**Вариант использования Оформление поставки** 14](#_Toc90396961)

# **Задание. Описание предметной области**

В рамках курсового проекта необходимо определить конечных пользователей информационной системы. Для каждого из них составить варианты использования системы. Выделить основной вариант использования информационной системы, т.е. основной бизнес-процесс. Разработать систему авторизации пользователей информационной системы. Помимо этого, необходимо необходима UML-диаграмма вариантов использования. Для каждого варианта использования необходимо сделать:

1. Сценарий
2. BPMN-диаграмму контроллера
3. Требования к шаблонам
4. Программная архитектура реализации вариантов использования

***Предметной областью*** информационной системы является **склад**. При этом накапливаются следующие данные:

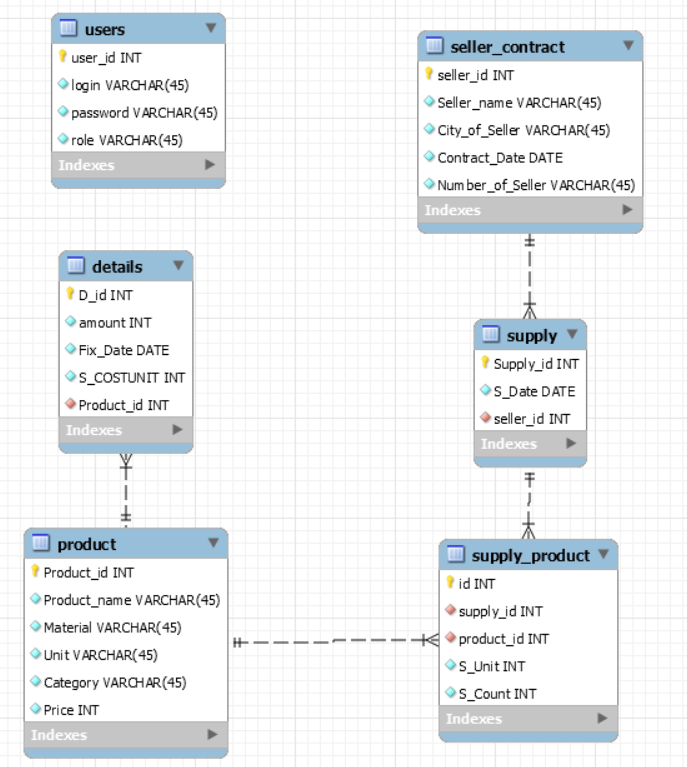
* Уникальный шифр товара
* Название товара
* Материал
* Единица измерения
* Категория товара
* Шифр поставщика товара
* Ф. И. О. поставщика
* Город поставщика
* Номер долгосрочного договора
* Дата заключения долгосрочного договора
* Телефон поставщика
* Количество конкретного товара на складе
* Дата, на которую зафиксировано количество конкретного товара
* Номер поставки
* Дата поставки
* Цена единицы товара в поставке
* Количество товара в поставке

***Описание бизнес-процесса***

С каждым поставщиком предварительно заключается долгосрочный договор на поставки товаров. Далее каждый поставщик может многократно поставлять разнообразные товары на склад.

Каждая поставка имеет уникальный номер и дату поставки. При поставке Поставщик указывает количество поставленного товара и назначает цену единицы товара. Количество одноименного товара на складе, имеющего одинаковую цену, суммируется и запоминается вместе с датой последнего изменения количества.

# **Модель базы данных**



# **Определение конечных пользователей**

В разрабатываемой информационной системе у нас будет 3 вида конечных пользователей:

* Начальник склада
* Менеджер
* Администратор

# **Разработка UML-диаграммы вариантов использования**

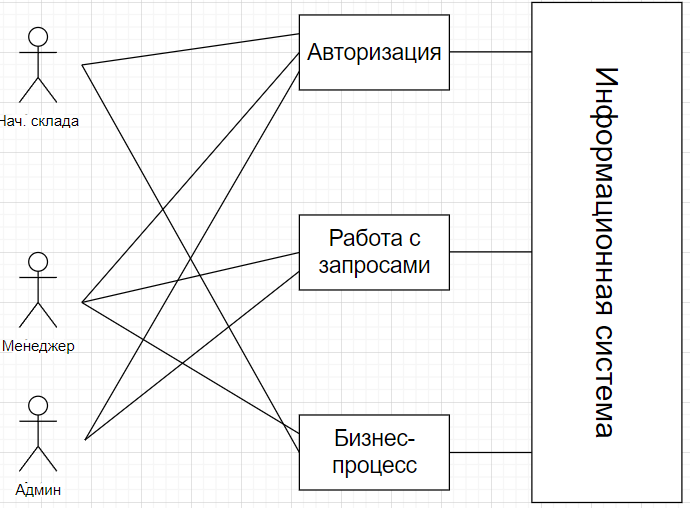
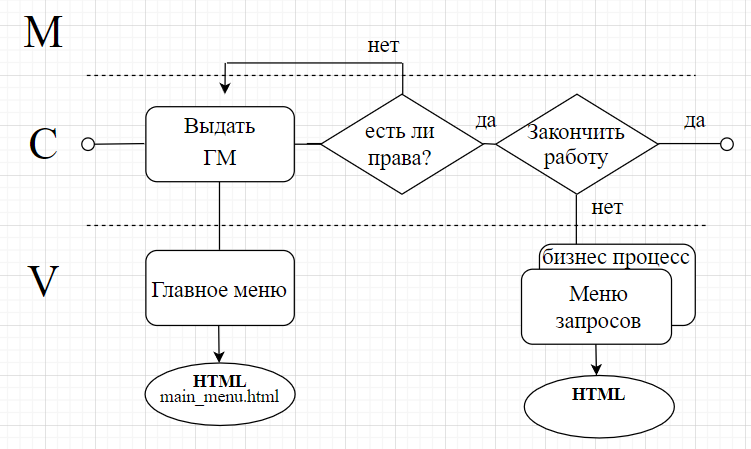


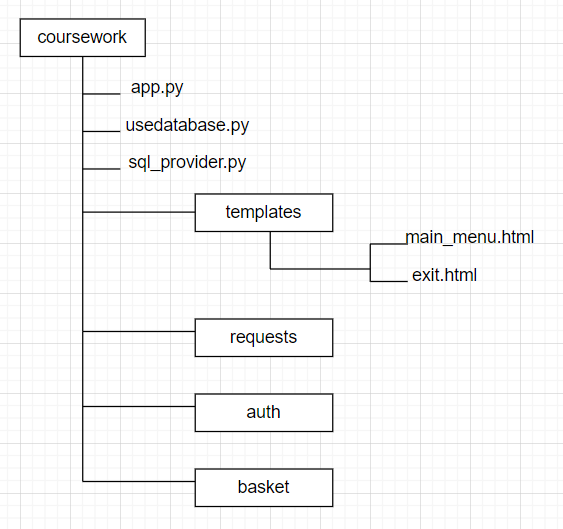
Рис.1. UML-диаграмма вариантов использования

# **Вариант использования Главное меню**

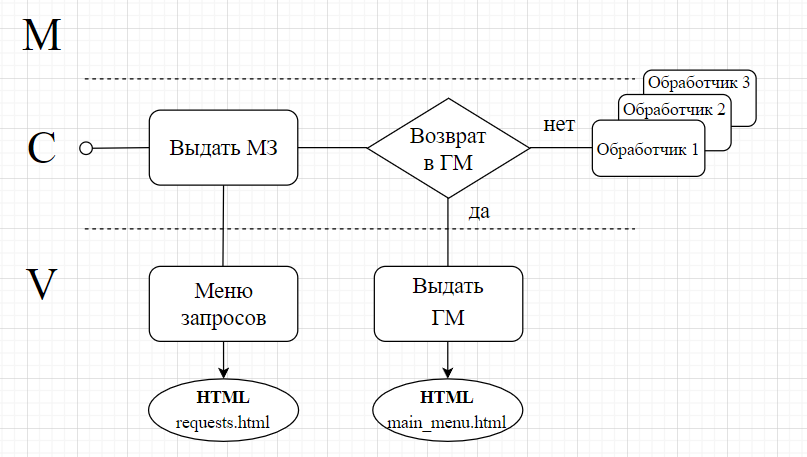
1. Сценарий
   1. Пользователь запускает сценарий
   2. Система присылает главное меню
   3. Пользователь выбирает пункт использования
   4. Система передает управление соответствующему контроллеру
2. Исключения
   1. У пользователя нет прав для выбранного им варианта использования
3. BPMN-диаграмма

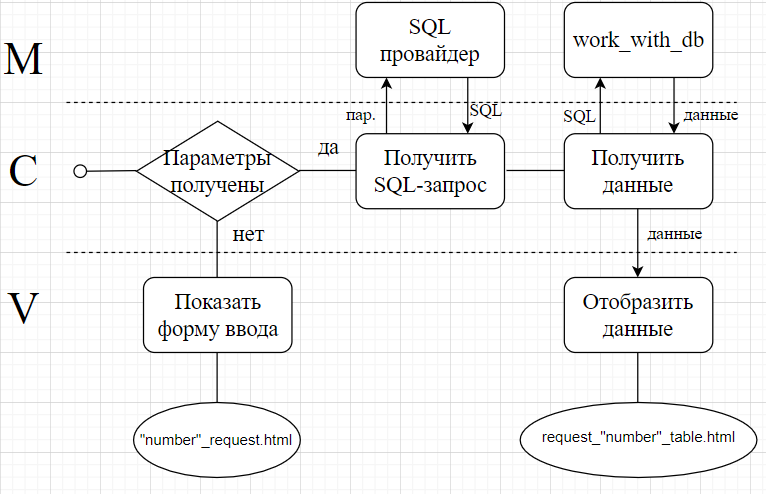


1. Требования к шаблонам
   1. Контроллер работы с запросами('/requests') с вызовом шаблона HTML(requests.html)
   2. Контроллер авторизации('/auth/login') с вызовом шаблона HTML(login.html)
   3. Контроллер оформления поставки('/basket') с вызовом шаблона HTML(suppliers.html) для выбора поставщика
   4. Выход из системы('/exit') с очисткой сессии с вызовом шаблона HTML(exit.html)
2. Программная архитектура



# **Вариант использования Работа с запросами**

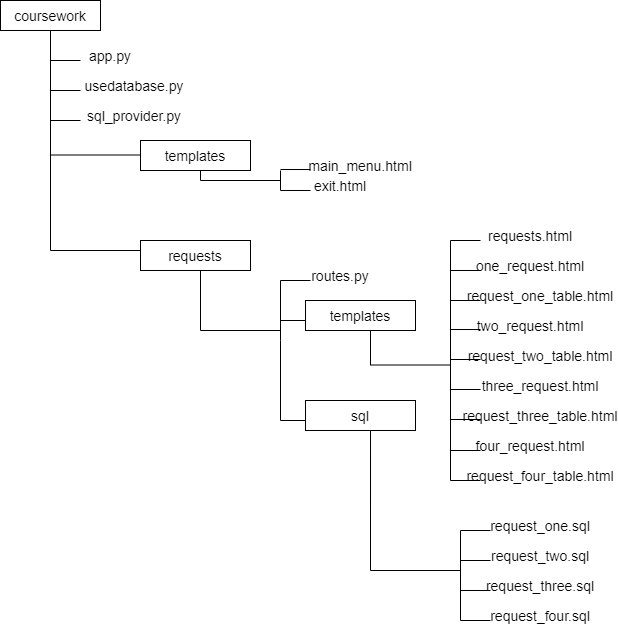
1. Сценарий
   1. Меню запросов
      1. Пользователь запускает сценарий
      2. Система присылает меню запросов
      3. Пользователь выбирает запрос
      4. Система передает управление соответствующему контроллеру
   2. Запрос
      1. Пользователь запускает сценарий
      2. Система присылает форму для ввода параметров
      3. Пользователь вводит параметры
      4. Система выполняет запрос и присылает результаты с возможностью выхода в меню запросов
2. Исключения
   * 1. Система не находит данные по запросу
     2. Пользователь не вводит данные
     3. Пользователь вводит некорректные данные
3. BPMN-диаграмма



1. Требования к шаблонам
   1. Меню запросов
      1. Контроллер первого запроса('/requests/first') c вызовом шаблона HTML(one\_request.html).

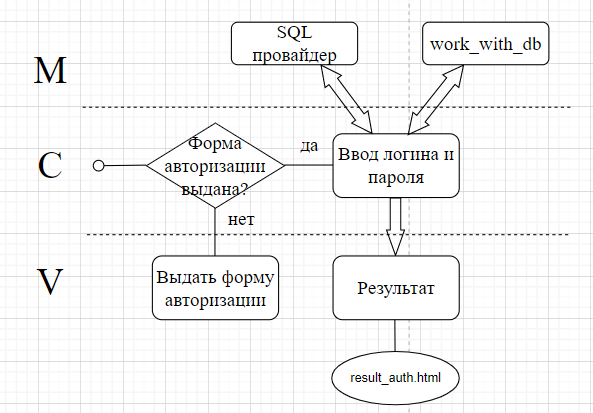
* **Запрос**: показать отчет о запасах товара XXX на складе
* Метод передачи параметров – POST
* В форме необходимо разместить поле типа «text» для ввода названия товара
* В шаблон передается словарь, в котором по ключу 'schema' можно получить имена заголовков, а по ключу 'data' список словарей со значениями
* Поле типа «submit» для выполнения SQL-запроса
* Поле типа «submit» для возврата в меню запросов
  + 1. Контроллер второго запроса('/requests/second') c вызовом шаблона HTML(two\_request.html)
* **Запрос**: показать шифры и названия товаров, которые поставщик XXX поставил в XXX году.
* Метод передачи параметров – POST
* В форме необходимо разместить поле типа «text» для ввода названия товара, а также выпадающий список поставщиков, с которыми заключен контракт.
* В шаблон передается словарь, в котором по ключу 'schema' можно получить имена заголовков, а по ключу 'data' список словарей со значениями
* Поле типа «submit» для выполнения SQL-запроса
* Поле типа «submit» для возврата в меню запросов
  + 1. Контроллер третьего запроса('/requests/third') c вызовом шаблона HTML(three\_request.html)
* **Запрос**: показать поставщика самого дешевого товара с названием XXX
* Метод передачи параметров – POST
* В форме необходимо разместить поле типа «text» для ввода названия товара
* В шаблон передается словарь, в котором по ключу 'schema' можно получить имена заголовков, а по ключу 'data' список словарей со значениями
* Поле типа «submit» для выполнения SQL-запроса
* Поле типа «submit» для возврата в меню запросов
  + 1. Контроллер четвертого запроса('/requests/fourth') c вызовом шаблона HTML(four\_request.html)
* **Запрос**: показать название товара, который ни разу не поставлялся в XXX году
* Метод передачи параметров – POST
* В форме необходимо разместить поле типа «text» для ввода года
* В шаблон передается словарь, в котором по ключу 'schema' можно получить имена заголовков, а по ключу 'data' список словарей со значениями
* Поле типа «submit» для выполнения SQL-запроса
* Поле типа «submit» для возврата в меню запросов
  + 1. Главное меню('/') c вызовом шаблона HTML(main\_menu.html)
  1. Запрос(форма ввода параметров)
     1. Содержит поля для ввода параметров
     2. Содержит кнопку «Получить информацию» с переходом на нее ('/requests/''number\_ of\_request/parametrs'')
     3. Идет передача в шаблон HTML(request\_''number''\_table.html) словаря с именами полей и значениями в этих полях.
     4. Ссылка в меню запросов('/requests') с вызовом шаблона HTML(requests.html)
     5. Ссылка на выход из системы('/exit') с вызовом шаблона HTML(exit.html)

1. Тестовые данные
   1. Запрос 1
      1. Набор данных: Book
      2. Результат: 3 строчки
   2. Запрос 2
      1. Набор данных: Zolin, 2012
      2. Результат: 2 строчки
   3. Запрос 3
      1. Набор данных: Perfume
      2. Результат: 1 строчка
   4. Запрос 4
      1. Набор данных: 2012
      2. Результат: 3 строчки
2. Программная архитектура



# **Вариант использования Авторизация**

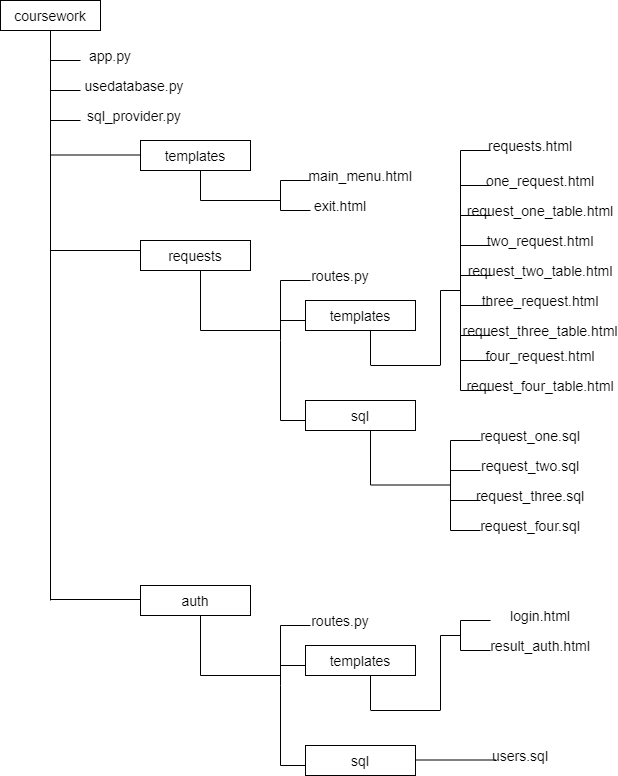
1. Сценарий
   * 1. Пользователь выбирает пункт «Авторизация»
     2. Система выдает форму для ввода логина и пароля
     3. Пользователь вводит логин и пароль и отправляет форму
     4. Если логин и пароль верны (присутствуют в БД), то система сообщает пользователю об успешной авторизации, иначе выдает сообщение об ошибке
2. Исключения
   * 1. Пользователь не вводит данные
     2. Пользователь вводит данные несуществующего пользователя
3. BPMN-диаграмма



1. Требования к шаблонам
   * 1. Форма ввода содержит поля для ввода логина и пароля и кнопку подтверждения(login.html) с возможностью вернуться назад(/auth/login)

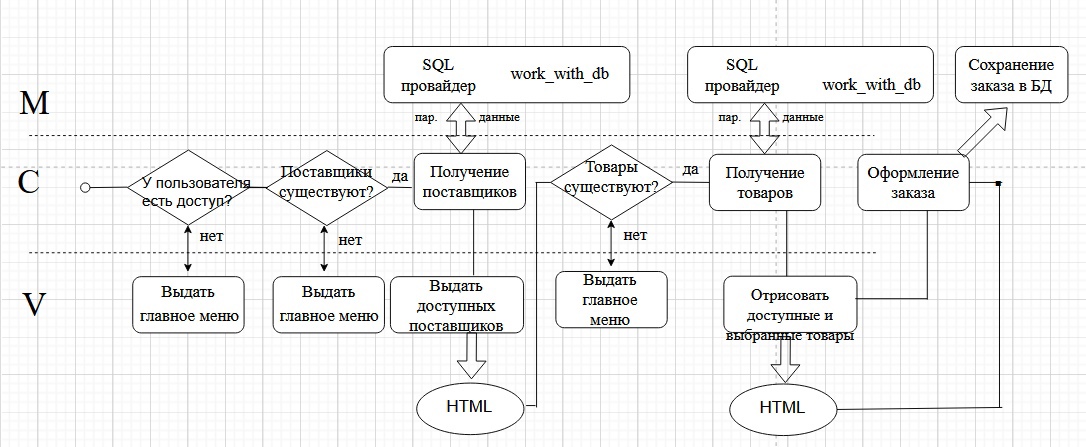
* Метод передачи параметров – POST
* Поле для ввода логина типа «text»
* Поле для ввода пароля типа «password»
* В шаблон получения результата передается flag для проверки успешности входа
* Поле типа «submit» для подтверждения попытки входа
* Поли типа «submit» для возврата в главное меню
  + 1. Выдается HTML страничка (result\_auth.html) результата попытки авторизации с возможностью вернуться в главное меню (/auth/login)

1. Тестовые данные
   1. **Тест №1**
      1. Логин: manager, Пароль: manager
      2. Результат: Вы авторизовались как manager
   2. **Тест №2**
      1. Логин: abc, Пароль: abc
      2. Результат: Ошибка авторизации! Неверно введен логин и/или пароль
2. Программная архитектура



# **Вариант использования Оформление поставки**

1. Сценарий
   * 1. Пользователь запускает сценарий
     2. Система выдает форму оформления поставки
     3. Пользователь выбирает поставщика
     4. Система выдает форму для выбора товаров
     5. Пользователь выбирает товары и нажимает «Оформить»
     6. Система оформляет поставку
2. Исключения
   * 1. Пользователь нажал кнопку «оформить», не выбрав товары.
3. BPMN-диаграмма



1. Требования к шаблонам
   * 1. Контроллер выбора поставщика для поставки('/basket')

* В форме необходимо разместить выпадающий список поставщиков, с которыми заключен контракт
* Метод передачи параметров – POST
  + 1. Контроллер корзины('/basket/list')
* Колонка с доступными для поставки товарами
* Колонка с добавленными для поставки товарами
* Поле очистки корзины типа «submit»
* Поле оформления поставки типа «submit» с переходом в ('/basket/buy')
* Поле возврата в главное меню типа «submit» с переходом в ('/')

1. Программная архитектура

