

**НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ ЧАСТНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ УНИВЕРСИТЕТ «СИНЭРГИЯ»
Факультет Информационных технологий**

ДИПЛОМНЫЙ ПРОЕКТ

на тему Разработка Информационной системы учета продаж игровых
аккаунтов на примере "ООО Фанпей"

Обучающийся

Хайдуров Николай Николаевич



(Фамилия, Имя, Отчество)

(подпись)

Руководитель

Джебилов Александр Валерьевич



(Фамилия, Имя, Отчество)

(подпись)

МОСКВА 2025 г.



A.B. Захаров

Факультет
Кафедра

Информационных технологий Цифровой экономики

ЗАДАНИЕ
на дипломный проект
обучающемуся
Хайдуров Николай Николаевич

1. Тема дипломного проекта:

Разработка Информационной системы учета продаж игровых аккаунтов на примере "ООО Фанпей"

2. Структура дипломного проекта:

Введение

Глава 1. Аналитическая часть

1.1. Технико-экономическая характеристика предметной области и предприятия

- 1.1.1. Характеристика предприятия и его деятельности
 - 1.1.2. Организационная структура управления предприятием
 - 1.1.3. Программная и техническая архитектура ИС предприятия
 - 1.2. Характеристика комплекса задач, задачи и обоснование необходимости автоматизации
 - 1.2.1. Выбор комплекса задач автоматизации и характеристика существующих бизнес-процессов
 - 1.2.2. Определение места проектируемой задачи в комплексе задач и ее описание
 - 1.2.3. Обоснование необходимости использования вычислительной техники для решения задачи
 - 1.2.4. Анализ системы обеспечения информационной безопасности и защиты информации
 - 1.3. Анализ существующих разработок и выбор стратегии автоматизации
 - 1.3.1. Анализ существующих разработок для автоматизации задачи
 - 1.3.2. Выбор и обоснование стратегии автоматизации задачи
 - 1.3.3. Выбор и обоснование способа приобретения ИС для автоматизации задачи

Глава 2. Проектная часть

2.1. Обоснование проектных решений

2.1.1. Обоснование проектных решений по информационному обеспечению

2.1.2. Обоснование проектных решений по программному обеспечению

2.1.3. Обоснование проектных решений по техническому обеспечению

2.2. Разработка проекта автоматизации

2.2.1. Этапы жизненного цикла проекта автоматизации

2.2.2. Характеристика нормативно-справочной, входной и оперативной информации

2.2.3. Характеристика результатной информации

2.3. Программное обеспечение задачи

2.3.1. Общие положения (дерево функций и сценарий диалога)

2.3.2. Характеристика базы данных

2.3.3. Структурная схема пакета (дерево вызова программных модулей)

2.3.4. Описание программных модулей

2.4. Контрольный пример реализации проекта и его описание

Глава 3. Обоснование экономической эффективности проекта

3.1. Выбор и обоснование методики расчёта экономической эффективности

3.2. Расчёт показателей экономической эффективности проекта

Заключение

Список использованной литературы

Приложения

3. Основные вопросы, подлежащие разработке:

Введение. Во «Введении» необходимо обосновать актуальность выбранной темы дипломного проекта, сформулировать цель и задачи. Сформулировать объект, предмет, научную и информационную базу дипломного проекта. Перечень задач целесообразно отразить по главам, т.е. указать, какие задачи будут решаться в рамках каждой главы.

В главе 1 необходимо представить обоснование актуальности выбора автоматизируемой задачи, проектных решений по информационному, программному и аппаратному обеспечению, дать ее развернутое описание, отразить взаимосвязь с другими задачами, изложить используемую стратегию автоматизации и способ приобретения информационной системы.

В разделе 1.1 необходимо привести краткое описание компании и таблицу показателей ее деятельности, рисунок организационной структуры и его описание, рисунки программной и технической архитектуры, а также их описание.

В разделе 1.2 следует обосновать актуальность выбора автоматизируемой задачи, для этого необходимо провести анализ организационной, программной и технической архитектуры с целью определения перечня задач, которые необходимо автоматизировать и выбора

наиболее приоритетной из них. С целью определения информационных потоков выбранной задачи необходимо привести соответствующие IDEF диаграммы. Далее необходимо привести рисунок, отражающий документооборот автоматизируемой задачи, таблицу прагматических характеристик соответствующих документов (периодичность, время на обработку и так далее) и определить комплекс программно-аппаратных средств, обеспечивающих информационную безопасность и защиту информации в рамках решаемой задачи.

В разделе 1.3 необходимо провести анализ готовых программных решений автоматизируемой задачи (в виде сравнительной таблицы наиболее интересных решений), определить стратегию автоматизации и способ приобретения информационной системы.

В главе 2 необходимо представить проектные решения в соответствии с выбранной стратегией автоматизации и разработки информационной системы и моделью жизненного цикла: начиная с анализа и выбора стратегии внедрения и заканчивая примером ее опытной эксплуатации.

В разделе 2.1 необходимо провести анализ и обосновать выбор средств, используемых для автоматизации задачи: технического обеспечения (компьютеры и периферия, сетевое оборудование, офисная техника, дополнительные устройства), программного обеспечения (операционная система, СУБД, среда разработки, другое прикладное ПО), информационного обеспечения (классификаторы, справочники, документы, информационные базы).

В разделе 2.2 необходимо провести выбор стандарта и модели жизненного цикла, соответствующих автоматизируемой задаче, а также стратегии внедрения проектируемой информационной системы. После этого следует провести анализ нормативно-справочной, входной, оперативной информации и результатной информации: документы, файлы, экранные формы, алгоритмы расчета результатных показателей и так далее.

Раздел 2.3 отражает процесс проектирования программного обеспечения задачи и должен содержать:

- рисунки дерева функций и сценария диалога, а также их описание;
- рисунок ER – модели (обязательно указание ключевых полей и связей между таблицами), описание структуры записей каждой таблицы;
- рисунок дерева вызова программных модулей, а также таблицу с перечнем и назначением модулей;
- блок-схему основного/ расчетного модуля, описание блок-схем алгоритмов основных расчетных модулей.

В разделе 2.4 следует представить экранные формы, демонстрирующие работу системы (не менее 7) и их описание.

В главе 3 приводится методика расчета показателей экономической эффективности и расчеты, сделанные в соответствии с изложенной методикой. Расчетные данные следует представить в виде таблиц и диаграмм, отражающие сравнение базового и предлагаемого вариантов.

Заключение.

В «Заключении» необходимо подвести итоги дипломного проектирования. Раскрыть содержание основных выводов, сделанных обучающимся, представить краткую характеристику результатов, полученных в ходе решения поставленных во «Введении» задач и, тем самым, ответить на основной вопрос дипломного проекта: о степени достижимости поставленной цели.

Список использованной литературы.

В «Список использованной литературы» приводятся только те информационные источники, которые автор лично использовал при написании данного дипломного проекта. Причем ссылки на данную литературу и информационные источники обязательны по всему тексту работы. Заимствованные чужие тексты в обязательном порядке заключаются в кавычки, как принадлежащие другому автору. Сноски приводятся постранично нарастающим итогом от № 1 до № N. Сноски, используемые обучающимся, должны быть отражены в списке использованной литературы в конце работы.

Приложение обязательно должно содержать фрагмент листинга программного кода (на исходном языке программирования отложенных основных расчетных модулей – около 400 операторов языка высокого уровня или адаптированных программных средств, использованных в работе), также могут быть приведены:

- схемы или таблицы из основной части дипломного проекта;
- результаты выполнения контрольного примера;
- диаграммы потоков данных, демонстрирующих существующую технологию решения задач;
- диаграммы потоков данных, демонстрирующих предлагаемую технологию решения задач;
- схемы документооборота;
- примеры классификаторов;
- формы первичных и результатных документов;
- распечатки меню, экранных форм ввода, получаемых отчетов в разработанной системе;
- а также другие материалы дипломного проекта, кроме текстов договоров с клиентами и иных «шаблонных документов» (в тех случаях, когда для их существенных реквизитов проектируется форма, а по результатам ввода и сохранения в информационную базу имеется возможность распечатки документа «по шаблону»).

В одном приложении нельзя размещать различные по смыслу таблицы или рисунки. Не допускается дублирование в приложении материала, размещенного в основной части дипломного проекта.

С детальным рассмотрением содержания каждого пункта, а также примерами схем и таблиц необходимо ознакомиться в «Методических

рекомендациях по выполнению дипломного проекта».

4. Исходные данные по дипломному проекту:

Основная литература:

1. Федеральный закон от 27.07.2006 № 149-ФЗ «Об информации, информационных технологиях и о защите информации» // <https://docs.cntd.ru/document/901990051/titles/64U0IK?ysclid=ltfq8y68gb365029912>

2. Грекул, В. И. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / В. И. Грекул, Н. Л. Коровкина, Г. А. Левочкина. — 2-е изд. — Москва : Издательство Юрайт, 2024. — 423 с. — (Профессиональное образование). — ISBN 978-5-534-17836-4. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/543034>.

3. Зараменских, Е. П. Информационные системы: управление жизненным циклом : учебник и практикум для среднего профессионального образования / Е. П. Зараменских. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 497 с. — (Профессиональное образование). — ISBN 978-5-534-16179-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/542807>.

4. Стружкин, Н. П. Базы данных: проектирование : учебник для среднего профессионального образования / Н. П. Стружкин, В. В. Годин. — Москва : Издательство Юрайт, 2024. — 477 с. — (Профессиональное образование). — ISBN 978-5-534-11635-9. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/542792>.

5. Системы управления технологическими процессами и информационные технологии : учебное пособие для среднего профессионального образования / В. В. Троценко, В. К. Федоров, А. И. Забудский, В. В. Комендантov. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2024. — 136 с. — (Профессиональное образование). — ISBN 978-5-534-09939-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539749>.

Дополнительная литература:

1. Богатырев, В. А. Надежность информационных систем : учебное пособие для среднего профессионального образования / В. А. Богатырев. — Москва : Издательство Юрайт, 2023. — 318 с. — (Профессиональное образование). — ISBN 978-5-534-15205-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/520442>.

2. Иванов, В. М. Интеллектуальные системы : учебное пособие для среднего профессионального образования / В. М. Иванов ; под научной редакцией А. Н. Сесекина. — Москва : Издательство Юрайт, 2024. — 93 с. — (Профессиональное образование). — ISBN 978-5-534-07819-0. — Текст :

электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/541299>.

3. Проектирование информационных систем : учебник и практикум для среднего профессионального образования / Д. В. Чистов, П. П. Мельников, А. В. Золотарюк, Н. Б. Ничепорук. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2024. — 293 с. — (Профессиональное образование). — ISBN 978-5-534-16217-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/538370>.

Интернет-ресурсы:

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Особенности проектирования информационных систем (cyberleninka.ru)	https://cyberleninka.ru/article/n/osobennosti-proektirovaniya-informatsionnyh-sistem/viewer
2.	Проектирование информационных систем	https://moodle.kstu.ru/course/view.php?id=4638
3.	Техническое задание на разработку информационной системы	https://dynamicsun.ru/blog/tz-na-inform-systemu.html
4.	Информационная система (ИС)	https://processmi.com/terms/informacionnaya-sistema-is/
5.	Информационные системы, понятие состав и структура	https://helpiks.org/8-10945.html

Руководитель:


подпись

Джебилов Александр Валерьевич
расшифровка

Обучающийся задание получил: «13» декабря 2024 г.

Обучающийся:


подпись

Хайдуров Н.Н
расшифровка

ВВЕДЕНИЕ

Актуальность темы дипломного проекта обусловлена стремительным развитием рынка цифровых товаров, в частности — рынка игровых аккаунтов. Рост числа пользователей, желающих приобрести игровые аккаунты для различных целей, приводит к необходимости автоматизации процессов учёта, управления продажами и защиты информации. ООО «Фанпей» является компанией, предоставляющей услуги по продаже игровых аккаунтов. Для повышения эффективности и прозрачности деятельности предприятия требуется внедрение информационной системы учёта продаж.

Цель работы: разработать информационную систему для автоматизации учёта продаж игровых аккаунтов на примере ООО «Фанпей».

Задачи дипломного проекта:

проанализировать текущее состояние учёта и продажи игровых аккаунтов;

исследовать предметную область, определить основные бизнес-процессы;

разработать требования к автоматизации;

спроектировать архитектуру и базу данных информационной системы;

реализовать прототип программного обеспечения;

провести оценку экономической эффективности внедрения.

Объект исследования: процессы автоматизации учёта и управления продажами игровых аккаунтов.

Предмет исследования: методы и средства проектирования и реализации информационных систем в сфере цифровых товаров.

Методы исследования: анализ бизнес-процессов, методы структурного и объектно-ориентированного проектирования, использование CASE-средств, методы экономической оценки эффективности.

ГЛАВА 1. АНАЛИТИЧЕСКАЯ ЧАСТЬ

1.1. Технико-экономическая характеристика предметной области и предприятия

1.1.1. Характеристика предприятия и его деятельности

ООО «Фанпей» — Биржа игровых аккаунтов и не только, в последнее время на ней появились и услуги, связанные с играми и ПО: подписки (например, Spotify Premium), “Буст” (поднятие игрового рейтинга аккаунта посредством игры на нём более умелого игрока) игровых аккаунтов, и прочее. На сайте доступны тысячи наименований, и каждый может начать продавать товары и оказывать услуги. Главным конкурентом на данный момент является форум LolzTeam, который обладает таким же функционалом. Так же можно привести в пример недавно зашедшего на данный рынок игрока Playerok.

Рассмотрим показатели за 2024 год:

Показатель	Значение за 2024
Количество проданных аккаунтов	8 543 952
Количество уникальных пользователей	10 326 653
Объём продаж, руб.	16 250 000
Средняя цена аккаунта, руб.	730
Количество обращений в поддержку	2540
Среднее время решения запроса, ч.	1.5

Таблица 1.

1.1.2. Организационная структура управления предприятием

Управление компанией «Фанпей» осуществляется по функциональной структуре. Основные отделы: технический (IT), отдел поддержки клиентов, финансовый отдел, отдел безопасности, маркетинговый отдел.

Взаимодействие между отделами организовано через программные средства и корпоративные коммуникации.

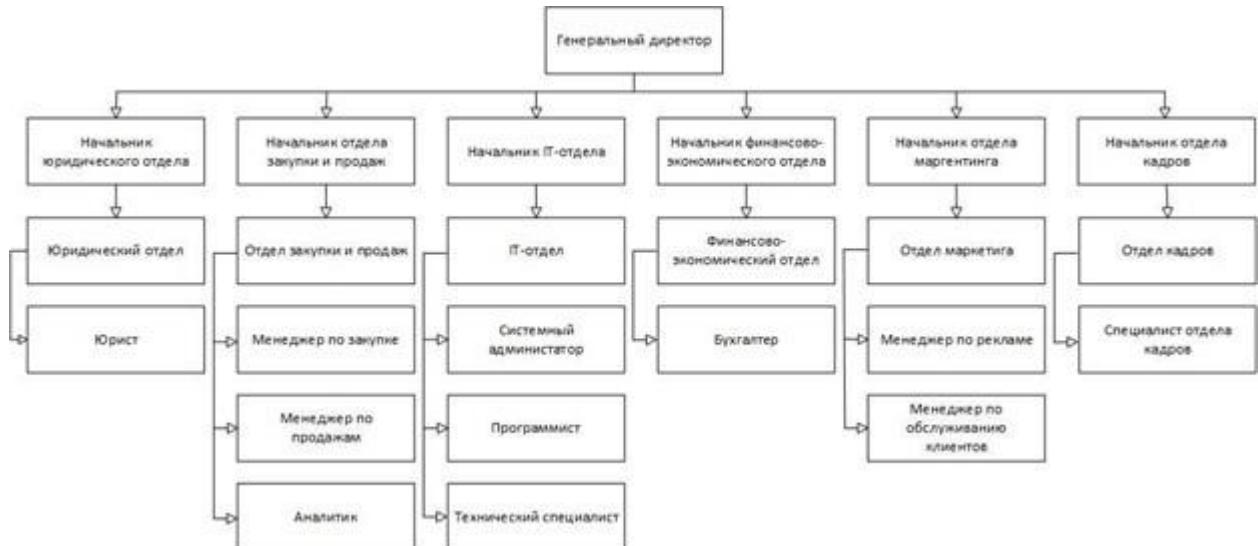


Рис.1 Организационная схема

1.1.3. Программная и техническая архитектура ИС предприятия

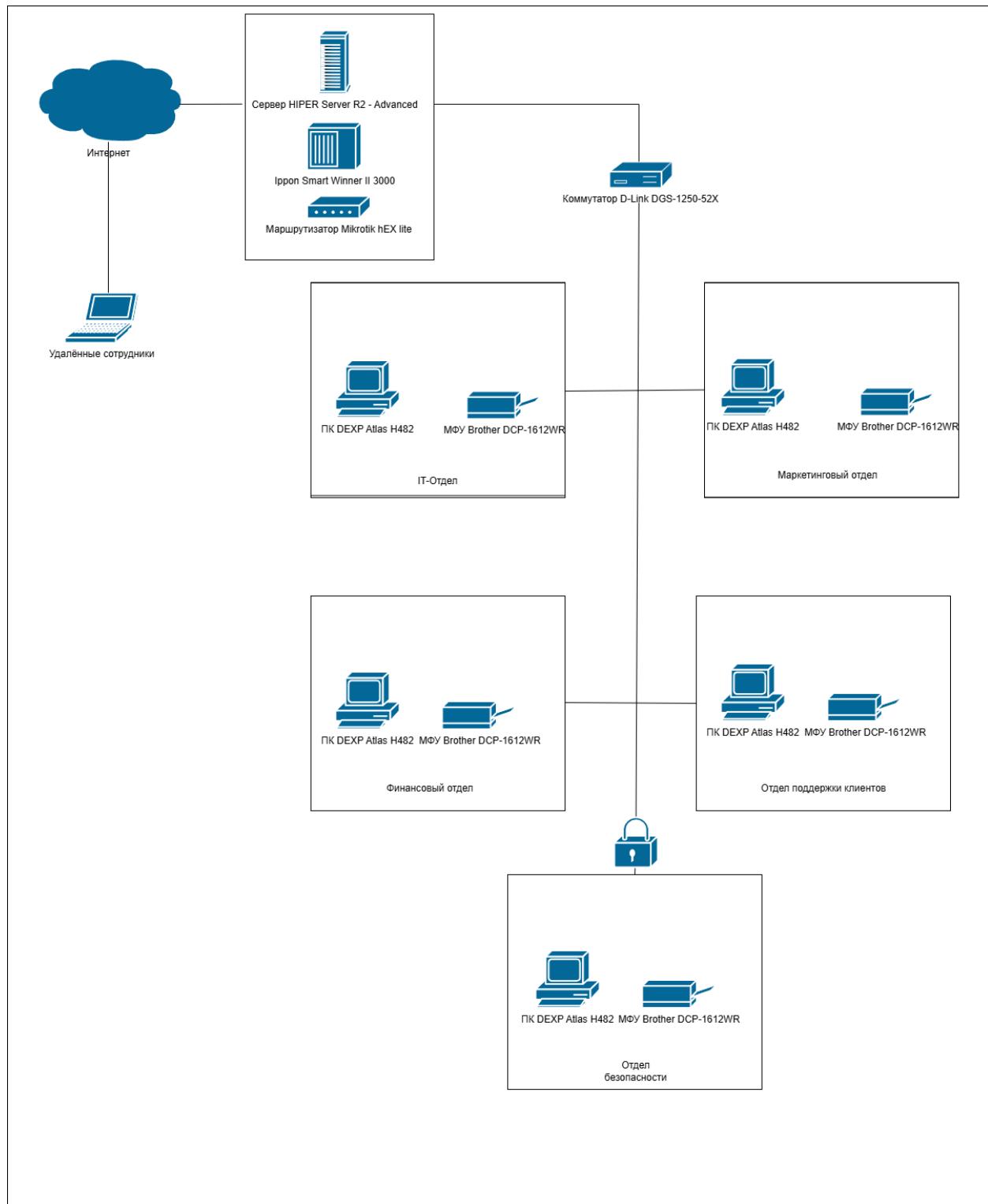


Рис.2 Техническая архитектура

№	Наименование устройства	Вид оборудования	Технические характеристики	Место установки
1	HIPER Server R2 - Advanced	Сервер	Xeon Silver 4210, 32 ГБ ОЗУ, 2×SSD 1 ТБ, RAID	Серверная
2	Mikrotik hEX lite	Маршрутизатор	5×Ethernet, 850 МГц CPU, 256 МБ ОЗУ	Серверная
3	Ippon Smart Winner II 3000	Источник бесперебойного питания	3000 ВА, LCD, AVR, время работы 10 мин	Серверная
4	D-Link DGS-1250-52X	Коммутатор	48 портов 1G, 4×10G SFP+, Layer 2	Серверная
5	ПК DEXP Atlas H482	Персональный компьютер	Intel i5-10400, 8 ГБ ОЗУ, SSD 512 ГБ, Windows 11	Рабочие места сотрудников
6	МФУ Brother DCP-1612WR	МФУ (принтер/сканер)	Лазерный, Wi-Fi, A4	Рабочие места сотрудников

Таблица 2

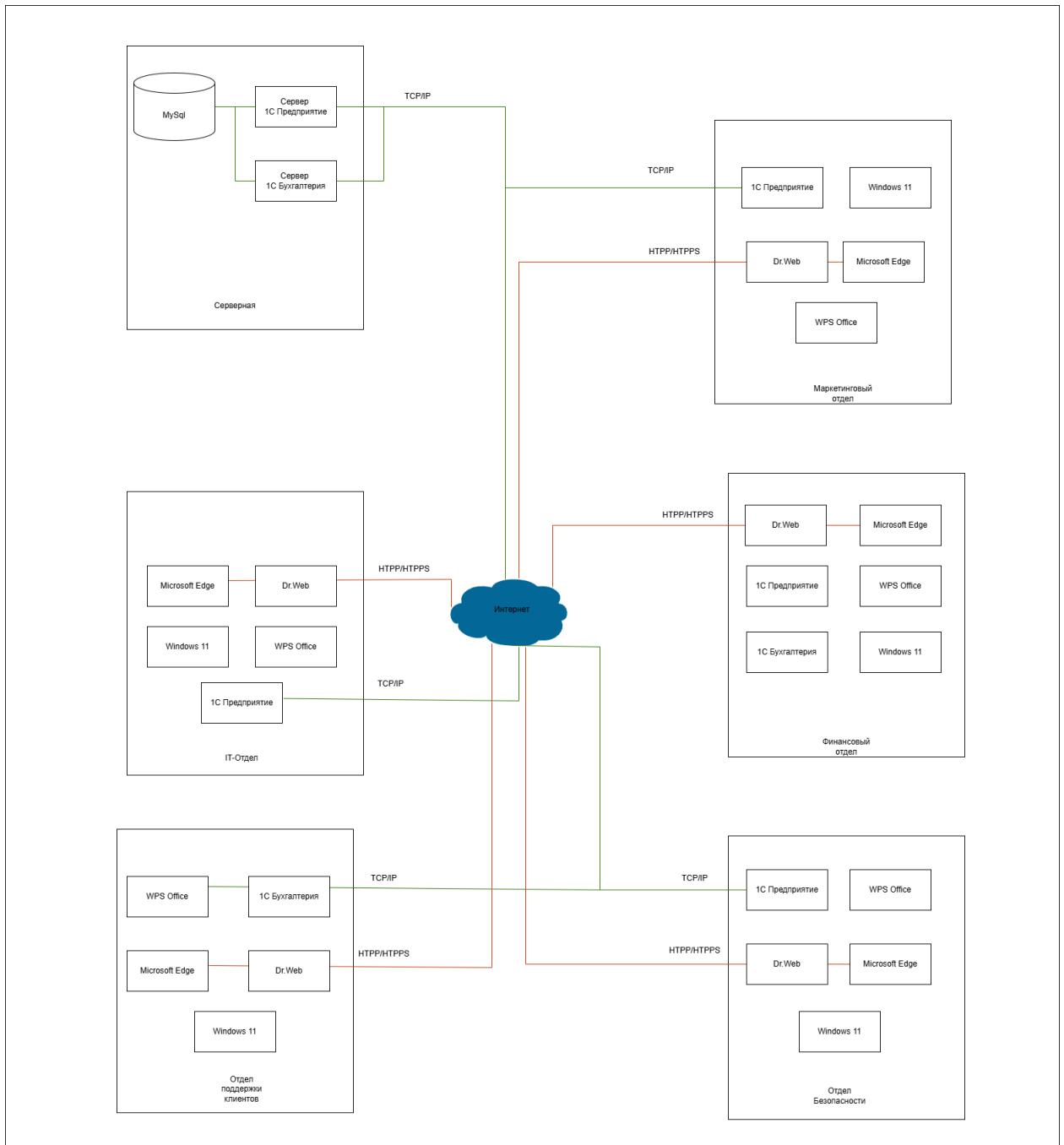


Рис.3 Программная архитектура

Техническая архитектура «Фанпей» включает в себя серверы баз данных, пользовательские рабочие станции. Программное обеспечение состоит из серверной части (СУБД MySQL), клиентских приложений (Windows Forms), Веб-браузера Microsoft Edge, антивируса Dr.Web, ОС Windows 11, WPS Office, 1С Бухгалтерия, 1С Предприятие

Вставить схемы архитектуры и описание состава программного/технического обеспечения.

1.2. Характеристика комплекса задач, задачи и обоснование необходимости автоматизации

1.2.1. Выбор комплекса задач автоматизации и характеристика существующих бизнес-процессов

В качестве основного объекта автоматизации выбран процесс учёта и продажи игровых аккаунтов. Анализ бизнес-процессов показал, что ручное ведение учёта приводит к ошибкам, затрудняет обработку заказов, усложняет формирование отчётов и увеличивает издержки. Автоматизация учёта продаж позволит повысить скорость обработки заявок, снизить вероятность ошибок и упростить анализ финансовых показателей.

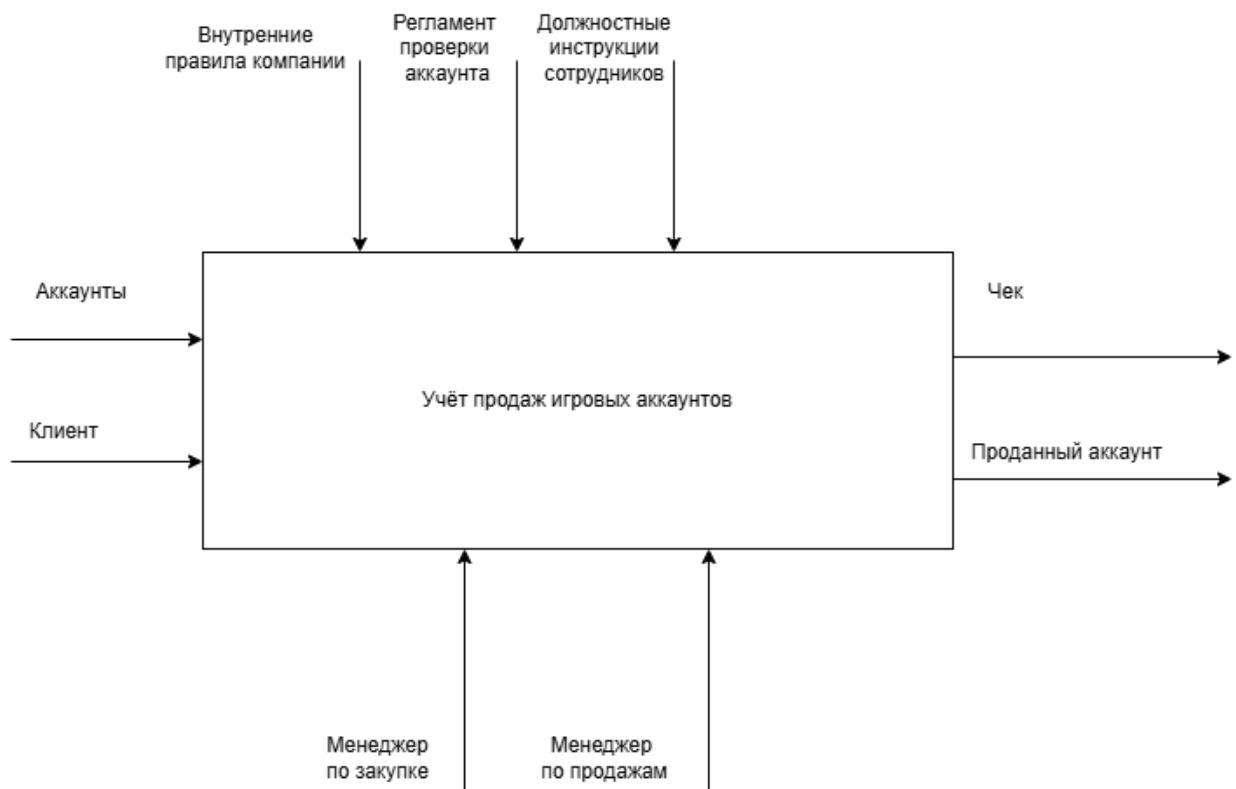


Рис.4 Диаграмма бизнес-процесса учёта продаж

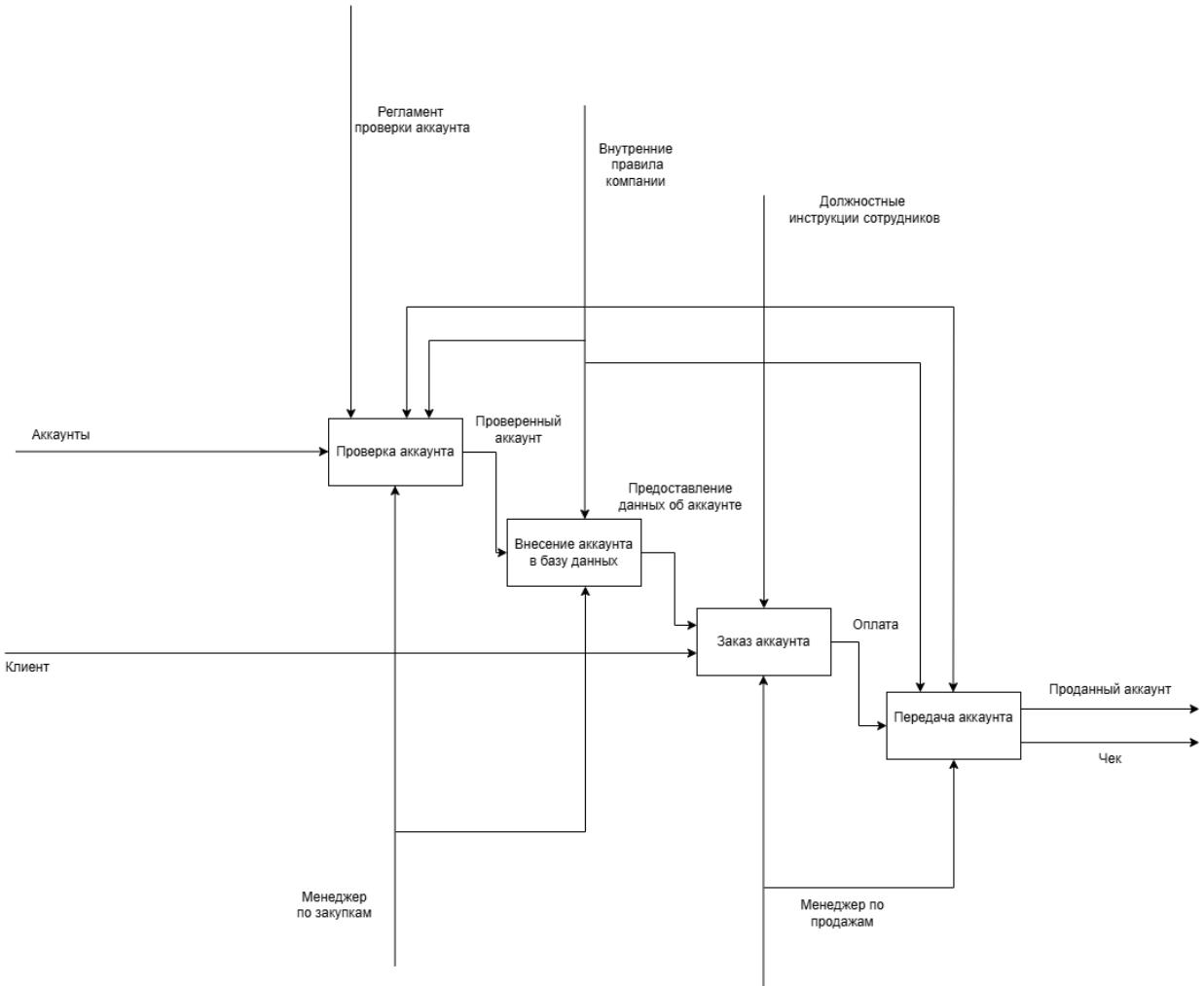


Рис5. Декомпозиция

1.2.2. Определение места проектируемой задачи в комплексе задач и её описание

Учёт продаж является центральной задачей, интегрированной с задачами авторизации, управления пользователями, формирования отчётов и администрирования. Проектируемая система должна обеспечивать регистрацию пользователей, учёт их ролей (администратор, пользователь), продажу, покупку и учёт аккаунтов.

1.2.3. Обоснование необходимости использования вычислительной техники для решения задачи

Использование вычислительной техники необходимо из-за большого объёма информации, необходимости оперативной обработки и хранения данных, а также из-за требований к безопасности и конфиденциальности информации.

1.2.4. Анализ системы обеспечения информационной безопасности и защиты информации

Важным аспектом является защита персональных данных пользователей, учетных записей и финансовых операций. Применяются современные

методы шифрования, разграничение прав доступа, двухфакторная аутентификация и резервное копирование базы данных.

1.3. Анализ существующих разработок и выбор стратегии автоматизации

1.3.1. Анализ существующих разработок для автоматизации задачи

На рынке существуют различные решения для автоматизации продажи цифровых товаров и учёта пользователей. В рамках анализа были рассмотрены платформы типа Plati.ru, G2G, а также специализированные решения для интернет-магазинов. Каждый из вариантов обладает определёнными достоинствами, но не учитывает специфики рынка игровых аккаунтов.

1.3.2. Выбор и обоснование стратегии автоматизации задачи

Выбор сделан в пользу индивидуальной разработки информационной системы, учитывающей особенности бизнеса ООО «Фанпей», требования к безопасности, учёту и управлению ролями пользователей.

1.3.3. Выбор и обоснование способа приобретения ИС для автоматизации задачи

Для ООО «Фанпей» наиболее целесообразна собственная разработка и внедрение программного продукта, что позволяет обеспечить полную адаптацию под потребности компании и гибко модернизировать систему в дальнейшем.

ГЛАВА 2. ПРОЕКТНАЯ ЧАСТЬ

2.1. Обоснование проектных решений

2.1.1. Обоснование проектных решений по информационному обеспечению

Информационное обеспечение информационной системы FunPayApp включает в себя структурированную базу данных для хранения информации о пользователях, аккаунтах, сделках и других элементах системы. В качестве СУБД выбран MySQL как надёжное и производительное решение, поддерживающее масштабирование и безопасность.

Основные информационные объекты:

- Пользователи системы (логин, пароль, роль, список аккаунтов).
- Аккаунты для продажи (игра, логин, стоимость, статус).
- Сделки (ID, покупатель, продавец, дата, итоговая цена).
- История операций и изменений.

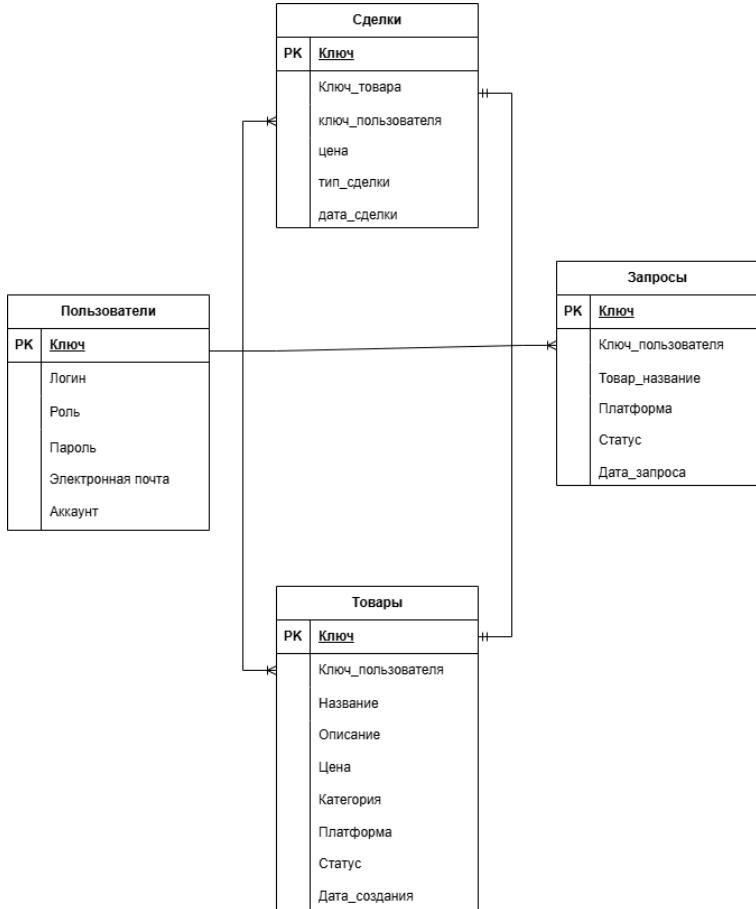


Рис.6 Логическая модель БД

2.1.2. Обоснование проектных решений по программному обеспечению
 В качестве среды разработки выбран Microsoft Visual Studio с использованием языка программирования C# и технологий .NET Framework/Windows Forms. Это обусловлено доступностью, широким сообществом и удобством для создания десктопных приложений с графическим интерфейсом.

- Программное обеспечение обеспечивает:
- Надёжную работу с базой данных через библиотеку MySql.Data.
- Безопасную обработку пользовательских данных.
- Удобный графический интерфейс для пользователя и администратора.

2.1.3. Обоснование проектных решений по техническому обеспечению
 Для функционирования системы требуется минимальный набор техники:

- Сервер или ПК с установленной MySQL.
- Рабочие станции пользователей (клиентское приложение FunPayApp).

- Локальная сеть или облачный сервис для передачи данных между клиентом и сервером.

2.2. Разработка проекта автоматизации

2.2.1. Этапы жизненного цикла проекта автоматизации

В проекте FunPayApp выбрана итерационная модель жизненного цикла, что позволяет поэтапно реализовывать, тестировать и совершенствовать систему, внедряя новые функции на каждом этапе.

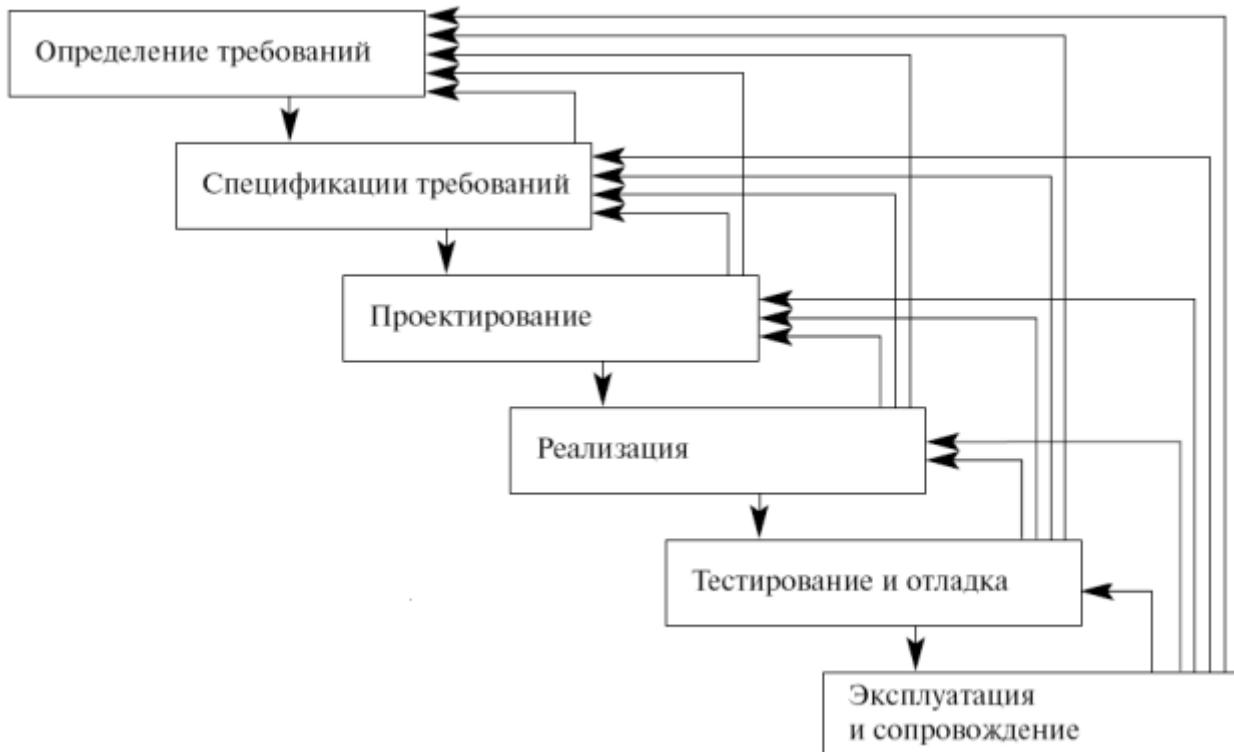


Рис.7 Итерационная модель ЖЦ

2.2.2. Характеристика нормативно-справочной, входной и оперативной информации

- Нормативно-справочная информация — списки поддерживаемых игр, шаблоны для размещения аккаунтов, роли пользователей.
- Входная информация — данные, вводимые пользователями (логин, пароль, информация об аккаунте, параметры сделки).
- Оперативная информация — данные, формируемые в ходе работы: новые аккаунты, сделки, изменения статусов.

2.2.3. Характеристика результатной информации

Результатная информация отображается в виде:

- Списков пользователей, аккаунтов и сделок в таблицах (DataGridView).
- Статистических данных по продажам.
- Отчётов о совершенных операциях.

2.3. Программное обеспечение задачи

2.3.1. Общие положения (дерево функций и сценарий диалога)

Структура приложения FunPayApp предусматривает разделение по ролям:

- Регистрация и авторизация пользователей.
- Работа с аккаунтами (добавление, удаление, изменение статуса).
- Администрирование (управление пользователями, изменение ролей).

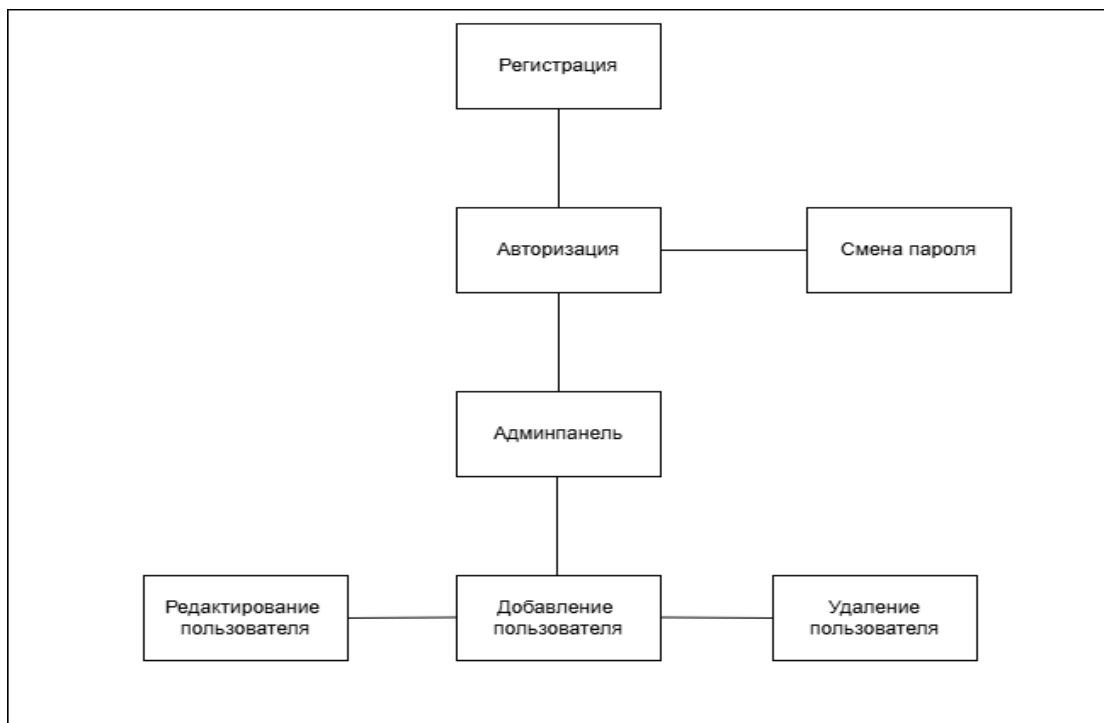


Рис.8 Дерево сценария диалога

2.3.2. Характеристика базы данных

База данных содержит следующие основные таблицы:

Наименование поля	Наименование на англ.	Тип данных	Длина / Точность	Ключевое свойство	Описание
Идентификатор	id	Числовой (int)	11	Первичный ключ	Уникальный ID пользователя
Логин	login	Текстовый	55	Уникальный	Логин пользователя
Роль	role	Текстовый	20	-	Роль пользователя (admin/user)
Пароль	password	Текстовый	55	-	Пароль пользователя
Email	email	Числовой (int)	11	-	Email пользователя
Аккаунт	account	Текстовый	255	-	Привязанный аккаунт

Таблица 3. Пользователи

Наименование поля	Наименование на англ.	Тип данных	Длина / Точность	Ключевое свойство	Описание
Идентификатор	id	Числовой (int)	10	Первичный ключ	Уникальный ID товара
Продавец	seller_id	Числовой (int)	10	Внешний ключ	ID пользователя-продавца
Название	title	Текстовый (varchar)	200	-	Название аккаунта/товара
Описание	description	Текстовый (text)	-	-	Описание
Цена	price	Числовой (decimal)	10,2	-	Цена
Категория	category	Текстовый (varchar)	100	-	Категория
Платформа	platform	Текстовый (varchar)	50	-	Игровая платформа

Наименование поля	Наименование на англ.	Тип данных	Длина / Точность	Ключевое свойство	Описание
Статус	status	Перечисление (enum)	-	-	Статус: "в наличии", "продано", "скрыто"
Дата создания	created_at	Дата/время (datetime)	-	-	Дата добавления записи

Таблица 4. Товары

Наименование поля	Наименование на англ.	Тип данных	Длина / Точность	Ключевое свойство	Описание
Идентификатор	id	Числовой (int)	10	Первичный ключ	Уникальный ID сделки
Товар	product_id	Числовой (int)	10	Внешний ключ	ID аккаунта/товара
Покупатель	buyer_id	Числовой (int)	10	Внешний ключ	ID пользователя-покупателя
Итоговая цена	final_price	Числовой (decimal)	10,2	-	Итоговая сумма
Тип сделки	deal_type	Перечисление (enum)	-	-	Покупка/Продажа
Дата сделки	deal_date	Дата/время (datetime)	-	-	Дата и время проведения сделки

Таблица 5. Сделки

Наименование поля	Наименование на англ.	Тип данных	Длина / Точность	Ключевое свойство	Описание
Идентификатор	id	Числовой (int)	10	Первичный ключ	Уникальный ID заявки
Пользователь	user_id	Числовой (int)	10	Внешний ключ	пользователя, создавшего заявку
Название товара	product_title	Текстовый (varchar)	200	-	Название запрашиваемого

Наименование поля	Наименование на англ.	Тип данных	Длина / Точность	Ключевое свойство	Описание
Платформа	platform	Текстовый (varchar)	50	-	о аккаунта/товара Платформа, для которой требуется аккаунт
Статус	status	Перечисление (enum)	-	-	Статус заявки: "активен", "выполнено"
Дата заявки	request_date	Дата/время (datetime)	-	-	Дата и время подачи заявки

Таблица 6. Заявки

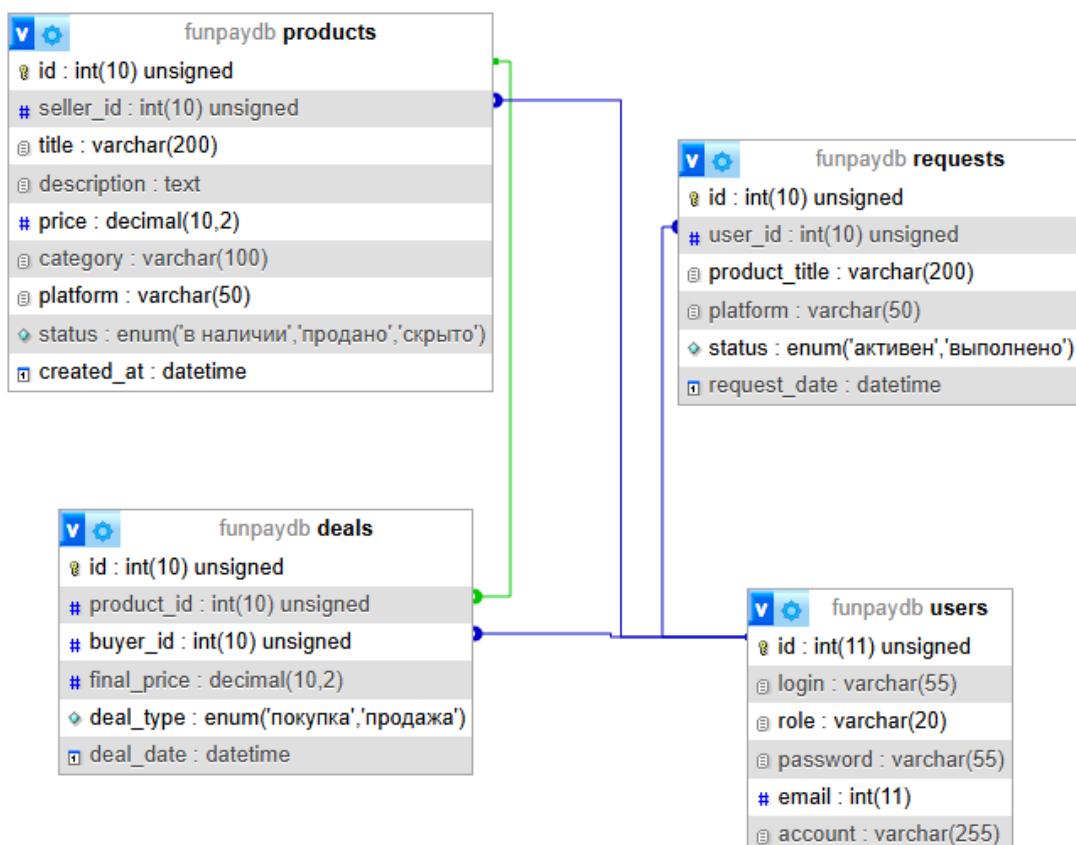


Рис.10 Физическая модель БД.

2.3.3. Структурная схема пакета (дерево вызова программных модулей)

Модули приложения:

- Главная форма (MainForm)
- Форма регистрации (RegisterForm)
- Форма авторизации (LoginForm)
- Админ-панель (AdminForm)
- Модуль работы с базой данных (database.cs)
- Модули управления аккаунтами и сделками

2.3.4. Описание программных модулей

- RegisterForm.cs — регистрация новых пользователей.
- LoginForm.cs — авторизация, проверка прав доступа.
- AdminForm.cs — управление пользователями, изменение ролей.
- AccountForm.cs — добавление/редактирование аккаунтов.
- DealForm.cs — оформление и история сделок.
- database.cs — подключение и работа с MySQL.

Каждый модуль реализует свою часть функционала, взаимодействуя через единый доступ к базе данных.



РЕГИСТРАЦИЯ



Логин:

Пароль:

E-mail:

Зарегистрироваться

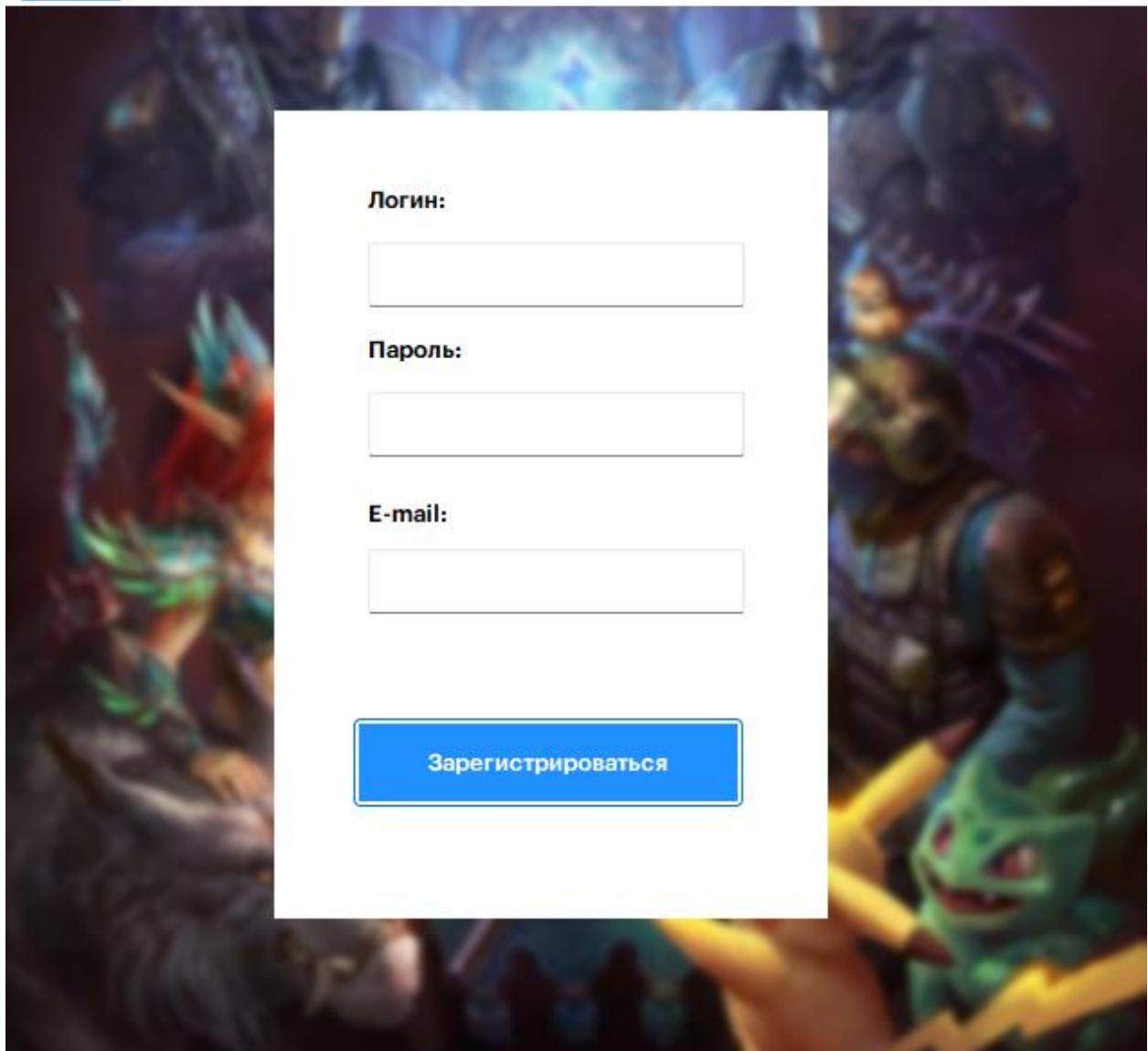


Рис.11 Форма регистрации

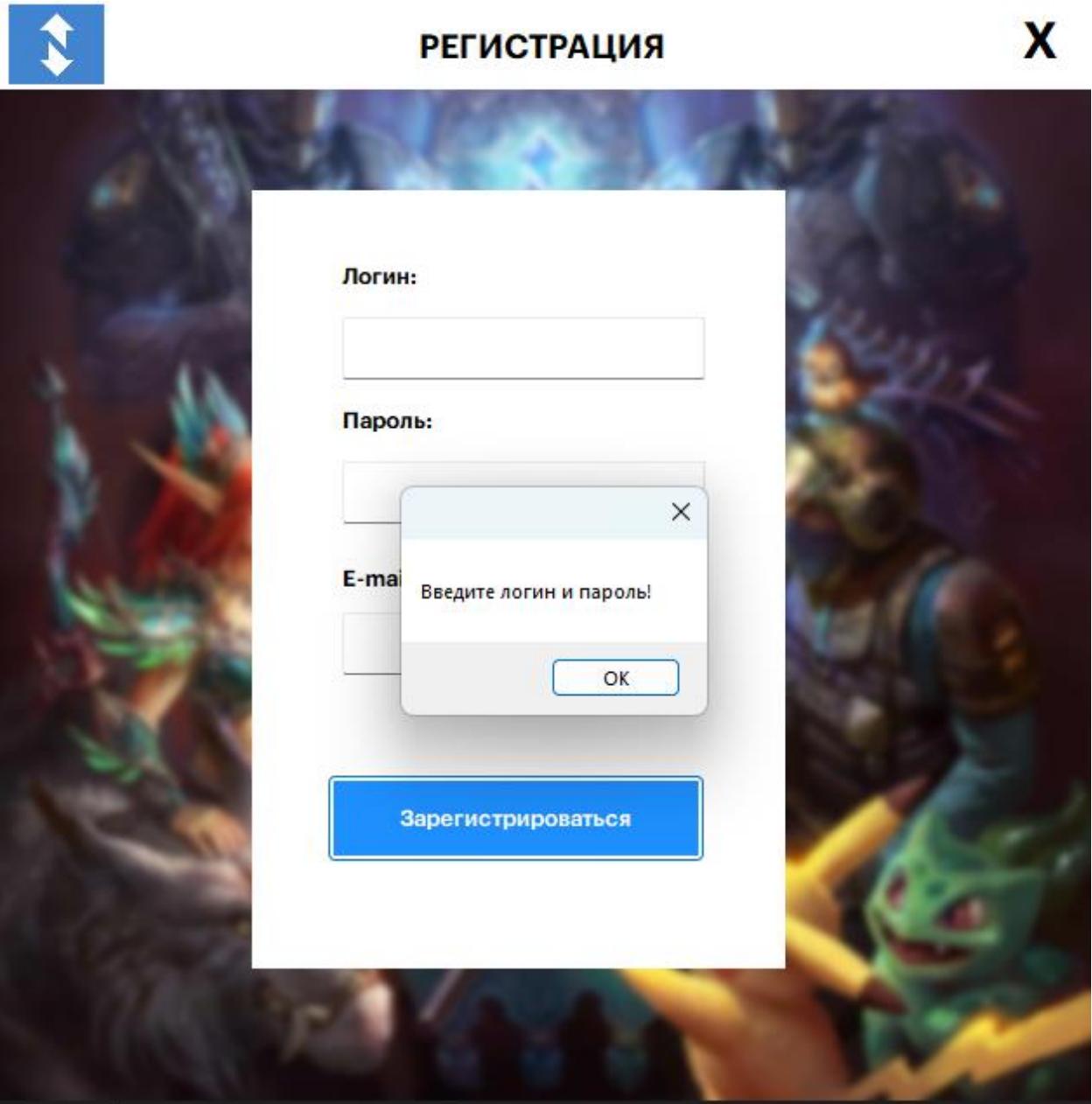


Рис.12 Попытка входа без ввода данных



РЕГИСТРАЦИЯ

X

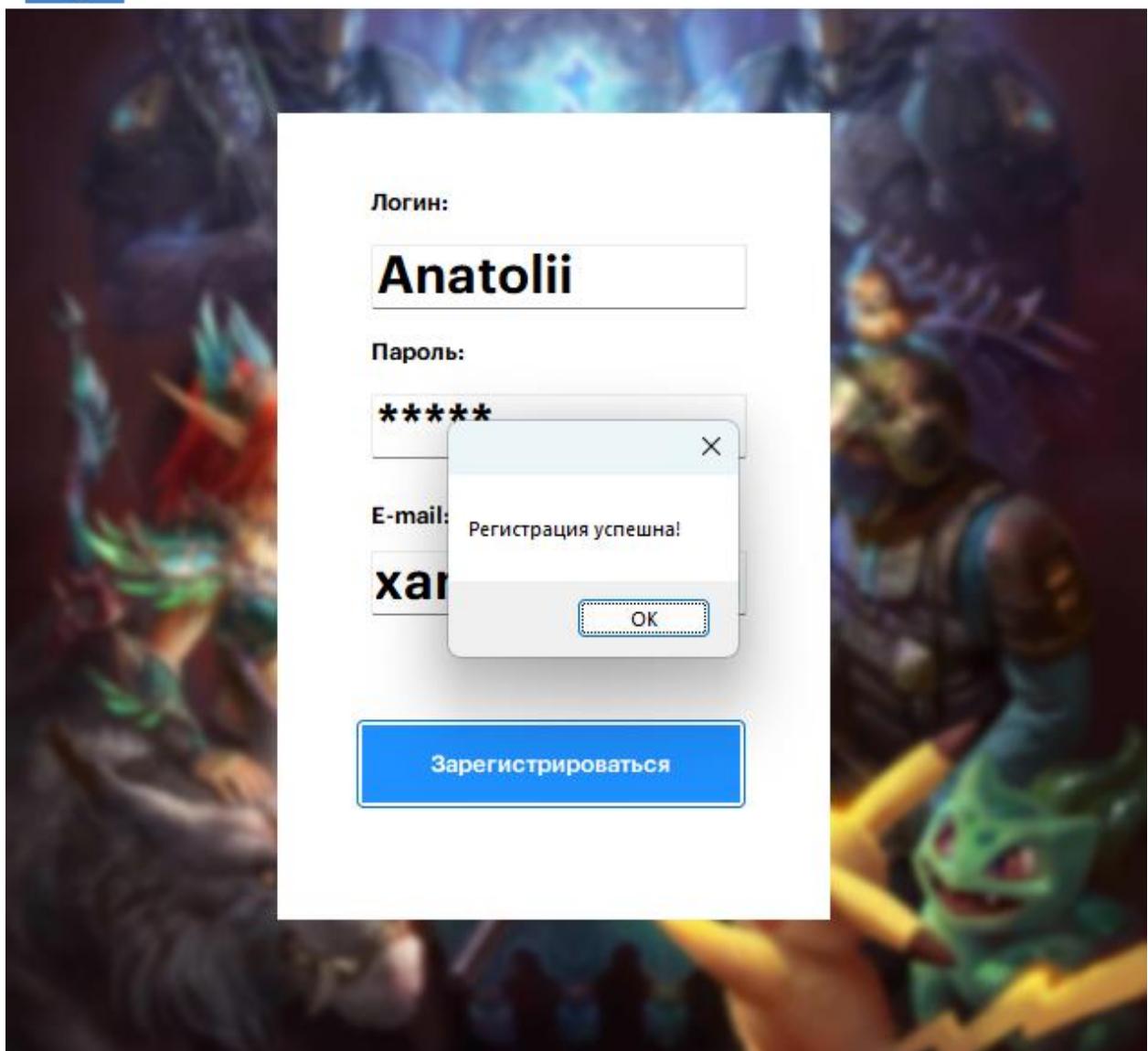


Рис.13 Успешный вход

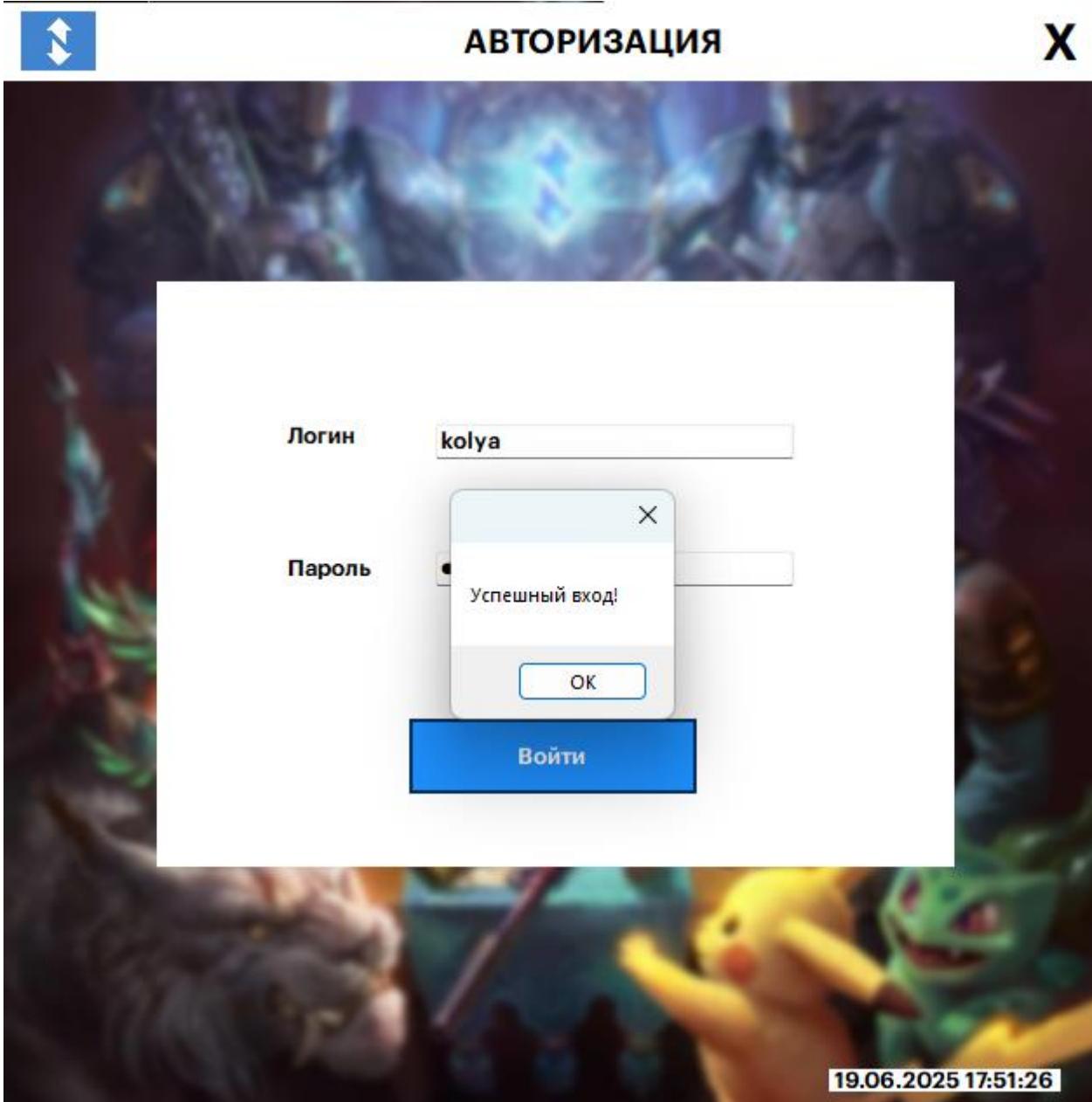


Рис.14 Успешная авторизация



АВТОРИЗАЦИЯ

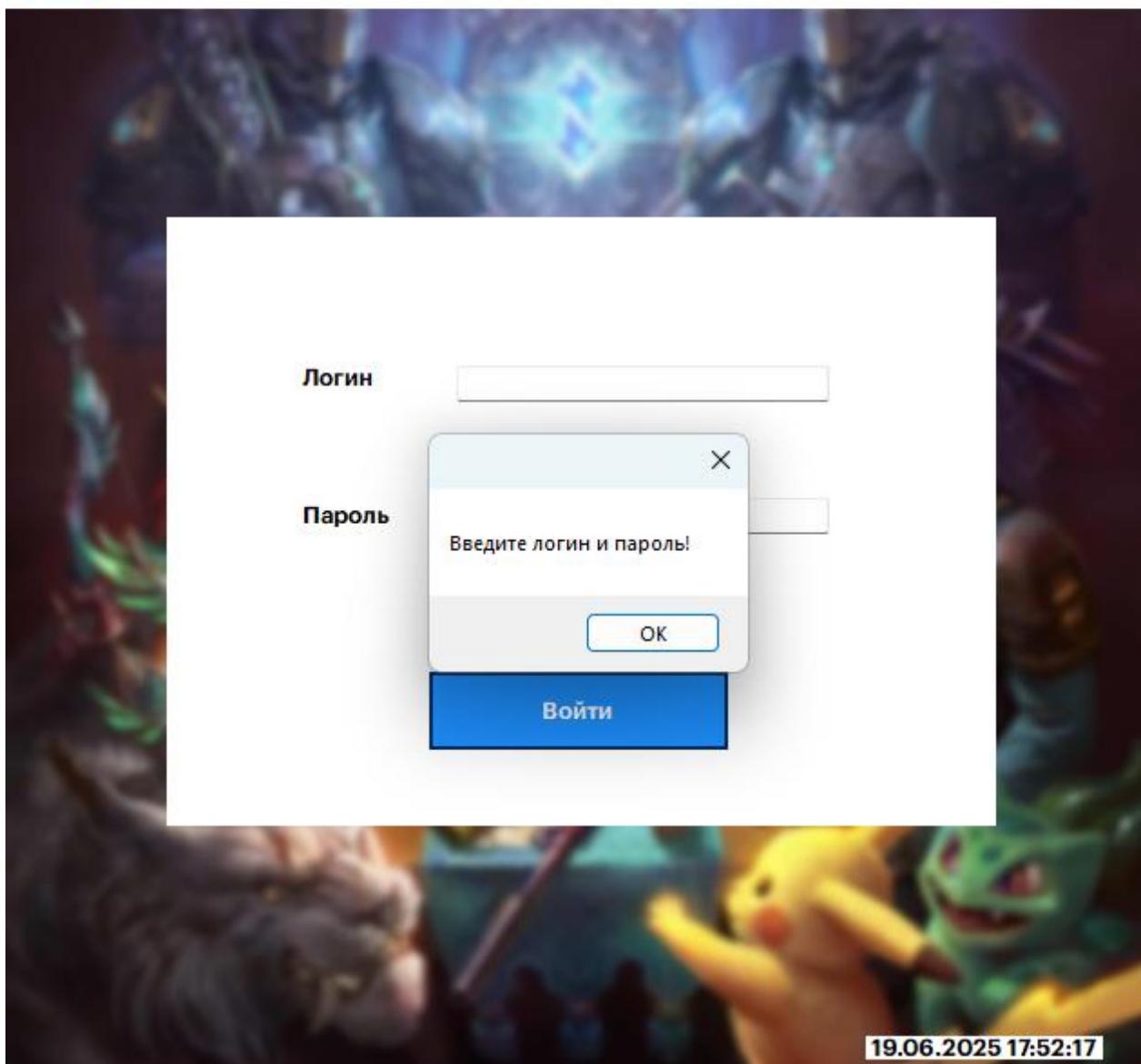


Рис.15 Вход без введённых данных

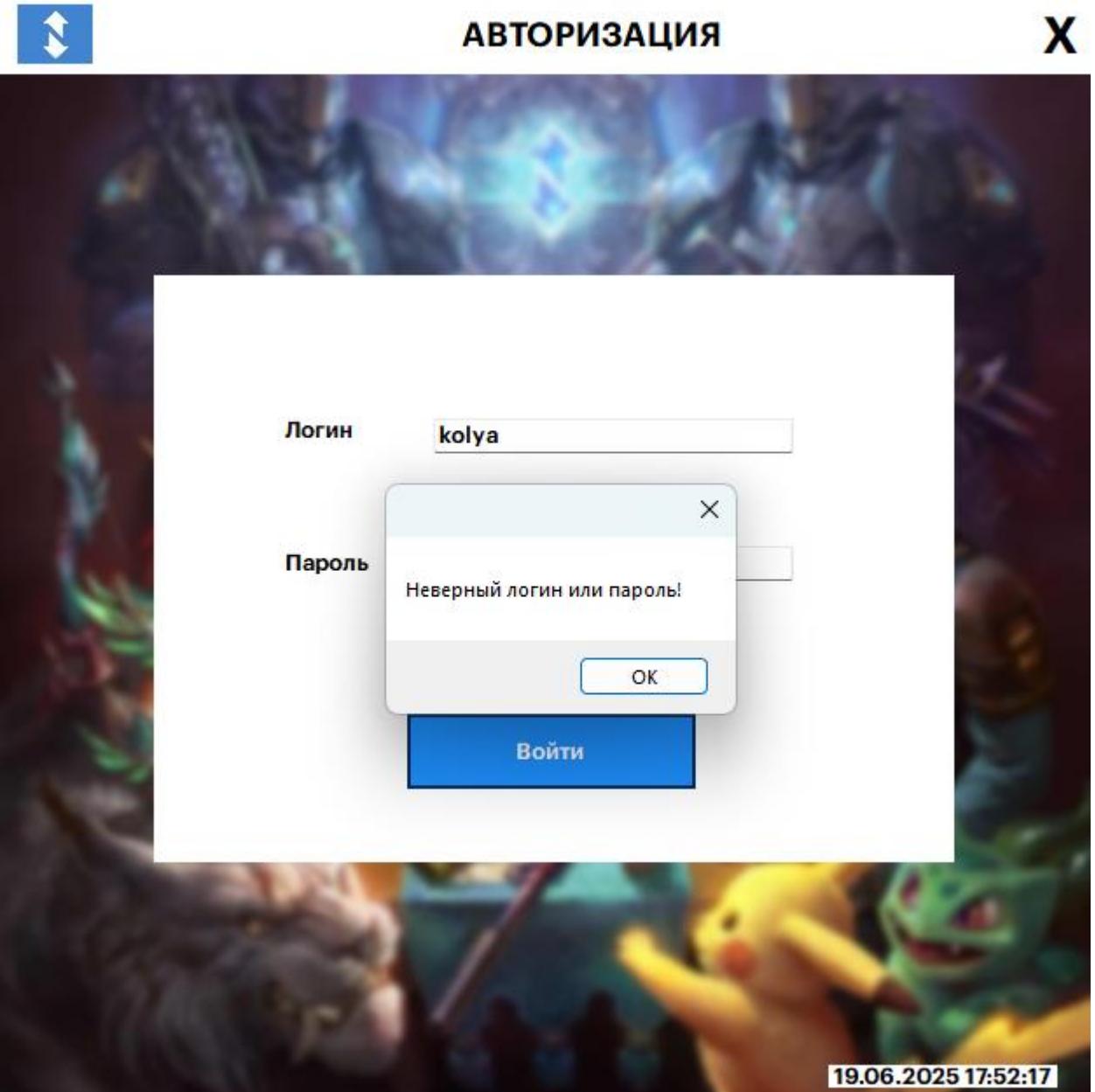


Рис.16 Вход с неверно введёнными данными

The screenshot shows an admin panel interface. On the left is a table listing users with columns: id, login, role, and account. The table contains 11 rows of data. On the right is a form for adding a new user, with fields for Login, Password, Role (dropdown), and Account. Below the form are three buttons: 'Добавить пользователя' (Add user), 'Сохранить изменения' (Save changes), and 'Обновить' (Update).

	id	login	role	account
1	ognikolaichh	admin		
2	GeorgeBushJr	user	Dota 2 7000MMR	
3	SmiRan	user	Minecraft (Bedro...	
4	firstpack_	user	Terraria	
5	wlehaa	user	GTAV	
6	Hunter251	user	CSGO	
7	kolya	user	iCloud 50GB	
8	alex	user	Battlefield 2042	
9	2435w34	user	WoT	
10	MCar	admin	Soundcloud+	
11	Anatolii	user	Я.Музыка	
*				

Рис.17 Админ-панель

The screenshot shows the same admin panel as in Figure 17, but with a modal dialog box in the center. The dialog has a title 'Успех' (Success) and a message 'Пользователь добавлен!' (User added). It also features a blue info icon and an 'OK' button. The rest of the interface remains the same, including the user list and the add form.

Рис.18 Добавление пользователя

ГЛАВА 3. ОБОСНОВАНИЕ ЭКОНОМИЧЕСКОЙ ЭФФЕКТИВНОСТИ ПРОЕКТА

3.1. Выбор и обоснование методики расчёта экономической эффективности

Для оценки эффективности внедрения информационной системы учёта продаж игровых аккаунтов (FunPayApp) применяется комплексный подход, включающий анализ трудозатрат и экономии средств в процессе автоматизации основных бизнес-операций.

Экономическая эффективность оценивается по следующим критериям:
Абсолютное снижение трудовых затрат (ΔT , в часах/минутах за год):
 $\Delta T = T_0 - T_1$ где T_0 — трудозатраты в часах (или минутах) на обработку информации до внедрения автоматизации;

T_1 — трудозатраты после внедрения информационной системы.

Индекс снижения трудовых затрат (YT) — показатель повышения производительности труда:

$$YT = T_0 / T_1$$

Абсолютное снижение стоимостных затрат (ΔC):

$$\Delta C = C_0 - C_1, \text{ где}$$

C_0 — расходы на выполнение операций в ручном режиме;

C_1 — расходы при использовании автоматизированной системы.

Срок окупаемости проекта (Ток):

$$Tok = K_p / \Delta C, \text{ где}$$

K_p — затраты на внедрение и разработку системы;

ΔC — ежегодная экономия средств.

3.2. Расчёт показателей экономической эффективности проекта

В состав стоимости реализации проекта включены следующие категории затрат:

- Оплата труда специалистов, участвующих в проекте;
- Затраты на расходные материалы и компоненты (офисные расходы);
- Прочие накладные расходы (например, аренда, электроэнергия).

Должность	Ставка за 1 час, руб.	Кол-во часов	Общая сумма, руб.
Разработчик	350	160	56 000
Системный администратор	300	80	24 000
Менеджер по продукту	400	15	6 000
Тестировщик	350	25	8 750
Бухгалтер	300	10	3 000
Итого			97 750

Таблица 7. Расчёт затрат на оплату труда специалистов

Материал/компонент Цена за 1 ед., руб. Кол-во Итог, руб.

Бумага для печати	350	2	700
Картридж	3700	2	7 400
Итого			8 100

Таблица 8. Расчёт затрат на расходники

Итого совокупные затраты на проект:
 $S = 9775 \text{ руб.} + 8100 \text{ руб.} = 105850 \text{ руб.}$

Расчёт трудозатрат:

Допустим, до внедрения автоматизации для обработки аккаунтов требовалось:

Среднее кол-во обработанных аккаунтов в год: $150 \times 12 = 1800$

Время обработки одного аккаунта: 8 минут (до автоматизации), 2 минуты (после автоматизации)

Число специалистов: 2

До внедрения:

$T_0 = 1800 \times 8 \text{ мин} \times 2 = 28800 \text{ мин} = 480 \text{ ч}$

После внедрения:

$T_1 = 1800 \times 2 \text{ мин} \times 2 = 7200 \text{ мин} = 120 \text{ ч}$

Экономия времени:

$\Delta T = T_0 - T_1 = 480 \text{ ч} - 120 \text{ ч} = 360 \text{ ч}$

Индекс производительности:

$YT = T_0 / T_1 = 480 / 120 = 4$ (производительность выросла в 4 раза)

Расчёт снижения стоимостных затрат:

Если специалисты получают по 45 000 руб./мес. и можно уменьшить кол-во специалистов с 2 до 1 (или перераспределить их функции):

До автоматизации:

$C_0 = 12 \times 2 \times 45000 = 1080000 \text{ руб.}$

После автоматизации:

$C_1 = 12 \times 1 \times 45000 = 540000 \text{ руб.}$

Экономия:

$\Delta C = 1080000 - 540000 = 540000 \text{ руб./год}$

Срок окупаемости проекта:

$T_{ok} = K_p / \Delta C = 105850 / 540000 \approx 0,2 \text{ года (около 2-3 месяцев)}$

Вывод

Таким образом, внедрение системы FunPayApp позволяет существенно снизить трудозатраты и производственные расходы, обеспечивая быструю окупаемость инвестиций. Полученные показатели свидетельствуют о высокой экономической эффективности проекта автоматизации учёта продаж игровых аккаунтов.

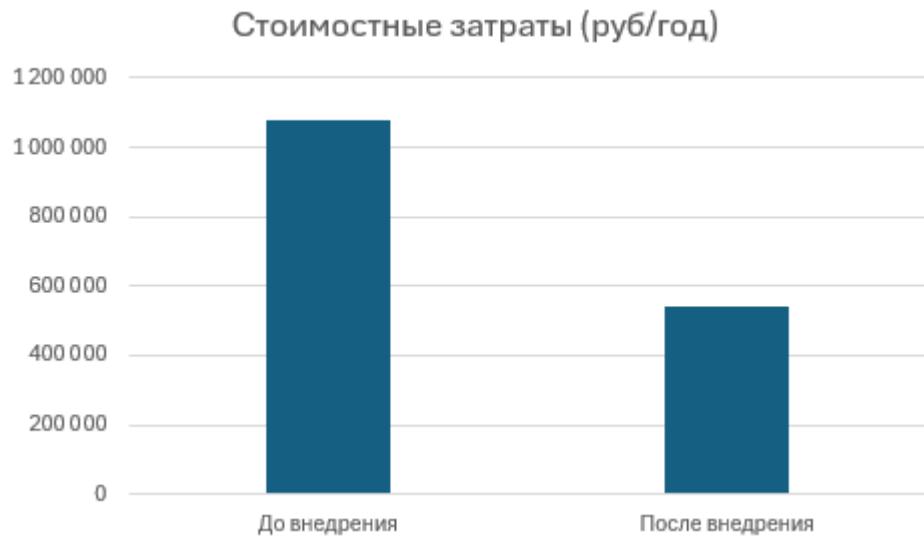


Рис.20 Стоимостные затраты

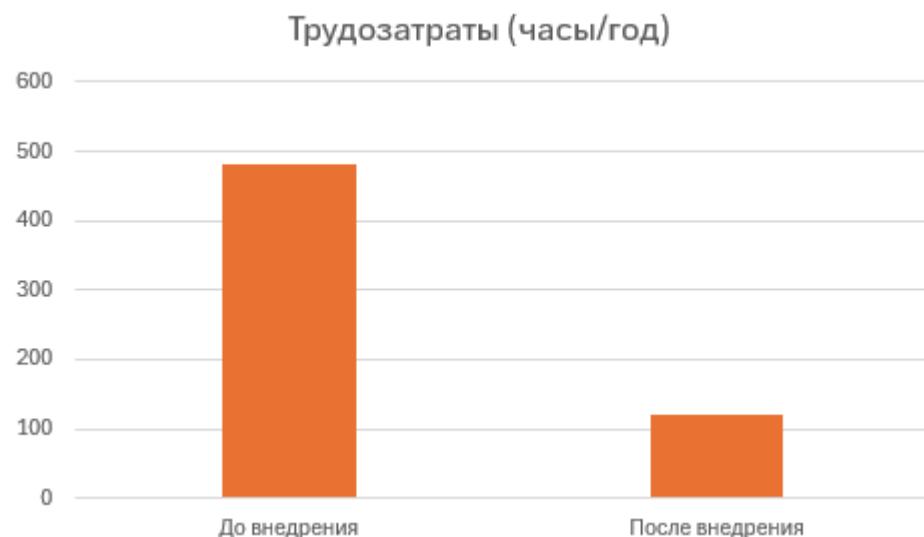


Рис.21 Трудозатраты

ЗАКЛЮЧЕНИЕ

В результате выполнения дипломного проекта была разработана и реализована информационная система учёта продаж игровых аккаунтов для ООО «Фанпей». В ходе работы был проведён анализ предметной области, определены основные требования к системе, спроектирована архитектура и база данных, реализовано программное обеспечение и проведена оценка экономической эффективности внедрения. Система обеспечивает автоматизацию ключевых процессов, способствует сокращению издержек, повышению скорости и надёжности работы предприятия.

СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

- 1) Программирование. Базовый курс С# / Авторы: Подбельский В. В. / Вид издания: Учебное пособие / Год издания: 2020. – 369 с.
- 2) Основы информационной безопасности: учебное пособие / Авторы: Будников Г. И., Петренко Ю. А. / Вид издания: Учебное пособие / Год издания: 2020. – 304 с.
- 3) Организация баз данных: учебное пособие / Авторы: Сергеев С. А., Степанов В. Г. / Вид издания: Учебное пособие / Год издания: 2018. – 352 с.
- 4) Проектирование информационных систем: учебник / Авторы: Гасанов Р. И., Акопова И. А. / Вид издания: Учебник / Год издания: 2017. – 328 с.
- 5) Информационные технологии в бизнесе: учебное пособие / Автор: Сидоров А. С. / Вид издания: Учебное пособие / Год издания: 2021. – 416 с.
- 6) Разработка информационных систем: учебник для вузов / Авторы: Шнейдер И. Г., Карлов А. А. / Вид издания: Учебник / Год издания: 2019. – 448 с.
- 7) Управление информационными ресурсами предприятия / Автор: Федоров Д. А. / Вид издания: Учебное пособие / Год издания: 2018. – 256 с.
- 8) Методы и средства защиты информации: учебник для вузов / Автор: Кузнецов А. И. / Вид издания: Учебник / Год издания: 2019. – 336 с.
- 9) Анализ и проектирование информационных систем: учебник для вузов / Авторы: Иванов А. П., Петров В. Г. / Вид издания: Учебник / Год издания: 2017. – 384 с.
- 10) Программирование на языке Java: учебник / Автор: Алексеев С. В. / Вид издания: Учебник / Год издания: 2020. – 432 с.
- 11) Основы информационных технологий: учебное пособие / Авторы: Новиков А. Г., Корнев В. М. / Вид издания: Учебное пособие / Год издания: 2018. – 320 с.
- 12) Программирование и анализ данных: учебное пособие / Автор: Рыбников А. Л. / Вид издания: Учебное пособие / Год издания: 2021. – 288 с.
- 13) Базы данных и знаний: учебное пособие / Автор: Петров И. И. / Вид издания: Учебное пособие / Год издания: 2019. – 352 с.
- 14) Программирование на языке С#: учебное пособие / Авторы: Смирнов В. В., Кузнецов А. А. / Вид издания: Учебное пособие / Год издания: 2020. – 368 с.

- 15) Информационные технологии в управлении предприятием: учебник для вузов / Автор: Горбунов А. С. / Вид издания: Учебник / Год издания: 2018. – 400 с.
- 16) Программирование на языке PHP: учебник / Автор: Попов С. А. / Вид издания: Учебник / Год издания: 2021. – 384 с.
- 17) Базы данных и системы управления базами данных: учебное пособие / Автор: Марков А. П. / Вид издания: Учебное пособие / Год издания: 2017. – 336 с.
- 18) Информационные технологии в экономике: учебник для вузов / Автор: Дмитриев А. В. / Вид издания: Учебник / Год издания: 2019. – 416 с.
- 19) Программирование на языке JavaScript: учебное пособие / Автор: Козлов А. С. / Вид издания: Учебное пособие / Год издания: 2018. – 352 с.
- 20) Методы и модели принятия решений: учебное пособие / Авторы: Иванов П. П., Сидоров Д. Г. / Вид издания: Учебное пособие / Год издания: 2020. – 384 с.
- 21) Сайт проекта FunPay – биржа игровых ценностей: <https://funpay.com/>

ПРИЛОЖЕНИЯ

Форма авторизации

```
using System;
using System.Drawing;
using System.Windows.Forms;
using MySql.Data.MySqlClient;

namespace FunPayApp
{
    public partial class Авторизация : Form
    {
        public Авторизация()
        {
            InitializeComponent();
            lblDateTime.Text = DateTime.Now.ToString();
        }

        private void close_Click(object sender, EventArgs e)
        {
```

```

        this.Close();
    }

private void lgbt_Click(object sender, EventArgs e)
{
    string login = lgtxb.Text.Trim();
    string password = pstxb.Text.Trim();

    if (string.IsNullOrWhiteSpace(login) ||
string.IsNullOrWhiteSpace(password))
    {
        MessageBox.Show("Введите логин и пароль!");
        return;
    }

    try
    {
        database db = new database();
        db.OpenConnection();

        string query = "SELECT COUNT(*) FROM users WHERE login =
@login AND password = @password";
        MySqlCommand command = new MySqlCommand(query,
db.GetConnection());
        command.Parameters.AddWithValue("@login", login);
        command.Parameters.AddWithValue("@password", password);

        int count = Convert.ToInt32(command.ExecuteScalar());
        db.CloseConnection();

        if (count > 0)
        {
            MessageBox.Show("Успешный вход!");
            this.Hide();
            AdminForm adminForm = new AdminForm();
            adminForm.Show();
        }
        else
        {
            MessageBox.Show("Неверный логин или пароль!");
        }
    }
    catch (Exception ex)
    {

```

```

        MessageBox.Show($"Ошибка при авторизации: {ex.Message}");
    }
}

private void pictureBox1_Click(object sender, EventArgs e)
{
    try
    {
        System.Diagnostics.Process.Start("https://funpay.com");
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка при открытии сайта: {ex.Message}");
    }
}

Point lastPoint;
private void Авторизация_MouseDown(object sender, MouseEventArgs e)
{
    lastPoint = new Point(e.X, e.Y);
}

private void Авторизация_MouseMove(object sender, MouseEventArgs e)
{
    if (e.Button == MouseButtons.Left)
    {
        this.Left += e.X - lastPoint.X;
        this.Top += e.Y - lastPoint.Y;
    }
}

private void lblDateTime_Click(object sender, EventArgs e) { }
private void LoginForm_Load(object sender, EventArgs e) { }
private void dateTimePicker1_ValueChanged(object sender, EventArgs e) { }
private void pictureBox1_MouseMove(object sender, MouseEventArgs e) { }
}
}

```

Форма регистрации

```

using System;
using System.Windows.Forms;
using MySql.Data.MySqlClient;

```

namespace FunPayApp

```

{
    public partial class RegisterForm : Form
    {
        public RegisterForm()
        {
            InitializeComponent();
        }

        private void RegisterForm_Load(object sender, EventArgs e)
        {

        }

        private void btnRegister_Click(object sender, EventArgs e)
        {
            string login = loginfield.Text.Trim();
            string password = passfield.Text.Trim();
            string role = "user";

            if (string.IsNullOrWhiteSpace(login) ||
                string.IsNullOrWhiteSpace(password))
            {
                MessageBox.Show("Введите логин и пароль!");
                return;
            }

            try
            {
                database db = new database();
                db.OpenConnection();

                MySqlCommand checkUser = new MySqlCommand("SELECT
COUNT(*) FROM users WHERE login = @login", db.GetConnection());
                checkUser.Parameters.AddWithValue("@login", login);

                if (Convert.ToInt32(checkUser.ExecuteScalar()) > 0)
                {
                    MessageBox.Show("Пользователь с таким логином уже
существует!");
                    db.CloseConnection();
                    return;
                }
            }
        }
    }
}

```

```
    MySqlCommand insertUser = new MySqlCommand("INSERT INTO
users (login, password, role) VALUES (@login, @password, @role)",
db.GetConnection());
    insertUser.Parameters.AddWithValue("@login", login);
    insertUser.Parameters.AddWithValue("@password", password);
    insertUser.Parameters.AddWithValue("@role", role);
    insertUser.ExecuteNonQuery();

    db.CloseConnection();

    MessageBox.Show("Регистрация успешна!");
    this.Hide();
    new Авторизация().Show();
}

catch (Exception ex)
{
    MessageBox.Show($"Ошибка регистрации: {ex.Message}");
}
}

private void close_Click_1(object sender, EventArgs e)
{
    this.Close();
}

private void pictureBox1_Click(object sender, EventArgs e)
{
    try
    {
        System.Diagnostics.Process.Start("https://funpay.com");
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка при открытии сайта: {ex.Message}");
    }
}
}
```

Форма администрирования

```
using System;  
using System.Data;  
using System.Windows.Forms;
```

```

using MySql.Data.MySqlClient;

namespace FunPayApp
{
    public partial class AdminForm : Form
    {
        private readonly database db = new database();
        private DataTable _usersTable;
        private DataTable _changesTable;

        public AdminForm()
        {
            InitializeComponent();
            InitializeChangesTable();
            LoadUsers();
        }

        private void InitializeChangesTable()
        {
            _changesTable = new DataTable();
            _changesTable.Columns.Add("id", typeof(int));
            _changesTable.Columns.Add("column_name", typeof(string));
            _changesTable.Columns.Add("new_value", typeof(object));
        }

        private void LoadUsers()
        {
            try
            {
                db.OpenConnection();
                var adapter = new MySqlDataAdapter("SELECT id, login, role, account
FROM users", db.GetConnection());
                _usersTable = new DataTable();
                adapter.Fill(_usersTable);
                dataGridViewUsers.DataSource = _usersTable;
            }
            catch (Exception ex)
            {
                MessageBox.Show($"Ошибка загрузки: {ex.Message}", "Ошибка",
MessageBoxButtons.OK, MessageBoxIcon.Error);
            }
            finally
            {
                db.CloseConnection();
            }
        }
    }
}

```

```

        }

    }

private void btnAddUser_Click(object sender, EventArgs e)
{
    if (string.IsNullOrWhiteSpace(txtLogin.Text) ||
    string.IsNullOrWhiteSpace(txtPassword.Text))
    {
        MessageBox.Show("Введите логин и пароль!", "Ошибка",
    MessageBoxButtons.OK, MessageBoxIcon.Warning);
        return;
    }

    try
    {
        db.OpenConnection();
        var connection = db.GetConnection();

        var checkCmd = new MySqlCommand("SELECT COUNT(*) FROM
users WHERE login = @login", connection);
        checkCmd.Parameters.AddWithValue("@login", txtLogin.Text);

        if (Convert.ToInt32(checkCmd.ExecuteScalar()) > 0)
        {
            MessageBox.Show("Пользователь уже существует!", "Ошибка",
    MessageBoxButtons.OK, MessageBoxIcon.Warning);
            return;
        }

        var insertCmd = new MySqlCommand("INSERT INTO users (login,
password, role, account) VALUES (@login, @password, @role, @account)",
connection);
        insertCmd.Parameters.AddWithValue("@login", txtLogin.Text);
        insertCmd.Parameters.AddWithValue("@password", txtPassword.Text);
        insertCmd.Parameters.AddWithValue("@role",
cmbRole.SelectedItem?.ToString() ?? "user");
        insertCmd.Parameters.AddWithValue("@account", txtAccount.Text);
        insertCmd.ExecuteNonQuery();

        MessageBox.Show("Пользователь добавлен!", "Успех",
MessageBoxButtons.OK, MessageBoxIcon.Information);
        txtLogin.Clear();
        txtPassword.Clear();
        txtAccount.Clear();
    }
}

```

```

        LoadUsers();
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка: {ex.Message}", "Ошибка",
MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    finally
    {
        db.CloseConnection();
    }
}

private void dataGridViewUsers_CellValueChanged(object sender,
DataGridViewCellEventArgs e)
{
    if (e.RowIndex < 0 || dataGridViewUsers.Rows.Count == 0) return;
    var row = dataGridViewUsers.Rows[e.RowIndex];
    var userId = row.Cells["id"].Value;
    var columnName = dataGridViewUsers.Columns[e.ColumnIndex].Name;
    var newValue = row.Cells[e.ColumnIndex].Value;
    _changesTable.Rows.Add(userId, columnName, newValue);
}

private void btnSaveChanges_Click(object sender, EventArgs e)
{
    if (_changesTable.Rows.Count == 0)
    {
        MessageBox.Show("Нет изменений для сохранения!",
"Информация", MessageBoxButtons.OK, MessageBoxIcon.Information);
        return;
    }

    try
    {
        db.OpenConnection();
        var connection = db.GetConnection();

        foreach (DataRow change in _changesTable.Rows)
        {
            var userId = change["id"];
            var columnName = change["column_name"].ToString();
            var newValue = change["new_value"];

```

```

        var query = $"UPDATE users SET {columnName} = @value
WHERE id = @id";
        using (var cmd = new MySqlCommand(query, connection))
{
    cmd.Parameters.AddWithValue("@value", newValue);
    cmd.Parameters.AddWithValue("@id", userId);
    cmd.ExecuteNonQuery();
}
MessageBox.Show("Изменения успешно сохранены!", "Успех",
MessageBoxButtons.OK, MessageBoxIcon.Information);
_changesTable.Clear();
LoadUsers();
}
catch (Exception ex)
{
    MessageBox.Show($"Ошибка сохранения: {ex.Message}", "Ошибка",
MessageBoxButtons.OK, MessageBoxIcon.Error);
}
finally
{
    db.CloseConnection();
}
}

private void btnRefresh_Click(object sender, EventArgs e)
{
    _changesTable.Clear();
    LoadUsers();
}

private void AdminForm_Load(object sender, EventArgs e) { }

private void dataGridViewUsers_CellContentClick(object sender,
DataGridViewCellEventArgs e) { }

private void close_Click(object sender, EventArgs e)
{
    this.Close();
}

private void pictureBox1_Click(object sender, EventArgs e)
{
    try

```

```
        {
            System.Diagnostics.Process.Start("https://funpay.com");
        }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка при открытии сайта: {ex.Message}");
    }
}
```

Форма смены пароля

```
using System;
using System.Windows.Forms;
using MySql.Data.MySqlClient;

namespace FunPayApp
{
    public partial class ChangePasswordForm : Form
    {
        private readonly int _userId;

        public ChangePasswordForm(int userId)
        {
            InitializeComponent();
            _userId = userId;
        }

        private void btnChangePassword_Click(object sender, EventArgs e)
        {
            string oldPassword = txtOldPassword.Text.Trim();
            string newPassword = txtNewPassword.Text.Trim();
            string confirmPassword = txtConfirmPassword.Text.Trim();

            if (string.IsNullOrWhiteSpace(oldPassword) ||
                string.IsNullOrWhiteSpace(newPassword) ||
                string.IsNullOrWhiteSpace(confirmPassword))
            {
                MessageBox.Show("Заполните все поля.");
                return;
            }
        }
    }
}
```

```

if (newPassword != confirmPassword)
{
    MessageBox.Show("Пароли не совпадают.");
    return;
}

try
{
    database db = new database();
    using (var conn = db.GetConnection())
    {
        conn.Open();

        var checkCmd = new MySqlCommand("SELECT COUNT(*) FROM
users WHERE id = @id AND password = @oldPassword", conn);
        checkCmd.Parameters.AddWithValue("@id", _userId);
        checkCmd.Parameters.AddWithValue("@oldPassword",
oldPassword);

        int match = Convert.ToInt32(checkCmd.ExecuteScalar());
        if (match == 0)
        {
            MessageBox.Show("Старый пароль введён неверно.");
            return;
        }

        var updateCmd = new MySqlCommand("UPDATE users SET
password = @newPassword WHERE id = @id", conn);
        updateCmd.Parameters.AddWithValue("@newPassword",
newPassword);
        updateCmd.Parameters.AddWithValue("@id", _userId);
        updateCmd.ExecuteNonQuery();

        MessageBox.Show("Пароль успешно изменён!");
        this.Close();
    }
}
catch (Exception ex)
{
    MessageBox.Show("Ошибка: " + ex.Message);
}
}

```

private void ChangePasswordForm_Load(object sender, EventArgs e)

```
    }

}

private void close_Click(object sender, EventArgs e)
{
    this.Close();
}

private void pictureBox1_Click(object sender, EventArgs e)
{
    try
    {
        System.Diagnostics.Process.Start("https://funpay.com");
    }
    catch (Exception ex)
    {
        MessageBox.Show($"Ошибка при открытии сайта: {ex.Message}");
    }
}
```

Класс базы данных

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using MySql.Data.MySqlClient;

namespace FunPayApp
{
    internal class database
    {
        MySqlConnection conn = new
MySqlConnection("server=localhost;database=funpaydb;user=root;password=");

        public void OpenConnection()
        {
            if (conn.State == System.Data.ConnectionState.Closed)
                conn.Open();
        }
    }
}
```

```
}

public void CloseConnection()
{
    if (conn.State == System.Data.ConnectionState.Open)
        conn.Close();
}

public MySqlConnection GetConnection()
{
    return conn;
}

}
```

Дипломный проект выполнен мной совершенно самостоятельно. Все использованные в проекте материалы и концепции из опубликованной литературы и других источников имеют ссылки на них. Дипломный проект прошел проверку на корректность заимствования в системе «Антиплагиат.ВУЗ».

Настоящим подтверждаю, что даю разрешение Университету «Синергия» на размещение полного текста моего дипломного проекта, отзыва на дипломный проект в электронно-библиотечной системе Университета «Синергия».



подпись

Хайдуров Н.Н

Фамилия ИО

«20» июня 2025