Титульный лист

МИНИСТЕРСТВО ОБРАЗОВАНИЯ МОСКОВСКОЙ ОБЛАСТИ

Государственное бюджетное профессиональное образовательное учреждение Московской области «Ногинский колледж»

Курсовой проект

по МДК.09.01 Проектирование и разработка веб-приложений

ПМ.09 Проектирование, разработка и оптимизация веб-приложений

Тема:

Использование фреймворка Yii2 в разработке веб-приложения «Интернет-магазин «ТЕХНО У АНТОШКИ»

Разработчик: студент группы 3ИСПР2

Безруков Д.А

(подпись)

Оценка защиты курсового проекта

« »

Дата защиты

« » 2024

Руководитель проекта:

преподаватель Степанов С.О.

(подпись)

Ногинск, 2024 г

Содержание

Оглавление

[Введение 3](#_Toc169383963)

[Основная часть 5](#_Toc169383964)

[**1.** **Инсталляция всех необходимых программ и фреймворка, предпроектная подготовка к работе** 5](#_Toc169383965)

[**2.** **Разработка сайта** 6](#_Toc169383966)

[2.1. Html и Scss. 6](#_Toc169383967)

[2.2. Php (YII2) 10](#_Toc169383968)

[2.3. JavaScript (JQuery) 19](#_Toc169383969)

# Введение

**Цель:** Главная задача в курсовой работе изучить новый backend и framefork и с помощью них разработать сайт с требуемым функционалом.

**Актуальность:** Курсовая работа актуальна тем что сегодня у людей есть возможность получить практически любой товар или услугу онлайн, что делает интернет-магазины незаменимым инструментом для бизнеса. Множество современных магазинов имеют или создают свои интернет-магазины.

На сегодняшний день система интернет-магазинов на рынке представлена многочисленными площадками, предлагающими разнообразные товары и услуги. В связи с ростом популярности онлайн-покупок, конкуренция среди интернет-магазинов становится все более острой, что заставляет их постоянно совершенствоваться и применять новые подходы для привлечения клиентов.

Система интернет-магазинов характеризуется широким разнообразием товаров, начиная от одежды и обуви, заканчивая бытовой техникой и товаром для дома. Многие площадки также предлагают услуги доставки, собственные бренды, программы лояльности и скидок, что помогает им удерживать клиентов и стимулировать продажи.

Важным трендом в системе интернет-магазинов является стремительное развитие мобильной торговли и мобильных приложений. Пользователи предпочитают совершать покупки через мобильные устройства из-за удобства и доступности, поэтому многие интернет-магазины активно развивают мобильные версии своих сайтов или создают собственные приложения.

Система интернет-магазинов также сталкивается с вызовами в области безопасности данных и защиты конфиденциальности пользователей. В связи с ростом онлайн-мошенничества и утечек данных, у интернет-магазинов возрастает ответственность за защиту личной информации клиентов и обеспечение безопасных платежей.

В целом, система интернет-магазинов на рынке демонстрирует стабильный рост и разнообразие, предлагая покупателям широкий выбор товаров и услуг, удобные способы оплаты и доставки, а также возможность пользоваться скидками и акциями. Для успешной деятельности в этой системе важно постоянно следить за трендами, улучшать сервис и адаптироваться к изменяющимся потребностям клиентов.

Задачи:

1. Выбрать тему курсовой работы.
2. Изучить yii2.
3. Изучить jQuery +jQuery UI.
4. Сделать сайт с помощью выбранных backend и framefork.
5. Написать и защитить курсовой проект.

# Основная часть

## **Инсталляция всех необходимых программ и фреймворка, предпроектная подготовка к работе**

Первым что я скачал это было PhpStorm — коммерческая кросс-платформенная интегрированная среда разработки для PHP. Разрабатывается компанией JetBrains на основе платформы IntelliJ IDEA. Для установки надо зайти на официальный сайт скачать exe файл и запустив его установить приложение.

Для удобной работы с PhpStorm нужно скачать Composer — это пакетный менеджер уровня приложений для языка программирования PHP, который предоставляет средства по управлению зависимостями в PHP-приложении. Для его установки так же зайдя на официальный сайт скачать exe файл и при установке так же я скачал нужную мне версию Php7.4 чтобы выбрать его для работы composer.

Установка yii2 — это высокоэффективный, основанный на компонентной структуре PHP-фреймворк для быстрой разработки крупных веб-приложений. Он позволяет максимально применить концепцию повторного использования кода и может существенно ускорить процесс веб-разработки. Благодаря composer будет сделана с помощью команды в терминале composer create-project --prefer-dist yiisoft/yii2-app-basic basic. Которая создаст всё необходимое для работы с yii2.

Для установки jQuery — это бесплатная JavaScript-библиотека с открытым исходным кодом. Она заметно упрощает веб-разработку, позволяя писать меньше кода, чем на ванильном JavaScript. Нужно с официального сайта скачать сжатый файл с кодом библиотеки, данный файл я поместил в папку js. После чего для работы jQuery в тег head нужно поместить <script defer src="js/jquery-3.6.3.js"></script>.

Также для работы с sass чтобы он компилировал в css нужно скачать Node.js — программная платформа, основанная на движке V8, превращающая JavaScript из узкоспециализированного языка в язык общего назначения. Чтобы установить надо зайти на официальный сайт и оттуда скачать exe, после его установки в PhpStorm файлы scss стал компилироваться в css.

## **Разработка сайта**

### Html и Scss.

Первым что я сделал был header для него сделал логотип в виде названия после добавил меню навигации в котором расположены главная, товары, логин, выход из аккаунта, мой профиль, после идёт блок где находятся ссылки на соц.сети такие как vk и whatsapp дальше идёт номер телефона, а в конце расположен вход для админа и в корзину. Для стилизации я задал задний фон чёрным,весь текст сделал белым, в левую часть поставил логотип. Меню навигации выстроил в линию после чего сделал небольшой отступ для соц.сетей и для номера, а в правую часть поставил корзину и переход на админ панель.



Рис1.header

Следующая секция promo в первый блок в виде заголовка я добавил название сайта и к нему небольшой слоган. Вторым блоком я сделал список где написаны плюсы нашего сайта. Для стиля на задний фон я добавил картинку во всю секцию, первый блок я поместил по центру секции и задал шрифт и большой размер текста под названием сайта поставил слоган, а уже список я поместил ниже с помощью отступов и выстроил этот список в ряд ограничил для текста ширину.

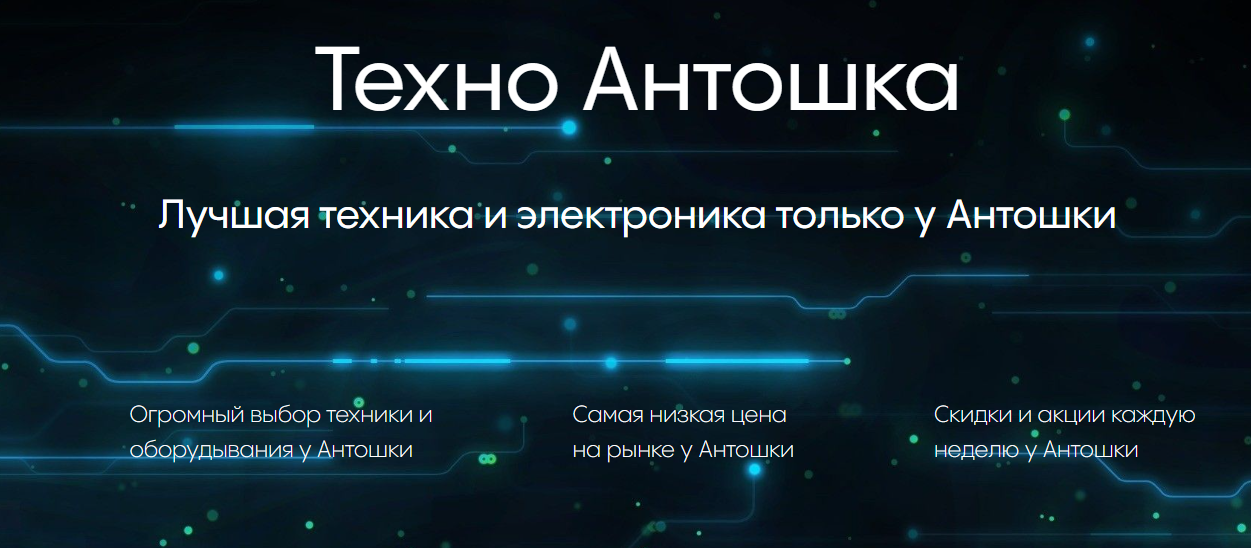


Рис2.promo

Вторая секция предназначена для карточек товара и сортировки но первым делом я сделал только сортировку. Первый блок в секции имеет в себе заголовок и кнопку смотреть всё. Второй блок сделан списком в котором находится кнопка для с названием категории под ней блок для выбора подкатегорий для каждого товара. И в конце блок с кнопками для сортировки по цене и по алфавиту. Для стиля заголовок и кнопку смотреть всё я поместил в линию сделав небольшой отступ между ними, дальше в ряд стоят кнопки категорий которые я закруглил, сделал чёрную обводку а задний фон белый в центр кнопок поставил текст, под каждой категорией расположил свои подкатегории и уже ниже отдельным рядом сделал кнопки сортировки подобно категориям и так же я заранее прописал для них смену цвета при активации но сделал функционал этого позже через jquery. Карточками я занялся позже когда перешёл к php.

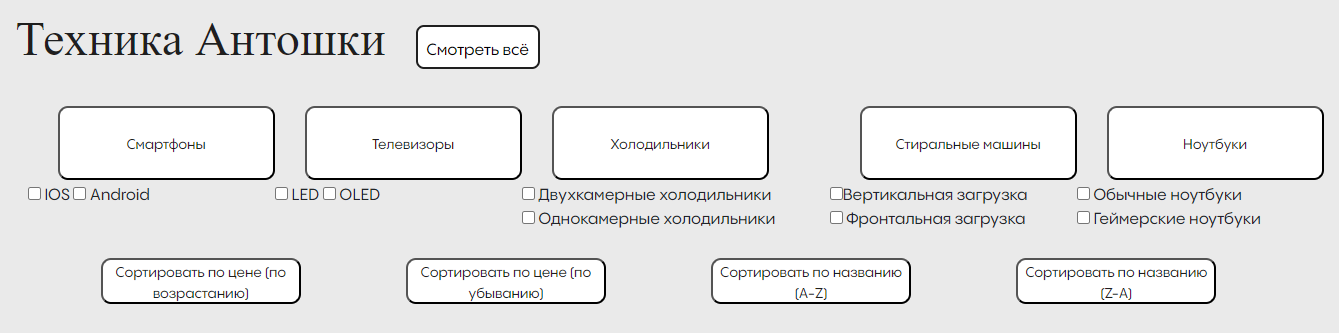


Рис3.Вид сортировки

Четвёртая секция акции. В ней два блока, в первом блоке заголовок про выгодные предложения и для него описание. Во втором блоке находится фотка товара, его название и микро блок для описания и цены без скидки со скидкой. Для заднего фона я выбрал поставить фотографию, сами же блоки расположил друг на против друга так же для заголовка последние два слова сделал фиолетовым текст фиолетовым а под ним расположил описание ограничив ему и заголовку ширину. Справа я решил сделать задний фон белый и закруглить этот блок, на большую часть блока сделал фотку, а под ней название товара и внизу названия описание справа от которого цена.



Рис4.Вид акции

Пятая секция про огромный выбор техники. Сначала идёт заголовок, а после два одинаковых блока в которых фотографии, заголовок и текст. Под заголовком поместил эти два блока стоящие в ряд с отступом друг от друга, каждый блок имеет задний белый фон и закругленный вид, большую часть блока занял фотографией а уже под ней расположил заголовок и в самом конце блока текст.

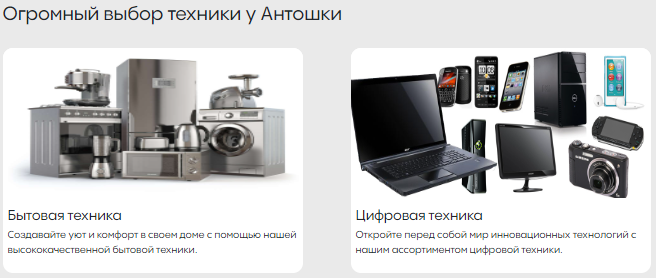


Рис5.Вид пятой секции

Следующая секция подразумевает причины покупать на сайте. Для неё я сделал список где в каждом блоке два div в первом изображение с иконкой под причину, а во втором название причины и её описание. Для этого заголовок я оставил заголовок вверху, а уже сами причины с помощью grid выстроил четыре причины по две с каждой стороны, сами иконки и текст я поставил в линию и изменил размер текста его ширину и размер иконок

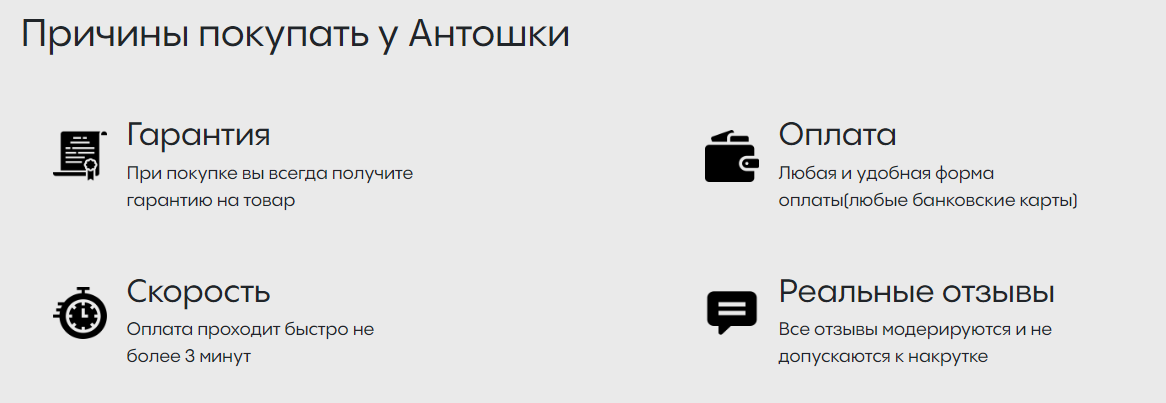


Рис6. Вид причин

Последней секцией является связь с поддержкой если возникли вопросы. В самой секции есть два блока один из них это фотография сотрудницы, а во втором заголовок номер телефона с информацией как быстро с вами свяжутся, ниже находится форма заполнения имени и номера телефона, а так же кнопка оставить запрос и под кнопкой возможность поставить галочку с соглашением на обработку данных. Для стиля я в ряд выставил фотку и второй блок для которого сделал небольшой отступ. Сначала идёт заголовок под которым небольшим размером сделал номер телефона и описание. Сделав средний отступ я разместил форму которую закруглил как и кнопку под формой для которой задал чёрный цвет под кнопкой я расположил галочку и половину текста сделал серым, а другую черным

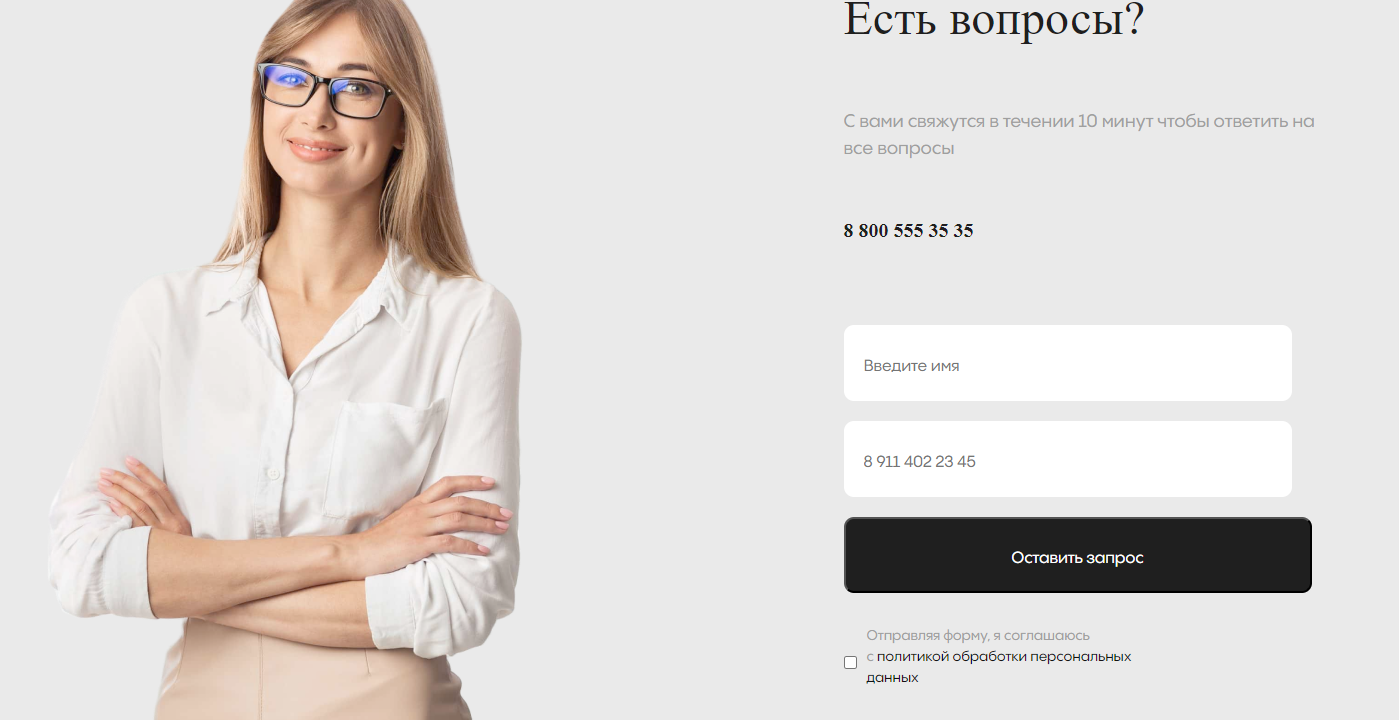


Рис7.Вид связи с поддержкой

Последние это footer для него я поделил на три блока в первом у меня находятся заголовок контакты под ним ссылка на карту с пометкой пункта выдачи так же номер телефона и кнопка задать вопросы. Во втором блоке списком предоставлена навигация по сайту разделенная на три небольших списка. В третьем блоке идут соц.сети и адрес с годом создания сайта. Для стилизации первые два блока я выстроил в ряд, первый блок идёт в колонку где кнопка имеет белый фон и фиолетовую обводку где текст находится в центре. Во втором блоке небольшие списки я выстроил в колонну но расположил их рядом друг с другом. Последний третий блок я огородил линией где внизу на одном конце расположил соц.сети, а в другом конце адрес и под ним название сайта с годом создания

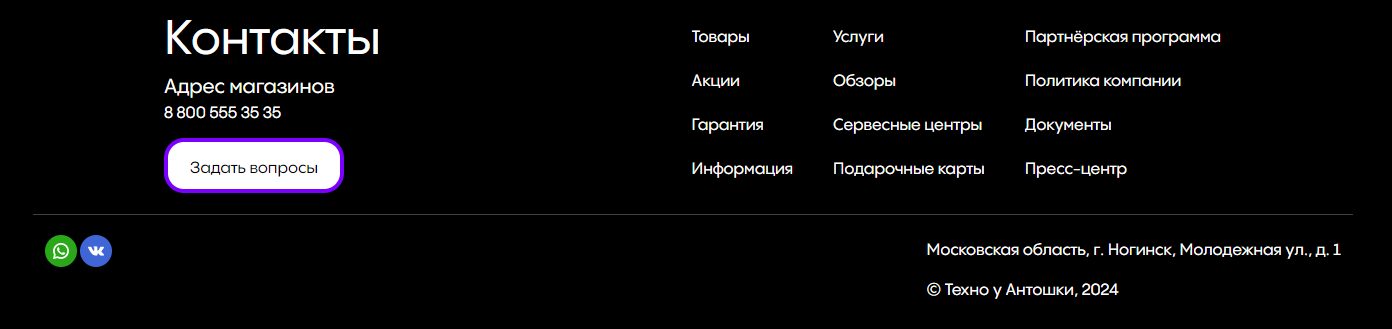


Рис8. Вид footer

### Php (YII2)

Первым делом я занялся выводом карточек товаров из базы данных. Бд была сделана с помощью phpmyadmin и подключена в phpstorm. Для вывода сначала я написал код который будет подключать в файле мою базу данных и чтобы все данные брал из таблицы texno.

<?php  
$pdo = new PDO('mysql:host=localhost;dbname=tovarin', 'root', '');  
  
// Получение данных о товарах из базы  
$statement = $pdo->query('SELECT \* FROM TEXNO');  
$texnos = $statement->fetchAll(PDO::*FETCH\_ASSOC*);  
  
?>

Затем я создал div в котором карточки представлены в виде списка, перед началом li я открыл php который берёт переменную texnos и заменил её на index и texno. В самой карточке идёт сначала блок с фотографией товара, затем название, модель, минимальное описание, цена и две кнопки подробнее и добавить которые отвечают за добавление в корзину и открытие отдельной странички с всеми данными товара. После в местах где должна находится информация я открывал php и вводил переменную texno для выбора столбца из которого выводятся данные о товаре.

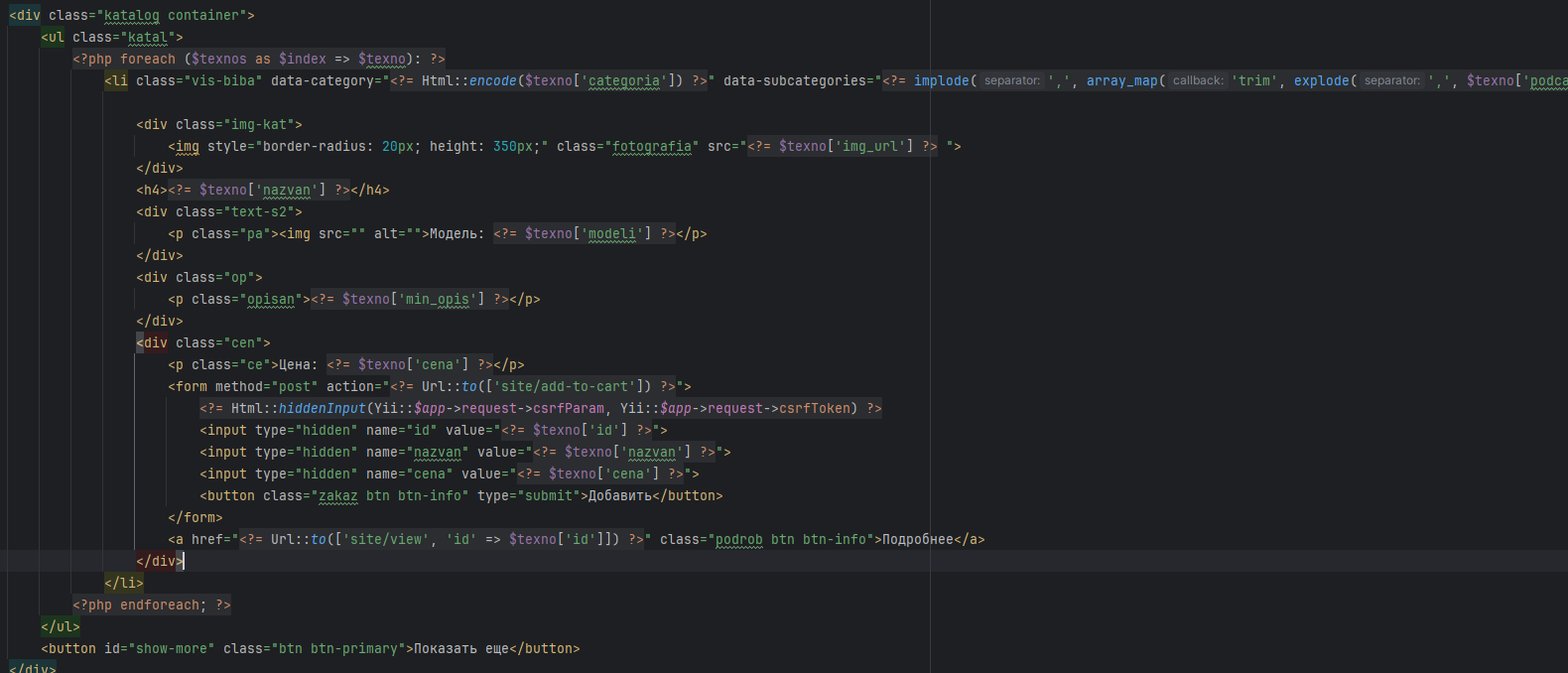


Рис9.Код для карточки товара

После я занялся корзиной и добавлением товара. Для начала я создал модель Texno в которой первым делом вызвал таблицу texno для которой написал правило брать столбцы и для них прописал тип данных который используется, а так же ниже прописал атрибуты для Labels. Следующим шагом я создал модель Сart в которой написал функцию добавления товара где используется как идентификатором является id товара, после задал если id товара совпадает то он добавляет количество для товара и после иначе он берёт id из таблицы имя, цену и задал изначальное количество равное 1, затем написал функцию removeItem которое удаляет товар в соответствии с его id. Следующим этапом я начал добавлять функции для котроллера. Первая функция предназначена для добавления товара на страничку корзины. Для работы он берёт с помощью метода post id товара из таблички Texno дальше задал цикл if в котором для переменной cart задал ключ cart и создание новой записи которую он берёт из атрибутов модели cart и выводится в корзине.

public function actionAddToCart()

{

$id = Yii::$app->request->post('id');

$tovar = Texno::findOne($id);

if ($tovar) {

$cart = Yii::$app->session->get('cart', new Cart());

$cart->addItem($tovar->attributes);

Yii::$app->session->set('cart', $cart);

}

return $this->redirect(['site/cart']);

}

Второй функцией было удаление товара из корзины где я так же задал ключ для переменной cart и чтобы эта переменная брала удаление товара в соответствии с функцией remove в модели cart и выводил всё это в корзине. Третьей функцией было обновление товара, она предназначена для того чтобы на самой страницы корзины можно было выбрать кол-во товара которое вы хотите и обновляет количество товара.



Рис10.Код для функций remove и update

И последней функцией идёт Cart которая выводит работу функций на страничку корзины. Последним этапом я сделал представление для корзины в котором сначала идет заголовок с тем что это корзина, затем открыл php с цикл if где если никого товара нет то пишется что корзина, а уже если товар был добавлен я написал табличку где прописаны названия для полей которые выводят и затем открыл php с циклом foreach где берётся массив с товаром и заменил его на массив item. После написал вывод id, название, цены и написал форму для изменения количества товара который обращается к функции update и берёт id товара и при смене его кол-во он умножает эту цену на значение количества, для изменения этого значения я добавил кнопку изменить, а так же добавил кнопку для удаления товара которая использует функцию remove.

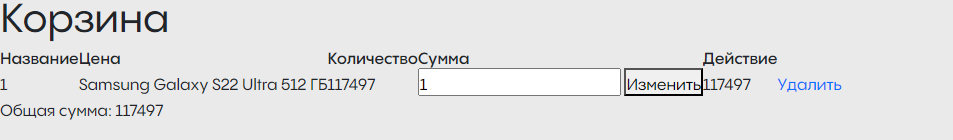


Рис11. Вид корзины

Следующим этапом я сделал регистрацию и логин. Для начала я создал новую таблицу в базе данных где находится username, password, ключ аутентификации, роли и дата создания и обновления. Следующим этапом я создал модель user для которой прописал константу ролей user и admin, затем вызвал таблицу user для которой создал функцию правил для столбцов которые используются в бд где и использовал для определения ролей константы. Затем прописал функции для нахождения и валидации данных столбцов таких как id, username, password, ключ идентификации, а также для скрытия пароля и ключа в базе данных. После я создал модель для регистрации где сначала вызвал атрибуты имени, пароля и повторения пароля. Задал функцию правил для этих атрибутов так чтобы username ещё выбирался из модели user. Затем сделал функцию регистрации где с помощью цикла if создавалась новая запись пользователя где надо будет ввести username, пароль и повторить этот пароль, ключ аутентификации присваивается автоматически, если всё правильно то данные сохраняются в таблице если нет то выводится ошибка

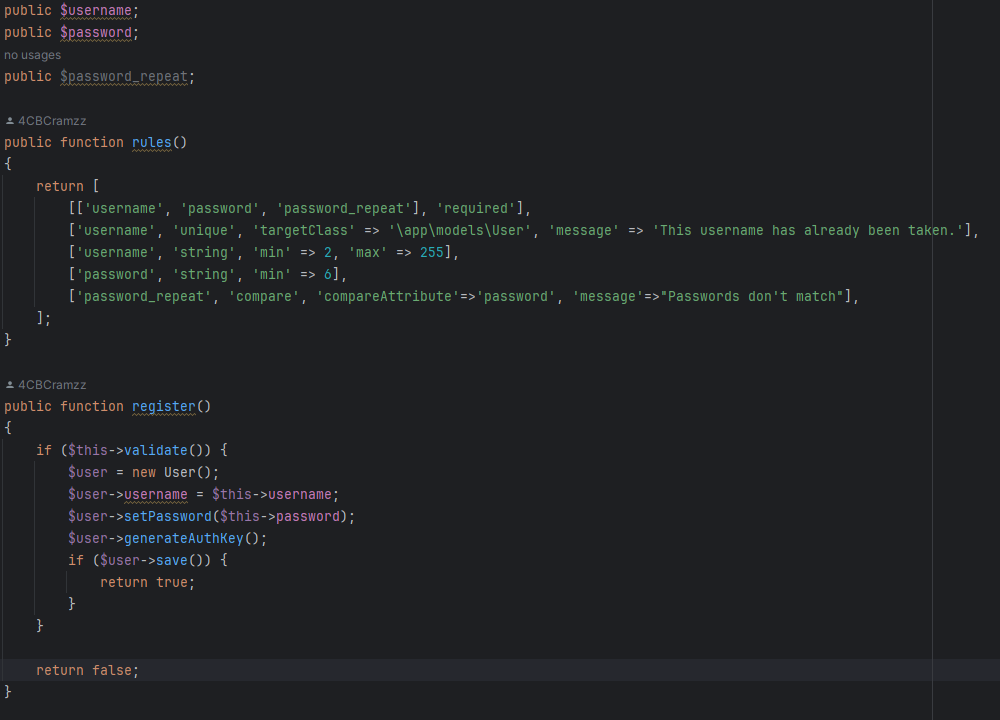


Рис12. Код модели регистрации

Для регистрации в контролере я прописал функцию которая создаёт нового пользователя после внесённых данных и если всё хорошо то пользователя переносит на страницу логина и прописал вывод на страницу register. Для представления странички я открыл div в котором сначала идёт заголовок затем небольшой текст, а после открыл php для которого задействовал функцию регистрации из контролера и прописал формы для заполнения которые использую атрибуты из модели, после сделал кнопку для подтверждения регистарции.

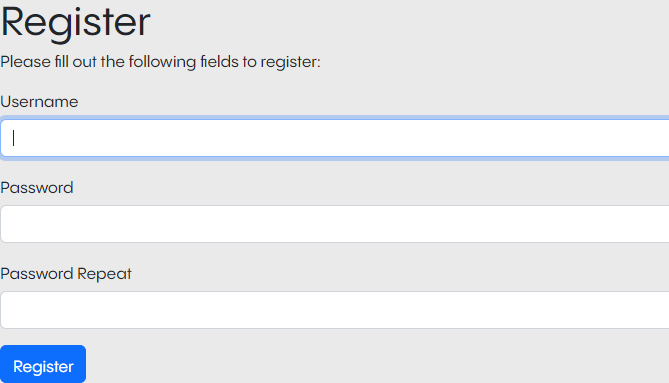


Рис13.Вид регистрации

Следующим я сделал модель логина в которой вызвал атрибуты username, password, галочки remeberme. Написал функцию правил для атрибутов так чтобы пароль проходил валидацию, для этого создал ещё одну функцию валидации пароля в которой цикл if проверяет данные из таблицы user и если всё правильно то не выдаёт ошибки. И также создал функцию для логина которая даёт пользователю войти в аккаунт. Для логина в контролере создал функцию которая создаёт новую форму в которой если пользователь зашёл в аккаунт то его кидает на главную модель, но если у пользователя роль админа то его кидает на страницу админа. Выводится всё в станице login. В представлении создал страничку логина где сначала идёт заголовок затем небольшой текст и открыл php в написал форму для заполнения где берутся атрибуты из модели login. Дальше для вида создал возможность перезаписать пароль и ниже кнопку для подтверждения логина, а также переход на регистрацию совершается с станицы логина.

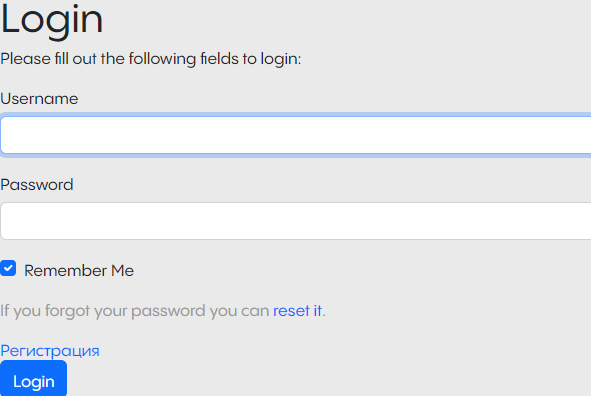


Рис14.Вид логина.

Следующим необходимо создать оформление заказа для чего и были созданы регистрация и логин. Для начала я создал таблицу ordres в которой id заказа, id пользователя который в аккаунте (связал с таблицей user), name, phone, address, items, total, дата создания и статус заказа. После я написал модель order в которой сделал константы для статуса заказа, затем вызвал таблицу orders и создал функцию правил для всех столбцов из таблицы, для статуса прописал константы. Второй моделью я сделал orderform которая понадобится для формы заполнения оформления заказа. В ней я прописал атрибуты для name, phone и address для которых задал правила. В контролере для функции cart добавил значения функции orderform, затем создал функцию placeorder в которой создаётся новая запись для модели orderform, с помощью цикла if в которой создаётся новая запись для таблицы orders где заполняется id пользователя из таблицы user, имя, телефон и адрес добавляются в соответствии с заполненными данными пользователем. Для заполнения данных товара используется json который берёт данные из карточки товара, а статус берёт константы из модели order и так же для записи создания используется time. Если всё правильно пользователя переносит на страницу thank-you для этой станицы я так же создал функцию которая выводит всё на представление.

public function actionPlaceOrder()  
{  
 $model = new OrderForm();  
  
 if ($model->load(Yii::*$app*->request->post()) && $model->validate()) {  
 $order = new Order();  
 $order->user\_id = Yii::*$app*->user->id;  
 $order->name = $model->name;  
 $order->phone = $model->phone;  
 $order->address = $model->address;  
 $order->items = json\_encode(Yii::*$app*->cart->getItems());  
 $order->total = Yii::*$app*->cart->getTotal();  
 $order->status = Order::*STATUS\_PENDING*; // Set the default status  
 $order->created\_at = time();  
  
 if ($order->save()) {  
 Yii::*$app*->cart->clear();  
 return $this->redirect(['site/thank-you']);  
 }  
 }  
  
 return $this->render('checkout', [  
 'model' => $model,  
 ]);  
}

В представление корзины я добавил div в котором открыв php я создал форму заполнения имени телефона и адреса которые используют атрибуты из модели orderform. Так же эта форма подключена к функции из контролера placeorder, и под формами заполнения я добавил кнопку для подтверждения оформления заказа. А так же для представления thank-you я сделал заголовок, текст о том что заказ был успешно оформлен и кнопку для возвращения на главную страницу.

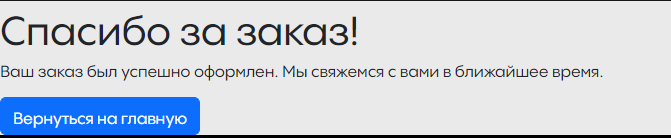


Рис15.Вид странички thank-you

Следующая задача была создать страничку профиля пользователя. Для неё уже есть модель order и таблица в базе данных orders откуда и будет браться информация. Сначала в контролере я прописал функцию profile в которой в соответствии с id пользователя для того чтобы из таблицы orders он находил данные связанные с id юзера и датой создания заказа, и написал вывод в представление profile. Затем я начал писать само представление где есть заголовок мои заказы, после я открыл php и прописал через цикл if если ничего нет то выдаёт у вас нет заказов. В случае если всё же есть заказ я написал табличку с названиями информации которая выводится и открыл php с циклом foreach в котором прописал вывод данных из переменной order такие как id, дата создания, сумма и статус заказа.

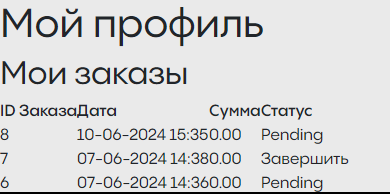


Рис16. Вид профиля

Следующим что я сделал это была панель администратора для которой мне нужно было сделать две задачи, первая эта возможность работать с таблицей товаров, а вторая это возможность изменять статус заказа у пользователей. Для первой задачи мне понадобилась модель texno, в контролере я создал функции создания, обновления и удаления которые имеют тот же принцип действия и похожий код который был использован для корзины. После я создал представление для панели администратора где сначала идёт заголовок затем кнопка для перехода к страничке второй задачи. Затем открыл php где с помощью цикла if я прописал кнопку для создания нового товара с использованием функции create из контроллера, а потом написал какие колонки должны выводится, а это все из таблицы texno. Так же прописал кнопки для обновления и удаления которые используют функции из контроллера, затем прописал else с циклом if где если пользователь нажимает создать или обновить товар то у него открывается формы для заполнения всех столбцов из таблицы, так же для категорий и подкатегорий прописал какие именно значения берутся из таблицы и написал им тоже самое название. Дальше создал кнопку которая отвечает либо за создание либо за обновление товара.

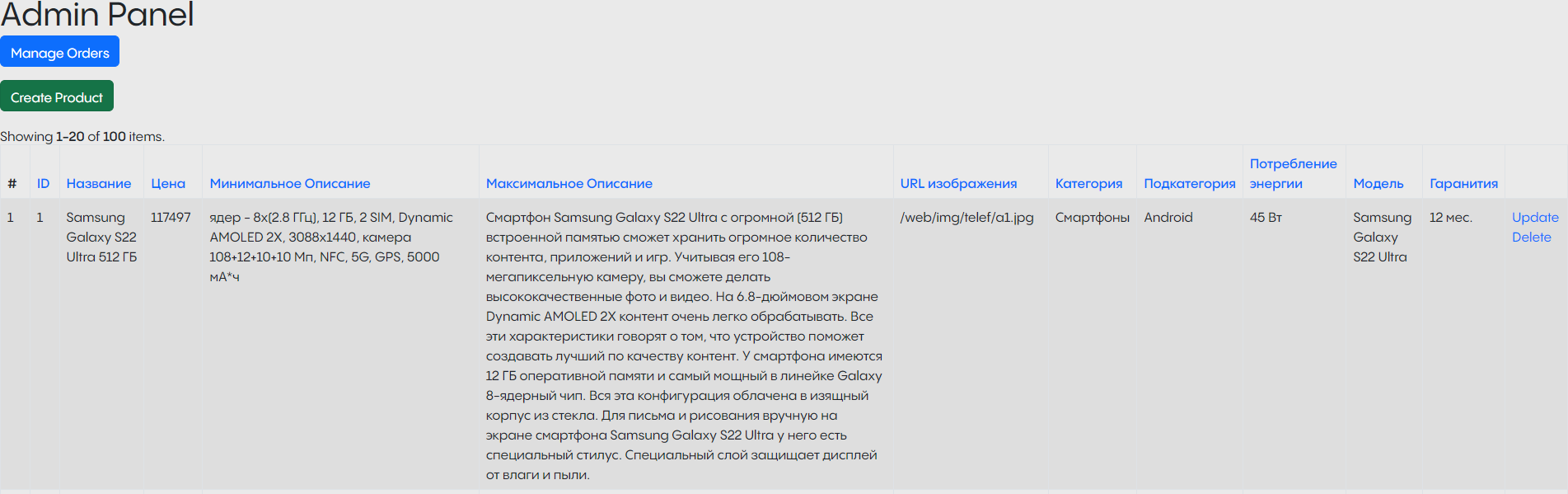


Рис17.Вид панели администратора

Для второй задачи уже есть модель order, в контроллере я прописал первую функцию manageorders в котором берутся данные и таблички orders и выводится на страничке представления. Вторая функция отвечает за изменения статуса заказа, для этого я прописал через метод post вызов id и status которые он берёт из модели order, затем через цикл if прописал все константы статуса заказа и ещё раз открыл if где при сохранения изменения выдаёт либо что всё прошло успешно либо ошибка, а так же прописал ошибку если заказ некорректный или статус и так же вывод для представления.

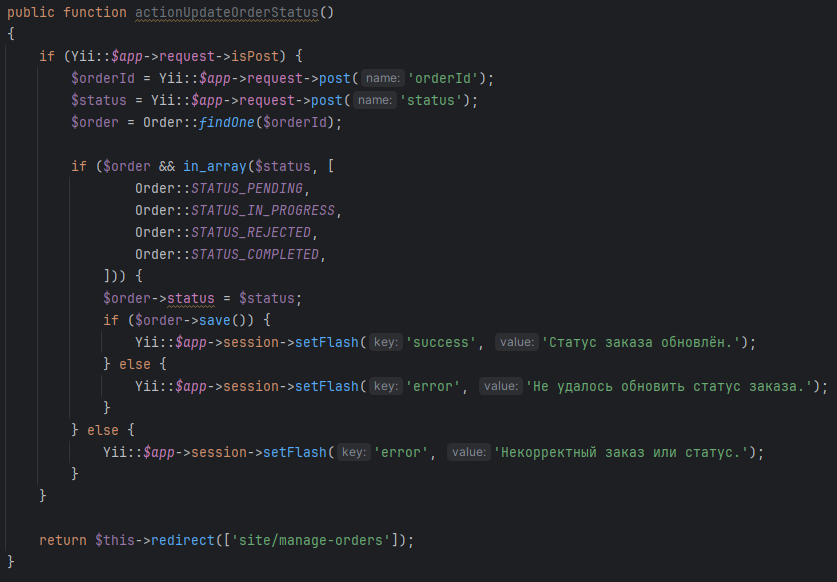


Рис18.Код для обновления статуса заказа

Следующим я создал представление в котором сначала идёт заголовок, потом открыл php в котором как и для панели администратора написал вывод всех столбцов из таблицы orders и добавил кнопку для перехода к обновлению статуса заказа. Затем в следующем div я написал с помощью php форму для изменения статуса заказа который использует функцию из контролера и кнопку для обновления этого статуса.

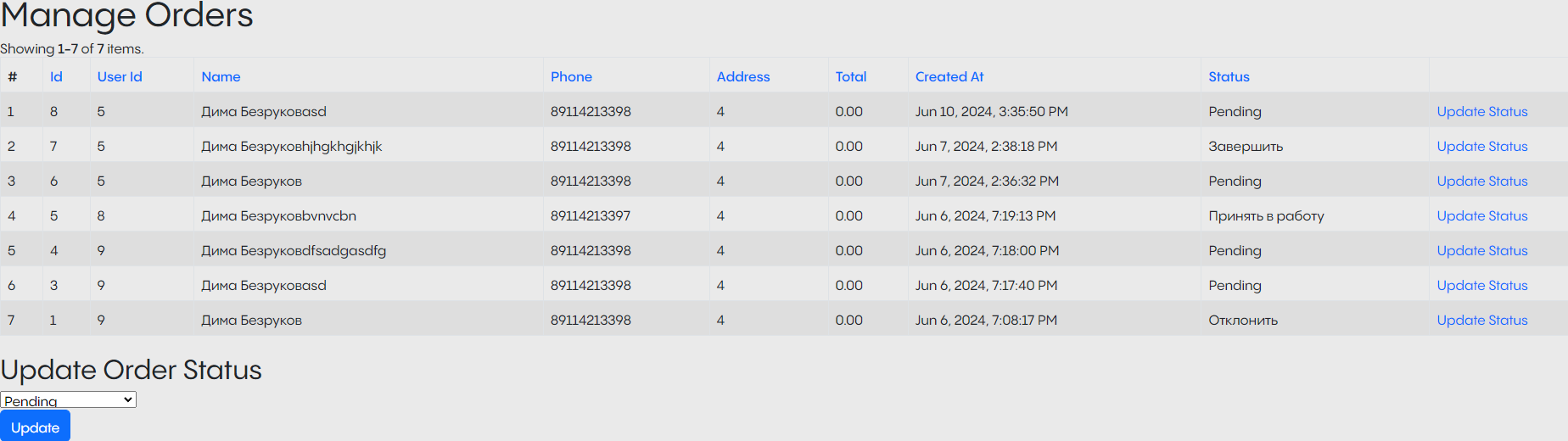


Рис19.Вид manageorders

А так же для header я прописал ссылку для перехода на панель администратора которая видна только если у пользователя роль админа.

Предпоследней задачей я сделал поиск товара по его имени. Для этого в контролере я прописал функцию search в которой сначала прописал query которая используется для хранения запроса в поиске, а затем прописал чтобы при поиске он брал данные из таблицы texno а именно данные столбца nazvan и выводил это в представление search. Для него я написал заголовок с названием товара который искал пользователь и открыв php прописал цикл if где если такой товар не существует то выдаёт что по запросу ничего не найдено, а если есть то выводятся с помощью php данные из таблички texno такие как название, минимальное описание и цена, а так же добавил кнопку для добавления товара в корзину. Для того чтобы поиск можно было осуществить в header я добавил форму которая переходит на страничку представления search, так же строку заполнения для названия товара и кнопку найти.

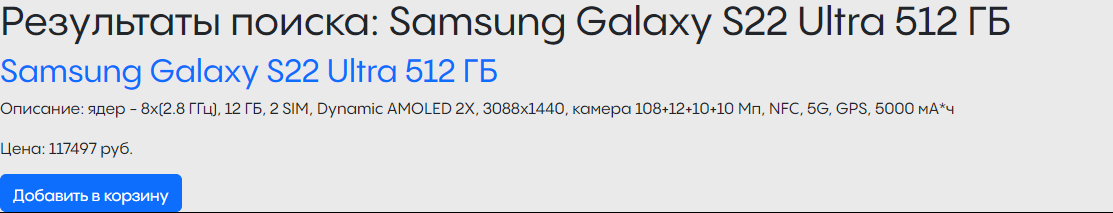


Рис20.Вид странички поиска

Последняя задача была отдельная страничка где более подробно написано о товаре. Для неё в контролере я задал поиск из таблички texno с помощью id товара и вывод на страничку представления view. В нём я прописал заголовок заголовок который берёт название из столбца nazvan, затем создал контейнер в котором разделил два div первый фотография товара берущеюся из img\_url и второй где предоставлена вся остальная информация о товаре из таблицы texno.

<?php  
// In views/product/view.php  
*/\* @var $this yii\web\View \*/  
/\* @var $product app\models\Product \*/*use yii\helpers\Html;  
  
$this->title = $texno->nazvan;  
?>  
<h1 class="container"><?= Html::*encode*($this->title) ?></h1>  
<div class="product-view container">  
 <div class="product-image">  
 <img src="<?= $texno->img\_url ?>" alt="<?= Html::*encode*($texno->nazvan) ?>" style="width:600px; border-radius: 20px;">  
 </div>  
 <div class="product-details">  
 <p><strong>Модель:</strong> <?= Html::*encode*($texno->modeli) ?></p>  
 <p><strong>Описание:</strong> <?= Html::*encode*($texno->min\_opis) ?></p>  
 <p><strong>Цена:</strong> <?= Html::*encode*($texno->cena) ?></p>  
 </div>  
</div>

Для перехода на эту страничку в карточке товара я добавил кнопку подробнее которая переходит на представление view проверяя для начала id этого товара в базе данных

### JavaScript (JQuery)