

Министерство образования Республики Беларусь
Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Компьютерных технологий

Кафедра проектирования информационных компьютерных систем

Дисциплина "Современные технологии проектирования информационных систем"

К защите допустить:
Руководитель курсовой работы
старший преподаватель
кафедры
_____ А.В.Михалькевич
10.07.2025

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к курсовой работе
на тему

Электронная энциклопедия компьютерных игр

БГУИР КР 1-40 05 01-10 № 174 ПЗ

Студент _____ (подпись студента)

Курсовая работа
представлена на проверку
10.07.2025

(подпись студента)

Реферат

БГУИР КР 1-40 05 01-10 № 174 ПЗ, гр. 784371

, Электронная энциклопедия компьютерных игр, Минск: БГУИР - 2025.

Пояснительная записка 196533 с., 32 рис., 6 табл.

Ключевые слова:

Предмет Современные технологии проектирования информационных систем,
А.В.Михалькевич

-

-

Содержание

[Введение](#)

[1 Описание предметной области](#)

[1.1 С](#)

[2 Описание основных процессов предметной области](#)

[3 Спецификация вариантов использования системы](#)

[4 Информационная модель системы и её описание](#)

[5 Обоснование выбора компонентов и технологий для реализации курсового проекта](#)

[6 Модели представления системы и их описание](#)

[7 Описание применения паттернов проектирования](#)

[8 Руководство по развёртыванию системы](#)

[9 Результаты тестирования разработанной системы, и оценка выполнения задач](#)

[10 Приложение А. Листинг программы](#)

[Заключение](#)

[Список использованных источников](#)

[Приложения](#)

Введение

Курсовой проект посвящен изучению теории и практики разработки и проектирования распределённых баз данных. В данном курсовом проекте объектом исследования является среда WEB, как платформа приложений баз данных в информационных системах. Предметом исследования является система организации и ведения распределённых баз данных в Интернет. Целью курсового проекта является разработка электронной энциклопедии компьютерных игр. Курсовой проект предполагает выполнение следующих задач: 1) спроектировать базу данных электронной энциклопедии компьютерных игр, описать предметную область, создать диаграмму классов и диаграмму состояний; 2) разработать с использованием Вэб-интерфейса, алгоритма работы электронной энциклопедии компьютерных игр. В качестве практической части в рамках курсового проекта создаётся электронная энциклопедия компьютерных игр с использованием технологий языка программирования PHP и СУБД MySQL. Выбранная тема считается актуальной на сегодняшний день, так как сегодня миллионы людей ежедневно, не выходя из дому, просматривают электронные энциклопедии разного характера. В мире, а в частности в Беларуси, огромными темпами растёт количество пользователей Internet и, как следствие, количество «электронных» пользователей и потенциальных «электронных» пользователей.

1 Описание предметной области

Электронные энциклопедии – это электронные страницы с текстами и графическими изображениями классического энциклопедического характера. Главное преимущество такой энциклопедии перед обычными бумажными формами – мощная система поиска информации, основанная на специальных программных алгоритмах поиска, электронные каталоги, функция поиска связанной информации, поддержка видео и анимации, и, безусловно, небольшой физический объем. 50 томов книг-энциклопедий умещаются на одном CD-диске.

Компьютерные игры – это сложные программы, которые служат человеку для обучения или развлечения и рассчитаны на разные возрастные группы. Игры – это неперенные спутники развития человечества с момента его зарождения, когда древние люди в свободное время занимали себя различными предметами: камнями, палками, шкурками и другими. Каждый временной период характеризуется наличием у людей различных игрушек. С развитием технологий усовершенствовались и игрушки. И не секрет, что пик игрушечного производства пришелся именно на пик технологического прогресса середины XX века. В это же время вместе с изобретением компьютеров появились и компьютерные игры. Первый прототип компьютерной игры был создан в США, в Технологическом Университете штата Массачусетс. Эта игра получила название Space War и была рассчитана на двух игроков. Суть ее заключалась в том, что два космических корабля должны были сбить друг друга с помощью своих ракет. Однако в те времена игра не получила должного внимания и популярности из-за дороговизны персональных компьютеров и всего, что было с ними связано.

В этот же период были выпущены еще две игры:

«Теннис», разработчиком которой был Уильям Хигинботем;
«ОХО» (компьютерная версия игры «крестики-нолики»), разработчиком которой стал Санди Дуглас.

Спустя несколько десятилетий, в 1972 году, Нолан Башнелл выпустил всеми известную игру под названием «Pong», сутью которой было удержание мячика на игровом поле с помощью специальных тарелок. Это событие стало поистине одним из самых важных в истории развития компьютерных игр. Далее стали появляться первые игры в жанре бродилок, приключений, гонок и других. Еще через несколько лет была выпущена первая игровая приставка с одной единственной игрой, а затем и ее аналоги с несколькими играми. С тех пор персональные компьютеры и игровые приставки постоянно усовершенствовались, а игры для них становились функциональней, интересней и ярче. Появился целый рынок компьютерной индустрии, который по своим оборотам и вложениям обгонял даже рынок киноиндустрии.

Сейчас все компьютерные игры мы можем классифицировать по следующим тематикам:

приключенческие игры;
аркадные игры, которые характеризуются поэтапным прохождением игры и набором очков, где в финале необходимо выполнить определенную задачу;
игры-стратегии, где для достижения цели нужно уметь принимать правильные логические решения;
ролевые игры, где определена цель игры, а на пути главных героев к этой цели встречаются препятствия, которые необходимо преодолеть;

логические игры, которые представляют собой компьютерные версии различных головоломок.

1.1 С

2 Описание основных процессов предметной области

Объектом автоматизации является электронная энциклопедия компьютерных игр, которая отображает всю самую необходимую информацию о компьютерных играх. Основная задача работы электронной энциклопедии компьютерных игр - хранение, сбор, и в некотором случае призыв к продаже компьютерных игр клиентам.

Основными задачами электронной энциклопедии являются:

- оперативное размещение справочной информации на своих электронных страницах;
- ведение справочно-поискового аппарата: жанров, тематик и производителей компьютерных игр;
- улучшение работы электронной энциклопедии на основании внедрения современных технологий и компьютеризации информационных процессов.

Работа с пользователем организована следующим образом: у каждого пользователя, который использует электронную энциклопедию, имеется некий скидочный купон на просмотр дополнительной информации. Пользователь поясняет, какая именно компьютерная игра ему нужна, после чего идет проверка, имеется ли она в наличии. Если компьютерная игра имеется в наличии, то администратор оговаривает сроки, в которые компьютерная игра может подлежать ознакомлению в электронной энциклопедии.

Основные нотации концептуального моделирования: нотация IDEF0/IDEF3, нотация ARIS, и нотация UML [1]

Нотация IDEF0 - это нотация, применяемая для моделирования широкого класса систем. Результатом данного моделирования является модель системы, которая состоит из иерархии диаграмм, текста документации, которые связаны друг с другом при помощи перекрестных ссылок [2, с.19]

Нотация ARIS - предполагает построение большого числа диаграмм, для описания динамики и статистики. Данные диаграммы классифицируются по видам, типам, уровням и ракурсам описания [3, с. 6]

Нотация UML - семейство графических нотаций, в основе которого лежит единая метамодель. Нотация UML помогает в описании и проектировании программных систем, в особенности систем, которые построены с применением объектно-ориентированных технологий [4, с. 27]

Осуществим сравнение данных нотаций, и представим результаты сравнения в таблицу 2.1

Таблица 2.1 - Сравнительный анализ нотация

Критерий	Нотация IDEF0	Нотация ARIS	Нотация UML
----------	---------------	--------------	-------------

Легкость в изучении и понимании	Легок в освоении	Очень сложный в освоении	Сложный в освоении
Подход к проектированию	Функциональный	Процессный	Объектно-ориентированный
Области применения	Бизнес-процессы, программное обеспечение	Бизнес-процессы	Бизнес-процессы, программное обеспечение

Как видно из таблицы 2.1, наиболее подходящей является нотация IDEF0, поскольку нотации ARIS и UML достаточно сложны в освоении.

В настоящее время существует множество CASE средств, поддерживающих функциональное моделирование в стандарте IDEF0. Однако наиболее популярной, и легкой в понимании является AllFusion Process Modeler (BPwin).

AllFusion Process Modeler – это мощный инструмент моделирования, который создала фирма Computer Associates Technologies, и который применяется для анализа, документирования и реорганизации сложных бизнес-процессов [5].

На данный момент в электронной энциклопедии ведется ручной не компьютеризованный учет. Для построения контекстной модели, необходимо определить входные, выходные данные, управляющая информация и механизм.

В данном случае, входной информацией будет: скидочной купон; данные пользователя; данные игры.

Выходной информацией будет описание компьютерных игр.

Управляющей информацией будет: устав электронной энциклопедии компьютерных игр; и нормативно-справочная информация.

Механизмом управления будет администратор.

Построенная контекстная диаграмма показана на рисунке 2.1.

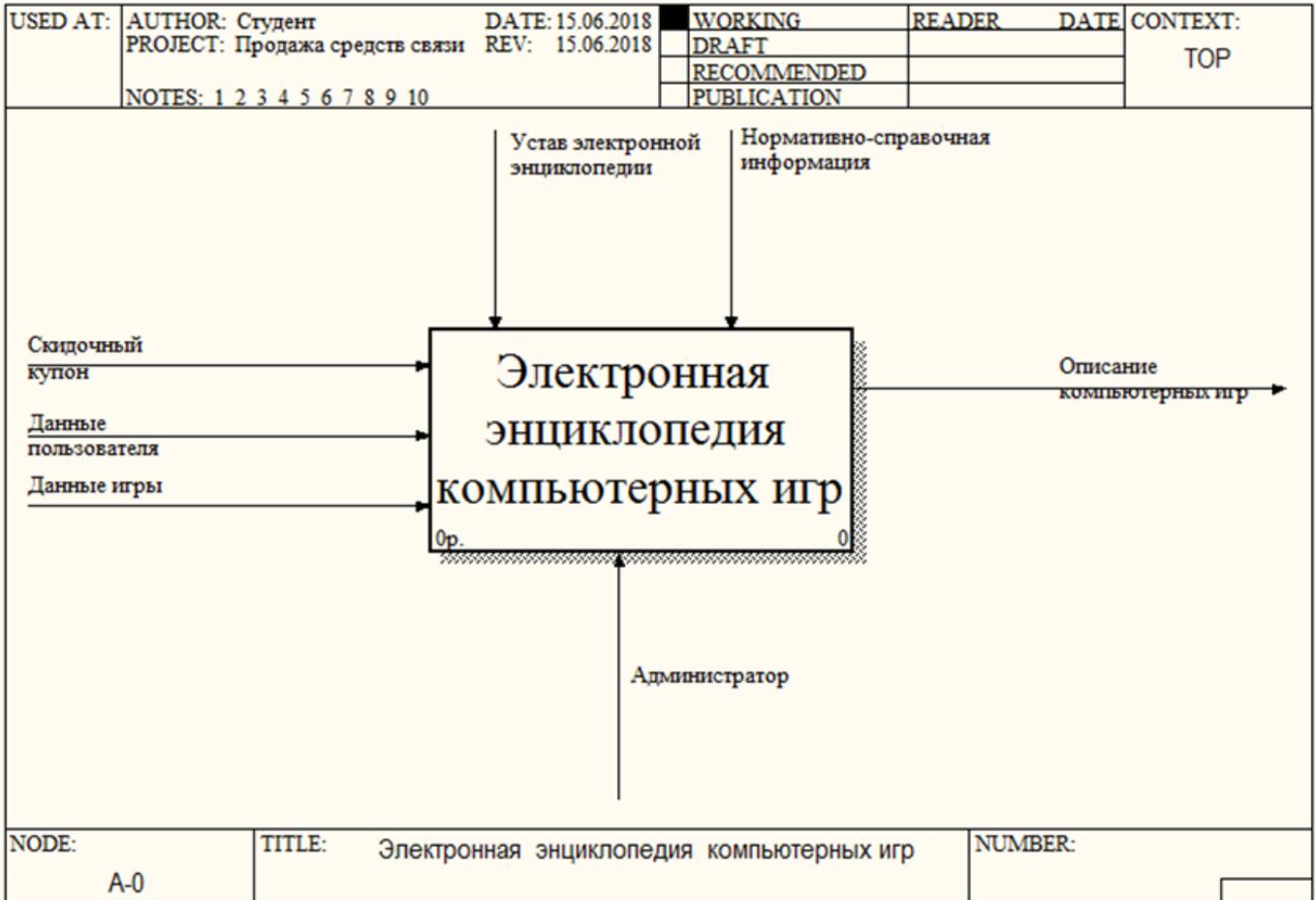


Рисунок 2.1 - Контекстная диаграмма

Декомпозируем контекстную диаграмму - рисунок 2.2.

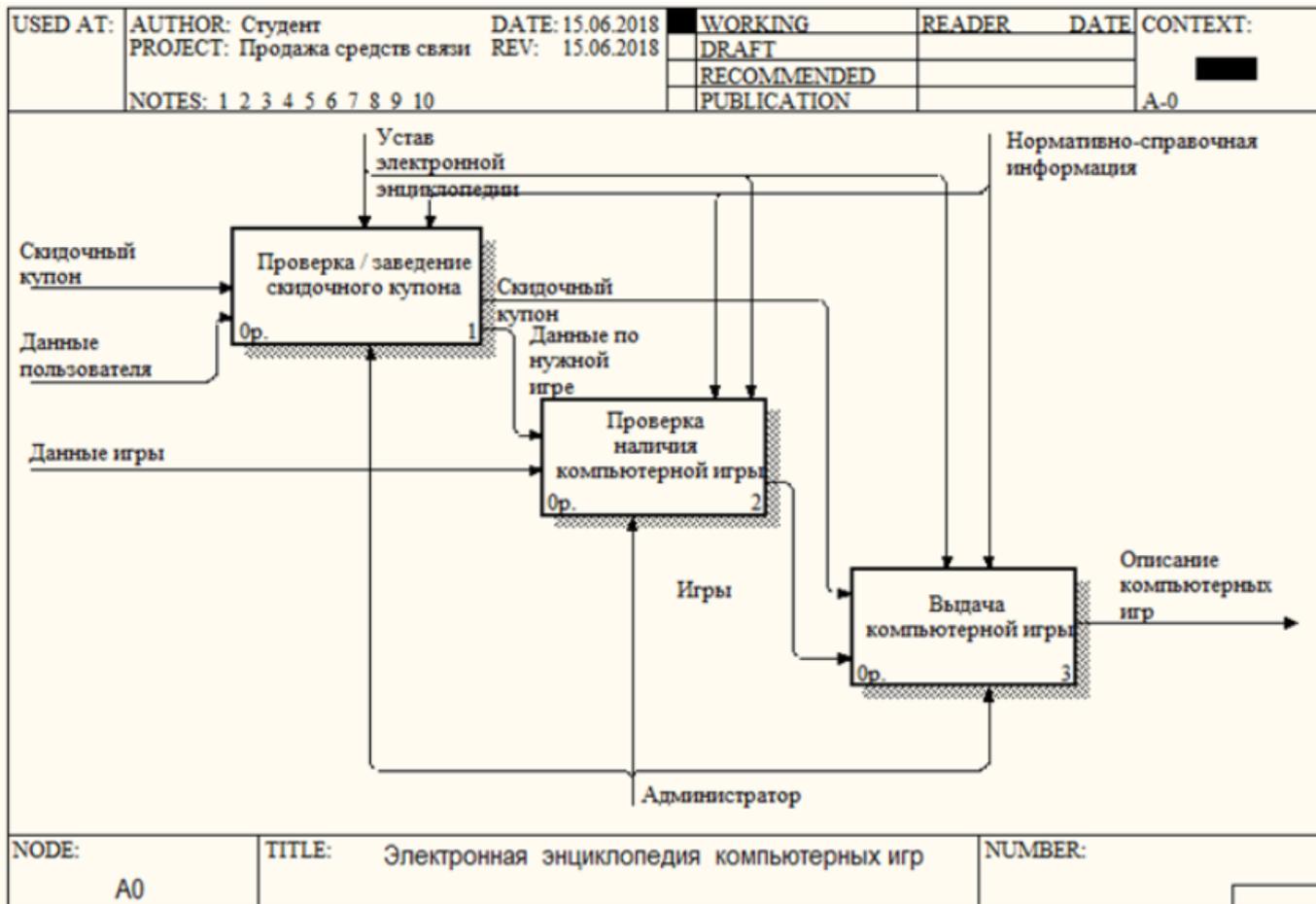


Рисунок 2.2 - Диаграмма декомпозиции контекстной диаграммы

Как видно из рисунка 2.2, диаграмма декомпозиции состоит из трех процессов: проверка / заведение скидочного купона (рисунок 2.3); проверка наличия компьютерной игры (рисунок 2.4); выдача компьютерной игры (рисунок 2.5).

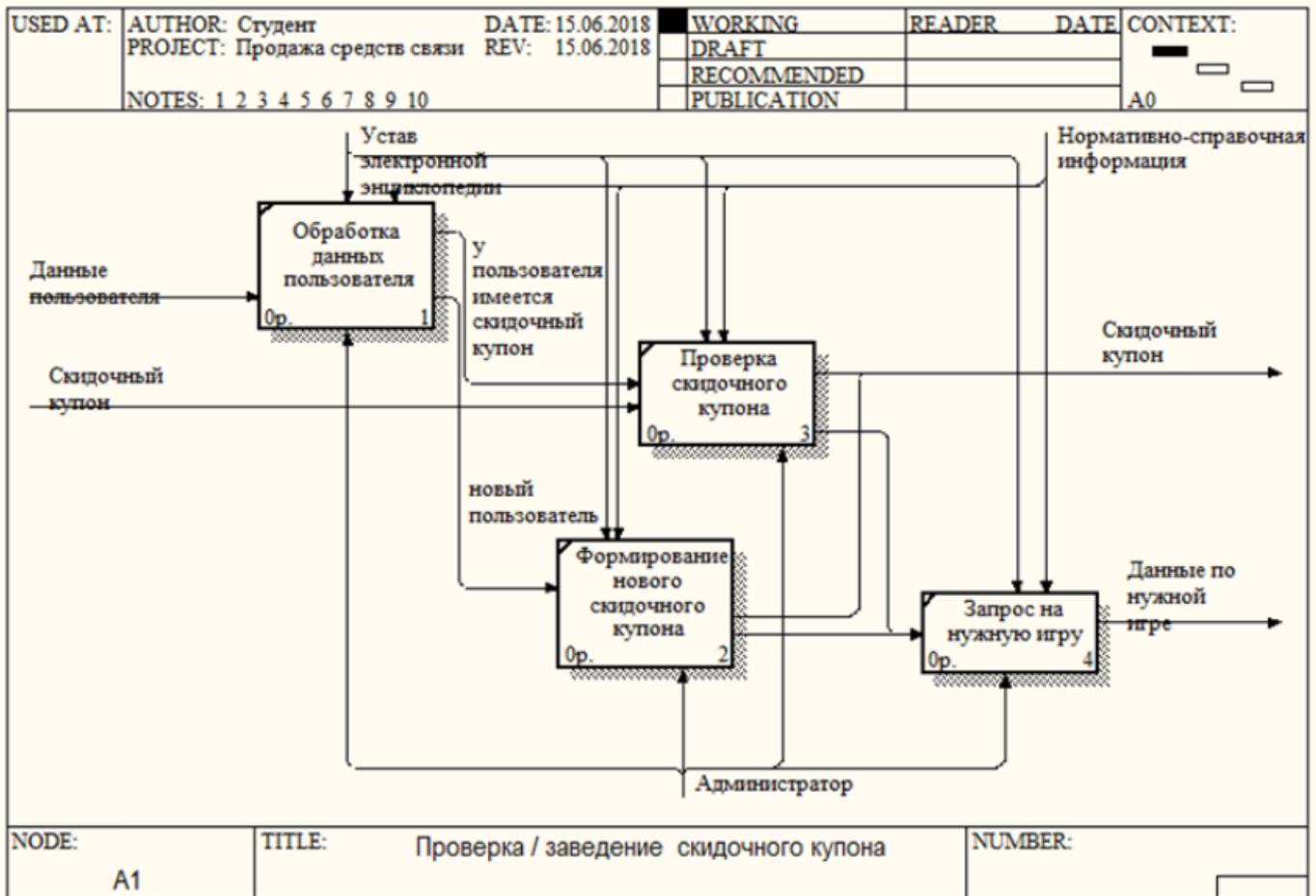


Рисунок 2.3 - Диаграмма декомпозиции процесса «Проверка / заведение скидочного купона»

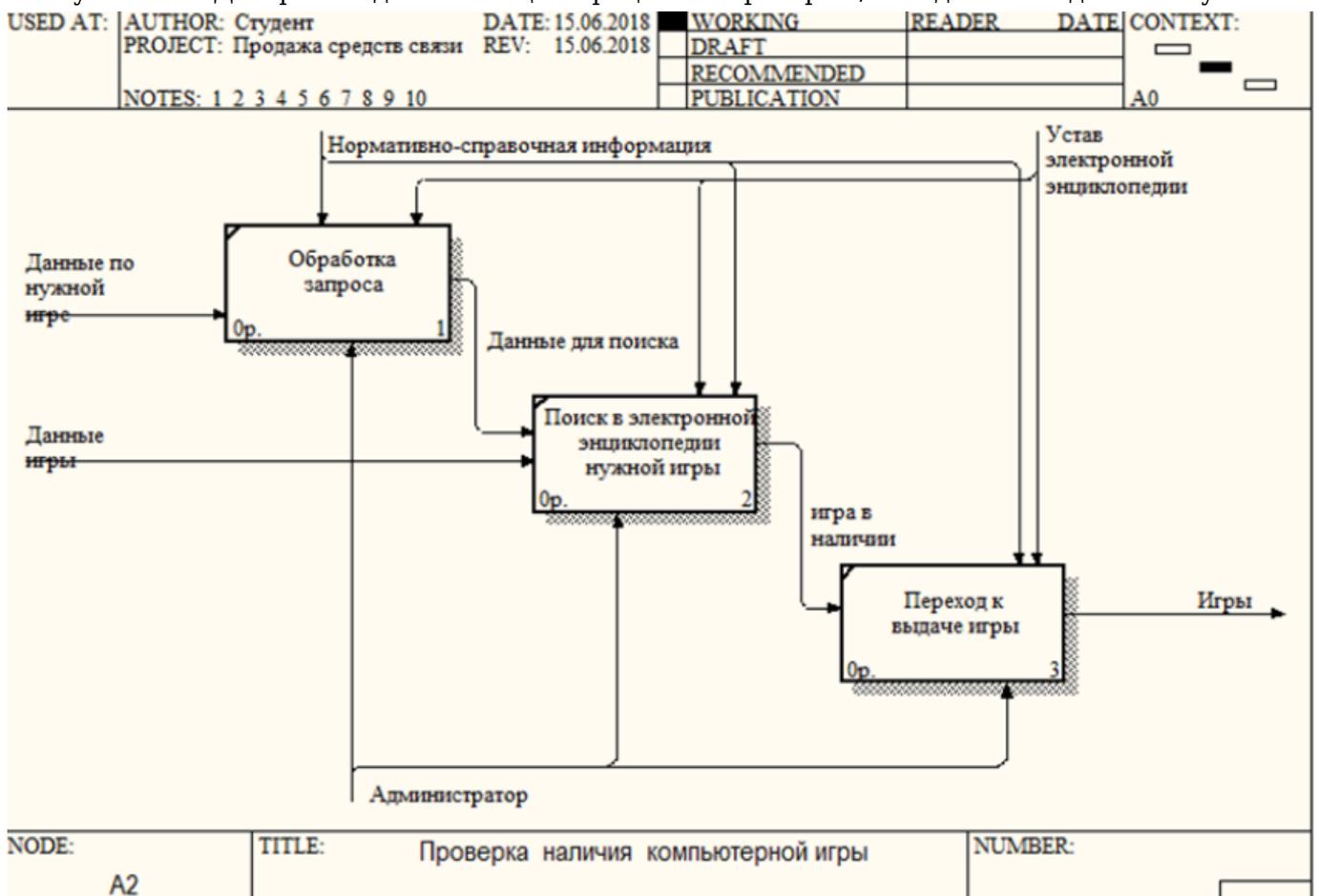


Рисунок 2.4 - Диаграмма декомпозиции процесса «Проверка наличия компьютерной игры»

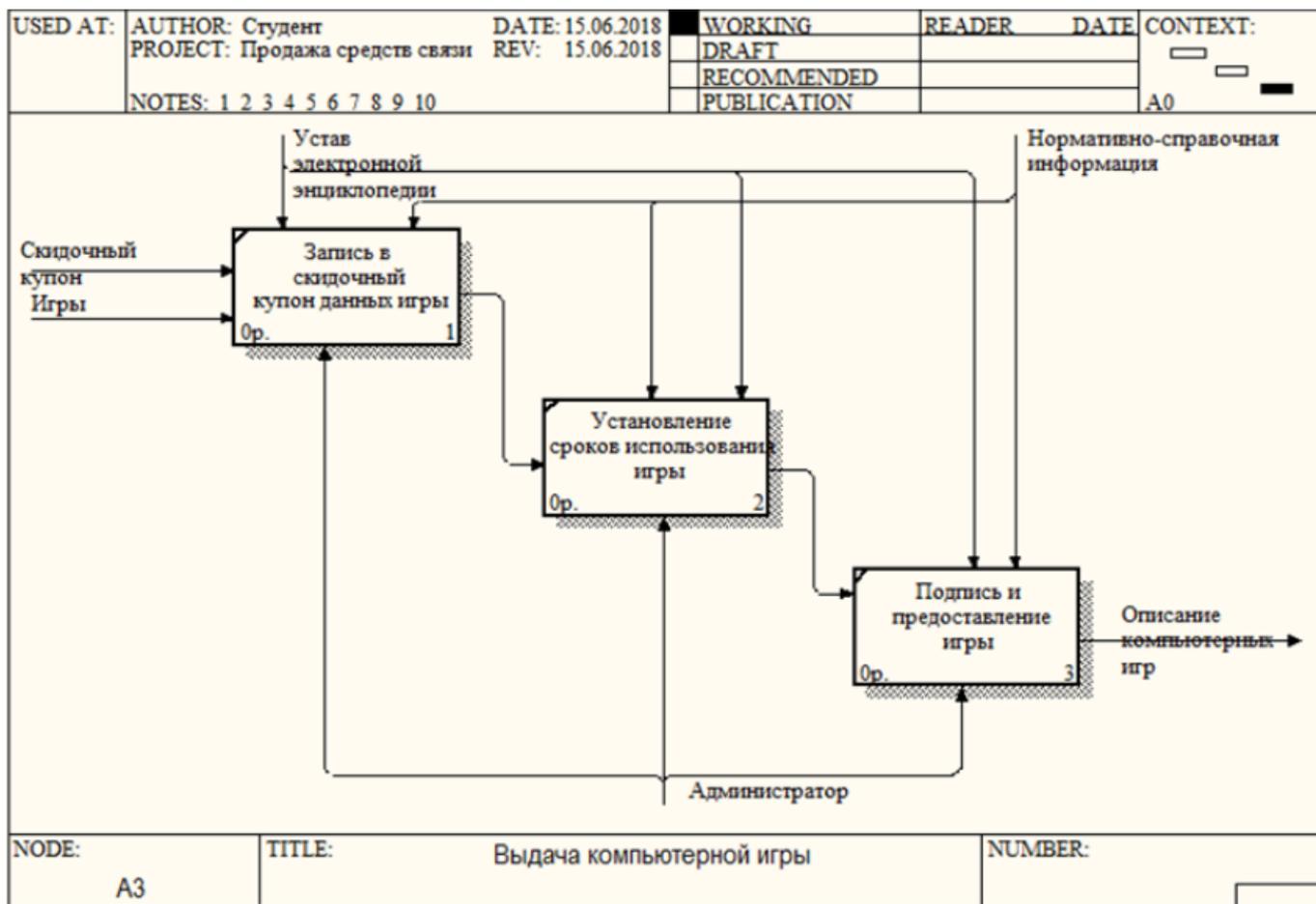


Рисунок 2.5 - Диаграмма декомпозиции процесса «Выдача компьютерной игры»

Процесс деятельности электронной энциклопедии обладает следующими недостатками:

- бумажные скидочные купоны, которые имеют возможность теряться;
- оформление всей документации вручную, что занимает достаточное большое время;
- возможность допущения ошибок;
- поиск информации по пользователю, занимает большое количество времени;
- поиск игр, также занимает большое количество времени, поскольку необходимо пересмотреть все имеющиеся компьютерные игры (по критерию), что бы выявить имеется ли требуемая игра или он уже на руках у пользователя;
- при ручном ведении игры, может произойти и потеря данных по компьютерным играм либо же потеря самих игр.

Данные недостатки можно будет решить при помощи разработки электронной энциклопедии компьютерных игр.

Для формулировки основных требований к внедряемой ИСУ, построим контекстная модель «КАК ДОЛЖНО БЫТЬ», которая показывает будущее предполагаемое состояние предметной области.

Входная и управляющая информация остается такая же, как и в модели «Как есть», однако в механизм управления добавится ИСУ электронной энциклопедии. А к выходящей информации добавится еще и отзывы компьютерных игр.

Контекстная модель показана на рисунке 2.6.



Рисунок 2.6 - Контекстная диаграмма

Декомпозиция контекстной диаграммы представлена на рисунке 2.7.

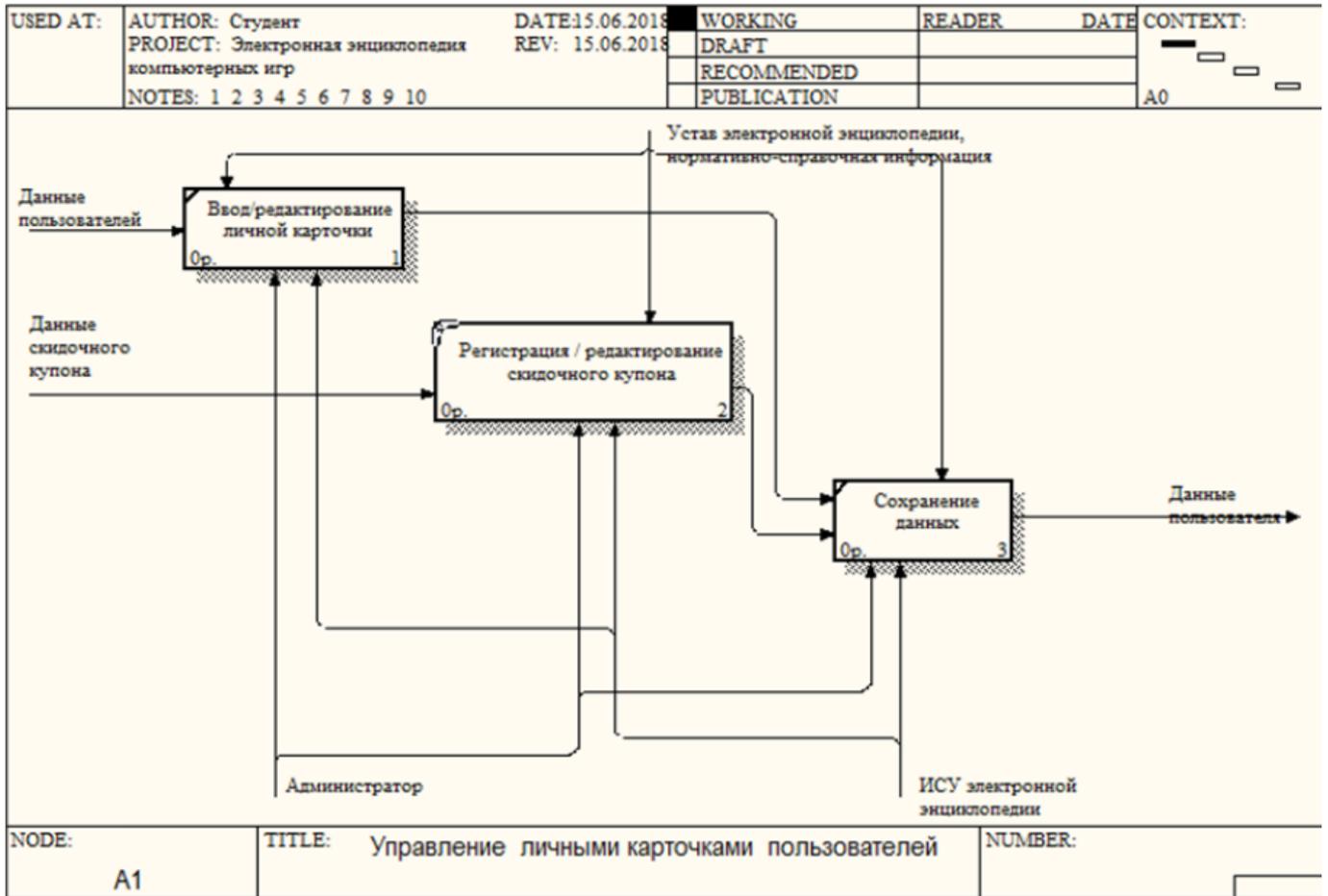


Рисунок 2.7 - Декомпозиция контекстной диаграммы

Из рисунка 2.7 видно, что декомпозиция контекстной диаграммы состоит из следующих процессов: управление личными карточками пользователей (рисунок 2.8); управление компьютерными играми (рисунок 2.9); выдача / прием компьютерных игр (рисунок 2.10); и формирование отзывов по компьютерным играм (рисунок 2.11).

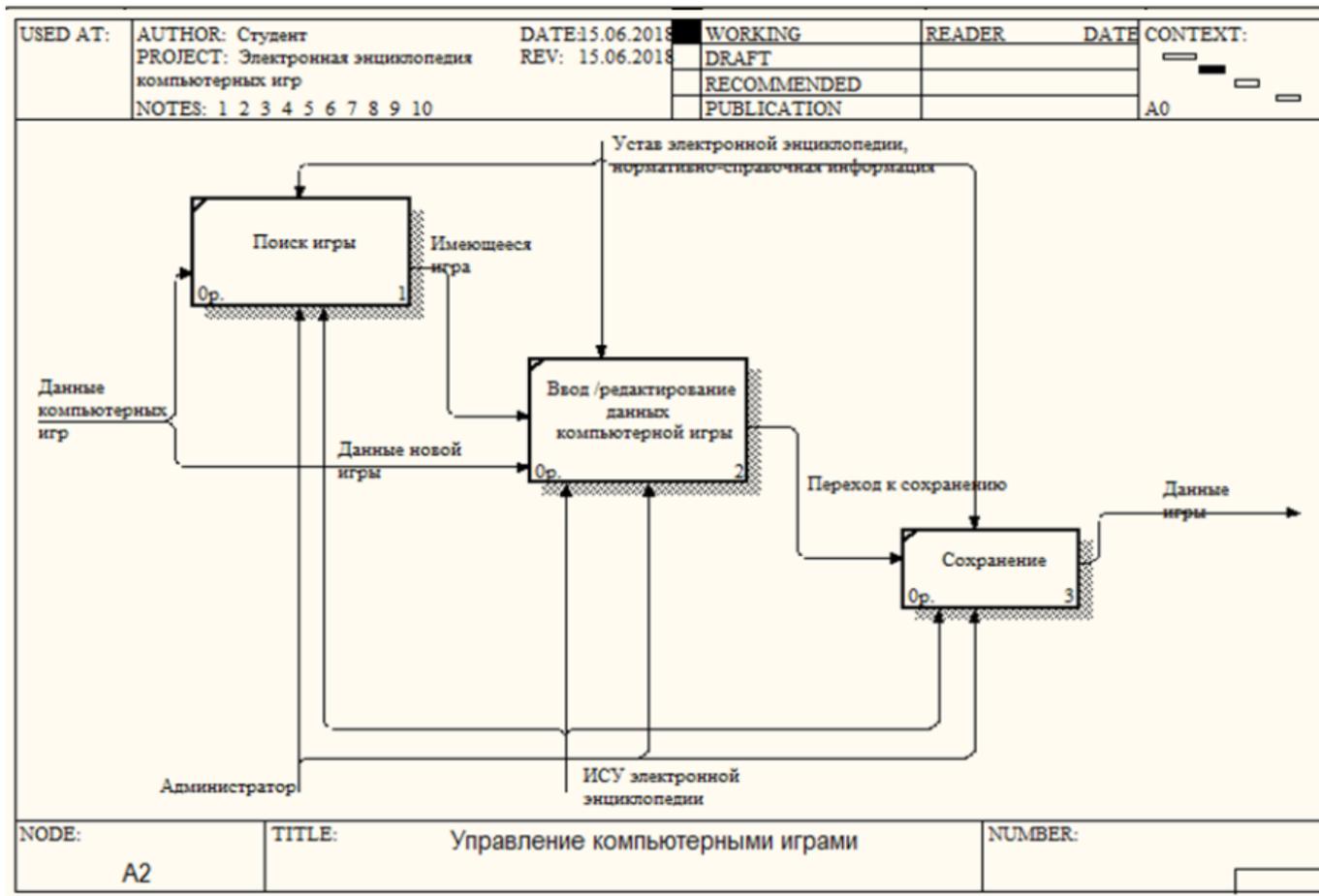


Рисунок 2.8 - Декомпозиция процесса «Управление личными карточками пользователей»

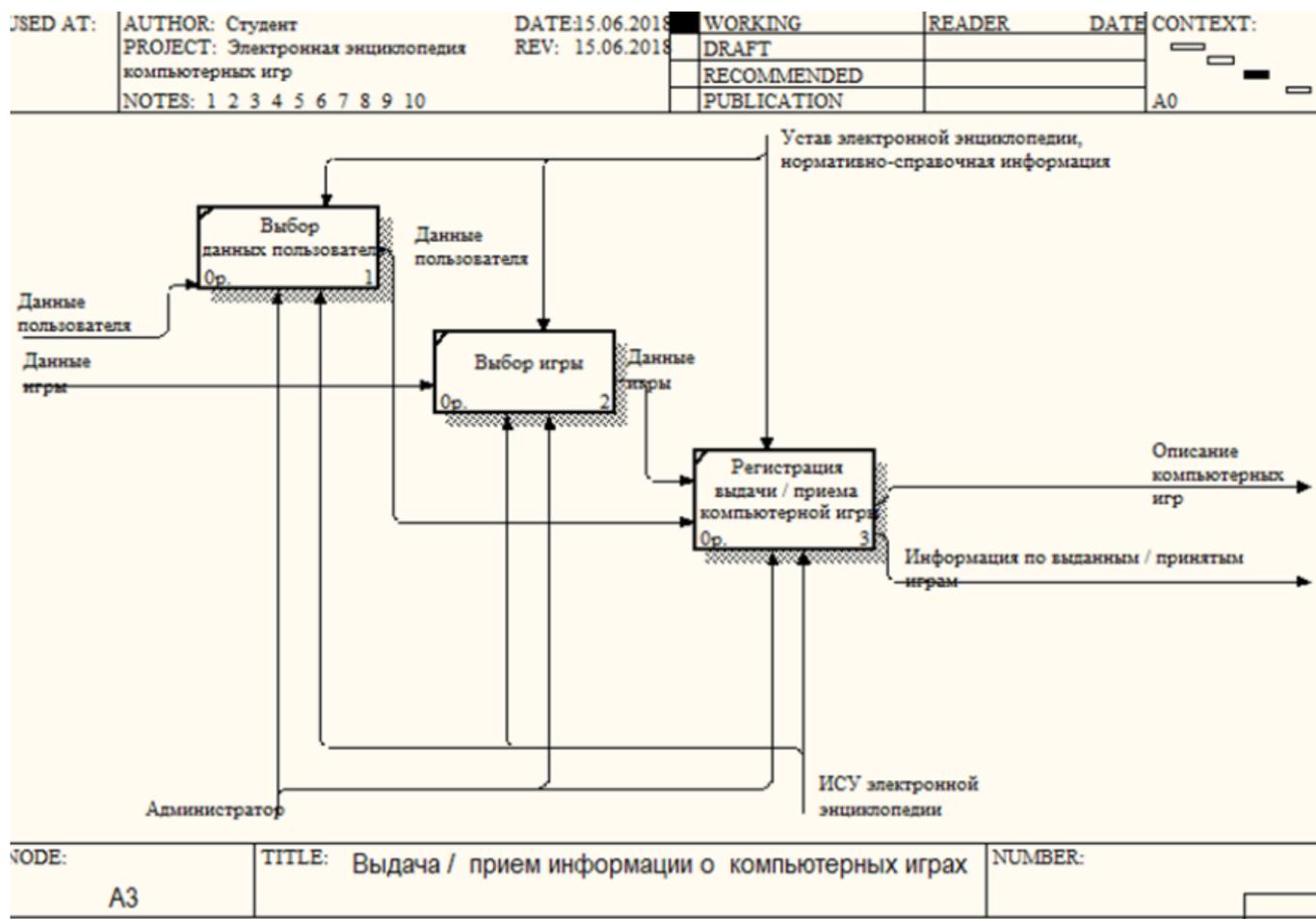


Рисунок 2.9 - Декомпозиция процесса «Управление компьютерными играми»

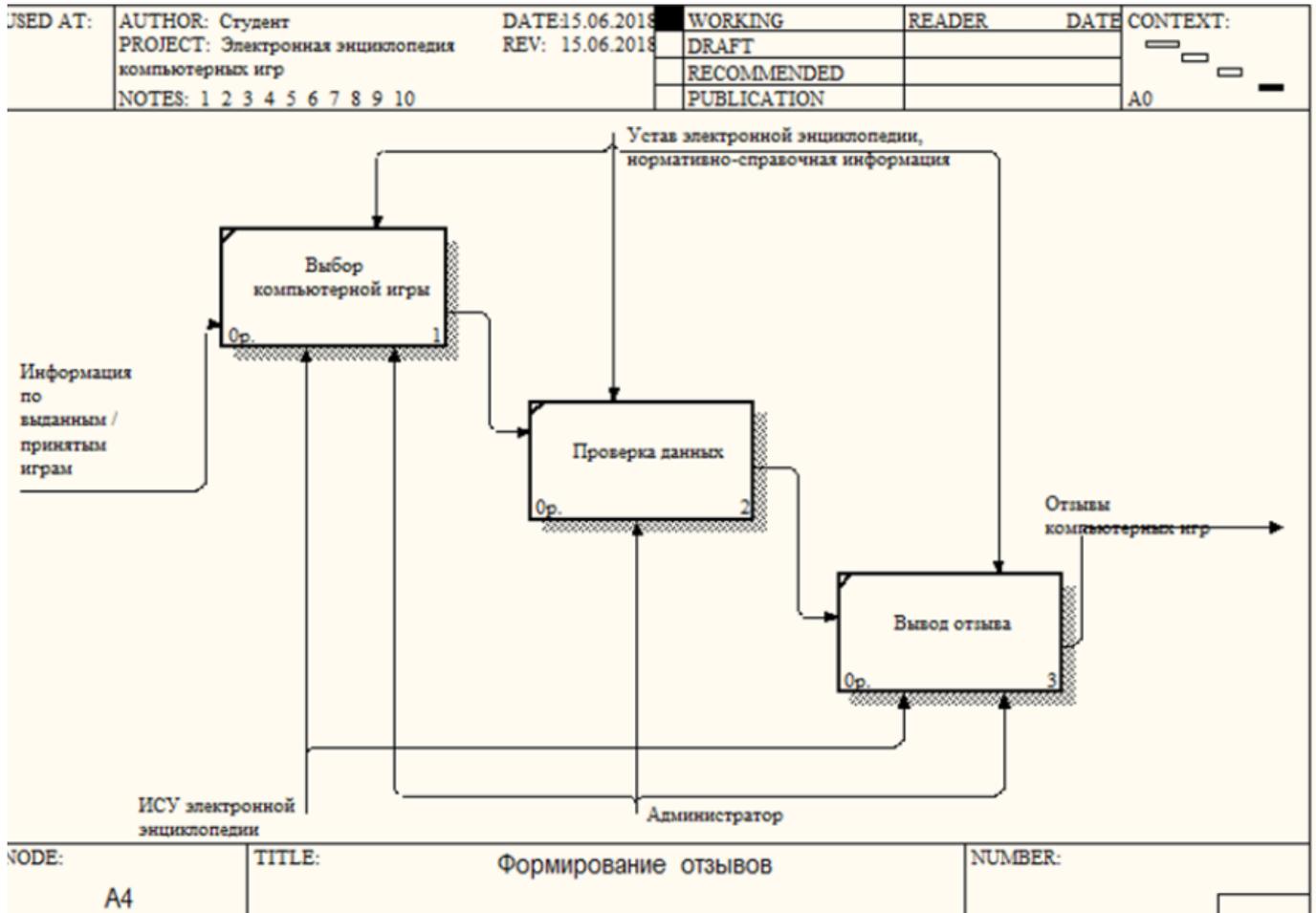


Рисунок 2.10 - Декомпозиция процесса «Выдача / прием информации о компьютерных играх»

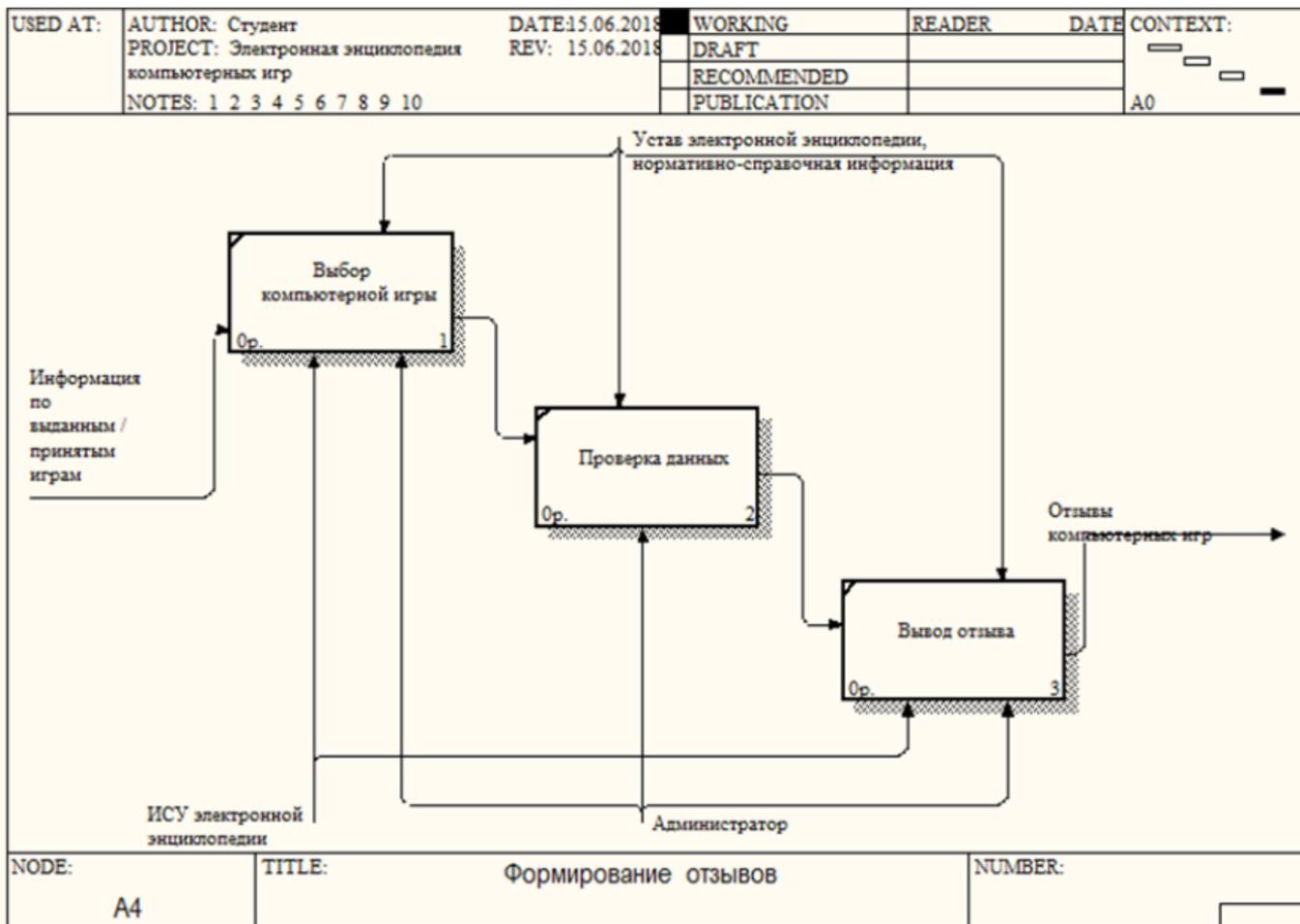


Рисунок 2.11 – Декомпозиция процесса «Формирование отзывов»

После построения модели «Как должно быть», можно сформулировать требования к новой ИСУ.

Функциональные требования:

- создание и редактирование карточек пользователей;
- формирование и печать скидочных купонов;
- добавление новых компьютерных игр и создание по ним купонов;
- редактирование купонов компьютерных игр;
- поиск информации;
- учет предоставления информации о компьютерных играх;
- учет приема информации об играх;
- получение информации о пользователях;
- получение информации по совершенным действиям пользователей;
- получение информации по совершенным действиям с определенной компьютерной игрой;
- формирование отзывов по компьютерным играм.

Нефункциональные требования:

- интерфейс электронной энциклопедии должен быть простым и легким в освоении;
- восстановление ИСУ после сбоя - не более 24 часов;
- ИСУ должна иметь многопользовательский режим работы;
- отклик ИСУ должен составлять не более 5 секунд.

3 Спецификация вариантов использования системы

Спецификация – документ, который точно, полностью и в поддающейся проверке форме определяет требования, устройство, поведение или другие особенности системы, компонента, продукта, результата или услуги, а также процедуры, способные определить, были ли выполнены эти условия.

Спецификация может содержать:

- описательное название, номер или другой идентификатор спецификации;
- время последнего пересмотра и отметку, кем был выполнен пересмотр;
- логотип или торговую марку, указывающую, кому принадлежит право на копирование, владельца и происхождение документа;
- содержание документа, если документ длинный;
- ответственное лицо или организацию по вопросам по спецификации, по обновлениям и отклонениям;
- важность, область применения спецификации и её назначение;
- термины, определения и аббревиатуры для пояснения сути спецификации;
- способы проверки для всех установленных требований и характеристик;
- материальные требования: физические, механические, электрические, химические и другие. Целевые и допустимые;
- требования по эксплуатационному тестированию. Целевые и допустимые;
- изображения, фотографии или технические иллюстрации;
- требования по мастерству;
- требования к сертифицированности;
- требования по технике безопасности;
- экологические требования;
- контроль по обеспечению качества, образец для проверки, проверка, критерий приёма работы;
- лицо или организация ответственное за выполнение спецификации;
- выполнение и доставка;
- условия по отклонениям, перепроверке, пересмотре, корректировке измерений и характеристик;
- ссылки и цитаты в тексте спецификации, которые могут потребоваться для установки ясности документа;
- подписи и разрешения, если они необходимы;
- контроль изменений (с помощью специальных компьютерных программ) для последовательной разработки, проверки и выполнения, если документ предназначен для внутреннего использования;
- приложения, которые раскрывают детали, добавляют ясности, или пояснения по оплате.

Сценарий использования, вариант использования, прецедент использования (англ. use case) – в разработке программного обеспечения и системном проектировании это описание поведения системы, когда она взаимодействует с кем-то (или чем-то) из внешней среды. Система может отвечать на внешние запросы Актёра (англ. actor) (может применяться термин Актант), может сама выступать инициатором взаимодействия. Другими словами, сценарий использования описывает, «кто» и «что» может сделать с рассматриваемой системой, или что система может сделать с «кем» или «чем». Методика сценариев использования применяется для выявления требований к поведению системы, известных также как пользовательские и

функциональные требования.

В системном проектировании сценарии использования применяются на более высоком уровне, чем при разработке программного обеспечения, часто представляя цели заинтересованных лиц или миссии. На стадии анализа требований сценарии использования могут быть преобразованы в ряд детальных требований и задокументированы с помощью диаграмм требований SysML или других подобных механизмов.

Диаграмма прецедентов (диаграмма вариантов использования) в UML - диаграмма, отражающая отношения между актёрами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Прецедент - возможность моделируемой системы (часть её функциональности), благодаря которой пользователь может получить конкретный, измеримый и нужный ему результат. Прецедент соответствует отдельному сервису системы, определяет один из вариантов её использования и описывает типичный способ взаимодействия пользователя с системой. Варианты использования обычно применяются для спецификации внешних требований к системе.

Диаграмма вариантов использования представлена на рисунке 3.1.

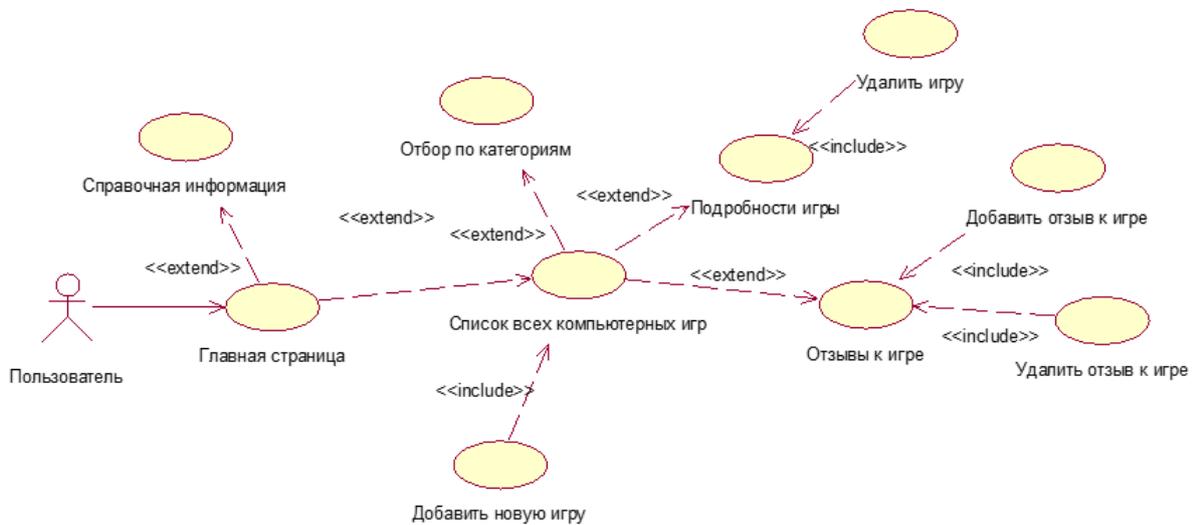


Рисунок 3.1 - Диаграмма вариантов использования

Разработка веб-приложения «Электронная энциклопедия компьютерных игр» включает в себя следующий функционал:

- поиск игр по сайту;
- применение фильтров к компьютерным играм по категории (новинки, лидеры продаж,

- распродажа);
- отображение подробной информации о компьютерных играх;
- добавление и удаления компьютерной игры;
- добавление и удаления отзывов к компьютерной игре;
- отображение справочной информации;
- отображение компьютерных игр таблицей.

Подробное описание всех разработанных функций представлено в приложении А.

4 Информационная модель системы и её описание

Созданная база данных состоит из четырех таблиц. Структура базы данных отображена на рисунке 4.1. Описание таблиц базы данных приведено в таблице 4.1.

Таблица 4.1 - Состав таблиц базы данных

Имя таблицы	Описание
1 category	Справочник категорий компьютерных игр
2 list_games	Справочник компьютерных игр
3 uploads_images	Справочник загруженных фотографий к играм
4 reviews_game	Справочник отзывов к компьютерным играм

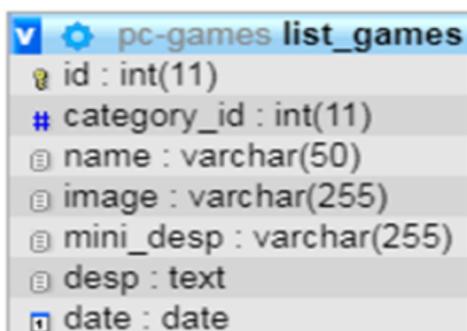
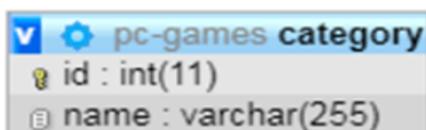


Рисунок 4.1 - Структура базы данных

В таблицах 4.2 - 4.5 приведено описание состава таблиц спроектированной базы данных.

Таблица 4.2 – Категории компьютерных игр

Ключ (Да/Нет)	Поле	Тип	Значение по умолчанию	Пустые значения (Да/Нет)
РК	id	число	Нет	Нет
Нет	name	строка	Нет	Нет

Таблица 4.3 – Компьютерные игры

Ключ (Да/Нет)	Поле	Тип	Значение по умолчанию	Пустые значения (Да/Нет)
РК	id	число	Нет	Нет
FK	category_id	число	Нет	Нет
Нет	name	строка	Нет	Нет
Нет	image	строка	Нет	Нет
Нет	mini_desp	строка	Нет	Нет
Нет	desp	текст	Нет	Нет
Нет	date	дата	Нет	Нет

Таблица 4.4 – Отзывы к товарам

Ключ (Да/Нет)	Поле	Тип	Значение по умолчанию	Пустые значения (Да/Нет)
РК	id	число	Нет	Нет
FK	game_id	число	Нет	Нет
Нет	name	строка	Нет	Нет
Нет	comment	текст	Нет	Нет
Нет	date	дата	Нет	Нет

Таблица 4.6 – Загруженные фотографии к играм

Ключ (Да/Нет)	Поле	Тип	Значение по умолчанию	Пустые значения (Да/Нет)
РК	id	число	Нет	Нет
Нет	image	строка	Нет	Нет

В результате была спроектирована база данных для ведения электронной энциклопедии компьютерных игр. База данных включает в себя информацию о категориях компьютерных игр, компьютерных играх, отзывах к компьютерным играм, загруженным фотографиям к компьютерным играм.

5 Обоснование выбора компонентов и технологий для реализации курсового проекта

Наиболее распространенным языком разработки сайта является Язык разметки гипертекстовых страниц (HTML – Hypertext Markup Language) представляет собой язык, разработанный специально для создания Web-документов. Он определяет синтаксис и

размещение специальных инструкций (тегов), которые не выводятся на экран, но указывают браузеру, как отображать содержимое документа. Он также используется для создания ссылок на другие документы, локальные или сетевые, например, находящиеся в сети Интернет [9, с. 37].

Стандарт HTML и другие стандарты для Web разработаны под руководством консорциума W3C (World Wide Web Consortium). Стандарты, спецификации и проекты новых предложений можно найти на сайте <http://www.3w.org/>. В настоящее время действует спецификация HTML 4.0, поддержка которой со стороны основных браузеров постоянно растет.

На практике на стандарт HTML большое влияние оказывает наличие тегов, предложенных и поддерживаемых наиболее известными браузерами, такими как Microsoft Internet Explorer и Netscape Navigator. Эти теги в данный момент могут, как входить, так и не входить в состав действующей спецификации HTML.

PHP – язык программирования, используемый на стороне Web-сервера для динамической генерации HTML-страниц. Об этом говорит и расшифровка его названия: PHP – Personal HyperText Processor.

PHP – один из немногих языков программирования, созданных специально для разработки Web-приложений. Поэтому он включает в себя все функции, необходимые именно для работы на Web-сервере, и при этом лишен избыточности, свойственной многим его конкурентам.

Очень приятная особенность PHP – то, что его команды включаются в обычные HTML-страницы с помощью специальных тегов, которые и заставляют PHP-машину выполнять на сервере нужные действия. Программам на PHP не нужны специальные CGI-директории с особыми правами доступа. Более того, на одной страничке можно произвольно чередовать «простой» HTML и PHP-код [3, с. 110].

PHP не зависит от платформы. PHP прекрасно интегрируется во все популярные Web-серверы: Apache и IIS, Zens и Netscape Enterprise Server, работает под Windows и OS/2, MacOS и практически всеми UNIX-подобными системами. Как следствие – PHP работает практически у всех хостеров, разрешающих собственные выполняемые скрипты.

Замечательная особенность PHP – его интегрированность практически со всеми современными интернет-технологиями. PHP поддерживает большинство современных Web-протоколов: IMAP, FTP, POP, XML, SNMP и другие. PHP прекрасно работает с базами данных. Трудно найти СУБД, поддержка которой не была бы реализована в PHP. MySQL и MS SQL Server, PostgreSQL и Oracle, Sybase и Interbase. Один только перечень баз данных, поддерживаемых PHP, займет, наверное, целый экран [1, с. 321].

PHP включает в себя огромное количество встроенных функций: обработки строк и массивов, работы с файловой системой и с HTTP, электронной почтой, датой и временем, кириллицей и другими национальными алфавитами, и большим количеством встроенных функций. Благодаря им многие алгоритмы, требующие в большинстве языков написания программного кода размером в несколько экранов, реализуются на PHP одной командой (точнее, вызовом одной функции).

Современные тенденции развития языков программирования не обошли стороной и PHP. Средства объектно-ориентированного программирования появились еще в PHP3. А в объектной

модели PHP в полном объеме реализованы классические понятия объектно-ориентированного программирования: наследование, инкапсуляция и полиморфизм.

Основное отличие от CGI-скриптов, написанных на других языках, типа Perl или C – это то, что в CGI-программах вы сами пишете выводимый HTML-код, а, используя PHP – вы встраиваете свою программу в готовую HTML-страницу, используя открывающий и закрывающий теги.

Отличие PHP от JavaScript, состоит в том, что PHP-скрипт выполняется на сервере, а клиенту передается результат работы, тогда как в JavaScript-код полностью передается на клиентскую машину и только там выполняется.

Диаграмма последовательности (англ. sequence diagram) – диаграмма, на которой для некоторого набора объектов на единой временной оси показан жизненный цикл какого-либо определённого объекта (создание-деятельность-уничтожение некой сущности) и взаимодействие актёров (действующих лиц) ИС в рамках какого-либо определённого прецедента (отправка запросов и получение ответов). Используется в языке UML.

Основными элементами диаграммы последовательности являются обозначения объектов (прямоугольники с названиями объектов), вертикальные «линии жизни» (англ. lifeline), отображающие течение времени, прямоугольники, отражающие деятельность объекта или исполнение им определенной функции (прямоугольники на пунктирной «линии жизни»), и стрелки, показывающие обмен сигналами или сообщениями между объектами.

Диаграмма последовательности представлена на рисунке 5.1.

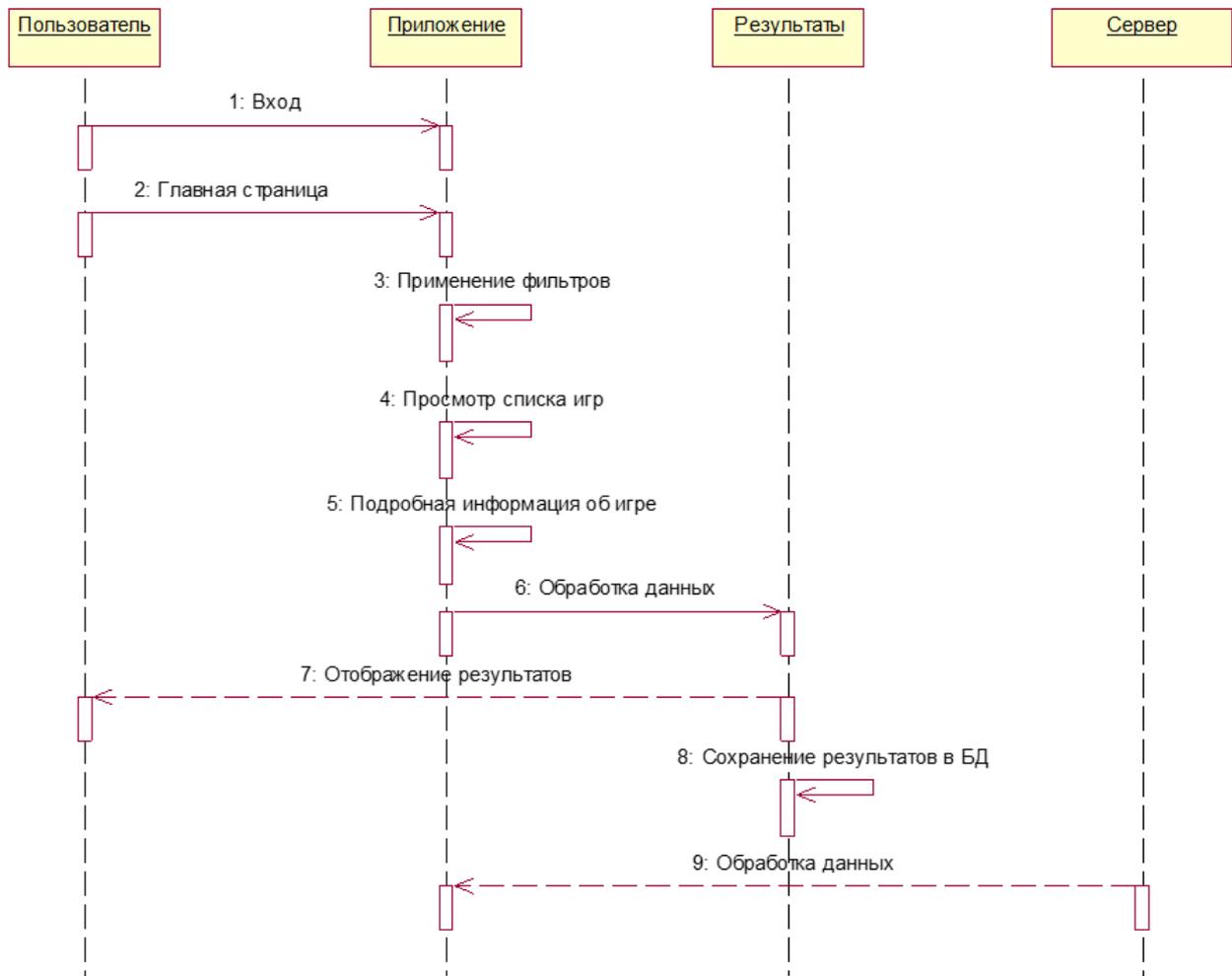


Рисунок 5.1 – Диаграмма последовательности

Диаграмма компонентов (англ. Component diagram) – элемент языка моделирования UML, статическая структурная диаграмма, которая показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами. В качестве физических компонентов могут выступать файлы, библиотеки, модули, исполняемые файлы, пакеты и т. п.

Диаграмма компонентов представлена на рисунке 5.2.

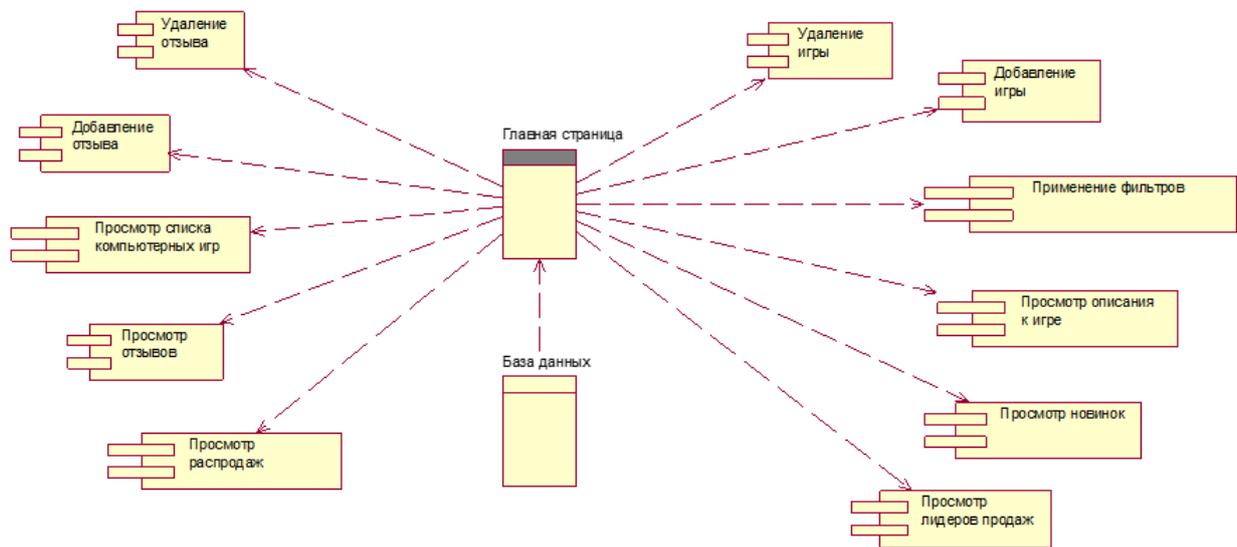


Рисунок 5.2 - Диаграмма компонентов

Диаграмма развёртывания, Deployment diagram в UML моделирует физическое развёртывание артефактов на узлах. Например, чтобы описать веб-сайт диаграмма развёртывания должна показывать, какие аппаратные компоненты («узлы») существуют (например, веб-сервер, сервер базы данных, сервер приложения), какие программные компоненты («артефакты») работают на каждом узле (например, веб-приложение, база данных), и как различные части этого комплекса соединяются друг с другом (например, JDBC, REST, RMI).

Диаграмма развёртывания представлена на рисунке 5.3.

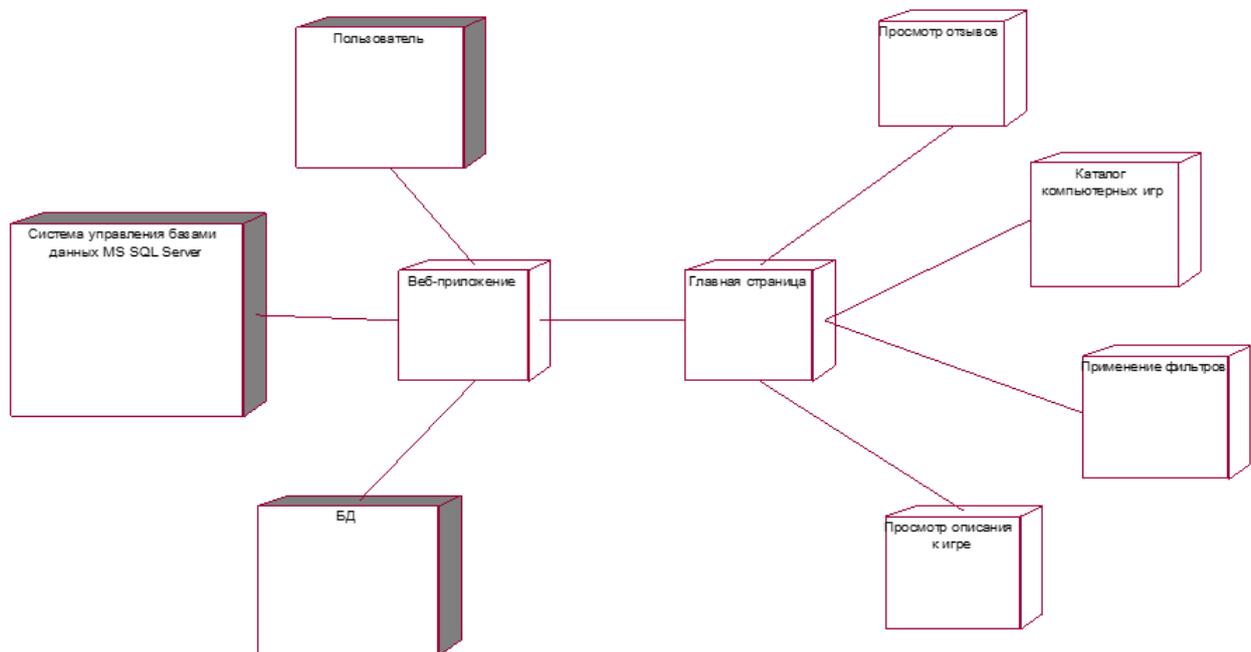


Рисунок 5.3 - Диаграмма развертывания

6 Модели представления системы и их описание

В данном разделе будет продемонстрировано моделирование системы с помощью стандарта UML, который использует графические обозначения для создания абстрактной модели системы и предназначен для определения, визуализации, проектирования и документирования в основном программных систем. UML позволяет описать систему практически со всех возможных точек зрения и разные аспекты поведения системы.

Для данной курсовой работы были построены такие диаграммы, как диаграммы вариантов использования, последовательности, состояний, классов, развертывания и компонентов.

Диаграмма вариантов использования состоит из актеров, для которых система производит действие и собственно действия Use Case, которое описывает то, что актер хочет получить от системы. Диаграмма вариантов использования представлена в приложении Б.

Диаграмма состояний предназначена для отображения состояний объектов системы, имеющих сложную модель поведения. Она показывает пространство состояний системы или ее элементов, события, которые влекут переход из одного состояния в другое, действия, которые происходят при изменении состояния. Объекты меняют своё состояние в ответ на

происходящие события и течением времени. Диаграмма состояний представляет состояния объекта и переходы между ними, а также начальное и конечное состояние объекта. Диаграмма состояний представлена в приложении Б и на рисунке 6.1.

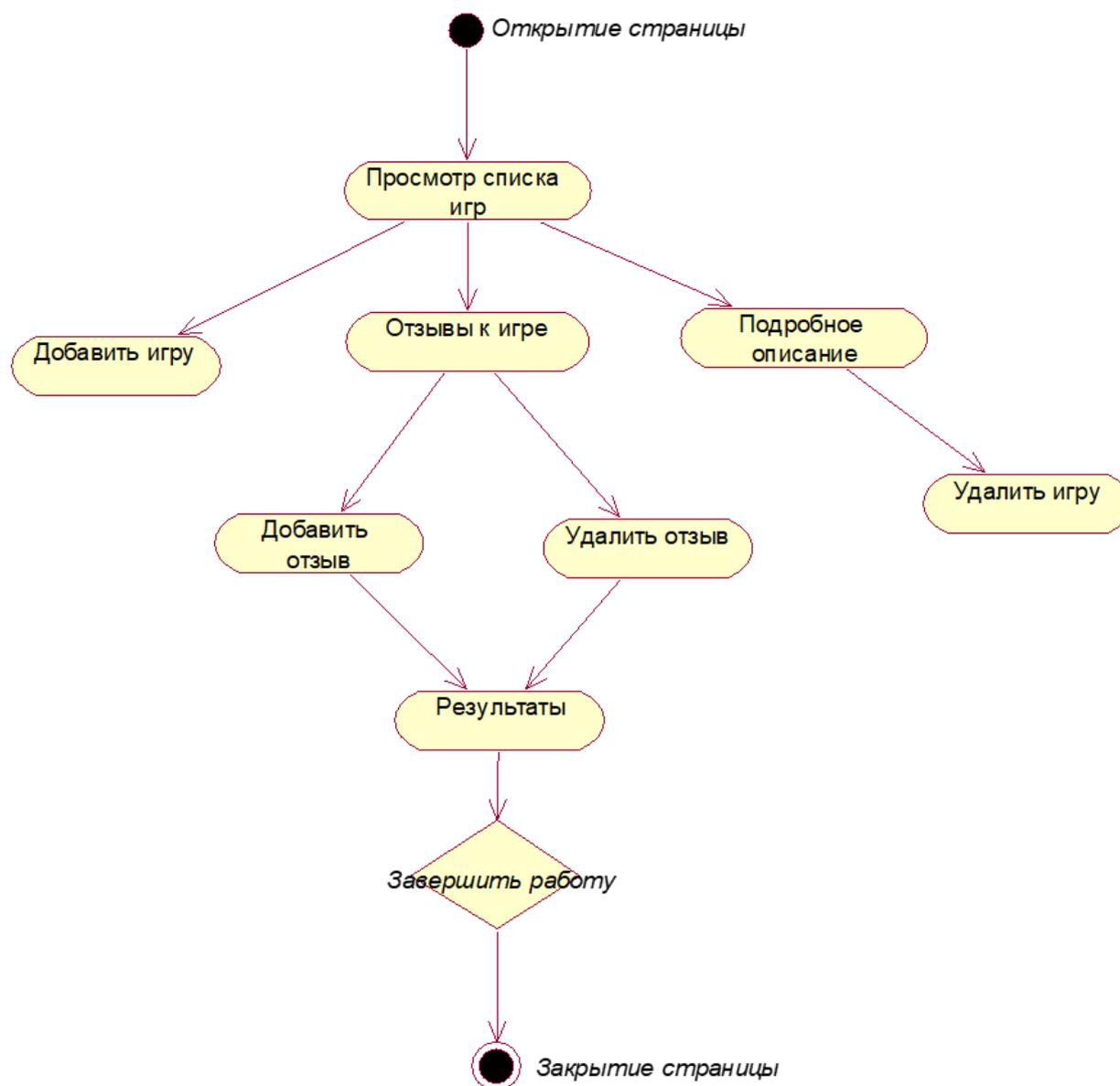


Рисунок 6.1 - Диаграмма состояний

Для моделирования взаимодействия объектов во времени в языке UML используются диаграммы последовательностей. Диаграмма последовательности представлена в приложении Б.

Диаграмма классов описывает структуру системы, показывая её классы, их атрибуты и операторы, а также взаимосвязи этих классов. Диаграмма классов представлена в приложении Б.

Диаграмма компонентов показывает разбиение программной системы на структурные

компоненты и связи между компонентами. Диаграмма компонентов представлена в приложении Б.

Диаграмма развёртывания предназначена для визуализации элементов и компонентов программы, существующих лишь на этапе ее исполнения. Диаграмма развёртывания представлена в приложении Б.

Перед программной реализацией необходимо определиться с содержанием сайта. Информация, которая должна представляться на странице, должна удовлетворять следующим критериям:

- соответствовать целям разработки сайта;
- учитывать особенности целевого сегмента потребителей;
- быть уникальной, чтобы привлечь внимание посетителей;
- быть оперативной и достоверной, то есть информация на сайте постоянно должна обновляться.

Разработка графического дизайна электронной энциклопедии компьютерных игр:

Лицом сайта является главная страница.

Первым этапом разработки сайта является разработка дизайна. При разработке дизайна необходимо решить некоторые задачи, а именно соответствие сайта стилю предприятия, использование логотипа и цветов предприятия, а также удобство сайта для посетителей [19, с. 200].

Графическое оформление сайта подразумевает выбор цветового оформления, создание статических и динамических элементов, подбор шрифтов и подбор фона.

Оформление для электронной энциклопедии компьютерных игр было выбрано стандартное, то есть в данное оформление не включается разработка оригинальных графических элементов, а используются оригиналы графических элементов, представленные предприятием и подбор подходящего шаблона.

Разрабатываемый дизайн будет соответствовать следующим требованиям:

- поскольку электронная энциклопедия компьютерных игр несет информационную нагрузку, то графическое оформление должно быть легким, и не раздражающим глаза;
- цвета, шрифты и графика должны быть выдержаны в едином стиле для всех страниц сайта.
- В дизайне использоваться цвета: голубой - серый - белый;
- графика должна быть качественной и сочетаться с остальными составляющими страницы;
- текст должен легко читаться и не сливаться с фоном.

Моделирование и разработка электронной энциклопедии компьютерных игр:

Этап разработки структуры имеет особое значение, поскольку от него зависит удобство пользования сайтом. На данном этапе разрабатывается документ, который служит исходным материалом для создания сайта: разработки сценария, графической концепции и структуры, программных инструментов, обеспечивающих необходимые функциональные ресурсы, и так далее.

Структура разрабатываемого сайта будет довольно простой и будет иметь все элементы навигации для удобного перемещения пользователя по сайту, структура разрабатываемого

сайта приведена на рисунке 6.2.



Рисунок 6.2 - Структура электронной энциклопедии компьютерных игр

Блок-схема алгоритма - графическое изображение алгоритма в виде связанных между собой с помощью стрелок (линий перехода) и блоков - графических символов, каждый из которых соответствует одному шагу алгоритма. Внутри блока дается описание соответствующего действия.

Блок «процесс» применяется для обозначения действия или последовательности действий, изменяющих значение, форму представления или размещения данных. Для улучшения наглядности схемы несколько отдельных блоков обработки можно объединять в один блок. Представление отдельных операций достаточно свободно.

Блок «решение» используется для обозначения переходов управления по условию. В каждом блоке «решение» должны быть указаны вопрос, условие или сравнение, которые он определяет.

Блок «модификация» используется для организации циклических конструкций. (Слово

«модификация» означает «видоизменение, преобразование»). Внутри блока записывается параметр цикла, для которого указываются его начальное значение, граничное условие и шаг изменения значения параметра для каждого повторения.

Блок «предопределенный процесс» используется для указания обращений к вспомогательным алгоритмам, существующим автономно в виде некоторых самостоятельных модулей, и для обращений к библиотечным подпрограммам.

Блок-схема алгоритма веб-приложения представлена на рисунке 6.3.

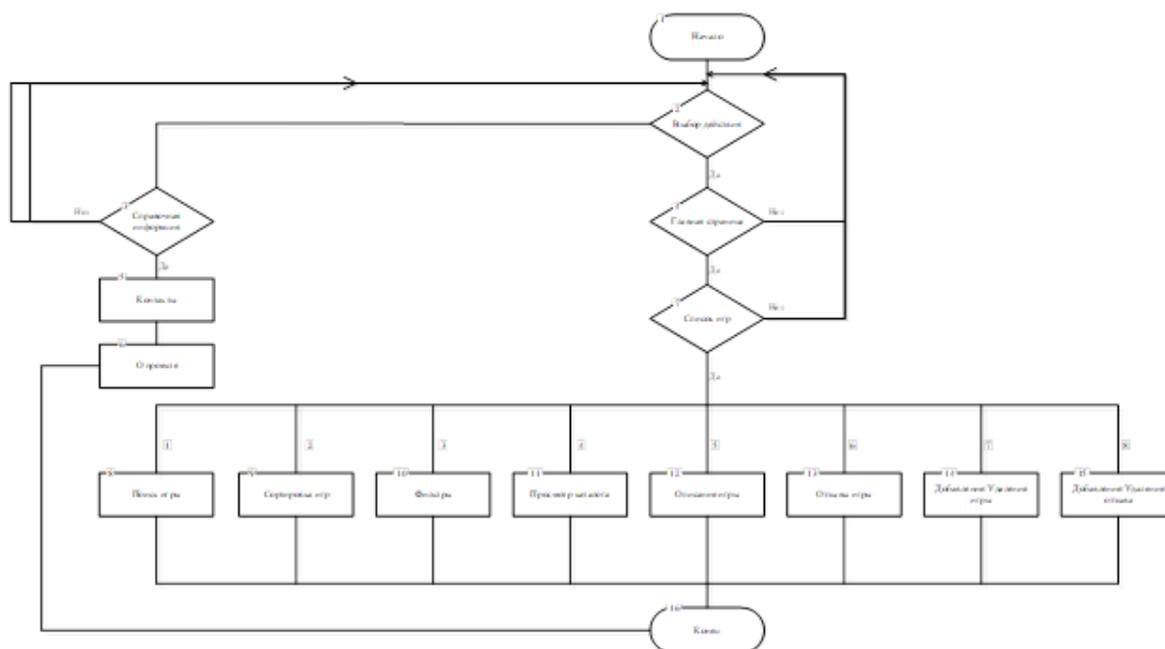


Рисунок 6.3 – Блок-схема

В фазе анализа и проектирования системы бизнес-логика воплощается в классах и методах классов, в случае использования объектно-ориентированных языков программирования, или процедур и функций, в случае применения процедурных языков.

На жаргоне разработчиков ПО бизнес-логикой также называются программные модули, её реализующие, и уровень системы, на котором эти модули находятся (Business Logic Layer, Domain Logic Layer).

В многоуровневых информационных системах этот уровень взаимодействует с нижележащим уровнем инфраструктурных сервисов (Infrastructure Layer), например, интерфейсом к базе данных или файловой системе (Data-Access Layer, DAL) и вышележащим уровнем сервисов приложения (Application Services Layer), который уже, в свою очередь, взаимодействует с уровнем пользовательского интерфейса (User Interface Layer) или внешними системами.

7 Описание применения паттернов проектирования

Паттерн проектирования – это часто встречающееся решение определённой проблемы при проектировании архитектуры программ.

В отличие от готовых функций или библиотек, паттерн нельзя просто взять и скопировать в программу. Паттерн представляет собой не какой-то конкретный код, а общую концепцию решения той или иной проблемы, которую нужно будет ещё подстроить под нужды вашей программы.

Паттерны часто путают с алгоритмами, ведь оба понятия описывают типовые решения каких-то известных проблем. Но если алгоритм – это чёткий набор действий, то паттерн – это высокоуровневое описание решения, реализация которого может отличаться в двух разных программах.

Если привести аналогии, то алгоритм – это кулинарный рецепт с чёткими шагами, а паттерн – инженерный чертёж, на котором нарисовано решение, но не конкретные шаги его реализации.

Описания паттернов обычно очень формальны и чаще всего состоят из таких пунктов:

- проблема, которую решает паттерн;
- мотивации к решению проблемы способом, который предлагает паттерн;
- структуры классов, составляющих решение;
- примера на одном из языков программирования;
- особенностей реализации в различных контекстах;
- связей с другими паттернами.

Такой формализм в описании позволил создать обширный каталог паттернов, проверив каждый из них на состоятельность.

Диаграмма классов (англ. Static Structure diagram) – диаграмма, демонстрирующая классы системы, их атрибуты, методы и взаимосвязи между ними. Входит в UML.

Существует два вида:

- статический вид диаграммы рассматривает логические взаимосвязи классов между собой;
- аналитический вид диаграммы рассматривает общий вид и взаимосвязи классов, входящих в систему.

Существуют разные точки зрения на построение диаграмм классов в зависимости от целей их применения:

- концептуальная точка зрения – диаграмма классов описывает модель предметной

области, в ней присутствуют только классы прикладных объектов;
точка зрения спецификации – диаграмма классов применяется при проектировании информационных систем;
точка зрения реализации – диаграмма классов содержит классы, используемые непосредственно в программном коде (при использовании объектно-ориентированных языков программирования).

Диаграмма классов является ключевым элементом в объектно-ориентированном моделировании. На диаграмме классы представлены в рамках, содержащих три компонента:

в верхней части написано имя класса. Имя класса выравнивается по центру и пишется полужирным шрифтом. Имена классов начинаются с заглавной буквы. Если класс абстрактный – то его имя пишется полужирным курсивом.

посередине располагаются поля (атрибуты) класса. Они выровнены по левому краю и начинаются с маленькой буквы.

нижняя часть содержит методы класса. Они также выровнены по левому краю и пишутся с маленькой буквы.

Диаграмма классов представлена на рисунке 7.1.

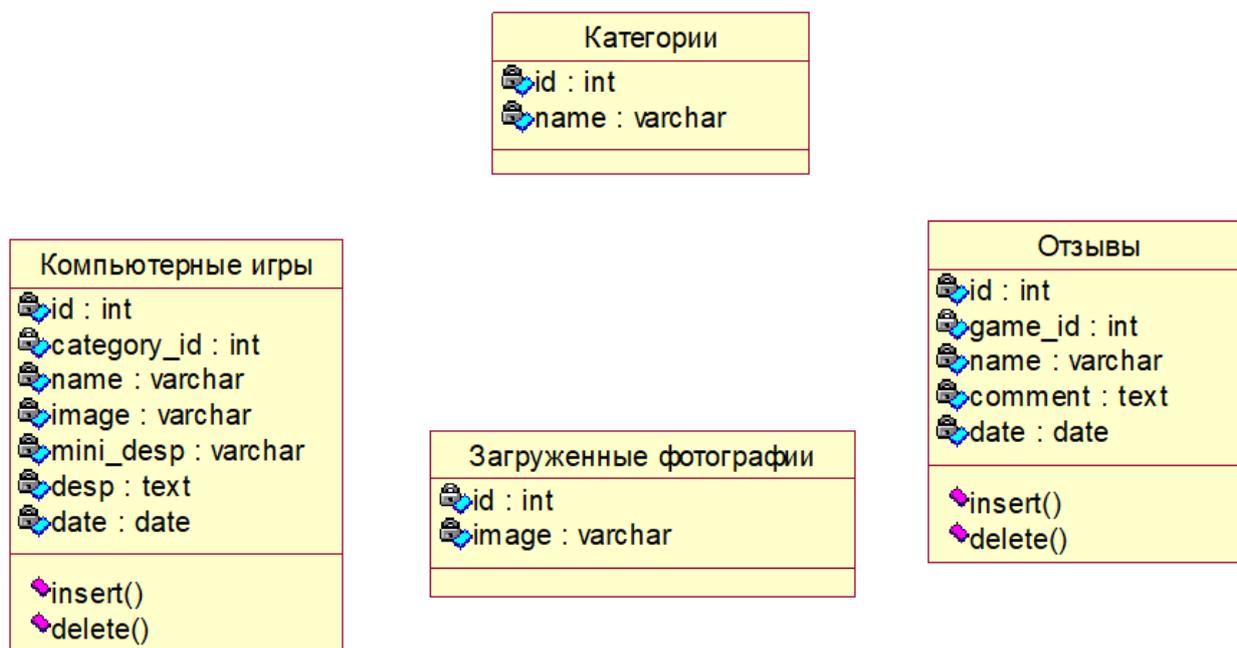


Рисунок 7.1 – Диаграмма классов

На рисунке 7.1 представлена диаграмма классов электронной энциклопедии компьютерных игр. Пользователю сайта для начала необходимо авторизоваться или зарегистрироваться для приобретения товара. Для поиска нужного товара пользователь может воспользоваться фильтрами или полем для поиска. После того как пользователь определился с товаром,

ознакомился с его описанием, ценой и просмотрев фотогалерею к товару он может добавить товар в корзину где, будет производиться оформление заказа. В корзине заказов будут отображены все товары, которые хочет приобрести пользователь и их общую сумму. Для приобретения необходимо оставить контактные данные и оплатить товар любым доступным ему способом.

8 Руководство по развёртыванию системы

Для установки системы на рабочем ПК необходимо иметь локальный веб-сервер для Windows (Open Server Panel), спроектированную базу данных и исходный код проекта. Панель Open Server представлена на рисунке 8.1.

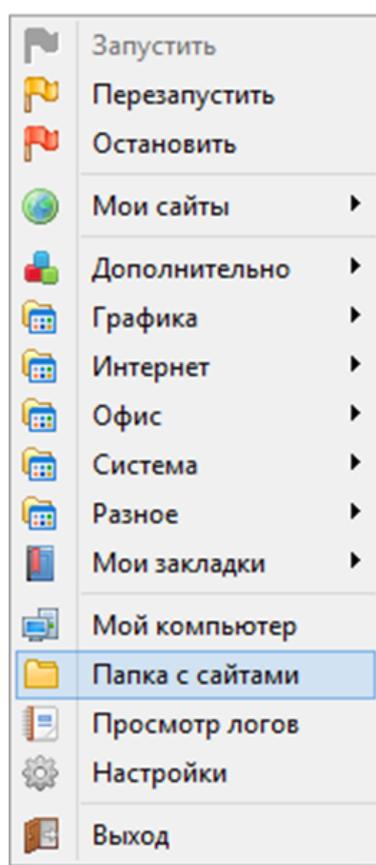


Рисунок 8.1 - Панель

Для начала необходимо запустить локальный сервер, после перейти на вкладку «Дополнительно» - «RНРМуAdmin». Перед программистом в окне браузера откроется страница с авторизацией (рисунок 8.2).

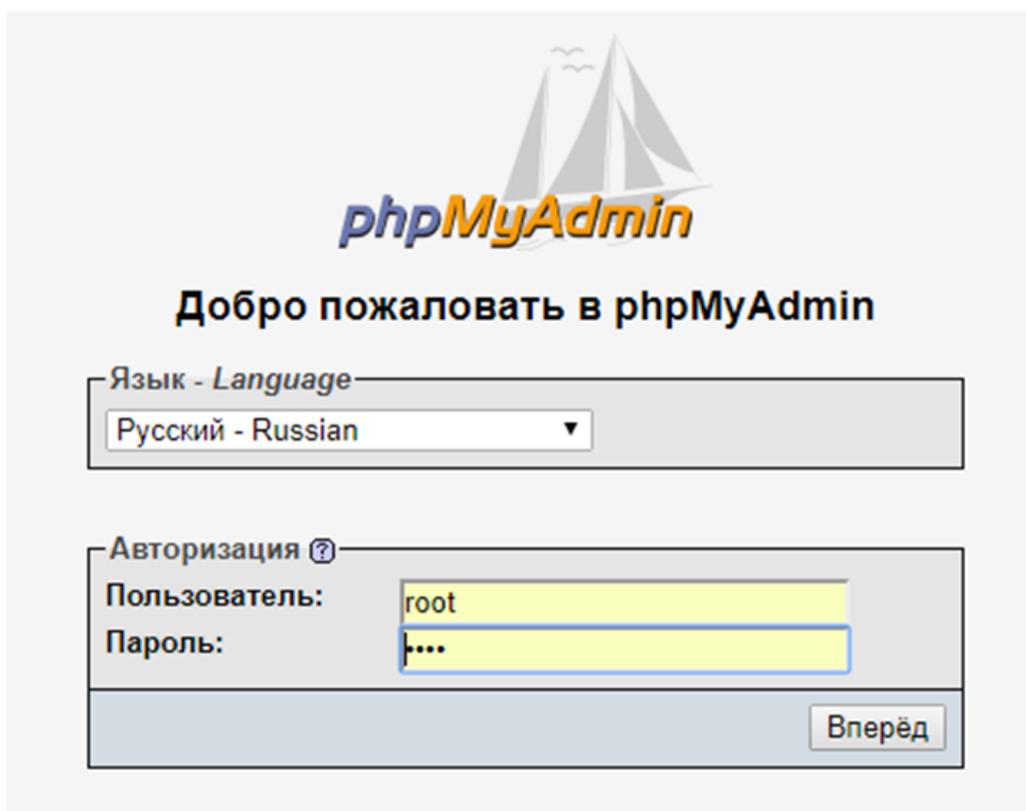


Рисунок 8.2 - Авторизация РНРMyAdmin

Далее система перенаправит программиста на главную страницу СУБД для дальнейшего импорта спроектированной базы данных. Главная страница представлена на рисунке 8.3.

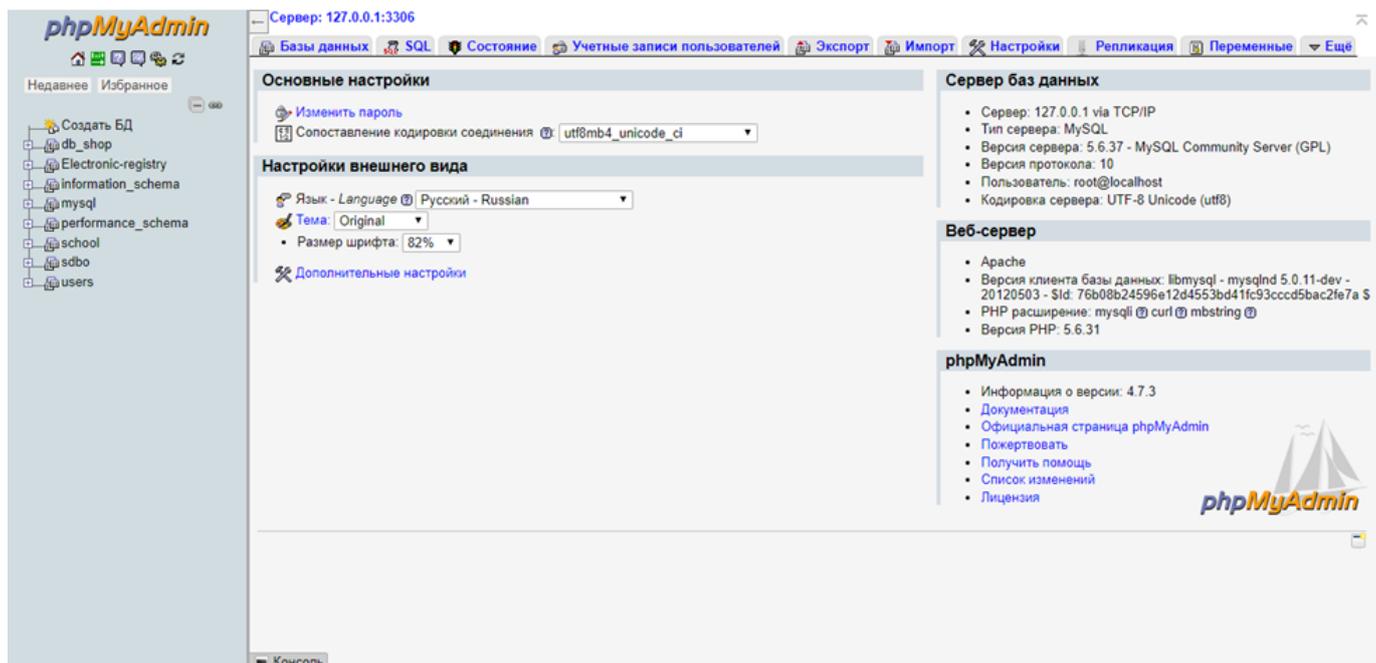


Рисунок 8.3 – Главная страница СУБД

С главной страницы программист создает новую базу данных с названием «pc-games» и кодировкой UTF-8. Далее внутри базы нажимает импорт, выбирает файл *.sql спроектированной базы данных и нажимает кнопку «Вперед». Результат импорта данных представлен на рисунке 8.4.



Рисунок 8.4 - Импорт базы данных

Проект с исходным кодом необходимо поместить в корень локального сервера. После в папку «domains». Перезапустить сервер и выбрать вкладку «Мои сайты» где будет отображаться сайт. Результаты данной процедуры представлены на рисунке 8.5.

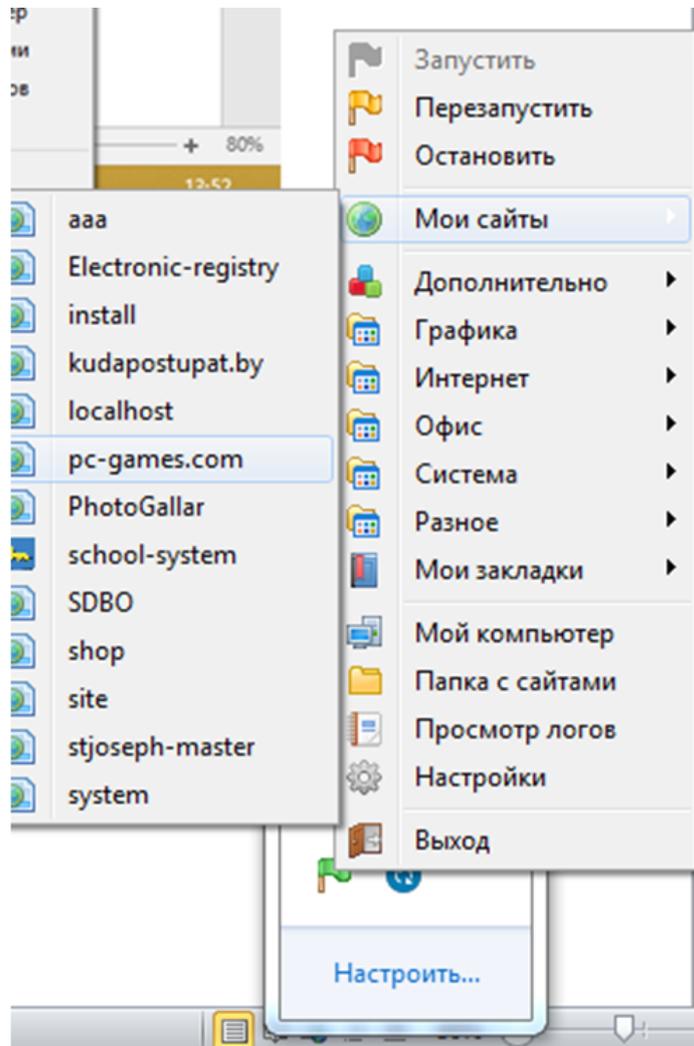


Рисунок 8.5 – Запуск системы

Перед программистом откроется браузер, взаимодействующий с локальным сервером и страница авторизации, которая была описана в предыдущем разделе.

9 Результаты тестирования разработанной системы, и оценка выполнения задач

После запуска исполняемого файла перед пользователем открывается главная страница сайта, главная страница сайта при входе пользователя приведена на рисунке 9.1.

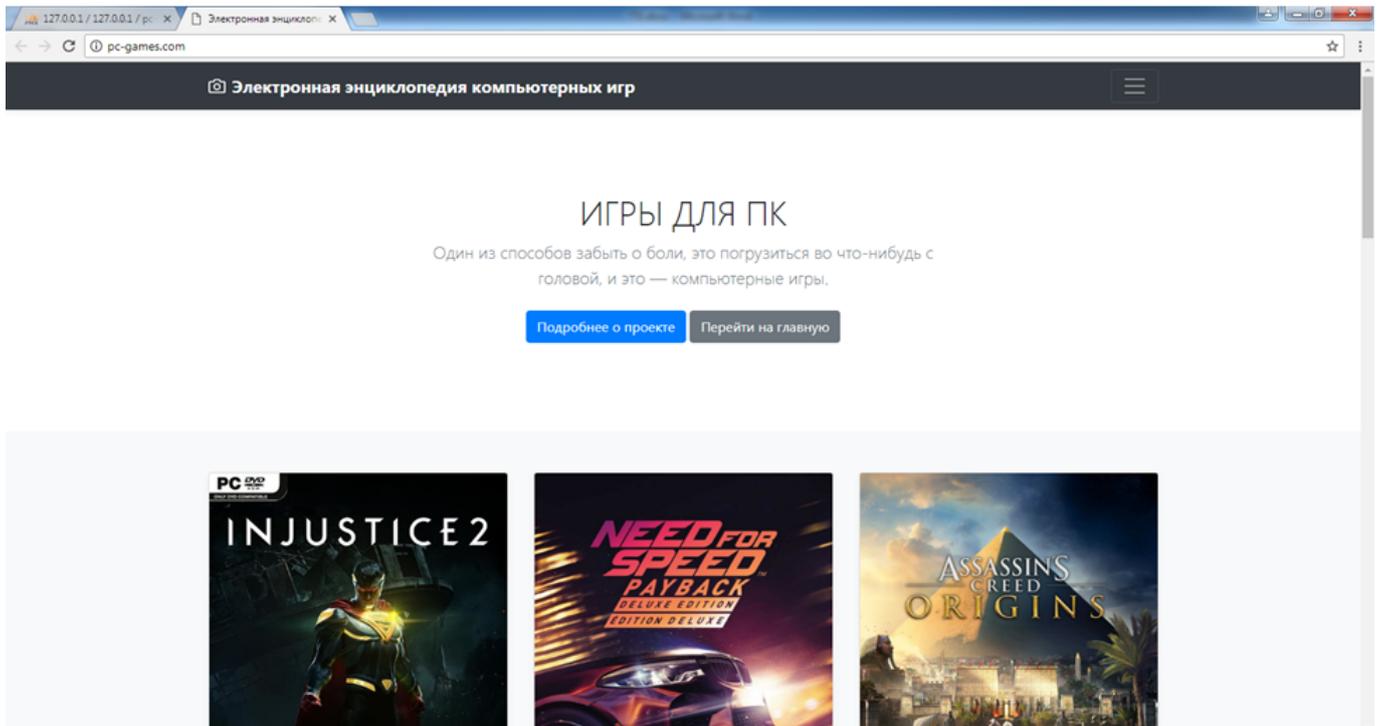


Рисунок 9.1 – Главная страница электронной энциклопедии компьютерных игр

С главной страницы можно просмотреть справочной информации о проекте, которая представлена на рисунке 9.2. Также можно применить фильтры к списку компьютерных игр, результат представлен на рисунке 9.3.

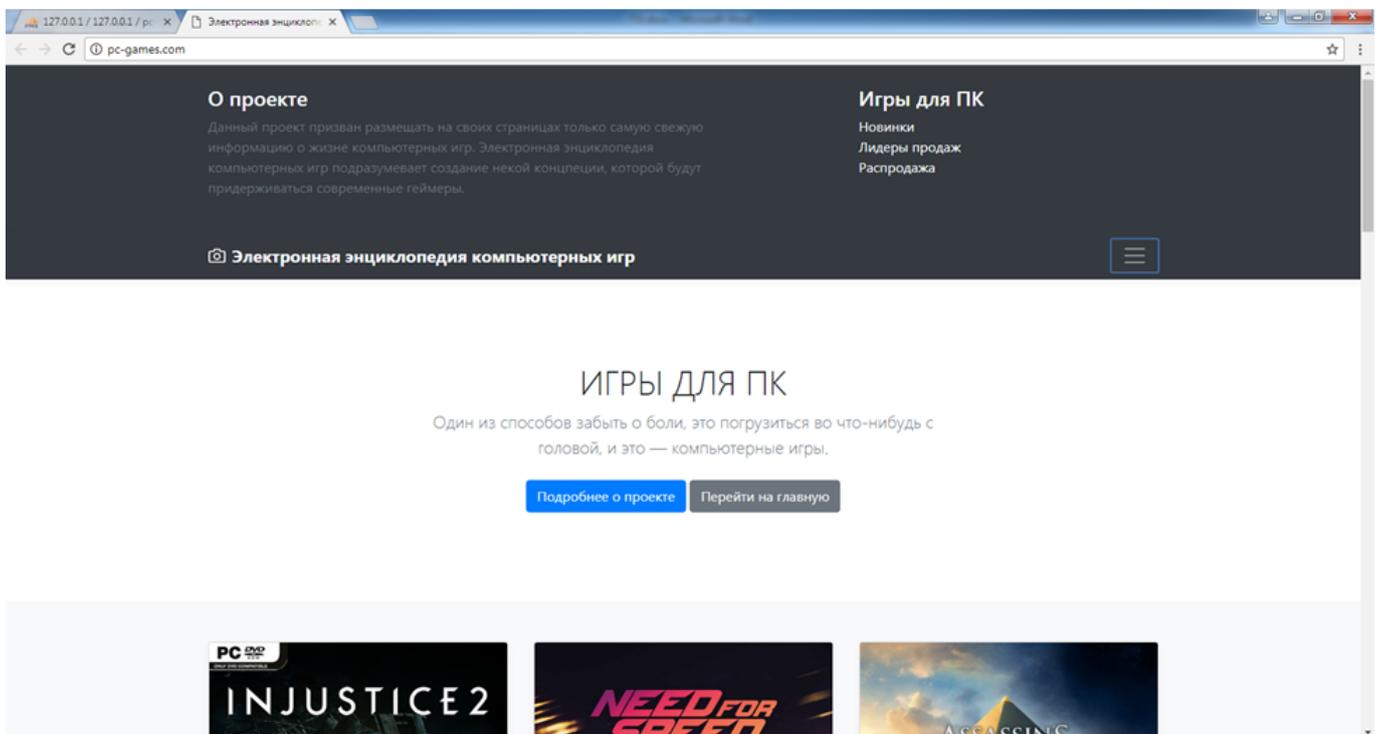


Рисунок 9.2 – Информация о проекте

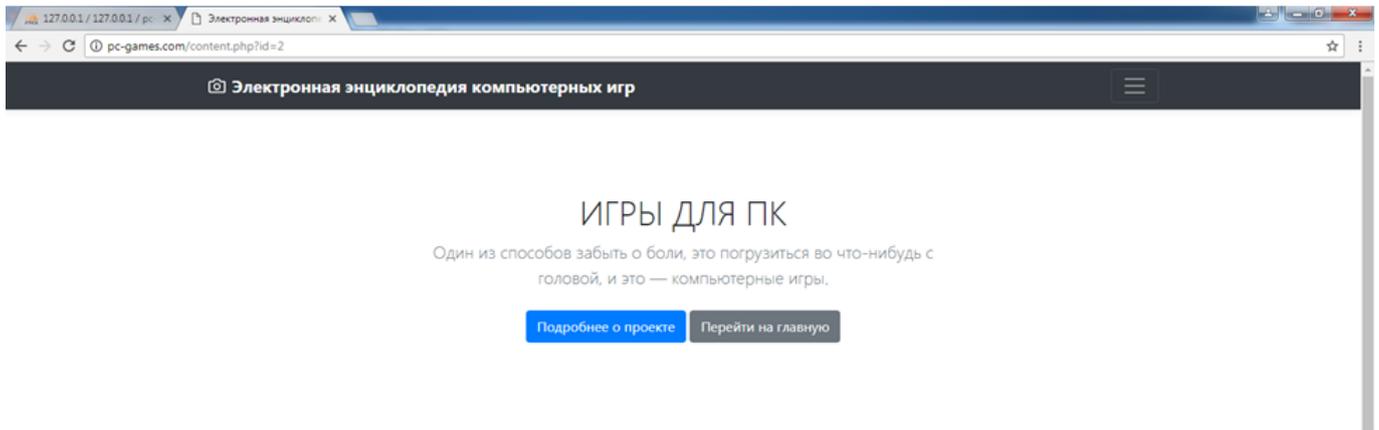


Рисунок 9.3 – Результаты применения фильтров

Существующие игры можно просмотреть подробно. Для этого необходимо нажать кнопку «Подробнее» и пользователь будет перенаправлен на страницу с подобной информацией о компьютерной игре. Результат такого действия представлен на рисунке 9.4.

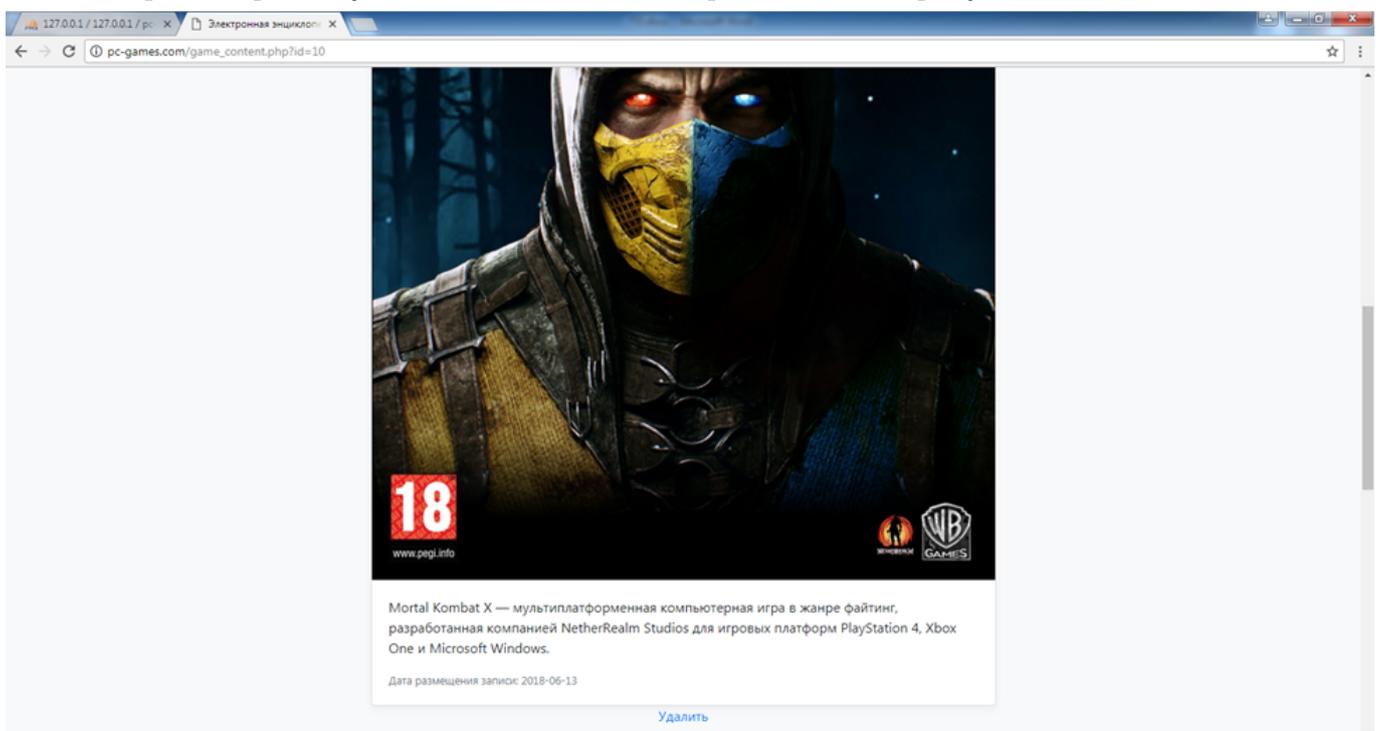


Рисунок 9.4 – Подробная информация о компьютерной игре

Также приложение предусмотрена функция добавления новой игры в энциклопедию с полным её описанием и всеми возможностями. Результат данного действия представлен на рисунке 9.5.

The screenshot shows a web browser window with the URL `pc-games.com/index.php`. The page title is "Добавление новой компьютерной игры". The form contains the following fields and elements:

- Статус игры:** A dropdown menu with "Новинки" selected.
- Название:** A text input field with "Заголовок" as a placeholder.
- Обложка:** A file upload area with a "Выберите файл" button and the text "Файл не выбран".
- Краткое описание:** A text area with "Краткое описание игры" as a placeholder.
- Полное описание:** A text area with "Полное описание игры" as a placeholder.

Рисунок 9.5 – Результат формы для добавления игры

Также предусмотрена возможность удаления игры, добавления отзыва и удаления отзыва. Результат формы для добавления отзыва представлен на рисунке 9.6.

The screenshot shows a web browser window with the URL `pc-games.com/game_content.php?id=12`. The page title is "Отзывы". The form contains the following elements:

- A box displaying "Дата размещения записок: 2018-06-13" and a blue "Удалить" button.
- Имя:** A text input field with "Имя" as a placeholder.
- Комментарий:** A text area with "Комментарий" as a placeholder.
- A yellow "Отправить" button.
- Footer text: "2018 © Лучшие компьютерные игры. Электронная энциклопедия компьютерных игр." and a blue "Вернуться вверх" link.

Рисунок 9.6 – Результат формы для добавления отзыва

Как мы видим на рисунке 9.6, перед формой добавления отзыва доступна ссылка, по нажатию на которую компьютерная игра будет удалена с энциклопедии. Аналогично с

удалением отзывов к игре. Список отзывов к одной из игр представлен на рисунке 9.7.

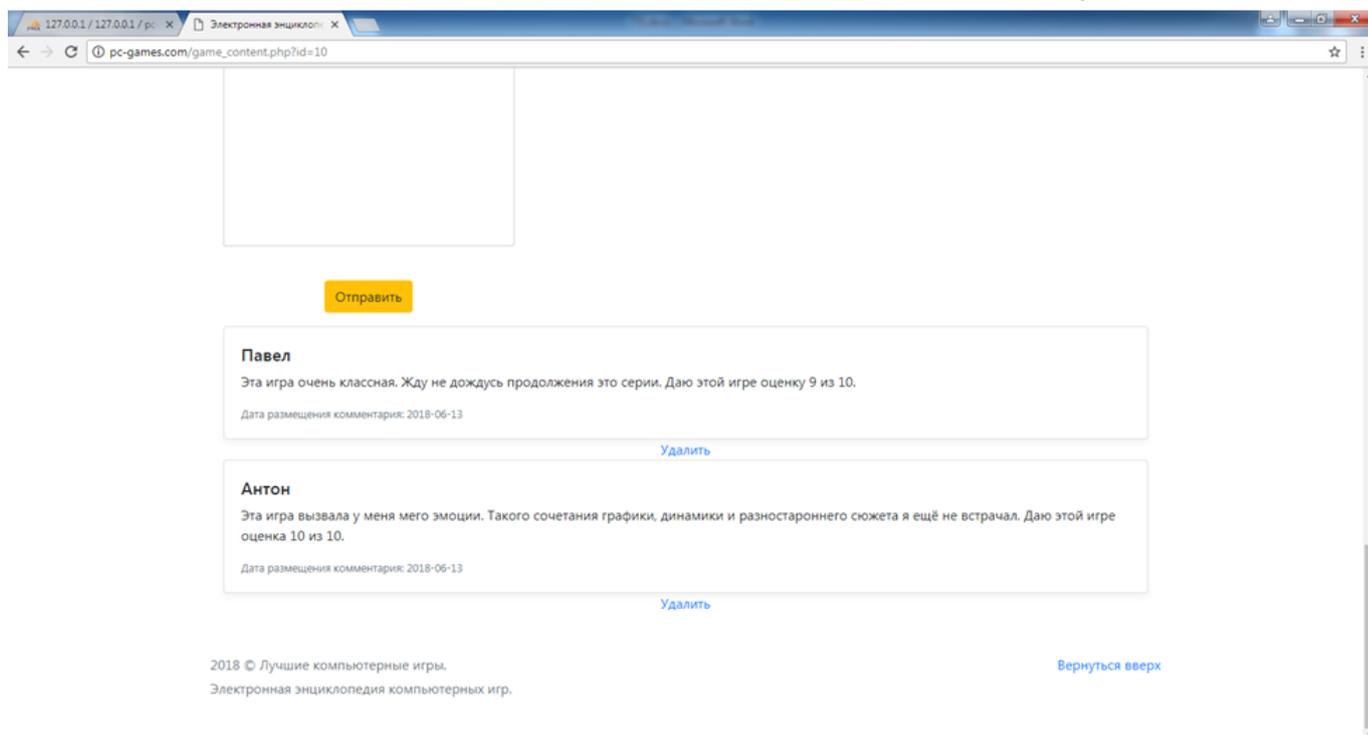


Рисунок 9.7 – Список отзывов

Таким образом, была разработана электронная энциклопедия компьютерных игр, позволяющая отображать самую актуальную информацию в жизни компьютерных технологий. Разработанным веб-приложением предусмотрены добавления/удаления игры, добавления/удаления отзыва, применение фильтров, отображение справочной информации и многое другое.

10 Приложение А. Листинг программы

```
<?php
if ($_POST["submit"])
{
include("/include/db_connect.php");
mysql_query("INSERT INTO list_games(category_id, name, image, mini_desp, desp, date)
VALUES(
'".$_POST["category"]."',
'".$_POST["name"]."',
'".$_POST["upload_image"]."',
'".$_POST["mini_desp"]."',
'".$_POST["desp"]."',
NOW()
)");
$id = mysql_insert_id();
if (empty($_POST["upload_image"]))
```

```
{
    include("/include/upload-image.php");
    unset($_POST["upload_image"]);
}
mysql_close();
header('Location: index.php');
exit;
}
?>
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="icon" href="">

    <title>Электронная энциклопедия компьютерных игр</title>

    <!-- Bootstrap core CSS -->
    <link href="/dist/css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom styles for this template -->
    <link href="/dist/css/album.css" rel="stylesheet">

    <style type="text/css">
        .link_game {
            text-decoration: none;
            color: #6c757d;
        }

        .link_game:hover {
            text-decoration: none;
            color: #ffffff;
        }
    </style>

</head>
```

```

<body>

<header>
  <div class="collapse bg-dark" id="navbarHeader">
    <div class="container">
      <div class="row">
        <div class="col-sm-8 col-md-7 py-4">
          <h4 class="text-white">О проекте</h4>
          <p class="text-muted">Данный проект призван размещать на своих страницах
только самую свежую информацию о жизни компьютерных игр. Электронная энциклопедия
компьютерных игр подразумевает создание некой концепции, которой будут придерживаться
современные геймеры.</p>
        </div>
        <div class="col-sm-4 offset-md-1 py-4">
          <h4 class="text-white">Игры для ПК</h4>
          <ul class="list-unstyled">

            <?php

                $link=mysqli_connect("localhost", "root", "root", "pc-games");
                $result = mysqli_query($link,"SELECT id, name FROM category");

                If (mysqli_num_rows($result) > 0)
                {
                    $row = mysqli_fetch_array($result);
                    do
                    {
                        echo '
                            <li><a href="content.php?id='.$row["id"].'" class="text-
white">'.$row["name"].'</a></li>
                        ';
                    } while ($row = mysqli_fetch_array($result));
                    echo'
                </ul>
                ';
                }

            ?>

          </ul>
        </div>
      </div>
    </div>
  </div>

```

```

        </div>
    </div>
</div>
<div class="navbar navbar-dark bg-dark box-shadow">
    <div class="container d-flex justify-content-between">
        <a href="#" class="navbar-brand d-flex align-items-center">
            <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24
24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="mr-2"><path d="M23 19a2 2 0 0 1-2 2H3a2 2 0 0 1-2-2V8a2 2 0 0 1
2-2h4l2-3h6l2 3h4a2 2 0 0 1 2 2z"></path><circle cx="12" cy="13" r="4"></circle></svg>
            <strong>Электронная энциклопедия компьютерных игр</strong>
        </a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarHeader" aria-controls="navbarHeader" aria-expanded="false" aria-label="Toggle
navigation">
            <span class="navbar-toggler-icon"></span>
        </button>
    </div>
</div>
</header>

<main role="main">

    <section class="jumbotron text-center">
        <div class="container">
            <h1 class="jumbotron-heading">ИГРЫ ДЛЯ ПК</h1>
            <p class="lead text-muted">Один из способов забыть о боли, это погрузиться во что-
нибудь с головой, и это — компьютерные игры.</p>
            <p>
                <a href="info.php" class="btn btn-primary my-2">Подробнее о проекте</a>
                <a href="index.php" class="btn btn-secondary my-2">Перейти на главную</a>
            </p>
        </div>
    </section>

    <div class="album py-5 bg-light">
        <div class="container">

            <div class="row">

```

```

<?php

$link=mysqli_connect("localhost", "root", "root", "pc-games");
$result = mysqli_query($link,"SELECT * FROM list_games ORDER BY id DESC");

If (mysqli_num_rows($result) > 0)
{
$row = mysqli_fetch_array($result);

do
{
if (strlen($row["image"]) > 0 && file_exists("./uploads_images/".$row["image"]))
{
$img_path = './uploads_images/'.$row["image"];
$max_width = 500;
$max_height = 300;
list($width, $height) = getimagesize($img_path);
$ratiow = $max_width/$width;
$ratioh = $max_height/$height;
$ratio = min($ratioh, $ratiow);
// New dimensions
$width = intval($ratio*$width);
$height = intval($ratio*$height);
}else
{
$img_path = "";
$width = auto;
$height = auto;
}

echo '

<div class="col-md-4">
<div class="card mb-4 box-shadow">

<div class="card-body">
<p class="card-text">'.$row["mini_desp"].'</p>
<div class="d-flex justify-content-between align-items-center">

```

```
        <div class="btn-group">
            <button type="button" class="btn btn-sm btn-outline-secondary"><a
class="link_game" href="game_content.php?id='.$row["id"].'">Подробнее</a></button>
            <button type="button" class="btn btn-sm btn-outline-secondary"><a
class="link_game" href="game_content.php?id='.$row["id"].'">Отзывы</a></button>
        </div>
        <small class="text-muted">'.$row["date"].'</small>
    </div>
</div>
</div>
</div>
</div>
';

    } while ($row = mysqli_fetch_array($result));
    echo'
</ul>
';
}

?>

</div>
</div>
</div>
```

```
<div class="container containerform">
```

```
<form class="well form-horizontal" enctype="multipart/form-data" method="post">
<fieldset>
```

```
<!-- Form Name -->
```

```
<legend class="text-center">Добавление новой компьютерной игры</legend>
```

```
<!-- Select input-->
```

```
<div class="form-group">
```

```
<label class="col-md-4 control-label">Статус игры</label>
```

```
<div class="col-md-4 inputGroupContainer">
```

```

<div class="input-group">
  <?php
  include('/include/db_connect.php');
  $query = 'SELECT * FROM category';
  $result = mysql_query ( $query ) or die ( 'Query failed: ' . mysql_error ( ) );
  echo "<select class='form-control' name=category>";

  while ( $line = mysql_fetch_array ( $result, MYSQL_ASSOC ) ) {
    echo '<option value="" . $line [ 'id' ] . "'>' . $line [ 'name' ] . "'>';
  }
  echo "</select>";
  ?>
</div>
</div>
</div>

```

```

<!-- Text input-->
<div class="form-group">
  <label class="col-md-4 control-label">Название</label>
  <div class="col-md-4 inputGroupContainer">
    <div class="input-group">
      <span class="input-group-addon"><i class="glyphicon glyphicon-
pencil"></i></span>
      <input name="name" placeholder="Заголовок" class="form-control"
type="text">
    </div>
  </div>
</div>

```

```

<!-- Text input-->
<div class="form-group">
  <label class="col-md-4 control-label" >Обложка</label>
  <div class="col-md-4 inputGroupContainer">
    <div class="input-group">
      <span class="input-group-addon"><i class="glyphicon glyphicon-
picture"></i></span>
      <input type="hidden" name="MAX_FILE_SIZE" class="form-control"
value="5000000"/>
      <input type="file" name="upload_image" class="form-control"/>
    </div>
  </div>

```

```

</div>

<!-- Text area -->
<div class="form-group">
  <label class="col-md-4 control-label">Краткое описание</label>
  <div class="col-md-4 inputGroupContainer">
    <div class="input-group">
      <span class="input-group-addon"><i class="glyphicon glyphicon-
pencil"></i></span>
      <textarea class="form-control" name="mini_desp" placeholder="Краткое
описание игры" rows="10" cols="45" style="resize: none;"></textarea>
    </div>
  </div>
</div>

<!-- Text area -->
<div class="form-group">
  <label class="col-md-4 control-label">Полное описание</label>
  <div class="col-md-4 inputGroupContainer">
    <div class="input-group">
      <span class="input-group-addon"><i class="glyphicon glyphicon-
pencil"></i></span>
      <textarea class="form-control" name="desp" placeholder="Полное описание
игры" rows="10" cols="45" style="resize: none;"></textarea>
    </div>
  </div>
</div>

<!-- Button -->
<div class="form-group">
  <label class="col-md-4 control-label"></label>
  <div class="col-md-4 text-center">
    <input type="submit" class="btn btn-warning" id="submit" name="submit"
value="Добавить"/>
  </div>
</div>

</fieldset>
</form>
</div>

```

```
</main>
```

```
<footer class="text-muted">
```

```
<div class="container">
```

```
<p class="float-right">
```

```
<a href="#">Вернуться вверх</a>
```

```
</p>
```

```
<p>2018 &copy; Лучские компьютерные игры.</p>
```

```
<p>Электронная энциклопедия компьютерных игр.</p>
```

```
</div>
```

```
</footer>
```

```
<!-- Bootstrap core JavaScript
```

```
===== -->
```

```
<!-- Placed at the end of the document so the pages load faster -->
```

```
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KcRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
```

```
<script>window.jQuery || document.write('<script src="/assets/js/vendor/jquery-
slim.min.js"></script>')</script>
```

```
<script src="/assets/js/vendor/popper.min.js"></script>
```

```
<script src="/dist/js/bootstrap.min.js"></script>
```

```
<script src="/assets/js/vendor/holder.min.js"></script>
```

```
</body>
```

```
</html>
```

```
<?php
```

```
if ($_POST["submit"])
```

```
{
```

```
include("/include/db_connect.php");
```

```
mysql_query("INSERT INTO list_games(category_id, name, image, mini_desp, desp, date)
```

```
VALUES(
```

```
    "$_POST["category"].",
```

```
    "$_POST["name"].",
```

```
    "$_POST["upload_image"].",
```

```
    "$_POST["mini_desp"].",
```

```
    "$_POST["desp"].",
```

```
    NOW()
```

```
    )", $link);
```

```
$id = mysql_insert_id();
```

```
if (empty($_POST["upload_image"]))
{
    include("/include/upload-image.php");
    unset($_POST["upload_image"]);
}
mysql_close();
header('Location: index.php');
exit;
}
?>
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <meta name="description" content="">
    <meta name="author" content="">
    <link rel="icon" href="">

    <title>Электронная энциклопедия компьютерных игр</title>

    <!-- Bootstrap core CSS -->
    <link href="/dist/css/bootstrap.min.css" rel="stylesheet">

    <!-- Custom styles for this template -->
    <link href="/dist/css/album.css" rel="stylesheet">

    <style type="text/css">
        .link_game {
            text-decoration: none;
            color: #6c757d;
        }

        .link_game:hover {
            text-decoration: none;
            color: #ffffff;
        }
    </style>

</head>
```

```
<body>
```

```
<header>
```

```
<div class="collapse bg-dark" id="navbarHeader">
```

```
<div class="container">
```

```
<div class="row">
```

```
<div class="col-sm-8 col-md-7 py-4">
```

```
<h4 class="text-white">О проекте</h4>
```

```
<p class="text-muted">Данный проект призван размещать на своих страницах  
только самую свежую информацию о жизни компьютерных игр. Электронная энциклопедия  
компьютерных игр подразумевает создание некой концепции, которой будут придерживаться  
современные геймеры.</p>
```

```
</div>
```

```
<div class="col-sm-4 offset-md-1 py-4">
```

```
<h4 class="text-white">Игры для ПК</h4>
```

```
<ul class="list-unstyled">
```

```
<?php
```

```
$link=mysqli_connect("localhost", "root", "root", "pc-games");
```

```
$result = mysqli_query($link,"SELECT id, name FROM category");
```

```
If (mysqli_num_rows($result) > 0)
```

```
{
```

```
    $row = mysqli_fetch_array($result);
```

```
    do
```

```
    {
```

```
        echo '
```

```
            <li><a href="content.php?id='.$row["id"].'" class="text-  
white">'.$row["name"].'</a></li>
```

```
        ';
```

```
    } while ($row = mysqli_fetch_array($result));
```

```
    echo'
```

```
</ul>
```

```
!;
```

```
}
```

```
?>
```

```
</ul>
```

```
        </div>
    </div>
</div>
</div>
<div class="navbar navbar-dark bg-dark box-shadow">
    <div class="container d-flex justify-content-between">
        <a href="#" class="navbar-brand d-flex align-items-center">
            <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24
24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="mr-2"><path d="M23 19a2 2 0 0 1-2 2H3a2 2 0 0 1-2-2V8a2 2 0 0 1
2-2h4l2-3h6l2 3h4a2 2 0 0 1 2 2z"></path><circle cx="12" cy="13" r="4"></circle></svg>
            <strong>Электронная энциклопедия компьютерных игр</strong>
        </a>
        <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarHeader" aria-controls="navbarHeader" aria-expanded="false" aria-label="Toggle
navigation">
            <span class="navbar-toggler-icon"></span>
        </button>
    </div>
</div>
</header>

<main role="main">

<section class="jumbotron text-center">
    <div class="container">
        <h1 class="jumbotron-heading">ИГРЫ ДЛЯ ПК</h1>
        <p class="lead text-muted">Один из способов забыть о боли, это погрузиться во что-
нибудь с головой, и это — компьютерные игры.</p>
        <p>
            <a href="#" class="btn btn-primary my-2">Подробнее о проекте</a>
            <a href="index.php" class="btn btn-secondary my-2">Перейти на главную</a>
        </p>
    </div>
</section>

<div class="album py-5 bg-light">
    <div class="container">

        <div class="row">
```

<p style="text-align: justify">

PC-Games — это проект о компьютерных играх. Сайт предлагает своим посетителям самые свежие новости игрового мира, обзоры и анонсы популярных игр, подробную энциклопедию, охватывающую свыше 29 тысяч наименований, а также коды, видеоролики, спецматериалы и многое другое.

</p>

<p style="text-align: justify">

Увлекаюсь играми с 2015 года, когда получила в подарок Dendy. Убежденный программист.

Стараюсь уделять внимание всем AAA-новинкам, но особый трепет испытываю перед спортсимами, онлайн-шутерами и иммерсив РПГ. Также не дурак в тактику и стратегии, начиная от Героев, Эпохи Империй и Цивы, и заканчивая ХСОМ и всякими ККИ.

Игрой 2016 года считаю Firewatch.

</p>

<p style="text-align: justify">

РЕДАКТОР - СТУДЕНТ БГУИР

Недобой Анастасия Александровна

</p>

</div>

</div>

</div>

</main>

<footer class="text-muted">

<div class="container">

<p class="float-right">

Вернуться вверх

</p>

<p>2018 © Лучшие компьютерные игры.</p>

<p>Электронная энциклопедия компьютерных игр.</p>

</div>

</footer>

```
<!-- Bootstrap core JavaScript
===== -->
<!-- Placed at the end of the document so the pages load faster -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkVYIK3UENzmM7KcRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
<script>window.jQuery || document.write('<script src="/assets/js/vendor/jquery-
slim.min.js"></script>')</script>
<script src="/assets/js/vendor/popper.min.js"></script>
<script src="/dist/js/bootstrap.min.js"></script>
<script src="/assets/js/vendor/holder.min.js"></script>
</body>
</html>

<?php
$id = $_GET['id'];
?>
<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">
<link rel="icon" href="">

<title>Электронная энциклопедия компьютерных игр</title>

<!-- Bootstrap core CSS -->
<link href="/dist/css/bootstrap.min.css" rel="stylesheet">

<!-- Custom styles for this template -->
<link href="/dist/css/album.css" rel="stylesheet">

<style type="text/css">
.link_game {
text-decoration: none;
color: #6c757d;
}
}
```

```

        .link_game:hover {
            text-decoration: none;
            color: #ffffff;
        }
    </style>

</head>

<body>

<header>
    <div class="collapse bg-dark" id="navbarHeader">
        <div class="container">
            <div class="row">
                <div class="col-sm-8 col-md-7 py-4">
                    <h4 class="text-white">О проекте</h4>
                    <p class="text-muted">Данный проект призван размещать на своих страницах
только самую свежую информацию о жизни компьютерных игр. Электронная энциклопедия
компьютерных игр подразумевает создание некой концепции, которой будут придерживаться
современные геймеры.</p>
                </div>
                <div class="col-sm-4 offset-md-1 py-4">
                    <h4 class="text-white">Игры для ПК</h4>
                    <ul class="list-unstyled">

                        <?php

                            $link=mysqli_connect("localhost", "root", "root", "pc-games");
                            $result = mysqli_query($link,"SELECT id, name FROM category");

                            If (mysqli_num_rows($result) > 0)
                            {
                                $row = mysqli_fetch_array($result);
                                do
                                {
                                    echo '
                                        <li><a href="content.php?id='.$row["id"].'" class="text-
white">'.$row["name"].'</a></li>
                                    ';
                                } while ($row = mysqli_fetch_array($result));
                                echo'

```

```
</ul>
';
}

?>

</ul>
</div>
</div>
</div>
</div>
<div class="navbar navbar-dark bg-dark box-shadow">
  <div class="container d-flex justify-content-between">
    <a href="#" class="navbar-brand d-flex align-items-center">
      <svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24
24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-
linejoin="round" class="mr-2"><path d="M23 19a2 2 0 0 1-2 2H3a2 2 0 0 1-2 2V8a2 2 0 0 1
2-2h4l2-3h6l2 3h4a2 2 0 0 1 2 2z"></path><circle cx="12" cy="13" r="4"></circle></svg>
      <strong>Электронная энциклопедия компьютерных игр</strong>
    </a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarHeader" aria-controls="navbarHeader" aria-expanded="false" aria-label="Toggle
navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
  </div>
</div>
</header>

<main role="main">

<section class="jumbotron text-center">
  <div class="container">
    <h1 class="jumbotron-heading">ИГРЫ ДЛЯ ПК</h1>
    <p class="lead text-muted">Один из способов забыть о боли, это погрузиться во что-
нибудь с головой, и это — компьютерные игры.</p>
    <p>
      <a href="info.php" class="btn btn-primary my-2">Подробнее о проекте</a>
      <a href="index.php" class="btn btn-secondary my-2">Перейти на главную</a>
    </p>
  </div>
</section>
</main>
```

```
</section>
```

```
<div class="album py-5 bg-light">
```

```
<div class="container">
```

```
<div class="row">
```

```
<?php
```

```
$link=mysqli_connect("localhost", "root", "root", "pc-games");
```

```
$result = mysqli_query($link,"SELECT * FROM list_games WHERE category_id = $id  
ORDER BY id DESC");
```

```
If (mysqli_num_rows($result) > 0)
```

```
{
```

```
    $row = mysqli_fetch_array($result);
```

```
do
```

```
{
```

```
    if (strlen($row["image"]) > 0 && file_exists("./uploads_images/".$row["image"]))
```

```
    {
```

```
        $img_path = './uploads_images/'.$row["image"];
```

```
        $max_width = 500;
```

```
        $max_height = 300;
```

```
        list($width, $height) = getimagesize($img_path);
```

```
        $ratioh = $max_height/$height;
```

```
        $ratiow = $max_width/$width;
```

```
        $ratio = min($ratioh, $ratiow);
```

```
        // New dimensions
```

```
        $width = intval($ratio*$width);
```

```
        $height = intval($ratio*$height);
```

```
    }else
```

```
    {
```

```
        $img_path = "";
```

```
        $width = auto;
```

```
        $height = auto;
```

```
    }
```

```
echo '
```

```
<div class="col-md-4">
  <div class="card mb-4 box-shadow">
    <img class="card-img-top" src="".$img_path.'" data-src="" alt="">
    <div class="card-body">
      <p class="card-text">'.$row["mini_desp"].'</p>
      <div class="d-flex justify-content-between align-items-center">
        <div class="btn-group">
          <button type="button" class="btn btn-sm btn-outline-secondary"><a
class="link_game" href="game_content.php?id='.$row["id"].'">Подробнее</a></button>
          <button type="button" class="btn btn-sm btn-outline-
secondary">Отзывы</button>
        </div>
        <small class="text-muted">'.$row["date"].'</small>
      </div>
    </div>
  </div>
</div>
';

} while ($row = mysqli_fetch_array($result));
echo'
</ul>
';
}

?>

</div>
</div>
</div>

</main>

<footer class="text-muted">
  <div class="container">
```

```
<p class="float-right">
  <a href="#">Вернуться вверх</a>
</p>
<p>2018 &copy; Лучишие компьютерные игры.</p>
<p>Электронная энциклопедия компьютерных игр.</p>
</div>
</footer>
```

```
<!-- Bootstrap core JavaScript
===== -->
<!-- Placed at the end of the document so the pages load faster -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KcRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
<script>window.jQuery || document.write('<script src="/assets/js/vendor/jquery-
slim.min.js"></script>')</script>
<script src="/assets/js/vendor/popper.min.js"></script>
<script src="/dist/js/bootstrap.min.js"></script>
<script src="/assets/js/vendor/holder.min.js"></script>
</body>
</html>
```

```
<?php
$id = $_GET['id'];
?>
<?php
if ($_POST["submit"])
{
include("/include/db_connect.php");
mysql_query("INSERT INTO reviews_game(game_id, name, comment, date)
VALUES(
'".$_GET['id']."',
'".$_POST["name"]."',
'".$_POST["comment"]."',
NOW()
)",$link);
mysql_close();
header('Location: game_content.php?id='.$_GET['id'].');
exit;
}
?>
```

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
  <meta name="description" content="">
  <meta name="author" content="">
  <link rel="icon" href="">

  <title>Электронная энциклопедия компьютерных игр</title>

  <!-- Bootstrap core CSS -->
  <link href="/dist/css/bootstrap.min.css" rel="stylesheet">

  <!-- Custom styles for this template -->
  <link href="/dist/css/album.css" rel="stylesheet">

  <style type="text/css">
    .link_game {
      text-decoration: none;
      color: #6c757d;
    }

    .link_game:hover {
      text-decoration: none;
      color: #ffffff;
    }
  </style>

</head>

<body>

<header>
  <div class="collapse bg-dark" id="navbarHeader">
    <div class="container">
      <div class="row">
        <div class="col-sm-8 col-md-7 py-4">
          <h4 class="text-white">О проекте</h4>
          <p class="text-muted">Данный проект призван размещать на своих страницах
```

только самую свежую информацию о жизни компьютерных игр. Электронная энциклопедия компьютерных игр подразумевает создание некой концепции, которой будут придерживаться современные геймеры.</p>

```
</div>
```

```
<div class="col-sm-4 offset-md-1 py-4">
```

```
<h4 class="text-white">Игры для ПК</h4>
```

```
<ul class="list-unstyled">
```

```
<?php
```

```
$link=mysqli_connect("localhost", "root", "root", "pc-games");
```

```
$result = mysqli_query($link,"SELECT id, name FROM category");
```

```
If (mysqli_num_rows($result) > 0)
```

```
{
```

```
    $row = mysqli_fetch_array($result);
```

```
    do
```

```
    {
```

```
        echo '
```

```
            <li><a href="content.php?id='.$row["id"].'" class="text-  
white">'.$row["name"].'</a></li>
```

```
        ';
```

```
    } while ($row = mysqli_fetch_array($result));
```

```
    echo'
```

```
</ul>
```

```
!;
```

```
}
```

```
?>
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="navbar navbar-dark bg-dark box-shadow">
```

```
<div class="container d-flex justify-content-between">
```

```
<a href="#" class="navbar-brand d-flex align-items-center">
```

```
<svg xmlns="http://www.w3.org/2000/svg" width="20" height="20" viewBox="0 0 24  
24" fill="none" stroke="currentColor" stroke-width="2" stroke-linecap="round" stroke-  
linejoin="round" class="mr-2"><path d="M23 19a2 2 0 0 1-2 2H3a2 2 0 0 1-2 2V8a2 2 0 0 1
```

```
2-2h4l2-3h6l2 3h4a2 2 0 0 1 2 2z"></path><circle cx="12" cy="13" r="4"></circle></svg>
    <strong>Электронная энциклопедия компьютерных игр</strong>
  </a>
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarHeader" aria-controls="navbarHeader" aria-expanded="false" aria-label="Toggle
navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
  </div>
</div>
</header>
```

```
<main role="main">
```

```
<section class="jumbotron text-center">
  <div class="container">
    <h1 class="jumbotron-heading">ИГРЫ ДЛЯ ПК</h1>
    <p class="lead text-muted">Один из способов забыть о боли, это погрузиться во что-
нибудь с головой, и это — компьютерные игры.</p>
    <p>
      <a href="info.php" class="btn btn-primary my-2">Подробнее о проекте</a>
      <a href="index.php" class="btn btn-secondary my-2">Перейти на главную</a>
    </p>
  </div>
</section>
```

```
<div class="album py-5 bg-light">
```

```
<div class="container">
```

```
<div class="row">
```

```
<?php
```

```
$link=mysqli_connect("localhost", "root", "root", "pc-games");
```

```
$result = mysqli_query($link,"SELECT * FROM list_games WHERE id = '$id' ORDER
BY id DESC");
```

```
If (mysqli_num_rows($result) > 0)
```

```
{
```

```

$row = mysqli_fetch_array($result);

do
{
    if (strlen($row["image"]) > 0 && file_exists("./uploads_images/".$row["image"]))
    {
        $img_path = './uploads_images/'.$row["image"];
        $max_width = 500;
        $max_height = 300;
        list($width, $height) = getimagesize($img_path);
        $ratioh = $max_height/$height;
        $ratiow = $max_width/$width;
        $ratio = min($ratioh, $ratiow);
        // New dimensions
        $width = intval($ratio*$width);
        $height = intval($ratio*$height);
    }else
    {
        $img_path = "";
        $width = auto;
        $height = auto;
    }

    echo '

```

```
<div class="col-md-2"></div>
```

```
<div class="col-md-8">
```

```
<center><h1 class="card-text">'.$row["name"].'</h1></center>
```

```
<div class="card mb-8 box-shadow">
```

```

```

```
<div class="card-body">
```

```
<p class="card-text">'.$row["desp"].'</p>
```

```
<div class="d-flex justify-content-between align-items-center">
```

```
<small class="text-muted">Дата размещения записи: '.$row["date"].'</small>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
<center><a href="include/delete-games.php?id='.$row["id"].'">Удалить</a></center>
```

```
</div>
```

```

<div class="col-md-2"></div>
';

} while ($row = mysqli_fetch_array($result));
echo'
</ul>
';
}

?>

</div>
</div>
</div>

<div class="container containerform">

<form class="well form-horizontal" enctype="multipart/form-data" method="post">
<fieldset>

<!-- Form Name -->
<legend class="text-center">Отзывы</legend>

<!-- Text input-->
<div class="form-group">
<label class="col-md-4 control-label">Имя</label>
<div class="col-md-4 inputGroupContainer">
<div class="input-group">
<span class="input-group-addon"><i class="glyphicon glyphicon-
pencil"></i></span>
<input name="name" placeholder="Имя" class="form-control" type="text">
</div>
</div>
</div>

<!-- Text area -->

```

```

    <div class="form-group">
        <label class="col-md-4 control-label">Комментарий</label>
        <div class="col-md-4 inputGroupContainer">
            <div class="input-group">
                <span class="input-group-addon"><i class="glyphicon glyphicon-
pencil"></i></span>
                <textarea class="form-control" name="comment" placeholder="Комментарий"
rows="10" cols="45" style="resize: none;"></textarea>
            </div>
        </div>
    </div>

    <!-- Button -->
    <div class="form-group">
        <label class="col-md-4 control-label"></label>
        <div class="col-md-4 text-center">
            <input type="submit" class="btn btn-warning" id="submit" name="submit"
value="Отправить"/>
        </div>
    </div>

</fieldset>
</form>

<?php

$link1=mysqli_connect("localhost", "root", "root", "pc-games");
$result1 = mysqli_query($link1,"SELECT * FROM reviews_game WHERE game_id = '$id'
ORDER BY id DESC");

If (mysqli_num_rows($result1) > 0)
{
    $row1 = mysqli_fetch_array($result1);

do
{
    echo '
    <div class="col-md-12">

```

```

        <div class="card mb-12 box-shadow">
            <div class="card-body">
                <h5 class="card-text">'. $row1["name"].'</h5>
                <p class="card-text">'. $row1["comment"].'</p>
                <div class="d-flex justify-content-between align-items-center">
                    <small class="text-muted">Дата размещения комментария:
'. $row1["date"].'</small>
                </div>
            </div>
        </div>
        <div class="text-align-right">
            <center><a href="include/delete-
reviews.php?id='.$row1["id"].'">Удалить</a></center>
        </div>
    ';

} while ($row1 = mysqli_fetch_array($result1));
echo'
</ul>
';
}

?>

</div>

</main>

<footer class="text-muted">
    <div class="container">
        <p class="float-right">
            <a href="#">Вернуться вверх</a>
        </p>
        <p>2018 &copy; Лучшие компьютерные игры.</p>
        <p>Электронная энциклопедия компьютерных игр.</p>
    </div>
</footer>

<!-- Bootstrap core JavaScript

```

```
===== -->
<!-- Placed at the end of the document so the pages load faster -->
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KcRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG5KkN"
crossorigin="anonymous"></script>
<script>window.jQuery || document.write('<script src="/assets/js/vendor/jquery-
slim.min.js"></script>')</script>
<script src="/assets/js/vendor/popper.min.js"></script>
<script src="/dist/js/bootstrap.min.js"></script>
<script src="/assets/js/vendor/holder.min.js"></script>
</body>
</html>
```

Заключение

Результатом выполнения курсового проекта является разработанная электронная энциклопедия компьютерных игр. В ходе проектирования веб-приложения были рассмотрены актуальные вопросы разработки и создания современного дизайна главной страницы сайта. При этом мною были решены следующие частные задачи: 1. Проанализированы существующие Веб-продукты отображения и продажи компьютерных игр. 2. Рассмотрены предложения по переработке существующего программного обеспечения. 3. Разработаны макеты сайта в графическом редакторе. 4. Создана главная страница сайта. 5. Отформатировано его содержимое средствами CSS. 6. Разработаны клиентские сценарии средствами JavaScript. 7. Произведено тестирование разработанного Веб-продукта. В результате проведенных работ на базе выбранных технологий была создана электронная энциклопедия компьютерных игр.

Список использованных источников

1. [url] **Курсовой проект**
https://drive.google.com/file/d/1i7yMftFueGBV_sm1kjeeam8cOuIMQ8pE/view?usp=sharing

Приложения