Министерство образования Республики Беларусь

Учреждение образования

«БелорусскиЙ государственный университет

информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра проектирования информационно-компьютерных систем

Дисциплина «Объектно-ориентированное программирование»

|  |
| --- |
| *К защите допустить*: |
| Руководитель курсовой работы  ассистент |
| \_\_\_\_\_\_\_\_\_\_\_\_А.П.Горбач  \_\_\_.\_\_\_. 20\_\_\_ |

**Пояснительная записка**

к курсовой работе

на тему

**ПРОГРАММНОЕ СРЕДСТВО С МИНИ ИГРАМИ "СИМУЛЯТОР КАЗИНО"**

БГУИР КР 1-40 05 01-10 014 ПЗ

|  |  |  |
| --- | --- | --- |
| Студент | (подпись студента) | Н.В.Верховодко |
|  |  | Курсовая работа представлена на проверку \_\_\_.\_\_\_. 20\_\_\_ |
|  |  | (подпись студента) |

Минск 2022

СОДЕРЖАНИЕ

[Введение 3](#_Toc115801855)

[**1.1.Анализ исходных данных к курсовому проекту.** 3](#_Toc115801856)

[**1.2 Обоснование и описание выбора языка программирования, средств разработки, используемых технологий и сторонних библиотек.** 4](#_Toc115801857)

[2 Перечень графического материала 6](#_Toc115801858)

[**2.1 Схема алгоритма.** 6](#_Toc115801859)

[3 Использование системы контроля версий GIT 6](#_Toc115801860)

# **Введение**

В современном мире нет необходимости посещать реальные игровые клубы Вегаса или же Макао для получения незабываемых эмоций. Компьютерные технологии позволили создать множество виртуальных площадок для любителей азартных игр.

Казино — программа, дающие возможность играть в азартные игры на компьютере. Казино, также называемые виртуальными казино, позволяют играть в разнообразные, основанные на случайности азартные игры.

Я считаю на данных момент эту тему как никогда актуальной, поскольку несколько лет назад виртуальное казино стало весьма популярным в Беларуси и не только. В реальных азартных клубах невозможно играть бесплатно, только посредством ставки реальных финансов. В виртуально казино можно играть как на реальные деньги, так и «на фантики», то есть не делая денежных депозитов, но без возможности выиграть/проиграть деньги.

Цель данной курсовой работы — это разработать казино, в котором пользователи смогут играть на виртуальные деньги. После установки на компьютере, можно ознакомиться с правилами и управлением. Даже неопытный пользователь сможет быстро освоить игру и увлекательно провести свое свободное время. Основными задачами являются создание интуитивно понятного пользовательского интерфейса, реализация нескольких атмосферных мини игр, которые заставят пользователей делать огромные ставки.

Солидные казино отличаются:

1. оригинальным, красочным дизайном;
2. удобным интерфейсом;
3. интересными звуковыми эффектами;
4. простым управлением и понятными правилами;
5. большим разнообразием слотов с захватывающим сюжетом.

Для достижения поставленной цели необходимо решить следующие задачи:

1. Изучение современных инструментов разработки приложений;
2. Разработка самой Windows Form;
3. Создание баз данных для хранения информации;
4. Создание сервиса для игры в мини игры;

## **1.1.Анализ исходных данных к курсовому проекту.**

Тема курсовой работы: программное средство с мини играми "симулятор казино"**.**

Язык программирования C#.

Среда разработки Visual Studio 2022.

Парадигма программирования – объектно-ориентированная.

Способ организации данных – классы.

Способ хранения данных – БД в MySQL.

К защите курсовой работы представляются:

1. программное средство,
2. схема алгоритма (формат А2/А3),
3. UML диаграмма классов (плакат, формат А2/А3),
4. диаграмма состояний (плакат, формат А2/А3),
5. структура графического пользовательского интерфейса (плакат, формат А2/А3) и пояснительная записка.

Текст пояснительной записки оформляется в соответствии со стандартом предприятия СТП 01–2017.

## **1.2 Обоснование и описание выбора языка программирования, средств разработки, используемых технологий и сторонних библиотек.**

Я решил выбрать создание приложения Windows Form потому что мне понравился язык C#.

Данный язык использует объектно-ориентированный подход к программированию во всем. Это означает, что тебе нужно будет описывать абстрактные конструкции на основе предметной области, а потом реализовывать между ними взаимодействие. Данный подход пользуется большой популярностью, потому что позволяет не держать в голове всю информацию, а работать по принципу черного ящика.

Также в языке вместо того, чтобы писать много строк кода, можно просто использовать готовую конструкцию, а компилятор сделает за тебя всю сложную работу. Но некоторые такие конструкции являются не самыми оптимальными с точки зрения производительности. Но все это перекрывается за счет удобочитаемости кода и высокой скоростью разработки. Так же к плюсам можно отнести строгую типизацию.

Еще стоит упомянуть, что все это работает на базе платформы .NET Framework. Так как окончательная компиляция из промежуточного кода выполняется в живую на вашей конкретной вычислительной машине, то возможно увеличение производительности за счет использования специфических команд именно вашего процессора.

Лично для меня одним из самых важных плюсов является наличие большого количества библиотек и шаблонов, позволяющих не тратить время на изобретение своего собственного велосипеда, из костылей. Можно нужное тебе решение из nuget и начинаешь его использовать. В большинстве своем они бесплатны. Сюда же можно отнести большое количество обучающего и справочного материала. Практически на любой свой вопрос ты сможешь найти ответ на формуах.

Немаловажно наличие хороших инструментов разработки, и здесь все очень хорошо. Абсолютное большинство разработчиков используют интегрированную среду разработки Visual Studio, которая предоставляет много возможностей, использовать которые ты конечно же не будешь.

Windows Forms — это интеллектуальная клиентская технология для .NET Framework, набора управляемых библиотек, которые упрощают общие задачи приложений, такие как чтение и запись в файловую систему. При использовании среды разработки, такой как Visual Studio, вы можете создавать смарт-клиентские приложения Windows Forms, которые отображают информацию, запрашивают ввод данных от пользователей и обмениваются данными с удаленными компьютерами по сети.

В Windows Forms форма — это визуальная поверхность, на которой вы отображаете информацию для пользователя. Обычно вы создаете приложения Windows Forms, добавляя элементы управления в формы и разрабатывая ответы на действия пользователя, такие как щелчки мышью или нажатия клавиш. Элемент управления — это дискретный элемент пользовательского интерфейса (UI), который отображает данные или принимает ввод данных.

Windows Forms содержит различные элементы управления, которые можно добавлять в формы: элементы управления, отображающие текстовые поля, кнопки, раскрывающиеся списки, переключатели и даже веб-страницы. Список всех элементов управления, которые можно использовать в форме, см. в разделе Элементы управления для использования в Windows Forms . Если существующий элемент управления не соответствует вашим потребностям, Windows Forms также поддерживает создание собственных настраиваемых элементов управления с помощью класса UserControl.

Windows Forms имеет богатые элементы управления пользовательского интерфейса, которые эмулируют функции высокопроизводительных приложений, таких как Microsoft Office. При использовании элементов управления ToolStrip и MenuStrip можно создавать панели инструментов и меню, содержащие текст и изображения, отображать подменю и размещать другие элементы управления, такие как текстовые поля и поля со списком.

# **2 Перечень графического материала**

## **2.1 Схема алгоритма.**



Рисунок 2.1 – Схема алгоритма

# **3 Использование системы контроля версий GIT**

Система управления версиями (Version Control System) — программное обеспечение для облегчения работы с изменяющейся информацией. Система управления версиями позволяет хранить несколько версий одного и того же документа, при необходимости возвращаться к более ранним версиям, определять, кто и когда сделал то или иное изменение, и многое другое.

Такие системы наиболее широко используются при разработке программного обеспечения для хранения исходных кодов разрабатываемой программы. Однако они могут с успехом применяться и в других областях, в которых ведётся работа с большим количеством непрерывно изменяющихся электронных документов.

Git — распределённая система управления версиями. Система спроектирована как набор программ, специально разработанных с учётом их использования в сценариях. Это позволяет удобно создавать специализированные системы контроля версий на базе Git или пользовательские интерфейсы. Удалённый доступ к репозиториям Git обеспечивается git-демоном, SSH- или HTTP-сервером.

GitHub — крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Веб-сервис основан на системе контроля версий Git. Сервис бесплатен для проектов с открытым исходным кодом и небольших частных проектов, предоставляя им все возможности (включая SSL), а для крупных корпоративных проектов предлагаются различные платные тарифные планы.

Создатели сайта называют GitHub «социальной сетью для разработчиков». Кроме размещения кода, участники могут общаться, комментировать правки друг друга, а также следить за новостями знакомых.

С помощью широких возможностей Git программисты могут объединять свои репозитории — GitHub предлагает удобный интерфейс для этого и может отображать вклад каждого участника в виде дерева. Для проектов есть личные страницы, небольшие Вики и система отслеживания ошибок. Прямо на сайте можно просмотреть файлы проектов с подсветкой синтаксиса для большинства языков программирования. Можно создавать приватные репозитории, которые будут видны только вам и выбранным вами людям.

Ссылка на репозиторий с проектом: <https://github.com/NVERKHOVODKO/OOP-CourseWork>