|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |   Институт кибернетики | |
| Кафедра программного обеспечения систем радиоэлектронной аппаратуры | |
|  |
|  |
|  |
| **КУРСОВАЯ РАБОТА** | |

по дисциплине \_\_\_\_\_\_«Системный анализ программного обеспечения» \_\_\_\_\_   
 (наименование дисциплины)

**Тема курсовой работы:** Разработка ПО «Клиент магазина видеоигр»

**Студент группы** \_КМБО-02-18, Новожилов Дмитрий Владимирович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 (учебная группа, фамилия, имя, отчество студента) (подпись студента)

**Руководитель курсовой работы** Свинцова А.А. \_\_\_\_\_\_\_\_\_\_\_\_\_ **(**Должность, звание, ученая степень) (подпись руководителя)

**Консультант** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 (Должность, звание, ученая степень) (подпись руководителя)

Работа представлена к защите «\_\_\_\_» \_\_\_\_\_\_\_\_\_20\_\_г.

Допущены к защите «\_\_\_\_» \_\_\_\_\_\_\_\_\_20\_\_г.

Москва 2020

Оглавление

[1. Постановка задачи 4](#_Toc57988726)

[1.1 Исследование заинтересованных лиц 4](#_Toc57988727)

[1.1.1 Покупатели 4](#_Toc57988728)

[1.1.2 Разработчики 5](#_Toc57988729)

[1.1.3 Издатели 5](#_Toc57988730)

[1.1.4 Возможные проблемы 6](#_Toc57988731)

[1.2 Анализ конкурентов 6](#_Toc57988732)

[1.3 Диаграмма классов 9](#_Toc57988733)

[1.3.1 Описание диаграммы классов 10](#_Toc57988734)

[2. Концепция 11](#_Toc57988735)

[2.1 Общее описание 11](#_Toc57988736)

[2.2 Масштаб и границы системы 11](#_Toc57988737)

[2.3 Диаграмма прецедентов 12](#_Toc57988738)

[2.4 Функциональные возможности 17](#_Toc57988739)

[2.4.1 Выбор метода сортировки 18](#_Toc57988740)

[2.4.2 Поиск по названию 19](#_Toc57988741)

[2.4.3 Открытие страницы игры 19](#_Toc57988742)

[2.4.4 Добавление игры в желаемое 19](#_Toc57988743)

[2.4.5 Удаление игры из желаемого 20](#_Toc57988744)

[2.4.6 Заполнение корзины 20](#_Toc57988745)

[2.4.7 Удаление из корзины 20](#_Toc57988746)

[2.4.8 Выбор способа оплаты 21](#_Toc57988747)

[2.4.9 Оплата покупки 21](#_Toc57988748)

[2.4.10 Возврат товара 21](#_Toc57988749)

[2.4.11 Просмотр библиотеки 22](#_Toc57988750)

[2.4.12 Запуск игр 22](#_Toc57988751)

[2.4.13 Сохранение прогресса в облаке 22](#_Toc57988752)

[2.4.14 Предоставление доступа к своей библиотеке для друзей 23](#_Toc57988753)

[2.4.15 Получение достижений 23](#_Toc57988754)

[2.4.16 Создание обсуждений 23](#_Toc57988755)

[2.4.17 Комментирование на страницах пользователей 24](#_Toc57988756)

[2.4.18 Комментирование в обсуждениях 24](#_Toc57988757)

[2.4.19 Переписка с другими пользователями 24](#_Toc57988758)

[2.4.20 Добавление пользователей в друзья 25](#_Toc57988759)

[2.4.21 Удаление пользователей из друзей 25](#_Toc57988760)

[2.4.22 Добавление пользователей в чёрный список 25](#_Toc57988761)

[2.4.23 Удаление пользователей из чёрного списка 26](#_Toc57988762)

[2.4.24 Загрузка продукта на сервера 26](#_Toc57988763)

[2.4.25 Управление ценой продукта 26](#_Toc57988764)

[2.4.26 Загрузка обновлений продукта 27](#_Toc57988765)

[2.4.27 Поиск по именам 27](#_Toc57988766)

[2.4.28 Поиск по жанрам 27](#_Toc57988767)

[3. Требования 28](#_Toc57988768)

[3.1 Функциональные требования 28](#_Toc57988769)

[3.1.1 Просмотр каталога игр 28](#_Toc57988770)

[3.1.2 Покупка игр 28](#_Toc57988771)

[3.1.3 Прохождение игр 29](#_Toc57988772)

[3.1.4 Общение с другими пользователями 29](#_Toc57988773)

[3.1.5 Продажа видеоигр и программ 30](#_Toc57988774)

[3.1.6 Просмотр каталога разработчиков 30](#_Toc57988775)

[3.2 Нефункциональные требования 31](#_Toc57988776)

[3.2.1 Требование к возможности и простоте локализации 31](#_Toc57988777)

[3.2.2 Требование к переносимости 31](#_Toc57988778)

[3.2.3 Требование к производимости 31](#_Toc57988779)

[3.2.4 Требования к безопасности 31](#_Toc57988780)

[3.2.5 Требование к надёжности 32](#_Toc57988781)

[3.2.6 Требование к времени хранения данных 32](#_Toc57988782)

[3.3 Ограничения проектирования 32](#_Toc57988783)

# 1 Постановка задачи

## 1.1 Исследование заинтересованных лиц

Создавая клиент магазина видеоигр, необходимо опираться на 3 основополагающие группы пользователей:

· Покупатели;

· Разработчики;

· Издатели.

Каждая группа имеет разные интересы и видит в нашем ПО различные цели. Мы в свою очередь должны суметь подстроиться под каждую из них, при этом не ограничив ничьи права и возможности. Итак, рассмотрим возможные интересы.

### 1.1.1 Покупатели

Главная цель покупателя – совершить покупку, получив при этом пользу и удовольствие. В сферу их интересов входит в первую очередь приобретение игр для проведения досуга. Каждый покупатель, а по совместительству игрок, имеет разные навыки и предпочтения в играх, поэтому им должна быть предложена возможность настраивать рекомендации, исходя из различных деталей. Приятным бонусом будет возможность общения с остальными пользователями площадки, получение новостей о новинках и грядущих играх, возможность влиять на рейтинг товаров. Невозможно в наше время без механизма возврата средств, ведь ожидания не всегда совпадают с реальностью или же игра может оказаться банально слишком современной для игровой системы пользователя. Также каждый человек хочет чувствовать себя уверенным, что в магазине проводятся работы по обнаружению и предотвращению мошенничества.

### 1.1.2 Разработчики

Главная цель разработчика – выпустить продукт в массы. Разработчики могут быть заинтересованы в быстром и бесплатном создании страниц внутри магазина, где будут выложены их игр. Уточним, что их задача – загрузить их проект на сервер, остальную работу выполнят сотрудники площадки. Разработчикам крайне важно получать отзывы от игроков и критиков, чтобы понимать, каким образом они могут улучшать контент в дальнейшем. Естественно, нельзя говорить о магазине, в котором был бы невозможен заработок денег и честное конкурирование. Сама же плата за то, чтобы получить возможность опубликовать игру на площадке должна быть приемлемой. Это нужно для того, чтобы огородить мошенников и халтурщиков от честных разработчиков, а потраченные средства будут выплачены сотрудникам магазина, занимающимся контролем качества.

### 1.1.3 Издатели

Издатели являются самым крупным игроком на игровых площадках. Их основной целью является привлечение новых разработчиков под своё крыло. Обычно они заинтересованы в увеличении узнаваемости своей торговой марки. Чем крупнее и надёжнее становится издатель, тем выше шанс, что игроки будут в дальнейшем покупать игры, выпущенные именно им. Для издателей важно иметь представление о текущих трендах внутри площадки, так они смогут искать независимых разработчиков, которые достигли определённого успеха и которые бы удовлетворили желанием целевой аудитории издателя. Кроме этого, в сферу их деятельности входит маркетинг уже существующих проектов собственного издания.

### 1.1.4 Возможные проблемы (их нужно для каждой группы лиц занести в пункты выше)

Основной проблемой может стать то, на какую из групп пользователей мы обратим большее внимание. Если отдать приоритет крупным издательствам, то они будут использовать все свои возможности для отстаивания собственных интересов и интересов партнёров в ущерб покупателям, и работающим на него разработчикам, страдающим от плохих условий работы. Конечно же, стоит стремиться к тому, чтобы предприниматели были честны и справедливы с клиентами. Но всему есть пределы и всегда приходится идти на компромиссы: например, крайняя степень «дружественности к покупателю» — бесплатная раздача товаров. Очевидно, что при отсутствии компромиссов магазин не сможет отстаивать собственные интересы и в конечном счёте обанкротится. Если же разработчикам удастся «перетянуть одеяло» на себя, то могут возникнуть ситуации, когда они смогут получать большую часть, а то и все деньги с продажи за продукты, к которым они могут не иметь никакого отношения, вследствие динамичного выбора процента с продаж и отсутствии контроля качества.

## 1.2 Анализ конкурентов

На данный момент существует уже достаточное количество онлайн-сервисов распространения компьютерных игр, поэтому нам неизбежно придётся столкнуться с конкуренцией. В нашем случае невозможно говорить об альтернативах, ведь магазин, не удовлетворяющий целям хотя бы одной из групп пользователей, магазином являться не может.

Итак, главным конкурентом является цифровая площадка Steam. Он в течение длительного времени не имел аналогов, поэтому смог достичь уровня некой монополии в данном сегменте рынка. Одним из его главных преимуществ является наличие торговой площадки. Торговая площадка – это внутренний сервис, который позволяет игрокам продавать предметы, полученные во время игры, и на вырученные средства совершать новые покупки. Таким образом, игроки получают мотивацию играть больше, чтобы получить из этого наибольшую выгоду. Всё это удачно дополняется социальными площадками платформы: системой достижений, форумами для каждой игры, мастерской, в которой игроки могут собственноручно создавать улучшения для любимых игр. В число минусов входит практически полное отсутствие модерации новых игр, что засоряет площадку некачественной продукцией, а также высокий процент для издателей, что отталкивает их от продажи в Steam.

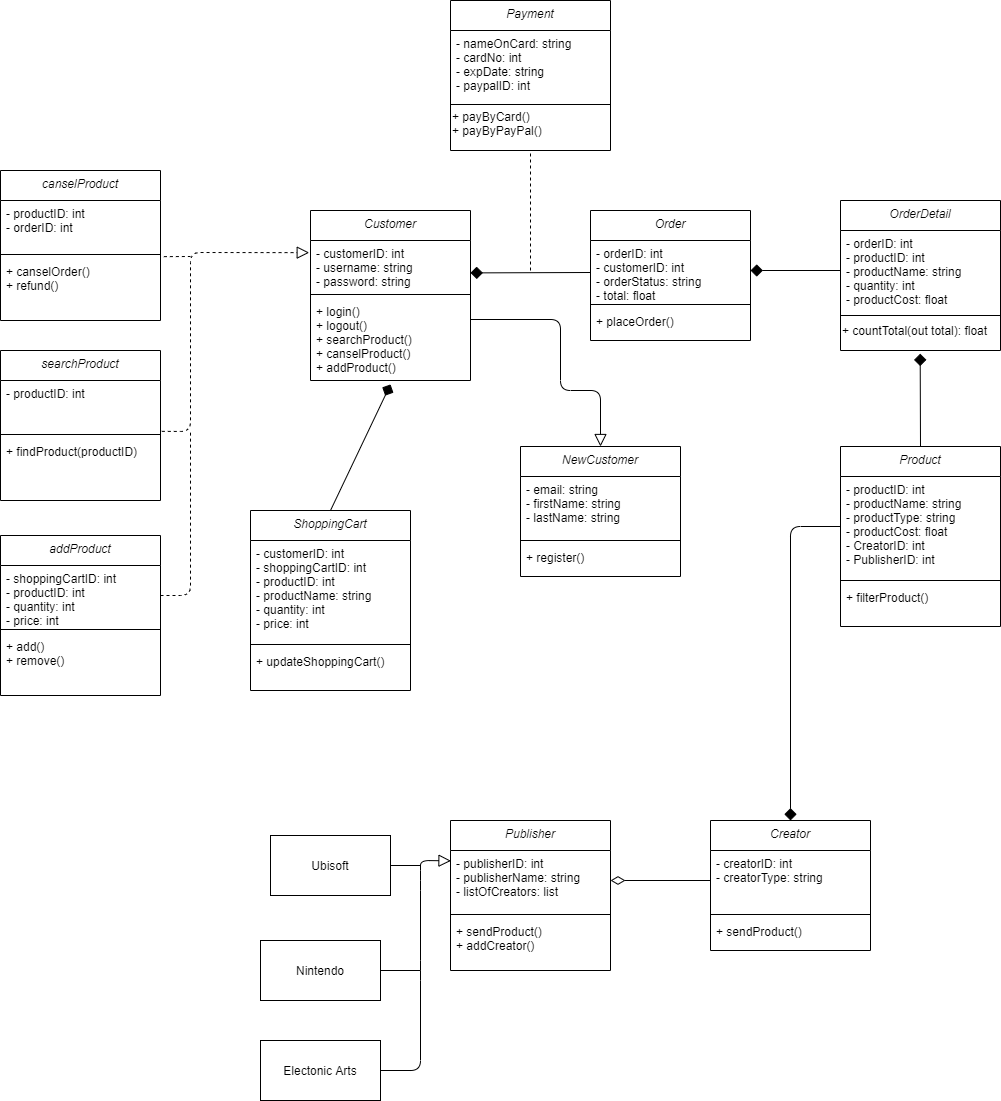
Вторым по популярности конкурентом является магазин Epic Games. Его успех обязан максимальной дружелюбностью к покупателям. Площадка известна своими низкими ценами, а также еженедельными бесплатными раздачами игр. Эти два фактора не могут не привлекать огромное количество игроков, особенно внутри категории лиц школьного возраста, не имеющих стабильного заработка для совершения сделок. Также магазин радует большим количеством эксклюзивов, которые нельзя приобрести ни на одной другой площадке. Из минусов выделяются невозможность вернуть игру и средства, если продукт не удовлетворил ожиданиям покупателя, а также крайне малая наполненность лаунчера функционалом. Единственное, с чем он справляется хорошо, так это с продажей игр. Социальное взаимодействие между пользователями находится на очень скудном уровне.

Итак, наш последний конкурент, магазин itch.io, который на первое место ставит разработчиков, и заходит в этом довольно далеко. Во-первых, itch обеспечивает бесплатный хостинг и магазин без курирования. Не дожидаясь разрешения, за несколько минут можно подготовить страницу магазина и люди смогут сразу же скачивать игру и даже платить за неё деньги. Более того, можно указать долю, которую вы хотите отдавать itch, можно даже выбрать 0%! А для опытных разработчиков itch предлагает самый лучший в своём классе инструмент командной строки под названием *[butler](https://github.com/itchio/butler)*. В нём используются современные алгоритмы патчинга различий между версиями, что делает загрузки удобными и быстрыми, а что важнее всего — их можно автоматизировать. Какие же недостатки есть у itch? Во-первых, там не так много покупателей. Это невероятно нишевый сайт, в своём текущем состоянии это в основном инструмент для самостоятельной публикации. Во-вторых, в магазине уже были случаи массового обмана покупателей, когда мошенники скачивали дистрибутивы игр и выкладывали их на itch по собственным ценам, выбрав при этом функцию оплаты “Direct to you” (транзакции направлялись непосредственно в руки преступников). То есть, на сайте абсолютно отсутствует слежка за публикуемыми материалами, что может вызвать только недовольство у обманутых покупателей.

Подводя итог всему вышесказанному, мы хотим в первую очередь создать клиент, в котором пользователи смогут не только покупать игры, но и вступать во взаимодействие друг с другом при помощи социального функционала, в котором будет соблюдён баланс между процентами с продаж для издателей и независимых разработчиков, а также нельзя обойтись без надлежащего контроля за качеством.

## 1.3 Диаграмма классов

Ниже вы сможете ознакомиться с диаграммой классов (рис. 1), которая поможет понять примерный механизм устройства нашего клиента.



(Рисунок 1 – Диаграмма классов)

### 1.3.1 Описание диаграммы классов

Одним из основных классов нашего ПО является класс *Customer*, который является наследником класса *NewCustomer.* Здесь новые пользователи вводят свои регистрационные данные, после чего получают уникальный ключ и способность авторизовываться и выходить из учётной записи. Без существования объекта класса Customer невозможно существование объектов классов *Order* и *ShoppingCart*, ведь все они привязаны к ключу *customerID*.

Класс Order получает данные из класса *OrderDetail*, в котором происходят подсчёты общей суммы заказа, после чего будет ожидаться оплата посредством вызова объекта класса *Payment*. В последнем покупатель может оплатить заказ с помощью карточки, как банковской, так и виртуальной (в нашем случае PayPal). В *OrderDetail* попадают данные из класса *Product*, который не может существовать, если нет ни одного объекта класса *Creator*. Класс *Product* содержит все данные о товарах, которые выставлены на продажу: идентификатор товара, название, тип товара (игра, дополнение или визуальные модификации), идентификаторы разработчика и издателя (последний может равняться нулю, если игра выпущена независимым разработчиком).

С классом Creator в свою очередь связан класс *Publisher*. Он же вполне может существовать отдельно, пополняя собственное содержимое списком из объектов класса Creator. Здесь хранятся данные об издателях и независимых разработчиках. Они могут отправлять продукты в магазин, издатели также имеют возможность пополнять своё содержимое независимыми разработчиками. В качестве примеров наследников класса *Publisher* указаны *Ubisoft*, *Nintendo* и *Electronic* *Arts*.

Каждый покупатель, осуществляя покупки, должен иметь буфер, в котором хранятся отмеченные им товары. Таким буфером (в дальнейшем корзина) является класс *ShoppingCart*. Она будет пополняться всякий раз, когда покупатель добавляет товар с помощью метода *addProduct*(), саму корзину можно обновлять. Также в классе *Customer* реализованы методы поиска товара внутри магазина и отправки запроса на отмену заказа/возврат средств, *searchProduct*() и *canselProduct*() соответственно.

# 2 Концепция

## 2.1 Общее описание

Создаваемое нами ПО в первую очередь является онлайн-сервисом цифрового распространения компьютерных игр и программ. После совершения покупки игру можно будет скачать и запустить из клиента магазина, по мере выхода обновлений они будут автоматически загружаться и устанавливаться. Помимо этого, пользователям предоставляются возможности для социального взаимодействия друг с другом, реализованные в виде небольшой социальной сети внутри клиента. Пользователи имеют доступ к следующим функциям: добавление друзей, общение, создание модификаций к купленным играм, получение достижений, редактирование профиля и др.

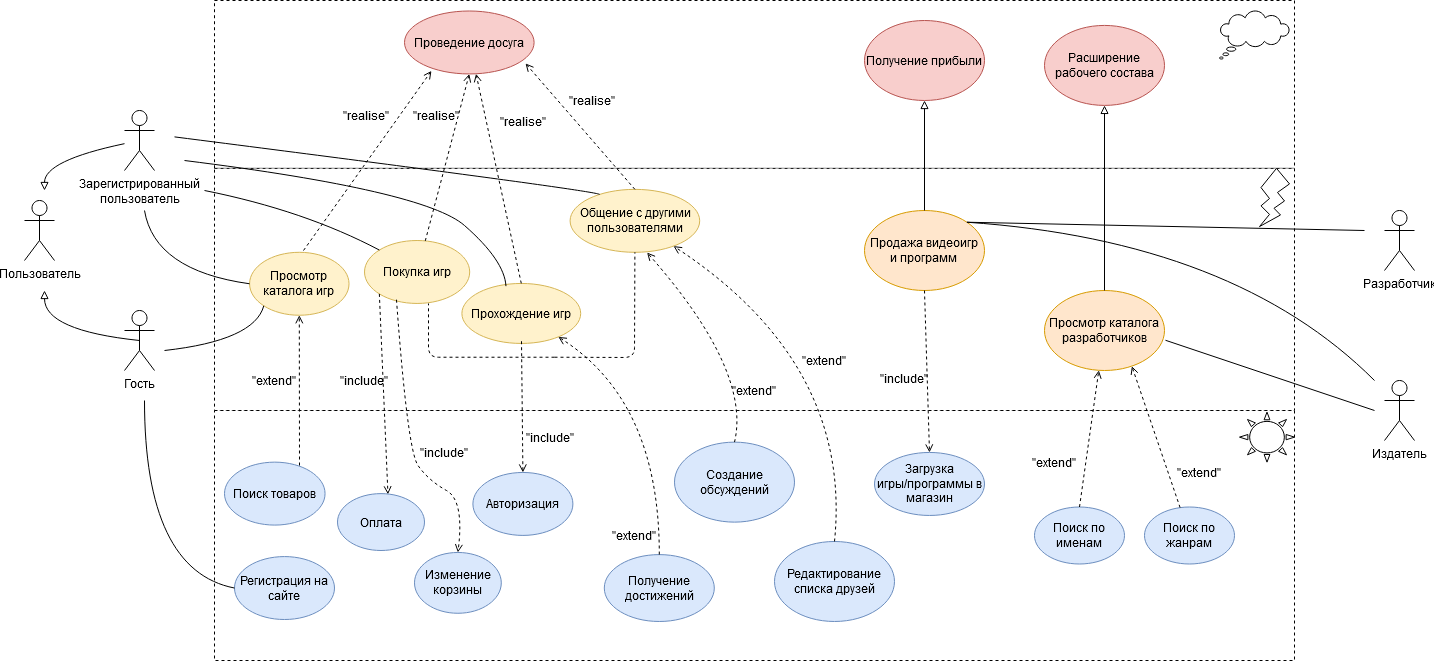
Для разработчиков и издателей также имеются функции, позволяющие выставлять на продажу свои товары и получать с них доход.

## 2.2 Масштаб и границы системы

Наше ПО не имеет ограничений по регионам распространения и может быть свободно скачано из Интернета. На первых этапах развития клиент планируется выпустить с локализацией на английском, русском, немецком, испанском и французском языках; позже планируется пополнить этот список языками азиатской группы и более мелкими европейскими языками. Пользователи нашего ПО будут ограничены только нормами международного права и законами их государства.

## 2.3 Диаграмма прецедентов

Ниже вы можете изучить диаграмму прецедентов (рис. 2), иллюстрирующую отношения между акторами (участниками системы) и прецедентами (вариантами использования).



(Рисунок 2 – Диаграмма прецедентов)

2.3.1 Описание диаграммы

Для актора *Пользователь* на глобальном уровне находится прецедент *Проведение досуга*. Он включает в себя все формы проведения свободного времени, на диаграмме таковыми являются:

* *Просмотр каталога игр.* Пользователи (зарегистрированные и нет) имеют возможность свободно перемещаться по каталогу магазина, посещать страницы игр, читать описание и отзывы других пользователей. Пользователей могут заинтересовать какие-то конкретные игры, поэтому существует возможность навигации внутри каталога, что входит в прецедент *Поиск товаров.*
* *Покупка игр.* Зарегистрированные пользователи помимо изучения каталога имеют возможность покупать игры, тем самым сохраняя их в своей библиотеке. *Изменение корзины* – для того, чтобы совершить покупку, товары сначала надо сохранить в корзине. *Оплата* – пользователи могут выбирать способ оплаты (банковская карта, электронный кошелёк, терминал).
* *Прохождение игр.* Каждую купленную игру можно использовать по её прямому назначению. *Авторизация* – пользователь должен быть авторизован в своём аккаунте, чтоб иметь доступ к своей библиотеке и получить доступ к облачным сохранениям. *Получение достижений* – по ходу игры пользователь будет получать внутриигровые достижения. Они могут выполнять как декоративную, так и соревновательную функции.
* *Общение с другими пользователями.* Не обязательно приобретать игры, чтобы получать удовольствия от использования нашего магазина. Пользователи могут общаться с другими людьми в групповых обсуждениях (*Создание обсуждений*), личных сообщениях; работать со списком друзей (*Редактирование списка друзей*).

Глобальной целью акторов *Разработчик* и *Издатель* является *Получение прибыли*. Данные группы пользователей получают прибыль за счёт проданного ПО (*Продажа видеоигр и программ*), созданного или издаваемого ими соответственно. *Загрузка игры/программы в магазин* – чтобы продажа была возможна, в первую очередь необходимо иметь возможность загружать товары на сервера нашего магазина.

У *Издателя* также есть также глобальная цель *Расширение рабочего состава*. Издательство растёт и становится узнаваемым за счёт игр, созданных курируемыми им разработчиками. Поэтому издатели должны иметь возможность просматривать всех активных на данный момент разработчиков (*Просмотр каталога разработчиков*). Для упрощения данной задачи они могут использовать *Поиск по жанрам* и *Поиск по именам*.

2.3.2 Описание прецедента «Покупка игр»

|  |  |  |
| --- | --- | --- |
| Название | Покупка игр | |
| Контекст | Желание расслабиться за компьютером | |
| Область действия | Игры существуют только в электронном формате, покупку невозможно соединить без выхода в сеть и наличия компьютера | |
| Уровень | Цели пользователя | |
| Основное действующее лицо | Зарегистрированный пользователь | |
| Участники и интересы | Участник | Интерес |
|  | Зарегистрированный пользователь | Обладание желаемой игрой в электронной библиотеке (быстро и надёжно) |
| Предусловие | У пользователя есть стабильное соединение с интернетом, он зарегистрирован и авторизован | |
| Минимальные гарантии | Уведомление о невозможности совершить покупку | |
| Гарантии успеха | Уведомление об успешной покупке игры | |
| Триггер | Покупка игр начинается после заполнения корзины товарами | |
| Описание | | |
| 1 | Выбираем игры из каталога | |
| 2 | Заполняем корзину | |
| 3 | Нажимаем кнопку «Оплатить» | |
| 4 | Выбираем способ оплаты | |
| 5 | Вводим платёжные данные | |
| 6 | Переход к пункту 1 | |
| Расширения | | |
| 1а | В каталоге нет интересных игр | |
| 1а1 | Завершение прецедента | |
| 2а | В корзину попали лишние товары | |
| 2а1 | Удаляем лишние товары из корзины | |
| 2а2 | Переход к пункту 3 | |
| 4а | Отсутствует желаемый способ оплаты | |
| 4а1 | Пишем в тех.поддержку | |
| 4а2 | Выбираем другой способ оплаты или завершаем прецедент | |
| 5а | Получаем сообщение об ошибке | |
| 5а1 | Проверяем введённые данные | |
| 5а2 | Повторяем пункт 5 | |
| 5b | Получаем повторное сообщение об ошибке (данные проверены и верны) | |
| 5b1 | Пишем в тех.поддержку | |
| 5b2 | Завершение прецедента | |
| 6а | Отсутствие желания покупать новые игры | |
| 6а1 | Завершение прецедента | |
| Изменения в технологии и данных | | |
|  |  | |
| Комментарии | | |
|  |  | |

2.3.3 Описание прецедента «Просмотр каталога игр»

|  |  |  |
| --- | --- | --- |
| Название | Просмотр каталога игр | |
| Контекст | Желание получить информацию о новинках и скидках | |
| Область действия | Можно ознакомиться только с кратким описанием игр, подробности можно узнать из интернета и других источников | |
| Уровень | Цели пользователя | |
| Основное действующее лицо | Пользователь | |
| Участники и интересы | Участник | Интерес |
|  | Пользователь | Формирование мнения на основе полученной информации |
| Предусловие | У пользователя есть стабильное соединение с интернетом | |
| Минимальные гарантии | Уведомление о недоступности каталога | |
| Гарантии успеха | Открытый и рабочий каталог | |
| Триггер | Каталог можно просматривать как только открыт клиент или сайт магазина | |
| Описание | | |
| 1 | Открываем главную страницу | |
| 2 | Смотрим раздел со скидками | |
| 3 | Смотрим раздел с новинками | |
| 4 | Смотрим раздел с рекомендациями | |
| 5 | Переход к пункту 1 | |
| Расширения | | |
| 1а | Хочется найти определённую игру | |
| 1а1 | Пользуемся поиском внутри каталога | |
| 1а2 | Переход к пункту 2 или завершение прецедента | |
| 2а | Хочется отсортировать скидки по убыванию/возрастанию | |
| 2а1 | Пользуемся кнопкой сортировки по убыванию/возрастанию | |
| 2а2 | Переход к пункту 3 или завершение прецедента | |
| 4а | Набор рекомендаций не устраивает | |
| 4а1 | Сортируем список, удаляя ненужные игры с помощью тэга «Неинтересно» | |
| 4а2 | Возврат к пункту 1 или завершение прецедента | |
| Изменения в технологии и данных | | |
|  |  | |
| Комментарии | | |
| 1 | В каждом разделе существуют и другие способы сортировки, а именно сортировка по названию, по дате выхода, по цене, по рейтингу. | |

## 2.4 Функциональные возможности

|  |  |
| --- | --- |
| Возможности | Функции |
| Просмотр каталога игр | Выбор метода сортировки (по цене, жанру, дате выхода, названию) |
| Поиск по названию |
| Открытие страницы игры |
| Добавление игры в желаемое |
| Удаление игры из желаемого |
| Покупка игр | Заполнение корзины |
| Удаление из корзины |
| Выбора способа оплаты |
| Оплата покупки |
| Возврат товара |
| Прохождение игр | Просмотр библиотеки |
| Запуск игр |
| Получение достижений |
| Сохранение прогресса в облаке |
| Предоставление доступа к своей библиотеке для друзей |
| Общение с другими пользователями | Создание обсуждений |
| Комментирование на страницах пользователей |
| Комментирование в обсуждениях |
| Переписка с другими пользователями |
| Добавление пользователей в друзья |
| Удаление пользователей из друзей |
| Добавление пользователей в чёрный список |
| Удаление пользователей из чёрного списка |
| Продажа видеоигр и программ | Загрузка продукта на сервера |
| Управление ценой продукта |
| Загрузка обновлений продукта |
| Просмотр каталога разработчиков | Поиск по именам |
| Поиск по жанрам |

### 2.4.1 Выбор метода сортировки

Статус: включённая

Приоритет: важная

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.1

Назначение: мы

Источник: учебный план

### 2.4.2 Поиск по названию

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.3 Открытие страницы игры

Статус: включённая

Приоритет: критический

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.4 Добавление игры в желаемое

Статус: включённая

Приоритет: полезная

Трудоёмкость: средняя

Риск: низкий

Стабильность: высокая

Целевая версия: 1.5

Назначение: мы

Источник: учебный план

### 2.4.5 Удаление игры из желаемого

Статус: включённая

Приоритет: полезная

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.5

Назначение: мы

Источник: учебный план

### 2.4.6 Заполнение корзины

Статус: включённая

Приоритет: критическая

Трудоёмкость: средняя

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.7 Удаление из корзины

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.8 Выбор способа оплаты

Статус: включённая

Приоритет: критическая

Трудоёмкость: средняя

Риск: средний

Стабильность: средняя

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.9 Оплата покупки

Статус: включённая

Приоритет: критическая

Трудоёмкость: высокая

Риск: высокий

Стабильность: средняя

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.10 Возврат товара

Статус: включённая

Приоритет: важная

Трудоёмкость: высокая

Риск: высокий

Стабильность: высокая

Целевая версия: 1.2

Назначение: мы

Источник: учебный план

### 2.4.11 Просмотр библиотеки

Статус: включённая

Приоритет: критическая

Трудоёмкость: средняя

Риск: низкий

Стабильность: средняя

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.12 Запуск игр

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.13 Сохранение прогресса в облаке

Статус: включённая

Приоритет: полезная

Трудоёмкость: высокая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.5+

Назначение: мы

Источник: учебный план

### 2.4.14 Предоставление доступа к своей библиотеке для друзей

Статус: включённая

Приоритет: полезная

Трудоёмкость: средняя

Риск: средний

Стабильность: средняя

Целевая версия: 1.5+

Назначение: мы

Источник: учебный план

### 2.4.15 Получение достижений

Статус: включённая

Приоритет: важная

Трудоёмкость: средняя

Риск: низкий

Стабильность: высокая

Целевая версия: 1.1

Назначение: мы

Источник: учебный план

### 2.4.16 Создание обсуждений

Статус: включённая

Приоритет: важная

Трудоёмкость: средняя

Риск: низкий

Стабильность: высокая

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.17 Комментирование на страницах пользователей

Статус: включённая

Приоритет: полезная

Трудоёмкость: средняя

Риск: средний

Стабильность: средняя

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.18 Комментирование в обсуждениях

Статус: включённая

Приоритет: важная

Трудоёмкость: низкая

Риск: низкий

Стабильность: средняя

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.19 Переписка с другими пользователями

Статус: включённая

Приоритет: критическая

Трудоёмкость: средняя

Риск: средний

Стабильность: средняя

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.20 Добавление пользователей в друзья

Статус: включённая

Приоритет: критическая

Трудоёмкость: средняя

Риск: низкий

Стабильность: высокая

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.21 Удаление пользователей из друзей

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: средний

Стабильность: высокая

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.22 Добавление пользователей в чёрный список

Статус: включённая

Приоритет: критическая

Трудоёмкость: средняя

Риск: средний

Стабильность: высокая

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.23 Удаление пользователей из чёрного списка

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.3

Назначение: мы

Источник: учебный план

### 2.4.24 Загрузка продукта на сервера

Статус: включённая

Приоритет: критическая

Трудоёмкость: высокая

Риск: средний

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.25 Управление ценой продукта

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.26 Загрузка обновлений продукта

Статус: включённая

Приоритет: критическая

Трудоёмкость: средняя

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.27 Поиск по именам

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

### 2.4.28 Поиск по жанрам

Статус: включённая

Приоритет: критическая

Трудоёмкость: низкая

Риск: низкий

Стабильность: высокая

Целевая версия: 1.0

Назначение: мы

Источник: учебный план

# 3 Требования

## 3.1 Функциональные требования

### 3.1.1 Просмотр каталога игр

Каждый товар хранится на серверах магазина с атрибутами *releaseDate* (дата выхода), *productName* (название игры/ПО), *productGenre* (жанр игры), *price* (цена). У пользователя должна быть возможность при помощи метода *sortBy()* отсортировать список товаров, выбрав желаемый атрибут. По умолчанию игры сортируются по релевантности (вычисляется при помощи алгоритмов на серверах). Каждый товар имеет собственную уникальную ссылку, которая ведёт к странице этого товара. На этой странице возможно добавить и удалить товар из списка желаемого (*addToDesired() removeFromDesired()*), после чего продукт будет направлен в список listDesired.

### 3.1.2 Покупка игр

Покупка игр осуществляется посредством работы со списком корзины *listCart*. Здесь хранятся все товары, которые вы планируете купить. Редактирование корзины происходит с помощью методов *addToCart()* и *removeFromCart()* (добавление и удаление товаров соответственно). Как только пользователь убедился, что корзина составлена правильно, он нажимает на кнопку оплатить покупку, после чего выбирает способ оплаты, по которому будет осуществляться покупка. В качестве примеров можно выбрать оплату картой – *payByCard()* или оплату с помощью сервиса PayPay – *payByPayPal().* После выбора способа оплаты магазин связывается с серверами платёжных сервисов и происходит покупка товара. Если покупатель остался не удовлетворён совершенной покупкой, он всегда может создать запрос на возвращение средств.

### 3.1.3 Прохождение игр

Пользователь может запустить игру, которая находится в его библиотеке, то есть куплена, с помощью метода *launchGame(),* который получает уникальный ID игры. Для каждой игры есть собственный список достижений *listAchievment*. Пользователь открывает достижения за различные действия внутри игры. Весь игровой процесс может сохраняться как на облачных серверах сервиса, так и на устройстве пользователя. Пользователи могут делиться идентификатором своей библиотеки *libraryID* с другими пользователями, чтобы те могли получить доступ к тем же играм. Одновременно к одной библиотеке могут быть подключены до 10 пользователей.

### 3.1.4 Общение с другими пользователями

Сервера нашего магазина также поддерживают коммуникацию с другими пользователями. Функция обсуждений позволяет нескольким пользователям участвовать в общении одновременно. Обсуждения создаются и удаляются с помощью методов *createDiscussion()* и *deleteDiscussion().* Первый метод создаёт обсуждение с уникальным *discussionID*, привязанный к *userID*, второй метод проверяет *userID*, а затем получает *discussionID* обсуждения, на которое направлено действие. Общение в обсуждениях и личной переписке с пользователями поддерживает все необходимые для этого функции. Пользователи могут писать сообщения *writeMessage(),* редактировать их *redactMessage(),* прикреплять файлы *attachToMessage().* Также каждый пользователь обладает уникальным списком друзей и заблокированных пользователей – *listFriends* и *listBlocked* соответственно. Работа с ними осуществляется с помощью методов *addToFriends(),* *removeFromFriends(),* *addToBlocked()* и *removeFromBlocked()*.

### 3.1.5 Продажа видеоигр и программ

Разработчики при использовании нашего сервиса в первую очередь должны иметь возможность загружать ПО на наши сервера. Разработчик заполняет форму, в которую передают основные атрибуты товара, которые после окончания загрузки будут отображаться в каталоге для всех пользователей. Как только продукт будет загружен, разработчик будет иметь возможность продолжать поддерживать его, вносить правки в атрибуты (например менять цену (*price*), описание (*productDescription*) и др.), загружать обновления или даже закрыть лот, скрыв его из каталога. При этом продукт продолжит храниться на серверах в течение месяца.

### 3.1.6 Просмотр каталога разработчиков

Издатели могут работать не только с каталогом игр, но и с каталогом разработчиков. Все разработчики хранятся в списке *listDevelopers*, у каждого есть своё уникальное имя и у их игр есть жанры, которые могут заинтересовать издателей. Сортировка списка осуществляется ранее названным методом *sortBy().*

## 3.2 Нефункциональные требования

### 3.2.1 Требование к возможности и простоте локализации

Система должна иметь файлы, отвечающие за выбор локализации. Всё, что предоставляется с нашей стороны, должно быть локализовано с самого начала, разработчики же должны иметь возможность вместе загружать файлы локализации для своего ПО вместе с этим самым ПО. В противном случае перевод всех страниц будет осуществлять интегрируемое оборудование. Список языков, локализация которых должна быть обеспечена в релизной версии: английский, русский, немецкий, испанский и французский языки.

### 3.2.2 Требование к переносимости

Базовые функции приложения должны быть написаны на таком уровне, чтобы перенос на другие платформы осуществлялся за наиболее короткий промежуток времени. При переносе на мобильные устройства должны остаться функции общения, просмотра каталога, библиотеки и покупки игр.

### 3.2.3 Требование к производимости

Сервера должны иметь возможность выдерживать колоссальную нагрузку в несколько миллионов пользователей (не меньше 10 миллионов) и не меньше %sampleNumber% транзакций одновременно. Скорость и пропускная способность каналов связи должна достигать как минимум %sampleNumber% Мбит/с.

### 3.2.4 Требования к безопасности

Так как наш сервис обеспечивает предоставление финансовых услуг и хранит в себе данные о наших пользователях, то необходимо наладить систему защиты этих данных. Во-первых, пользователь может подключить двухфакторную аутентификацию. В таком случае при подключении с нового или не сохранённого устройства наши сервера попросят подтвердить, что человек за этим устройством является владельцем аккаунта. Сделать это можно при помощи кода, который будет отправлен на мобильный телефон. Затем, сервисы должны обладать современной защитой от DDOS-атак.

### 3.2.5 Требование к надёжности

Перед проведением запланированных технических работ система должна предупредить всех авторизованных пользователей за час и за полчаса о том, что им лучше закрыть приложение, чтобы гарантированность сохранность их данных. В этот период новые пользователи не смогут попасть на сервера. Также система автоматически сохраняет все важные изменения, производимые внутри неё. Каждую ночь в системе очищается кэш действий и изменений, которые уже утратили силу. Все важные личные данные пользователей дублируются на резервных серверах.

### 3.2.6 Требование к времени хранения данных

Вся информация об активных пользователях и имеющая значимость должна бессрочно храниться на серверах. Выше уже было отмечено, что разработчики имеют возможности снять товар с продажи. В таком случае он будет храниться на серверах ещё месяц, после чего навсегда удалится.

## 3.3 Ограничения проектирования

Программа должна быть написана таким образом, чтобы для её работы был как минимум необходим ПК или ноутбук, имеющий доступ к сети и снабжённый ОС Windows 7 и старше, Linux или Mac OS. При установке программа должна выделить для себя как минимум 500 МБ свободного места.