**Лабораторная работа №23**

**Техническое задание**

**Введение**

**1.5.1.1 Наименование программы**

Наименование программного продукта – сайт-конструктор «Папа Карло»

**1.5.1.2 Краткая характеристика области применения**

Программа предназначена для помощи в планировании и конструировании шкафов/комодов/шкафов-купе.

**1.5.2 Основания для разработки**

Основанием для разработки является задание на практику на тему «Распил и конструирование шкафов/шкафов-купе/комодов». Тема согласована с преподавателем.

**1.5.3 Назначение разработки**

Основным назначением программы является изучение принципов создания

программного обеспечения на языке программирования C# и применение в них баз данных, создания программной документации.

**1.5.4 Требования к программе или программному изделию**

**1.5.4.1 Требования к применению**

Пользователь, который использует программу должен обладать базовыми

навыками воображения

**1.5.4.2 Требования к производительности**

Программа должна запускаться на слабо конфигурированных системах.

Минимальные системные требования:

* Intel Core i3 второго поколения;
* 4 ГБ ОЗУ;
* 100 МБ свободного места на жестком диске;
* Windows 8 и выше;

Максимальное время на запуск программы при минимальных требованиях должно составлять не более 15 секунд

**1.5.4.3 Требования к функциональным характеристикам**

Чтобы обеспечить возможность полного взаимодействия всех частей

разрабатываемого приложения, необходимо автоматизировать следующие задачи:

**Для сайта:**

**Гость:**

1. Регистрация
2. Просмотр страницы
3. Выбор региона
4. Переход в конструктор и просмотр предложенных моделей по заданным параметрам
5. Добавление в избранное
6. Просмотр готовых изделий
7. Обзор материалов

**Пользователь:** все те же функции, что может выполнять гость, но так же появляется:

1. Авторизация
2. Покупка
3. Доставка на дом
4. Добавление в лист ожидания
5. Оформление заказа распила
6. Выход из личного кабинета
7. Создание и редактирование профиля
8. Добавление заказа чертежа
9. Обзор поставщиков

**Администратор:**

1. авторизация
2. ответы на email и по телефону
3. доступ к базе данных
4. добавление/удаление категории
5. редактирование страниц сайта

**Поставщик:**

* + 1. отслеживание заказов и доставок
    2. обновление ежедневного ассортимента
    3. просмотр товара у других поставщиков

**Для конструктора:**

**Гость:**

1. переход в конструктор
2. задание параметров
3. просмотр готовых изделий
4. просмотр материалов/расцветок/фурнитуры

**Пользователь:**

1. конструирование собственного шкафа/комода по заданным параметрам
2. оформление заказа распила
3. автоматическое создание чертежа с заданными размерами
4. автосохранение созданного элемента в библиотеку
5. просмотр/выбор материалов/расцветок/фурнитуры
6. вывод итоговой цены

**Администратор:**

1. переход в конструктор прием заказа
2. анализ данного заказа
3. редактирование неточностей и согласование с заказчиком
4. добавление/удаление поставщиков
5. редактирование страниц конструктора

**Поставщик:**

1. ежедневное обновление ассортимента
2. добавление/удаление материалов/расцветок/фурнитуры9+

**1.5.4.4 Требования к реализации**

Для достижения максимальной производительности и оптимизации программа должна быть разработана на языке JavaScript в среде Visual Studio 2019.

**1.5.4.5 Требование к надежности**

Приложение должно производить самовосстановление после сбоя работы.

**1.5.4.6 Требования к интерфейсу**

Интерфейс должен быть с приятной цветовой гаммой и понятной для пользователя. Следовательно, каждое окно должно иметь ясную визуальную иерархию своих элементов. Фрагменты текста должны располагаться на экране так, чтобы пользователя было просто и понятно принимать информацию.

Пользователь не должен испытывать какого-либо дискомфорта в плане восприятия информация, отображённой на экране. Объекты (рисунки и символы) не должны быть слишком мелкие.

**1.5.5 Требования к программной документации**

Для удобства использования программного продукта пользователем, необходимо добавить файл справки, в котором будет находиться вся необходимая информация о работе с программой.

**1.5.6 Технико-экономические показатели**

Ориентировочная экономическая эффективность не рассчитываются.

Предполагаемое число использования программой в год – 365 сеансов на 1 рабочем месте.

**1.5.7 Стадии и этапы разработки**

**1.5.7.1 Стадии разработки**

Разработка должна быть проведена в три стадии: разработка технического

задания, рабочее проектирование, внедрение.

**1.5.7.2 Этапы разработки**

На стадии разработки технического задания должен быть выполнен этап

разработки, согласования и утверждения настоящего технического задания.

На стадии рабочего проектирования должны быть выполнены

перечисленные ниже этапы работ:

* разработка программы;
* разработка программной документации;
* испытания программы.

На стадии внедрения должен быть выполнен этап разработки – подготовка и

передача программы.

**1.5.7.3 Содержание работ по этапам**

На этапе разработки технического задания должны быть выполнены

перечисленные ниже работы:

1. Постановка задачи;

2. Определение и уточнение требований к техническим средствам;

3. Определение требований к программе;

4. Определение стадий, этапов и сроков разработки программы и

документации на неё;

5. Выбор ЯП;

6. Согласование и утверждение технического задания.

На этапе разработки программной документации должна быть выполнена

работа по программированию (кодированию) и отладке программы.

На этапе разработки программной документации должна быть выполнена

разработка программных документов в соответствии с требованиями

ГОСТ 19.101-77 с требованием п. Предварительный состав программной

документации настоящего технического задания.

На этапе испытаний программы должны быть выполнены перечисленные

ниже виды работ:

* Разработка, согласование и утверждение программы и методики испытаний;
* Проведение приемо-сдаточных испытаний;
* Корректировка программы и программной документации по результатам испытаний.

На этапе подготовки и передачи программы должна быть выполнена работа

по подготовке и передаче программы и программной документации в

эксплуатацию на объектах Заказчика.

**1.5.8 Порядок контроля и приемки**

**1.5.8.1 Виды испытаний**

Приемо-сдаточные испытания программы должны проводиться согласно

разработанной Исполнителем и согласованной Заказчиком программы и методик испытаний.

Ход проведения приемо-сдаточных испытаний Заказчик и Исполнитель

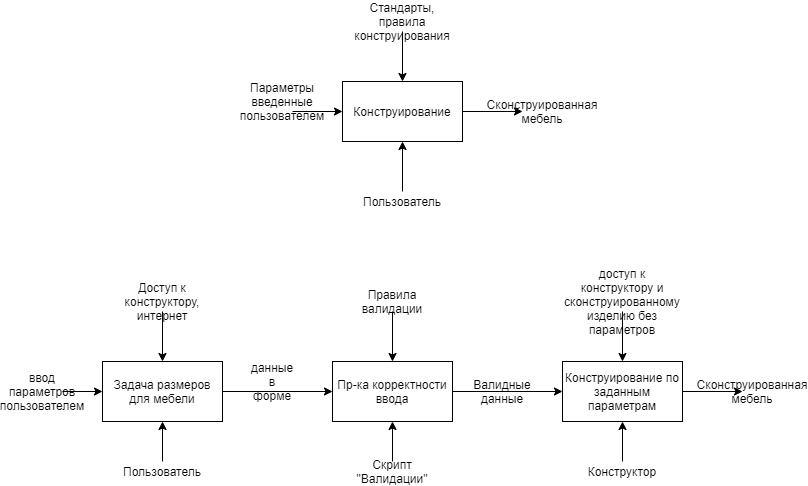
документируют в Протоколе проведения испытаний.

**1.5.8.2 Общие требования к приемке работы**

На основе Протокола проведения испытаний Исполнитель совместно с

Заказчиком подписывают Акт приемки-сдачи программы в эксплуатацию.

Функциональная модель



Структура сайта

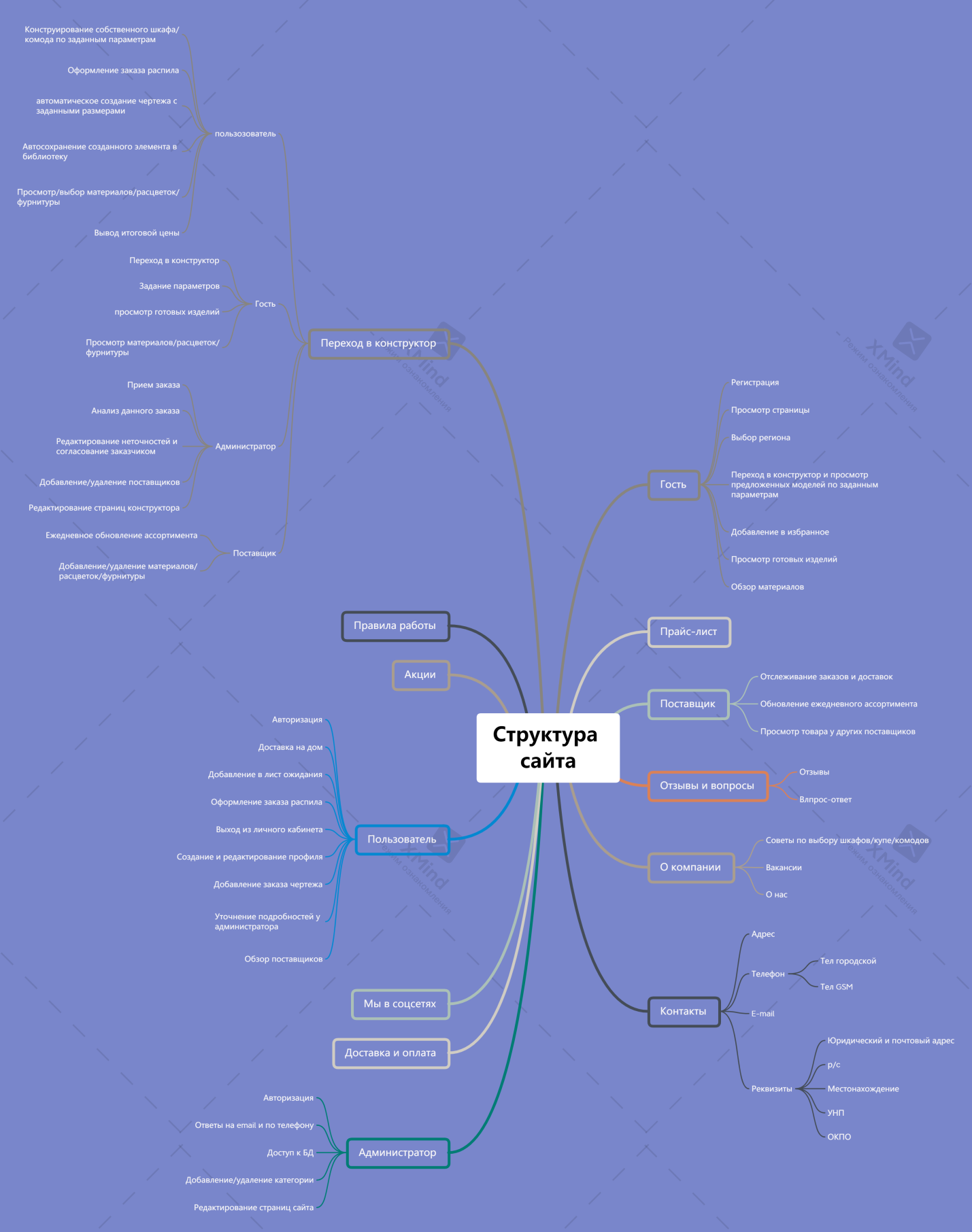


Диаграмма состояний

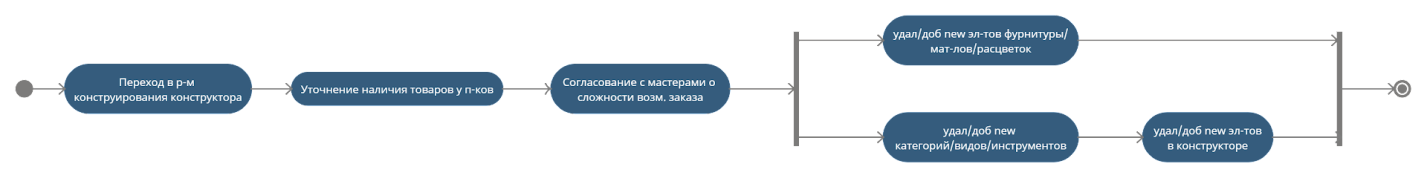


Диаграмма последовательности

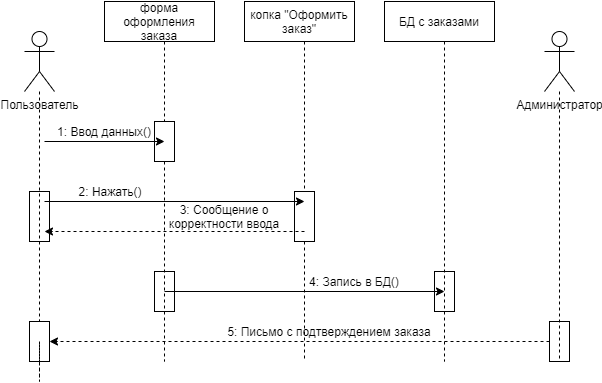
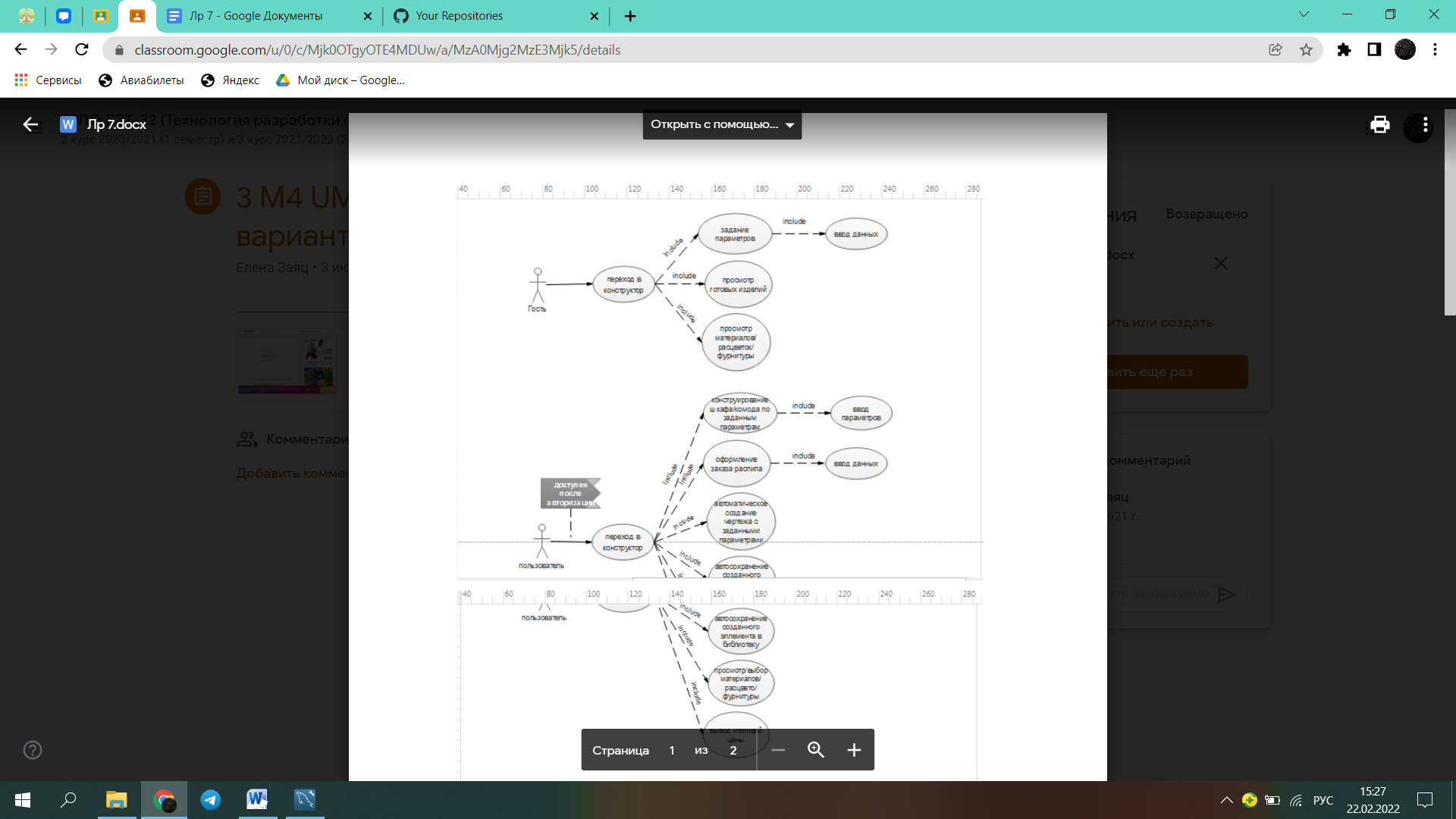
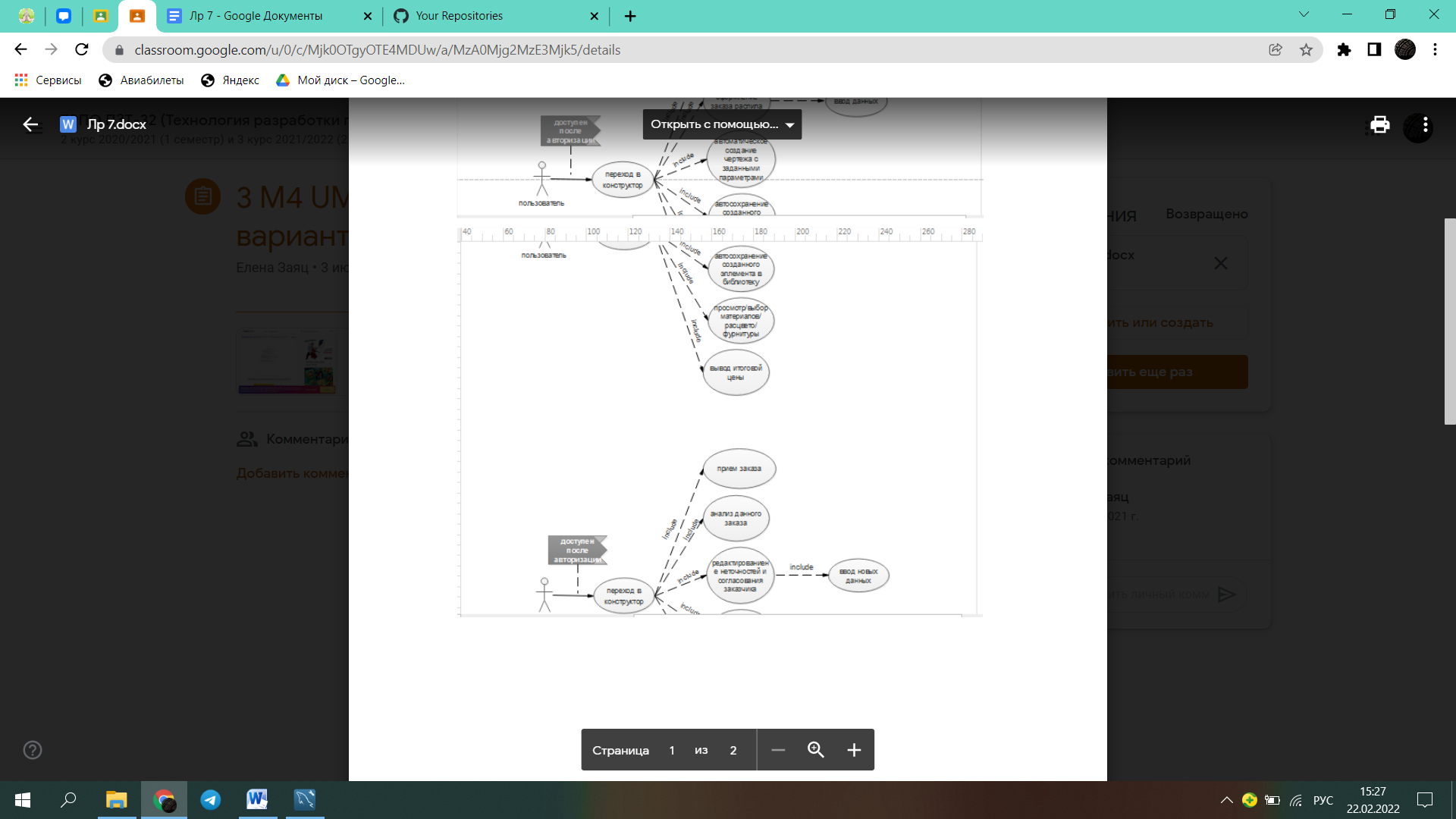


Диаграмма вариантов использования





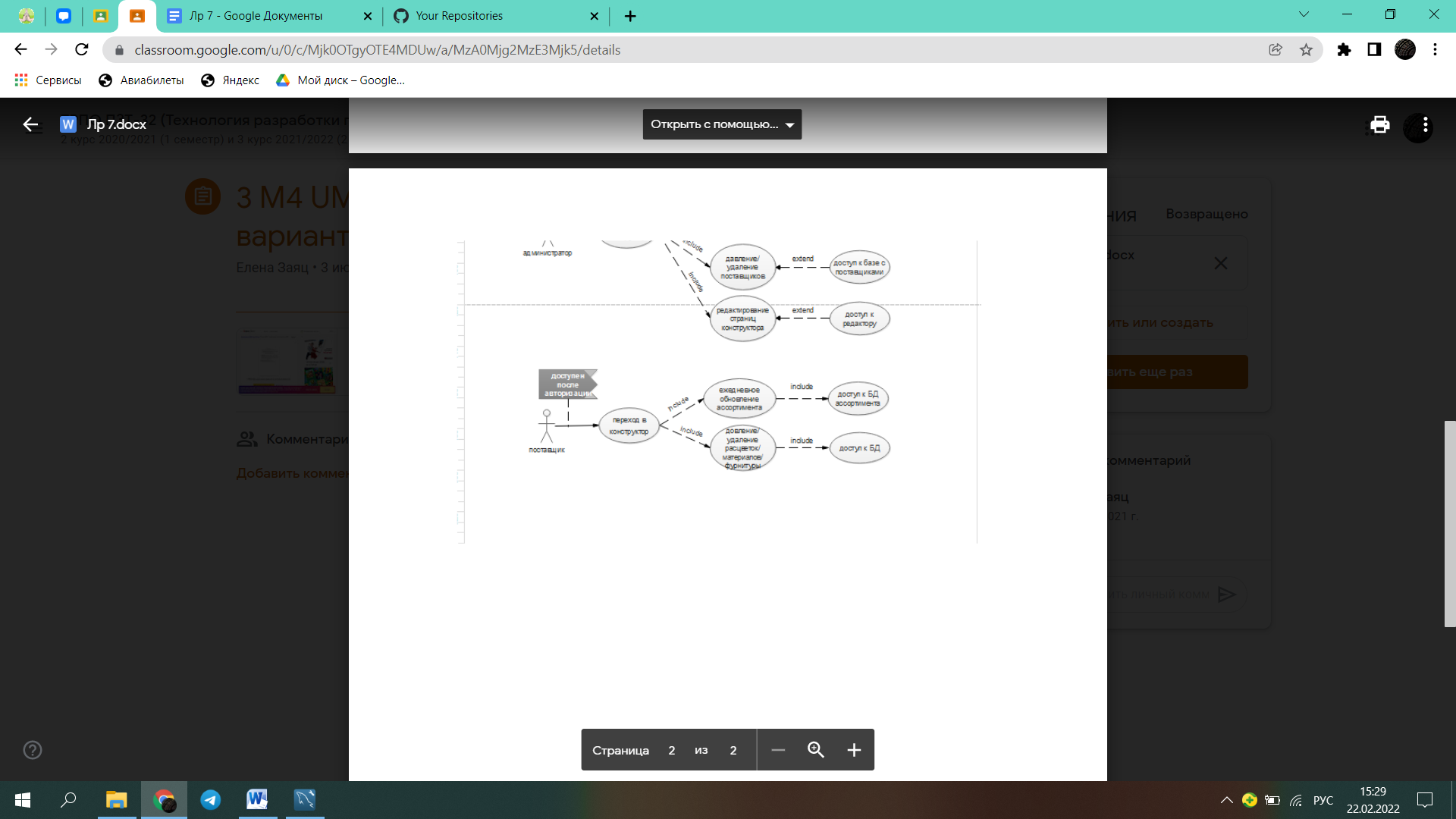


Диаграмма коопераций



Диаграмма деятельности

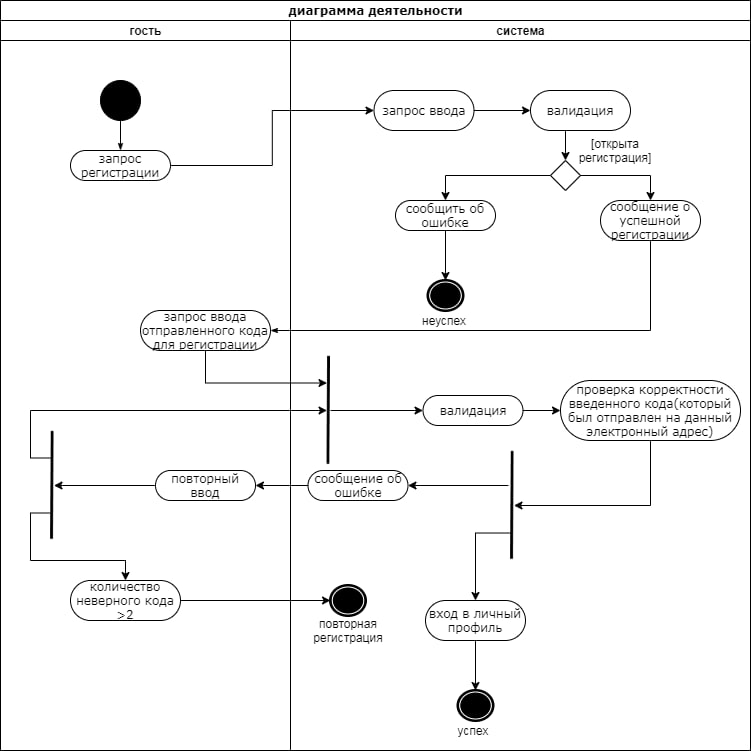
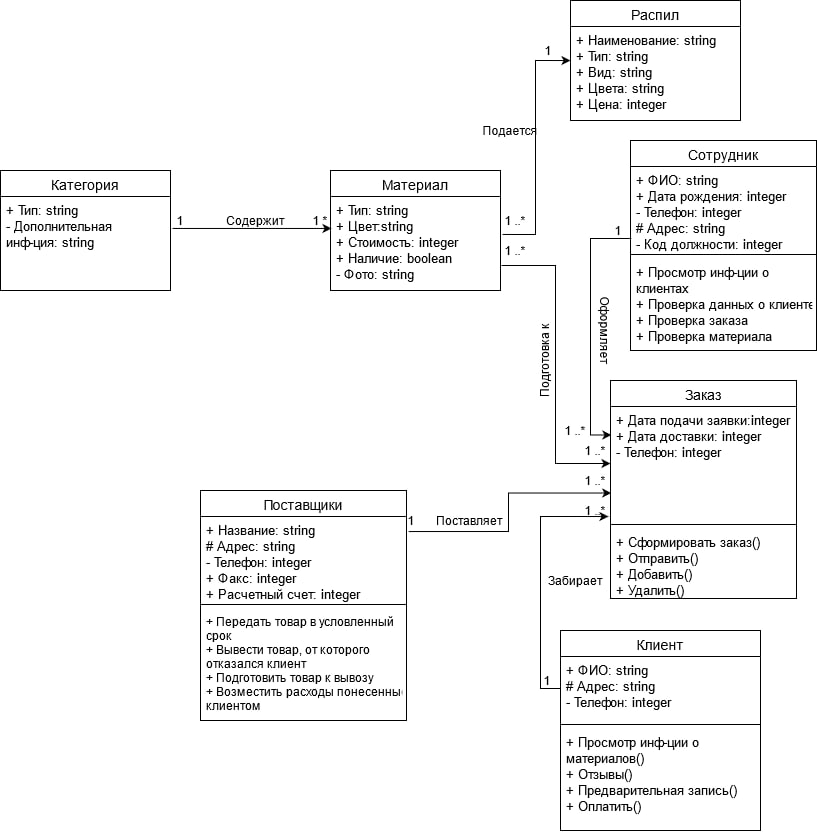
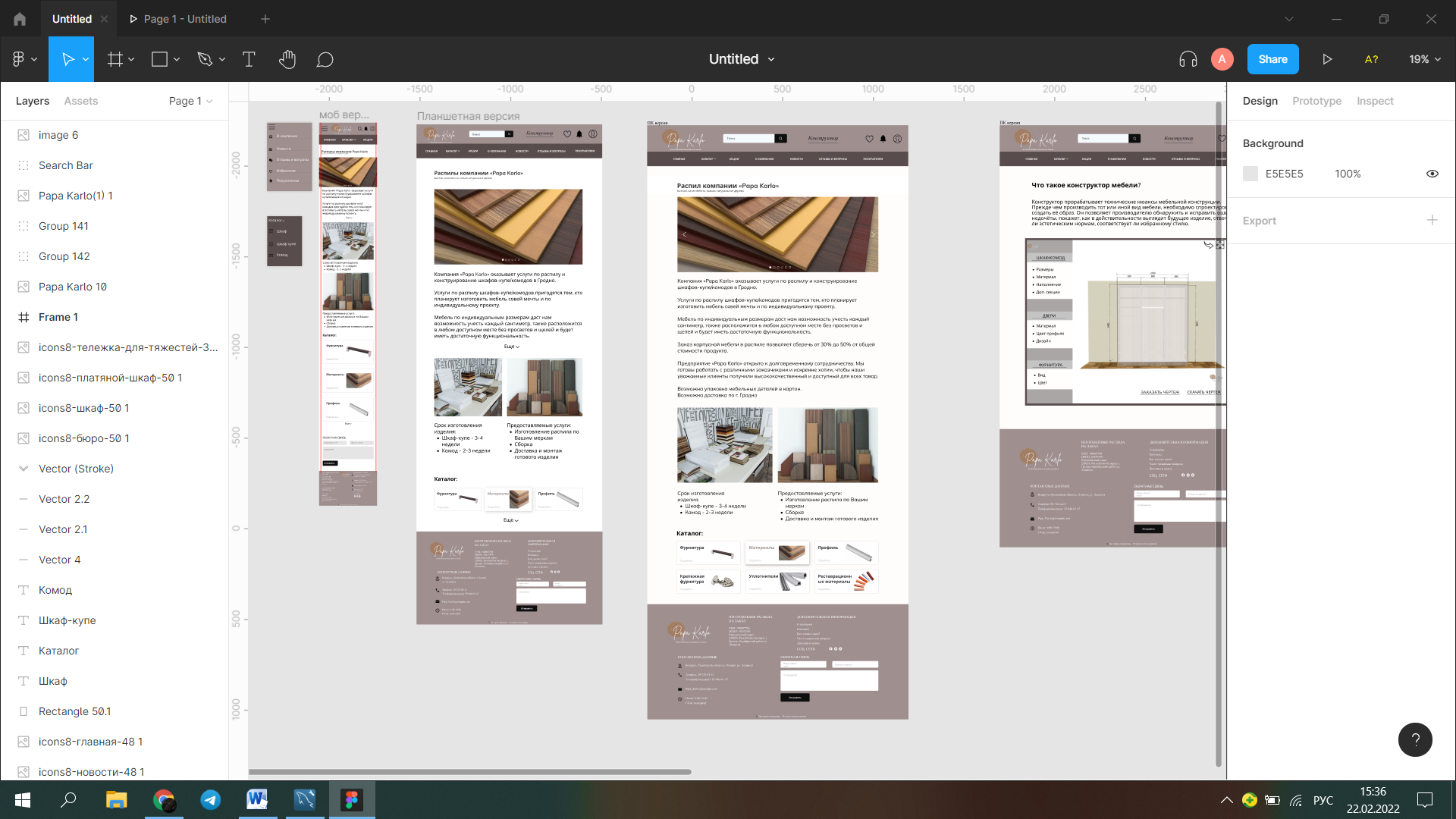
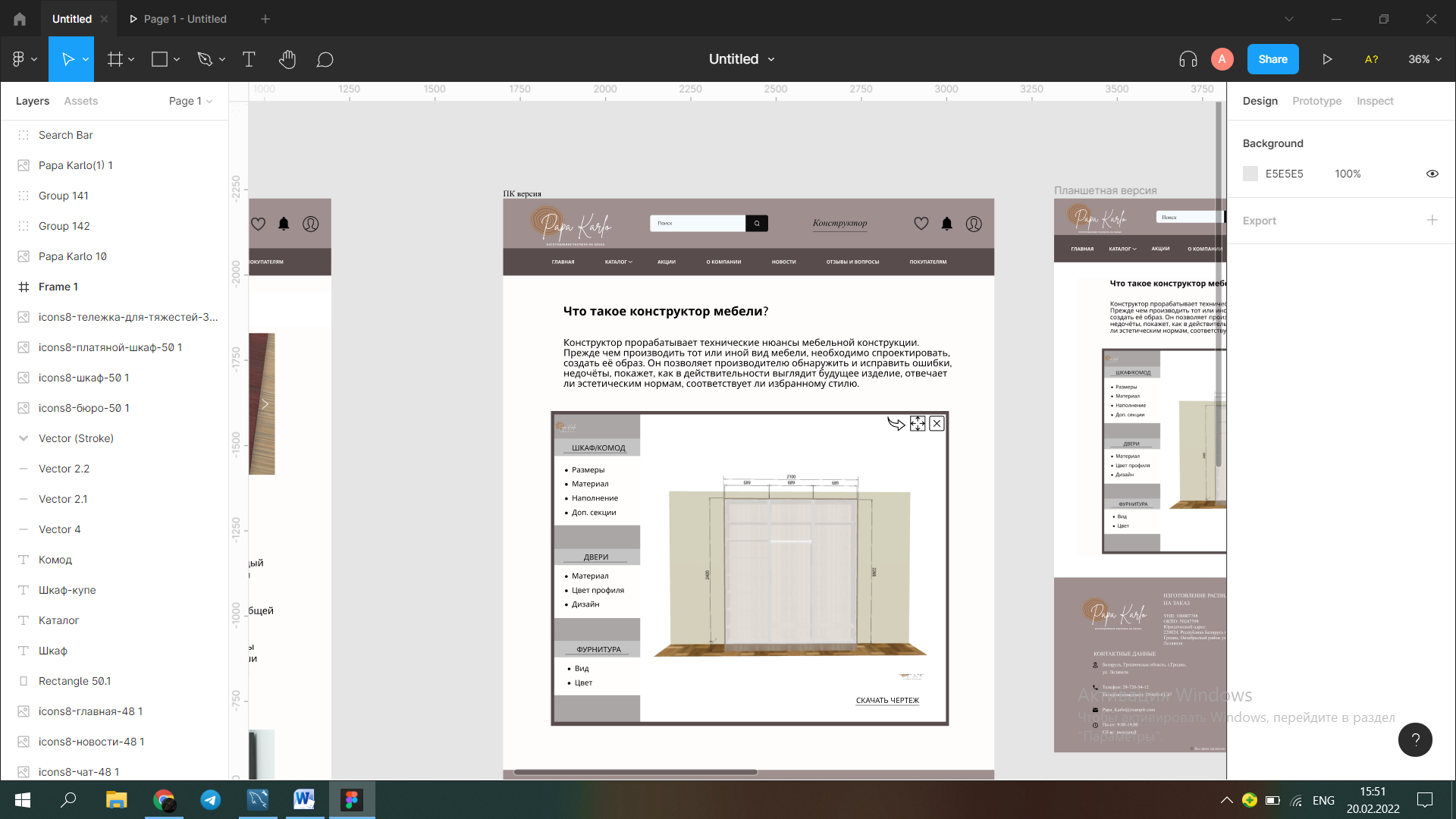


Диаграмма классов

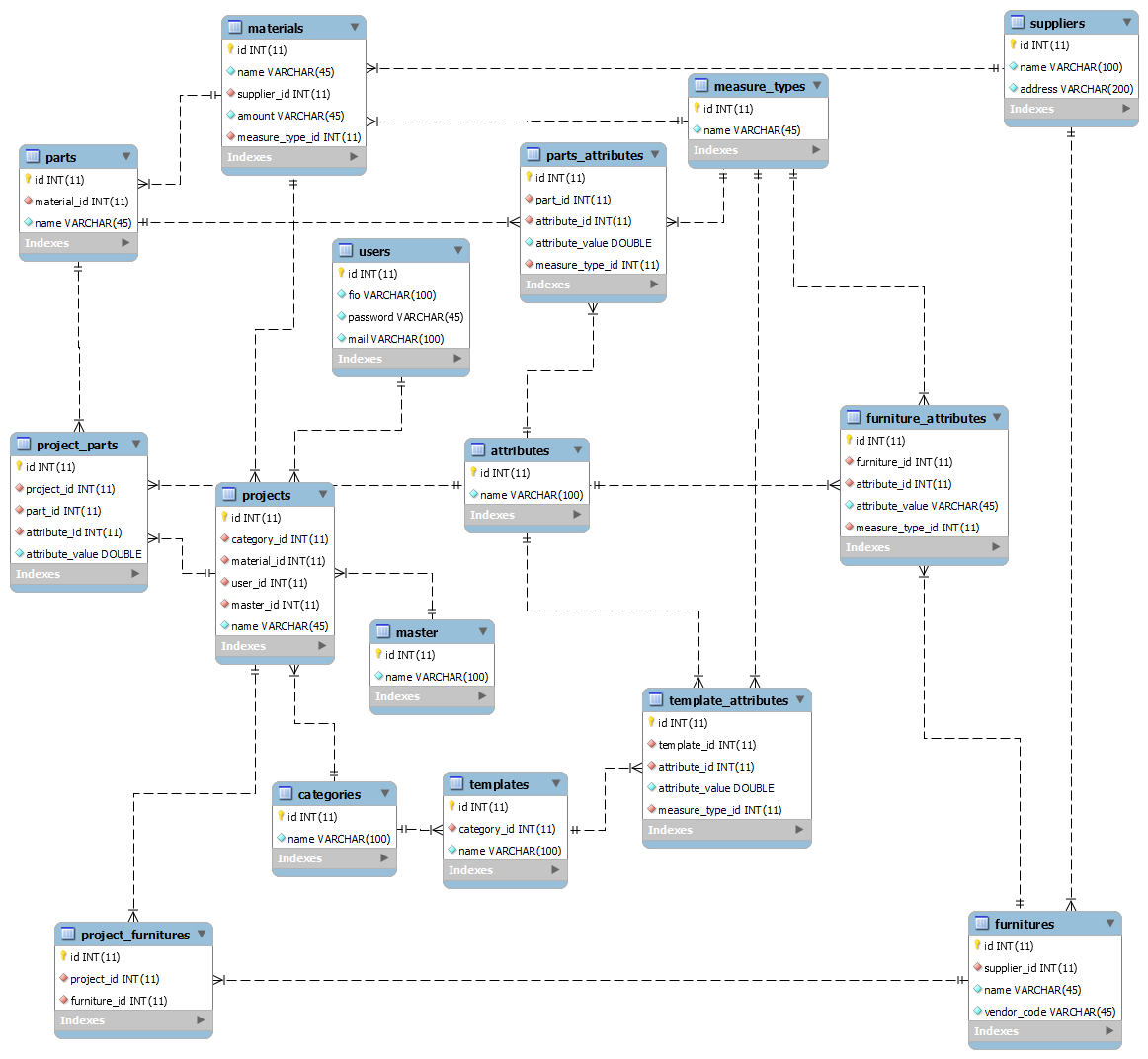


Пользовательский интерфейс





Прототип программного продукта



1. **Руководство программиста**

**2.1 Организация данных**

Определив атрибуты для каждой сущности и установив связи между ними, получили следующую схему данных (рисунок 1).

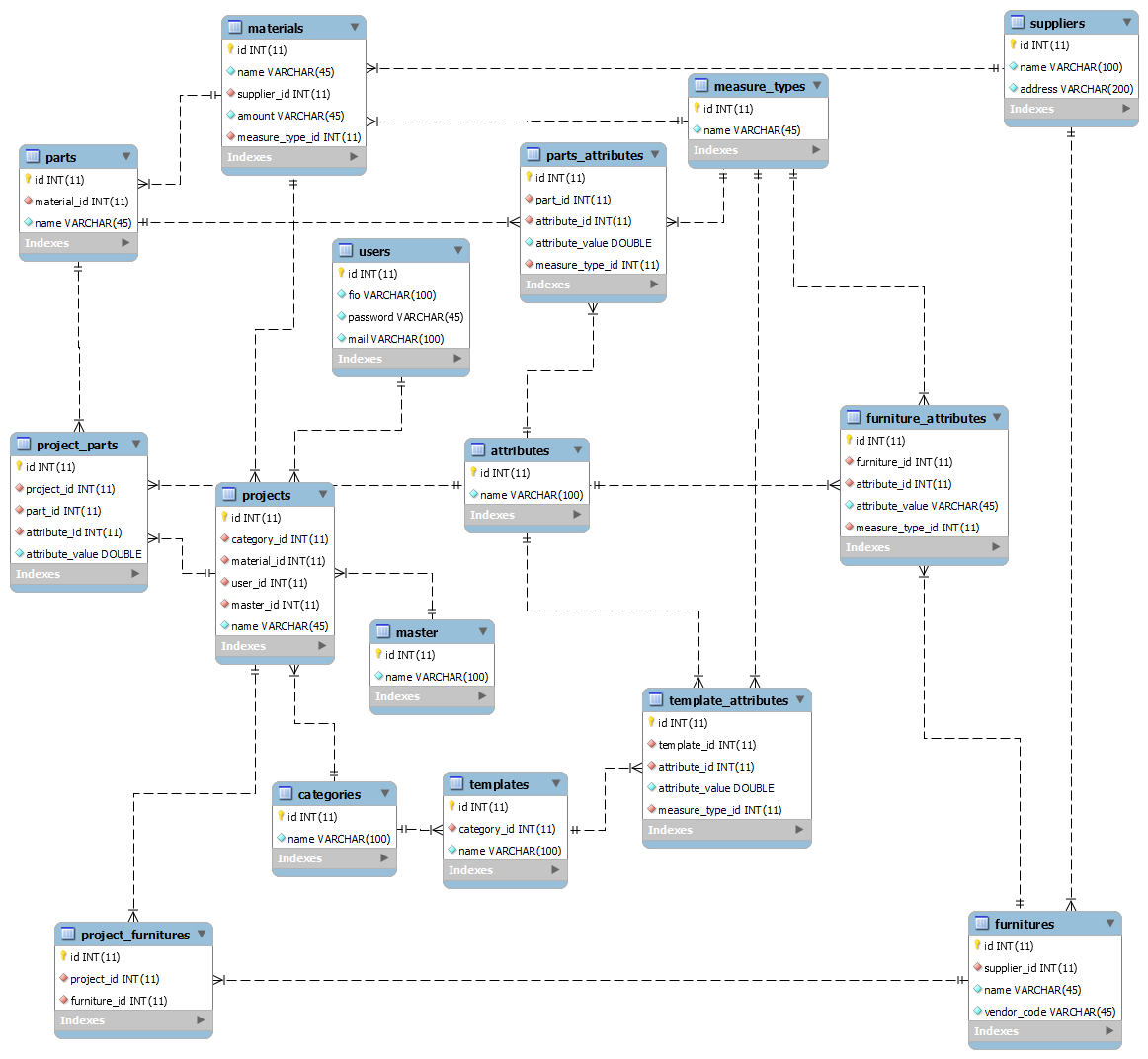


Рисунок 1 – схема данных

В данной БД представлено 16 сущностей. Описание каждой из них представлено в таблице 1.

Таблица 1 – Сущности

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Сущность | Имя поля | Тип данных | Описание |
| 1 | user | Id | INT | Содержит информацию о пользователе: его ФИО и пароль. |
| name | VARCHAR |
| password | INT |
| 2 | Master | Id | INT | Содержит информацию о мастере выполняющем заказ по распилу: его ФИО. |
| name | VARCHAR |
| 3 | suppliers | Id | INT | Храниться информация о поставщиках |
| name | VARCHAR |
| 4 | categories | Id | INT | Храниться информация о доступных категориях |
| name | VARCHAR |
| 5 | attributes | Id | INT | Храниться информация о параметрах(высота, ширина, глубина) |
| name | VARCHAR |
| 6 | measure\_types | Id | INT | Храниться информация о единицах измерения |
| name | VARCHAR |
| 7 | templates | Id | INT | Храниться информация о подкатегориях |
| category\_id | INT |
| name | VARCHAR |
| 8 | furnitures | Id | INT | Храниться информация о поставляемой фурнитуре |
| supplier\_id | INT |
| name | VARCHAR |
| vendor\_code | VARCHAR |
| 9 | material | Id | INT | Храниться информация о поставляемом материале |
| supplier\_id | INT |
| name | VARCHAR |
| measure\_type\_id | INT |
| 10 | parts | Id | INT | Храниться информация о изготавливаемых частях |
| material\_id | INT |
| 11 | templates\_attributes | Id | INT | Храниться информация о размерах предлагаемых изделий |
| template\_id | INT |
| attribute\_id | INT |
| measure\_id | INT |
| attribute\_value | DOUBLE |

Продолжение таблицы 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 12 | furniture\_attributes | | Id | INT | Храниться информация о характеристиках фурнитуры |
| furniture\_id | INT |
| attribute\_id | INT |
| measure\_id | INT |
| attribute\_value | DOUBLE |
| 13 | | parts\_attributes | Id | INT | Храниться информация о характеристиках изготавливаемой детали |
| part\_id | INT |
| attribute\_id | INT |
| measure\_id | INT |
| attribute\_value | DOUBLE |
| 14 | | project | Id | INT | Храниться информация о конкретном проекте для определенного заказчика |
| category\_id | INT |
| material\_id | INT |
| user\_id | INT |
| master\_id | INT |
| name | VARCHAR |
| 15 | | project\_furnitures | Id | INT | Храниться информация о фурнитуре для конкретного проекта |
| project\_id | INT |
| furniture\_id | INT |
| amount | INT |
| 16 | | project\_parts | Id | INT | Храниться информация о изготавливаемых деталях для конкретного проекта |
| progect\_id | INT |
| part\_id | INT |
| attribute\_id | INT |
| attribute\_value | INT |
| amount | INT |

2.2 Структура программы

В базе данных были разработаны запросы на добавление, удаление, обновление и выборку. Перечень запросов представлен в таблице 2.

Таблица 2 – Запросы

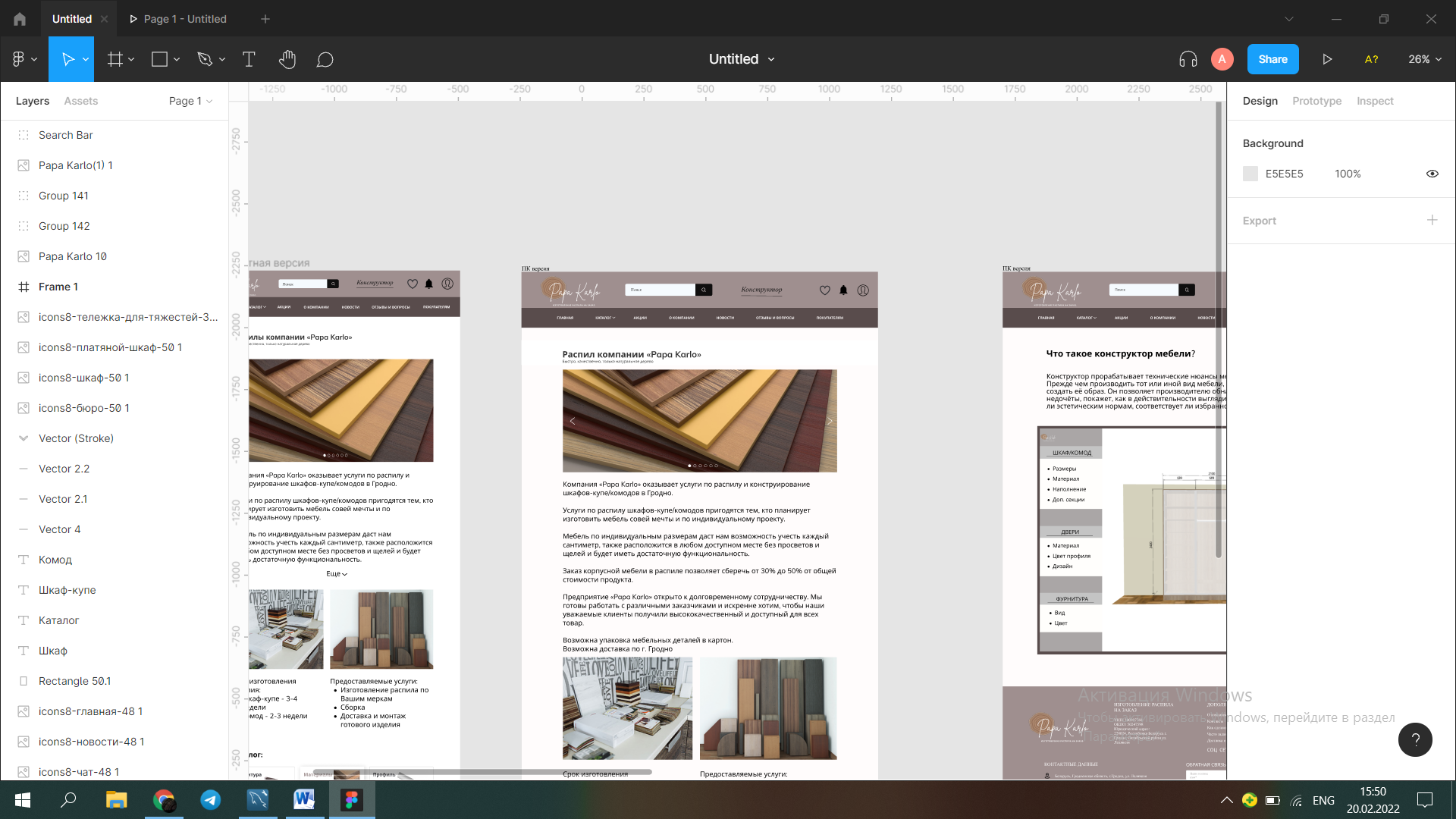
|  |  |  |
| --- | --- | --- |
| № | Название | SQL |
| 1 | Добавление пользователя в БД | INSERT INTO users (`fio`, `password`, `mail`) VALUES ('Грирорук Александра Денисовна', '89gfd8sfg5', 'grisha2003@gmail.com'); |
| 2 | Вывод информации о проекте с id=1, а точнее о частях которые были использованы в этом проекте, их размеры и количество | SELECT parts\_attributes.id, parts.id AS part\_id, parts.name AS part, attributes.name AS attribute, parts\_attributes.attribute\_value AS value, measure\_types.name AS measure\_type, project\_parts.amount AS amount FROM parts  INNER JOIN parts\_attributes ON parts\_attributes.part\_id = parts.id  INNER JOIN attributes ON attributes.id = parts\_attributes.attribute\_id  INNER JOIN measure\_types ON measure\_types.id = parts\_attributes.measure\_type\_id  INNER JOIN project\_parts ON project\_parts.part\_id = parts.id  INNER JOIN projects ON projects.id = project\_parts.project\_id  WHERE projects.id = 1 |
| 3 | Вывод информации о проекте, категории, мастере который выполняет заказ и заказчике под id 1 | SELECT users.id, users.fio, users.mail, projects.name AS project\_name, categories.name AS category\_name, master.name AS master FROM users  INNER JOIN projects ON projects.user\_id = users.id  INNER JOIN categories ON categories.id = projects.category\_id  INNER JOIN master ON master.id = projects.master\_id  WHERE users.id = 1 |
| 4 | Вывод информации о том из какого материала сделаны части проекта | SELECT parts.\*, materials.name FROM parts  INNER JOIN materials ON materials.id = parts.material\_id |
| 5 | Вывод информации о том какая фурнитура использовалась в проекте с конкретным заказчиком | SELECT \* FROM furnitures  WHERE furnitures.supplier\_id = 2 |
| 6 | Обновление почты пользователя с id=2 | UPDATE users SET `mail` = 'grusti1233@gmail.com' WHERE (`id` = '2'); |
| 7 | Удаление пользователя с id=2 | DELETE FROM users WHERE (`id` = '2'); |
| 8 | Вывод поставщиков поставляющих фурнитуру, с названием фунитуры | SELECT suppliers.\*, furnitures.name AS furniture, furnitures.vendor\_code AS vendor\_code FROM suppliers  INNER JOIN furnitures ON furnitures.supplier\_id = suppliers.id |
| 9 | Вывод информации из БД с названием furniture\_attributes | SELECT furnitures.\*, attributes.name as attribute, furniture\_attributes.attribute\_value AS value, measure\_types.name AS measure\_type FROM furnitures  INNER JOIN furniture\_attributes ON furniture\_attributes.furniture\_id = furnitures.id  INNER JOIN attributes ON attributes.id = furniture\_attributes.attribute\_id  INNER JOIN measure\_types ON measure\_types.id = furniture\_attributes.measure\_type\_id |

**2.3 Структура и описание процедур и функций пользователя**

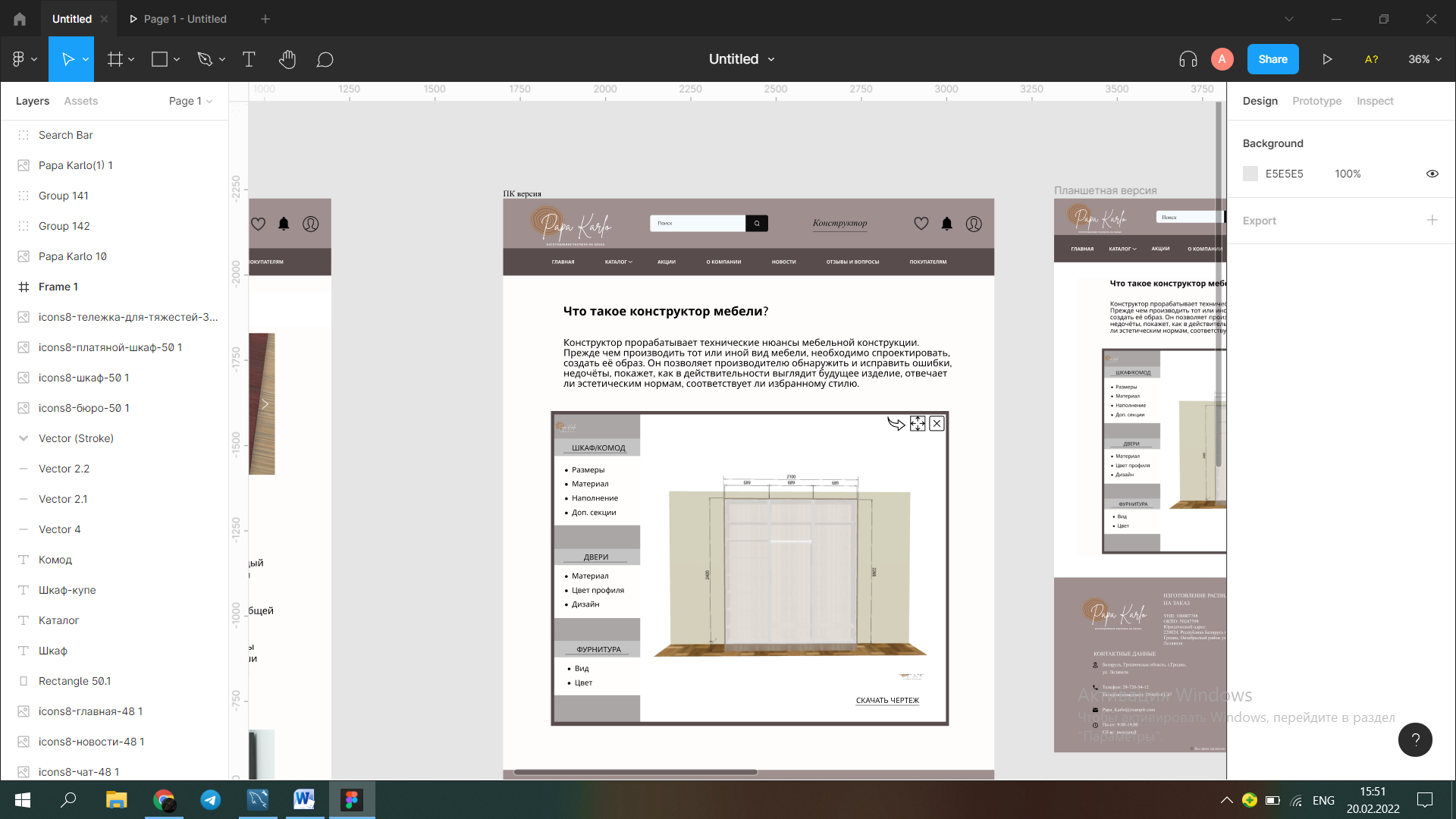
3. Руководство пользователя

3.1 Запуск конструктора

Запуск конструктора осуществляется автоматически при переходе на страницу «Конструктор» с главной страницы сайта.



При переходе на данную страницу отображается окно конструктора, где пользователь может изменять параметры создаваемого объекта на свое усмотрение, так же пользователь может выбрать, то что он хочет создать шкаф/комод.



3.2 Инструкция по работе с программой

В меню конструктора пользователь может изменять параметры по своему усмотрению.

Рассмотрим блок шкаф/комод (рис.1):



Рисунок 1- блок Шкаф/комод

На пример во вкладке размеры он может устанавливать индивидуальные размеры (рис. 2).



Рисунок 2- вкладка «Размеры»

Во вкладке материал, пользователь может подобрать свой идеальный оттенок (рис.3).



Рисунок 3- вкладка «Материал»

Во вкладке наполнение пользователь может редактировать внутренне строение шкафа (рис.4).



Рисунок 4 - вкладка «Наполнение»

Вкладка доп. секции отвечает за добавление новых секций (рис.5).



Рисунок 5- вкладка «Доп. секции»

Рассмотрим блок двери (рис.6):



Рисунок 6- блок Двери

Вкладка материал позволяет пользователю выбрать материал дверей (рис.7).



Рисунок 7- вкладка «Материал»

Вкладка профиль содержит в себе виды профиля доступные к заказу на данный момент, там пользователь может подобрать как и цвет, так и модель подходящую именно к его шкафу (рис.8).



Рисунок 8- вкладка «Цвет профиля»

Вкладка дизайн предлагает пользователю готовые варианты комбинации дсп и стекла, тут пользователь ориентируясь на свои личные предпочтения подбирает подходящий дизайн именно ему (рис.9).



Рисунок 9- вкладка «Дизайн»

Рассмотри раздел «Фурнитура» (рис.10):



Рисунок 1- блок Фурнитура

Во вкладке фурнитура пользователю предоставляется просмотр имеющийся в наличии фурнитуры. Пользователь подбирает подходящую ориентируясь на свои предпочтения (рис.11).



Рисунок 11- вкладка «Вид»

Вкладка цвет предполагает выбор цвета фурнитуры из наличия (рис.12).



Рисунок 12- вкладка «Цвет»

Функция «Скачать/заказать чертеж» (рис.13)



Рисунок 13- функция «Скачать/заказать чертеж»

Данная функция предлагается пользователю для того, чтобы он мог заказать или скачать свою работу и в последующем ее собрать, т.е. данная функция идет как создание подробной инструкции по сборке готового распила.

Так же пользователь может «поделиться» своим проектом, сделать конструктор на целый экран своего устройства или выйти из него (рис.14).



Рисунок 14- функция «Поделиться», «На весь экран», «Закрыть»

3.3 Завершение использования программы

Завершить использования конструктора легко, достаточно будет нажать на крестик в верхнем правом углу окна конструктора или просто прейти на другую страницу сайта. Волноваться, что программа удалит чертеж не надо так как в конструкторе присутствует автосохранение прогресса разработки изделия и сохраненный проект всегда можно найти в личном кабинете (рис.15).



Рисунок 15- функция «Закрыть»

Тестирование программного продукта методом белого ящика.

|  |  |  |  |
| --- | --- | --- | --- |
| № | Наименование функции | Описание тестового сценария с исходными данными | Ожидаемый результат |
| 1 | Добавление пользователя в БД | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Добавление пользователя в БД |
| 2 | Вывод информации о проекте с id=1, а точнее о частях которые были использованы в этом проекте, их размеры и количество | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Вывод информации о проекте с id=1, а точнее о частях которые были использованы в этом проекте, их размеры и количество |
| 3 | Вывод информации о проекте, категории, мастере который выполняет заказ и заказчике под id 1 | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Вывод информации о проекте, категории, мастере который выполняет заказ и заказчике под id=1 |
| 4 | Вывод информации о том из какого материала сделаны части проекта | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Вывод информации о том из какого материала сделаны части проекта |
| 5 | Вывод информации о том какая фурнитура использовалась в проекте с конкретным заказчиком | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Вывод информации о том какая фурнитура использовалась в проекте с конкретным заказчиком |
| 6 | Обновление почты пользователя с id=2 | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Обновление почты пользователя с id=2 |
| 7 | Удаление пользователя с id=2 | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Удаление пользователя с id=2 |
| 8 | Вывод поставщиков поставляющих фурнитуру, с названием фурнитуры | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Вывод поставщиков поставляющих фурнитуру, с названием фурнитуры и ее акртикулом |
| 9 | Вывод информации из БД с названием furniture\_attributes | 1. Нажатие на кнопку триггер 2. Вывод итоговой таблицы БД, полученной в ходе запуска | Вывод информации из БД с названием furniture\_attributes |