*Федеральное государственное бюджетное образовательное учреждение высшего образования*

*«Владимирский государственный университет*

*имени Александра Григорьевича и Николая Григорьевича Столетовых»*

*Кафедра информационных систем и программной инженерии*

***КУРСОВОЙ ПРОЕКТ***

***по дисциплине «Технологии программирования»***

*Разработка программной системы для*

*автоматизации работы магазина компьютерной техники*

*Выполнил: студент гр. ПРИ-120*

*Парахин К.В.*

*Принял: доц. Вершинин В.В.*

*Владимир, 2022*

Оглавление

[Этап 1 3](#_Toc33137296)

## Этап 1

**Цель работы**: разработать автоматизированную систему работы электронного компьютерного магазина, которая включает в себя подсистему хранения данных, которая будет позволять сохранять и фиксировать изменения данных о заказах, отгрузках и доставках заказов устройств компьютерной техники, о регистрации и действиях пользователей, имеющих различные привилегии в системе.

Для достижения поставленной цели необходимо решить следующие задачи:

* хранение данных о производителях компьютерной техники
* хранение списка категорий (типов устройств)
* хранение списка устройств, имеющихся в наличии на складах
* хранение данных о пользователях, авторизованных или зарегистрированных в системе
* хранение данных о системных пользователях (работников системы), их привилегий и функций
* хранение корзины пользователей, их отзывови оценок
* функции сбора итоговой информации о продажах компьютерной техники в иечение определнного промежутка времени (анализ маркетинга)

**Описание предметной области**

У каждого компьютерного устройства есть принадлежность к какому-либу типу устройств, свои технические характеристики, производитель, его цена и наличие на складах. Сеть официально получает разрешение на ведение торговой деятельности, покупает разнообразную компьютерную технику (например, ноутбуки, смартфоны, наушники, системные блоки, затем устанавливает свои цены на приобретенные товары для потенциальных покупателей в зависимости от различных факторов. Кроме того сеть может менять цену на устройства, делая скидки постоянным клиентам или в целях ускоренной продажи и продвижения определенных товаров (ведение маркетинга – чем занимается менеджер).

Сеть имеет свою собственную автоматизированную программную систему, представленную для пользователей системы в виде электоронного сайта, на котором пользователи системы могут совершать определенные действия, которые заложены в их функциональных ролях в этой системе.

Например, пользователи электронного сайта имеют возможность просто авторизоваться в системе, зарегистрироваться в личном кабинете, стать постоянным пользователем услуг сети.

Также стоит сказать, что сеть открывает в городе несколько пунктов выдачи, где операторы (работники системы) могут выдавать заказы покупателям (заказ должен быть доставлен в пункт выдачи – что не является частью автоматизированной электронной системы).

Ниже перечислим основные типы пользователей данной системы.

**Пользователи разрабатываемой подсистемы**

*Администратор системы –* сотрудник сети, который может редактировать содержание сайта, личного кабинета клиентов (то есть работает с блоками и сообщениями на сайте), а также осуществлять упраление аккаунтами пользователей системы

*Менеджер системы –* сотрудник сети, который осуществляет бухгалтерский (экономический) учет работы сети, может регулировать поставки тех или иных товаров на склады, а также регулировать категории товаров на сайте (тем самым осуществляя маркетинг)

*Оператор системы –* сотрудник сети, который осуществляет выдачу заказов покупателям, работает в пункте выдачи сети, обладает собственным функционалом в системе.

*Обычный пользователь системы (или покупатель) –* любой пользователь, осуществляющий доступ к системе, может просматривать электронный каталог, делать заказы и получать их, а также оставлять отзывы.

**Словарь предметной области:**

*Пункт выдачи* – отдельный торговый офис сети, содержащий некоторый кадровый состав и каталог имеющихся в наличии телефонов (часто вместо этого термина используется устоявшийся вариант – магазин).

Атрибуты: адрес, контакты, кадровый состав (операторы)

*Корзина* – страница, на которую клиент отправляет товары, чтобы затем приобрести их.

Атрибуты: клиент, товар, кол-во товаров (того или иного вида), дата добавления, информация о карте.

*Заказ –* это специльный бизнесс-процесс, в результате которого осуществляется поручение на доставку (и соответственно, оплату) товаров покупателю из электронного магазина.

Атрибуты: товары, покупатель, дата создания, стоимость заказа, дата доставки, адрес пункта выдачи

*Производитель* – компания, осуществляющая создание компьютерной техники, их продвижение и продажу в разных странах мира.

Атрибуты: название, описание, официальный сайт

*Тип устройства* (DeviceType)– один из основных шаблонов компьютерных устройств.

Атрибуты: название, описание

Артикул – некоторый идентификатор товара, по которому он определяется и находится в системной базе данных.

Атрибуты: товар, название, идентификатор

*Смартфон* – устройство, объединяющее в себе функции персонального органайзера и мобильного телефона

*Систе́мный блок* — это устройство, которое физически представляет собой корпус, наполненный аппаратным обеспечением для создания компьютера

*Ноутбук* — переносной компьютер, обладающий небольшими размерами и автономнойстью работы, в корпусе которого объединены типичные компоненты ПК.

*Модель устройства* - определенный вид (шаблон) устройств конкретного производителя, обладающий постоянным набором технических характеристик и стоимостью

Атрибуты: производитель, линейка, технические характеристики, базовая стоимость

*Каталог* – полный список предоставляемого ассортимента устройств, содержащий информацию о наличии различных компьютерных устройств (вообще в сети, и конкретно по наличии на складах).

Атрибуты: сеть, склаж, модель устройства, отметка о наличии

*Личный кабинет* – сервис на электронном сайте сети, в которой могут регистрироваться клиенты для удобного доступа к различным функциям системы с разных устройств и бразеров.

Атрибуты: электронный адрес, входные данные, пользователь системы, логин.

*Склад* – помещение, предназначенное для хранения товаров.

Атрибуты: адрес, товар, наличие товара(ов), их количество.

*Пользовтель (клиент)* – человек, который пользуется функционалом, предоставляемыми автоматизированной электронной системой.

Атрибуты: фамилия, имя, отчество, контактный телефон, личный кабинет, скидки и бонусы

*Роль –* это набор полномочий (или привилегий), который необходим пользователю или группе пользователей для выполнения определённых рабочих задач

Атрибуты: пользователь, привилегии, функции (функционал).

*Привилегия* – это право некоторой учетной записи выполнять определенную функцию.

Атрибуты: учетная запись, пользователь, логин, роль, функции.

*Скидка* – вознаграждение сетью постоянных клиентов за частое пользование услугами торговой сети, проявляющееся в снижение стоимости некоторых товаров при пользовании картой.

Атрибуты: постоянный клиент, скидка, бонус.

**Разработка UML-диаграмм**

Начнем с разработки **диаграммы типа Use Case (диаграммы прецедентов)** в нотации UML. Для этого требуется выполнить анализ прецедентов работы с программной системой и промоделировать взаимоотношения автоматизируемой информационной системы и ее элементов с внешней средой.

**В ходе проектирования были выделены следующие прецеденты:**

- Управлять своим аккаунтом (пользователь системы)

- Авторизоваться

- Зарегистироваться

- Управлять своим аккаунтом

- Выбирать устройство в каталоге (покупатель)

- Использовать поиск товара на сайте

- Применять фильтрацию товаров по категориям

- Создать заказ (покупатель)

- Добавить товар в корзину

- Управлять содержимым корзины

- Оплатить заказ

- Оценить товар из заказа

- Оставить отзыв о заказе

- Управлять товарами (менеджер)

- Огранизовать поставки товаров со складов

- Продвинуть товар (добавить рекламу)

- Администрировать сайт (администратор)

- Добавить новый блок (сообщение) на сайт

- Осуществлять модерацию отзывов

**Опишем основные прецеденты пользователей в системе:**

**Прецедент «Управлять своим аккаунтом»**

Предусловие: пользователь системы авторизован в личном кабинете системы, имеет некоторый идентификатор в сети.

Действующее лицо: Пользователь системы

Основной поток: Управлять своим аккаунтом (авторизоваться – получить логин)

Клиент открывает в браузере электронный сайт магазина, заходит на страницу авторизации (в верхней правой части экрана присутствует ссылка)

Если пользователь не авторизован в системе, то система предложит ему при заходе на страницу авторизации авторизоваться.

Альтернативный поток: пользователь авторизован в системе. Тогда он может зарегистрироваться в личном кабинете (чтобы ему были доступны функции личного кабинета, а также накопление денег на балансе и возможности сохранения доставочной информации).

Если пользователь не собирается регистрироваться в системе – он может просто выйти с этой страницы и пользоваться простым функционалом системы.

**Прецеденты покупателей:**

**Прецедент «Выбирать устройство в каталоге»**

Предусловие: пользователь (клиент) авторизован в личном кабинете системы, имеет некоторый идентификатор в сети.

Действующее лицо: Покупатель

Основной поток: Выбирать устройство в каталоге

Клиент открывает в браузере электронный сайт (представление веб-приложения), на котором отображается артикул всех моделей, доступных для покупки.

Пользуется средствами поиска по каталогу и операциями фильтрации (может вводить название производителя, отмечать требуемые значения характеристик искомых устройств и т.д.)

Отмечает найденную интересуемую модель устройства и нажимает кнопку «Добавить в корзину» - прецедент «Добавить в корзину».

Система при этом сохраняет в базу данных выбранное устройство и при надобности перенаправляет клиента в «корзину», в которой находится список всех выбранных пользователем устройств на сайте.

Альтернативный поток: нужная модель отсутствует в каталоге (не в наличии)

На втором шаге клиент не находит требующееся устройство (например, какой-то смартфон). В этом случае он:

- либо начинает поиск устройства другой модели

- либо корректирует условия поиска или фильтрации исходного каталога

Постусловие: если требующееся устройство было все-таки найдено, то должно быть произведено создание заказа, в который будет включено это устройство (и остальные устройства из списка выбранных).

В случае, если заказ не укомплектован и не выполнен, то не гарантируется удержание данного устройства в наличии (то есть другие пользователи могут в это время заказать его сами и оно уже не будет в будущем в наличии).

**Прецедент «Создать заказ».**

Предусловие: пользователь (клиент) выбрал устройство (а), добавил их в корзину.

Действующее лицо: Покупатель

Основной поток: Создать заказ

Клиент открывает в браузере электронный сайт (представление веб-приложения), заходит на страницу под названием «Корзина», в ней выбирает те устройства, которые он в данный момент хочет купить, помечает их галочками. Затем нажимает кнопку «Составить заказ», переходит на форму оплаты заказа, осуществляет все необходимые действия и оставялет свою контакную информацию. Затем ожидает некоторое время доставку своего заказа и получает заказ: прецедент «Получить заказ».

Система при этом сохраняет в базу данных выбранный заказ, укомплектовывает его и отправляет его в выбранный пользователем пункт выдачи.

Альтернативный поток: устройства, которые пользователь хочет купить, теперь отсуствуют на сайте, и он не может их добавить в заказ.

На втором шаге клиент не находит требующееся устройство в статусе «В наличии» среди списка выбранных устройств в корзине:

- либо выбирает другие устройства и совершает заказ

- либо корректирует условия поиска и заново ищет нужное устройство («Выбирать устройство»)

Постусловие: если требующееся устройство было все-таки найдено, то должно быть произведено создание заказа, в который будет включено это устройство (и остальные устройства из списка выбранных).

Кроме того, в данном взаимодействии еще присуствуют такие прецеденты, как: «Искать устройство», «Передать заказ» (в пункт выдачи), «Получить заказ».

Кроме основного действующего лица «Клиент» в данной схеме присутствует еще «Оператор», который занимается передачей заказа клиенту.

**Прецеденты менеджера:**

**Прецедент «Управлять категориями товаров».**

Предусловие: На сайте присутствует некоторый базовый каталог товаров

Действующее лицо: Менеджер

Основной поток: Управлять категориями товаров.

Менеджер имеет возможность управлять каталогом электронного сайта, регулировать категории товаров, сформированные ранее, настраивать фильтры товаров, гибко подстраиваясь под основные интересы клиентов (например, присулшиваясь к их отзывам). Кроме этого, он управляет не только каталогом (то есть не только способами отобразить имеющиеся товары в наличии), но и может управлять отдельными товарами, выводить более востребованные товары и заниматься модерацией прочих, подобно менеджеру обычного магазина.

Альтернативный поток: менеджер не может управлять категориями, потому что такой блок не был добавлен на сайт администратором. Тогда он должен обратиться с помощью к администратору.

Ниже изображена UML – диаграмма прецедентов для основных пользователей системы интернет-магазина компьютерной техники (рис.1 – измененная по сравнению с прошлым этапом диаграмма) :

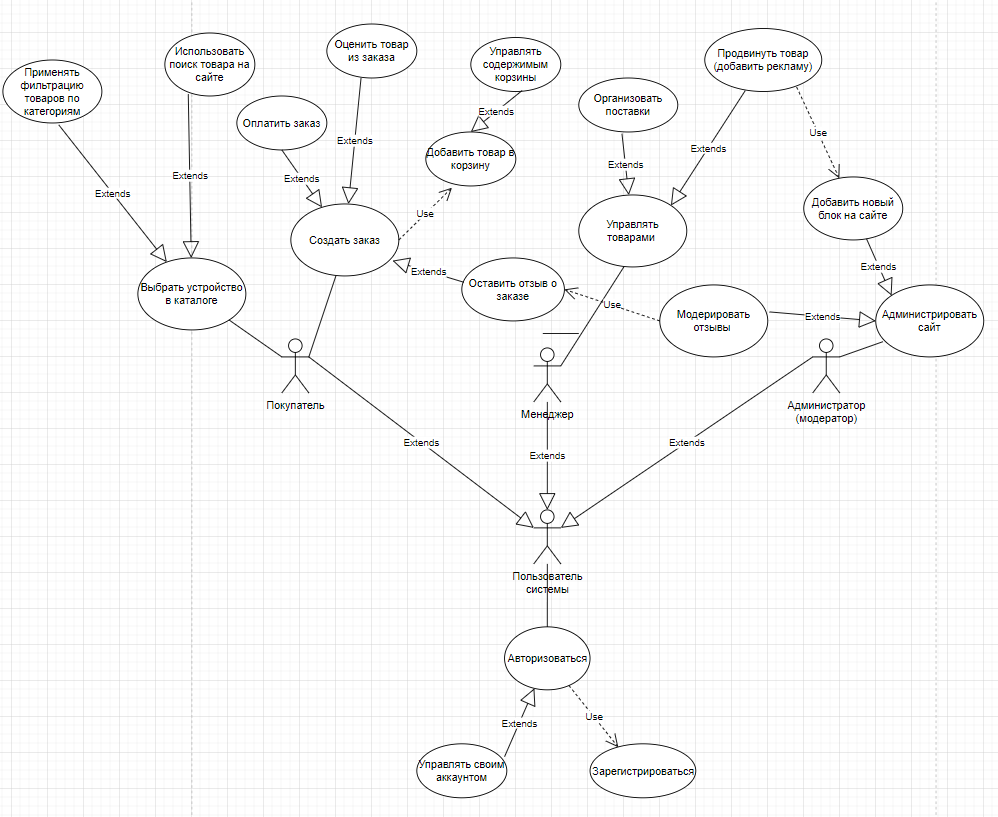


Рисунок 1. Диаграмма прецедентов

Далее разработаем **диаграмму классов (UML Class Diagram)** в нотации UML.

Класс – это описание совокупности объектов с общими атрибутами, операциями, отношениями и семантикой.

Выделим такие основные сущности в системе, как:

- **Producer** (производитель (и его требования))

- **DeviceType** (тип устройства: смартфон, ноутбук, системный блок, планшет, наушники и т.д.)

- **Product** (товар, являющийся некоторой моделью устройства)

- **SummUpProduct** (совокупность товаров одного артикула, которая может находиться в корзине)

- **Order** (заказ, его статус и информация о нем)

- **Basket** (корзина клиента с данными о товарах и личными клиентскими данными)

- **User** (абстрактный пользователь с его данными в системе)

- **Role** (описание привилегии, которой может быть наделен пользователь системы с набором опций. Включает в себя такие привилегии, как: покупатель (обычный пользователь), менеджер и администратор системы)

- **Option** (особая функция, которой может быть наделена некоторая привилегия. Считается также, что если какая-то более низкая привилегия обладает какой-то опцией, то все привилегии (роли) выше нее по уровню также обладают этой опцией).

- **Notification** (системное уведомление пользовтелю на сайте)

- **Rewiew** (оценки конкретного товара и отзывы о нем)

- **Warehouse** (склад для хранения товаров)

- **Shipping** (информация о доставке товаров)

Ниже изображена UML – диаграмма классов системы интернет – магаизна компьютерной техники, на которой используются такие основные виды отношений, как ассоциация, агрегация и композиция.

Примеры связей и отношений между сущностями (классами) можно посмотреть на рисунке 2 (измененная по сравнению с прошлым этапом диаграмма)

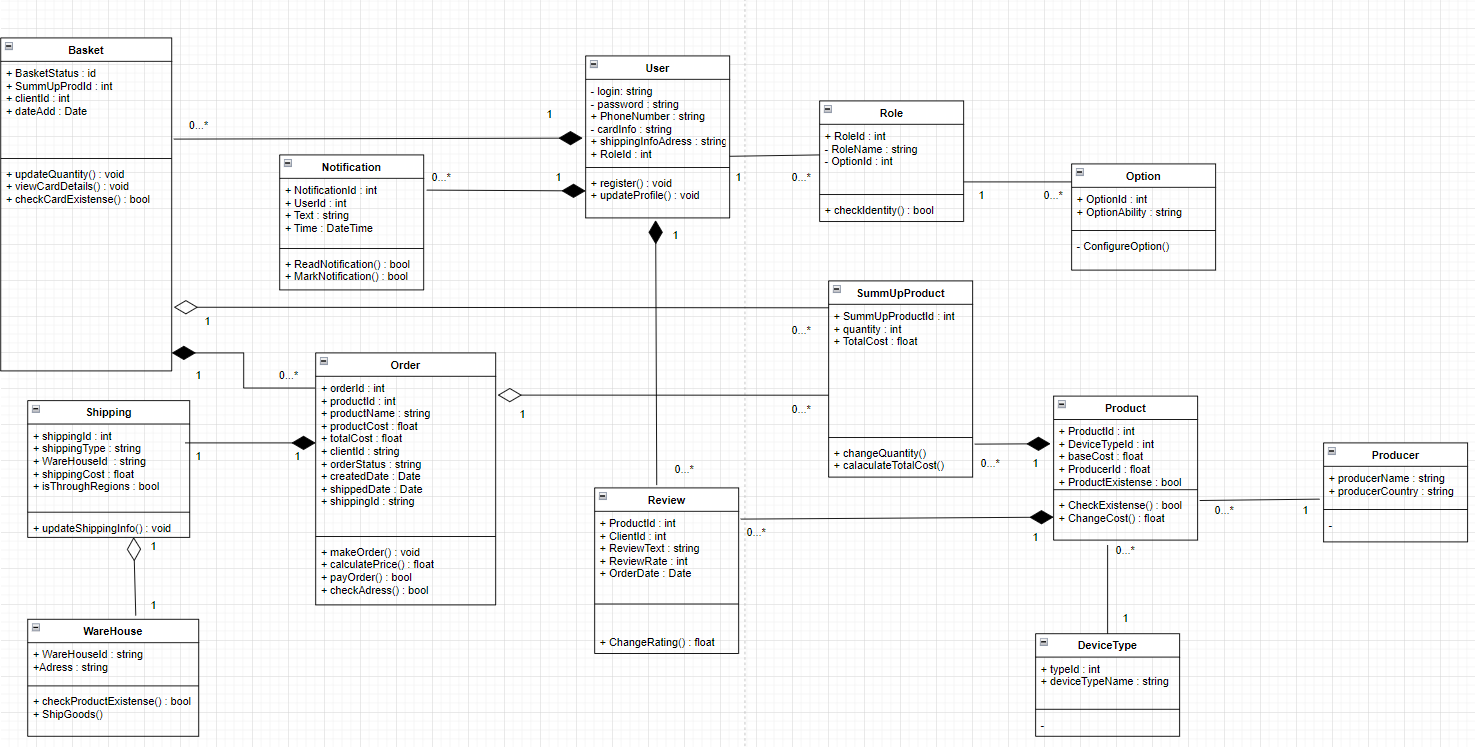


Рисунок 2. Диаграмма классов

***Этап 2***

**Динамическое моделирование системы**

Начнем с анализа состояний в автоматизируемой системе электронного компьютерного магазина для построения диаграммы состояний (**UML StateChart Diagram**):

Сделаем это на примере состояний объекта «Заказ» для данной системы (учитывая тот факт, что сущность «Заказ» построена на сущности «Корзина» (смотреть диаграмму классов – рис.2)

Состояния объекта «Заказ в системе»:

1. Покупатель собрал желаемые товары в корзине, указав их требуемое количество. Объект «Корзина» заполнен. Начинает создаваться объект «Заказ»
2. От системы создания заказа приходит форма для оформления заказа, в которой необходимо заполнить данные о способах оплаты, реквизитах пользователя, место для получения заказа.

А) Заказ не будет создан до того момента, пока не будут заполнены реквизиты и не будет выбрано доступное место получения заказа.

Б) Оплата заказа производится через внешнюю платежную систему, заказ будет обработан только после получения подтверждения о его оплате.

3) От системы проверки оформления заказа пришло сообщение (внутри системы) об отсутствии всех необходимых введенных данных.

Покупателю на сайте показывается соотвествующее уведомление, переходим к пункту 8

4) От платежной системы пришло сообщение об отказе в оплате или истек максимальный срок ожидания оплаты.

Заказ отменяется, пользователю приходит на электронную почту уведомление об отмене заказа. После этого переходим к пункту 8.

1. От платежной системы пришло подтверждение об оплате. Заказ помечается как «оплаченный».

Указанный покупателем адрес доставки проверяется во внешней системе проверки адресов, то есть заказ ожидает подтверждение о проверке адреса.

1. От системы проверки адресов пришло сообщение об ошибке в указанном для доставки адресе. Покупателю приходит уведомление на почту об отмене заказа и возврате средств, платежная система получает команду вернуть деньги за отмененный заказ на счет пользовтеля.
2. От системы проверки адресов пришло подтверждение правильности введенного адреса. Заказ помечается как «доставляемый» и переходит в состояние ожидания подтверждения о доставке.

Пользовтелю приходит уведомление на почту о передаче заказа в службу доставки.

1. Заказ помечается как «не созданный» и «ждёт» дооформления.
2. Заказ помечается как «отмененный» (и «ждёт» переоформления)
3. Заказ приходит в пункт выдачи и помечается как «досталенный». Покупатель получает соответствующее уведомление о доставке товара.

Диаграмма состояний для объекта «Заказ» в системе(Statechart Daiagram UML):

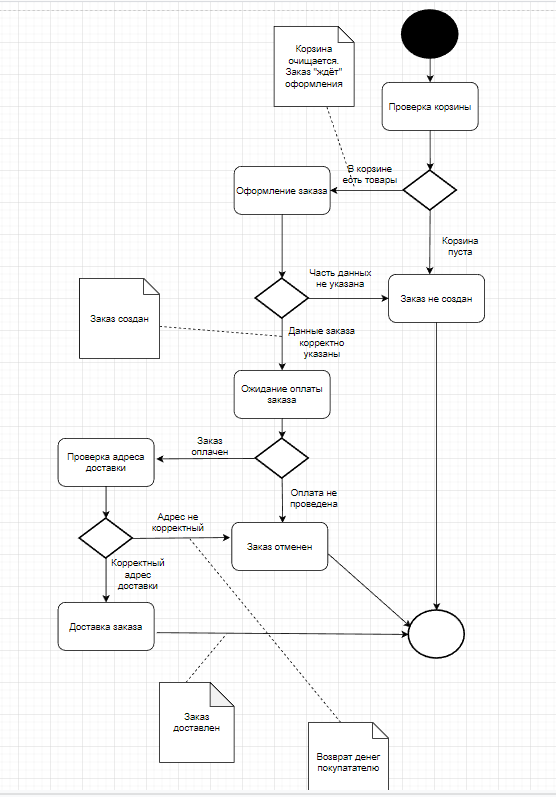


Рисунок 3. Пример диаграммы состояний системы для сущности «Заказ»

Далее проанализируем виды деятельности в системе и построим диаграмму последовательностей **(UML Sequence Diagram)**

Используем для это логику перехода между состояниями объекта «Заказ» из построенной и описанной ранее UML - диаграммы состояний.

Выделим такие основные последовательности (последовательные «остановки» для создаваемого заказа в системе, вызываемые функции и сопровождаемые ими действия, а также ответные реакции системы (тоже в виде вызова функций – что представляет концептуальную схему взаимодействия сущностей в системе с их контроллерами).

Последовательные «звенья» системы оформления заказа:

- Покупатель

Функции:

- Register (Зарегистрироваться) : ответное действие – уведомление о регистрации

- Login (Авторизоваться) : ответное действие – уведомление о результате проведенной авторизации

- AddProductToBasket (Добавить товар в корзину) : ответное действие – ShowProductPrompt (сообщение о изменении количества товаров данного артикула в корзине)

- ChangeProductsQuantity (Изменить количество товаров в корзине) : ответное действие – ChangeBasketConditionList (Изменить список товаров в корзине)

- OpenBasket (Открыть страницу «Корзина)»: ответное действие – DisplayBasketList (показать текущее состояние корзины на новой странице электронного сайта магазина)

- ClearBasket (Очистить корзину) : ответное действие – GetNewCleanedBasketStatus (уведомление об успешной очистке корзины)

- CreateOrder (Создать заказ) : ответное действие – ShowOrderInfo (показать статус созданного заказа и перейти к следующим шагам подтверждения оформления заказа (заказ может иметь на этом этапе состояния «Создан» или «Не создан» (по причине указания неполноты данных или их некорректности))

- PayOrder (Оплатить заказ) : ответное действие – GetPaymentNotification (показать состояние оплаты заказа (заказ на данном этапе может иметь состояния «Оплачен» или «Не оплачен»))

Используемые контроллеры:

- Profile Controller (требуется для регистрации пользователей в системе, проверке данных, вводимых пользователем при его авторизации в системе)

Ответные действия :

* RegistrationNotification (уведомление о регистрации)
* Login\Logout Notifications (уведомление о проведенной авторизации)

- BasketManager

Ответные действия :

– ShowProductPrompt() : Product (сообщение о изменении количества товаров данного артикула в корзине)

- ChangeBasketConditionList() : Basket (Изменить список товаров в корзине)

- DisplayBasketList() : Basket (Показать текущее состояние корзины на новой странице «Корзина»)

- GetNewCleanedBasketStatus() : Basket (Уведомление об успешной очистке страницы «Корзина»)

- OrderManager

Функции:

- GetPaymentOfOrder() : Order (Проверить состояние оплаты счета по оформленному заказу)

Ответное действие системы – ChangeStatusPaymentOrder() – уведомление о текущем статусе оплаты счета.

- CheckAdress() : Order (Проверить корректность указанного в заказе адреса и его существования)

Ответное действие системы – GetAdressExistenseStatus() – уведомление о результате проверки указанного адреса

Используемые контроллеры:

- Payment System (система оплаты счетов по создаваемым в интернет-магазине заказам)

- CheckingAdressSystem (система проверки адресов на карте)

Диаграмма последовательностей (Sequence UML Diagram):

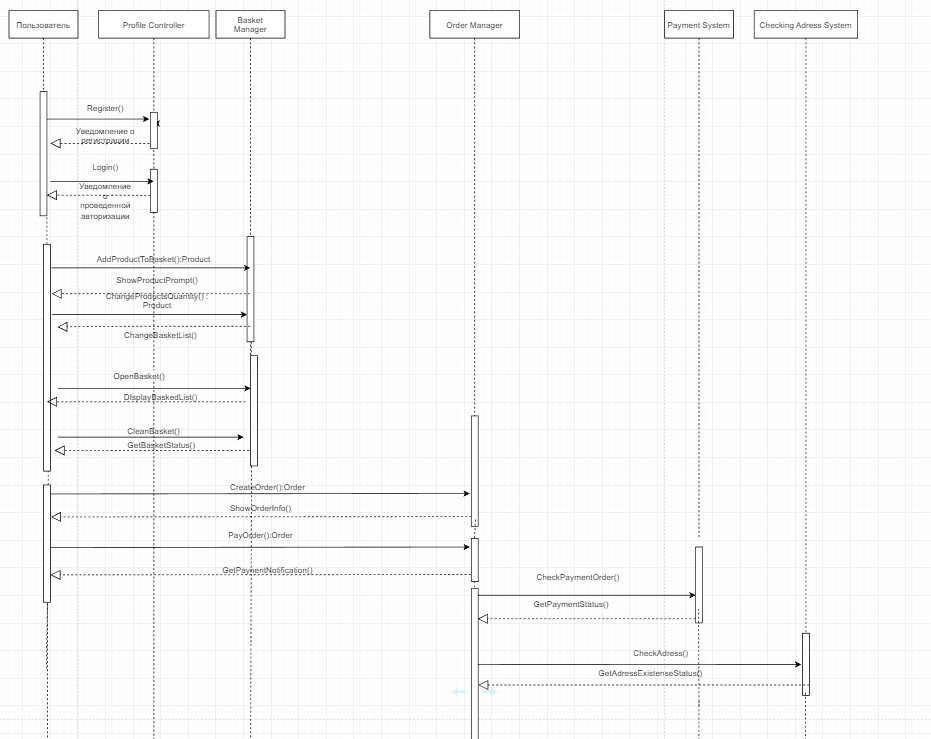


Рисунок 4. Диаграмма последовательностей в системе

Анализ видов деятельности и построение диаграммы видов деятельности (**UML Activity Diagram**):

Основные действующие лица в системе: клиент (покупатель), менеджер, администратор.

Кроме этого, основные действия по оформлению заказа, составлении платежного счета и проверке его оплаты, по проверке корректности указанного адреса, будет производить автоматизированная система интернет-магазина компьютерной техники.

Менеджер в случае отсутствия каких-то товаров из заказа в наличии на внутренних складах осуществляет по своему усмотрению поставку товаров со внешних складов (складов поставщика). В случае, если заказ пришел, а клиент его не получил, то менеджер делает обратный запрос на возвращение товаров (на внутренние склады).

Администратор занмается проверкой выдачи заказов, ставит подтверждение при явке покупателя и получении им заказа, либо в случае неявки он отзывает товары из заказа.

(После получения заказа и его закрытия в системе, покупатель может оставить о нем отзыв на электронном сайте и поставить оценку товарам, администратор также может модерировать эти отзывы)

Ниже будут представлена диаграмма активностей работы интернет-магазина компьютерной техники (рис. 5.1 и рис. 5.2):

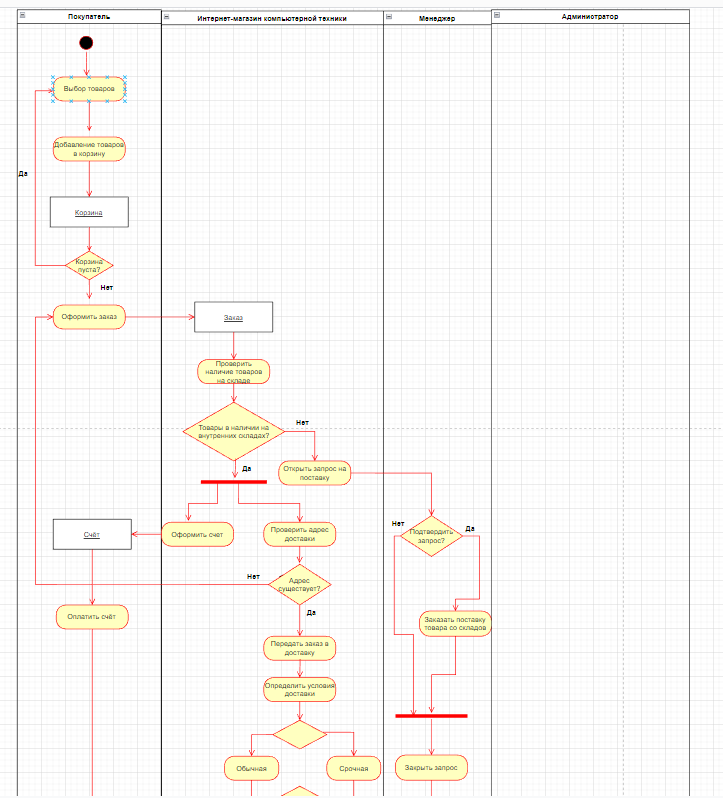


Рисунок 5.1. Верхний фрагмент диаграммы активностей

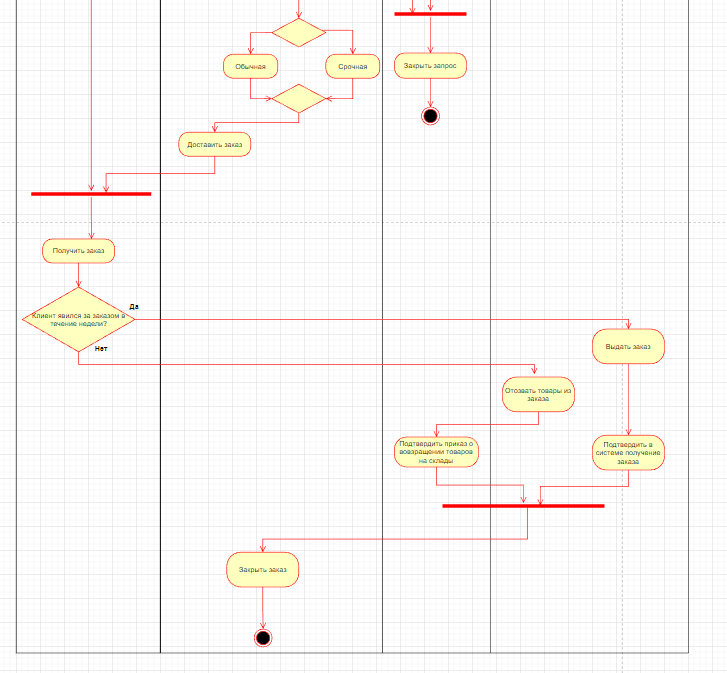


Рисунок 5.2. Нижний фрагмент диаграммы активностей

4Д